METODOS NUMERICOS APLICADOS CON SOFTWARE

SHOICHIRO NAKAMURA

PEARSON
Prentice Hall ®

METODOS NUMERICOS APLICADOS CON SOFTWARE

METODOS NUMERICOS APLICADOS CON SOFTWARE

PRIMERA EDICION

Shoichiro Nakamura
The Ohio State University

Traducción:
Oscar Alfredo Palmas Velasco
Maestría Facultad de Ciencias, UNAM

Revisión Técnica:
Victor Hugo Ibarra Mercado
Lic. en Matemáticas ESFM-IPN

PEARSON Educación ®

México • Argentina • Brasil • Colombia • Costa Rica • Chile • Ecuador
España • Guatemala • Panamá • Perú • Puerto Rico • Uruguay • Venezuela

EDICION EN INGLES
Editorial/production supervision and
interior design: Fred Dahl
Cover design: Ben Santora
Manufacturing buyer: Lori Bulwin

METODOS NUMERICOS APLICADOS CON SOFTWARE, 1/E

Traducido de la primera edición en inglés de:

APPLIED NUMERICAL METHODS WITH SOFTWARE

Prohibida la reproducción total o parcial de esta obra, por cualquier medio o método sin autorización escrita del editor.

DERECHOS RESERVADOS © 1992 respecto a la primera edición en español por
PRENTICE-HALL HISPANOAMERICANA, S.A.

Atlacomulco Núm. 500-5° Piso
Col. Industrial Atoto
53519, Naucalpan de Juárez, Edo. de México

ISBN 968-880-263-8

Miembro de la Cámara Nacional de la Industrial
Editorial, Reg. Núm. 1524

Original English Language Edition Published by
Copyright MCMXCI Prentice Hall Inc.
All Rights Reserved
ISBN 0-13-041047-0

IMPRESO EN MEXICO / PRINTED IN MEXICO

Contenido

Programas, *vii*

Prefacio, *ix*

Antes de leer y usar los programas de este libro, *xiii*

1 Causas principales de errores en los métodos numéricos, *1*

1.1 Introducción, 1
1.2 Series de Taylor, 1
1.3 Números en las computadoras, 5

2 Interpolación polinomial, *22*

2.1 Introducción, 22
2.2 Interpolación lineal, 22
2.3 Fórmula de interpolación de Lagrange, 24
2.4 Interpolaciones de Newton hacia adelante y hacia atrás en puntos con igual separación, 32
2.5 Interpolación de Newton en puntos con separación no uniforme, 40
2.6 Interpolación con raíces de Chebyshev, 43
2.7 Polinomios de interpolación de Hermite, 47
2.8 Interpolación en dos dimensiones, 50
2.9 Extrapolaciones, 51

3 Solución de ecuaciones no lineales, *62*

3.1 Introducción, 62
3.2 Método de bisección, 63

3.3 Método de la falsa posición y método de la falsa posición modificada, 68
3.4 Método de Newton, 73
3.5 Método de la secante, 77
3.6 Método de sustitución sucesiva, 79
3.7 Método de Bairstow, 82

4 Integración numérica, *109*

4.1 Introducción, 109
4.2 Regla del trapecio, 110
4.3 Regla de 1/3 de Simpson, 115
4.4 Regla de 3/8 de Simpson, 119
4.5 Fórmulas de Newton-Cotes, 120
4.6 Cuadraturas de Gauss, 123
4.7 Integración numérica con límites infinitos o singularidades, 130
4.8 Integración numérica en un dominio bidimensional, 135

5 Diferenciación numérica, *155*

5.1 Introducción, 155
5.2 Uso del desarrollo de Taylor, 156
5.3 Algoritmo genérico para obtener una aproximación por diferencias, 163
5.4 Uso de los operadores de diferencias, 166
5.5 Uso de la diferenciación de los polinomios de interpolación de Newton, 168
5.6 Aproximación de derivadas parciales por diferencias, 171

6 Algebra lineal numérica, *184*

6.1 Introducción, 184
6.2 Eliminaciones de Gauss y Gauss-Jordan para problemas ideales sencillos, 185
6.3 Pivoteo y eliminación canónica de Gauss, 191
6.4 Problemas sin solución única, 195
6.5 Matrices y vectores, 196
6.6 Inversión de una matriz, 203

6.7 Descomposición LU, 207
6.8 Determinantes, 212
6.9 Problemas mal condicionados, 216
6.10 Solución de N ecuaciones con M incógnitas, 218

7 Cálculo de valores propios de una matriz, *238*

7.1 Introducción, 238
7.2 Método de interpolación, 243
7.3 Método de Householder para una matriz simétrica, 246
7.4 Métodos de potencias, 250
7.5 Iteración QR, 253

8 Ajuste de curvas, *274*

8.1 Introducción, 274
8.2 Regresión lineal, 274
8.3 Ajuste de curvas con un polinomio de orden superior, 278
8.4 Ajuste de curvas mediante una combinación lineal de funciones conocidas, 280

9 Problemas de ecuaciones diferenciales ordinarias con valor o condición inicial, *289*

9.1 Introducción, 289
9.2 Métodos de Euler, 292
9.3 Métodos de Runge-Kutta, 299
9.4 Métodos predictor-corrector, 312
9.5 Más aplicaciones, 321
9.6 EDO rígidas, 329

10 Problemas de ecuaciones diferenciales con valores en la frontera, *351*

10.1 Introducción, 351
10.2 Problemas con valores en la frontera para varillas y láminas, 353
10.3 Algoritmo de solución por medio de sistemas tridiagonales, 358

10.4 Coeficientes variables y retícula con espaciamiento no uniforme en la geometría laminar, 360
10.5 Problemas con valores en la frontera para cilindros y esferas, 364
10.6 Problemas de ecuaciones diferenciales ordinarias no lineales con valores en la frontera, 366
10.7 Problemas de valores propios en ecuaciones diferenciales ordinarias, 368
10.8 Análisis de convergencia de los métodos iterativos, 375
10.9 Doblamiento y vibración de una viga, 379

11 Ecuaciones diferenciales parciales elípticas, *407*

11.1 Introducción, 407
11.2 Ecuaciones en diferencias, 409
11.3 Panorama de los métodos de solución para las ecuaciones en diferencias elípticas, 426
11.4 Métodos de relajación sucesiva, 427
11.5 Análisis de convergencia, 433
11.6 Cómo optimizar los parámetros de iteración, 442
11.7 Método implícito de la dirección alternante (IDA), 447
11.8 Métodos de solución directa, 450

12 Ecuaciones diferenciales parciales parabólicas, *470*

12.1 Introducción, 470
12.2 Ecuaciones en diferencias, 471
12.3 Análisis de estabilidad, 478
12.4 Métodos numéricos para problemas parabólicos bidimensionales, 484

13 Ecuaciones diferenciales hiperbólicas, *489*

13.1 Introducción, 489
13.2 Método de características, 491
13.3 Métodos de diferencias (exactas) de primer orden, 495
13.4 Análisis del error por truncamiento, 501
13.5 Esquemas de orden superior, 504

13.6 Esquemas de diferencias en la forma conservativa, 508
13.7 Comparación de métodos mediante ondas de pruebas, 512
13.8 Esquemas numéricos para EDP hiperbólicas no lineales, 512
13.9 Esquemas de flujo corregido, 516

Apéndices

A **Error de las interpolaciones polinomiales,** *524*

B **Polinomios de Legendre,** *529*

C **Cálculo de diferencias de orden superior con el operador de traslación,** *531*

D **Obtención de EDP hiperbólicas unidimensionales para problemas de flujo,** *533*

E **Disminución de la variación total (TVD),** *535*

F **Obtención de las ecuaciones modificadas,** *537*

G **Interpolación con splines cúbicos,** *540*

H **Interpolación transfinita bidimensional,** *549*

Indice, *565*

Programas

PROGRAMA 1-1 Conversión de decimal a binario, 18

PROGRAMA 2-1 Interpolación de Lagrange, 52
PROGRAMA 2-2 Tabla de diferencias hacia adelante, 53
PROGRAMA 2-3 Tabla de diferencias divididas, 54

PROGRAMA 3-1 Método de bisección, 87
PROGRAMA 3-2 Búsqueda de raíces, 90
PROGRAMA 3-3 Graficación de una función en BASIC, 92
PROGRAMA 3-4 Método de la falsa posición modificada, 94
PROGRAMA 3-5 Método de Newton, 96
PROGRAMA 3-6 Método de Newton para raíces complejas, 98
PROGRAMA 3-7 Método de Bairstow, 100

PROGRAMA 4-1 Reglas extendidas del trapecio y de Simpson, 139
PROGRAMA 4-2 Fórmulas cerradas de Newton-Cotes, 141
PROGRAMA 4-3 Fórmulas abiertas de Newton-Cotes, 143
PROGRAMA 4-4 Cuadratura de Gauss, 143
PROGRAMA 4-5 Integración de una función singular, 146
PROGRAMA 4-6 Integración doble, 148

PROGRAMA 5-1 Cálculo de aproximaciones por diferencias, 173

PROGRAMA 6-1 Eliminación de Gauss, 223
PROGRAMA 6-2 Inversión de una matriz, 225
PROGRAMA 6-3 Descomposición LU, 227
PROGRAMA 6-4 M Ecuaciones con N incógnitas, 230

PROGRAMA 7-1 Método de interpolación, 258
PROGRAMA 7-2 Householder/Bisección, 261
PROGRAMA 7-3 Iteración *QR*, 265

PROGRAMA 8-1 Curvas por mínimos cuadrados, 282

PROGRAMA 9-1 Método de Runge-Kutta de segundo orden, 334

PROGRAMA 9-2 Esquema de Runge-Kutta de cuarto orden, 336
PROGRAMA 9-3 Método de Runge-Kutta de cuarto orden para un conjunto de EDO, 338
PROGRAMA 9-4 Método predictor-corrector de tercer orden, 340

PROGRAMA 10-1 Solución de problemas lineales con valores en la frontera, 382
PROGRAMA 10-2 Solución de problemas no lineales con valores en la frontera, 384
PROGRAMA 10-3 Método de la potencia inversa, 386
PROGRAMA 10-4 Método de la potencia inversa con desplazamiento, 389
PROGRAMA 10-5 Doblamiento de una viga, 392
PROGRAMA 10-6 Vibración de una viga, 395

PROGRAMA 11-1 SOR, 453

PROGRAMA 11-2 Método iterativo extrapolado de Jacobi (EJ), 456
PROGRAMA 11-3 Optimización del parámetro del SOR, 457
PROGRAMA 11-4 Demostración del theta óptimo, 459

PROGRAMA 12-1 Ecuación de la conducción del calor, 486

Prefacio

Este libro describe los métodos numéricos aplicados que aprenden los estudiantes de ingeniería y de ciencias pertenecientes a una amplia gama que abarca desde el segundo año de la licenciatura hasta el primero del posgrado. Los primeros nueve capítulos se basan en el material que he enseñado en dos cursos introductorios de métodos numéricos. Los últimos cuatro se apoyan en el material enseñado a nivel de posgrado, aunque las primeras secciones de los últimos cuatro capítulos se han escrito de manera que resultan comprensibles a los estudiantes de licenciatura de los niveles superiores.

La importancia de los métodos numéricos ha aumentado de forma drástica en la enseñanza de la ingeniería y la ciencia, lo cual refleja el uso actual y sin precedentes de las computadoras. Al aprender los métodos numéricos, nos volvemos aptos para: 1) entender esquemas numéricos a fin de resolver problemas matemáticos, de ingeniería y científicos en una computadora; 2) deducir esquemas numéricos básicos; 3) escribir programas y resolverlos en una computadora, y 4) usar correctamente el software existente para dichos métodos. El aprendizaje de los métodos numéricos no sólo aumenta nuestra habilidad para el uso de computadoras, también amplía la pericia matemática y la comprensión de los principios científicos básicos. En mi opinión, los métodos numéricos son más benéficos si su enseñanza comienza en el segundo año de la licenciatura.

Entre los objetivos de este libro están: 1) que sea fácilmente comprensible para los estudiantes de licenciatura con un conocimiento mínimo de matemáticas; 2) capacitar a los estudiantes para que practiquen los métodos en una microcomputadora; 3) proporcionar programas cortos que puedan usarse de manera sencilla en aplicaciones científicas con o sin modificaciones, y 4) proporcionar software que resulte fácil de comprender.

El capítulo 1 analiza los errores de truncamiento y redondeo, los cuales son temas preparatorios para el cálculo numérico. Con el objeto de explicar las causas de estos errores, se examinan brevemente las series de Taylor y cómo se calculan y almacenan los números en las computadoras.

El capítulo 2 describe las interpolaciones de Lagrange y Newton como temas básicos. A continuación, se estudia la interpolación usando los puntos de Chebyshev, la interpolación de Hermite, la interpolación bidimensional y se da un breve

análisis de la extrapolación. Las causas y el comportamiento de los errores de interpolación se explican mediante un enfoque intuitivo.

El capítulo 3 describe los métodos iterativos que se usan para resolver ecuaciones no lineales, incluyendo los métodos de bisección, de la falsa posición, de Newton, de la secante, de iteración de punto fijo y finalmente el método de Bairstow.

El capítulo 4 abarca los métodos de integración numérica, comenzando con los métodos de integración simples (pero fundamentales), como la regla del trapecio y la de Simpson. A continuación, se presentan las fórmulas de Newton-Cotes. Además, el capítulo 4 incluye el método de integración de Gauss y el método numérico para integrarles impropias y dobles. Los métodos numéricos para las integrales impropias se basan en la regla del trapecio y en la transformación exponencial doble.

El capítulo 5 analiza los conceptos básicos de la diferenciación numérica. Desarrolla un método sistemático para obtener aproximaciones por diferencias con el término de error por truncamiento. El enfoque se implanta en un programa de computadora.

El capítulo 6 examina los métodos computacionales básicos para resolver las ecuaciones lineales no homogéneas. Primero se estudian los métodos de eliminación de Gauss y de Gauss-Jordan sin pivoteo y después con pivoteo. El efecto del pivoteo se ilustra tanto con precisión simple como en precisión doble. Después de presentar la notación matricial, se dan los conceptos de inversión de matrices, de descomposición LU y de determinante. Se explican los problemas mal condicionados usando la matriz inversa y el determinante. Por último se describe la solución de n ecuaciones m incógnitas.

El capítulo 7 abarca los métodos selectos para el cálculo de valores propios de una matriz. Primero se explican los aspectos básicos de las ecuaciones lineales homogéneas. Después se proporcionan el método de interpolación. Este enfoque deberá ayudar a los estudiantes a comprender la relación entre los valores propios y las raíces de la ecuación característica. En el resto del capítulo se dan los métodos iterativos y tridiagonal de Householder, así como la iteración QR.

El capítulo 8 describe el ajuste de curvas de datos experimentales con base en los métodos de mínimos cuadrados.

El capítulo 9 analiza los métodos numéricos de las ecuaciones diferenciales ordinarias, incluyendo los métodos de Runge-Kutta y el predictor-corrector. Se ilustrarán las aplicaciones de los métodos a numerosos problemas de ingeniería.

El capítulo 10 describe los métodos numéricos para problemas de ecuaciones diferenciales con valores en la frontera, incluyendo los problemas de valores propios. Este capítulo también pueden servir como preparación para los métodos numéricos en ecuaciones diferenciales parciales que se estudiarán a continuación.

Los últimos tres capítulos examinan los métodos numéricos para las ecuaciones diferenciales parciales. El capítulo 11 analiza las ecuaciones diferenciales parciales elípticas. En sus primeras secciones se estudia la obtención de ecuaciones en diferencias y la implantación de condiciones en la frontera. Después describe los métodos iterativos, incluyendo la sobrerrelajación sucesiva, el método iterativo extra-

polado de Jacobi basado en la propiedad bicíclica y el método ADI. Se presentan los análisis de convergencia y la optimización de parámetros iterativos usando un modelo unidimensional. En el capítulo también se da el concepto de métodos de solución directa.

El capítulo 12 trata de las ecuaciones diferenciales parciales parabólicas. Incluye los métodos numéricos basados en los métodos explícito, implícito y de Euler modificado. Se brindan los conceptos de análisis de estabilidad basados en funciones propias, al igual que el desarrollo de Fourier.

El capítulo 13 analiza las ecuaciones diferenciales parciales hiperbólicas. Al restringirse a una sencilla ecuación de onda de primer orden, comienza con los métodos de características, a los que siguen los métodos fundamentales de diferencia finita. Se describen los análisis de errores por truncamiento con el concepto de ecuación modificada. También se desarrolla una fórmula simplificada para obtener ecuaciones modificadas.

El libro contiene aproximadamente 40 programas, la mayor parte de los cuales se listan en FORTRAN y unos pocos en BASIC. Las versiones en FORTRAN se pueden ejecutar en una microcomputadora con un compilador o intérprete FORTRAN apropiado; también se pueden correr en una supercomputadora (*mainframe*). Los programas se pueden transferir a ese tipo de computadora por medio de un software de comunicación y un módem. Los programas se pueden modificar con facilidad, de tal forma que los estudiantes puedan correr en forma inmediata tanto los programas existentes como los que acaban de modificar. Los métodos numéricos descritos en este libro no se limitan a las microcomputadoras; son esencialmente independientes del tipo de computadoras que se usen.

El gran número de programas de este libro no significa que se deja de enfatizar la importancia de la práctica de programación por parte de los estudiantes. El desarrollo de un programa siempre es importante en el aprendizaje de métodos numéricos. El instructor debe asignar programas cortos para que los estudiantes los desarrollen. Por otro lado, el desarrollo de cada programa a partir de la nada consume tiempo y a menudo es ineficaz y frustrante. Además, es imposible dar todas las instrucciones necesarias para la programación, las protecciones aritméticas, el formateo y la prueba. Un enfoque eficaz es el de asignar frecuentemente a los estudiantes proyectos para modificar partes de los programas de este libro.

Quisiera agradecer a muchos estudiantes de licenciatura que usaron mi primer manuscrito como texto para los cursos de métodos numéricos en el departamento de ingeniería mecánica y en el de ciencias de la computación e información de la Ohio State University. Las preguntas planteadas por los estudiantes me ayudaron a reexaminar y mejorar el manuscrito. Muchos estudiantes de posgrado, en realidad demasiados para nombrarlos de manera individual, me ayudaron en la corrección de estilo, la verificación del software y el desarrollo de las claves para las respuestas. Sin la ayuda de ellos, este libro no habría existido. El estímulo y ayuda que me dio el profesor Helmer Nielsen —jefe del departamento de ingeniería mecánica de San Jose State University— en la primera etapa de escritura, fueron invaluables. Las críticas y sugerencias de muchos revisores, en particular los del profesor Mike Khonsari de la Universidad de Pittsburgh, el profesor Robert Skeel de la Universidad de

Illinois, y las del profesor Terry Shoup de la Florida Atlantic University fueron muy instructivas para mí. Estoy en deuda con los profesores Henry Busby, Terry Conlisk, Yann Guezennec, y M.J. Liou de la Ohio State University, quienes enseñaron métodos numéricos usando mis primeros manuscritos como texto. La terminación de este libro hubiera sido imposible sin el estímulo del profesor Larry Kennedy, jefe del departamento de ingeniería mecánica de la Ohio State University. Para terminar, quisiera agradecer a mi familia, la cual ha sido extremadamente paciente durante mi trabajo en este libro.

S. NAKAMURA
Columbus, Ohio

Antes de leer y usar los programas de este libro

La mayor parte de los programas en este libro se escribieron en FORTRAN, para una computadora VAX con unos pocos escritos en BASIC. Todos los programas se desarrollaron originalmente en BASIC y después se tradujeron a FORTRAN. En realidad, la mayoría de los ejemplos de cálculo se desarrollaron usando las versiones de BASIC para una IBM PC. El lector debe estar consciente de que los resultados de un programa a veces difieren si se ejecutan en computadoras distintas. Aunque dichas discrepancias suelen ser muy pequeñas, quizá sean significativas si el cálculo es sensible a los errores de redondeo.

En las explicaciones de los programas que están al final de cada capítulo, se usan dos símbolos:

- S-33 indica la línea de la instrucción número 33 en los programas de FORTRAN.
- L-33 se usa para indicar el número de línea 33 en los programas de BASIC.

Los programas en FORTRAN incluyen numerosos comentarios después del signo "!", los cuales están escritos en las instrucciones ejecutables. Estos comentarios son para conveniencia de los lectores del programa, pero por desgracia se deben eliminar los comentarios con dicho signo (excepto cuando los programas se corren en una VAX).

1

Causas principales de errores en los métodos numéricos

1.1 INTRODUCCION

Existen dos causas principales de errores en los cálculos numéricos. La primera es el error de truncamiento y la segunda es el error de redondeo. El error de truncamiento se debe a las aproximaciones utilizadas en la fórmula matemática del modelo. La serie de Taylor es el medio más importante que se emplea para obtener modelos numéricos y analizar los errores de truncamiento.

Los errores de redondéo se asocian con el número limitado de dígitos con que se representan los números en una computadora. Para comprender la naturaleza de estos errores, es necesario aprender las formas en que se almacenan los números y cómo se llevan a cabo las sumas y restas dentro de una computadora.

Este capítulo analiza las series de Taylor y los números; ambos con temas fundamentales en métodos numéricos.

1.2 SERIES DE TAYLOR

Las soluciones numéricas son, en su mayoría, aproximaciones de las soluciones exactas. Gran parte de los métodos numéricos se basan en la aproximación de funciones por medio de polinomios, aún cuando esto no sea evidente. Se construyen algoritmos más avanzados conjuntando los algoritmos básicos. Por lo tanto, cuando se objeta el error de un método numérico, hay que investigar la precisión con la que el polinomio aproxima a la función verdadera.

El desarrollo de Taylor, que es una serie infinita de potencias, representa de manera exacta a una función dentro de un cierto radio alrededor de un punto dado.

Por lo tanto, mediante la comparación del desarrollo polinomial de la solución numérica con la serie de Taylor de la solución exacta —particularmente al descubrir el orden en donde aparece la discrepancia— es posible evaluar el error, el cual se conoce como *error de truncamiento* [Conte/de Boor; King; Hornbeck].

También se usa la serie de Taylor para obtener métodos numéricos. Si se ignoran todos los términos de la serie de Taylor, excepto unos pocos, se puede obtener un polinomio que se aproxime a la función verdadera. A este polinomio se le llama una *serie de Taylor truncada* y se usa como punto de partida para obtener métodos numéricos [Morris; Cheney/Kincaid]. Sin embargo, el error del método numérico se origina en el truncamiento.

Desarrollo de Taylor para funciones unidimensionales. Se dice que una función $f(x)$ es analítica en $x = a$ si $f(x)$ se puede representar por medio de una serie de potencias en términos de $h = x - a$ dentro de un radio de convergencia, $D > |x - a| > 0$. Una condición necesaria para que una función sea analítica es que todas sus derivadas sean continuas tanto en $x = a$, como en alguna vecindad alrededor de ese punto.

Un punto en donde una función $f(x)$ no es analítica recibe el nombre de *punto singular*. Si $f(x)$ es diferenciable en todas partes en la vecindad de x_0 excepto en x_0, entonces x_0 es un punto singular. Por ejemplo, tan (x) es analítica excepto en $x = \pm (n + \frac{1}{2})\ \pi$, $n = 0, 1, 2, \ldots, \infty$, los cuales son puntos singulares. Los polinomios son analíticos en todas partes.

Si f es analítica alrededor de $x = a$, se puede representar $f(x)$ de manera exacta en la vecindad de $x = a$ por medio de su serie de Taylor, que es una serie de potencias dada por

$$f(x) = f(a) + hf'(a) + \frac{h^2}{2} f''(a) + \frac{h^3}{6} f'''(a) + \frac{h^4}{24} f''''(a) + \frac{h^5}{5!} f'''''(a) + \cdots + \frac{h^m}{m!} f^{(m)}(a) + \cdots \tag{1.2.1}$$

donde

$h = x - a$

Por ejemplo, los desarrollos de Taylor de e^{-x} sen (x), alrededor de $x = 1$ son, respectivamente,

$$\begin{aligned} \exp(-x) &= \exp(-1) - h \exp(-1) + \frac{h^2}{2} \exp(-1) - \frac{h^3}{6} \exp(-1) + \frac{h^4}{24} \exp(-1) - \cdots \\ \text{sen}\,(x) &= \text{sen}\,(1) + h \cos(1) - \frac{h^2}{2} \text{sen}\,(1) \end{aligned} \tag{1.2.2}$$

$$-\frac{h^3}{6}\cos(1)+\frac{h^4}{24}\operatorname{sen}(1)+\cdots$$

donde

$h = x - 1.$

La serie de Taylor es única. Esto quiere decir que no existe otra serie de potencias en $h = x - a$ para representar $f(x)$.

El desarrollo de Taylor de una función alrededor de $x = 0$ recibe el nombre de *serie de Maclaurin*. Por ejemplo, las series de Maclaurin de las funciones exp (x), sen (x), cos (x), y ln $(x + 1)$ son, respectivamente,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

$$\operatorname{sen}(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots$$

$$\ln(x+1) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots$$

En las aplicaciones prácticas, hay que truncar la serie de Taylor después de un término de cierto orden, ya que es imposible incluir un número infinito de términos. Si la serie de Taylor se trunca después del término de orden N, se expresa como

$$f(x) = f(a) + hf'(a) + \frac{h^2}{2}f''(a) + \frac{h^3}{6}f'''(a) + \frac{h^4}{24}f''''(a) + \frac{h^5}{5!}f'''''(a) + \cdots + \frac{h^N}{N!}f^{(N)}(a) + 0(h^{N+1}) \tag{1.2.3}$$

donde $h = x - a$ y $0(h^{N+1})$ representa el error provocado por el truncamiento de los términos de orden $N + 1$ y superiores. Sin embargo, el error global se puede expresar como

$$0(h^{N+1}) = f^{(N+1)}(a + \xi h)\frac{h^{N+1}}{(N+1)!}, \quad 0 \leqslant \xi \leqslant 1 \tag{1.2.4}$$

puesto que ξ no se puede calcular con exactitud, es frecuente aproximar el término del error haciendo $\xi = 0$.

$$0(h^{N+1}) \simeq f^{(N+1)}(a)\frac{h^{N+1}}{(N+1)!} \tag{1.2.5}$$

que es el término dominante de los términos truncados.

Si $N = 1$, por ejemplo, la serie de Taylor truncada es

$$f(x) \simeq f(a) + f'(a)h, \quad h = x - a \tag{1.2.6}$$

Si se incluye el efecto del error, también se expresa como

$$f(x) = f(a) + f'(a)h + 0(h^2) \tag{1.2.7}$$

donde

$$0(h^2) = f''(a + \xi h)\frac{h^2}{2}, \quad 0 < \xi < 1 \tag{1.2.8}$$

Serie de Taylor para una función bidimensional. El desarrollo de Taylor de una función de dos dimensiones $f(x, y)$ alrededor de (a, b) está dada por

$$\begin{aligned} f(x, y) = f(a, b) &+ hf_x + gf_y + \frac{1}{2}[h^2 f_{xx} + 2hgf_{xy} + g^2 f_{yy}] \\ &+ \frac{1}{6}[h^3 f_{xxx} + 3h^2 g f_{xxy} + 3hg^2 f_{xyy} + g^3 f_{yyy}] \\ &+ \frac{1}{24}[h^4 f_{xxxx} + 4h^3 g f_{xxxy} + 6h^2 g^2 f_{xxyy} + 4hg^3 f_{xyyy} + g^4 f_{yyyy}] + \cdots \end{aligned} \tag{1.2.9}$$

donde

$h = x - a, \quad g = y - b,$

$f_x = \dfrac{\partial}{\partial x} f(x, y)|_{x=a, y=b}$

$f_y = \dfrac{\partial}{\partial y} f(x, y)|_{x=a, y=b}$

y las notaciones análogas tales como $f_{x\ldots x}$, $f_{xy\ldots}$, y $f_{yy\ldots}$ son las derivadas parciales de f en $x = a$ y $y = b$; cada x o y en los subíndices indica una diferenciación parcial con respecto de x o y, respectivamente.

Resumen de Esta Sección

a) Las series de Taylor son la herramienta más importante para obtener métodos numéricos y para analizar errores.
b) La serie de Taylor alrededor de $x = 0$ recibe el nombre de serie de Maclaurin.

1.3 NUMEROS EN LAS COMPUTADORAS

Al resolver un problema matemático por medio de una calculadora de bolsillo, estamos conscientes de que los números decimales que calculamos quizá no sean exactos. Estos números casi siempre se redondean cuando los registramos. Aun cuando los números no se redondeen de manera intencional, el número limitado de dígitos de la calculadora puede provocar errores de redondeo. (Una calculadora de bolsillo diseñada para cálculos científicos puede tener 10 u 11 dígitos, pero una más económica a menudo sólo tiene 6.)

En una computadora electrónica, los errores de redondeo aparecen por las mismas razones y afectan los resultados de los cálculos [Wilkinson]. En algunos casos, los errores de redondeo causan efectos muy serios y hacen que los resultados de los cálculos carezcan por completo de sentido. Por lo tanto, es importante aprender algunos aspectos básicos de las operaciones aritméticas en las computadoras y comprender bajo qué circunstancias pueden ocurrir severos errores de redondeo. Muchos de los problemas de error por redondeo se pueden evitar por medio de prácticas de programación adecuadas.

1.3.1 Base de los números

El sistema numérico que usamos cotidianamente se llama *sistema decimal*. La base del sistema numérico decimal es 10. Sin embargo, las computadoras no usan el sistema decimal en los cálculos ni en la memoria, sino que usan el binario. Este sistema es natural para las computadoras ya que su memoria consiste de un enorme número de dispositivos de registro magnético y electrónico, en los que cada elemento sólo tiene los estados de "encendido" y "apagado".

Sin embargo, si examinamos los lenguajes de máquina, pronto nos percatamos que se usan otros sistemas numéricos, en particular el octal y el hexadecimal [Hannula; Bartee]. Estos sistemas son parientes cercanos del binario y pueden traducirse con facilidad al o del binario. Las expresiones en octal o hexadecimal son más cortas que en binario, por lo que es más sencillo que las personas las lean y comprendan. El hexadecimal también proporciona un uso más eficiente del espacio de la memoria para los números reales (como se explicará más adelante).

La base de un sistema numérico también recibe el nombre de *raíz*. Para el sistema decimal ésta es 10; para el sistema octal es 8 y 2 para el binario. La raíz del sistema hexadecimal es 16.

La base de un número se denota por medio de un subíndice; por ejemplo, $(3.224)_{10}$ es 3.224 en base 10 (decimal), $(1001.11)_2$ es 1001.11 en base 2 (binario) y $(18C7.90)_{16}$ es 18C7.90 en base 16 (hexadecimal).*

El valor decimal de un número en base r, por ejemplo,

$$(\text{abcdefg} \cdot \text{hijk})_r$$

*En hexadecimal, cada dígito puede variar desde 0 hasta 15. Los dígitos del 10 al 15 se expresan por medio de las letras mayúsculas, A, B, C, D, E y H. Los valores decimales de los dígitos hexadecimales se muestran a continuación:

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

se calcula como

$$ar^6 + br^5 + cr^4 + dr^3 + er^2 + fr + g + hr^{-1} + ir^{-2} + jr^{-3} + kr^{-4}$$

Los números que aparecen sin subíndice en este libro están en base 10, a menos que se indique lo contrario.

1.3.2 Rango de constantes numéricas

Las constantes numéricas que se usan en un programa se clasifican en tres categorías: a) enteros, b) números reales y c) números complejos. Sin embargo, un lenguaje de programación no siempre puede manipular de manera directa números complejos. Los números que se usan en los programas son decimales, a menos que se indique lo contrario.

En BASIC para IBM PC [IBM; Goldstein/Goldstein], el mayor entero posible es 32767 y el menor entero negativo es -32768. La magnitud del segundo es mejor que la del primero por 1. En el lenguaje estándar FORTRAN 77 para IBM 370 (y otras computadoras mainframe IBM), el rango de los enteros es desde $2^{31} - 1$ hasta $-(2^{31} - 1)$ inclusive.

La menor y la mayor magnitud de un número real que se pueden representar en una computadora varían de acuerdo con el diseño tanto del hardware como del software. En la IBM PC (BASIC) el rango aproximado es de 2.9×10^{-39} hasta 1.7×10^{38}. El rango aproximado en la IBM 370 es desde 5.4×10^{-79} (o 2147483647) hasta 7.2×10^{75}. Véase la tabla 1.1 para datos análogos de otras computadoras.

Debemos comprender que los números reales en una computadora no son continuos. Si nos fijamos en los números cercanos al cero, el número positivo más pequeño en la IBM PC es 2.9×10^{-39}. Por lo tanto, no se pueden representar números entre cero y 2.9×10^{-39}. El intervalo entre el número positivo (2.9×10^{-39}) y el siguiente menor número positivo es aproximadamente de 3.45×10^{-46} que es mucho menor que 2.9×10^{-39}.

En la IBM PC, la diferencia entre 1 y el menor número mayor que 1 (pero distinguible de 1) es de 1.19×10^{-7}. Este intervalo se llama el *épsilon de la máquina* [Forsythe/Malcolm/Moler; Cheney/Kincaid; Yakowitz/Szidarovsky]. El intervalo entre cualquier número real y el siguiente es igual a

$$(\text{épsilon de la máquina}) \times R$$

donde R es el número real. Posteriormente se darán más detalles sobre el épsilon de la máquina.

Tabla 1.1 Rango de los números reales (precisión simple)

IBM PC (BASIC)	2.9E − 39	1.7E + 38
IBM 370	5.4E − 79	7.2E + 75
Cray XMP	4.6E − 2476	5.4E + 2465
VAX	2.9E − 39	1.7E + 38

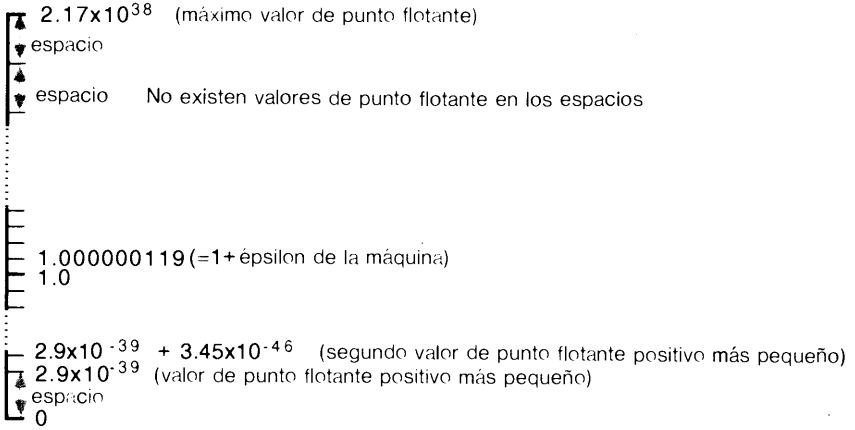


Figura 1.1 Distribución de los números reales en la IBM PC (BASIC). (Las marcas representan los números reales.)

Los números reales positivos se muestran de manera gráfica en la figura 1.1, tal y como se representarían en la computadora.

1.3.3 Números dentro del hardware de la computadora

Un *bit* es la abreviatura de dígito binario (*bi*nary digi*t*) y representa un elemento de memoria que consta de posiciones de encendido y apagado, a la manera de un dispositivo semiconductor o un punto magnético en una superficie de registro. Un *byte* es un conjunto de bits considerado como una unidad, que normalmente está formado por 8 bits.

Las formas en que se usan los bits para los valores enteros y de punto flotante varían según el diseño de una computadora. El resto de esta sección describe ejemplos característicos del uso de los bits para almacenar números.

Enteros. En el sistema de numeración binario, la expresión matemática de un entero es

$$\pm\, a_k a_{k-1} a_{k-2} \cdots a_2 a_1 a_0 \tag{1.3.1}$$

donde a^i es un bit con valor 0 o 1. Su valor decimal es

$$I = \pm [a_k 2^k + a_{k-1} 2^{k-1} + \cdots a_2 2^2 + a_1 2 + a_0] \tag{1.3.2}$$

Por ejemplo, el número binario dado por

$$\pm\ 110101$$

es igual a

$$\begin{aligned} I &= \pm [(1)(2^5) + (1)(2^4) + (0)(2^3) + (1)(2^2) + (0)(2) + (1)] \\ &= \pm [32 + 16 + 0 + 4 + 0 + 1] = \pm 53 \end{aligned} \tag{1.3.3}$$

En una computadora, el valor máximo de k en la ecuación (1.3.1) se limita, debido al diseño del hardware. En la IBM PC (BASIC), se usan 2 bytes (o, de manera equivalente, 16 bits) para representar un entero. El primer bit registra el signo: positivo si es 0, negativo si es 1. Los restantes 15 bits se usan para los a^i. Por lo tanto, el máximo entero positivo es

Binario:	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
(bit no:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15)	(1.3.4)

El valor decimal de lo anterior es

$$\sum_{i=0}^{14} 2^i = 32767$$

Una forma de almacenar un número negativo es utilizar los mismos dígitos que el número positivo de la misma magnitud, excepto que el primer bit se pone en 1. Sin embargo, muchas computadoras usan el *complemento a dos* para almacenar números negativos. Por ejemplo, el complemento a dos para $(-32767)^{10}$ es

Binario:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
(bit no:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15)	(1.3.5)

Los bits de la ecuación (1.3.5) se obtienen a partir de la ecuación (1.3.4), cambiando los 0 por 1, los 1 por 0 y añadiendo 1 al resultado para el número 32767. En el complemento de dos, se determina primero el valor decimal como si los 16 bits expresaran un número positivo. Si este número es menor que 2^{15}, o 32768, se le interpreta como positivo. Si es mayor o igual, entonces se transforma en un número negativo restándole 2^{16}. En el ejemplo anterior del número binario, el equivalente decimal de éste en la ecuación (1.3.5) es $Z = 2^{15} + 1$, por lo que la resta da

$$32768 + 1 - 2^{16} = 32768 + 1 - 65536 = -32767 \tag{1.3.6}$$

El entero negativo de menor magnitud se representa por

$$(1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111)_2$$

que es igual a -1 en decimal.

En la IBM 370, se usan 4 bytes para un entero. Por lo tanto, el máximo número positivo es $2^{32-1} - 1 = 2147483648 - 1 = 2147483647$.

Números reales. El formato para un número real en una computadora difiere según el diseño de hardware y software. Así, nos centraremos en la IBM PC (BASIC) y en la mainframe red IBM (FORTRAN 77) como ejemplos principales.

Los números reales de la IBM PC (BASIC) se almacenan en el formato de punto flotante formalizado en binario. En precisión simple, se usan 4 bytes, o 32 bits, para almacenar un número real. Si se introduce como dato un número decimal, primero se convierte al binario más cercano en el formato normalizado:

$$(\pm 0.abbbbbbbb \ldots bbbb)_2 \times 2^z \tag{1.3.7}$$

donde a siempre es 1, cada b es un dígito binario 0 o 1 y z es un exponente que también se expresa en binario. Existen 24 dígitos para la mantisa incluyendo la a y las b.

Los 32 bits se distribuyen de la manera siguiente. El primer bit se usa para el signo de la mantisa, los siguientes 8 bits para el exponente z y los últimos 23 para la mantisa.

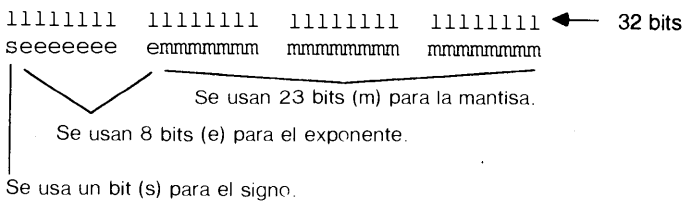


Figura 1.2 Distribución de 32 bits en la IBM PC (BASIC)

En el formato de punto flotante normalizado, el primer dígito de la mantisa siempre es 1, por lo que no se almacena físicamente. Esto explica por qué una mantisa de 24 bits se almacena en 23.

Si los 8 bits asignados al exponente se usan sólo para enteros positivos, el exponente puede representar desde 0 hasta $2^8 - 1 = 255$, aunque puede incluir números negativos. Para registrar exponentes positivos y negativos, el exponente en decimal es sesgado (o sumado) con 128 y después convertido a binario (complemento a dos). Por ejemplo, si el exponente es -3, entonces $-3 + 128 = 125$ se convierte a binario y se almacena en los 8 bits. Por lo tanto, los exponentes que se pueden almacenar en 8 bits van desde $0 - 128 = -128$ hasta $255 - 128 = 127$.

En el FORTRAN 77 de las computadoras mainframe IBM (como la IBM 370) se usa el formato de punto flotante normalizado en hexadecimal, que se escribe como

$$x = (0.abbbbb)_{16} \times 16^k \tag{1.3.8}$$

donde a es un dígito hexadecimal distinto de cero, cada b es un dígito hexadecimal que puede ser cero y k es un exponente expresado en binario. La mantisa tiene 6 dígitos hexadecimales.

En la IBM 370, un valor de punto flotante de precisión simple usa 32 bits; el primero registra el signo de la mantisa, los siguientes 7 bits son para el exponente y los últimos 24 son para la mantisa. El exponente es sesgado por $(64)_{10}$ y almacenado en los 7 bits. Un dígito hexadecimal se representa con 4 bits. Por lo tanto, 6 dígitos hexadecimales de la mantisa necesitan 24 bits; los primeros 4 representan el primer dígito hexadecimal, los siguientes 4 el segundo dígito hexadecimal, y así sucesivamente.

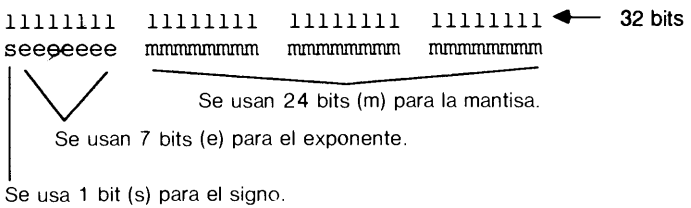


Figura 1.3 Distribución de 32 bits para un valor de punto flotante en la IBM 370

Ahora podemos explicar por qué el número positivo más grande para las mainframe IBM es 7.2×10^{75}. La máxima mantisa positiva que se puede representar con 6 dígitos hexadecimales es $(0.FFFFFF)_{16}$, que es igual a

$$1 - (16)^{-6} \tag{1.3.9}$$

en decimal. El máximo exponente que se pueda representar por el binario de 7 dígitos es

$$127 - 64 = 63$$

donde 64 es el sesgo. Por lo tanto, el máximo valor positivo de punto flotante es

$$(1 - 16^{-6})16^{63} \simeq 7.23 \times 10^{75} \tag{1.3.10}$$

Por otro lado, el menor número positivo es $(0.100000)_{16} \times 16^{-64}$, que es igual a

$$16^{-1} \times 16^{-64} = 16^{-65} = 5.39 \times 10^{-79} \tag{1.3.11}$$

en decimal.

El épsilon de la máquina en mainframe IBM es

$$16^{-6} \times 16^{1} = 16^{-5} = 9.53 \times 10^{-7} \tag{1.3.12}$$

El número de bits que se usan en las computadoras comunes se resume en la tabla 1.2.

Tabla 1.2 Número de bits para los números de punto flotante

	Precisión simple		Precisión doble	
	Mantisa	Exponente	Mantisa	Exponente
IBM PC, AT, XT[a]	23	8	55	8
IBM 370	24	7	56	7
CDC 7600 y Cyber	48	11	96	11
VAX 11	23	8	55	8
Cray XMP	48	11	96	11

[a]Basado en Microsoft Basic [IBM].

1.3.4 Errores de redondeo en una computadora

Errores de redondeo al almacenar un número en memoria. La causa fundamental de errores en una computadora se atribuye al error de representar un número real mediante un número limitado de bits.

Como ya se explicó, el épsilon de la máquina, ε, es el tamaño del intervalo entre 1 y el siguiente número mayor que 1 distinguible de 1. Esto significa que ningún número entre 1 y $1 + \varepsilon$ se puede representar en la computadora. En el caso de la IBM PC (BASIC), cualquier número $1 + \alpha$ se redondea a 1 si $0 < \alpha < \varepsilon/2$, o se redondea a $1 + \varepsilon$ si $\varepsilon/2 \leqslant \alpha$. Así, se puede considerar que $\varepsilon/2$ es el máximo error posible de redondeo para 1. En otras palabras, cuando se halla 1.0 en la memoria, el valor original pudo ser alguno de entre $1 - \varepsilon/2 < x < 1 + \varepsilon/2$.

El épsilon de la máquina se puede determinar mediante el programa siguiente:

```
10  E=1
20  IF E+1>1 THEN PRINT E ELSE STOP
30  E=E/2: GOTO 20
```

El último número impreso por el programa es igual al épsilon de la máquina. Los épsilon en precisión simple para algunas computadoras son:

Precisión	IBM PC	IBM 370	VAX 11	Cray XMP
Simple	1.19E − 7	9.53E − 7	5.96E − 8	3.55E − 15
Doble	2.77E − 17	2.22E − 16	1.38E − 17	1.26E − 29

El error de redondeo implicado en el almacenamiento de cualquier número real R en memoria es aproximadamente igual a $\varepsilon R/2$, si el número se redondea por exceso y εR si se redondea por defecto.

EFECTOS DE LOS ERRORES POR REDONDEO. Si se suman o restan números, la representación exacta del resultado quizá necesite un número de dígitos mucho mayor que el necesario para los números sumados o restados.

Existen dos tipos de situaciones en los que aparecen muchos errores por redondeo: a) cuando se suma (o se resta) un número muy pequeño de uno muy grande y b) cuando un número se resta de otro que es muy cercano.

Para probar el primer caso en la computadora, sumemos 0.00001 a la unidad diez mil veces. El diseño de un programa para este trabajo sería:

```
10  suma=1
20  for i=1 to 10000
30      suma=suma+0.00001
40  next
50  print"SUMA=";suma
```

El resultado de este programa en la IBM PC es

$$\text{SUMA} = 1.100136$$

Puesto que la respuesta exacta es 1.1, el error relativo de este cálculo es

$$\frac{1.1 - 1.100136}{1.1} = -0.000124 \text{ o bien } -0.0124\% \tag{1.3.13}$$

Otro problema molesto es que dos números que debiesen ser matemáticamente idénticos no siempre lo son en las computadoras. Por ejemplo, consideremos la ecuación

```
y=A/B
w=y*B
z=A-w
```

donde A y B son constantes. Desde un punto de vista matemático, w es igual a A, por lo que z debe anularse. Si estas ecuaciones se calculan en una computadora, z se anula o es un valor no nulo pero muy pequeño, dependiendo de los valores de A y B. Pruebe el programa siguiente:

```
    A=COS(0.3)
    DO 10 K=1,20
       B=SIN(FLOAT(K))
       Z=A/B
       W=Z*B
       Y=A-W
       PRINT *,A,B,W,Y
10  CONTINUE
    END
```

Lo que ocurre en la computadora, es que aparece un error de redondeo cuando se calculan Z = A/B y W = Z*B y se almacenan. Así, W = Z*B en la quinta línea no es exactamente igual a A. La magnitud relativa del error por redondeo atribuida a la multiplicación o división entre una constante y al almacenamiento del resultado en la memoria es casi igual al épsilon de la máquina.

El error de un número provocado por el redondeo aumenta cuando el número de operaciones aritméticas también se incrementa [Wilkinson].

Causas de errores por redondeo. Para explicar cómo surgen los errores por redondeo, consideremos el cálculo de 1 + 0.00001 en la IBM PC. Las representaciones binarias de 1 y 0.00001 son, respectivamente,

$$\begin{aligned}(1)_{10} &= (0.1000\ 0000\ 0000\ 0000\ 0000\ 0000)_2 \times 2^1 \\ (0.00001)_{10} &= (0.1010\ 0111\ 1100\ 0101\ 1010\ 1100)_2 \times 2^{-16}\end{aligned} \tag{1.3.15}$$

La suma de estos dos números es

$$\begin{aligned}&(1)_{10} + (0.00001)_{10} \\ &\quad = (0.1000\ 0000\ 0000\ 0000\ 0101\ 0011\ 1110\ 0010\ 1101\ 0110\ 0)_2 \times 2^1 \\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad *\end{aligned} \tag{1.3.16}$$

Sin embargo los números a partir del asterisco (*) se redondean ya que la mantisa tiene 24 bits. Por lo tanto, el resultado de este cálculo se guarda en memoria como

$$(1)_{10} + (0.00001)_{10} \simeq (0.1000\ 0000\ 0000\ 0000\ 0101\ 0100)_2 \times 2^1 \tag{1.3.17}$$

que es equivalente a $(1.0000\ 1001\ 36)_{10}$.

Así, siempre que se sume 0.00001 a 1, el resultado agrega 0.0000000136 como error. Al repetir diez mil veces la suma de 0.00001 a 1, se genera un error de exactamente diez mil veces 0.0000000136. Aunque el resultado calculado se incrementa en el presente ejemplo, puede disminuir si algunos dígitos se redondean por defecto. La pérdida y ganancia se conocen como *error de redondeo*.

A continuación ilustramos el efecto de los errores por redondeo implicados al restar un número. Del cálculo sabemos que

$$\lim_{\theta \to 0} \frac{f(x+\theta) - f(x)}{\theta} = f'(x) \tag{1.3.18}$$

Con el fin de ilustrar, hacemos $f(x)$ = sen (x) y calculamos

$$d = \frac{\text{sen}\,(1+\theta) - \text{sen}\,(1)}{\theta} \tag{1.3.19}$$

con valores decrecientes de θ. Los resultados calculados en una IBM PC se muestran a continuación.

θ	d	(Valor exacto — d)
0.1	0.49736	0.042938
0.01	0.53607	0.004224
0.001	0.53989	0.000403
0.0001	0.54061	−0.003111
0.00001	0.53644	0.003860
0.000001	0.53644	0.003860
0.0000001	0.59604	−0.055744
Valor exacto = cos (1) 0.54030		

Se observa que cuando θ decrece, d se acerca al valor exacto hasta que θ llega a 0.0001, pero entonces el error empieza a crecer al seguir decreciendo θ. El incremento de los errores al decrecer θ ocurre debido a que cuando la diferencia entre $f(1 + \theta)$ y $f(1)$ se vuelve pequeña, aumenta el error de redondeo con respecto a θ. Los errores de d para valores grandes de θ se deben a los errores de truncamiento de la aproximación, véase la ecuación (1.3.19).

Para analizar el redondeo en la resta, consideremos el cálculo de 1.00001 − 1. Puesto que ya sabemos que 1.00001 se almacena en binario como

$$(0.1000\ 0000\ 0000\ 0000\ 0101\ 0100)_2 \times 2^1$$

entonces, 1.00001 − 1 es

$$\begin{aligned}(0.1000\ 0000\ 0000\ 000\ 0101\ 0100)_2 \times 2^1 &- (0.1)_2 \times 2^1 \\ &= (0.0000\ 0000\ 0000\ 0000\ 0101\ 0100)_2 \times 2^1 \\ &= (0.1010\ 1)_2 \times 2^{-16} \qquad (1.3.20)\end{aligned}$$

Su valor decimal es 1.00136×10^{-5}. Al comparar esto con el valor exacto, 0.00001, el error relativo es

$$(0.0000100136 - 0.00001)/0.00001 = 0.00136$$

o 0.136%

Estrategias. Los efectos de redondeo se pueden minimizar cambiando el algoritmo de cálculo, aunque éste debe diseñarse caso por caso. Algunas estrategias útiles son:

Doble precisión [McCracken]
Agrupamiento
Desarrollos de Taylor
Cambio de definición de variables
Reescritura de la ecuación para evitar restas

Las aplicaciones de estos enfoques se ilustran en el ejemplo 1.1.

Ejemplo 1.1

Sume 0.00001 diez mil veces a la unidad usando a) el método de agrupamiento y b) el de doble precisión.

(Solución)

a) Cuando se calcula la suma de muchos números pequeños, su agrupación ayuda a reducir los errores por redondeo. Un programa para sumar 0.00001 a 1 diez mil veces se podría escribir como

```
(BASIC)
10 SUMA=1
20 FOR I=1 to 100
22 TOTAL=0
25   FOR K=1 to 100
30     TOTAL=TOTAL+0.00001
40   NEXT
45   SUMA=SUMA+TOTAL
47 NEXT
60 PRINT "SUMA=";SUMA
```

```
(FORTRAN)
   SUMA=1
   DO 47 I=1,100
      TOTAL=0
      DO 40 K=1,100
         TOTAL=TOTAL+0.00001
40     CONTINUE
      SUMA=SUMA+TOTAL
47 CONTINUE
   PRINT *,SUMA
```

En el programa anterior, los valores pequeños se agrupan en conjuntos de 100, se calcula el total del grupo y después se acumulan los totales de grupo. La respuesta de este programa revisado es

SUMA = 1.100000

Otro enfoque es el de usar la doble precisión para la suma SUMA como sigue:

```
10  SUMA#=0
20  FOR i=1 to 10000
30     SUMA  =SUMA#+0.00001
40  NEXT
50  PRINT "SUMA=";SUMA #
```

donde el signo = después de SUMA indica que SUMA es una variable de doble precisión en BASIC. El resultado de esta versión es

SUMA = 1.09999999747

La comparación de los dos enfoques muestra que el agrupamiento es más efectivo que el uso de la doble precisión.

Ejemplo 1.2

Cuando θ tiende a 0, la precisión de una evaluación numérica para

$$d \equiv \frac{\text{sen}\,(1+\theta) - \text{sen}(1)}{\theta}$$

se vuelve muy pobre debido a los errores por redondeo. Por medio del *desarrollo de Taylor*, podemos reescribir la ecuación de tal forma que se mejore la exactitud para θ.

(Solución)

El desarrollo de Taylor es útil cuando se va a calcular una pequeña diferencia de dos valores funcionales. El desarrollo de Taylor de sen $(1+\theta)$ es

$$\text{sen}\,(1+\theta) = \text{sen}(1) + \theta\cos(1) - 0.5\,\theta^2\,\text{sen}(1)\cdots$$

Si aproximamos sen $(1+\theta)$ con los primeros tres términos, d es

$$d = \cos(1) - 0.5\theta\,\text{sen}(1) \qquad \text{(A)}$$

Los valores calculados para varias θ son

θ	d
0.1	0.49822
0.01	0.53609
0.001	0.53988
0.0001	0.54026
0.00001	0.54030
0.000001	0.54030
0.0000001	0.54030
Valor exacto =	0.54030

La precisión de la aproximación aumenta cuando θ tiende a 0.

Ejemplo 1.3

Si la siguiente ecuación se calcula de modo directo en un programa, aparecen errores de redondeo cuando x tiende a $+\infty$ y $-\infty$.

$$y = \frac{1}{(a-z)(b-z)} \qquad \text{(A)}$$

donde

$$z = \frac{a + b + (b-a)\tanh(x)}{2} \qquad \text{(B)}$$

Reescriba las ecuaciones, de forma que no ocurran errores importantes de redondeo.

(Solución)

Puesto que $-1 < \tanh(x) < 1$, el dominio de z es $a < z < b$. Cuando x tiende a ∞, z tiende a b; y cuando x tiende a $-\infty$, z tiende a b. Hay que considerar dos causas de errores por redondeo.

Si $b = 0$, el numerador de la ecuación B) se convierte en $a[1 - \tanh(x)]$. Por lo tanto, habrá un error de redondeo severo cuando $\tanh(x)$ está muy cerca de 1. La resta de números similares en la ecuación B) se puede evitar reconociendo la relación:

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \qquad \text{(C)}$$

al incorporar la ecuación C) en B7 y reescribirla se obtiene

$$z = \frac{b \exp(x) + a \exp(-x)}{\exp(x) + \exp(-x)}$$

Cuando z tiende a a o b, se tiene un error por redondeo en el denominador de la ecuación A). Para evitar esto, se puede dividir el cálculo de la ecuación A) en dos casos:

Caso 1 $\quad a < z < (a + b)/2$

Caso 2 $\quad (a + b)/2 \leqslant z < b$

Para el caso 1, se escribe el término $a - z$ del denominador de la ecuación A) como

$$a - z = -\frac{b \exp(x) + a \exp(-x)}{\exp(x) + \exp(-x)}$$

$$= \frac{(a - b) \exp(x)}{\exp(x) + \exp(-x)}$$

Para el caso 2, se escribe $b - z$ como

$$b - z = b - \frac{b \exp(x) + a \exp(-x)}{\exp(x) + \exp(-x)} = \frac{(b - a) \exp(-x)}{\exp(x) + \exp(-x)}$$

Resumen de Esta Sección

a) La máxima magnitud de los enteros positivos y negativos se limita mediante el número de bytes utilizados. Con el complemento a dos, el último es más grande que el primero por 1.

b) El intervalo entre 1 y el siguiente número real se llama *épsilon de la máquina*. El intervalo entre cualesquier dos números reales consecutivos en una computadora —excepto en 0— es aproximadamente igual al número real por el épsilon de la máquina.

c) Los errores de redondeo son causantes de errores en los cálculos. Una cantidad significativa de errores por redondeo aparece, de manera particular cuando se suma un número pequeño a uno grande o cuando un número se resta de un número similar.

d) Los efectos de los errores por redondeo se pueden reducir mediante varios enfoques, incluyendo el uso de la doble precisión, la reescritura de las ecuaciones y la expansión de una función en polinomios. Sin embargo, la que es mejor depende de la naturaleza del cálculo a realizar.

PROGRAMA

PROGRAMA 1-1 Conversión de decimal a binario

A) Explicaciones

Este programa convierte un valor decimal positivo menor de 1.0E + 38 a un binario en el formato de punto flotante normalizado con una mantisa de 24 bits. El programa trabaja de manera interactiva. Al ejecutarlo, pide un valor decimal, que después se convierte en binario. Se imprimen por separado la mantisa y el exponente cuando se termina la conversión.

B) Variables

A(k): bit *k*-ésimo en la mantisa normalizada
B: 2^I
I: Parámetro
L: mantisa
X: de entrada valor decimal

C) Listado

```
C-----CSL/F1-1.FOR        CONVERSION DE DECIMAL A BINARIO (FORTRAN)
      INTEGER A(255)
      PRINT *
      PRINT *, 'CSL/F1-1     CONVERSION DE DECIMAL A BINARIO (FORTRAN)'
      PRINT *
   10 PRINT *,' INTRODUZCA EL VALOR DECIMAL ?'
      READ *, X
      L=0
      K=1
      I=LOG(X)/LOG(2.0) + 2
   70 I=I-1
      IF (I.LT.-200) STOP
      B=2.0**FLOAT(I-1)
      IF (X.GE.B) THEN
         A(K)=1
         X=X-B
         IF (L.EQ.0) M=I
         IF (L.EQ.0) L=1
      ELSE
         IF (K.GT.1) A(K)=0
      END IF
      IF (L.GT.0) K=K+1
```

```
      IF (K.LT.25) GOTO 70
      PRINT *
      PRINT *,'----------------------------------------'
      PRINT *,'BINARIO'
      PRINT 30, (A(K), K=1,24)
   30 FORMAT( 1X,' MANTISA=    '    ,10(4I1,1X))
      PRINT 40,M
   40 FORMAT(' EXPONENTE=',I3)
      PRINT *,'----------------------------------------'
      PRINT *
      PRINT *
      PRINT*,' OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR'
      READ *,K
      IF(K.EQ.1) GOTO 10
      PRINT*
      END
```

D) Ejemplo de salida

```
CSL/F1-1    CONVERSION DE DECIMAL A BINARIO (FORTRAN)
----------------------------------------
INTRODUZCA EL VALOR DECIMAL:?
0.5
BINARIO
   MANTISA=  1000 0000 0000 0000 0000 0000
 EXPONENTE=  0
----------------------------------------
INTRODUZCA EL VALOR DECIMAL:?
64.0
BINARIO
   MANTISA=  1000 0000 0000 0000 0000 0000
 EXPONENTE=  7
----------------------------------------
INTRODUZCA EL VALOR DECIMAL:?
0.0001751
BINARIO
 MANTISA=    1011 0111 1001 1011 0000 1100
 EXPONENTE=-12
----------------------------------------
```

PROBLEMAS

1.1) Si se usan 8 bits para representar los enteros positivos y negativos en complemento a dos, ¿cuál es el entero positivo más grande y el negativo más pequeño (en magnitud) en decimal?

1.2) Se tienen dos números binarios de 16 bits en complemento a dos:

a) Binario:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
(bit no:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15)
b) Binario:	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
(bit no:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15)

Determine los valores decimales de los dos números binarios.

1.3) Hallar el épsilon de la máquina para una IBM PC en WATFOR-77.

1.4) Repita el problema anterior con la computadora mainframe a la que tenga acceso.

1.5) Evalúe

$$\exp(x) - 1$$

para $x = 0.0001$, aplicando el desarrollo de Taylor para $\exp(x)$. Use los primeros tres términos.

1.6) Desarrolle las siguientes funciones en serie de Maclaurin.

$$1/(1 + x^2)$$
$$\tan(x)$$
$$1/(1 - x)$$
$$\ln(1 + x)$$

1.7) Muestre que el desarrollo de Taylor de $\ln[(1 + x)/(1 - x)]$ alrededor de $x = 1$ es

$$2 \sum_{n=1}^{\infty} \frac{x^{2n-1}}{2n - 1}$$

1.8) Por medio del desarrollo de Maclaurin de e^x y e^{-x}, obtenga el desarrollo de Maclaurin de senh (x) y cosh (x), donde

$$\text{senh}(x) = \frac{1}{2}(e^x - e^{-x})$$
$$\cosh(x) = \frac{1}{2}(e^x + e^{-x})$$

1.9 a) Si la siguiente función se escribe en un programa, ¿en cuál rango de x aparecerá un desborde o una división entre cero originados por el error de redondeo?

$$f(x) = \frac{1}{1 - \tanh(x)}$$

Suponga que el número positivo más pequeño es 3×10^{-39} y el épsilon de la máquina es 1.2×10^{-7}.

b) Reescriba la ecuación de tal forma que no se necesite restar.

BIBLIOGRAFIA

Bartee, T.C. *Digital Computer Fundamentals*, McGraw-Hill, 1981.

Cheney, W., y D. Kincaid, *Numerical Mathematics and Computing*, Brooks/Cole, 1985.

Conte, D.C., y S.D. de Boor, *Elementary Numerical Analysis: An Algorithmic Approach* McGraw-Hill, 1980.

Forsythe, G.E., M.A. Malcolm y C.B. Moler, *Computer Methods for Mathematical Computations*, Prentice-Hall, 1977.

Goldstein, L.J. y M. Goldstein, *IBM PC*, Prentice-Hall and Communications, 1984.

Hannula, R. *Computing and Programming*, Houghton Mifflin, 1974.

Hornbeck, W.H., *Numerical Methods*, Quantum, 1975.

International Business Machines Corporation (IBM), *IBM System/360 and System/370* Fortran IV Language, GC28-6515-9.

IBM, "Basic by Microsoft Corp.", IBM Personal Computer Hardware Reference Library 6025013, 1982.

King, J.T., *Introduction to Numerical Computations*, McGraw-Hill, 1984.

Kline, R.M., *Structured Digital Design Including MSI/LSI Components and Microprocessors*, Prentice-Hall, 1983.

Leventhal, L.A., *Introduction to Microprocessors*, Software, Programming, Prentice-Hall, 1978.

McCracken, D.D., *Computing for Engineers and Scientists with Fortran 77*, Wiley, 1984.

Morris, J.L., *Computational Methods in Elementary Numerical Analysis,* Wiley, 1983.

Sterbenz, P.H., *Floating-Point Computations*, Prentice-Hall, 1974.

Wilkinson, J.H., *Rounding Errors in Algebraic Processes*, Prentice-Hall, 1963.

Yakowitz, S. y F. Szidarovszky, *An Introduction to Numerical Computations*, Mcmillan, 1986.

2
Interpolación polinomial

2.1 INTRODUCCION

Una función de *interpolación* es aquella que pasa a través de puntos dados como datos, los cuales se muestran comúnmente por medio de una tabla de valores o se toman directamente de una función dada.

La interpolación de los datos puede hacerse mediante un polinomio, las funciones *spline*, una función racional o las series de Fourier entre otras posibles formas [Stoer/Burlish]. La interpolación polinomial (ajustar un polinomio a los puntos dados) es uno de los temas más importantes en métodos numéricos, ya que la mayoría de los demás modelos numéricos se basan en la interpolación polinomial. Por ejemplo, los modelos de integración numérica se obtienen integrando fórmulas de interpolación polinomial, y los modelos de diferenciación numérica se obtienen derivando las interpolaciones polinomiales. Por esto, el objetivo del capítulo es analizar los aspectos básicos de la interpolación polinomial y sus aplicaciones. La interpolación de splines cúbicos y la interpolación transfinita se describen en los apéndices G y H, respectivamente.

Los datos obtenidos mediante una medición pueden interpolarse, pero en la mayoría de los casos no es recomendable una interpolación directa debido a los errores aleatorios implicados en la medición. Así pues, el ajuste de una curva a los datos obtenidos de esta forma se describe por separado en el capítulo 8.

La tabla 2.1 da un breve panorama de los métodos de interpolación descritos en este capítulo.

2.2 INTERPOLACION LINEAL

Esta interpolación es la base para varios modelos numéricos fundamentales. Al integrar la interpolación lineal, se deduce el modelo de integración llamado *regla del*

Tabla 2.1 Resumen de los esquemas de interpolación en dimensión uno

Esquema de interpolación	Ventajas	Desventajas
Interpolación de Lagrange	Forma conveniente. Fácil de programar.	Difícil de manejar para los cálculos manuales.
Interpolación de Newton	El orden del polinomio puede cambiarse sin problemas. La evaluación de errores es fácil.	Se debe preparar una tabla de diferencias o de diferencias divididas.
Interpolación de Lagrange mediante puntos de Chebyshev	Los errores se distribuyen más uniformemente que en la malla que presenta igual separación.	Los puntos de la malla no están distribuidos de manera uniforme.
Interpolación de Hermite	Alta precisión debido a que el binomio se ajusta también a las derivadas.	Necesita los valores de las derivadas.
Spline cúbico (apéndice G)	Aplicable a cualquier número de datos.	Se necesitan resolver ecuaciones simultáneas.

trapecio. El gradiente de la interpolación lineal es una aproximación a la primera derivada de la función. El objetivo de esta sección es introducir la interpolación lineal y analizar a continuación sus errores.

La interpolación lineal da como resultado una recta que se ajusta a dos puntos dados. La interpolación lineal que se muestra en la figura 2.1 está dada por

$$g(x) = \frac{b-x}{b-a} f(a) + \frac{x-a}{b-a} f(b) \tag{2.2.1}$$

donde $f(a)$ y $f(b)$ son valores conocidos de $f(x)$ en $x = a$ y $x = b$ respectivamente.

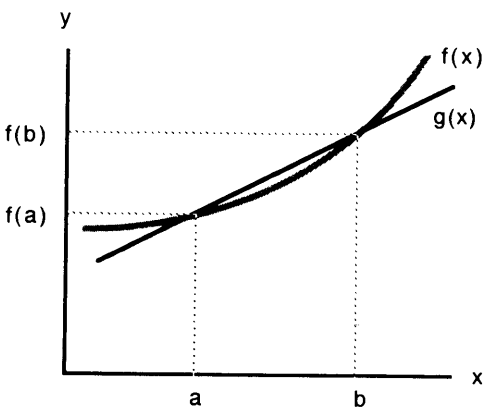


Figura 2.1 Interpolación lineal

El error de la interpolación lineal se puede expresar en la forma:

$$e(x) = \frac{1}{2}(x - a)(x - b)f''(\xi), \quad a \leqslant \xi \leqslant b \tag{2.2.2}$$

donde ξ (una letra griega llamada "xi") depende de x pero está en algún lugar entre a y b (la demostración de la ecuación (2.2.2) está dada en el apéndice A). La ecuación (2.2.2) es una función un poco difícil de manejar ya que no tenemos forma de evaluar a ξ exactamente. Sin embargo, es posible analizar $x(x)$ cuando $f''(x)$ se aproxima mediante una constante en $a \leqslant x \leqslant b$ como se explica a continuación.

Si f'' es una función con poca variación, o si el intervalo $[a, b]$ es pequeño, de forma que f'' cambie un poco, podemos aproximar $f''(\xi)$ mediante $f''(x_m)$, donde x_m es el punto medio entre a y b: $x_m = (a + b)/2$. La ecuación (2.2.2) indica entonces que:

a) El error máximo aparece aproximadamente en el punto medio entre los datos dados.
b) El error aumenta cuando $b - a$ crece.
c) El error también se incrementa cuando $|f''|$ crece.

Una excepción a estas tendencias es cuando f'' tiene una raíz en el intervalo $[a, b]$ porque la afirmación de que f'' es aproximadamente constante no es válida.

Resumen de Esta Sección

a) Por interpolación lineal se obtiene una recta que se ajusta a dos datos dados.
b) Si el signo de $f(x)$ no cambia en $a \leqslant x \leqslant b$, el error máximo de una interpolación lineal aparece aproximadamente en el punto medio y su magnitud es proporcional a la segunda derivada de la función aproximada.

2.3 FORMULA DE INTERPOLACION DE LAGRANGE

¿Pueden ajustarse tres o cuatro datos por medio de una curva? Uno de los métodos fundamentales para encontrar una función que pase a través de datos dados es el de usar un polinomio (véase la figura 2.2).

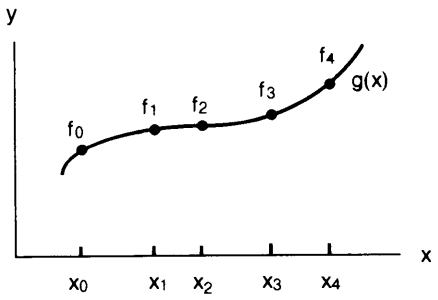


Figura 2.2 Datos ajustados por un polinomio

La interpolación polinomial se puede expresar en varias formas alternativas que pueden transformarse entre sí. Entre éstas se encuentran las series de potencias, la interpolación de Lagrange y la interpolación de Newton hacia atrás y hacia adelante.

Como se verá después con más detalle, un polinomio de orden N que pasa a través de $N + 1$ puntos es único. Esto significa que, independientemente de la fórmula de interpolación, todas las interpolaciones polinomiales que se ajustan a los mismos datos son matemáticamente idénticas.

Suponga que se dan $N + 1$ puntos como

$$\begin{matrix} x_0 & x_1 & \cdots & x_N \\ f_0 & f_1 & \cdots & f_N \end{matrix}$$

donde $x_0, x_1, \ldots$ son las abscisas de los puntos (puntos de la malla) dados en orden creciente. Los espacios entre los puntos de la malla son arbitrarios. El polinomio de orden N que pasa a través de los $N + 1$ puntos se puede escribir en una serie de potencias como

$$g(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_N x^N \tag{2.3.1}$$

donde los a_i son coeficientes. El ajuste de la serie de potencias a los $N + 1$ puntos dados da un sistema de ecuaciones lineales:

$$\begin{aligned} f_0 &= a_0 + a_1 x_0 + a_2 x_0^2 + \cdots + a_N x_0^N \\ f_1 &= a_0 + a_1 x_1 + a_2 x_1^2 + \cdots + a_N x_1^N \\ &\vdots \\ f_N &= a_0 + a_1 x_N + a_2 x_N^2 + \cdots + a_N x_N^N \end{aligned} \tag{2.3.2}$$

Aunque los coeficientes a_i pueden determinarse resolviendo las ecuaciones simultáneas por medio de un programa computacional, dicho intento no es deseable por dos razones. Primera, se necesita un programa que resuelva un conjunto de ecuaciones lineales; y segunda, la solución de la computadora quizá no sea precisa. (Realmente, las potencias de x_i en la ecuación pueden ser números muy grandes, y si es así, el efecto de los errores por redondeo será importante.) Por fortuna, existen mejores métodos para determinar una interpolación polinomial sin resolver las ecuaciones lineales. Entre éstos están la fórmula de interpolación de Lagrange y la fórmula de interpolación de Newton hacia adelante y hacia atrás.

Para presentar la idea básica que subyace en la fórmula de Lagrange, considere el producto de factores dados por

$$V_0(x) = (x - x_1)(x - x_2) \cdots (x - x_N)$$

que se refiere a los $N + 1$ puntos dados antes. La función V_0 es un polinomio de or-

den N de x, y se anula en $x = x_1, x_2, \ldots, x_N$. Si dividimos $V_0(x)$ entre $V_0(x_0)$, la función resultante

$$V_0(x) = \frac{(x - x_1)(x - x_2)\cdots(x - x_N)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_N)}$$

toma el valor de uno para $x = x_0$, y de cero para $x = x_1$, $x = x_2, \ldots, x = x_N$. En forma análoga, podemos escribir V_i como

$$V_i(x) = \frac{(x - x_0)(x - x_1)\cdots(x - x_N)}{(x_i - x_0)(x_i - x_1)\cdots(x_i - x_N)}$$

donde el numerador no incluye $(x - x_i)$ y el denominador no incluye $(x_i - x)$. La función $V_i(x)$ es un polinomio de orden N y toma el valor de uno en $x = x_i$ y de cero en $x = x_j$, $j \neq$ i. Así, si multiplicamos $V_0(x)$, $V_1(x), \ldots, V_N(x)$ por $f_0, f_1, \ldots f_N$, respectivamente y las sumamos, el resultado será un polinomio de orden a lo más N e igual a f_i para cada $i = 0$ hasta $i = N$.

La fórmula de interpolación de Lagrange de orden N así obtenida se escribe como sigue [Conte/de Boor]:

$$\begin{aligned} g(x) = {} & \frac{(x - x_1)(x - x_2)\cdots(x - x_N)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_N)} f_0 \\ & + \frac{(x - x_0)(x - x_2)\cdots(x - x_N)}{(x_1 - x_0)(x_1 - x_2)\cdots(x_1 - x_N)} f_1 \\ & \vdots \\ & + \frac{(x - x_0)(x - x_1)\cdots(x - x_{N-1})}{(x_N - x_0)(x_N - x_1)\cdots(x_N - x_{N-1})} f_N \end{aligned} \tag{2.3.3}$$

La ecuación (2.3.3) es equivalente a la serie de potencias que se determina resolviendo la ecuación lineal. Parece complicado, pero incluso la memorización no es difícil si se entiende la estructura.

Ejemplo 2.1

a) Las densidades de sodio para tres temperaturas están dadas como sigue:

i	Temperatura T_i	Densidad ρ_i
0	94° C	929 kg/m³
1	205	902
2	371	860

Escriba la fórmula de interpolación de Lagrange que se ajusta a los tres datos.

b) Determine la densidad para $T = 251\,°C$ utilizando la interpolación de Lagrange (al calcular el valor de $g(x)$, no desarrolle la fórmula en una serie de potencias).

(Solución)

a) Ya que el número de datos es tres, el orden de la fórmula de Lagrange es $N = 2$. La interpolación de Lagrange queda

$$g(T) = \frac{(T-205)(T-371)}{(94-205)(94-371)}(929) + \frac{(T-94)(T-371)}{(205-94)(205-371)}(902) + \frac{(T-94)(T-205)}{(371-94)(371-205)}(860)$$

b) Sustituyendo $T = 251$ en la ecuación anterior, obtenemos

$$g(251) = 890.5\ \text{kg/m}^3$$

(Comentarios: al evaluar $g(x)$ por un valor dado x, no se debe desarrollar la fórmula de interpolación de Lagrange en una serie de potencias, porque no sólo es molesto sino además se incrementa la posibilidad de cometer errores humanos.)

La ecuación (2.3.3) es particularmente larga si el orden N es grande. Sin embargo, su escritura en un programa de computación necesita únicamente un número pequeño de líneas: Observando la ecuación (2.3.3), se reconoce que el primer término es f_0 veces un producto de

$$\frac{(x - x_i)}{(x_0 - x_i)}$$

para toda i excepto para $i = 0$. El segundo término es f_1 veces un producto de

$$\frac{(x - x_i)}{(x_1 - x_i)}$$

para toda i excepto para $i = 1$. Los otros términos siguen el mismo patrón. Por lo tanto, la ecuación (2.3.3) puede programarse con dos ciclos DO en FORTRAN como sigue:

```
G=0
DO I=0,N
  Z=F(I)
  DO J=0,N
     IF (I.NE.J) Z=Z*(XA-X(J))/(X(I)-X(J))
  END DO
  G=G+Z
END DO
```

Donde las notaciones significan:

```
X(I), F(I), I=0,1,..N : data points
G  : result of interpolation calculation
Z  : product of factors
XA : x
```

VEA EL PROGRAMA 2-1 para una implantación real.

La ecuación (2.3.3) es un polinomio de orden menor o igual que N, ya que cada término del lado derecho es un polinomio de orden N. El orden de un polinomio es menor que N si f_i se obtiene de un polinomio $f(x)$ de orden menor que N. En este caso $g(x)$ es exactamente igual a $f(x)$.

El polinomio de interpolación de orden N que se ajusta a $N + 1$ puntos es único. Esto es importante, ya que indica que todos los polinomios de orden N que se ajustan a un conjunto dado de $N + 1$ puntos son matemáticamente idénticos, aun cuando sus formas sean distintas.

La unicidad del polinomio de interpolación se puede demostrar considerando la hipótesis de que la interpolación de Lagrange no es un polinomio único. Si no es único, debe existir otro polinomio de orden N, $k(x)$ que pasa por los mismos $N + 1$ puntos. La diferencia entre la interpolación de Lagrange $g(x)$ y $k(x)$ definida como

$$r(x) = g(x) - k(x) \tag{2.3.4}$$

debe ser un polinomio de orden menor o igual que N, ya que $g(x)$ y $k(x)$ son ambos polinomios de orden N. Por otro lado, puesto que $g(x)$ y $k(x)$ coinciden ambos en los $N + 1$ puntos dados, $r(x)$ se anula en los $N + 1$ puntos. Esto significa que $r(x)$ tiene $N + 1$ raíces, es decir, $r(x)$ es un polinomio de orden $N + 1$. Esto contradice el hecho de que $r(x)$ sea un polinomio de orden menor o igual que N, lo cual demuestra que la hipótesis es incorrecta.

Cuando una función conocida $f(x)$ se aproxima mediante un polinomio de interpolación, lo que nos interesa es el error del polinomio. El error se define como

$$e(x) = f(x) - g(x) \tag{2.3.5}$$

donde $f(x)$ es la función de la cual se muestrean los datos: $f_i = f(x_i)$. La distribución y magnitud de $e(x)$ se ven afectadas por los siguientes parámetros:

a) La distribución de las abscisas en los datos.
b) El tamaño del dominio de interpolación.
c) El orden del polinomio (o equivalentemente el número de puntos utilizados en la interpolación, menos uno).

Estos aspectos se analizan con más detalle en los siguientes párrafos.

La distribución de x_i que se elige con más frecuencia es la de los puntos con igual espaciamiento (con intervalos espaciados de manera uniforme entre dos absci-

sas consecutivas), pero los x_i con espaciamientos no uniformes, también se usan a menudo (véase la sección 2.5). Aquí supondremos que las x_i están uniformemente espaciadas. Sin embargo, en una malla con espaciamiento uniforme, la magnitud de $e(x)$, a saber $|e(x)|$, tiende a ser pequeña en los intervalos cercanos al centro del dominio y tiende a crecer rápidamente hacia los extremos.

El tamaño del dominio de interpolación definido como

$$D = x_N - x_0 \tag{2.3.6}$$

tiene un defecto significativo en la magnitud y distribución de $e(x)$. En general, el valor máximo de $|e(x)|$ tiende a cero al decrecer D. Por otro lado, si D crece, el valor máximo de $|e(x)|$ aumenta e incluso puede dominar a $|g(x)|$, particularmente para un orden grande N.

Si D permanece fijo pero N se incrementa a partir de un valor pequeño, el error máximo tiende a decrecer hasta un cierto valor de N. A partir de ahí, el error máximo puede empezar a crecer. Debemos entender que nada garantiza que las interpolaciones $g(x)$ convergerán a $f(x)$ al crecer N.

El error de la fórmula de interpolación de Lagrange está dado por una fórmula análoga a la ecuación (2.2.2) para el caso de la interpolación lineal. En el apéndice A se describe cómo se deduce, por lo que podemos escribir aquí sin demostración:

$$e(x) = f(x) - g(x) = L(x)f^{(N+1)}(\xi), \quad x_0 \leqslant \xi \leqslant x_N \tag{2.3.7}$$

donde $N + 1$ es el número de datos, $f^{(N+1)}$ es la $(N + 1)$-ésima derivada de $f(x)$ y

$$L(x) = \frac{(x - x_0)(x - x_1)\cdots(x - x_N)}{(N + 1)!} \tag{2.3.8}$$

En la ecuación (2.3.7), ξ depende de x, pero satisface $x_0 \leqslant \xi \leqslant x_N$. Conviene observar que si $f(x)$ es un polinomio de orden menor o igual que N, la $(N + 1)$-ésima derivada de $g(x)$ se anula. Por lo tanto, el error también se anula. Si $f(x)$ no es de este tipo, entonces tenemos la misma dificultad que en el caso de la ecuación (2.2.3), puesto que ξ depende de una x que no se conoce.

Sin embargo, para un intervalo pequeño $[a, b]$ en el que $f^{N+1}(\xi)$ se pueda aproximar mediante una constante, la ecuación (2.3.7) se escribe como

$$e(x) \simeq L(x)f^{(N+1)}(x_m) \tag{2.3.9}$$

donde x_m es el punto medio entre los dos extremos del intervalo $[a, b]$. Entonces, $e(x)$ es aproximadamente proporcional a $L(x)$ dada por la ecuación (2.3.8). Se puede estimar un valor aproximado de $f^{(N+1)}$ en el intervalo (si se dispone de un dato más) utilizando una aproximación por diferencias. Sin embargo, el hecho de aproximar $f^{(N+1)}(\xi)$ mediante una constante no es apropiado en las siguientes situaciones: cuando $[a, b]$ es un intervalo grande o $f^{(N+1)}(x)$ cambia de manera sustancial, y cuando $f^{(N+1)}(x)$ cambia de signo en la parte media del dominio.

La función $L(x)$ tiene efectos significativos en la distribución del error, de la manera siguiente:

a) En una retícula con espaciamiento uniforme, la amplitud de oscilación de $L(x)$ es mínima en el centro del rango de interpolación, pero crece hacia los extremos. Esto provoca un incremento de los errores hacia los extremos.
b) Al aumentar el tamaño del rango de interpolación la amplitud de oscilación crece rápidamente.

La figura 2.3 muestra $L(x)$ para la interpolación de Lagrange cuando se usan seis puntos en la malla, en un rango de $0 \leqslant x \leqslant 5$. El máximo de $|L(x)|$ aparece en los intervalos de extrema izquierda y de extrema derecha, mientras que el máximo local de $|L(x)|$ en cualquier intervalo de la retícula es más pequeño en el intervalo que se encuentra en el centro del dominio.

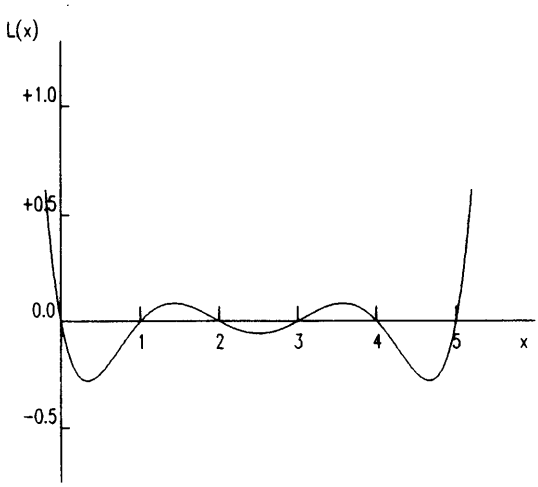


Figura 2.3 Distribución de $L(x)$

Ejemplo 2.2

Una tabla de valores para $f(x) = \log_{10}(x)$ es la siguiente:

i	x_i	$f(x_i)$
0	1	0
1	2	0.30103
2	3	0.47712
3	4	0.60206

Si la función se aproxima mediante la interpolación de Lagrange que se ajusta a estos datos, estime los errores en $x = 1.5, 2.5, 3.5$.

(Solución)

La estimación del error dada por la ecuación (2.3.9) es

$$e(x) \simeq \frac{(x-1)(x-2)(x-3)(x-4)f''''(2.5)}{(4!)}$$

La cuarta derivada de $f(x)$ es

$$f''''(x) = \left(\frac{d}{dx}\right)^4 \log_{10}(x) = \frac{\left(\frac{d}{dx}\right)^4 \log_e(x)}{\log_e(10)}$$

$$= \frac{-6}{x^4 \log_e(10)}$$

Así, se obtienen los siguientes $e(x)$ para $x = 1.5$, 2.5 y 3.5, los cuales se comparan con los valores exactos:

x	$e(x)$	Exacto
1.5	0.0026	0.0053
2.5	−0.0015	−0.0021
3.5	0.0026	0.0026

Es bueno que coincidan las estimaciones de los errores con sus valores exactos. Las discrepancias entre las estimaciones y los valores exactos surgen de la aproximación de $f''''(\xi)$ mediante $f''''(2.5)$.

A menudo es necesario escribir una fórmula de interpolación en la forma de serie de potencias. Por desgracia, la fórmula de interpolación de Lagrange no es adecuada para obtener una forma en serie de potencias, ya que es molesto desarrollar la interpolación de Lagrange en una serie de potencias. Un mejor enfoque —en particular cuando los puntos de la retícula tienen un espaciamiento uniforme —es el de utilizar la fórmula de interpolación de Newton hacia adelante y transformarla a la forma de series de potencias por medio de los coeficientes de Markov (véanse las secciones 2.5 y 7.3 para mayores detalles).

Resumen de Esta Sección

a) Un polinomio de orden N ajustado a $N + 1$ datos es único.

b) El polinomio de interpolación se puede expresar en varias formas distintas, entre las que estudiamos la forma de series de potencias y la fórmula de interpolación de Lagrange. Ambas son matemáticamente iguales debido al teorema de unicidad.

c) Cuando las abscisas de los datos tienen un espaciamiento uniforme, la amplitud de la oscilación del error de la interpolación tiende a ser más pequeña en el centro o cerca de él. La amplitud de oscilación aumenta hacia los extremos.

d) Si una función se aproxima mediante un polinomio de interpolación, no hay garantía de que dicho polinomio converja a la función exacta al aumentar el número de datos. En general, la interpolación mediante un polinomio de orden grande debe evitarse o utilizarse con precauciones extremas.
e) Aunque no existe un criterio para determinar el orden óptimo del polinomio de interpolación, generalmente se recomienda utilizar uno con orden relativamente bajo en un pequeño rango de x.

2.4 INTERPOLACIONES DE NEWTON HACIA ADELANTE Y HACIA ATRAS EN PUNTOS CON IGUAL SEPARACION

En la sección anterior, estudiamos la fórmula de interpolación de Lagrange y analizamos su error en puntos con igual separación. Esta fórmula es adecuada tanto para puntos con igual separación, como para los que no tienen ese espaciamiento. Sin embargo, las desventajas de la interpolación de Lagrange son las siguientes:

a) La cantidad de cálculos necesaria para una interpolación es grande.
b) La interpolación para otro valor de x necesita la misma cantidad de cálculos adicionales, ya que no se pueden utilizar partes de la aplicación previa.
c) Cuando el número de datos tiene que incrementarse o decrementarse, no se pueden utilizar los resultados de los cálculos previos.
d) La evaluación del error no es fácil.

El uso de las fórmulas de interpolación de Newton salva estas dificultades.

Para escribir una interpolación de Newton para un conjunto dado de datos se tiene que desarrollar una tabla de diferencias. Una vez hecho esto, las fórmulas de interpolación que pasan por distintos conjuntos de datos consecutivos como $i = 0, 1, 2, 3$; $i = 3, 4, 5, 6$ o $i = 2, 3, 4$, etc.) se pueden escribir con mucha facilidad. Por lo tanto, el orden de un polinomio de interpolación se puede incrementar rápidamente con datos adicionales. El error de la fórmula de interpolación de Newton también se puede estimar con comodidad. La interpolación de Newton es más adecuada que la interpolación de Lagrange para obtener otros modelos numéricos —por ejemplo las aproximaciones de derivada por diferencias (véase la sección 5.4)— o para desarrollar una interpolación por medio de una serie de potencias (véase la sección 7.3).

En el resto de esta sección se analizan dos versiones de la interpolación de Newton (hacia adelante y hacia atrás). Ambas son matemáticamente equivalentes pero representan expresiones distintas. Una puede ser más conveniente que la otra, dependiendo de cómo se aplique la fórmula. Por ejemplo, se prefiere la segunda al obtener el método predictor-corrector en la sección 9.4, puesto que todos los datos están en posiciones hacia atrás. Sin embargo, al ajustar los datos dados en una tabla, a menudo es más conveniente la interpolación de Newton hacia adelante.

2.4.1 Tabla de diferencias hacia adelante y coeficientes binomiales

Supondremos que las abscisas de los datos tienen igual separación con un tamaño de intervalo h. Los puntos se denotarán por (x_i, f_i).

Para evaluar una fórmula de interpolación de Newton hacia adelante, son necesarios una tabla de diferencias hacia adelante y los coeficientes binomiales [Gerald/Wheatley]. Por lo tanto, primero definimos las diferencias hacia adelante como

$$\Delta^0 f_i = f_i \qquad \text{(diferencia hacia adelante de orden cero)} \tag{2.4.1}$$

$$\Delta f_i = f_{i+1} - f_i \qquad \text{(diferencia hacia adelante de orden uno)} \tag{2.4.2}$$

$$\Delta^2 f_i = \Delta f_{i+1} - \Delta f_i \qquad \text{(diferencia hacia adelante de orden dos)} \tag{2.4.3}$$

$$\Delta^3 f_i = \Delta^2 f_{i+1} - \Delta^2 f_i \qquad \text{(diferencia hacia adelante de orden tres)} \tag{2.4.4}$$

$$\vdots$$

$$\Delta^k f_i = \Delta^{k-1} f_{i+1} - \Delta^{k-1} f_i \qquad \text{(diferencia hacia adelante de orden } k\text{)} \tag{2.4.5}$$

Una diferencia de orden superior se puede obtener fácilmente utilizando el operador de desplazamiento (véase el apéndice C para los detalles).

La tabla de diferencias ilustrada en la tabla 2.2, es un medio conveniente para evaluar las diferencias para un conjunto dado de datos. En la tabla 2.2, la primera columna es el índice de los datos, la segunda son las ordenadas de los datos. La tercer columna lista las diferencias de primer orden calculadas a partir de la segunda columna. La cuarta columna muestra las diferencias de segundo orden calculadas a partir de la columna anterior, etc. Cada renglón proporciona un conjunto de diferencias hacia adelante de los puntos correspondientes.

Tabla 2.2 Tabla de diferencias

i	f_i	Δf_i	$\Delta^2 f_i$	$\Delta^3 f_i$	$\Delta^4 f_i$	$\Delta^5 f_i$
0	f_0	Δf_0	$\Delta^2 f_0$	$\Delta^3 f_0$	$\Delta^4 f_0$	$\Delta^5 f_0$
1	f_1	Δf_1	$\Delta^2 f_1$	$\Delta^3 f_1$	$\Delta^4 f_1$	
2	f_2	Δf_2	$\Delta^2 f_2$	$\Delta^3 f_2$		
3	f_3	Δf_3	$\Delta^2 f_3$			
4	f_4	Δf_4				
5	f_5					

Uno debe saber lo siguiente acerca de la tabla de diferencias: Si f_i se toma como $f_i = f(x_i)$ (donde $f(x)$ es un polinomio de orden digamos L, y los x_i tienen igual separación), entonces la columna para la diferencia de orden L se convierte en una constante y la siguiente columna ($[L + 1]$-ésima diferencia) se anula. Si esto ocurre, sabemos que los datos pertenecen a un polinomio de orden L. Sin embargo, si una columna de diferencias tiene uno o más valores anormalmente grandes, es probable que existan algunos errores humanos en el proceso de desarrollo de la tabla o en el conjunto de datos.

Los coeficientes binomiales están dados por

$$\binom{s}{0} = 1$$

$$\binom{s}{1} = s$$

$$\binom{s}{2} = \frac{1}{2!}\, s(s-1)$$

$$\binom{s}{3} = \frac{1}{3!}\, s(s-1)(s-2)$$

$$\vdots$$

$$\binom{s}{n} = \frac{1}{n!}\, s(s-1)(s-2)\cdots(s-n+1)$$

donde s es una coordenada local definida por $s = (x - x_0)/h$ y h es el intervalo uniforme de la retícula.

Ejemplo 2.3

Desarrolle una tabla de diferencias hacia adelante para el conjunto de datos dado a continuación

i	0	1	2	3	4	5	6
x_i	0.1	0.3	0.5	0.7	0.9	1.1	1.3
$f(x_i)$	0.99750	0.97763	0.93847	0.88120	0.80752	0.71962	0.62009

(Solución)

La tabla de diferencias hacia adelante es como sigue:

i	x_i	f_i	Δf_i	$\Delta^2 f_i$	$\Delta^3 f_i$	$\Delta^4 f_i$	$\Delta^5 f_i$	$\Delta^6 f_i$
0	0.1	0.99750	−0.01987	−0.01929	0.00118	0.00052	−0.00003	−0.00006
1	0.3	0.97763	−0.03916	−0.01811	0.00170	0.00049	−0.00009	
2	0.5	0.93847	−0.05727	−0.01641	0.00219	0.00040		
3	0.7	0.88120	−0.07368	−0.01422	0.00259			
4	0.9	0.80752	−0.08790	−0.01163				
5	1.1	0.71962	−0.09953					
6	1.3	0.62009						

Comentario: las diferencias de orden superior tienden a anularse pero quizá no lleguen a valer exactamente cero. A menudo la causa se debe a errores de redondeo en los datos. Así, esto puede ocurrir aunque los datos pertenezcan a un polinomio de orden bajo.

2.4.2 La fórmula de interpolación de Newton hacia adelante

La fórmula de interpolación de Newton hacia adelante que pasa por $k + 1$ puntos, $f_0, f_1, f_2, \ldots, f_k$, se escribe como

$$g(x) = g(x_0 + sh) = \sum_{n=0}^{k} \binom{s}{n} \Delta^n f_0 \tag{2.4.6}$$

Por ejemplo, cuando $k = 2$, la ecuación (2.4.6) es

$$g(x_0 + sh) = f_0 + s(f_1 - f_0) + \frac{s(s-1)}{2}(f_2 - 2f_1 + f_0) \tag{2.4.7a}$$

o equivalentemente

$$g(x_0 + sh) = f_0 + (sh)\frac{-f_2 + 4f_1 - 3f_0}{2h} + \frac{(sh)^2}{2}\,\frac{f_2 - 2f_1 + f_0}{h^2} \tag{2.4.7b}$$

La ecuación (2.4.6) es un polinomio de orden k ya que $\binom{s}{n}$ es un polinomio de orden n, y su máximo orden es k. La ecuación (2.4.6) es igual a $f_0, f_1, f_2, \ldots, f_k$ en $x = x_0, x_1, \ldots, x_k$, respectivamente, lo cual se muestra a continuación:

$$\begin{aligned}
s = 0&: \quad g(x_0) = g(x_0 + 0) = f_0 \\
s = 1&: \quad g(x_1) = g(x_0 + h) = f_0 + \Delta f_0 = f_1 \\
s = 2&: \quad g(x_2) = g(x_0 + 2h) = f_0 + 2\Delta f_0 + \Delta^2 f_0 = f_2 \\
&\vdots \\
s = k&: \quad g(x_k) = g(x_0 + kh) = f_0 + k\Delta f_0 + \frac{k(k-1)}{2}\Delta^2 f_0 + \ldots = f_k
\end{aligned} \tag{2.4.8}$$

Los primeros $m + 1$ términos de la ecuación (2.4.6) forman un polinomio de interpolación de orden m ajustado a los $m + 1$ puntos en $x_0, x_1, x_2, \ldots, x_m$. De la misma forma, los primeros $m + 2$ términos forman un polinomio de interpolación de orden $m + 1$ ajustado a $m + 2$ puntos. Así, el orden de un polinomio de interpolación se puede cambiar fácilmente modificando el número de diferencias que se toman del primer renglón de la tabla 2.2.

Si se remplazan x_0 y f_0 de la ecuación (2.4.6) por x_2 y f_2, respectivamente, la ecuación se convierte en

$$g(x_2 + sh) = \sum_{n=0}^{k} \binom{s}{n} \Delta^n f_2 \tag{2.4.9}$$

donde s se define como $s = (x - x_2)/h$, que es una coordenada local. El valor s se vuelve 0 en $x = x_2$, y 1, 2, 3, ... en $x = x_3, x_4, x_5, \ldots$ respectivamente. La ecuación (2.4.9) es un polinomio de orden k ajustado a $x_2, x_3, \ldots, x_{k+2}$ y utiliza las diferencias del tercer renglón de la tabla 2.2; esto ilustra que, una vez desarrollada una tabla

de diferencias como la tabla 2.2, se pueden obtener sin problemas las fórmulas de interpolación que se ajustan a distintos conjuntos de datos.

Ejemplo 2.4

Obtenga los polinomios de interpolación de Newton hacia adelante ajustados a los datos en a) $i = 0, 1, 2$, b) $i = 0, 1, 2, 3, 4$, c) $i = 2, 3, 4$ y d) $i = 4, 5, 6$ dados en la siguiente tabla

i	0	1	2	3	4	5	6
x_i	0.1	0.3	0.5	0.7	0.9	1.1	1.3
$f(x_i)$	0.99750	0.97763	0.93847	0.88120	0.80752	0.71962	0.62009

(Solución)

La tabla de diferencias se desarrolló en el ejemplo 2.3.

a) La interpolación de Newton hacia adelante que pasa por los puntos $i = 0, 1, 2$ se obtiene utilizando los tres valores del renglón correspondiente a $i = 0$ en la tabla de diferencias del ejemplo 2.3 y se escribe como

$$y = 0.99750 - 0.01987s - \frac{0.01929}{2}s(s-1)$$

$$s = \frac{x - x_0}{h}$$

b) $$y = 0.99750 - 0.01987s - \frac{0.01929}{2}s(s-1) + \frac{0.00118}{6}s(s-1)(s-2) + \frac{0.00052}{24}s(s-1)(s-2)(s-3)$$

$$s = \frac{x - x_0}{h}$$

c) $$y = 0.93847 - 0.05727s - \frac{0.01641}{2}s(s-1)$$

$$s = \frac{x - x_2}{h}$$

d) $$y = 0.80752 - 0.08790s - \frac{0.01163}{2}s(s-1)$$

$$s = \frac{x - x_4}{h}$$

Debido a la equivalencia entre las fórmulas de interpolación de Newton y las de interpolación de Lagrange el error del polinomio de interpolación de Newton debe ser idéntico al de la fórmula de interpolación de Lagrange. Así, se puede escribir como

$$e(x) = f(x) - g(x) = L(x)f^{(N+1)}(\xi) \quad x_0 < \xi < x_N \tag{2.4.10}$$

donde $f(x)$ es la función exacta y $g(x)$ es la interpolación de Newton; sin embargo, la evaluación de la ecuación (2.4.10) para la interpolación de Newton es mucho más fácil que para la interpolación de Lagrange.

Consideremos la ecuación (2.4.6) con $k = N$. Si k se incrementa de N a $N + 1$, el término adicional es

$$\begin{aligned}\binom{s}{N+1}\Delta^{N+1}f_0 &= \frac{s(s-1)(s-2)\cdots(s-N)}{(N+1)!}\Delta^{N+1}f_0 \\ &= \frac{(x-x_0)(x-x_1)\cdots(x-x_N)}{(N+1)!} \times \frac{\Delta^{N+1}f_0}{h^{N+1}}\end{aligned} \tag{2.4.11}$$

donde se utilizan $s = (x - x_0)/h$ y $x_n = x_0 + nh$. Se puede mostrar que el segundo término de la ecuación (2.4.11) es una aproximación de $f(N + 1)$, a saber

$$\Delta^{N+1}f_0/h^{N+1} \simeq f^{(N+1)}(x_m)$$

donde

$$x_m = \frac{1}{2}(x_0 + x_N)$$

Por lo tanto, la ecuación (2.4.11) es aproximadamente igual al lado derecho de la ecuación (2.4.10). Es decir, el error está representado por el siguiente término que aparece, si el orden del polinomio se incrementa en uno con un punto adicional x_{N+1}.

¿Qué podemos hacer si no disponemos del siguiente punto? En este caso, hay que verificar si se dispone de un punto adicional del otro lado, a saber, $f(x_{-1})$. Si está disponible, se puede calcular, $\Delta^{N+1}f_{-1}$ y utilizarla como una aproximación de $\Delta^{N+1}f_0$.

Ejemplo 2.5

Evalúe el error de la ecuación a) del ejemplo 2.4 para $x = 0.2$.

(Solución)

La ecuación a) del ejemplo 2.4 se ajusta a $i = 0, 1, 2$. Así, el término adicional que proviene del ajuste de la interpolación en $i = 3$ es

$$\frac{0.00118}{6}s(s-1)(s-2)$$

Por lo tanto, al introducir $s = (x - x_0)/h = (x - 0.1)/0.2 = 0.5$ para $x = 0.2$, el error es

$$e(x) \simeq \frac{0.00118}{6}s(s-1)(s-2) = 7.4 \times 10^{-5}$$

Compare esto con el error real, 4.4×10^{-5}.

2.4.3 Interpolación de Newton hacia atrás

El polinomio de interpolación de Newton hacia atrás es otra fórmula de uso frecuente y se escribe en términos de las diferencias hacia atrás y los coeficientes binomiales. Consideramos puntos con igual separación $x_0, x_{-1}, x_{-2}, \ldots, x_{-k}$ con un espacio constante igual a $h = x_i - x_{i-1}$.

Las diferencias hacia atrás se definen como

$$\nabla^0 f_i = f_i \qquad \text{(diferencia hacia atrás de orden cero)}$$

$$\nabla f_i = f_i - f_{i-1} \qquad \text{(diferencia hacia atrás de orden uno)}$$

$$\nabla^2 f_i = \nabla f_i - \nabla f_{i-1} \qquad \text{(diferencia hacia atrás de orden dos)}$$

$$\nabla^3 f_i = \nabla^2 f_i - \nabla^2 f_{i-1} \qquad \text{(diferencia hacia atrás de orden tres)}$$

$$\vdots$$

$$\nabla^k f_i = \nabla^{k-1} f_i - \nabla^{k-1} f_{i-1} \qquad \text{(diferencia hacia atrás de orden } k\text{)}$$

Se puede desarrollar una tabla de diferencias hacia atrás, como se muestra en el ejemplo 2.6.

Ejemplo 2.6

Elabore una tabla de diferencias hacia atrás para el mismo conjunto de datos dados en el ejemplo 2.4:

i	0	1	2	3	4	5	6
x_i	0.1	0.3	0.5	0.7	0.9	1.1	1.3
$f(x_i)$	0.99750	0.97763	0.93847	0.88120	0.80752	0.71962	0.62009

(Solución)

La tabla de diferencias hacia atrás es como sigue:

i	x_i	f_i	∇f_i	$\nabla^2 f_i$	$\nabla^3 f_i$	$\nabla^4 f_i$	$\nabla^5 f_i$	$\nabla^6 f_i$
0	0.1	0.99750						
1	0.3	0.97763	−0.01987					
2	0.5	0.93847	−0.03916	−0.01929				
3	0.7	0.88120	−0.05727	−0.01811	0.00118			
4	0.9	0.80752	−0.07368	−0.01641	0.00170	0.00052		
5	1.1	0.71962	−0.08790	−0.01422	0.00219	0.00049	−0.00003	
6	1.3	0.62009	−0.09953	−0.01163	0.00259	0.00040	−0.00009	−0.00006

Los coeficientes binomiales que se utilizan en las interpolaciones de Newton hacia atrás son los siguientes:

$$\binom{s-1}{0} = 1$$

$$\binom{s}{1} = s$$

$$\binom{s+1}{2} = \frac{1}{2!}(s+1)s$$

$$\binom{s+2}{3} = \frac{1}{3!}(s+2)(s+1)s$$

$$\vdots$$

$$\binom{s+n-1}{n} = \frac{1}{n!}(s+n-1)(s+n-2)\cdots(s+1)s$$

La interpolación de Newton hacia atrás ajustada a los puntos en $x = x_0$, $x = x_{-1}$, $x = x_{-2}, \ldots$ y $x = x_{-k}$ se escribe como

$$g(x) = g(x_i + sh) = \sum_{n=0}^{k} \binom{s+n-1}{n} \nabla^n f_i, \quad -k \leqslant s \leqslant 0 \tag{2.4.12}$$

donde s es una coordenada local definida por $s = (x - x_i)/h$; $\binom{s+n-1}{n}$ es un coeficiente binomial y $\nabla^n f_i$ es la diferencia hacia atrás.

Una relación de equivalencia entre la diferencia hacia adelante y la diferencia hacia atrás está dada por

$$\nabla^n f_i = \Delta^n f_{i-n} \tag{2.4.13}$$

Por lo tanto, la ecuación (2.4.12) se puede expresar en términos de las diferencias hacia adelante como

$$g(x) = \sum_{n=0}^{k} \binom{s+n-1}{n} \Delta^n f_{i-n}, \quad -k \leqslant s \leqslant 0 \tag{2.4.14}$$

o en forma más explícita,

$$\begin{aligned} g(x) = g(x_i + sh) &= f_i + s(f_i - f_{i-1}) + \frac{1}{2}(s+1)s(f_i - 2f_{i-1} + f_{i-2}) \\ &+ \frac{1}{6}(s+2)(s+1)s(f_i - 3f_{i-1} + 3f_{i-2} - f_{i-3}) + \cdots \\ &+ \frac{1}{k!}(s+k-1)(s+k-2)\cdots(s+1)s\Delta^k f_{i-k} \end{aligned} \tag{2.4.15}$$

Ejemplo 2.7

Determine los polinomios de interpolación de Newton hacia atrás ajustado a los tres puntos $i = 3, 4, 5$, en la tabla de valores del ejemplo 2.6.

(Solución)

Debido a que el número de puntos es 3, el orden del polinomio es 2. El polinomio de interpolación de Newton hacia atrás dado por la ecuación (2.4.12) es, en este caso,

$$g(x) = g(x_5 + sh) = \sum_{n=0}^{2} \binom{s+n-1}{n} \nabla^n f_5$$

$$= f_5 + s\nabla f_5 + \frac{1}{2}(s+1)s\nabla^2 f_5, \quad -2 \leqslant s \leqslant 0$$

donde $s = (x - x_5)/h$. Utilizando los valores de f_5, ∇f_5 y $\nabla^2 f_5$ en las tablas de diferencias del ejemplo 2.6, la ecuación anterior se convierte en

$$g(x) = 0.71962 - 0.08790s - \frac{0.01422}{2}(s+1)s$$

o, en forma equivalente, si se emplea $s = (x - x_5)/h$,

$$g(x) = 0.71962 - \frac{0.08790(x_5 - x)}{h} - \frac{0.00711(x_5 - x)(x_4 - x)}{h^2}$$

Resumen de Esta Sección

a) Los coeficientes de las interpolaciones de Newton hacia adelante y hacia atrás se evalúan desarrollando una tabla de diferencias.
b) La L-ésima columna de diferencias en la tabla llega a ser constante y la $(L + 1)$-ésima columna de diferencias es nula si el conjunto de los datos pertenece a un polinomio de orden L. Sin embargo, debido a los errores de redondeo en el conjunto de datos, la diferencia de orden $(L + 1)$ puede no anularse exactamente.
c) Los polinomios de interpolación de Newton son iguales a la fórmula de interpolación de Lagrange si se utiliza el mismo conjunto de datos.
d) El error en la fórmula de interpolación de Newton se representa mediante el término adicional que resulta de un punto extra de los datos.

2.5 INTERPOLACION DE NEWTON EN PUNTOS CON SEPARACION NO UNIFORME

Las fórmulas de interpolación de Newton descritas en la sección anterior se restringen a puntos con igual separación. Sin embargo, a menudo aparece la necesidad de escribir un polinomio de interpolación para puntos con separación no uniforme. El

modelo de interpolación de Newton puede extenderse a los puntos con separación no uniforme utilizando las diferencias divididas [Isaacson/Keller; Carnahan/Luther/Wilkes]. Así, el polinomio de interpolación de Lagrange en una retícula con espaciamiento no uniforme se puede expresar de manera equivalente en la forma de un polinomio de interpolación de Newton.

Denotemos al polinomio de interpolación de Lagrange ajustando a x_0, x_1, $x_2, \ldots, x_m$ como

$$P_{0,1,2,\ldots,m}(x)$$

y el ajustado a $x_1, x_2, \ldots, x_{m+1}$

$$P_{1,2,3,\ldots,m+1}(x)$$

El número de subíndices de $P_{a,b,c,\ldots,j}$ menos uno es el orden del polinomio de interpolación, por lo que los dos polinomios dados son de orden m. Entonces, es obvio que el polinomio ajustado a $x_0, x_1, \ldots, x_{m+1}$ está dado por

$$P_{0,1,2,\ldots,m+1}(x) = \frac{(x - x_0)P_{1,2,\ldots,m+1}(x) + (x_{m+1} - x)P_{0,1,2,\ldots,m}(x)}{x_{m+1} - x_0} \tag{2.5.1}$$

Si la ecuación (2.5.1) se desarrolla en una serie de potencias, el coeficiente del término de mayor orden se llama *coeficiente principal*. El coeficiente principal de $P_{a,b,c,\ldots,j}$ utilizando los mismos subíndices. Al aplicar esta regla, el coeficiente dominante de $P_{0,1,2,\ldots,m+1}$ es $f_{0,1,2,\ldots,m+1}$. De manera similar, los coeficientes dominantes de $P_{0,1,2,\ldots,m}$ y $P_{1,2,\ldots,m+1}$ son, respectivamente, $f_{0,1,2,\ldots,m}$ y $f_{1,2,\ldots,m+1}$. Mediante la inspección de la ecuación (2.5.1), su coeficiente dominante para el lado izquierdo se relaciona con los de los dos polinomios de interpolación del lado derecho mediante

$$f_{0,1,2,\ldots,m+1} = \frac{f_{1,2,\ldots,m+1} - f_{0,1,2,\ldots,m}}{x_{m+1} - x_0} \tag{2.5.2}$$

La ecuación (2.5.2) es una "diferencia dividida" de orden $m + 1$, ya que está dada por la diferencia de los coeficientes dominantes de orden m dividida entre la distancia de los puntos más exteriores. Por medio de la ecuación (2.5.2), se pueden calcular en forma recursiva los coeficientes dominantes —a partir de una tabla de valores— como se muestra de manera simbólica en la tabla 2.3.

Por medio de la diferencia, se puede obtener —un polinomio de interpolación, por ejemplo, $P_{a,b,c,\ldots,d}$— como

$$\begin{aligned} P_{a,b,c,\ldots,j}(x) = f_a &+ f_{a,b}(x - x_a) + f_{a,b,c}(x - x_a)(x - x_b) + \cdots \\ &+ f_{a,b,c,\ldots,j}(x - x_a)(x - x_b)(x - x_c)\cdots(x - x_{j-1}) \end{aligned} \tag{2.5.3}$$

El error de la interpolación se evalúa de la misma forma que la interpolación de Newton en una malla igualmente espaciada. En realidad, el error de la ecuación

Tabla 2.3 Tabla de diferencias divididas

x_0	f_0	$f_{0,1} = \dfrac{f_1 - f_0}{x_1 - x_0}$	$f_{0,1,2} = \dfrac{f_{1,2} - f_{0,1}}{x_2 - x_0}$	$f_{0,1,2,\ldots,m+1} = \dfrac{f_{1,2,\ldots,m+1} - f_{0,1,\ldots,m}}{x_{m+1} - x_0}$
x_1	f_1	$f_{1,2} = \dfrac{f_2 - f_1}{x_2 - x_1}$	$f_{1,2,3} = \dfrac{f_{2,3} - f_{1,2}}{x_3 - x_1}$	
x_2	f_2	$f_{2,3} = \dfrac{f_3 - f_2}{x_3 - x_2}$	$f_{2,3,4} = \dfrac{f_{3,4} - f_{2,3}}{x_4 - x_2}$	
x_m	f_m	$f_{m,m+1} = \dfrac{f_{m+1} - f_m}{x_{m+1} - x_m}$		

(2.5.3) es aproximadamente igual al término que se añadiría a la ecuación (2.5.3) si la interpolación se extendiera para ajustarse a otro punto más, $j + 1$; es decir, el error es

$$e(x) \simeq f_{a,b,c,\ldots,j,j+1}(x - x_a)(x - x_b)(x - x_c)\cdots(x - x_j)$$

Ejemplo 2.8

a) Elabore una tabla de diferencias divididas para los siguientes datos:

i	x_i	f_i
0	0.1	0.99750
1	0.2	0.99002
2	0.4	0.96040
3	0.7	0.88120
4	1.0	0.76520
5	1.2	0.67113
6	1.3	0.62009

b) Escriba la fórmula de interpolación utilizando la tabla de diferencias divididas ajustada a los puntos: 1) $i = 0$ hasta 6 y 2) $i = 2$ hasta 4.

c) Obtenga una estimación del error de las interpolaciones.

d) Evalúe los polinomios de interpolación en $x = 0.3$ y $x = 0.55$.

e) Estime los errores de la interpolación para los dos valores del inciso d) y compárelos con los valores exactos de los errores. (Los valores exactos son $f(0.3) = 0.97763$ y $f(0.55) = 0.92579$, respectivamente.)

(Solución)

En primer término, la tabla de diferencias divididas se elabora.

i	x_i	f_i	$f_{i,i+1}$	$f_{i,\ldots,i+2}$	$f_{i,\ldots,i+3}$	$f_{i,\ldots,i+4}$	$f_{i,\ldots,i+5}$	$f_{i,\ldots,i+6}$
0	0.1	0.99750	−0.07480	−0.24433	0.02088	0.01478	−0.00236	0.00122
1	0.2	0.99002	−0.14810	−0.23180	0.03418	0.01218	−0.00090	
2	0.4	0.96040	−0.26400	−0.20445	0.04636	0.01119		
3	0.7	0.88120	−0.38667	−0.16736	0.05643			
4	1.0	0.76520	−0.47035	−0.13350				
5	1.2	0.67113	−0.51040					
6	1.3	0.62009						

i) El polinomio ajustado de $i = 0$ hasta 5 es entonces

$$\begin{aligned} P_{0,1,\ldots,5}(x) = 0.99750 &- 0.07480(x - 0.1) - 0.24433(x - 0.1)(x - 0.2) \\ &+ 0.02088(x - 0.1)(x - 0.2)(x - 0.4) \\ &+ 0.01478(x - 0.1)(x - 0.2)(x - 0.4)(x - 0.7) \\ &- 0.00236(x - 0.1)(x - 0.2)(x - 0.4)(x - 0.7)(x - 1) \end{aligned}$$

Una estimación del error de esta interpolación es

$$e(x) = 0.00122(x - 0.1)(x - 0.2)(x - 0.4)(x - 0.7)(x - 1)(x - 1.2)$$

Los resultados calculados se resumen abajo:

	$P_{0,1,\ldots,5}$	Error (estimado)	Error (exacto)
$x = 0.3$	0.97762	$6.17E - 7$	$5.2E - 6$
$x = 0.55$	0.92580	$-1.26E - 7$	$-5.2E - 6$

Los errores mostrados antes requieren cierto examen. Los valores exactos de $f(0.3)$ y $f(0.55)$ se dan sólo hasta la quinta cifra decimal, por lo que están sujetos a errores de redondeo de a lo más $\pm 5.0\ E - 6$. Por lo tanto, los valores exactos que se muestran abajo no tienen significado, excepto para ilustrar el efecto del error de redondeo de una resta. Por otro lado, los valores estimados son menores que estos valores. Así, concluimos que los resultados de la interpolación son exactos dentro de los errores de redondeo.

ii) El polinomio ajustado de $i = 2$ hasta 4 es

$$P_{1,2,3,4}(x) = 0.96040 - 0.26400(x - 0.4) - 0.20445(x - 0.4)(x - 0.7)$$

El error estimado de esta interpolación es aproximadamente

$$e(x) = 0.04636(x - 0.4)(x - 0.7)(x - 1)$$

Los valores calculados se resumen abajo:

	$P_{1,2,3,4}$	Error (estimado)	Error (exacto)
$x = 0.3$	0.97862	-0.00130	-0.00099
$x = 0.55$	0.92540	0.00047	0.00039

Los errores estimados coinciden con los errores exactos.

Resumen de Esta Sección

a) La fórmula de interpolación de Newton con diferencias divididas es una variación de la interpolación de Newton hacia adelante para puntos con igual separación.
b) Se puede aplicar a puntos que tienen igual o distinta separación.
c) El error de la fórmula de interpolación se representa mediante el término adicional que proviene de un punto más de los datos.

2.6 INTERPOLACION CON RAICES DE CHEBYSHEV

Como se mencionó en la sección anterior, la interpolación polinomial que utiliza puntos con igual separación —ya sea que se exprese mediante la fórmula de interpo-

lación de Lagrange o a través de un polinomio de interpolación de Newton— es más precisa en el rango medio del dominio de interpolación, aunque el error de la interpolación crece hacia los extremos. Esto se atribuye al comportamiento de $L(x)$ en la ecuación (2.3.7).

El esquema descrito en esta sección determina los puntos mediante un polinomio de Chebyshev [Carnahan/et al.; Abramowitz/Stegun]. La separación determinada por un polinomio de Chebyshev es mayor en el centro del dominio de interpolación y decrece hacia los extremos. Como resultado, los errores se distribuyen de una forma más regular en todo el dominio y sus magnitudes son menores que en el caso de los puntos separados de manera uniforme. La interpolación con los puntos de Chebyshev se usa ampliamente en las subrutinas matemáticas al igual que en los cálculos numéricos generales.

Los polinomios de Chebyshev se pueden expresar de dos formas distintas pero equivalentes: una utiliza funciones coseno y la otra series de potencias. En la primera expresión, el polinomio de Chebyshev normalizado de orden K se define como

$$T_K(x) = \cos(K \cos^{-1}(x)), \quad -1 \leqslant x \leqslant 1 \tag{2.6.1}$$

Los polinomios de Chebyshev en la serie de potencias están dados por

$$\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_2(x) &= 2x^2 - 1 \\
T_3(x) &= 4x^3 - 3x \\
T_4(x) &= 8x^4 - 8x^2 + 1 \\
T_5(x) &= 16x^5 - 20x^3 + 5x \\
T_6(x) &= 32x^6 - 48x^4 + 18x^2 - 1
\end{aligned} \tag{2.6.2}$$

Los polinomios de Chebyshev de cualquier orden superior en la serie de potencias se pueden generar utilizando la relación recursiva,

$$T_j(x) = 2xT_{j-1}(x) - T_{j-2}(x) \tag{2.6.3}$$

La forma de coseno de los polinomios de Chebyshev en la ecuación (2.6.1) indican que el mínimo y máximo local en $-1 \leqslant x \leqslant 1$ son -1 y 1, respectivamente. Conviene observar también que todos los polinomios de Chebyshev valen 1 en $x = 1$ y $+1$ o -1 en $x = -1$, como se ilustra en la figura 2.4. Puesto que la función coseno se anula en $\pm\ \pi/2$, $\pm 3\ \pi/2, \ldots$, las raíces de un polinomio de Chebyshev de orden K satisfacen

$$K \cos^{-1}(x_n) = \left(K + \frac{1}{2} - n\right)\pi, \quad n = 1, 2, \ldots, K \tag{2.6.4}$$

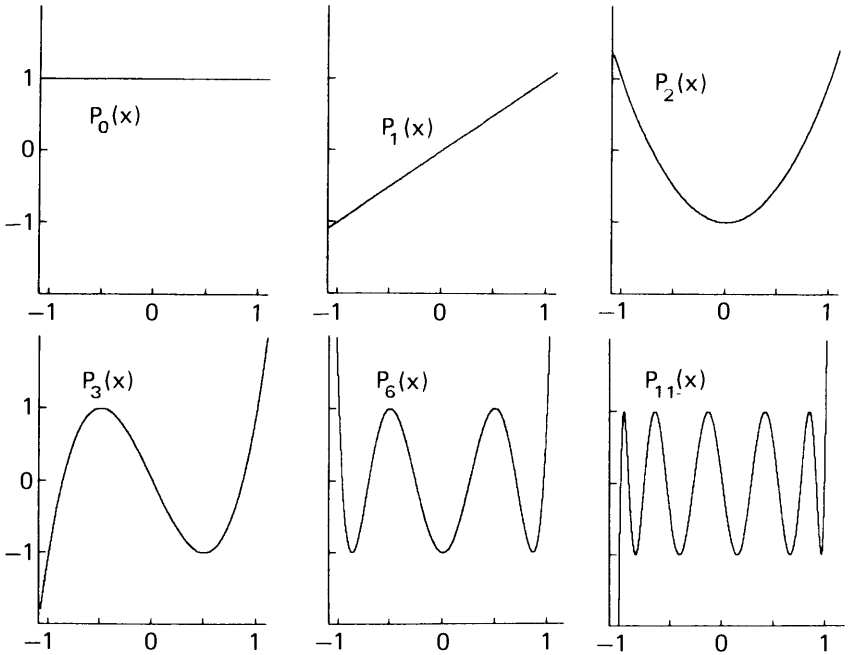


Figura 2.4 Polinomios de Chebyshev

o, más explícitamente,

$$x_n = \cos\left(\frac{K + 1/2 - n}{K}\pi\right), \quad n = 1, 2, \ldots, K \tag{2.6.5}$$

Si $K = 3$, por ejemplo, x_n para $n = 1, 2$ y 3 son -0.86602, 0, $+0.86602$, respectivamente.

Si el rango de interpolación es $[-1, 1]$, las K raíces x_n, $i = 1, 2, \ldots, K$, se pueden utilizar como las abscisas de los puntos en la interpolación de Lagrange, en vez de utilizar puntos con igual separación. Sin embargo, hay que observar que la numeración de los puntos al obtener los puntos de Chebyshev y la de la fórmula de interpolación de Lagrange de la ecuación (2.3.3) son distintas. Si se utilizan los tres puntos de Chebyshev de $K = 3$ como se mostró en el párrafo anterior, el orden de la fórmula de interpolación de Lagrange es $N = 2$ y los puntos x_i en la ecuación (2.3.3) son $x_0 = -0.86602$, $x_1 = 0$ y $x_2 = +0.086602$. Las ordenadas de los extremos —a saber, en $x = -1$ y $x = +1$ – no se utilizan. Por lo tanto, la fórmula de interpolación de Lagrange se utilizará como "extrapolación" en $[-1, -0.86602]$, al igual que en $[+0.86602, +1]$.

La interpolación polinomial de Chebyshev se puede aplicar en cualquier rango distinto de $[-1, 1]$, si se transforma a $[-1, 1]$ sobre el rango de interés. Si escribimos el rango de interpolación como $[a, b]$, la transformación está dada por

$$x = \frac{2z - a - b}{b - a} \tag{2.6.6}$$

o, en forma equivalente,

$$z = \frac{(b-a)x + a + b}{2} \tag{2.6.7}$$

donde

$$-1 \leqslant x \leqslant 1 \quad \text{y} \quad a \leqslant z \leqslant b.$$

por lo tanto, al sustituir los puntos de Chebyshev x_n en $[-1, 1]$ dados por la ecuación (2.6.5) en la ecuación (2.6.7), los puntos de Chebyshev z_n en $[a, b]$ son

$$z_n = \frac{1}{2}\left[(b-a)\cos\left(\frac{K + \frac{1}{2} - n}{K}\pi\right) + a + b\right], \quad n = 1, 2, \ldots, K \tag{2.6.8}$$

El error de una interpolación que utiliza raíces de Chebyshev también está dado por la ecuación (2.3.7). Sin embargo, el comportamiento de $L(x)$, es diferente del que se obtiene con los puntos separados uniformemente. En realidad, el propio $L(x)$ es un polinomio de Chebyshev ya que pasa por las raíces del polinomio de Chebyshev. En consecuencia, el error de la interpolación con las raíces de Chebyshev está distribuido de manera más uniforme que con los puntos con igual separación. Sin embargo, la distribución real del error $e(x)$ se desvía del polinomio de Chebyshev, ya que depende de x.

Ejemplo 2.9

a) Obtenga los tres puntos de Chebyshev en $2 \leqslant z \leqslant 4$.

b) Por medio de los tres puntos de Chebyshev, escriba la fórmula de interpolación ajustada a $\ln(z)$.

(Solución)

a) Al sustituir $a = 2$, $b = 4$ y $K = 3$ en la ecuación (2.6.8) y hacer $n = 1, 2, 3$, se encuentran los puntos de Chebyshev como

$$z_1 = 2.13397$$
$$z_2 = 3$$
$$z_3 = 3.86602$$

b) Ahora hacemos una tabla de valores con los puntos de Chebyshev como sigue:

z	$y = \ln(z)$
2.13397	0.757984
3	1.098612
3.86602	1.352226

La fórmula de interpolación de Lagrange ajustada al conjunto de datos es

$$g(z) = \frac{(z-3)(z-3.86602)}{(2.13397-3)(2.13397-3.86602)}(0.757984)$$
$$+\frac{(z-2.13397)(z-3.86602)}{(3-2.13397)(3-3.86602)}(1.098612)$$
$$+\frac{(z-2.13397)(z-3)}{(3.86602-2.13397)(3.86602-3)}(1.352226)$$

Resumen de Esta Sección

a) Los puntos de Chebyshev son raíces de un polinomio de Chebyshev.
b) Un polinomio de Chebyshev de orden K proporciona K puntos de Chebyshev. La fórmula de interpolación de Lagrange que utiliza K puntos de Chebyshev es un polinomio de orden $K - 1$.
c) La función $L(x)$ que representa el error dado por la ecuación (2.3.9) se convierte entonces en un polinomio de Chebyshev de orden K.
d) Al utilizar puntos de Chebyshev en la interpolación de Lagrange, el error se distribuye de manera más uniforme que con los puntos de igual separación.

2.7 POLINOMIOS DE INTERPOLACION DE HERMITE

Los esquemas de interpolación polinomial examinados anteriormente en este capítulo no utilizan la información de la derivada de la propia función ajustada. Sin embargo, un polinomio se puede ajustar no sólo a los valores de la función sino también a las derivadas de los puntos. Los polinomios ajustados a los valores de la función y su derivada se llaman *polinomios de interpolación de Hermite* o *polinomios osculatrices* [Isaacson/Keller].

Supóngase que se conocen los puntos $x_0, x_1, \ldots, x_N$, y los valores de la función y de todas sus derivadas hasta de orden p ($f_i, f_i' \ldots, f_i^{(p)}$, $i = 0, 1, \ldots, N$). El número total de datos es $K = (p + 1)(N + 1)$. Un polinomio de orden $K - 1$, a saber,

$$g(x) = \sum_{j=0}^{K-1} a_j x^j \tag{2.7.1}$$

se puede ajustar a los K datos, donde a_j es un coeficiente. Al igualar la ecuación (2.7.1) con los datos, obtenemos un conjunto de $K = (p + 1)(N + 1)$ ecuaciones

$$\begin{aligned} g(x_i) &= f_i, \quad i = 0, 1, \ldots, N \\ g'(x_i) &= f_i', \quad i = 0, 1, \ldots, N \\ &\vdots \\ g^{(p)}(x_i) &= f_i^{(p)} \quad i = 0, 1, \ldots, N \end{aligned} \tag{2.7.2}$$

Los coeficientes se pueden determinar resolviendo la ecuación (2.7.2) en forma exacta si K es pequeño.

Una expresión alternativa, análoga a la fórmula de interpolación de Lagrange, se puede escribir como

$$g(x) = \sum_{i=0}^{N} \alpha_i(x) f_i + \sum_{i=0}^{N} \beta_i(x) f'_i + \cdots + \sum_{i=0}^{N} \theta_i(x) f_i^{(p)} \tag{2.7.3}$$

Aquí,

$$\alpha_i(x_j) = \delta_{i,j} \tag{2.7.4}$$

y todas las derivadas de $\alpha_i(x)$ se anulan para cada $x = x_i$; $\beta_i(x)$ y todas sus derivadas se anulan para cada $x = x_i$ excepto

$$\left[\frac{d}{dx}\beta_i(x)\right]_{x=x_j} = \delta_{i,j} \tag{2.7.5}$$

De manera semejante, $\theta_i(x)$ y todas sus derivadas se anulan para cada $x = x_j$ excepto

$$\left[\frac{d^p}{dx^p}\theta_i(x)\right]_{x=x_j} = \delta_{i,j} \tag{2.7.6}$$

En realidad, la ecuación (2.7.3) es una extensión de la fórmula de interpolación de Lagrange, Se reduce a la interpolación de Lagrange si no se ajusta a la derivada.

Ejemplo 2.10

Suponga que una tabla de valores contiene los valores de la función y su primera derivada. Para cada intervalo, obtenga un polinomio que se ajuste a los valores de la función y las primeras derivadas en los extremos de ese intervalo.

(Solución)

Para cada intervalo, el número total de datos es cuatro, por lo que el orden del polinomio es tres. El polinomio se llama *polinomio cúbico de Hermite*.

Consideremos un intervalo entre x_{i-1} y x_i, como se muestra en la figura E2.10. El polinomio cúbico que se ajusta a f_{i-1}, f_{i-1}, f_i y f_i se escribe como

$$y(t) = a + bt + ct^2 + et^3 \tag{A}$$

donde se utiliza una coordenada local $t = x - x_{i-1}$. Al ajustar la ecuación (A) a los datos dados se obtiene

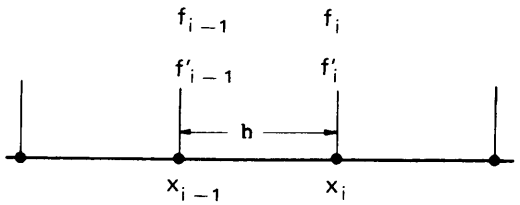


Figura E2.10 Un intervalo para la interpolación de Hermite

que da

$$f_{i-1} = a \tag{B}$$

$$f'_{i-1} = b \tag{C}$$

$$f'_i = a + bh + ch^2 + eh^3 \tag{D}$$

$$f'_i = b + 2ch + 3eh^2 \tag{E}$$

donde $h = x_i - x_{i-1}$. Al sustituir las ecuaciones (B) y (C) en las ecuaciones (D) y (E) y resolverlas en términos de c y e, se obtiene

$$c = \frac{3(f_i - f_{i-1}) - (f'_i + 2f'_{i-1})h}{h^2}$$
$$e = \frac{-2(f_i - f_{i-1}) + (f'_i + f'_{i-1})h}{h^3} \tag{F}$$

Así, el polinomio cúbico de interpolación de Hermite es

$$y(t) = f_{i-1} + f'_{i-1}t + [3(f_i - f_{i-1}) - (f'_{i-1} + 2f'_{i-1})h]\left(\frac{t}{h}\right)^2 + [-2(f_i - f_{i-1}) + (f'_i + f'_{i-1})h]\left(\frac{t}{h}\right)^3 \tag{G}$$

La ecuación (G) se puede expresar de manera equivalente como

$$y(t) = \alpha_{i-1}f_{i-1} + \alpha_i f_i + \beta_{i-1}f'_{i-1} + \beta_i f'_i \tag{H}$$

donde

$$\begin{aligned} \alpha_{i-1} &= 3(1-s)^2 - 2(1-s)^3 \\ \alpha_i &= 3s^2 - 2s^3 \\ \beta_{i-1} &= h[(1-s)^2 - (1-s)^3] \\ \beta_i &= h[s^2 - s^3] \end{aligned} \tag{I}$$

donde

$$s = \frac{t}{h} = \frac{x - x_{i-1}}{h}$$

Se puede mostrar fácilmente que $\alpha_i(x)$ vale uno para $x = x_i$ pero vale cero para $x = x_{i-1}$ y su primera derivada se anula tanto en x_{i-1}, su primera derivada vale uno en $x = x_i$ pero vale cero en $x = x_{i-1}$.

Resumen de Esta Sección

a) Una interpolación polinomial que se ajusta tanto a los valores de la función como a las derivadas, se llama *interpolación de Hermite*.

b) Se aplica una interpolación cúbica de Hermite en un intervalo en el que se especifican los valores de la función y los de la primera derivada en cada extremo. Si todo el dominio se divide en intervalos y el esquema se aplica a cada intervalo, el esquema global de interpolación se llama *interpolación cúbica de Hermite por partes*.

c) El valor de la función y la primera derivada en la interpolación cúbica de Hermite es continuo en el dominio de los enteros.

2.8 INTERPOLACION EN DOS DIMENSIONES

Los esquemas de interpolación en dos dimensiones se pueden clasificar en dos tipos. El primero utiliza dos veces la interpolación en dimensión uno y se llama *doble interpolación*. El segundo emplea polinomios de interpolación por partes en dos dimensiones. El primer tipo es adecuado para interpolar una tabla de valores de funciones en puntos con igual separación. El segundo se usa en los métodos de elemento finito [Becker/Carey/Oden].

Para explicar el primer tipo, supongamos que se conocen los valores de la función $f(x, y)$ en una malla rectangular de (x, y); a saber, (x_i, y_k). Denotamos el valor en el punto (x_i, y_j) como $f_{i,j} = f(x_i, y_j)$. La doble interpolación se lleva a cabo en dos etapas, en las que se utiliza una interpolación en dimensión uno. Suponga que hay que estimar el valor de la función en un punto localizado en el rectángulo definido $x_{i-1} \leqslant x \leqslant x_i$ y $y_{j-1} \leqslant y \leqslant y_j$, como lo muestra la figura 2.5. Para simplificar la explicación, supongamos que se utiliza la interpolación lineal en ambas etapas. La primera consiste en interpolar la tabla en la dirección de y y encontrar los valores en E y F, respectivamente, como

$$
\begin{aligned}
f_E &= \frac{y_j - y}{y_j - y_{j-1}} f_{i-1,j-1} + \frac{y - y_{j-1}}{y_j - y_{j-1}} f_{i-1,j} \\
f_F &= \frac{y_j - y}{y_j - y_{j-1}} f_{i,j-1} + \frac{y - y_{j-1}}{y_j - y_{j-1}} f_{i,j}
\end{aligned}
\tag{2.8.1}
$$

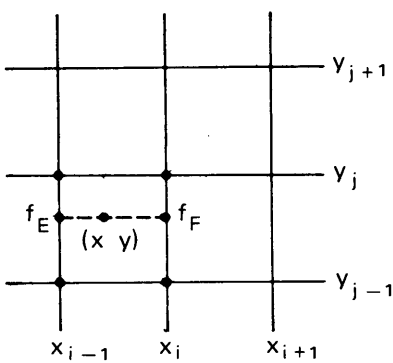


Figura 2.5 Interpolación bilineal en un dominio de dos dimensiones

La segunda etapa es interpolar entre f_E y f_F, mediante interpolación lineal, como

$$g(x, y) = \frac{x_i - x}{x_i - x_{i-1}} f_E + \frac{x - x_{i-1}}{x_i - x_{i-1}} f_F \tag{2.8.2}$$

Podemos combinar los dos pasos en una ecuación y escribir

$$\begin{aligned} g(x, y) = {} & [(x_i - x)(y_j - y)f_{i-1,j-1} + (x_i - x)(y - y_{j-1})f_{i-1,j} \\ & + (x - x_{i-1})(y_j - y)f_{i,j-1} + (x - x_{i-1})(y - y_{j-1})f_{i,j}]/ \\ & [(x_i - x_{i-1})(y_j - y_{j-1})] \end{aligned} \tag{2.8.3}$$

Se puede intercambiar el orden de las etapas de interpolación lineal. A saber, primero se hallan f_G y f_H mediante la interpolación lineal en la dirección de x y después se aplica una interpolación lineal en la dirección de y para calcular $g(x, y)$. El cambio no afecta el resultado.

Véase el apéndice H para analizar otra interpolación en dimensión dos llamada interpolación transfinita.

Resumen de Esta Sección

a) La interpolación en dos dimensiones se puede llevar a cabo de dos formas. En el primer enfoque, se aplica dos veces la interpolación en dimensión uno. En el segundo, se puede ajustar de manera directa un polinomio en dos dimensiones a los valores de la función.
b) En esta sección, se ilustra el primer enfoque por medio de la interpolación lineal. Este tipo de interpolación se puede remplazar por cualquier otra interpolación de dimensión uno.

2.9 EXTRAPOLACIONES

La extrapolación polinomial es exactamente igual a la interpolación polinomial, excepto que el polinomio ajustado se utiliza fuera de los dos puntos extremos de los datos.

En el dominio donde no se conoce la función, pero se cree que está bien representada, se utiliza la extrapolación extendiendo el uso de una fórmula de interpolación.

Al utilizar una extrapolación, hay que decidir el orden del polinomio a utilizar y qué tanto se extenderá la extrapolación. La extrapolación funciona de manera más confiable si un análisis teórico de la función por extrapolar indica un orden particular a emplear.

En general, el error de extrapolación crece al alejarse el punto de interés de los puntos dados. Si se utiliza una interpolación de orden superior para la extrapolación sin tener una base teórica, los errores pueden crecer rápidamente al aumentar el orden del polinomio. En el apéndice A se describe el análisis de los errores de extrapolación.

En varias partes de este libro se pueden ver aplicaciones de la extrapolación; por ejemplo, véase la fórmula de integración abierta de Newton-Cotes (sección 4.5), el método de integración de Romberg (sección 4.2), el método predictor-corrector (sección 9.4) y los parámetros de aceleración iterativa (sección 12.5). Para un mayor análisis, véase Stoer y Burlish.

PROGRAMAS

PROGRAMA 2-1 Interpolación de Lagrange

A) Explicaciones

Este programa interpola una tabla de valores de funciones mediante una fórmula de interpolación de Lagrange. El usuario define la tabla de valores en los enunciados de los datos. Después de ejecutar la definición, la computadora pide el valor de x para el que se evaluará la fórmula de interpolación.

Aunque el programa está diseñado para la interpolación, se puede utilizar también para la extrapolación. Sin embargo, en este caso se imprime un mensaje "X está en el rango de extrapolación".

B) Variables

K: número de datos
F(I), X(I): datos dados
YRES: valor numérico de la interpolación para un valor dado de x
XA: valor de x para el cual hay que evaluar la fórmula de interpolación

C) Listado

```
C-----CSL/F2-1.FOR      INTERPOLACION DE LAGRANGE
      DIMENSION F(0:10),X(0:10)
C           N ES EL ORDEN DEL POLINOMIO DE INTERPOLACION
      DATA N /3/
      DATA (X(I),I=0,3) /  1.,    2.,   3.,   4./
      DATA (F(I),I=0,3) / .671,.620,.567,.512 /
      PRINT *
      PRINT *,'CSL/F2-1       INTERPOLACION DE LAGRANGE'
      PRINT *
      PRINT *, 'TABLA DE VALORES UTILIZADOS'
      PRINT *, '-----------------------------------------'
      PRINT *, '            I      X(I)            F(I) '
        DO 37 I=0,N
          PRINT *, I,X(I),F(I)
37      CONTINUE
      PRINT *, '-----------------------------------------'
45    PRINT *, 'DAR X ?'
      READ *,XA
      IF (XA.LT.X(0). OR . XA.GT.X(N)) PRINT *,
     & '        ADVERTENCIA: X ESTA EN EL RANGO DE EXTRAPOLACION '
      YRES=0
```

```
      DO I=0,N
         Z=1.0
         DO J=0,N
            IF (I.NE.J) Z=Z*(XA-X(J))/(X(I)-X(J))
         END DO
         YRES=WYRES+Z*F(I)
      END DO
      PRINT 200,XA,YRES
200   FORMAT('  RESULTADO DE LA INTERPOLACION :  G(',1PE12.5,') =',1PE12.5)
      PRINT *
      PRINT*,' OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
      READ *,K
      IF(K.EQ.1) GOTO 45
      PRINT*
      END
```

D) Ejemplo de salida

```
CSL/F2-1        INTERPOLACION DE LAGRANGE

TABLA DE VALORES UTILIZADOS
-----------------------------------------
          I      X(I)          F(I)
          0    1.000000      0.6710000
          1    2.000000.     0.6200000
          2    3.000000      0.5670000
          3    4.000000      0.5120000
-----------------------------------------
DAR X ?
3.66
RESULTADO DE LA INTERPOLACION :     G( 3.66000E+00) = 5.30924E-01

DAR X ?
4.5
      ADVERTENCIA: X ESTA EN EL RANGO DE EXTRAPOLACION
RESULTADO DE LA INTERPOLACION:  G( 4.50000E+00) = 4.83750E-01

DAR X ?
0.1
      ADVERTENCIA: X ESTA EN EL RANGO DE EXTRAPOLACION
RESULTADO DE LA INTERPOLACION:  G( 1.00000E-01) = 7.15190E-01
```

PROGRAMA 2-2 Tabla de diferencias hacia adelante

A) Explicaciones

Este programa genera la tabla de diferencias hacia adelante para una tabla de una función dada, definida en la instrucción DATA.

B) Variables

x_0: valor inicial de los puntos de la malla

h: tamaño del intervalo, $h = x_{i+1} - x_i$.

C) Listado

```
C-----CSL/F2-2.FOR      TABLA DE DIFERENCIAS HACIA ADELANTE CON PUNTOS
C                       DE LA MALLA ESPACIADOS UNIFORMEMENTE
      DIMENSION F(0:10,0:10), X(0:10)
15    PRINT *
20    PRINT *,'CSL/F2-2        TABLA DE DIFERENCIAS HACIA ADELANTE (FORTRAN)
C                  ! NI es el orden máximo; NI + 1 es el número de datos
      DATA NI/6/
      DATA (X(I),I=0,6) /1,3,5,7,9,11,13 /
      DATA (F(I,0),I=0,6)/1.0, 0.5, 0.3333333, 0.25, 0.2, 0.1666666,
     1   0.14285714/
      DO K=1,NI
        J=NI-K
        DO I= 0,J
          F(I,K)= F(I+1,K-1)-F(I,K-1)
        END DO
      END DO
      PRINT *
      PRINT *,' I    X(I)        F(I)   1ER. ORDEN, 2DO. ORDEN. . .
        DO I=0, NI
          J=NI-I
          PRINT 440,I,X(I), (F(I,K),K=0,J)
          PRINT *
        END DO
400   PRINT *
440   FORMAT (1X,I2,8F9.5)
      END
```

D) Ejemplo de salida

```
CSL/F2-2       TABLA DE DIFERENCIAS HACIA ADELANTE (FORTRAN)
-----------------------------------------------------------------------------
                                  DIFERENCIAS DE ORDEN N
                          ---------------------------------------------------
I   X(I)        F(I)     N=1        2         3         4         5         6
0  1.00000  1.00000 -0.50000  0.33333 -0.25000  0.20000 -0.16667  0.14286
1  3.00000  0.50000 -0.16667  0.08333 -0.05000  0.03333 -0.02381 ,
2  5.00000  0.33333 -0.08333  0.03333 -0.01667  0.00952
3  7.00000  0.25000 -0.05000  0.01667 -0.00714
4  9.00000  0.20000 -0.03333  0.00952
5 11.00000  0.16667 -0.02381
6 13.00000  0.14286
-----------------------------------------------------------------------------
```

PROGRAMA 2-3 Tabla de diferencias divididas

A) Explicaciones

Este programa desarrolla una tabla de diferencias divididas. Todos los datos de entrada están definidos en las instrucciones DATA.

Antes de ejecutar el programa, el usuario debe definir la tabla de valores en las instrucciones DATA. Los valores muestra de las instrucciones DATA en el progra-

ma listado a continuación son del ejemplo 2.8. Las diferencias divididas se calculan en dos ciclos, uno para K y otro para I.

B) Variables

NI: número de puntos dados en la tabla de valores
J: número máximo de diferencias para cada k
K: orden de una diferencia
K(I): valores x de los puntos
F(K, I): diferencia dividida de orden k: F(0, I) es el valor de la función para el punto I.

C) Listado

```
C-----CSL/F2-3.FOR      TABLA DE DIFERENCIAS DIVIDIDAS CON
C                       PUNTOS DE LA MALLA SEPARADOS DE MANERA NO UNIFORME
      DIMENSION F(0:10,0:10), X(0:10)
      PRINT *
      PRINT *,'CSL/F2-3       TABLA DE DIFERENCIAS DIVIDIDAS '
      DATA NI/6/
      DATA (X(I),  I=0,6)/0.1, 0.2, 0.4, 0.7, 1.0, 1.2, 1.3/
      DATA (F(I,0),I=0,6)/.99750, .99002, .96040, .88120, .76520
     1    ,.67113, .62009/
      DO K=1,NI
        J=NI-K
        DO I= 0,J
          F(I,K)=(F(I+1,K-1)-F(I,K-1))/(X(I+K)-X(I))
        END DO
      END DO
      PRINT *
      PRINT *,' I   X(I)     F(I)     F(I,I+1)  F(I,I+2),..'
        DO I=0, NI
          J=NI-I
          PRINT 440,I,X(I), (F(I,K),K=0,J)
        END DO
      PRINT *
440   FORMAT (1X,I2,8F9.5)
      END
```

D) Ejemplo de salida

```
CSL/F2-3       TABLA DE DIFERENCIAS DIVIDIDAS
-----------------------------------------------------------------------
I   X(I)     F(I)     F(I,I+1), F(I,I+2),..
0  0.10000  0.99750 -0.07480 -0.24433  0.02089  0.01479 -0.00239  0.00128
1  0.20000  0.99002 -0.14810 -0.23180  0.03419  0.01215 -0.00085
2  0.40000  0.96040 -0.26400 -0.20444  0.04635  0.01122
3  0.70000  0.88120 -0.38667 -0.16737  0.05644
4  1.00000  0.76520 -0.47035 -0.13350
5  1.20000  0.67113 -0.51040
6  1.30000  0.62009
-----------------------------------------------------------------------
```

PROBLEMAS

2.1) Se tienen las siguientes parejas muestradas de $y = \cos(x)$:

a) $x = 0, \quad y = 1$
$x = 0.1, \quad y = 0.99500$

b) $x = 0, \quad y = 1$
$x = 0.2, \quad y = 0.98007$

c) $x = 0, \quad y = 1$
$x = 0.5, \quad y = 0.87758$

Aproxime el valor de y en el punto medio mediante interpolación y estime el error utilizando la ecuación (2.2.2). Compare el error estimado con el valor exacto del error evaluado al comparar con $\cos(x)$.

2.2) Si un conjunto de datos es (f_i, x_i), $i = 1, 2, \ldots, N$, se puede ajustar la interpolación lineal a cada pareja de datos consecutivos; a saber, (f_i, x_i) y (f_{i+1}). La interpolación lineal se puede escribir para cada intervalo de datos. Sin embargo, las fórmulas de interpolación se pueden expresar mediante una única ecuación.

a) Muestre que las fórmulas de interpolación lineal por partes para el rango $[x_0, x_N]$ se pueden expresar de manera compacta como

$$g(x) = \sum_{i=0}^{N} f_i \eta_i(x) \tag{a}$$

donde $n_i(x)$ está definida como

$$\begin{aligned} \eta_i(x) &= \frac{x - x_{i-1}}{x_i - x_{i-1}} \quad \text{para } x_{i-1} \leqslant x \leqslant x_i \\ &= \frac{x_{i+1} - x}{x_{i+1} - x_i} \quad \text{para } x_i \leqslant x \leqslant x_{i+1} \\ &= 0 \qquad \text{de otro modo} \end{aligned}$$

b) Grafique $n_i(x)$ y su derivada.

2.3) a) Escriba la fórmula de interpolación de Lagrange ajustada a los puntos $i = 2$, 3 y 4 dados en la siguiente tabla:

i	x_i	$f(x_i)$
1	0	0.9162
2	0.25	0.8109
3	0.5	0.6931
4	0.75	0.5596
5	1.0	0.4055

b) Si la tercera derivada de la función en $i = 3$ es $f''' = -0.26$, estime el error de la interpolación de Lagrange obtenido en el inciso a) en $x = 0.6$.

2.4) Una interpolación de Lagrange de orden N (con $N + 1$ puntos) para una función $f(x)$ es exacta, si $f(x)$ es un polinomio de orden menor o igual que N. Explique la razón de esto en dos formas distintas.

2.5) a) Escriba la interpolación de Lagrange que pasa por los siguientes puntos:

x	0	0.4	0.8	1.2
f	1.0	1.49182	2.22554	3.32011

b) Si sabe que $f''''(0.6) = 1.822$, estime el error en $x = 0.2$, 0.6 y 1.0 utilizando la ecuación (2.3.9) con $\xi = x_m$. (En el caso en que f'''' no se conoce, se puede calcular una aproximación para f'''' mediante una aproximación por diferencias, siempre y cuando se disponga de un punto más en la tabla de valores.)

c) Evalúe el valor exacto de la fórmula de interpolación en $x = 0.2$, 0.6 y 1.0 mediante $e(x) = f(x) - g(x) = \exp(x) - g(x)$.

2.6) Ajuste x sen (x) en $[0, \pi/2]$ con el polinomio de interpolación de Lagrange de orden 4 utilizando puntos con igual separación. Calcule el error de cada interpolación en cada incremento de $\pi/16$ y grafique.

2.7) a) Escriba un programa para evaluar la interpolación de Lagrange para $\sqrt{x}\cos(x)$ en [0, 2] con seis puntos de la malla que tengan igual separación y $h = 0.4$. **b)** Calcule el error del polinomio de interpolación para cada incremento de 0.1 de x. Grafique la distribución del error.

2.8) ajuste sen (x) en $[0, 2\pi]$ con el polinomio de interpolación de Lagrange de orden 4 y 8, utilizando puntos con igual separación (5 y 9 puntos, respectivamente). Grafique los polinomios de interpolación junto con sen (x) y las distribuciones de los errores.

2.9) a) Desarrolle una interpolación de Lagrange para $\log_e(x)$ en $1 \leqslant x \leqslant 2$ utilizando cuatro puntos con igual separación. **b)** Estime el error de la aproximación, utilizando la ecuación (2.3.9) en $x = 1, 1.2, 1.3, \ldots, 1.9$ y 2.0. **c)** Calcule el error exacto usando $e(x) = \log_e(x) - g(x)$.

2.10) Aproxime

$$y = \frac{1 + x}{1 + 2x + 3x^2}$$

en [0, 5] mediante la interpolación de Lagrange de orden 4 y evalúe el error exacto mediante $e(x) = y - g(x)$. Trabaje efectuando los pasos siguientes: **a)** determine los puntos, **b)** escriba la interpolación de Lagrange, **c)** calcule el error para cada incremento de 0.2 en x y **d)** grafique la distribución del error.

2.11) Si se ajusta un polinomio de interpolación de Lagrange a cuatro datos en $x = 1$, 2, 3 y 4, aparecen los siguientes polinomios cúbicos en la fórmula de interpolación:

a) $\dfrac{(x-2)(x-3)(x-4)}{(1-2)(1-3)(1-4)}$

b) $\dfrac{(x-1)(x-3)(x-4)}{(2-1)(2-3)(2-4)}$

c) $\dfrac{(x-1)(x-2)(x-4)}{(3-1)(3-2)(3-4)}$

d) $\dfrac{(x-1)(x-2)(x-3)}{(4-1)(4-2)(4-3)}$

Grafique las cuatro funciones anteriores y analice las implicaciones de la forma de cada una.

2.12) La fórmula de interpolación de Lagrange se puede escribir en forma compacta como

$$g(x) = \sum_{i=0}^{N} f_i \eta_i(x)$$

donde $\eta_i(x)$ es una función de forma definida por

$$\eta_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j}$$

Bosqueje la forma de la función.

2.13) Deduzca el polinomio de interpolación de Newton hacia adelante ajustado a los siguientes puntos del ejemplo 2.4:

a) $i = 1, 2, 3$

b) $i = 2, 3, 4, 5$

Estime también el error de las interpolaciones anteriores en $x = 0.75$, utilizando el método mostrado en el ejemplo 2.5.

2.14) Demuestre analíticamente que $\Delta^n x^{n-1} = 0$.

2.15) Si $f(x)$ es un polinomio de orden menor o igual que N, una interpolación de Newton hacia adelante de orden N será exactamente igual a $f(x)$, sin que importe el tamaño h del intervalo. Explique por qué.

2.16) Deduzca el polinomio de interpolación de Newton hacia adelante que pasa por los puntos $i = 2, 3, 4$ dados en la siguiente tabla:

i	x_i	$f(x_i)$
1	0	0.9162
2	0.25	0.8109
3	0.5	0.6931
4	0.75	0.5596
5	1.0	0.4055

2.17) La siguiente tabla de valores se muestreó del polinomio:

$$y = 2x^3 + 3x + 1$$

x	y
0.1	1.302
0.2	1.616
0.3	1.954
0.4	2.328
0.5	2.750

a) Elabore una tabla de diferencias hacia adelante y muestre que la diferencia de cuarto orden se anula. **b)** Explique por qué ocurre esto.

2.18) Haga la tabla de diferencias hacia adelante a partir de la siguiente tabla de valores:

i	x	$f(x)$
1	0.5	1.143
2	1.0	1.000
3	1.5	0.828
4	2.0	0.667
5	2.5	0.533
6	3.0	0.428

Por medio de las fórmulas de Newton hacia adelante, escriba los polinomios de interpolación ajustados a:

a) $i = 1, 2, 3$

b) $i = 4, 5, 6$

c) $i = 2, 3, 4, 5$

2.19) Formule una expresión aproximada del error en cada una de las fórmulas de interpolación obtenidas en el problema anterior.

2.20) Demuestre que, si $k = 3$ en la ecuación (2.4.12), $g(x)$ es igual a f_0, f_{-1}, f_{-2} y f_{-3} para $s = 0, -1, -2$ y -3 respectivamente.

2.21) El polinomio de interpolación de Newton hacia atrás ajustado a los puntos x_0, x_1 y x_2 se escribe como

$$g(x) = f_2 + s\nabla f_2 + \tfrac{1}{2}s(s+1)\nabla^2 f_2, \quad -2 \leqslant s \leqslant 0$$

donde

$s = (x - x_2)/h$

Por otro lado, el polinomio de interpolación de Newton hacia adelante ajustado a los mismos datos es

$$g(x) = f_0 + s\Delta f_0 + \tfrac{1}{2}(s-1)s\Delta^2 f_0, \quad 0 \leqslant s \leqslant 2$$

donde

$s = (x - x_0)/h$

Verifique la equivalencia de las ecuaciones.

2.22) ¿Es posible escribir una interpolación de Newton hacia atrás utilizando la tabla de diferencias hacia adelante? Explique cómo.

2.23) Escriba una tabla de diferencias divididas para la siguiente tabla de valores.

Viscosidad del agua
$(\mathrm{Ns/m^2}) \times 10^{-3}$

$T(C)$	μ
0	1.792
10	1.308
30	0.801
50	0.549
70	0.406
90	0.317
100	0.284

2.24) Obtenga el polinomio de interpolación ajustado a los siguientes puntos del ejemplo 2.8.

a) $i = 2, 3, 4$

b) $i = 0, 1, 2, 3$

Estime el error de las interpolaciones obtenidas en $x = 0.3$ y $x = 0.55$.

2.25) Demuestre que, si $P_{0,1,2,\ldots,m}$, dado por la ecuación (2.5.3) se desarrolla en una serie de potencias con respecto a x, el coeficiente del término de orden mayor, x^N, está dado por

$$a_N = \frac{f_N - P_{0,1,\ldots,m-1}}{\prod_{i=0}^{N-1} (x_N - x_i)}$$

donde f_N es el valor de la función en x_N.

2.26) Examine la validez de la ecuación (2.6.3) introduciendo los siguientes polinomios de Chebyshev de la ecuación (2.6.2):

a) T_1, T_2, T_3

b) T_3, T_4, T_5

2.27) **a)** Desarrolle una aproximación mediante interpolación de Lagrange para $\log_e(x)$ en $1 \leqslant x \leqslant 2$ utilizando cuatro puntos de Chebyshev. **b)** Estime el error de la aproximación por medio de la ecuación (2.3.9) en $x = 1, 1.2, 1.3, \ldots, 1.9$ y 2.0. **c)** Calcule el error real cuando $e(x) = \log_e(x) - g(x)$.

2.28) Obtenga una fórmula de interpolación cuadrática para $\log_e(x)$ en el intervalo $1 < x < 3$ utilizando tres puntos de Chebyshev. Escriba la ecuación para estimar el error y evalúela en $x = 2.5$.

2.29) Desarrolle un polinomio de interpolación de Lagrange ajustando a

$$y = \frac{x + 1}{1 + 2x + 3x^2}$$

en [1, 3] con tres puntos de Chebyshev.

2.30) Repita el problema del ejemplo 2.2 con los puntos de Chebyshev (use 3 y 5 puntos, respectivamente).

2.31) Aproxime e^x mediante las interpolaciones cúbicas de Hermite por partes en el intervalo [0, 2] con dos intervalos. Calcule el error de las interpolaciones de Hermite por partes para cada incremento de 0.2 en x y grafique el error.

2.32) Determine el polinomio ajustado a

$$f(0) = 1, \quad f(1) = 2, \quad f'(0) = 0.5$$

2.33) Determine el polinomio ajustado a

$$f(0) = 1, \quad f(1) = 2, \quad f'(0) = 0, \quad f'(1) = 1$$

2.34) Verifique que la ecuación (G) del ejemplo 2.10 es el polinomio que se ajusta a f_{i-1}, f_{i-1}, f_i y f_i.

2.35) Calcule los valores de las cuatro funciones en la ecuación (I) del ejemplo 2.10 en $s = 0$ y $s = h$.

2.36) Determine el polinomio de interpolación de Hermite de segundo orden (parabólico) ajustado a

$$f(1) = 2, \quad f(2) = 3, \quad f'(2) = 1.2$$

2.37) Defina el polinomio cúbico de interpolación de Hermite ajustado a

$$f(1) = 2, \quad f(2) = 3, \quad f'(2) = 0.5, \quad f(3) = 0$$

BIBLIOGRAFIA

Abramowitz, M. e I.A. Stegun, editores, *Handbook of Mathematical Functions*, National Bureau of Standards, 1970.

Becker, E. B., G. F. Carey y J. T. Oden, *Finite Elements: An Introduction*, Prentice-Hall, 1981.

Carnahan, B., H. A. Luther, y J. O. Wilkes, *Applied Numerical Methods*, Wiley, 1969.

Conte, S. D. y C. de Boor, *Elementary Numerical Analysis*, 3a. edición, McGraw-Hill, 1980.

Gerald, C. F. y P. O. Wheatley, *Applied Numerical Analysis*, 3a. edición, Addison-Wesley, 1984.

Isaacson, E. y H. B. Keller, *Analysis of Numerical Methods*, Wiley, 1966.

Stoer, J. y R. Burlish, *Introduction to Numerical Analysis*, Springer-Verlag, 1980.

3

Solución de ecuaciones no lineales

3.1 INTRODUCCION

Las soluciones de una ecuación no lineal se llaman *raíces o ceros*. Los siguientes son algunos ejemplos de ecuaciones no lineales:

a) $1 + 4x - 16x^2 + 3x^3 + 3x^4 = 0$

b) $f(x) - \alpha = 0, \quad a < x < b$

c) $\dfrac{x(2.1 - 0.5x)^{1/2}}{(1 - x)(1.1 - 0.5x)^{1/2}} - 3.69 = 0, \quad 0 < x < 1$

d) $\tan(x) = \tanh(2x)$

La primera es un ejemplo de ecuación polinomial, que puede aparecer como una ecuación característica para una ecuación diferencial ordinaria lineal, entre otros problemas. El segundo ejemplo es equivalente a evaluar $f^{-1}(\alpha)$, donde $f(x)$ es cualquier función y f^{-1} es su función inversa. El tercer ejemplo es un caso especial del inciso b). El cuarto ejemplo es una ecuación trascendental.

La razón principal para resolver ecuaciones no lineales por medio de métodos computacionales es que esas ecuaciones carecen de solución exacta, excepto para muy pocos problemas. La solución analítica de las ecuaciones polinomiales existe sólo hasta el orden cuatro [Abramowitz/Stegun, pág. 17], pero no existen soluciones en forma exacta para órdenes superiores. Por lo tanto, las raíces de esas ecuaciones no lineales se obtienen mediante métodos computacionales basados en procedimientos iterativos.

Los métodos numéricos diseñados para encontrar las raíces son poderosos, aunque cada uno tiene sus propias limitaciones y defectos. Por lo tanto, los estu-

Tabla 3.1 Resumen de los esquemas para encontrar raíces

Nombre	Necesidad de especificar un intervalo que contenga a la raíz	Necesidad de la continuidad de f'	Tipos de ecuaciones	Otras características especiales
Bisección	"sí"	no	cualquiera	Robusto, aplicable a funciones no analíticas
Falsa posición	"sí"	"sí"	cualquiera	Convergencia lenta en un intervalo grande
Falsa posición modificada	"sí"	"sí"	cualquiera	Más rápido que el método de la falsa posición
Método de Newton	no	"sí"	cualquiera	Rápido; se necesita calcular f'; aplicable a raíces complejas
Método de secante	no	"sí"	cualquiera	Rápido; no se requiere calcular f'
Sustitución sucesiva	no	"sí"	cualquiera	Puede no converger
Método de Bairstow	no	"sí"	polinomial	Factores cuadráticos

diantes deben aprender los pros y los contras de cada método —en particular sus dificultades— y familiarizarse con los métodos mediante la práctica en una computadora.

En la tabla 3.1 se resumen las características principales de los métodos numéricos para ecuaciones lineales descritos en este capítulo. Los primeros tres métodos de la tabla 3.1 (el de bisección, el de la falsa posición y el método de la falsa posición modificado) tienen una característica en común; a saber: estos esquemas pueden encontrar una raíz si se conoce un intervalo de x que contenga a la raíz. Por lo tanto, todos estos métodos necesitan un esfuerzo preliminar para estimar un intervalo adecuado que contenga a la raíz deseada. Los métodos de Newton y de la secante necesitan una estimación inicial, pero no es necesaria la estimación de un intervalo. El método de la sustitución sucesiva es un algoritmo iterativo simple, aunque su desventaja es que la iteración no siempre converge. El método de Bairstow se limita a los polinomios. No obstante, al aplicar varias veces este método, se pueden hallar todas las raíces —incluyendo las complejas— sin conocimientos previos de cualquier tipo, aunque a veces la iteración no converja en lo absoluto.

3.2 METODO DE BISECCION

Este método es el más simple, aunque también el más seguro y sólido para encontrar una raíz en un intervalo, dado donde se sabe que existe dicha raíz. Su única ventaja es que funciona aun para funciones no analíticas.

Suponga que el intervalo entre $x = a$ y $x = c$ denotado por $[a, c]$ —o equivalentemente $a \leqslant x \leqslant c$— tiene una sola raíz, como se muestra en la figura 3.1. El método de bisección se basa en el hecho de que, para que un intervalo $[a, c]$ tenga una raíz, basta que los signos de $y(x)$ en los dos extremos sean opuestos, o bien que $f(a)$ o $f(c)$ se anulen; es decir, $f(a)f(c) \leqslant 0$.

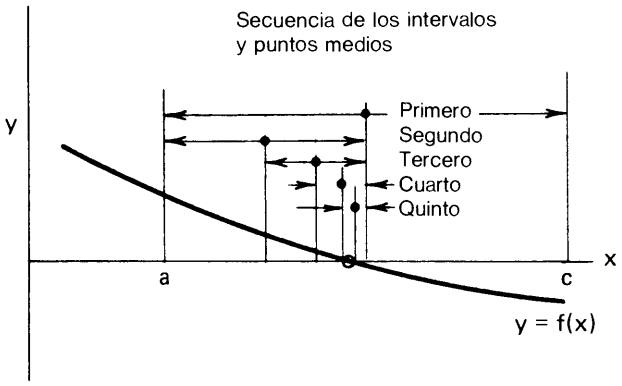


Figura 3.1 Método de bisección

El primer paso para utilizar este método es bisectar el intervalo $[a, c]$ en dos mitades; a saber, $[a, b]$ y $[b, c]$, donde $b = (a + c)/2$. Al verificar los signos de $f(a)f(b)$ y $f(b)f(c)$, se localiza la mitad del intervalo que contiene la raíz. Así, si $f(a)f(b) \leqslant 0$, el intervalo $[a, b]$ que incluye a $x = a$ y $x = b$ contiene a la raíz: en caso contrario, el intervalo $[b, c]$ tiene la raíz. El nuevo intervalo que contiene a la raíz se bisecta de nuevo. Al repetir este proceso, el tamaño del intervalo con la raíz se vuelve cada vez más pequeño. En cada paso, se toma el punto medio del intervalo como la aproximación más actualizada de la raíz. La iteración se detiene cuando la mitad del intervalo está dentro de una tolerancia dada ε. El PROGRAMA 3-1 está diseñado para encontrar una raíz por el método de bisección.

El tamaño del intervalo después de n pasos de la iteración es

$$\frac{(c - a)_0}{2^n}$$

donde el numerador es el tamaño del intervalo inicial. Esto también representa el máximo error posible cuando la raíz se aproxima mediante el n-ésimo punto medio. Por lo tanto, si la tolerancia del error está dada por ε, el número de pasos de iteración necesarios es el mínimo entero que satisface

$$\frac{(c - a)_0}{2^n} < \varepsilon \tag{3.2.1}$$

o, en forma equivalente

$$n \geqslant \log_2 \frac{(c - a)_0}{\varepsilon} \tag{3.2.2}$$

Por ejemplo, si $(c - a)_0 = 1$ y $\varepsilon = 0.0001$, entonces $n = 14$.

Ejemplo 3.1

Se sabe que la raiz de

$$e^x - 2 = 0$$

está en [0, 2]. Hallar un valor aproximado de la raiz con una tolerancia de $\varepsilon = 0.01$ mediante el método de la bisección.

(Solución)

El cálculo manual del método de bisección se puede llevar a cabo elaborando una tabla como se muestra abajo. Cuando empieza la primera iteración, los valores de $a = 0$ y $c = 2$ y el punto medio $b = (0 + 1)/2 = 1$ se escriben en la tabla en el renglón $i = 1$. También se calculan $f(a)$, $f(b)$ y $f(c)$ y se escriben en el mismo renglón. Al examinar los signos de estos tres valores de f, vemos que la raiz se localiza entre a y b. Por lo tanto, a y b del paso $i = 1$ se convierten respectivamente, en a y c para el paso $i = 2$. Así, $f(a)$ y $f(b)$ del paso $i = 1$ se copian a $f(a)$ y $f(c)$ para el paso $i = 2$. La b para el paso $i = 2$ es $b = (a + c)/2 = 0.5$ y se calcula $f(b)$, escribiendo su valor en la tabla. La iteración para el resto continúa de manera similar hasta que se alcanza la tolerancia. El último valor de b es la respuesta final.

Número de iteración, i	a	b	c	$f(a)$	$f(b)$	$f(c)$	Cota del error
1	0	1	2	−1	0.7182	5.3890	1
2	0	0.5	1	−1	−0.3512	0.7182	0.5
3	0.5	0.75	1	−0.3512	0.1170	0.7182	0.25
4	0.5	0.625	0.75	−0.3512	−0.1317	0.1170	0.125
5	0.625	0.6875	0.75	−0.1317	−0.0112	0.1170	0.0625
6	0.6875	0.7187	0.75	−0.0112	0.0518	0.1170	0.03125
7	0.6875	0.7031	0.7187	−0.0112	0.0200	0.0518	0.015625
8	0.6875	0.6953	0.7031	−0.0112	0.0043	0.0200	0.0078125

La octava aproximación para la raiz es $b = 0.6953$. Su cota de error (máximo error posible) es 0.0078, que está dentro de la tolerancia especifica.

Hemos supuesto que el intervalo inicial tiene sólo una raiz y que $f(a)f(b) \leqslant 0$. Sin embargo, $f(a)f(b) \leqslant 0$ se satisface siempre que el intervalo tenga un número impar de raices, como se ilustra en la figura 3.2. En este caso, el método de bisección encontrará una de las raices separadas en el intervalo dado. El método de bisección no puede encontrar una pareja de raices dobles, debido a que la función toca el eje x de manera tangencial en las raices dobles, como se muestra en la figura 3.3.

Un defecto del método de bisección es que éste puede atrapar una singularidad como si fuera una raiz, debido a que dicho método no reconoce la diferencia entre una raiz y una singularidad. Un punto singular es aquel en el que el valor de la función tiende a infinito, lo cual se ilustra en la figura 3.4. Para evitar este problema, el programa debe verificar si $|f(c) - f(a)|$ converge a cero cuando se lleva a cabo el método de bisección. Si esta cantidad diverge, el programa está atrapando una singularidad en vez de una raiz.

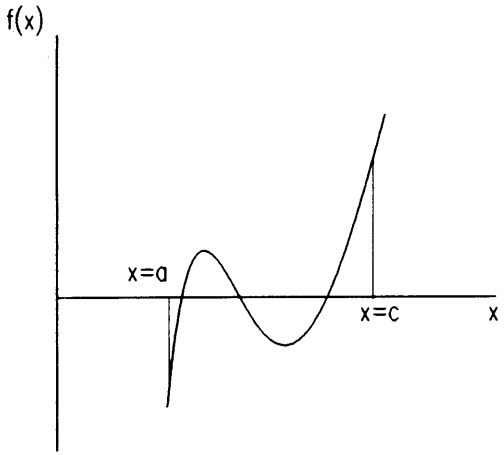


Figura 3.2 Número impar de raíces en un intervalo dado

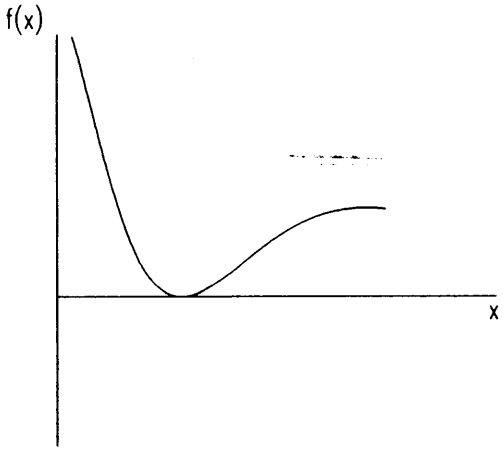


Figura 3.3 Función que toca al eje x en un punto

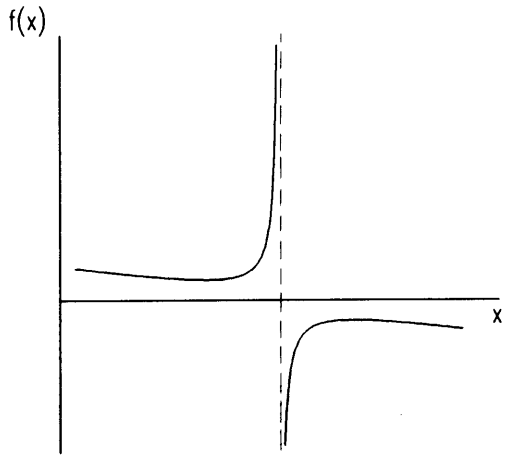


Figura 3.4 Función con una singularidad

Cuando no hay información previa acerca de los valores aproximados de las raíces, una forma sencilla para hallar intervalos de x que contengan una raíz es escribir una tabla de la función para valores de x con **separación uniforme** (véase el PROGRAMA 3-2), o graficar la función mediante computadora (véase el PROGRAMA 3-3). Si el signo del valor de la función cambia a través de un intervalo, existe al menos una raíz en ese intervalo. El enfoque gráfico es útil para localizar intervalos que contengan una raíz, en particular cuando la ecuación tenga varias raíces.

Ejemplo 3.2

a) Determinar intervalos de tamaño 1.0, tales que cada uno contenga una o más raíces (impares) de

$$y = -19(x - 0.5)(x - 1) + \exp(x) - \exp(-2x), \quad -10 < x < 10$$

b) Repita lo anterior utilizando el PROGRAMA 3-2 con tamaño del intervalo 0.1.

(Solución)

a) Calculamos y para $x = -10, -9, -8, \ldots, 10$ y hacemos una tabla de valores. Después, marcamos el intervalo en el que la función cambia de signo, como se muestra en la tabla E3.2.

Tabla E3.2 Tabla de valores

x	y
−10.0	−48.517E + 07
−9.0	−65.662E + 06
−8.0	−88.876E + 05
−7.0	−12.037E + 05
−6.0	−16.362E + 04
−5.0	−22.653E + 03
−4.0	−34.084E + 02
−3.0	−66.938E + 01
−2.0	−19.696E + 01
−1.0	−64.021E + 00
0.0	−95.000E − 01
1.0	25.829E − 01
2.0	−21.129E + 00
3.0	−74.917E + 00
4.0	−14.490E + 01
5.0	−19.359E + 01
6.0	−11.907E + 01
7.0	35.563E + 01
8.0	19.835E + 02
9.0	68.111E + 02
10.0	20.402E + 03

Así, se encuentran tres intervalos, [0, 1], [1, 2] y [6, 7], cada uno de los cuales contiene al menos una raíz.

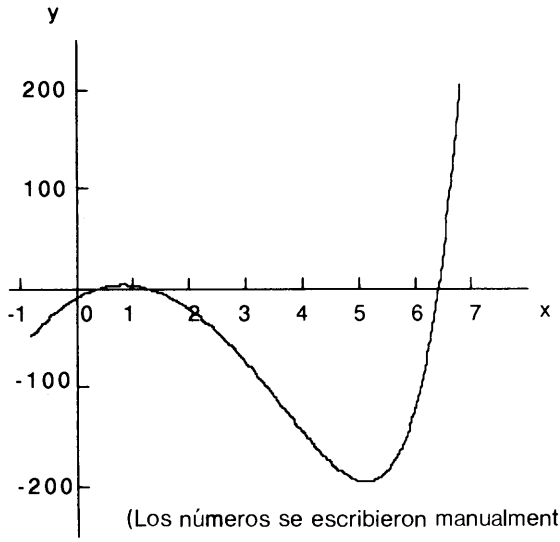


Figura E3.2 Muestra de la función realizada por el PROGRAMA 3-3 (versión BASIC)

b) Al ejecutar el PROGRAMA 3-2 para el rango [−10, 10] con el tamaño del intervalo $h = 1$, se encuentran tres intervalos, [0, 1], [1, 2] y [6, 7], cada uno de los cuales contiene al menos una raíz, como era de esperarse por lo dicho en a). Al ejecutar el PROGRAMA 3-2 nuevamente para cada uno de estos intervalos con $h = 0.1$, los tamaños de los intervalos se reducen a [0.4, 0.5], [1.2, 1.3] y [6.4, 6.5]. (El PROGRAMA 3-2 se puede ejecutar para un rango grande con un intervalo pequeño h en una sola ejecución, pero el tiempo de cómputo crece.) En la figura E3.2 se ilustra la gráfica de la función, la cual se obtuvo mediante el PROGRAMA 3-3.

Resumen de Esta Sección

a) El método de bisección encuentra una raíz de una función si se sabe que la raíz existe en un intervalo dado.

b) El método de bisección encuentra una raíz aun cuando la función no sea analítica.

c) Por otro lado, se puede atrapar una singularidad como si fuera una raíz, debido a que el método no distingue las raíces de las singularidades.

d) Una tarea importante que se debe realizar antes de aplicar el método de bisección es encontrar un intervalo que contenga a la raíz. La búsqueda de raíces se puede llevar a cabo listando una tabla de valores o graficando la función en la pantalla.

3.3 METODO DE LA FALSA POSICION Y METODO DE LA FALSA POSICION MODIFICADA

El método de la falsa posición, —basado en la interpolación lineal— es análogo al método de bisección, puesto que el tamaño del intervalo que contiene a la raíz se

reduce mediante iteración. Sin embargo, en vez de bisectar en forma monótona el intervalo, se utiliza una interpolación lineal ajustada a dos puntos extremos para encontrar una aproximación de la raíz. Así, si la función está bien aproximada por la interpolación lineal, entonces las raíces estimadas tendrán una buena precisión y, en consecuencia, la iteración convergerá más rápido que cuando se utiliza el método de bisección.

Dado un intervalo $[a, c]$ que contenga a la raíz, la función lineal que pasa por $(a, f(a))$ y $(c, f(c))$ se escribe como

$$y = f(a) + \frac{f(c) - f(a)}{c - a}(x - a) \tag{3.3.1}$$

o, despejando x,

$$x = a + \frac{c - a}{f(c) - f(a)}(y - f(a)) \tag{3.3.2}$$

La coordenada x en donde la línea intersecta al eje x se determina al hacer $y = 0$ en la ecuación (E.E.2); es decir,

$$b = a - \frac{c - a}{f(c) - f(a)} f(a) = \frac{af(c) - cf(a)}{f(c) - f(a)} \tag{3.3.3}$$

Después de encontrar b, el intervalo $[a, c]$ se divide en $[a, b]$ y $[b, c]$. Si $f(a)f(b) \leqslant 0$, la raíz se encuentra en $[a, b]$; en caso contrario, está en $[b, c]$. Los extremos del nuevo intervalo que contiene a la raíz se renombran a y c. El procedimiento de interpolación se repite hasta que las raíces estimadas convergen.

La desventaja de este método es que pueden aparecer extremos fijos, como lo muestra la figura 3.5, en donde uno de los extremos de la sucesión de intervalos no se mueve del punto original, por lo que las aproximaciones a la raíz, denotadas por $b_1, b_2, b_3, \ldots$ convergen a la raíz exacta solamente por un lado. Los extremos fijos no son deseables debido a que hacen más lenta la convergencia, en particular cuando el intervalo inicial es muy grande o cuando la función se desvía de manera significativa de una línea recta en el intervalo. El método de la falsa posición modificado que se explica a continuación elimina esta dificultad.

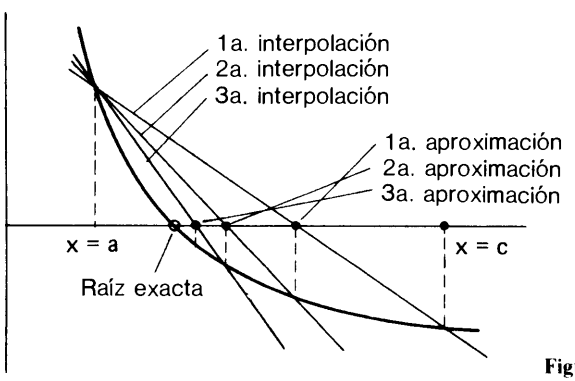


Figura 3.5 Método de la falsa posición

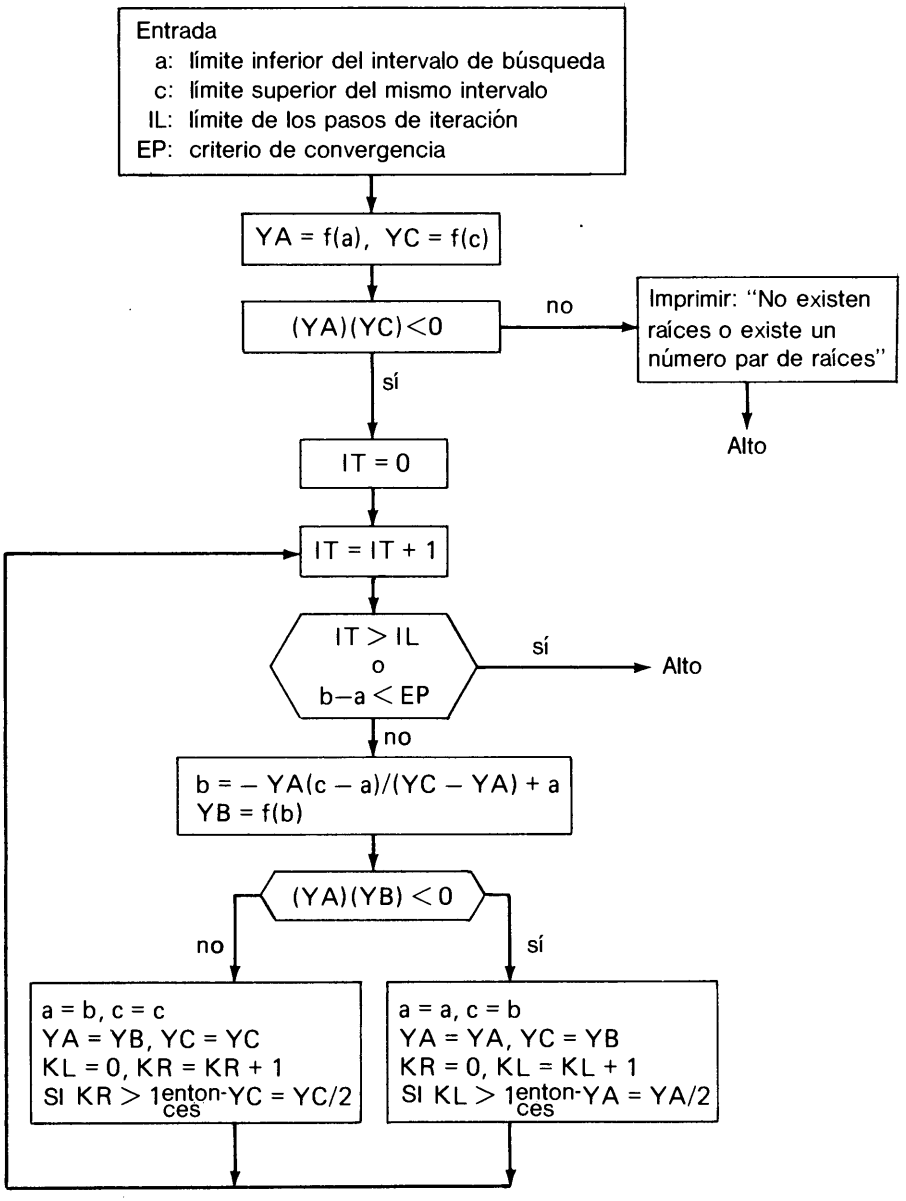


Figura 3.6 Diagrama de flujo del método de la falsa posición modificada

En este método, el valor de f en un punto fijo se divide a la mitad si este punto se ha repetido más de dos veces. El extremo que se repite se llama *extremo fijo*. La excepción a esta regla es que para $i = 2$, el valor de f en un extremo se divide entre 2 de inmediato si no se mueve.

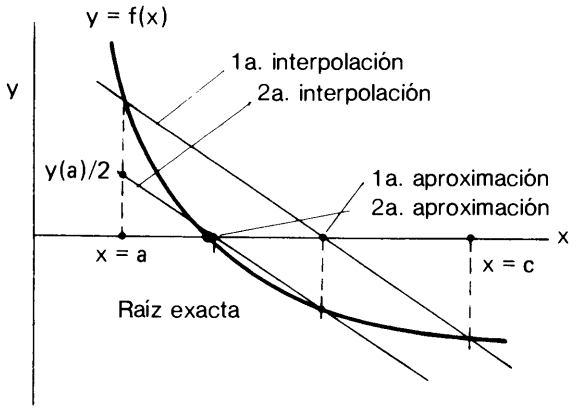


Figura 3.7 Método de la falsa posición modificada

El algoritmo se muestra de manera esquemática en la figura 3.6. El efecto de dividir el valor de y es que la solución de la interpolación lineal se hace cada vez más cercana a la verdadera raíz, como lo ilustra la figura 3.7.

Si este método se utiliza con una calculadora de bolsillo, se sugiere trabajar con un formato de tabla, como el del ejemplo 3.3.

Ejemplo 3.3

Por medio del método de la falsa posición, encontrar la mínima raíz positiva de

$$f(x) = \tan(x) - x - 0.5 = 0$$

la cual se sabe que se encuentra en $0.1 < x < 1.4$.

(Solución)

Los cálculos se muestran en la figura E3.3. En el renglón de la primera iteración ($i = 1$), se escriben los valores de $a = 0.1$, $c = 1.4$ y los valores calculados de $f(a)$ y $f(c)$. El valor b se halla mediante interpolación lineal,

$$b = a - \frac{c - a}{f(c) - f(a)} f(a)$$

y en consecuencia se calcula $f(b)$. Estos dos números se escriben en el mismo renglón. Al examinar los signos de $f(a)$, $f(b)$ y $f(c)$, se localiza la raíz entre b y c. Por lo tanto, los valores b y c de la primera iteración se copian en a y c para $i = 2$, respectivamente. El valor de $f(b)$ para $i = 1$ se copia en $f(a)$ para $i = 2$, pero $f(c)/2$ de $i = 1$ se copia a $f(c)$ para $i = 2$. El valor de $f^*(c)$. El valor de b para $i = 2$ se encuentra mediante la ecuación (3.3.3) de la misma forma que en el paso $i = 1$, excepto que se utiliza $f^*(c)$:

$$b = a - \frac{c - a}{f^*(c) - f(a)} f(a) \tag{3.3.4}$$

Programa 1.4 Esquema lineal modificado

```
COTA INFERIOR A=?0.1
COTA SUPERIOR C=? 1.4.
TOLERANCIA EP=? 0.00001
LIMITE DE ITERACIONES=? 13
ENTRADA: A= .1, C= 1.4, EP= .00001, IL= 13

It.No    a            b            c            f( a)        f( b)        f( c)
 1     1.0000E-01   2.4771E-01   1.4000E+00  -4.9967E-01  -4.9481E-01   3.8979E+00
                                                                        *0.5
 2     2.4771E-01   4.8102E-01   1.4000E+00  -4.9481E-01  -4.5911E-01   1.9489E+00
                                                                        *0.5
 3     4.8102E-01   7.7533E-01   1.4000E+00  -4.5911E-01  -2.9527E-01   9.7447E-01
                                                                        *0.5
 4     7.7533E-01   1.0110E+00   1.4000E+00  -2.9527E-01   8.4850E-02   4.8724E-01
 5     7.7533E-01   9.5842E-01   1.0110E+00  -2.9527E-01  -3.4845E-02   8.4850E-02
 6     9.5842E-01   9.7374E-01   1.0110E+00  -3.4845E-02  -2.7664E-03   8.4850E-02
                                                                        *0.5
 7     9.7374E-01   9.7603E-01   1.0110E+00  -2.7664E-03   2.1981E-03   4.2425E-02
 8     9.7374E-01   9.7501E-01   9.7603E-01  -2.7664E-03  -6.1393E-06   2.1981E-03
 9     9.7501E-01   9.7502E-01   9.7603E-01  -6.1393E-06   0.0000E+00   2.1981E-03

Se ha satisfecho la tolerancia
        Raiz aproximada = .9750172
```

Figura E3.3 Procedimiento de cómputo para el método de la falsa posición

después de terminar el renglón para $i = 2$, se examinan los signos de a, b y c; de nuevo, la raiz se localiza entre b y c. Por lo tanto, se repite el mismo proceso para $i = 3$ e incluso para $i = 4$.

En el renglón correspondiente a $i = 4$, la raiz se localiza entre a y b, por lo que tanto estos valores como $f(a)$ y $f(b)$ se copian en el siguiente renglón, para $i = 5$. Sin embargo, $f(a)$ no se divide a la mitad porque es la primera vez que a permanece fijo. Después de terminar el cálculo de b y $f(b)$ para $i = 6$, se copian b y c para $i = 7$ y $f(c)$ se divide entre 2 antes de copiarlo al renglón para el paso $i = 7$.

Después de modificar $f(a)$, se calcula el valor de b mediante

$$b = a - \frac{c - a}{f(c) - f^*(a)} f^*(a) \tag{3.3.5}$$

donde $f^*(a)$ es el valor modificado de $f(a)$.

Resumen de Esta Sección

a) El método de la falsa posición es esencialmente igual al método de la bisección, excepto que el segundo método se remplaza por la interpolación lineal.

b) El método de la falsa posición no necesariamente es más rápido que el método de bisección, debido a que un extremo puede permanecer fijo.

c) El método de la falsa posición modificada elimina los extremos fijos dividiendo a la mitad los valores de dichos puntos.

3.4 METODO DE NEWTON

Este método (también llamado método de Newton-Raphson) encuentra una raíz, siempre y cuando se conozca una estimación inicial para la raíz deseada. Utiliza las rectas tangentes que se evalúan analíticamente. El método de Newton se puede aplicar al dominio complejo para hallar raíces complejas. También se puede extender a las ecuaciones no lineales simultáneas (véase una aplicación en la sección 3.7).

El método de Newton se obtiene a partir del desarrollo de Taylor [Press/Flannery/Teukolsky/Vetterling; Cheney/Kincaid]. Supóngase que el problema es encontrar una raíz de $f(x) = 0$. Al utilizar el desarrollo de Taylor de $f(x)$ en torno a una estimación x_0, la ecuación se puede escribir como

$$f(x) = 0 = f(x_0) + f'(x_0)(x - x_0) + 0(h^2) \tag{3.4.1}$$

donde $h = x - x_0$. Al despejar x en la ecuación (3.4.1) no se obtiene el valor exacto debido al error de truncamiento, pero la solución se acerca en mayor medida al x exacto, que lo que se aproxima el estimado x_0. Por lo tanto, al repetir la solución utilizando el valor actualizado como una nueva estimación, se mejora la aproximación en forma sucesiva.

El algoritmo se muestra de manera gráfica en la figura 3.8. El valor x_0 es una estimación inicial para la raíz. A continuación se obtiene la función lineal que pasa por (x_0, y_0) en forma tangencial. La intersección de la recta tangente con el eje x se denota como x_1 y se considera como una aproximación de la raíz. Se repite el mismo procedimiento, utilizando el valor más actualizado como una estimación para el siguiente ciclo de iteración.

La recta tangente que pasa por $(x_0, f(x_0))$ es

$$g(x) = f'(x_0)(x - x_0) + f(x_0) \tag{3.4.2}$$

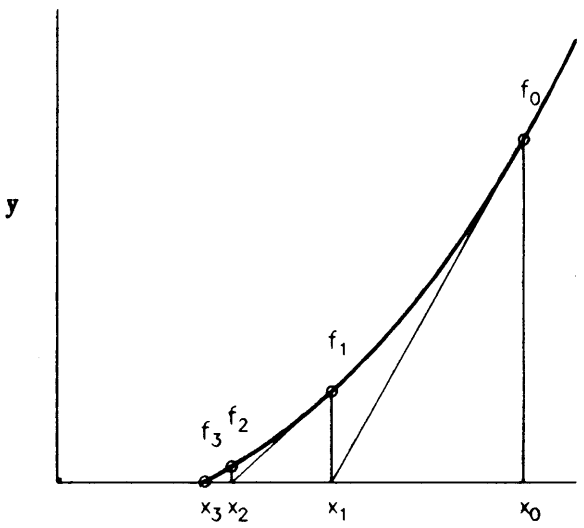


Figura 3.8 Método de Newton

La raíz de $g(x) = 0$ denotada por x_1 satisface

$$f'(x_0)(x_1 - x_0) + f(x_0) = 0$$

Al resolver la ecuación anterior se obtiene

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \tag{3.4.3}$$

Las aproximaciones sucesivas a la raíz se escriben como

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})} \tag{3.4.4}$$

Obtener la primera derivada de una función dada puede ser una tarea difícil o imposible. En tal caso, se puede evaluar $f'(x_i)$ en la ecuación (3.4.4) mediante una aproximación por diferencias, en vez de la forma analítica. Por ejemplo, se puede aproximar $f'(x_{i-1})$ mediante la aproximación por diferencias hacia adelante,

$$f'(x_{i-1}) = \frac{f(x_{i-1} + h) - f(x_{i-1})}{h} \tag{3.4.5}$$

donde h es un valor pequeño —como $h = 0.001$— o mediante la aproximación por diferencias hacia adelante, por

$$f'(x_{i-1}) = \frac{f(x_{i-1}) - f(x_{i-1} - h)}{h} \tag{3.4.6}$$

Los errores pequeños en la aproximación por diferencias no tienen un efecto observable en la razón de convergencia del método de Newton. La precisión del resultado final no se ve afectada por la aproximación por diferencias. Si la función no tiene singularidades en la vecindad de la raíz, ambas aproximaciones por diferencias funcionan bien. Sin embargo, debemos elegir una u otra si existe una singularidad cercana. (En el capítulo 5 se analizan más las aproximaciones numéricas a las derivadas.)

Como se indicó, el método de Newton se puede aplicar para hallar raíces complejas. Si el lenguaje de programación permite variables complejas, se puede aplicar fácilmente al caso de las raíces complejas un programa de computadora diseñado sólo para raíces reales, el PROGRAMA 3-7 está escrito en FORTRAN y puede encontrar raíces complejas. El método de Newton se puede aplicar a un conjunto de ecuaciones no lineales; en la sección 3.7 se proporciona un ejemplo. El método de Newton de segundo orden [James/Smith/Wolford] se obtiene utilizando el término de segundo orden en la ecuación (3.4.1). Converge más rápido que el método de Newton estándar analizado en esta sección, pero se paga el precio de evaluar la segunda derivada.

Ejemplo 3.4

Obtenga un esquema iterativo para encontrar la raíz cúbica de un número, basándose en el método de Newton. Determine la raíz cúbica de $a = 155$ mediante el esquema obtenido.

(Solución)

Suponga que deseamos calcular $x = \sqrt[3]{a}$. Este problema se puede formular de nuevo como determinar el cero de la función dada por

$$f(x) = x^3 - a$$

Por el método de Newton, se escribe un esquema iterativo como

$$\begin{aligned} x_{n+1} &= x^n - \frac{f(x_n)}{f'(x_n)} \\ &= x_n - \frac{x_n^3 - a}{3x_n^2} \\ &= \frac{2}{3}x_n + \frac{a}{3x_n^2} \end{aligned}$$

Para calcular la raíz cúbica de 155, definimos $a = 155$ y la estimación inicial $x_0 = 5$. Se obtienen los siguientes resultados iterativos

n	x
0	5
1	5.4
2	5.371834
3	5.371686 (exacto)

La solución exacta se obtiene sólo hasta después de tres pasos de iteración. Empecemos de nuevo con una estimación más pobre de $x_0 = 10$:

n	x
0	10
1	7.183334
2	5.790176
3	5.401203
4	5.371847
5	5.371686 (exacto)

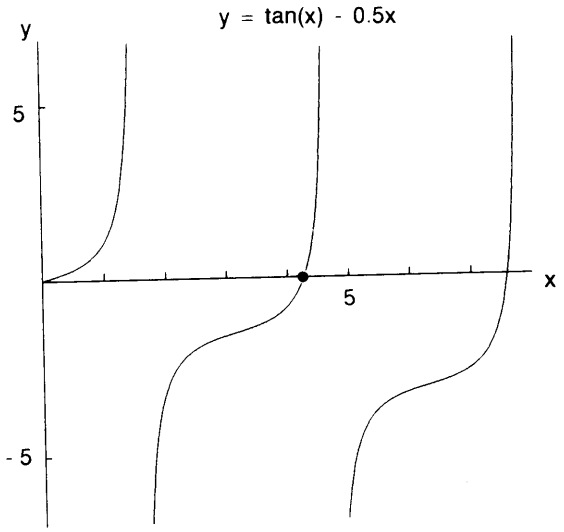
El valor exacto de la raíz cúbica se obtiene con cinco pasos de iteración.

Ejemplo 3.5

Calcule la raíz positiva más pequeña de $y = \tan(x) - 0.5x$ mediante el método de Newton.

(Solución)

La gráfica de y en la figura E3.5 muestra la mínima raíz positiva en una vecindad de 4.5, o entre 4 y $3\pi/2$. Aunque la expresión analítica de la primera



● : La menor raíz positiva a encontrar

Figura E3.5 Gráfica de $y = \tan(x) - 0.5x$

derivada de tan (x) se puede obtener fácilmente, utilizamos la ecuación (3.4.6) con $h = 0.001$. Así, aproximamos y' para tan (x) — 0.5 x mediante

$$y'(x) \simeq [\tan(x) - \tan(x - 0.001)]/0.001 - 0.5$$

El método de Newton con una estimación inicial de 4.0 se muestra en la tabla E3.5a.

```
CSL/F3-5        ESQUEMA DE NEWTON

TOLERANCIA?
0.0001
ESTIMACION INICIAL DE LA RAIZ?
4
 IT.NO.      X(N-1)          Y(N-1)          X(N)
    1      4.000000E+00  -8.421787E-01  4.458280E+00
    2      4.458280E+00   1.621111E+00  4.352068E+00
    3      4.352068E+00   4.781129E-01  4.288511E+00
    4      4.288511E+00   7.190108E-02  4.275191E+00
    5      4.275191E+00   2.075195E-03  4.274782E+00
    6      4.274782E+00  -2.861023E-06  4.274782E+00
--------------------------------
 SOLUCION FINAL =    4.274782
--------------------------------
```

Tabla E3.5a

Este problema es muy sensible a la elección de una estimación inicial. Si dicha estimación inicial se hace igual a 3.6, por ejemplo, la iteración converge a un valor irrelevante después de que los valores de x varían de forma errática, como lo muestra la tabla E3.5b.

```
CSL/F3-5        ESQUEMA DE NEWTON

TOLERANCIA?
0.0001
ESTIMACION INICIAL DE LA RAIZ ?
3.6
 IT.NO. N    X(N-1)          Y(N-1)          X(N)
    1      3.600000E+00 -1.306533E+00  5.358891E+00
    2      5.358891E+00 -4.004476E+00  7.131396E+00
    3      7.131396E+00 -2.431464E+00  8.494651E+00
    4      8.494651E+00 -5.588555E+00  1.092057E+01
    5      1.092057E+01  7.847680E+00  1.087581E+01
    6      1.087581E+01  2.872113E+00  1.083419E+01
    7      1.083419E+01  7.255301E-01  1.081511E+01
    8      1.081511E+01  7.328224E-02  1.081269E+01
    9      1.081269E+01  6.022453E-04  1.081267E+01
--------------------------------
 SOLUCION FINAL =    10.81267
--------------------------------
```

Tabla E3.5b

Resumen de Esta Sección

a) El método de Newton utiliza de forma iterativa las rectas tangentes que pasan por las aproximaciones consecutivas de la raíz.

b) El método requiere una buena estimación inicial. De otro modo, la solución iterativa puede diverger o converger a una solución irrelevante.

c) La razón de convergencia iterativa del método de Newton es alta, cuando funciona.

d) El método de Newton puede encontrar raíces complejas si las variables se definen como complejas.

3.5 METODO DE LA SECANTE

Este método es muy similar al de Newton. La principal diferencia con el método de Newton es que f' se aproxima utilizando los dos valores de iteraciones consecutivas de f. Esto elimina la necesidad de evaluar tanto a f como a f' en cada iteración. Por lo tanto, el método de la secante es más eficiente, particularmente cuando f es una función en la que se invierte mucho tiempo al evaluarla. El método de la secante también está íntimamente ligado con el método de la falsa posición, ya que ambos se basan en la fórmula de interpolación lineal, pero el primero utiliza extrapolaciones, mientras que el segundo utiliza únicamente interpolaciones [Press/et al.].

Las aproximaciones sucesivas para la raíz en el método de la secante están dadas por

$$x_n = x_{n-1} - y_{n-1}\frac{x_{n-1} - x_{n-2}}{y_{n-1} - y_{n-2}}, \quad n = 2, 3 \ldots \tag{3.5.1}$$

donde x_0 y x_1 son dos suposiciones iniciales para comenzar la iteración.

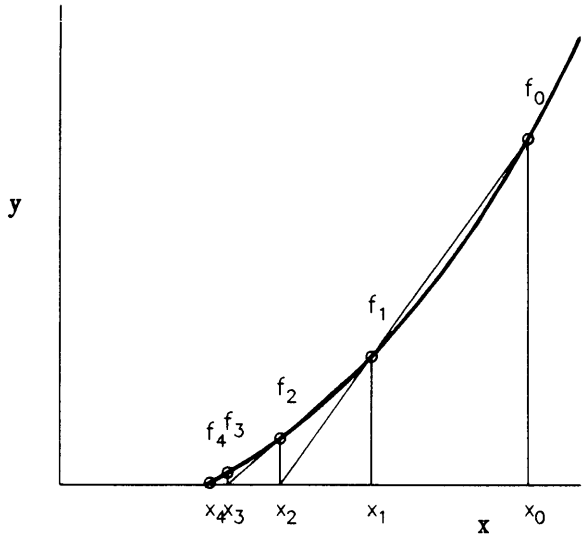


Figura 3.9 Método de la secante

Si los x_{n-1} y x_n consecutivos son muy cercanos, entonces también y_{n-1} y y_n están muy cercanos, por lo que aparece un error de redondeo significativo en la ecuación (3.5.1). Este problema se puede evitar de dos formas: a) cuando y_n es menor que un valor fijado de antemano, x_{n-2} y y_{n-2} en la ecuación (3.5.1) quedan fijos (o congelados) de ahí en adelante, o b) x_{n-2} y y_{n-2} se remplazan por $x_{n-2} + \xi$ y $y(x_{n-2} + \xi)$ donde ξ es un número pequeño prescrito pero lo suficientemente grande como para evitar serios errores de redondeo. El método de la secante puede converger a una raíz no deseada o puede no converger del todo si la estimación inicial no es buena.

Ejemplo 3.6

Un proyectil de $M = 2$ gm ha sido lanzado verticalmente al aire y está descendiendo a su velocidad terminal [Shames, pág. 417]. La velocidad terminal se determina mediante $gM = F_{drag}$ donde g es la gravedad y M es la masa; toda la ecuación se puede escribir, después de evaluar todas las constantes, como

$$\frac{(2)(9.81)}{1000} = 1.4 \times 10^{-5} v^{1.5} + 1.15 \times 10^{-5} v^2$$

donde y es la velocidad terminal en m/seg. El primer término del lado derecho representa la fuerza de fricción y el segundo término representa la fuerza de presión. Determinar la velocidad terminal por medio del método de la secante. Una estimación imperfecta está dada por $y \simeq 30$ m/seg.

(Solución)

El problema está definido como la determinación de la raíz de

$$y = f(v) = \frac{(2)(9.81)}{1000} - 1.4 \times 10^{-5} v^{1.5} - 1.15 \times 10^{-5} v^2 \qquad \text{(A)}$$

Hacemos $y_0 = 30$ y $y_0 = 30.1$ con base en la estimación imperfecta, para los que se evalúan y_0 y y_1 mediante la ecuación (A). La solución iterativa según la ecuación (3.5.1) es como sigue:

n	v_n	y_n
0	30.00000	1.9620001E − 02
1	30.10000	6.8889391E − 03
2	30.15411	6.8452079E − 03
3	38.62414	−8.9657493E − 04
4	37.64323	9.0962276E − 05
5	37.73358	9.9465251E − 07
6	37.73458	−1.8626451E − 09

Así, la velocidad terminal es $v = 37.7$ m/seg.

Resumen de Esta Sección

a) El método de la secante es una variación del método de Newton. Desde el punto de vista computacional, es más eficiente que el método de Newton.
b) Sin embargo, si dos aproximaciones sucesivas están demasiado cercanas, pueden aparecer errores de redondeo. Se han sugerido dos formas para prevenir los problemas por errores de redondeo.

3.6 METODO DE SUSTITUCION SUCESIVA

Si la ecuación $f(x) = 0$ se rearregla en la forma

$$x = \bar{f}(x) \tag{3.6.1}$$

entonces se puede escribir un método iterativo como

$$x^{(t)} = \bar{f}(x^{(t-1)}) \tag{3.6.2}$$

donde el índice t es el número de pasos en la iteración y $x^{(0)}$ es una estimación inicial. Este método se llama *método de sustitución sucesiva* o *iteración de punto fijo* [Conte/de Boor].

La ventaja de este método consiste en su gran sencillez y flexibilidad para elegir la forma de $\bar{f}$. Sin embargo, la desventaja es que la iteración no siempre converge con cualquier forma elegida de $\bar{f}(x)$. Para garantizar la convergencia de la iteración, se debe satisfacer la siguiente condición:

$$|\bar{f}'(x)| < 1 \text{ en la vecindad de la raíz} \tag{3.6.3}$$

La figura 3.10 muestra cómo afecta $\bar{f}'(x)$ la convergencia del método iterativo. Se puede observar que la convergencia es asintótica si $0 < \bar{f}' < 1$ y oscilatoria si -1

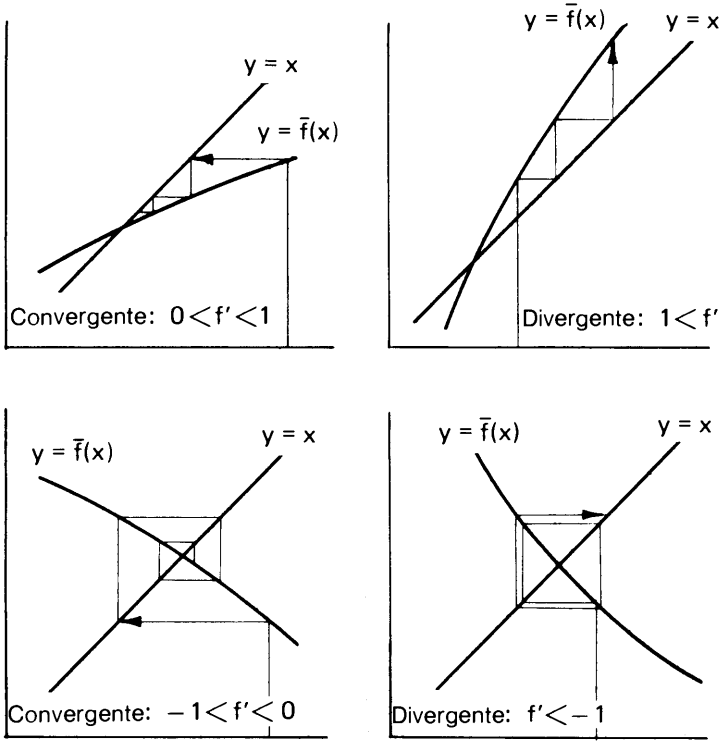


Figura 3.10 Convergencia del método de sustitución

$< \bar{f}' < 0$. Además, se puede mostrar fácilmente que la razón de convergencia es más rápida si f' tiende a cero en la vecindad de la raíz.

Ejemplo 3.7

Se sabe que la función $y = x^2 - 3x + e^x - 2$ tiene dos raíces: una negativa y otra positiva. Hallar la menor de éstas mediante el método de sustitución sucesiva.

(Solución)

Verificamos el signo de y en $x = -1$ y $x = 0$ (a saber, $y(-1) = 2.367$ y $y(0) = -1$) para localizar la raíz menor en $[-1, 0]$. Reescribimos la ecuación anterior como

$$x = \bar{f}(x) = \frac{x^2 + e^x - 2}{3} \qquad \text{(A)}$$

se puede escribir un método iterativo como

$$x^{(t)} = \bar{f}(x^{(t-1)}) \qquad \text{(B)}$$

La primera derivada de $f(x)$ satisface la ecuación (3.6.3) en el rango de $[-1, 0]$, por lo que el método anterior es convergente. Los valores numéricos de la iteración se dan a continuación:

Contador de iteraciones n	Aproximación sucesiva x_n
0	0 (estimación inicial)
1	−0.333333
2	−0.390786
3	−0.390254
4	−0.390272
5	−0.390272

Las ecuaciones alternativas son

$$x = -\sqrt{3x - e^x + 2} \tag{C}$$

y

$$x = \sqrt{3x - e^x + 2} \tag{D}$$

Sin embargo, las ecuaciones anteriores tienen discontinuidades en la vecindad de la raíz menor. Además, las primeras derivadas de ambas ecuaciones violan la condición de la ecuación (3.6.3) en la vecindad de la raíz. Por lo tanto, ninguna de las ecuaciones funciona.

Un camino sistemático para encontrar una forma de $\bar{f}(x)$ es hacer que

$$\bar{f}(x) = x - \alpha f(x) \tag{3.6.4}$$

por lo que el esquema iterativo queda como

$$x_n = x_{n-1} - \alpha f(x_{n-1}) \tag{3.6.5}$$

donde α es una constante. Si la iteración converge, el valor x obtenido mediante el esquema anterior satisface $f(x) = 0$. La constante α puede determinarse como sigue. Al sustituir la ecuación (3.6.4) en la ecuación (3.6.3), se ve que la iteración converge cuando

$$-1 < 1 - \alpha f'(x) < 1 \tag{3.6.6}$$

o, en forma equivalente,

$$0 < \alpha f'(x) < 2 \tag{3.6.7}$$

La ecuación (3.6.7) indica que, en primer lugar, α debe tener el mismo signo que f' y, en segundo lugar, la ecuación (3.6.5) siempre convergerá cuando α tienda a cero. La razón de convergencia es óptima cuando $\alpha \simeq 1/f'$.

El esquema actual se reduce al método de Newton si α se iguala a $1/f'(x_{n-1})$ para cada iteración.

Ejemplo 3.8

El tamaño critico de un reactor nuclear se determina resolviendo una ecuación de criticalidad [Lamarsh]. Supòngase que se da una versión sencilla de la ecuación de criticalidad como

$$\tan(0.1x) = 9.2e^{-x} \qquad \text{(A)}$$

La solución fisicamente significativa es la menor raiz positiva y se sabe que està en [3, 4] para la ecuación (A). Determine la minima raiz positiva.

(Solución)

Utilizamos el esquema iterativo de la ecuación (3.6.5) escribiendo

$$f(x) = \tan(0.1x) - 9.2e^{-x}$$

Se estima un valor aproximado de f' en [3, 4] como

$$f' = \frac{[f(4) - f(3)]}{(4 - 3)} = 0.40299$$

Por medio de la estimación anterior, se hace el parámetro α igual a $1/f' = 1/0.40299 = 2.4814$.

La iteración de la ecuación (3.6.5) converge de la manera siguiente:

Contador de iteraciones, n	x_n
0	4
1	3.36899
2	3.28574
3	3.29384
4	3.28280
5	3.29293
6	3.29292
7	3.29292

Resumen de Esta Sección

a) La sustitución sucesiva es una clase amplia de esquemas iterativos para encontrar una raiz de una función. El mètodo de Newton y el de la secante, descritos en la sección anterior, son casos especiales de la sustitución sucesiva.
b) Se ha analizado un criterio para la convergencia, de este mètodo.

3.7 METODO DE BAIRSTOW

Pocos mètodos se especializan en encontrar raices de polinomios casi automàticamente, entre èstos están el método de diferencias de cocientes [Gerald/Wheatley], el

método de Bairstow y la aplicación de la iteración QR. Aunque el método de diferencias de cociente es sencillo y fácil de usar, por desgracia falla muy a menudo. El uso de la iteración QR —el cual se explica en la sección 7.5— es el menor de los tres, pero no se puede utilizar sin comprender los valores propios de una matriz. El método de Bairstow tiene un problema de exactitud y a veces falla, pero es más confiable que el método de diferencias de cocientes.

El método de Bairstow es un esquema iterativo para encontrar un factor cuadrático de un polinomio en cada aplicación sin que se tenga ningún conocimiento previo. Al aplicar varias veces el método de Bairstow a los polinomios reducidos, se pueden calcular todos los factores cuadráticos de un polinomio.

Las raíces complejas de un polinomio con coeficientes reales siempre aparecen en parejas de complejos conjugados. Si un factor cuadrático, $x^2 + \bar{p}x + \bar{q}$ —que tiene una pareja de raíces conjugadas complejas— se factoriza del polinomio, la pareja de raíces complejas se puede calcular resolviendo $x^2 + \bar{p}x + \bar{q} = 0$. Así, se pueden calcular todas las raíces de un polinomio, sin utilizar variables complejas. Una desventaja del método de Bairstow es que la precisión de los resultados suele ser pobre, por lo que la precisión de las raíces calculadas se debe verificar o mejorar por algún otro medio, como el método de Newton.

Cualquier polinomio dado de orden N, escrito como

$$y = a_0 + a_1x + a_2x^2 + \cdots + a_Nx^N \tag{3.7.1}$$

se puede reescribir en la forma

$$y = (x^2 + px + q)G(x) + R(x) \tag{3.7.2}$$

donde p y q son valores arbitrarios, $G(x)$ es un polinomio de orden $N - 2$ y $R(x)$ es el residuo, que es un polinomio de orden 1, es decir, a lo más una función lineal. Si p y q se escogen de forma que el residuo $R(x)$ se anule, entonces $(x^2 + px + q)$ es un factor cuadrático. Las raíces de un factor cuadrático están dadas por la conocida fórmula

$$\frac{-p \pm \sqrt{p^2 - 4q}}{2}$$

Escribimos el polinomio de orden $N - 2$ y el residuo como

$$G(x) = b_2 + b_3x + b_4x^2 + \cdots + b_Nx^{N-2} \tag{3.7.3}$$

$$R(x) = b_0 + b_1x \tag{3.7.4}$$

respectivamente. Los valores de b_0 y b_1 dependen de los valores elegidos de p y q, por lo que se pueden considerar como funciones de p y q:

$$\begin{aligned} b_0 &= b_0(p, q) \\ b_1 &= b_1(p, q) \end{aligned} \tag{3.7.5}$$

Nuestro propósito es encontrar $p = \bar{p}$ y $q = q$ tales que $b_0(p, q) = b_1(p, q) = 0$, lo que $R(x) = 0$. Así $(x^2 + \bar{p}x + \bar{q})$ será un factor cuadrático.

Para obtener una fórmula explícita de la ecuación (3.7.5), sustituimos (3.7.3) y (3.7.4) y reescribimos la ecuación resultante como una serie de potencias. Puesto que la ecuación así obtenida debe ser igual a la ecuación (3.7.1), los coeficientes de iguales potencias de x en las dos ecuaciones deben ser idénticos. Al igualar los coeficientes para los mismos órdenes, encontramos las relaciones:

$$\begin{aligned}
a_N &= b_N \\
a_{N-1} &= b_{N-1} + pb_N \\
a_{N-2} &= b_{N-2} + pb_{N-1} + qb_N \\
&\vdots \\
a_2 &= b_2 + pb_3 + qb_4 \\
a_1 &= b_1 + pb_2 + qb_3 \\
a_0 &= b_0 \qquad\quad + qb_2
\end{aligned} \tag{3.7.6}$$

Al reescribir la ecuación (3.7.6), los coeficientes b_N hasta b_0 se pueden calcular en orden descendente como

$$\begin{aligned}
b_N &= a_N \\
b_{N-1} &= a_{N-1} - pb_N \\
b_{N-2} &= a_{N-2} - pb_{N-1} - qb_N \\
&\vdots \\
b_2 &= a_2 - pb_3 - qb_4 \\
b_1 &= a_1 - pb_2 - qb_3 \\
b_0 &= a_0 \qquad\quad - qb_2
\end{aligned} \tag{3.7.7}$$

Consideremos ahora p y q en la ecuación (3.7.5) como estimaciones arbitrarias para los valores exactos $\bar{p}$ y $\bar{q}$. Los términos de $b_0(\bar{p}, \bar{q})$ y $b_1(\bar{p}, \bar{q})$ se pueden desarrollar en una serie de Taylor alrededor de p y q:

$$b_0(\bar{p}, \bar{q}) = b_0(p, q) + \Delta p\left(\frac{\partial b_0}{\partial p}\right) + \Delta q\left(\frac{\partial b_0}{\partial q}\right) + \cdots \tag{3.7.8a}$$

$$b_1(\bar{p}, \bar{q}) = b_1(p, q) + \Delta p\left(\frac{\partial b_1}{\partial p}\right) + \Delta q\left(\frac{\partial b_1}{\partial q}\right) + \cdots \tag{3.7.8b}$$

donde

$$\Delta p = \bar{p} - p, \quad \Delta q = \bar{q} - q$$

y las derivadas parciales se evalúan en p y q. Obsérvese que los lados izquierdos de las ecuaciones (3.7.8a) y (3.7.8b) se anulan, debido a que $\overline{p}$ y $\overline{q}$ son valores exactos. Si truncamos los lados derechos de las ecuaciones (3.7.8a) y (3.7.8b) después de los términos de las derivadas de primer orden obtenemos

$$\Delta p\left(\frac{\partial b_0}{\partial p}\right) + \Delta q\left(\frac{\partial b_0}{\partial q}\right) = -b_0(p, q) \tag{3.7.9a}$$

$$\Delta p\left(\frac{\partial b_1}{\partial p}\right) + \Delta q\left(\frac{\partial b_1}{\partial q}\right) = -b_1(p, q) \tag{3.7.9b}$$

Los valores numéricos de los lados derechos de las ecuaciones (3.7.9a) y (3.7.9b) se evalúan mediante las últimas dos ecuaciones de la ecuación (3.7.7). Si se conocen las derivadas parciales, se pueden resolver las ecuaciones (3.7.9a) y (3.7.9b) para Δp y Δq.

Las derivadas parciales de las ecuaciones (3.7.9a) y (3.7.9b) se evalúan en forma recursiva, calculando las derivadas parciales de todas las ecuaciones que aparecen en la ecuación (3.7.7):

$$\begin{aligned}
(b_N)_p &= 0 \\
(b_{N-1})_p &= -b_N - p(b_N)_p \\
(b_{N-2})_p &= -b_{N-1} - p(b_{N-1})_p - q(b_N)_p \\
&\vdots \\
(b_2)_p &= -b_3 - p(b_3)_p - q(b_4)_p \\
(b_1)_p &= -b_2 - p(b_2)_p - q(b_3)_p \\
(b_0)_p &= \qquad\qquad\quad - q(b_2)_p
\end{aligned} \tag{3.7.10}$$

y

$$\begin{aligned}
(b_N)_q &= 0 \\
(b_{N-1})_q &= 0 \\
(b_{N-2})_q &= -b_N \\
&\vdots \\
(b_2)_q &= -p(b_3)_q - b_4 - q(b_4)_q \\
(b_1)_q &= -p(b_2)_q - b_3 - q(b_3)_q \\
(b_0)_q &= \qquad\qquad - b_2 - q(b_2)_q
\end{aligned} \tag{3.7.11}$$

donde los subíndices p y q denotan las derivadas parciales con respecto a p y q, respectivamente. Las últimas dos ecuaciones de (3.7.10) y las últimas dos ecuaciones de (3.7.11) dan los valores de las derivadas parciales en la ecuación (3.7.9).

Se hace una implantación del método de Bairstow como sigue:

a) Se efectúa una estimación inicial para p y q y se calculan b_0 y b_1 mediante la ecuación (3.7.7).
b) Se calculan $(b_0)_p$, $(b_1)_p$, $(b_0)_q$ y $(b_1)_q$ mediante las ecuaciones (3.7.10) y (3.7.11) (todas las ecuaciones deben evaluarse en forma recursiva).
c) Se resuelve la ecuación (3.7.9) en términos de Δp y Δq.
d) Se obtienen $\overline{p}$ y $\overline{q}$ mediante $\overline{p} = p + \Delta p$ y $\overline{q} = q + \Delta q$, respectivamente.

Se itera todo el proceso desde a) hasta d) utilizando los valores $\overline{p}$ y $\overline{q}$ del paso anterior como estimaciones actualizadas para p y q.

Una ventaja significativa del método de Bairstow es que en la mayoría de los problemas, la iteración converge a uno de los factores cuadráticos, independientemente de la estimación inicial para p y q, aunque a veces la iteración no converja. También se obtienen de manera automática los coeficientes del polinomio reducido $G(x)$. Así, para encontrar otro factor cuadrático, se puede aplicar nuevamente el método de Bairstow al polinomio reducido $G(x)$. Si se repite esto hasta que el orden del polinomio reducido sea menor que 2, se pueden encontrar todos los factores cuadráticos. Por otro lado, una desventaja es que la precisión de las raíces encontradas por el método quizá no sea buena. Por lo tanto, es recomendable mejorar la precisión aplicando el método de Newton para cada raíz. La precisión es pobre, particularmente si el polinomio tiene raíces múltiples. Se cuenta con el método de Bairstow mediante el PROGRAMA 3-7. Para obtener más información véase Isaacson/Keller; Shoup; Gerald/Wheatley.

La iteración QR —explicada en la sección 7.4— también se puede utilizar para encontrar las raíces de un polinomio.

Ejemplo 3.9

Por medio del PROGRAMA 3-7, encuentre los factores cuadráticos de la ecuación:

$$y = 3.3 + 0.5x + 2.3x^2 - 1.1x^3 + x^4$$

(Solución)

En la figura E3.9 se presenta la salida del PROGRAMA 3-7 para este problema. La salida muestra que: 1) $P = 0.9$ y $Q = 1.1$, por lo que el factor cuadrático es $x^2 + 0.9x + 1.1$; 2) las raíces del factor cuadrático son $-0.45 \pm 0.94736i$ y 3) el polinomio reducido es

$$x^2 - 2x + 3$$

el cual es otro factor cuadrático.

Los factores exactos son $(x^2 + 0.9x + 1.1)$ y $(x^2 - 2x + 3)$. Así, el primer factor cuadrático hallado es exacto, pero el segundo tiene cierto error. En general, el primer factor cuadrático hallado es el más preciso, mientras que los polinomios reducidos se vuelven cada vez menos precisos.

```
CSL/F3-7        ESQUEMA DE BAIRSTOW

ORDEN DEL POLINOMIO?
4
A( 0)?
3.3
A( 1)?
0.5
A( 2)?
2.3
A( 3)?
-1.1
A( 4)?
1
TOLERANCIA ?
0.00001
-----------------------------------------------------
P =     0.900000        Q =     1.100000
FACTOR CUADRATICO= X**2 + (   0.90000 X) + (   1.10000)

LAS RAICES DEL FACTOR CUADRATICO SON
  -0.450000 +     0.947365 I
  -0.450000 -     0.947365 I
COEFICIENTES DEL POLINOMIO REDUCIDO
        ORDEN    COEFICIENTES
         0     3.000000
         1    -2.000000
         2     1.000000
-----------------------------------------------------
```

Figura E3.9 Ejemplo de salida del PROGRAMA 3-7

Resumen de Esta Sección

a) El método de Bairstow encuentra un factor cuadrático de un polinomio, a partir del cual se calculan una pareja de raíces.

b) Puesto que las raíces complejas siempre aparecen como una pareja de complejos conjugados (cuando todos los coeficientes de un polinomio son raíces), se pueden calcular las raíces complejas sin álgebra compleja.

c) Al repetir la aplicación del método al polinomio reducido, se pueden hallar todos los factores cuadráticos.

d) Los errores de los polinomios reducidos y los factores cuadráticos aumentan al aplicar el método repetidamente.

e) La precisión de las raíces encontradas puede ser pobre, por lo que ésta debe mejorarse mediante otro método.

f) La iteración tal vez no converja para ciertos problemas.

PROGRAMAS

PROGRAMA 3-1 Método de bisección

A) Explicaciones

El PROGRAMA 3-1 encuentra una raíz de una ecuación mediante el método de bisección.

Antes de ejecutar el programa, hay que definir la ecuación a resolver en la función FUN, la cual tiene un polinomio cúbico como ejemplo en el listado que se muestra. Cuando el programa se ejecuta, se pide la entrada. Después de que el contador de la iteración se inicializa en cero, se encuentran los valores de y para x = A y x = C llamando a FUN, la cual detiene la ecuación a resolver: YA y YC son los valores de la función en x = A y x = C, respectivamente. Sin embargo, si F = 0 para x = A o x = B, el programa se detiene. Si el producto YA * YC es positivo, el programa se va a S-200 y se detiene después de imprimir un mensaje. Esto se debe a que no hay una raíz a encontrar cuando los signos de la función en los dos puntos extremos son iguales. Si el producto tiene un signo negativo en S-50, el programa pasa a S-60 donde el contador de las iteraciones IT se incrementa en uno. El punto medio B se calcula en S-70. Se encuentra el valor de la función para x = B y se guarda en YB. En S-90 se determina en cuál de los intervalos [A, B] o [B, C] se encuentra la raíz: Si el producto YA * YB no es positivo, el intervalo [A, B] contiene la raíz; en caso contrario, el intervalo [B, C] la contiene. De cualquier manera, el valor de C o A se actualiza en S-100 o S-110 y el programa regresa a S-60 para el siguiente paso de iteración.

B) Variables

A, C: valores de x en los puntos extremos actuales
EP: tolerancia
IL: máximo número de pasos de iteración permitidos
IT: contador de pasos de iteración
YA, YC: valores de la función en dos puntos extremos actuales
F: valor funcional en x

C) Listado

```
C-----CSL/F3-1.FOR      ESQUEMA DE BISECCION
      PRINT *
      PRINT *,'CSL/F3-1       ESQUEMA DE BISECCION
16    PRINT  *,'COTA INFERIOR: A ?   '
      READ   * ,A
      PRINT  *,'COTA SUPERIOR: C ?   '
      READ   * ,C
      PRINT  *,'TOLERANCIA: EP ?   '
      READ   *,EP                                  ! Tolerancia
      PRINT  *,'LIMITE DE ITERACIONES: IL ?   '
      READ   * ,IL                                 ! Límite de iteraciones
      PRINT  *,' IT.      A          B          C          F(A)       F(B)
     #      F(C)   ABS(F(C)-F(A))/2'

      IT=0                    ! Inicialización del contador de iteraciones
30    YA=FUN(A)
      YC=FUN(C)
50    IF (YA*YC .GT. 0) GOTO 200
60    IT=IT+1
70    B=(A+C)/2               ! B es el punto medio
      YB=FUN(B)
      PRINT  80,IT,A,B,C,YA,YB,YC,ABS(YC-YA)/2
```

```
80      FORMAT (I4, 3F9.4, 1X, 1P3E10.2, 2X, 1PE10.3)
        IF (IT .GT. IL) THEN      !  El número de iteraciones debe compararse con el límite
           PRINT  *,' SE HA EXCEDIDO EL LIMITE DE ITERACIONES
           GOTO 205
        ENDIF
71      IF (ABS(B-A) .LT. EP) THEN
           PRINT *,'            SE HA SATISFECHO LA TOLERANCIA '
           GOTO  205
        ENDIF
90      IF (YA*YB .LE. 0) THEN
           C=B
           YC=YB
        ELSE
110        A=B
           YA=YB
        ENDIF
           GOTO 60
C
200     PRINT  *,'YA*YC  ES POSITIVO '
        GOTO 210
205     PRINT  *
        PRINT  *,' RESULTADO FINAL:     RAIZ APROXIMADA =  ',B
        PRINT *
        PRINT *
210     PRINT *,' OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
        READ *,KS
        IF(KS.EQ.1) GOTO 16
        STOP
        END

C*******************
        FUNCTION FUN(X)        !  Define la función a resolver
        FUN=X*X*X-3*X*X-X+3
        RETURN
        END
```

D) Ejemplo de salida

```
CSL/F3-1       ESQUEMA DE BISECCION
COTA INFERIOR:  A ?
0
COTA SUPERIOR:  C ?
3
TOLERANCIA:  EP ?
0.0001
LIMITE DE ITERACIONES :  IL ?
20
```

IT.	A	B	C	F(A)	F(B)	F(C)	ABS(F(C)-F(A))/2
1	0.0000	1.5000	3.0000	3.00E+00	-1.88E+00	0.00E+00	1.500E+00
2	0.0000	0.7500	1.5000	3.00E+00	9.84E-01	-1.88E+00	2.438E+00
3	0.7500	1.1250	1.5000	9.84E-01	-4.98E-01	-1.88E+00	1.430E+00
4	0.7500	0.9375	1.1250	9.84E-01	2.50E-01	-4.98E-01	7.412E-01
5	0.9375	1.0313	1.1250	2.50E-01	-1.25E-01	-4.98E-01	3.739E-01
6	0.9375	0.9844	1.0313	2.50E-01	6.25E-02	-1.25E-01	1.874E-01
7	0.9844	1.0078	1.0313	6.25E-02	-3.12E-02	-1.25E-01	9.373E-02

```
 8   0.9844   0.9961   1.0078   6.25E-02  1.56E-02 -3.12E-02   4.687E-02
 9   0.9961   1.0020   1.0078   1.56E-02 -7.81E-03 -3.12E-02   2.344E-02
10   0.9961   0.9990   1.0020   1.56E-02  3.91E-03 -7.81E-03   1.172E-02
11   0.9990   1.0005   1.0020   3.91E-03 -1.95E-03 -7.81E-03   5.859E-03
12   0.9990   0.9998   1.0005   3.91E-03  9.76E-04 -1.95E-03   2.930E-03
13   0.9998   1.0001   1.0005   9.76E-04 -4.88E-04 -1.95E-03   1.465E-03
14   0.9998   0.9999   1.0001   9.76E-04  2.44E-04 -4.88E-04   7.323E-04
15   0.9999   1.0000   1.0001   2.44E-04 -1.22E-04 -4.88E-04   3.662E-04
          SE HA SATISFECHO LA TOLERANCIA

RESULTADO FINAL: RAIZ APROXIMADA = 1.000031
OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR
```

PROGRAMA 3-2 Búsqueda de raíces

A) Explicaciones

Este programa busca intervalos que contengan raíces de una función. La entrada consiste en los límites inferior y superior de x para la búsqueda y un tamaño del intervalo h. Los intervalos en donde cambia el signo de la función (que contienen una o un número impar de raíces) se imprimen. El programa FUNC define la ecuación a resolver y se puede cambiar para los problemas propios del lector.

Antes de ejecutar el programa, el usuario debe definir la ecuación a resolver en FUNC, que por el momento tiene el problema del ejemplo 3.2. Cuando se ejecuta el programa, la computadora pide en forma interactiva tres parámetros de entrada, A, B y H. Así, el valor de X se designa como el límite inferior para la búsqueda y el contador de intervalos es I = 0. En el subprograma se calcula el valor de la función para el valor actual de x. Para I = 1 el programa brinca a S-40. Para I > 1, se verifica el producto de Y y YB, donde YB es el valor de Y para el valor anterior de X. Si el producto es negativo, se imprimen los valores de X — H y X como un intervalo que contiene un número impar de raíces. En la línea siguiente a S-40, se remplaza YB por el actual Y; X se incrementa en H a continuación el programa va a S-20. El programa se detiene si X excede a B, el límite máximo.

B) Variables

A: límite inferior de x para la búsqueda
B: límite superior de x para la búsqueda
H: tamaño de los intervalos, h
Y: valor de la función en x
YB: valor de la función en $x - h$
I: contador de intervalos

C) Listado

```
C-----CSL/F3-2.FOR        BUSQUEDA DE RAICES
      print *
      PRINT *, 'CSL/F3-2    BUSQUEDA DE RAICES'
1     PRINT *
      PRINT *,'INICIAL X ? '
         READ *,A
      PRINT *,'FINAL X ? '
         READ *,B
      PRINT *,'INCREMENTO DE  X ?'
         READ *,H
         PRINT *
      I=0                        !  Inicialización del número de intervalos
      X=A
 20   I=I+1
      IF (X .GT. B) THEN
        GOTO 45
        ELSE
          Y = FUNC(X)
          IF (I.EQ.1.OR.Y*YB .GT. 0) GOTO 40
          PRINT *
          PRINT 90, X-H,X
        END IF
90    FORMAT(' UN INTERVALO QUE PUEDE CONTENER UNA RAIZ:   [',
     #       F10.6,','F10.6,']')
40    YB=Y
      X=X+H
      GOTO 20
 45   PRINT *
      PRINT *, '                    *** FIN DE LA BUSQUEDA '
      PRINT *
      PRINT *,'OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
      READ *,KS
      IF(KS.EQ.1) GO TO 1
      STOP
      END
C*********************
      FUNCTION FUNC(X)          !  Define la ecuación a resolver.
      FUNC = -19*(X-.5)*(X-1)+EXP(X)-EXP(-2*X)
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F3-2          BUSQUEDA DE RAICES

INICIAL X ?
-10
FINAL X ?
10
INCREMENTO DE  X ?
1
UN INTERVALO QUE PUEDE CONTENER UNA RAIZ:   [  0.000000,   1.000000]
UN INTERVALO QUE PUEDE CONTENER UNA RAIZ:   [  1.000000,   2.000000]
UN INTERVALO QUE PUEDE CONTENER UNA RAIZ:   [  6.000000,   7.000000]
                *** FIN DE LA BUSQUEDA
```

```
INICIAL X ?
0
FINAL X ?
1
INCREMENTO DE X ?
0.001
UN INTERVALO QUE PUEDE CONTENER UNA RAIZ:   [   0.405998,   0.406998]
                    *** FIN DE LA BUSQUEDA
```

PROGRAMA 3-3 Graficación de una función en BASIC

A) Explicaciones

Este programa está diseñado para graficar una función en la pantalla de una microcomputadora y está escrito en BASIC. Para ejecutar el programa, hay que definir la función en la línea 910. Al realizarse, el programa pide los datos de entrada como sigue:

Xmín: ¿mínimo de x de la figura?
Xmáx: ¿máximo de x de la figura?
Ymín: ¿mínimo de y de la figura?
Ymáx: ¿máximo de y de la figura?
M: ¿número de intervalos para graficar la curva?

La curva se grafica uniendo dos puntos consecutivos mediante líneas rectas. Si el valor de M dado en la entrada es pequeño, se dibujará una curva no lisa, pero si el número es muy grande se desperdiciará tiempo de cómputo. De cualquier forma, a veces será necesario un número muy grande, particularmente cuando una porción de la curva tenga una pendiente muy grande. El valor predeterminado para el número de intervalos es 50, por lo que si esta entrada se deja en blanco, se conectarán 50 puntos de la curva mediante líneas rectas.

B) Variables

A: cota izquierda de la abscisa
B: cota derecha de la abscisa
Ymín: cota inferior de la ordenada
Ymáx: cota superior de la ordenada
Y, X: coordenadas
PASOI: intervalo entre las marcas
M: número de intervalos; en cada uno de los cuales la curva se aproxima por una línea recta

C) Listado

```
1   PRINT: PRINT "CSL/B3-3              GRAFICACION   (BASIC, IBM PC)
2 :
11 PRINT
12 PRINT"*****************************************************************
13 PRINT" Para regresar a la pantalla normal después de graficar, oprima la tecla F10.
14 PRINT"*****************************************************************
15 PRINT
18 PRINT"INTRODUZCA A CONTINUACION LA FRONTERA IZQUIERDA, DERECHA, INFERIOR Y SUPERIOR
20 INPUT "Xmin ";A                                                  DE LA GRAFICA"
25 INPUT "Xmax ";B
30 INPUT "Ymin ";YMIN
35 INPUT "Ymax ";YMAX
40 INPUT "NUMERO DE INTERVALOS" ;M
60  DOTX=400: DOTY=250: SCREEN 0: SCREEN 2
70 PRINT "xmín=";A;" xmáx=";B:PRINT"ymín=";YMIN;" ymáx=";YMAX
80  SCREEN 2
90 DX=(B-A)/M :X=A:GOSUB 900: X0=X:Y0=Y
95 GOSUB 870
100  FOR X= A+DX TO B STEP DX
105    X1=X
110    GOSUB 900: Y1= Y
220    GOSUB 800
230    X0=X1:Y0=Y1
240    NEXT
250 X0=0:X1=0:Y0=YMIN:Y1=YMAX:GOSUB 800
260 Y0=0:Y1=0:X0=A    :X1=B    :GOSUB 800
265 :
266 PRINT: PRINT"Distancia entre las marcas de los intervalos:"
270 IX1= INT(A):IX2=INT(B):PASO=1: IF (IX2-IX1)>20 THEN PASO=10
275  FOR I= IX1 TO IX2 STEP PASO: X0=I:X1=I: Y0=0: Y1=(YMAX-YMIN)/50
276    GOSUB 800
277  NEXT :PRINT"  x:";ISTP
278 :
280 IX1= INT(YMIN):IX2=INT(YMAX):PASO=1: IF (IX2-IX1)>20 THEN ISTP=10
285  FOR I= IX1 TO IX2 STEP PASO: Y0=I:Y1=I: X0=0: X1=(B-A)/100
290    GOSUB 800
291  NEXT:PRINT"  y:"; PASO:PRINT
295 :
300 FOR WW=1 TO 10000:NEXT:STOP
310 :
800 REM---------- SUBRUTINA para trazar una línea
810 XX1=(X0-A)*AX+30
820 XX2=(X1-A)*AX+30
830 YY1=DOTY*1.1-(Y0-YMIN)*AY
840 YY2=DOTY*1.1-(Y1-YMIN)*AY: LINE (XX1,YY1)-(XX2,YY2)
850 RETURN
855 :
860 REM
870 AX=DOTX/(B-A)
880 AY=DOTY/(YMAX-YMIN): RETURN
885 :
900 REM ---------- SUBRUTINA para definir la función
910 Y=SIN(X): RETURN
```

D) Ejemplo de salida

```
CSL/B3-3            GRAFICACION  (BASIC)

*************************************************************************
 Si desea regresar a la pantalla normal después de graficar, oprima la tecla F10.
*************************************************************************

INTRODUZCA A CONTINUACION LA FRONTERA IZQUIERDA, DERECHA, INFERIOR Y SUPERIOR DE
                                                                    LA GRAFICA
Xmín ? -1
Xmáx ? 11
Ymín ? -5
Ymáx ? 5
¿Número de intervalos? 100
xmín=-1  xmáx= 11
ymín=-5  ymáx= 5

Distancia entre las marcas de los intervalos
 x: 1
 y: 1
```

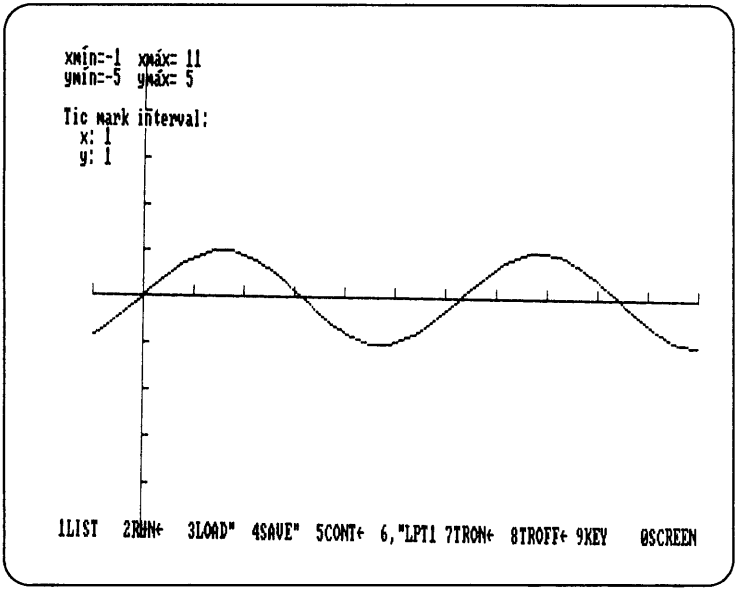


Figura 3.11 Ejemplo de una gráfica realizada por el PROGRAMA 3-3

PROGRAMA 3-4 Método de la falsa posición modificada

A) Explicaciones

Este programa encuentra la raíz de una función mediante el método de la falsa posición modificada.

El usuario debe definir la ecuación a resolver en el subprograma FUNC, la cual es actualmente $f(x) = \tan(x) - x - 0.5$. El flujo del cálculo es muy similar al del

PROGRAMA 3-1, excepto que YA o YC se dividen a la mitad cuando los contadores de extremos fijos, KI o KD, respectivamente, se vuelven mayores que uno.

B) Variables

Las mismas del PROGRAMA 3-1 excepto

KI: contador del extremo fijo izquierdo

KD: contador del extremo fijo derecho

C) Listado

```
C-----CSL/F3-4.FOR        ESQUEMA DE LA FALSA POSICION MODIFICADA
      PRINT *
      PRINT *,'CSL/F3-4.FOR    ESQUEMA DE LA FALSA POSICION MODIFICADA '
      PRINT *,'COTA INFERIOR: A ?'
         READ *,A
      PRINT *,'COTA SUPERIOR: C ?'
         READ *,C
      PRINT *,'TOLERANCIA: EP ?'
         READ *,EP
      PRINT *,'LIMITE DE ITERACIONES: IL ?'
         READ *,IL
      PRINT *
      YA=FUNC(A)
      YC=FUNC(C)
      PRINT 10
10    FORMAT
     #   (' IT.NO.',6X,'A',11X,'B',11X,'C',11X,'YA',10X,'YB',10X,'YC')
      IT=0
      KI=1
      KD=1
30    IT=IT+1
      IF (IT .GT. IL) THEN
         PRINT *,' SE HA EXCEDIDO EL LIMITE DE ITERACIONES'
         GO TO 110
      END IF
      GR=(YC-YA)/(C-A)
      BB=B
      B=-YA/GR+A
      YB= FUNC(B)
      PRINT 70,IT,A,B,C,YA,YB,YC
70    FORMAT(I3,3X,1P6E12.4)
C
      IF(ABS(BB-B) .LT. EP) GOTO 100
      IF(YA*YB .LE. 0) THEN
         YC=YB
         C=B
         KI=KL+1
         KD=0
         IF( KL .GT. 1) YA=YA/2
         GOTO 30
      ELSE
         YA=YB
         A=B
         KD=KD+1
         KI=0
         IF (KD .GT. 1) YC=YC/2
         GOTO 30
      END IF
```

```
100   PRINT *,'                                    SE HA SATISFECHO LA TOLERANCIA'
      PRINT *
      PRINT *,' ---------------------------------------------- '
110   PRINT *,'  RAIZ APROXIMADA =',B
      PRINT *,' ---------------------------------------------- '
      PRINT *
      STOP
      END
C**********************
      FUNCTION FUNC(X)    !-- Define la ecuación a resolver
      FUNC=TAN(X)-X-.5
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F3-4.FOR    ESQUEMA DE LA FALSA POSICION MODIFICADA
COTA INFERIOR: A ?
0
COTA SUPERIOR: C ?
1.5
TOLERANCIA: EP ?
0.0001
LIMITE DE ITERACIONES : IL ?
20

IT.NO.      A           B           C           YA          YB          YC
 1      0.0000E+00  5.9517E-02  1.5000E+00 -5.0000E-01 -4.9993E-01  1.2101E+01
 2      5.9517E-02  1.6945E-01  1.5000E+00 -4.9993E-01 -4.9836E-01  6.0507E+00
 3      1.6945E-01  3.5763E-01  1.5000E+00 -4.9836E-01 -4.8393E-01  3.0254E+00
 4      3.5763E-01  6.3451E-01  1.5000E+00 -4.8393E-01 -3.9846E-01  1.5127E+00
 5      6.3451E-01  9.3315E-01  1.5000E+00 -3.9846E-01 -8.3427E-02  7.5634E-01
 6      9.3315E-01  1.0356E+00  1.5000E+00 -8.3427E-02  1.5097E-01  3.7817E-01
 7      9.3315E-01  9.6961E-01  1.0356E+00 -8.3427E-02 -1.1622E-02  1.5097E-01
 8      9.6961E-01  9.7433E-01  1.0356E+00 -1.1622E-02 -1.4935E-03  1.5097E-01
 9      9.7433E-01  9.7552E-01  1.0356E+00 -1.4935E-03  1.0919E-03  7.5485E-02
10      9.7433E-01  9.7502E-01  9.7552E-01 -1.4935E-03 -1.6689E-06  1.0919E-03
11      9.7502E-01  9.7502E-01  9.7552E-01 -1.6689E-06  1.7881E-07  1.0919E-03
                                          SE HA SATISFECHO LA TOLERANCIA
 ---------------------------------------------
  RAIZ APROXIMADA=  0.9750172
 ---------------------------------------------
```

PROGRAMA 3-5 Método de Newton

A) Explicaciones

Este programa calcula una raíz real con una estimación inicial.

La ecuación a resolver y su primera derivada se definen en el subprograma FUNC. Se da como entrada la estimación inicial para x. En S-20, el contador de iteración se incrementa en uno. En cada iteración se encuentran X y XD llamando a FUNC, y a continuación se actualiza X mediante el método de Newton. La iteración

termina si la diferencia entre dos valores consecutivos de x es menor que la tolerancia especificada como entrada; el programa se detiene.

B) Variables

X: valor de x
XB: valor anterior de x
Y: valor de y para el valor actual de x
YD: y' para el valor actual de x
I: contador de pasos de iteración

C) Listado

```
C-----CSL/F3-5.FOR        ESQUEMA DE NEWTON
C   LA ECUACION A RESOLVER Y SU DERIVADA SE DEFINEN
C   EN LA SUBRUTINA FUNC
      PRINT*
      PRINT*,'CSL/F3-5    ESQUEMA DE NEWTON'
      PRINT*
      PRINT*,'TOLERANCIA ?'
      READ *,EP
      PRINT *
5     PRINT*,'ESTIMACION INICIAL PARA LA RAIZ ?'
      READ *, X
      XB=X
      I=0
      PRINT *
      PRINT *,' IT.NO. N    X(N-1)          Y(N-1)          X(N)'
20    I=I+1
      CALL FUNC(X,Y,YD)
      X = X - Y/YD                  !-- Esquema de Newton: encuentra la nueva x.
      PRINT 30, I,XB,Y,X
30    FORMAT(1X,I5,3X,1P4E14.6)
      IF (ABS(X-XB).GE.EP) THEN     !-- Prueba de convergencia
         XB=X
         GO TO 20
         END IF
      PRINT *
40    PRINT*,'--------------------------------'
      PRINT*,'  SOLUCION FINAL=',X
      PRINT*,'--------------------------------'
      PRINT *
      PRINT*
      PRINT*,'PARA CONTINUAR, OPRIMA 1'
      READ *,K
      IF(K.EQ.1) GOTO 5
      PRINT*
      END
C***************************
      SUBROUTINE FUNC(X,Y,YD)     !-- Calcula y y y'.
      Y=X**3 - 5.0*X**2 + 6.*X
      YD=3.0*X**2-10.0*X + 6.
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F3-5        ESQUEMA DE NEWTON

TOLERANCIA?
0.00001
ESTIMACION INICIAL PARA LA RAIZ?
4.0
 IT.NO.  N    X(N-1)          Y(N-1)          X(N)
    1      4.000000E+00  8.000000E+00  3.428571E+00
    2      3.428571E+00  2.099123E+00  3.127820E+00
    3      3.127820E+00  4.508972E-01  3.017077E+00
    4      3.017077E+00  5.240059E-02  3.000376E+00
    5      3.000376E+00  1.127243E-03  3.000000E+00
    6      3.000000E+00  1.907349E-06  3.000000E+00
--------------------------------
 SOLUCION FINAL =    3.000000
--------------------------------

ESTIMACION INICIAL PARA LA RAIZ ?
1.4
 IT.NO.  N    X(N-1)          Y(N-1)          X(N)
    1      1.400000E+00  1.344000E+00  2.033962E+00
    2      2.033962E+00 -6.673241E-02  1.999361E+00
    3      1.999361E+00  1.277924E-03  2.000000E+00
    4      2.000000E+00  9.536743E-07  2.000000E+00
--------------------------------
  SOLUCION FINAL=    2.000000
--------------------------------
```

PROGRAMA 3-6 Método de Newton para raíces complejas

A) Explicaciones

Esta es otra versión del método de Newton para determinar las raíces de un polinomio. Puesto que se utiliza el álgebra compleja, este programa también puede calcular raíces complejas. Sin embargo, para encontrar una raíz compleja, hay que dar un valor complejo como estimación inicial.

El orden del polinomio (N) y los coeficientes del polinomio se definen en la instrucción DATA. La entrada interactiva de los datos necesarios para cada ejecución son las partes real e imaginaria de la estimación inicial.

B) Variables

N: orden del polinomio
A(I): coeficientes de los términos de un polinomio
X: variable independiente (valor complejo)
F: valor de la función (variable compleja)
FD: derivada (variable compleja)

C) Listado

```
C-----CSL/F3-6.FOR      METODO DE NEWTON PARA ENCONTRAR LAS RAICES COMPLEJAS
C                       DE UN POLINOMIO
C
      COMPLEX F,FD,X,XB
      DIMENSION A(10)
      DATA N/2/                          !-- N es el orden del polinomio
      DATA A/4.0,-1.0,1.0, 7*0.0/        !-- Coeficientes de las potencias
      PRINT *
      PRINT *,'CSL/F3-6'
      PRINT *,'METODO DE NEWTON PARA ENCONTRAR LAS RAICES COMPLEJAS DE UN
      PRINT*                                                   POLINOMIO'
      NP=N+1
      PRINT *,' ESTIMACION INICIAL PARA X:'
1     PRINT *,'--- ¿PARTE REAL? '
      READ *, XR
      PRINT *,'--- PARTE IMAGINARIA (DISTINTA DE CERO) ? '
      READ *, XI
      X=CMPLX(XR,XI)
      IT=0                               !-- Se inicializa el contador de las iteraciones
      PRINT *
      PRINT *,'-------------------------------------------------------'
      PRINT *,' IT.NO.          X                          FUNCION '
      PRINT *,'-------------------------------------------------------'
30    IT=IT+1
          XB=X
          CALL FFD(N,A,X,F,FD)
          X=X-F/FD                       !-- Actualiza x mediante el método de Newton
          PRINT 40,IT,XB,F
40        FORMAT( I4, '  (',1P2E12.5,')  (',1P2E11.4,')')
          IF(CABS(X-XB).LT.0.00001) GOTO 60
          IF(IT.GT.50) GOTO 60
50    GO TO 30
60    PRINT *,'-------------------------------------------------------'
      PRINT*,'RESULTADO FINAL =',X
      PRINT *,'-------------------------------------------------------'
      PRINT*
      PRINT*,' OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
      READ *,K
      IF (K.EQ.1) THEN
         PRINT*
         PRINT*,' SIGUIENTE ESTIMACION DE X? '
         GOTO 1
      END IF
      PRINT *
      END
C*****************************
      SUBROUTINE FFD(N,A,X,F,FD)         !-- Encuentra los valores de f y f'para x
      COMPLEX F,FD,X
      DIMENSION A(1)
      F=CMPLX(A(1),0.0)
      FD=CMPLX(0.0,0.0)
      DO 10 I=1,N
          F=F+CMPLX(A(I+1),0.0)*X**I
          C=A(I+1)*FLOAT(I)
          FD=FD+CMPLX(C,0.0)*X**(I-1)
   10 CONTINUE
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F3-6
METODO DE NEWTON PARA ENCONTRAR LAS RAICES COMPLEJAS DE UN POLINOMIO

ESTIMACION INICIAL PARA X:
---PARTE REAL?
1.0
---PARTE IMAGINARIA DISTINTA DE CERO) ?
1.0

-------------------------------------------------------
 IT.NO.          X                          FUNCION
-------------------------------------------------------
  1   ( 1.00000E+00 1.00000E+00)   ( 3.0000E+00 1.0000E+00)
  2   ( 0.00000E+00 2.00000E+00)   ( 0.0000E+00-2.0000E+00)
  3   ( 4.70588E-01 1.88235E+00)   ( 2.0761E-01-1.1073E-01)
  4   ( 5.00854E-01 1.93703E+00)   (-2.0733E-03 3.3094E-03)
  5   ( 5.00000E-01 1.93649E+00)   ( 7.1526E-07 9.5367E-07)
-------------------------------------------------------
RESULTADO FINAL = (0.5000000,1.936492)
-------------------------------------------------------
```

PROGRAMA 3-7 Método de Bairstow

A) Explicaciones

Este programa calcula un factor cuadrático de un polinomio y en seguida encuentra las raíces del factor cuadrático.

Todos los datos de entrada se dan en forma interactiva. Los datos de entrada necesarios incluyen el orden del polinomio, los coeficientes del polinomio en orden creciente de potencias, así como la tolerancia de convergencia.

La parte principal del programa comprende desde S-70 hasta S-300. Antes de entrar a esta parte por primera vez, las estimaciones iniciales para p y q son cero. (Dichas estimaciones iniciales pueden cambiarse a otros valores, particularmente si aparece un error aritmético.) En S-300 se verifica la convergencia de la iteración: si la suma de los valores absolutos de Δp y Δq son mayores que la tolerancia, el programa regresa a S-70 para repetir los cálculos con los valores revisados de p y q. Si la prueba de convergencia es positiva, el programa imprime el resultado final.

Si aparece en S-280 "división entre cero" (un incidente raro pero impredecible), se vuelve a ejecutar el programa con un conjunto diferente de estimaciones iniciales para p y q.

B) Variables

A(I): coeficientes del polinomio, a_i (entrada)

B(I): coeficientes b_i en las ecuaciones (3.7.3) y (3.7.4)

BP(I), BQ(I): $(b_i)_p$ y $(q_i)_q$, respectivamente

TL: tolerancia de la convergencia

N: orden del polinomio (entrada)

P, Q: p y q, respectivamente

DN: determinante de los coeficientes de la ecuación (3.7.9)

DP, DQ: Δp y Δq, respectivamente

C) Listado

```
C-----CSL/F3-7.FOR      ESQUEMA DE BAIRSTOW

      DIMENSION A(0:10),B(0:10),BP(0:10),BQ(0:10)
      PRINT *
      PRINT *,'CSL/F3-7     ESQUEMA DE BAIRSTOW '
1     DO 10 I=0,10
         A(I)=0
         B(I)=0
         BP(I)=0
         BQ(I)=0
10    CONTINUE
      PRINT *
      PRINT *,'ORDEN DEL POLINOMIO ?'
      READ *,N                                  !  Lee los coeficientes
      DO 15 I=0, N
         WRITE (6,20) I
         READ *, A(I)
15    CONTINUE
20    FORMAT (' A(', I2,')?')
      PRINT *,'TOLERANCIA ?'
      READ (5,*) TL
      P=0          !  Inicializa P y Q (valores arbitrarios)
      Q=0
70    DO 78 I=N,1,-1                            !  Comienza el esquema de Bairstow
         B(I)=A(I)-P*B(I+1)-Q*B(I+2)
78    CONTINUE
32    B(0)=A(0)-Q*B(2)
      DO 140 I=N,1,-1
         BP(I) = -B(I+1)-P*BP(I+1)-Q*BP(I+2)
         BQ(I)=-P*BQ(I+1)-B(I+2)-Q*BQ(I+2)
140   CONTINUE
      BP(0)=-Q*BP(2)
      BQ(0)=-Q*BQ(2)-B(2)
      DN= BP(0)*BQ(1)-BP(1)*BQ(0)
      DP=(B(0)*BQ(1)-B(1)*BQ(0))/DN
      P=P-DP
      DQ=(B(1)*BP(0)-B(0)*BP(1))/DN
      Q=Q-DQ
300   IF (ABS(DQ)+ABS(DP).GT.TL) GOTO 70
       PRINT *                                  !   Pasa la prueba de convergencia
       PRINT *,'-------------------------------------------------------'
       PRINT 310,P,Q
310    FORMAT(' P =',F12.6, '          Q = ',F12.6)
       PRINT *
       PRINT 340, P,Q
340    FORMAT(' FACTOR CUADRATICO = X~2 + (',F10.5,' X) + (',F10.5,')')
       PRINT *
       PRINT *,' LAS RAICES DEL FACTOR CUADRATICO SON '
       ZZ=P*P-4*Q
```

```
      IF (ZZ.GE.0) THEN
        RT=SQRT(ZZ)
        PRINT 350,(-P+RT)/2,  (-P-RT)/2     ! Imprime una pareja de raíces reales
350     FORMAT( 2X,1P2E14.6)
        GO TO 322
      END IF
      IF (ZZ.LT.0) THEN
        RT=SQRT(-ZZ)
        PRINT 360 , -P/2,  RT/2             ! Imprime las raíces conjugadas complejas
360     FORMAT( F12.6,' +',F12.6,' I')
        PRINT 390 , -P/2, RT/2
390     FORMAT( F12.6,' -',F12.6,' I')
      END IF
322   PRINT *
      PRINT *,' COEFICIENTES DEL POLINOMIO REDUCIDO'
      PRINT *,'          ORDEN        COEFICIENTES '
      DO 325 I=2,N
        PRINT 380,I-2,B(I)        ! Imprime los coeficientes del polinomio reducido
325   CONTINUE
380   FORMAT( I10,F12.6)
      PRINT *,'-------------------------------------------------------'
      PRINT *
      PRINT*
      PRINT*,' OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR
      READ *,K
      IF(K.EQ.1) GOTO 1
      PRINT*
      END
```

D) Ejemplo de salida

Véase el ejemplo 3.9.

PROBLEMAS

3.1) Determine la raíz positiva de $x^2 - 0.9x - 1.52 = 0$ en el intervalo [1, 2] mediante el método de bisección, con una tolerancia de 0.001.

3.2) Encuentre la raíz de

$$x \operatorname{sen}(x) - 0.1 = 0, \quad 0 < x < 1.0$$

mediante el método de bisección, con una tolerancia de 0.001.

3.3) Calcule la raíz de $\tan(x) = 3.5$ en el intervalo $[0, \pi]$ mediante el método de bisección, con una tolerancia de 0.005.

3.4) a) Determine un intervalo de tamaño 0.5 para cada raíz positiva de las siguientes ecuaciones, utilizando el PROGRAMA 3-2:

i) $f(x) = 0.5e^{x/3} - \operatorname{sen} x \quad = 0, \quad x > 0$

ii) $f(x) = \log_e(1 + x) - x^2 = 0$

b) Grafique las funciones definidas anteriormente en el plano xy, utilizando el PROGRAMA 3-3 y verifique los resultados de a).

3.5) a) Determine un intervalo de tamaño 0.5 para cada raíz de las ecuaciones siguientes utilizando el PROGRAMA 3-2 y una modificación al subprograma:

i) $f(x) = e^x - 5x^2 = 0$

ii) $f(x) = x^3 - 2x - 1 = 0$

iii) $f(x) = \sqrt{x} + 2 - x = 0$

b) Grafique las funciones antes definidas en el plano xy utilizando el PROGRAMA 3-3 y verifique los resultados de a).

c) Calcule la máxima raíz de cada uno de los problemas de 3.5) mediante el método de bisección, con una tolerancia de 0.0001.

3.6) Encuentre la raíz de

$$\frac{x(2.1 - 0.5x)^{1/2}}{(1 - x)(1.1 - 0.5x)^{1/2}} = 3.69, \quad 0 < x < 1$$

en el intervalo [0, 1] por medio del PROGRAMA 3-1 y cambiando el subprograma, con una tolerancia de 0.001.

3.7) Encuentre todas las raíces positivas de las ecuaciones siguientes mediante el método de bisección con una tolerancia de 0.001. (Primero determine un intervalo apropiado para cada raíz mediante el PROGRAMA 3-3 o enlistando los valores de la función para valores escogidos de x.)

a) $\tan(x) - x + 1 = 0, \quad 0 < x < 3\pi$

b) $\operatorname{sen}(x) - 0.3e^x = 0, \quad x > 0$

c) $-x^3 + x + 1 = 0$

d) $16x^5 - 20x^3 + x^2 + 5x - 0.5 = 0$

3.8) Calcule intervalos apropiados para las raíces de las siguientes ecuaciones y determine después las raíces mediante el método de bisección (utilice el PROGRAMA 3-1) con una tolerancia de 0.001:

a) $0.1x^3 - 5x^2 - x + 4 + e^{-x} = 0$

b) $\log_e(x) - 0.2x^2 + 1 = 0$

c) $x + \dfrac{1}{(x + 3)x} = 0$

3.9) Un proyectil de $M = 2$ gm se ha lanzado en forma vertical al aire y está descendiendo a su velocidad terminal. Dicha velocidad se determina mediante la ecuación $gM = D_{\text{drag}}$ donde g es la gravedad y M es la masa; esta ecuación se puede escribir después de evaluar las constantes como

$$\frac{(2)(9.81)}{1000} = 1.4 \times 10^{-5}v^{1.5} + 1.15 \times 10^{-5}v^2$$

donde v es la velocidad terminal en m/seg. El primer término del lado derecho representa la fuerza de fricción y el segundo la fuerza de la presión. Determine la velocidad terminal mediante el método de bisección, con una tolerancia de 0.001.

3.10) La configuración superficial de la aeronave NACA 0012 de longitud de arco 1 m y con espesor máximo de 0.2 m está dada por

$$y(x) = \pm[0.2969\sqrt{x} - 0.126x - 0.3516x^2 + 0.2843x^3 - 0.1015x^4]$$

donde los signos más y menos se refieren a las superficies superior e inferior, respectivamente. Determine x, donde el espesor del aparato es 0.1 m por medio del método de bisección. Haga la tolerancia igual a 0.00001. (Existen dos soluciones.)

3.11) Una masa de 1 kg de CO está contenido en un recipiente a $T = 215°K$ y $p = 70$ bars. Calcule el volumen del gas utilizando la ecuación de estado de van der Waals para un gas no ideal, dada por [Moran/Shapiro]

$$P + \frac{a}{v^2}(v - b) = RT$$

donde $R = 0.08314$ bar m^3/(kg mol °K), $a = 1.463$ bar m^6/(kg mol)^2 y $b = 0.0394$ m^3/kg. Determine el volumen específico v (en m^3/kg) y compare los resultados con el volumen calculado por la ecuación del gas ideal, $Pv = RT$.

3.12) Encuentre la raíz de $f(x) = \text{sen}(x) - x + 1$ que se sabe está en $1 < x < 3$, mediante el método de la falsa posición modificada. Detenga los cálculos después de cuatro iteraciones.

3.13) Determine las raíces de las siguientes ecuaciones mediante el método de la falsa posición modificada:

a) $f(x) = 0.5 \exp(x/3) - \text{sen}(x), \quad x > 0$

b) $f(x) = \log(1 + x) - x^2$

c) $f(x) = \exp(x) - 5x^2$

d) $f(x) = x^3 + 2x - 1 = 0$

e) $f(x) = \sqrt{x + 2}$

3.14) La función de transferencia para un sistema está dada por

$$F(s) = \frac{H(s)}{1 + G(s)H(s)}$$

donde

$$G(s) = \frac{1}{s}\exp(-0.1s), \quad H(s) = \text{K}$$

Busque las raíces de la ecuación característica $1 + G(s)H(s) = 0$ para K = 1, 2 y 3 mediante el método gráfico y evalúelas después mediante el método de la falsa posición modificada.

3.15) Encuentre la raíz de

$$\tan(x) - 0.1x = 0$$

en $\pi < x < 1.5\pi$ mediante el método de Newton con una calculadora de bolsillo (la tolerancia es de 0.0001).

3.16) Encuentre las raíces de las ecuaciones del problema 3.7 mediante el método de Newton, con una tolerancia de 0.0001.

3.17) Las frecuencias naturales de vibración de una varilla uniforme sujetada por un extremo y libre por el otro [Thomson] son soluciones de

$$\cos(\beta l)\cosh(\beta l) + 1 = 0 \tag{A}$$

donde

$\beta = \rho\omega^2/EI$
$l =$ 1 (longitud de la varilla en metros)
$\omega =$ frecuencia en seg^{-1}
$EI =$ rigidez de flexión [Byars/Snyder/Plants]
$\rho =$ densidad del material de la varilla

Busque las raíces de la ecuación (A) primero mediante el método gráfico, y determine después los tres valores más pequeños de β que satisfagan la ecuación (A) mediante el método de Newton.

3.18) Las frecuencias naturales de vibración de una varilla sujeta en ambos extremos satisfacen

$$\tan(\beta l) = \tan h(\beta l), \quad \beta > 0$$

donde se supone que l es 1, como en el problema (3.17). Utilice el método de Newton con base en una aproximación por diferencias para evaluar la derivada, y determine los valores más pequeños de $\beta > 0$ que satisfagan la ecuación anterior. No incluya a $\beta = 0$ como respuesta. *Sugerencia*: $\tanh(x) = [\exp(x) - \exp(x)]/[\exp(x) + \exp(-x])$

3.19) Repita el problema (3.12) con el método de Newton.

3.20) Encuentre todas las raíces de la ecuación del problema (3.13) mediante el método de Newton.

3.21) Dos raíces complejas de

$$y = 2 - x + 2x^2 + x^4$$

son $-0.5 + 1.5i$ y $0.5 - 0.7i$, aproximadamente. Utilice estos valores como suposiciones iniciales y encuentre los valores exactos de las dos raíces complejas mediante el método de Newton (use el PROGRAMA 3-6).

3.22) Una mezcla equimolar de monóxido de carbono y oxígeno alcanza el equilibrio a 300°K y a una presión de 5 atm. La reacción teórica es

$$CO + \frac{1}{2}O_2 \rightleftharpoons CO_2$$

La reacción química real se escribe como

$$CO + O_2 \longrightarrow x\,CO + \frac{1}{2}(1 + x)O_2 + (1 - x)\,CO_2$$

La ecuación de equilibrio químico para determinar la fracción del CO restante, x, se escribe como

$$K_p = \frac{(1 - x)(3 + x)^{1/2}}{x(x + 1)^{1/2}P^{1/2}}, \quad 0 < x < 1$$

donde $K_p = 3.06$ es la constante de equilibrio para $CO + \frac{1}{2} O^2 = CO_2$ a 3000°K y $P = 5$ es la presión (Wark, pág. 608]. Determine el valor de x por medio del método de Newton.

3.23) Considere la misma reacción química del problema anterior, pero que ocurra con la presencia de N_2 a la presión atmosférica. La reacción real es

$$CO + O_2 + 3.76\ N_2 \longrightarrow x\ CO + \frac{1}{2}(1 + x)O_2 + (1 - x)CO_2 + 3.76\ N_2$$

La ecuación de equilibrio es

$$3.06 = \frac{(1 - x)(10.52 + x)^{1/2}}{x(1 + x)^{1/2}}$$

Determine el valor de x por medio del método de Newton.

3.24) Repita el problema 3.7) con el método de la secante.

3.25) Repita el problema 3.8) con el método de la secante.

3.26) La ecuación $x^2 - 2x - 3 = 0$ se puede reformular mediante el método de sustitución sucesiva como sigue

a) $x = \dfrac{(x^2 - 3)}{2}$

b) $x = \sqrt{2x + 3}$

c) $x = \dfrac{(2x + 3)}{\sqrt{x}}$

d) $x = x - 0.2(x^2 - 2x - 3)$

Las soluciones de la ecuación son $x = 3$ y $x = -1$. Determine en forma gráfica cuáles de las fórmulas anteriores convergen cuando se utilizan con la sustitución sucesiva para encontrar la raíz $x = -1$. Verifique los resultados del enfoque gráfico utilizando el criterio dado por la ecuación 3.7.3). Repita el mismo análisis para $x = 3$.

3.27) Encuentre todas las soluciones de las ecuaciones del problema 3.4 utilizando la sustitución sucesiva en la forma

$$x = x - \alpha f(x)$$

Sugerencia: determine α usando el gradiente de la interpolación lineal ajustada a los dos extremos del intervalo encontrados en el problema 3.4.

3.28) El coeficiente de la fricción f para el flujo turbulento en un tubo está dado por

$$\frac{1}{\sqrt{f}} = 1.14 - 2.0 \log_{10}\left(\frac{e}{D} + \frac{9.35}{R_e\sqrt{f}}\right)$$

(correlación de Colebrook)

donde R_e es el número de Reynolds, e es la rugosidad de la superficie del tubo y D es el diámetro del tubo [Shames]. **a)** Escriba un programa de computadora para resolver esta ecuación en términos de f, utilizando el método de sustitución sucesiva. **b)** Evalúe f llevando a cabo el programa para los siguientes casos:

i) $D = 0.1\text{m}, \quad e = 0.0025, \quad R_e = 3 \times 10^4$

ii) $D = 0.1\text{m}, \quad e = 0.0001, \quad R_e = 5 \times 10^6$

(*Sugerencia*: primero reescriba la ecuación en la siguiente forma:

$$f = \left(1.14 - 2.0 \log_{10}\left[\frac{e}{D} + \frac{9.35}{R_e\sqrt{f}}\right]\right)^{-2}$$

Introduzca una estimación inicial para f en el lado derecho. Reintroduzca de nuevo la f calculada en el lado derecho y repita esta iteración hasta que f converja. La estimación inicial puede igualarse a cero. Los resultados de estos cálculos se pueden verificar con una tabla de Moody que se puede encontrar en cualquier libro usual sobre mecánica de fluidos.)

3.29) Por medio del método de Bairstow, encuentre todos los factores cuadráticos de:

a) $x^4 - 5x^2 + 4 = 0$

b) $2x^3 + x^2 - x - 7 = 0$

c) $-x^4 - 4x^3 - 7x^2 + x - 3 = 0$

d) $-x^3 + 9x^3 - 18x + 16 = 0$

e) $x^4 - 16x^3 + 72x^2 - 96x + 24 = 0$

f) $x^6 - 6x^5 + 14x^4 - 18x^3 + 14x^2 - 6x + 1 = 0$

3.30) Si un polinomio tiene más de un factor cuadrático idéntico, la convergencia del método de Bairstow es pobre y los resultados poco precisos. Los siguientes polinomios tienen raíz séxtuple en $x = 1$:

$$x^6 - 6x^5 + 15x^4 - 20x^3 + 15x^2 - 6x + 1 = 0$$

Intente encontrar todos los factores cuadráticos mediante el método de Bairstow.

3.31) Encuentre las raíces de las siguientes ecuaciones por medio del método de Bairstow:

a) $y = 2 - x + 2x^2 + x^4$

b) $y = 2 + 4x + 3x^2 + x^3$

c) $y = 1.1 - 1.6x - 1.7x^2 + x^3$

d) $y = 11.55 + 0.325x - 9.25x^2 + 1.1x^3 + x^4$

e) $y = x^4 - 6x^3 + 13x^2 - 12x + 4$

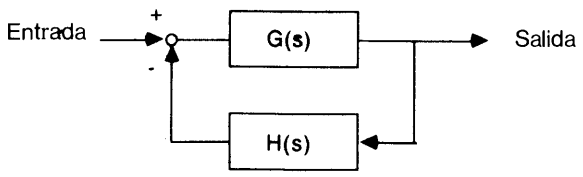
3.32) Los siguientes polinomios tienen raíces dobles:

a) $y = -8 + 12x - 2x^2 - 3x^3 + x^4$

b) $y = 4 - 12x + 13x^2 - 6x^3 + x^4$

Encuentre valores aproximados de las raíces de los polinomios anteriores mediante el método de Bairstow.

3.33) A continuación se muestra el diagrama de bloques de un sistema dinámico:



donde las funciones de transferencia están dadas por

$$G(s) = \frac{s+2}{s(s+3)}$$

$$H(s) = \frac{K}{s^2 + 2s + 2}$$

Por tanto, la función de transferencia total está dada por

$$Y(s) = \frac{G(s)}{1 + G(s)H(s)}$$

La respuesta transitoria del sistema está caracterizada por los polos de $Y(s)$, es decir, los ceros de la ecuación característica

$$1 + G(s)H(s) = 0$$

Encuentre todos los polos de $Y(s)$ para $K = 0$, 1 y 10 por el método de Bairstow.

3.34) Modifique el PROGRAMA 3-7 para que encuentre en forma automática todos los factores cuadráticos.

BIBLIOGRAFIA

Abramowitz, M. y L.A. Stegun, editores, *Handbook of Mathematical Functions*, National Bureau of Standards, 1970.

Byars, E. F., R. D. Snyder y H. L. Plants, *Engineering Mechanics of Deformable Bodies*, Harper & Row, 1983.

Cheney, W. y D. Kincaid, *Numerical Mathematics and Computing*, Brooks/Cole, 1985.

Conte, S.D. y C. de Boor, *Elementary Numerical Analysis,* 3a. ed. McGraw-Hill, 1980.

Gerald, C. F. y P. O. Wheatley, *Applied Numerical Analysis*, 3a. ed., Addison-Wesley, 1984.

Isaacson, E. y H. B. Keller, *Analysis of Numerical Methods*, Wiley, 1966.

James, M. L., G. M. Smith y J. C. Wolford, *Applied Numerical Methods for Digital Computation*, 3a. ed., Harper & Row, 1985.

Lamarsh, J. R., *Introduction to Nuclear Reactor Theory*, Addison-Wesley, 1966.

Moran, M. y H. N. Shapiro, *Fundamentals of Engineering Thermodynamics*, Wiley, 1988.

Press, W. H., B. P. Flannery, S. A. Teukolsky, W. T. Vetterling, *Numerical Recipes*, Cambridge University Press, 1986.

Shames, I. H., *Mechanics of Fluids*, McGraw-Hill, 1982.

Shoup, T. E., *Applied Numerical Methods for the Micro-Computer.* Prentice-Hall, 1984.

Thomson, W. T., *Theory of Vibration with Applications*, Prentice-Hall, 1981.

Wark, K., Jr., *Thermodynamics*, McGraw-Hill, 1988.

4

Integración numérica

4.1 INTRODUCCION

Los métodos de integración numérica se pueden utilizar para integrar funciones dadas, ya sea mediante una tabla o en forma analítica. Incluso en el caso en que sea posible la integración analítica, la integración numérica puede ahorrar tiempo y esfuerzo si sólo se desea conocer el valor numérico de la integral.

Este capítulo analiza los métodos numéricos que se utilizan para evaluar integrales de una variable:

$$I = \int_a^b f(x)\,dx$$

así como integrales dobles:

$$I = \int_a^b \int_{u(x)}^{v(x)} f(x, y)\,dy\,dx$$

donde las funciones $f(x)$ y $f(x, y)$ pueden estar dadas en forma analítica o mediante una tabla.

Los métodos de integración numérica se obtienen al integrar los polinomios de interpolación. Por consiguiente, las distintas fórmulas de interpolación darán por resultado distintos métodos de integración numérica. Los métodos que se estudian en las secciones 4.2 hasta la 4.5 se refieren a las fórmulas de Newton-Cotes, que se basan en las fórmulas de interpolación con puntos de separación uniforme y se deducen al integrar las fórmulas de interpolación de Newton hacia adelante y hacia atrás, así como la fórmula de interpolación de Lagrange. A su vez, las fórmulas de Newton-Cotes se subdividen en las de tipo cerrado y las de tipo abierto. Las reglas del trapecio y las dos reglas de Simpson pertenecen al tipo cerrado de las fórmulas de

Newton-Cotes. Las cuadraturas de Gauss, analizadas en la sección 4.6, se basan en la interpolación polinomial, usando las raíces de un polinomio ortogonal, como los polinomios de Legendre. Los métodos de integración examinados en la sección 4.7 se aplican a integrales con límites infinitos y a las integrales de funciones singulares. La última sección describe la integración numérica para integrales dobles.

En la tabla 4.1 aparece un resumen de las ventajas y desventajas de los métodos de integración numérica que se estudian en este capítulo.

Tabla 4.1 Resumen de los métodos de integración numérica

Método	Ventajas	Desventajas
Regla del trapecio	Sencillez. Optima para integrales impropias.	Necesita un gran número de subintervalos para una buena precisión.
Regla de 1/3 de Simpson	Sencillez. Más precisión que la regla del trapecio.	Sólo con un número par de intervalos.
Regla de 3/8 de Simpson	El mismo orden de precisión que la regla de 1/3.	Sólo con intervalos cuyo número sea múltiplo de tres.
Fórmulas de Newton-Cotes	Utiliza puntos con igual separación. Se dispone de fórmulas abiertas y cerradas.	Las fórmulas de orden superior no necesariamente son más precisas.
Cuadraturas de Gauss	Más precisión que las fórmulas de Newton-Cotes. No se utilizan los valores de la función en los extremos.	Los puntos no están separados uniformemente.
Transformación exponencial doble	Buena precisión para las integrales impropias.	Requiere de un cuidado especial para evitar el desbordamiento o la división por números muy pequeños.

4.2 REGLA DEL TRAPECIO

Esta regla es un método de integración numérica que se obtiene al integrar la fórmula de interpolación lineal. Se escribe en la forma siguiente:

$$I = \int_a^b f(x)\,dx = \frac{b-a}{2}\,[f(a) + f(b)] + E \tag{4.2.1}$$

donde el primer término del lado derecho es la regla del trapecio (fórmula de integración) y E representa su error. En la figura 4.1 se muestra gráficamente la integración numérica por medio de la ecuación (4.2.1). El área sombreada por debajo de la recta de interpolación (la cual puede denotarse como $g(x)$) es igual a la integral calculada mediante la regla del trapecio, mientras que el área por debajo de la curva $f(x)$ es el valor exacto. Por lo tanto, el error de la ecuación (4.2.1) es igual al área entre $g(x)$ y $f(x)$.

La ecuación (4.2.1) se puede extender a varios intervalos y se puede aplicar N veces al caso de N intervalos con una separación uniforme h (como se muestra en la figura 4.2) para así obtener la regla extendida del trapecio:

$$I = \int_a^b f(x) = \frac{h}{2}\,[f(a) + 2\sum_{j=1}^{N-1} f(a + jh) + f(b)] + E$$

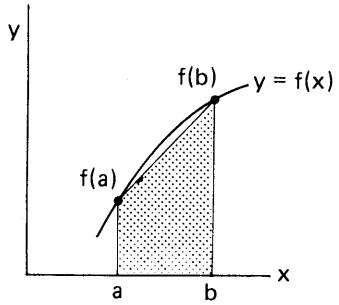


Figura 4.1 Regla del trapecio

donde $h = (b - a)/N$. La ecuación anterior se puede escribir en la siguiente forma equivalente:

$$I = \frac{h}{2}(f_0 + 2f_1 + 2f_2 + \cdots + 2f_{N-1} + f_N) + E \tag{4.2.2}$$

donde $f_0 = f(a)$, $f_1 = f(a + h)$, y $f_i = f(a + ih)$.

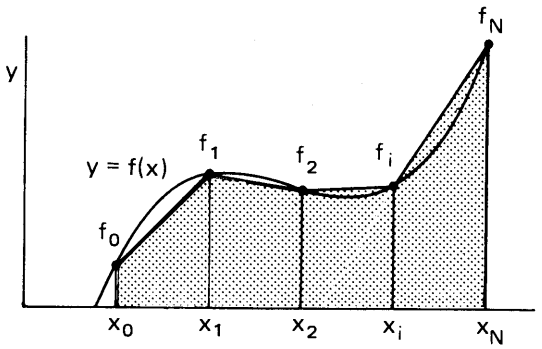


Figura 4.2 Regla extendida del trapecio

Ejemplo 4.1

El cuerpo de revolución que se muestra en la figura E4.1 se obtiene al girar la curva dada por $y = 1 + (x/2)^2$, $0 \leqslant x \leqslant 2$, en torno al eje x. Calcule el vo-

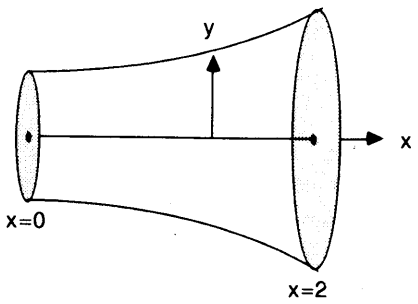


Figura E4.1 Un cuerpo de revolución

lumen utilizando la regla extendida del trapecio con $N = 2, 4, 8, 16, 32, 64$ y 128. El valor exacto es $I = 11.7286$. Evalúe el error para cada N.

(Solución)

El volumen está dado por

$$I = \int_0^2 f(x)\,dx$$

donde

$$f(x) = \pi\left(1 + \left(\frac{x}{2}\right)^2\right)^2$$

A continuación aparecen los cálculos para $N = 2$ y 4:

$$N = 2: \quad h = 2/2 = 1$$

$$I = \frac{1}{2}[f(0) + 2f(1) + f(2)] = 0.5\pi[1 + 2(1.5625) + 4]$$

$$= 12.7627$$

$$N = 4: \quad h = 2/4 = 0.5$$

$$I = (0.5/2)[f(0) + 2f(0.5) + 2f(1) + 2f(1.5) + f(2)] = 11.9895$$

Las integraciones con los demás valores de N se evalúan mediante el PROGRAMA 4-1. Los resultados se resumen en la tabla E4.2.

Tabla E4.2

N	h	I_h	E_h
2	1.	12.7627	−1.0341
4	0.5	11.9895	−0.2609
8	0.25	11.7940	−0.0654
16	0.125	11.7449	−0.0163
32	0.0625	11.7326	−0.0040
64	0.03125	11.7296	−0.0010
128	0.015625	11.7288	−0.0002

Valor exacto 11.7286

Se puede observar que el error decrece en forma proporcional a h^2.

El error de la regla del trapecio se define como

$$E = \int_a^b f(x)\,dx - \frac{b-a}{2}[f(a) + f(b)] \tag{4.2.3}$$

donde el primer término es la integral exacta y el segundo es la regla del trapecio: Para analizar la ecuación (4.2.3), utilizaremos los desarrollos en serie de Taylor de $f(x)$, $f(a)$ y $f(b)$ en torno a $\bar{x} = (a + b)/2$, con la hipótesis de que f es analítica en $a \leqslant x \leqslant b$.

El desarrollo de Taylor para $f(x)$ se escribe como

$$f(x) = f(\bar{x}) + zf'(\bar{x}) + \frac{z^2}{2} f''(\bar{x}) + \cdots \tag{4.2.4}$$

donde

$z = x - \bar{x}$

Por lo anterior, el primer término de la ecuación (4.2.3) se puede escribir como

$$\int_a^b f(x)\,dx = \int_{-h/2}^{h/2} \left[f(\bar{x}) + zf'(\bar{x}) + \frac{z^2}{2} f''(\bar{x}) + \cdots \right] dz \tag{4.2.5}$$

donde $z = -h/2$ para $x = a$ y $z = h/2$ para $x = b$. Al integrar obtenemos lo siguiente:

$$\int_a^b f(x)\,dx = hf(\bar{x}) + \frac{1}{24} h^3 f'' + \cdots \tag{4.2.6}$$

Por otro lado, el segundo término de la ecuación (4.2.3) se escribe

$$\begin{aligned} \frac{b-a}{2}[f(a) + f(b)] &= \frac{h}{2}\left[f(\bar{x}) - \frac{h}{2} f'(\bar{x}) + \frac{1}{2}\left(\frac{h}{2}\right)^2 f''(\bar{x}) - \cdots \right. \\ &\quad \left. + f(\bar{x}) + \frac{h}{2} f'(\bar{x}) + \frac{1}{2}\left(\frac{h}{2}\right)^2 f''(\bar{x}) + \cdots \right] \\ &= hf(\bar{x}) + \frac{1}{8} h^3 f''(\bar{x}) + \cdots \end{aligned} \tag{4.2.7}$$

Por lo que al sustituir las ecuaciones (4.2.6) y (4.2.7) en la ecuación (4.2.3) obtenemos

$$\begin{aligned} E &= \int_a^b f(x)\,dx - \frac{b-a}{2}[f(a) + f(b)] \\ &\simeq -\frac{1}{12} h^3 f''(\bar{x}) \end{aligned} \tag{4.2.8}$$

en donde se han truncado los términos de orden superior. La ecuación indica que el error de la regla del trapecio es proporcional a f'' y decrece en forma proporcional a h^3 cuando $h = b - a$ se reduzca. Como se señaló anteriormente, el análisis se basa en la hipótesis de que $f(x)$ es analítica en el intervalo. Si éste no es el caso, el error no es proporcional a h^3.

El error de la regla extendida del trapecio es la suma de los errores en todos los intervalos. Supongamos que se aplica la regla del trapecio a un intervalo $[a, b]$, el cual se divide en N intervalos mediante los $N + 1$ puntos $x_0, x_1, x_2, \ldots, x_N$, donde $x_0 = a$ y $x_N = b$. Puesto que el error para cada intervalo está dado por (4.2.5), el error de la regla extendida del trapecio está dado por

$$E \simeq -\frac{1}{12} \frac{(b-a)^3}{N^3} \sum_{i=1}^{N} f''(\bar{x}_i) \tag{4.2.9}$$

donde $h = (b - a)/N$ y x_i es el punto medio entre x_i y x_{i+1}. Si definimos $\bar{f}''$ como el promedio de f'', es decir,

$$\bar{f}'' = \sum_{i=1}^{N} f''(\bar{x}_i)/N$$

La ecuación (4.2.9) se puede reescribir como

$$E \simeq -\frac{1}{12}(b - a)h^2\bar{f}'' \tag{4.2.10}$$

La ecuación (4.2.10) muestra que el error de la regla extendida del trapecio es proporcional a h^2 para un intervalo fijo $[a, b]$. (Esto coincide con la observación hecha en el ejemplo 4.1).

En el análisis del error para la regla del trapecio, se tiene como hipótesis que $f(x)$ es analítica en $[a, b]$. De otra forma, el error no sería proporcional a h^2. Por ejemplo, si se integra $f(x) = \sqrt{x}$ en $[0, 1]$ por medio de la regla extendida del trapecio, el error decrece muy lentamente al disminuir el tamaño de los intervalos. Esto se debe a que $f = \sqrt{x}$ no es analítica en $x = 0$. En el caso de las funciones con una singularidad, se recomienda el uso de la transformación exponencial doble (sección 4.7).

Una aplicación importante del análisis del error para la regla del trapecio es la integración de Romberg. Supongamos que I_h es el resultado de la regla extendida del trapecio con intervalos de longitud $h = (b - a)/N$ y que I_{2h} es el resultado de otro cálculo con $h' = 2h$. Puesto que el error de la regla extendida del trapecio es proporcional a h^2, los errores con los intervalos h y $2h$ se pueden escribir respectivamente como

$$E_h \simeq Ch^2 \quad \text{y} \quad E_{2h} \simeq C(2h)^2 = 4Ch^2 \tag{4.2.11}$$

donde C es una constante. Por otro lado, la integral exacta se puede escribir como $I = I_h + E_h = I_{2h} + E_{2h}$, de lo cual obtenemos

$$E_h - E_{2h} = I_{2h} - I_h \tag{4.2.12}$$

Al sustituir la ecuación (4.2.1) en la ecuación (4.2.12) y despejar C obtenemos

$$C = \frac{1}{3}h^{-2}(I_h - I_{2h})$$

Así, la primera ecuación de (4.2.8) da el siguiente valor aproximado de E_h

$$E_h \simeq \frac{1}{3}(I_h - I_{2h}) \tag{4.2.13}$$

Si conocemos los valores de I_h y I_{2h} a partir de los cálculos hechos, obtenemos la siguiente integral, la cual es más precisa:

$$I = I_h + E_h \simeq I_h + \frac{1}{3}(I_h - I_{2h}) \tag{4.2.14}$$

El valor anterior de I no es exacto, puesto que tampoco lo es la ecuación (4.2.11), pero el error de la ecuación (4.2.14) es proporcional a h^4, término que tiene un orden

dos veces mayor al de I_h. Por lo tanto, la ecuación (4.2.14) da un resultado más exacto que el de I_h o el de I_{2h}. Esta técnica se llama integración de Romberg. Para más aplicaciones de la integración de Romberg, véase [James/Smith/Wolford; Ferziger y Gerald/Wheatley].

Ejemplo 4.2

En el ejemplo 4.1, la regla extendida del trapecio da como resultado $I_{0.5} = 11.9895$ y $I_{0.25} = 11.7940$. Determine un valor más exacto utilizando la integración de Romberg.

(Solución)

Si definimos $h = 0.25$ en las ecuaciones (4.2.11) a la (4.2.14), el valor de $E_{0.25}$ dado por la ecuación (4.2.13) es

$$E_{0.25} \simeq \frac{1}{3}(11.7940 - 11.9895) = -0.0652$$

Por lo tanto, se obtiene el siguiente valor más exacto de I, dado por la ecuación (4.2.14)

$$I = I_{0.25} + E_{0.25} \simeq 11.7940 - 0.0652 = 11.7288$$

Este resultado coincide con el resultado para $N = 128$ (con $h = 0.0156$) en el ejemplo 4.1.

Otra aplicación importante de la regla extendida del trapecio es la integración de una función desde $-\infty$ a ∞. El método óptimo para este tipo de problema es la regla extendida del trapecio. Al transformar un intervalo finito en toda la recta infinita, se puede integrar con exactitud cualquier función mediante la regla extendida del trapecio. Este enfoque se explica con mayor detalle en la sección 4.7.

Resumen de Esta Sección

a) La regla del trapecio se basa en la integración de las interpolaciones lineales.
b) La regla extendida del trapecio se obtiene al repetir la regla del trapecio.
c) Para un dominio de integración dado, el error de la regla extendida del trapecio es proporcional a h^2.
d) La integración de Romberg se basa en el hecho señalado en el inciso c). Por medio de los resultados de la regla extendida del trapecio para dos conjuntos de datos con distintas separaciones, se evalúa una integral con mayor exactitud.
e) Véase la sección 4.7 para un método de integración más avanzado basado en la regla extendida del trapecio.

4.3 REGLA DE 1/3 DE SIMPSON

La regla de 1/3 de Simpson se basa en la interpolación polinomial cuadrática (de segundo grado, véase la figura 4.3). El polinomio de Newton hacia adelante ajustado a

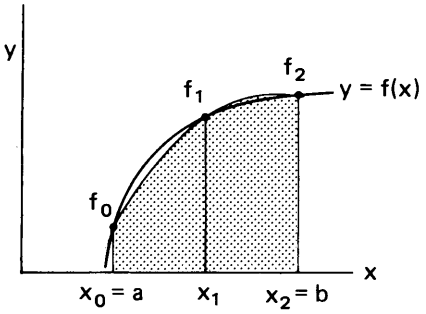


Figura 4.3 Regla de Simpson

tres puntos, x_0, x_1, x_2, está dado por la ecuación (2.4.7). Al integrar esta ecuación —haciendo el cambio de variable adecuado desde $x_0 = a$ hasta $x = b$ se obtiene la regla de 1/3 de Simpson:

$$I = \int_a^b f(x)\,dx = \frac{h}{3}[f(a) + 4f(\bar{x}) + f(b)] + E \tag{4.3.1}$$

donde $h = (b - a)/2$ y $x = (a + b)/2$. La ecuación (4.3.1) se puede escribir en la forma equivalente

$$I = \frac{h}{3}[f_0 + 4f_1 + f_2] + E \tag{4.3.2}$$

donde $f_i = f(x_i) = f(a + ih)$. Se mostrará posteriormente que el error es

$$E \simeq -\frac{h^5}{90} f^{iv}(\bar{x}) \tag{4.3.3}$$

El error se anula si $f(x)$ es un polinomio de orden menor o igual que 3. La regla de 1/3 de Simpson es fácil de aplicar con una calculadora. Su precisión es suficiente para muchas aplicaciones, como se ilustra en el ejemplo 4.3.

La regla extendida de 1/3 de Simpson es una aplicación repetida de la ecuación (4.3.2) para un dominio dividido en un número par de intervalos. Si denotamos el número total de intervalos por N (par), la regla extendida de 1/3 de Simpson se escribe como

$$I = \frac{h}{3}\Big[f(a) + 4\sum_{\substack{i=1\\ \text{impar } i}}^{N-1} f(a + ih) + 2\sum_{\substack{i=2\\ \text{par } i}}^{N-2} f(a + ih) + f(b)\Big] + E \tag{4.3.4}$$

donde $h = (b - a)/N$; la primera suma es únicamente sobre las i impares y la segunda es sólo sobre las i pares. La ecuación (4.3.4) se puede escribir en la forma equivalente

$$\begin{aligned} I &= \int_a^b f(x)\,dx \\ &= \frac{h}{3}[f_0 + 4f_1 + 2f_2 + 4f_3 + 2f_4 + \cdots + 2f_{N-2} + 4f_{N-1} + f_N] + E \end{aligned} \tag{4.3.5}$$

El término del error está dado por

$$E \simeq -\frac{N}{2}\frac{h^5}{90} f^{iv}(\bar{x}) = -(b-a)\frac{h^4}{180} f^{iv}(\bar{x}) \tag{4.3.6}$$

donde

$\bar{x} = (a+b)/2$

Para un dominio fijo $[a, b]$, el error es proporcional a h^4.

Ejemplo 4.3

Repita el problema del ejemplo 4.1 utilizando la regla extendida de 1/3 de Simpson con $N = 2, 4, 8, 16, 32$.

(Solución)

El tamaño del intervalo es $h = 2/N$. Los cálculos para $N = 2$ y 4 son como sigue:

$$N = 2: \quad I = \frac{1}{3}[f(0) + 4f(1) + f(2)]$$

$$= \frac{1}{3}\pi[1 + (4)(1.25^2) + 2^2] = 11.7809$$

$$N = 4: \quad I = \frac{0.5}{3}[f(0) + 4f(0.5) + 2f(1) + 4f(1.5) + f(2)]$$

$$= \frac{0.5}{3}\pi[1 + 4(1.0625) + 2(1.25)^2 + 4(1.5625)^2 + 2^2]$$

$$= 11.7318$$

Los cálculos para N mayores se pueden realizar de manera análoga. Los resultados y evaluaciones del error se muestran a continuación.

N	h	I_h	E_h
2	1.	11.7809	−0.0523
4	0.5	11.7318	−0.0032
8	0.25	11.7288	−0.0002
16	0.125	11.7286	0
32	0.0625	11.7286	0
64	0.03125	11.7286	0

Al comparar los resultados anteriores con los del ejemplo 4.1 se puede ver que la regla extendida de Simpson es mucho más precisa que la regla extendida del trapecio, utilizando el mismo número de intervalos. Por ejemplo, la exactitud de la regla extendida del trapecio con 32 intervalos es equivalente a la de la regla

extendida de Simpson con tan sólo 4 intervalos. El error de la regla extendida de Simpson es proporcional a h^4, por lo que es dos órdenes más grande que el de la regla extendida del trapecio. Debido al alto orden del error, la regla extendida de Simpson tiende a la solución exacta en forma más rápida que la regla extendida del trapecio cuando h se reduce.

Podríamos intentar obtener el término del error al integrar el error de la interpolación cuadrática dada por la ecuación (2.3.7), pero el resultado de la integración se anula y no representa el verdadero error. La razón para esta errónea consecuencia proviene de la aproximación $\xi = \bar{x}$. Por lo tanto, necesitamos un enfoque más preciso para obtener el término del error.

Utilizaremos el desarrollo de Taylor para la regla de 1/3 de Simpson. Los desarrollos de Taylor para f_0 y f_2 en torno de x_1, o en forma equivalente, $\bar{x} = (a + b)/2$, se escriben como

$$f_0 = f_1 - hf'_1 + \frac{1}{2}h^2f''_1 - \frac{1}{6}h^3f'''_1 + \frac{1}{24}h^4f''''_1 - \cdots$$

$$f_2 = f_1 + hf'_1 + \frac{1}{2}h^2f''_1 + \frac{1}{6}h^3f'''_1 + \frac{1}{24}h^4f''''_1 + \cdots$$

Al sustituir estos desarrollos en la ecuación (4.3.2), obtenemos

$$I = 2hf_1 + \frac{1}{3}h^3f''_1 + \frac{1}{36}h^5f''''_1 + \cdots + E \tag{4.3.7}$$

Por otro lado, el desarrollo de Taylor de $f(x)$ alrededor de x_1 es

$$f(x) = f_1 + zf'_1 + \frac{1}{2}z^2f''_1 + \frac{1}{6}z^3f'''_1 + \frac{1}{24}z^4f''''_1 + \cdots \tag{4.3.8}$$

donde

$x = x_1 + z$ o equivalentemente, $z = x - x_1$.

La integración analítica de este desarrollo en $[a, b]$ da como resultado

$$\int_a^b f(x)\,dx = 2hf_1 + \frac{1}{3}h^3f''_1 + \frac{1}{60}h^5f''''_1 + \cdots \tag{4.3.9}$$

que se considera como la integral exacta de la forma del desarrollo de Taylor. Restamos (4.3.7) de (4.3.9) y truncamos después del término principal, con lo que el error de la ecuación (4.3.7) está dado aproximadamente por

$$E \simeq -\frac{1}{90}h^5f''''_1 \tag{4.3.10}$$

donde $f_1'''' = f''''(x_1)$. Puesto que $x_1 = \bar{x} = (a + b)/2$, este resultado ya aparecía en la ecuación (4.3.3).

Una desventaja de la regla extendida de Simpson es que el número total de intervalos debe ser par. Por otro lado, la regla de 3/8 de Simpson, descrita en la

siguiente sección, se aplica únicamente a un número de intervalos que sea múltipo de tres. Por lo tanto, al combinar las reglas de 1/3 y 3/8, se puede considerar el caso tanto par como impar de intervalos.

Resumen de Esta Sección

a) La regla de 1/3 de Simpson se obtiene al integrar una fórmula de interpolación cuadrática.
b) Al aplicar repetidas veces la regla de 1/3 a un número par de intervalos, se obtiene la regla extendida de 1/3 de Simpson. Su error es proporcional a h^4.

4.4 REGLA DE 3/8 DE SIMPSON

La regla de 3/8 de Simpson se obtiene al integrar una fórmula de interpolación polinomial de tercer grado. Para un dominio $[a, b]$ dividido en tres intervalos, se escribe como

$$I = \int_a^b f(x)\,dx = \frac{3}{8}h[f_0 + 3f_1 + 3f_2 + f_3] + E \tag{4.4.1}$$

donde

$h = (b - a)/3$, $f_i = f(a + ih)$

y E representa el error. El término del error se escribe como

$$E \simeq -\frac{3}{80}h^5 f''''(\bar{x}) \tag{4.4.2}$$

donde

$\bar{x} = (a + b)/2$

La expresión anterior para el error se puede deducir utilizando el desarrollo de Taylor de manera análoga a la descrita en el caso de la regla de 1/3 de Simpson.

Como se explicó antes, la regla extendida de 1/3 se aplica a un número par de intervalos, mientras que la regla extendida de 3/8 se aplica a un número de intervalos que sea múltiplo de tres. Cuando el número de intervalos es impar pero sin ser múltiplo de tres, se puede utilizar la regla de 3/8 para los primeros tres o los últimos tres intervalos, y luego usar la regla de 1/3 para los intervalos restantes, que son un número par. Puesto que el orden del error de la regla de 3/8 es el mismo que el de la regla de 1/3, no se gana mayor exactitud que con la regla de 1/3 cuando uno puede elegir con libertad entre ambas reglas.

Resumen de Esta Sección

a) La regla de 3/8 de Simpson se obtiene al integrar un polinomio cúbico de interpolación. El orden de su error es el mismo que el de la regla de 1/3.
b) Esta regla se puede extender a un número de intervalos que sea múltiplo de tres.

c) La regla de 3/8 de Simpson es importante en combinación con la regla extendida de 1/3.

4.5 FORMULAS DE NEWTON-COTES

Los métodos de integración numérica que se obtienen al integrar las fórmulas de interpolación de Newton reciben el nombre de fórmulas de integración de Newton-Cotes. La regla del trapecio y las dos reglas de Simpson son casos de las fórmulas de Newton-Cotes, las cuales se dividen en fórmulas cerradas y abiertas.

Escribimos las fórmulas cerradas de Newton-Cotes en la forma:

$$\int_a^b f(x)\,dx = \alpha h[w_0 f_0 + w_1 f_1 + w_2 f_2 + \cdots + w_N f_N] + E \tag{4.5.1}$$

donde α y las w son las constantes que aparecen en la tabla 4.2 y

$$f_n = f(x_n), \quad x_n = a + nh, \quad \text{y} \quad h = (b - a)/N$$

La ecuación (4.5.1) recibe el nombre de *fórmula cerrada*, debido a que el dominio de integración está cerrado por el primer y último datos.

Por otro lado, la integración de la ecuación (4.5.1) se puede extender más allá de los puntos extremos de los datos dados. Las fórmulas abiertas de Newton-Cotes

Tabla 4.2 Constantes para las fórmulas cerradas de Newton-Cotes

N	α	w_i, $i = 0, 1, 2, \ldots, N$	E
1	1/2	1 1	$-\frac{1}{12}h^3 f''$
2	1/3	1 4 1	$-\frac{1}{90}h^5 f^{\text{iv}}$
3	3/8	1 3 3 1	$-\frac{3}{80}h^5 f^{\text{iv}}$
4	2/45	7 32 12 32 7	$-\frac{8}{945}h^7 f^{\text{vi}}$
5	5/288	19 75 50 50 75 19	$-\frac{275}{12096}h^7 f^{\text{vi}}$
6	1/140	41 216 27 272 27 216 41	$-\frac{9}{1400}h^9 f^{\text{viii}}$
7	7/17280	751 3577 1323 2989 2989 1323 3577 751	$-\frac{8183}{518400}h^9 f^{\text{viii}}$
8	14/14175	989 5888 −928 10946 −4540 10946 −928 5888 989	$-\frac{2368}{467775}h^{11} f^{\text{x}}$
9	9/89600	2857 15741 1080 19344 5788 5788 19344 1080 15741 2857	$-\frac{173}{14620}h^{11} f^{\text{x}}$
10	5/299376	16067 106300 −48525 272400 −260550 427368 −260550 272400 −48525 106300 16067	$-\frac{1346350}{326918592}h^{13} f^{\text{xii}}$

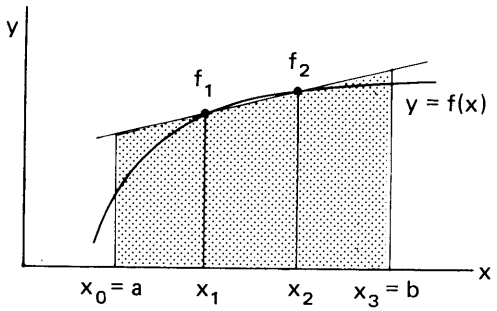


Figura 4.4 Fórmula abierta de Newton-Cotes ($N = 1$)

se obtienen al extender la integración hasta un intervalo a la izquierda del primer dato y un intervalo a la derecha del último dato (véase la figura 4.4). Dichas fórmulas se escriben como

$$\int_a^b f(x) = \alpha h[w_0 f_0 + w_1 f_1 + \cdots + w_{N+2} f_{N+2}] + E \tag{4.5.2}$$

donde $h = (b - a)/(N + 2)$. Las constantes α y w_i se listan en la tabla 4.3, en donde W_0 y W_{N+2} se igualan a cero debido a que corresponden a los extremos del dominio. Puesto que W_0 y W_{N+2} se anulan, f_0 y f_{N+2} son datos ficticios, que en realidad no son necesarios.

Tabla 4.3 Constantes para las fórmulas abiertas de Newton-Cotes

N	α	w_i, $i = 0, 1, \ldots, N + 2$	E
1	3/2	0 1 1 0	$\frac{1}{4}h^3 f''$
2	4/3	0 2 −1 2 0	$\frac{28}{90}h^5 f^{\text{iv}}$
3	5/24	0 11 1 1 11 0	$\frac{95}{144}h^5 f^{\text{iv}}$
4	6/20	0 11 −14 26 −14 11 0	$\frac{41}{140}h^7 f^{\text{vi}}$
5	7/1440	0 611 −453 562 562 −453 611 0	$\frac{5257}{8640}h^7 f^{\text{vi}}$
6	8/945	0 460 −954 2196 −2459 2196 −954 460 0	$\frac{3956}{14175}h^9 f^{\text{viii}}$

Si comparamos una fórmula abierta con una cerrada utilizando el mismo número N de datos, el error de la fórmula abierta es significativamente mayor que el de la fórmula cerrada. Por otro lado, se pueden utilizar las fórmulas abiertas cuando no se dispone de los valores de la función en los límites de integración.

Tanto en la tabla 4.2 como en la tabla 4.3 se observa que los valores de w para N grande también son grandes y cambian de signo. La resta de números grandes puede provocar errores de redondeo. Por esta razón, no son recomendables las fórmulas de Newton-Cotes de orden superior. Esta tendencia se ilustra en el ejemplo 4.4.

Ejemplo 4.4

La longitud de arco de una curva en coordenadas polares (véase la figura E4.4) está dada por

$$L = \int_a^b \sqrt{r^2 + \left(\frac{dr}{d\theta}\right)^2}\, d\theta$$

Calcular la longitud de arco de la curva dada por $r = 2(1 + \cos\theta)$, $0 < \theta < \pi$, utilizando cada una de las fórmulas cerradas de integración de Newton-Cotes que aparecen en la tabla 4.2.

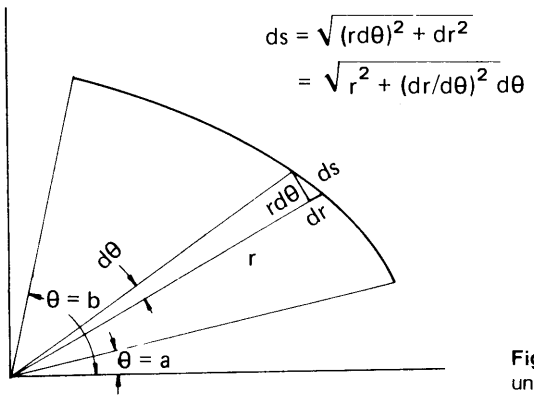


Figura E4.4 Aproximación lineal de una curva

(Solución)

Hacemos $a = 0$ y $b = \pi$ y utilizamos el PROGRAMA 4-2. A continuación se muestran los resultados computacionales:

Orden N	Integral L
2	8.01823
3	8.00803
4	7.99993
5	7.99996
6	8.00000
7	8.00000
8	8.00000
9	8.00197
10	7.99201
Valor exacto	8.00000

Estos resultados ilustran el efecto de los errores por redondeo. Es decir, cuando N crece, este resultado tiende el valor exacto, 8.00000. Sin embargo, después de $N = 8$, el error vuelve a crecer de manera gradual. El crecimiento de los errores se atribuye a los errores de redondeo en las sumas y restas de los números demasiado grandes de las fórmulas.

Resumen de Esta Sección

a) Las fórmulas cerradas y abiertas de Newton-Cotes se obtienen al integrar polinomios ajustados a datos con separación uniforme.

b) La regla del trapecio y las dos reglas de Simpson pertenecen a las fórmulas cerradas. Las fórmulas de Newton-Cotes de orden superior no son precisas debido a los errores de redondear provocados por los coeficientes grandes de signo variable.

4.6 CUADRATURAS DE GAUSS

4.6.1 Cuadraturas de Gauss-Legendre

Las cuadraturas de Gauss-Legendre (o simplemente de Gauss) son métodos de integración numérica que utilizan puntos de Legendre (raíces de polinomios de Legendre). Las cuadraturas de Gauss no se pueden utilizar para integrar una función dada en forma de tabla con intervalos de separación uniforme debido a que los puntos de Legendre no están separados de esa manera; sin embargo, son más adecuadas para integrar funciones analíticas. La ventaja de las cuadraturas de Gauss es que su precisión es mayor que la de las fórmulas de Newton-Cotes.

Antes de analizar la expresión general de las cuadraturas de Gauss, revisemos los términos del error en las fórmulas de Newton-Cotes. La ecuación (4.2.8) indica que el error de la regla del trapecio es proporcional a f''. Si se usa la regla del trapecio para integrar cada una de las funciones $f = 1, x, x^2, x^3, \ldots$, entonces los resultados serán exactos para $f = 1$ y $f = x$, pero existirán errores para x^2 y las potencias superiores de x. El error de la regla de Simpson dado por la ecuación (4.3.6) es proporcional a f^{iv}, por lo que es exacta si integramos $f = 1, x, x^2$ y x^3. En términos más generales, la fórmula cerrada de Newton-Cotes de orden impar N es exacta si el polinomio tiene orden menor o igual que N, pero cuando N es par, la fórmula resulta exacta cuando el integrando es un polinomio de orden menor o igual que $N + 1$.

Sin embargo, el siguiente ejemplo muestra que una integración numérica con dos puntos se puede hacer exacta para el caso de los polinomios de orden tres, si se optimizan los valores x de los datos.

Ejemplo 4.5

La fórmula de integración con dos puntos se puede hacer exacta cuando se integra un polinomio de orden tres. Determine los puntos.

(Solución)

Consideremos

$$I = \int_{-1}^{1} f(x)\,dx \tag{A}$$

y escribamos una fórmula de integración con dos puntos, como sigue

$$I = w_1 f(x_1) + w_2 f(x_2) + E \tag{B}$$

donde w_k, $k = 1, 2$, son pesos, x_k son puntos indeterminados y E es el término de error.

Ya que w_k y x_k son ambos indeterminados, requerimos que $E = 0$ (de modo que I es exacto) para $f(x) = 1, x, x^2$ y x^3. Introduciendo cada uno de los $f(x) = 1, x, x^2$, y x^3 en la ecuación (B) obtenemos cuatro ecuaciones:

$$2 = w_1 + w_2$$

$$0 = w_1x_1 + w_2x_2$$

$$\frac{2}{3} = w_1x_1^2 + w_2x_2^2$$

$$0 = w_1x_1^3 + w_2x_2^3$$

donde en el lado izquierdo aparecen los valores exactos.

Los límites de integración son -1 y 1 y simétricos con respecto a $x = 0$, por lo que hacemos $x_2 = -x_1$ y requerimos que los puntos estén situados en forma simétrica. De la primera y segunda ecuación obtenemos

$$w_1 = w_2 = 1$$

Con estos valores, la cuarta ecuación se satisface automáticamente. La tercera ecuación es

$$\frac{1}{3} = x_1^2$$

de la cual se obtiene

$$x_1 = \frac{1}{\sqrt{3}} = 0.577350269$$

y

$$x_2 = -x_1 = -0.577350269$$

Con estos pesos y puntos, la ecuación (B) es exacta para un polinomio de orden menor o igual a tres. Aunque consideramos el intervalo $[-1, 1]$ por simplicidad, puede cambiarse por cualquier intervalo arbitrario mediante una transformación de coordenadas.

La fórmula de integración que se deriva del Ejemplo 4.5 es el miembro más simple de las cuadraturas de Gauss.

No es sencillo derivar la cuadratura de Gauss con más de dos puntos por medio de la extensión del enfoque de este ejemplo. Por lo tanto, en lo que resta de esta sección, se da una fórmula general de las cuadraturas de Gauss y se mostrará entonces que la cuadratura de Gauss de orden N es exacta cuando se integra un polinomio de orden menor o igual a $2N - 1$.

Las cuadraturas de Gauss difieren en forma significativa de las fórmulas de Newton-Cotes ya que los N puntos de la retícula (llamados puntos de Gauss) se obtienen mediante las raíces del polinomio de Legendre $P_N(x) = 0$, donde $P_N(x)$ es el polinomio de Legendre de orden N. Sin duda alguna, x_1 y x_2 determinados en el ejemplo 4.5 son raíces de $P_2(x) = 0$. En el apéndice B aparecen más detalles acerca de los polinomios de Legendre.

La cuadratura de Gauss que se extiende sobre el intervalo [—1, 1] está dada por

$$\int_{-1}^{1} f(x)\,dx = \sum_{k=1}^{N} w_k f(x_k) \tag{4.6.1}$$

donde N es el número de puntos de Gauss, los w_i son los pesos y las x_i son los puntos de Gauss dados en la tabla 4.4. Los signos $\pm$ en la tabla significan que los valores de x de los puntos de Gauss aparecen son pares, uno de los cuales es positivo y el otro negativo. Por ejemplo, si $N = 4$, la ecuación (4.6.1) es

$$\begin{aligned}\int_{-1}^{1} f(x)\,dx = {} & 0.34785f(-0.86113) + 0.65214f(-0.33998) \\ & + 0.65214f(0.33998) + 0.34785f(0.86113)\end{aligned} \tag{4.6.2}$$

Tabla 4.4 Puntos de Gauss y pesos[a]

	$\pm x_i$	w_i
$N = 2$	0.577350269	1.00000000
$N = 3$	0	0.888888889
	0.774596669	0.555555556
$N = 4$	0.339981043	0.652145155
	0.861136312	0.347854845
$N = 5$	0	0.568888889
	0.538469310	0.478628670
	0.906179846	0.236926885
$N = 6$	0.238619186	0.467913935
	0.661209387	0.360761573
	0.932469514	0.171324492
$N = 8$	0.183434642	0.362683783
	0.525532410	0.313706646
	0.796666478	0.222381034
	0.960289857	0.101228536
$N = 10$	0.148874339	0.295524225
	0.433395394	0.269266719
	0.679409568	0.219086363
	0.865063367	0.149451349
	0.973906528	0.066671344

[a] Véase Abramowitz y Stegun para una tabla más completa.

La fórmula de integración de Gauss puede aplicarse a cualquier intervalo arbitrario $[a, b]$ con la transformación

$$x = \frac{2z - a - b}{b - a} \tag{4.6.3}$$

donde z es la coordenada original en $a < z < b$ y x es la coordenada normalizada en

$-1 \leqslant x \leqslant 1$. La transformación de x en z es

$$z = \frac{(b-a)x + a + b}{2} \tag{4.6.4}$$

Por medio de esta transformación, la integral se puede escribir como

$$\int_a^b f(z)\,dz = \int_{-1}^{1} f(z)(dz/dx)\,dx = \frac{b-a}{2}\sum_{k=1}^{N} w_k f(z_k) \tag{4.6.5}$$

donde $dz/dx = (b-a)/2$. Los valores de z_k se obtienen al sustituir x en la ecuación (4.6.4) por los puntos de Gauss; a saber,

$$z_k = \frac{(b-a)x_k + a + b}{2} \tag{4.6.6}$$

Por ejemplo, supongamos que $N = 2$, $a = 0$ y $b = 2$. Puesto que los puntos de Gauss x_k para $N = 2$ en la coordenada normalizada x, $-1 \leqslant x \leqslant 1$, son ± 0.57735 (de la tabla 4.4), los puntos correspndientes en z son

$$\begin{aligned} z_1 &= \frac{1}{2}[(2-0)(-0.57735) + 0 + 2] = 0.42265 \\ z_2 &= \frac{1}{2}[(2-0)(0.57735) + 0 + 2] = 1.57735 \end{aligned} \tag{4.6.7}$$

La derivada es $dz/dx = (b-a)/2 = 1$. Por lo tanto, la cuadratura de Gauss se escribe como

$$\int_0^2 f(z)\,dz = \int_{-1}^{1} f(z)(dz/dx)\,dx = (1)[(1)f(0.42264) + (1)f(1.57735)] \tag{4.6.8}$$

Mostramos que si $f(x)$ es un polinomio de orden menor o igual que $2N - 1$, la cuadratura de Gauss de orden N es exacta.

Supongamos que $f(x)$ en la ecuación (4.6.1) es un polinomio de orden menor o igual que $2N - 1$ y que se va a integrar en $[-1, 1]$. Mediante el polinomio de Legendre de orden N — a saber $P_N(x)$ — $f(x)$ se puede escribir como

$$f(x) = c(x)P_N(x) + r(x) \tag{4.6.9}$$

donde $c(x)$ y $r(x)$ son polinomios de orden menor o igual que $N - 1$. Las dos características de esta expresión que se presentan en seguida, serán importantes más adelante. En primer lugar, al integrar la ecuación (4.6.9) en $[-1, 1]$ se obtiene

$$\int_{-1}^{1} f(x)\,dx = \int_{-1}^{1} r(x)\,dx \tag{4.6.10}$$

donde el primer término de la ecuación (4.6.9) se anula después de integrar, debido a que $P_N(x)$ es otorgonal a cualquier polinomio de orden menor o igual que $N - 1$ (váse el apéndice B). En segundo lugar, si hacemos x igual a una de las raíces de $P_N(x) = 0$, el primer término de la ecuación (4.6.9) se anula, con lo que la ecuación (4.6.9) se reduce a $f(x_i) = r(x_i)$.

El integrando $r(x)$ en la ecuación (4.6.10) es un polinomio de orden menor o igual que $N - 1$, por lo que se puede expresar de forma exacta mediante la interpolación de Lagrange de orden $N - 1$ (véase la ecuación (2.3.3))

$$r(x) = \sum_{i=1}^{N} \left[\prod_{\substack{j=1 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j} \right] r(x_i) \tag{4.6.11}$$

donde $\prod$ denota una producto múltiple tomado con el subíndice j. De la ecuación (4.6.9), podemos ver que, debido a que x_i, $i = 1, 2, \ldots, N$, son las raíces de $P_N(x)$ y $r(x_i) = f(x_i)$, la ecuación (4.6.11) se puede escribir como

$$r(x) = \sum_{i=1}^{N} \left[\prod_{\substack{j=1 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j} \right] f(x_i) \tag{4.6.12}$$

La cuadratura de Gauss se obtiene al sustituir la ecuación (4.6.12) en la ecuación (4.6.10):

$$\int_{-1}^{1} f(x)\,dx = \sum_{i=1}^{N} f(x_i) \int_{-1}^{1} \left[\prod_{\substack{j=1 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j} \right] dx \tag{4.6.13}$$

La ecuación (4.6.13) se convierte en la ecuación (4.6.1) si definimos

$$w_i = \int_{-1}^{1} \left[\prod_{\substack{j=1 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j} \right] dx \tag{4.6.14}$$

Como hemos visto de la deducción anterior, la cuadratura de Gauss es exacta si el integrando $f(x)$ es un polinomio de orden menor o igual que $2N - 1$.

La tabla 4.4 indica que todos los pesos son positivos. Una ventaja de la cuadratura de Gauss es que no existen problemas con el error de redondeo, a menos que el integrando cambie de signo a mitad del dominio, debido a que no se realizan restas de números grandes como en el caso de las fórmulas de Newton-Cotes. Otra ventaja de las cuadraturas de Gauss es que se puede integrar una función con una singularidad en uno o ambos límites de integración, debido a que no se necesitan los valores en los límites.

En un intervalo de integración grande, el dominio se puede dividir en cierto número de intervalos pequeños y aplicar varias veces la cuadratura de Gauss en cada subintervalo. La idea es la misma que en las reglas extendidas del trapecio y de Simpson.

4.6.2 Otras cuadraturas de Gauss

Las cuadraturas de Gauss analizadas en la sección anterior se llaman cuadraturas de Gauss-Legendre porque se basan en la ortogonalidad de los polinomjos de Legendre. Existen cuadraturas análogas con base en los polinomios de Hermite, de Laguerre [Froeberg] y de Chebyshev, las cuales reciben los nombres de cuadraturas de Gauss-Hermite, Gauss-Laguerre y Gauss-Chebyshev, respectivamente.

Las cuadraturas de Gauss-Hermite son adecuadas para

$$\int_{-\infty}^{\infty} \exp(-x^2) f(x)\, dx \tag{4.6.15}$$

y están dadas por

$$\int_{-\infty}^{\infty} \exp(-x^2) f(x)\, dx = \sum_{k=1}^{N} w_k f(x_k) \tag{4.6.16}$$

En la ecuación (4.6.16), los x_k son raíces del polinomio de Hermite de orden N y los w_k son los pesos, algunos de los cuales se muestran en la tabla 4.5.

Tabla 4.5 Puntos de Hermite y sus pesos

	$\pm x_i$	w_i
$N = 2$	0.70710678	0.88622692
$N = 3$	0.00000000	1.18163590
	1.22474487	0.29540897
$N = 4$	0.52464762	0.80491409
	1.65068012	0.08131283
$N = 5$	0.00000000	0.94530872
	0.95857246	0.39361932
	2.02018287	0.01995324

Las cuadraturas de Gauss-Laguerre son adecuadas para

$$\int_{0}^{\infty} \exp(-x) f(x)\, dx \tag{4.6.17}$$

y están dadas por

$$\int_{0}^{\infty} \exp(-x) f(x)\, dx = \sum_{k=1}^{N} w_k f(x_k) \tag{4.6.18}$$

donde los x_k son las raíces del polinomio de Laguerre de orden N y los w_k son los pesos, algunos de los cuales se muestran en la tabla 4.6.

Las cuadraturas de Gauss-Chebyshev son adecuadas para

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)\, dx \tag{4.6.19}$$

Tabla 4.6 Puntos de Laguerre y sus pesos

	x_i	w_i
$N = 2$	0.58578643	0.85355339
	3.41421356	0.14644660
$N = 3$	0.41577455	0.71109300
	2.29428036	0.27851773
	6.28994508	$0.10389256E - 1$
$N = 4$	0.32254768	0.60315410
	1.74576110	0.35741869
	4.53662029	$0.38887908E - 1$
	9.39507091	$0.53929470E - 3$

y están dadas por

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)\,dx = \sum_{k=1}^{N} w_k f(x_k) \tag{4.6.20}$$

En la ecuación (4.6.20), los x_k son las raíces del polinomio de Chebyshev de orden N y los w_k son los pesos. Las raíces del polinomio de Chebyshev de orden N son

$$x_k = \cos \frac{k - 1/2}{N} \pi, \quad k = 1, 2, \ldots, N \tag{4.6.21}$$

Los pesos son

$$w_k = \frac{\pi}{N} \quad \text{para toda } k \tag{4.6.22}$$

Así, la ecuación (4.6.20) se reduce a

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)\,dx = \frac{\pi}{N} \sum_{k=1}^{N} f(x_k) \tag{4.6.23}$$

Los límites de integración de $[-1, 1]$ se pueden cambiar a un dominio arbitrario $[a, b]$ mediante la ecuación (4.6.4).

Las tres cuadraturas de Gauss explicadas en esta subsección son exactas si $f(x)$ es un polinomio de orden menor o igual que $2N - 1$. Para más detalles de las cuadraturas de Gauss véase [Carnahan/Luther/Wilkes y King].

Resumen de Esta Sección

a) Las cuadraturas de Gauss (–Legendre) se basan en la integración de un polinomio ajustado a los puntos dados por las raíces de un polinomio de Legendre.

b) El orden de exactitud de una cuadratura es aproximadamente el doble del de la fórmula cerrada de Newton-Cotes, al utilizar el mismo número de puntos.
c) Debido a que los coeficientes son positivos, no existen errores graves de redondeo, siempre y cuando el integrando no cambie de signo dentro de los límites de integración.
d) Existen otras cuadraturas de Gauss adecuadas para integrales especiales.

4.7 INTEGRACION NUMERICA CON LIMITES INFINITOS O SINGULARIDADES

En esta sección estudiaremos los siguientes tipos de integrales, los cuales merecen una atención especial:

$$I = \int_{-\infty}^{\infty} \exp(-x^2)\,dx \tag{4.7.1}$$

$$I = \int_0^1 \frac{1}{\sqrt{x}(e^x + 1)}\,dx \tag{4.7.2a}$$

$$I = \int_0^1 x^{0.7} \cos(x)\,dx \tag{4.7.2b}$$

En la ecuación (4.7.1), la integral se extiende en un dominio infinito. La ecuación (4.7.2a) involucra una singularidad del integrando en $x = 0$ (la función tiende a infinito cuando x tiende a 0). La ecuación (4.7.2b) parece no tener una singularidad; no obstante, no es un problema trivial para ninguno de los métodos descritos en las secciones anteriores. De hecho, si aplicáramos las reglas extendidas del trapecio y de Simpson, la respuesta cambiará conforme dupliquemos el número de intervalos. La razón para este comportamiento es que la función no es analítica en $x = 0$.

Esta sección analiza los métodos que se basan en la regla del trapecio y la transformación exponencial doble, métodos que son robustos y funcionan sin muchos problemas, en una amplia variedad de problemas. Sin embargo, para estudiar los enfoques más tradicionales, existe un resumen en Stoer y Burlish.

Una función integrable en un dominio infinito o semi-infinito es casi nula, excepto en cierta parte del dominio. La contribución principal a la integral proviene de un dominio relativamente pequeño, en donde la función es distinta de cero en forma significativa.

Se ha demostrado [Takahashi/Mori; Mori/Piessens] que si $f(x)$ es analítica en $(-\infty, \infty)$, el método más eficiente para la integración numérica de

$$I = \int_{-\infty}^{\infty} f(x)\,dx \tag{4.7.3}$$

es la regla extendida del trapecio

$$I = h \sum_{i=-M}^{M} f(x_i) \tag{4.7.4}$$

donde $x_i = ih$ y M es un entero suficientemente grande.

De hecho, el ejemplo 4.6 muestra que la regla extendida del trapecio produce resultados muy precisos con un número relativamente pequeño de puntos.

Ejemplo 4.6

Evalúe numéricamente

$$I = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \exp(-x^2)\, dx$$

mediante la regla extendida del trapecio, con $h = 0.5$, 0.1 y 0.01.

(Solución)

Remplazamos los límites de integración por -10 y 10, que son valores suficientemente grandes para este problema:

$$I = \frac{1}{\sqrt{\pi}} \int_{-10}^{10} \exp(-x^2)\, dx$$

Por medio de la regla extendida del trapecio en el PROGRAMA 4-1, obtenemos los siguientes resultados:

N	I
20	1.000104
40	1.000001
80	1.000000

donde N es el número de intervalos. El valor exacto es 1.0, por lo que la aproximación es muy buena con $N = 20$.

A continuación, consideremos el problema de integrar una función en un dominio finito, pero singular en uno o ambos límites. Véase la ecuación (4.7.2) como ejemplo. El dominio finito de integración, $[a, b]$ se puede transformar en $(-\infty, \infty)$ mediante una transformación de coordenadas. Una vez que se reduce el problema a la ecuación (4.7.1), se aplica la regla extendida del trapecio.

Sea

$$I = \int_b^a f(x)\, dx \tag{4.7.5}$$

donde a y b son finitos. La transformación se puede escribir como

$$z = z(x)$$

o, en forma equivalente

$$x = x(z) \tag{4.7.6}$$

donde z es una transformación que satisface

$$z(a) = -\infty$$
$$z(b) = \infty \tag{4.7.7}$$

Por lo tanto, la ecuación (4.7.5) se puede escribir como

$$I = \int_{-\infty}^{\infty} f[x(z)]\left(\frac{dx}{dz}\right)dz \tag{4.7.8}$$

Un ejemplo de tal transformación es la transformación exponencial dada por

$$x = \frac{1}{2}[a + b + (b - a)\tanh(z)] \tag{4.7.9}$$

o en forma equivalente

$$z = \tanh^{-1}\left(\frac{2x - a - b}{b - a}\right) \tag{4.7.10}$$

Sin embargo, la exactitud de la integración numérica se ve afectada por la elección de la transformación. La transformación exponencial doble dada por

$$x = \frac{1}{2}\left[a + b + (b - a)\tanh\left(\frac{\pi}{2}\operatorname{senh}(z)\right)\right] \tag{4.7.11}$$

se ha propuesto como la elección óptima. Con esta elección, dx/dz es

$$dx/dz = \frac{(b - a)\dfrac{\pi}{4}\cosh(z)}{\cosh^2\left[\dfrac{\pi}{2}\operatorname{senh}(z)\right]} \tag{4.7.12}$$

Al sustituir las ecuaciones (4.7.11) y (4.7.12) en la ecuación (4.7.8) y al aplicar la regla extendida del trapecio se tiene

$$I = h\sum_{k=-N}^{N} f(x_k)\left(\frac{dx}{dz}\right)_k \tag{4.7.13}$$

donde N es un entero suficientemente grande y

$$z_k = kh \tag{4.7.14}$$

$$x_k = \frac{1}{2}\left[a + b + (b - a)\tanh\left(\frac{\pi}{2}\,\underline{\text{s}}\text{enh}(z_k)\right)\right] \tag{4.7.15}$$

$$(dx/dz)_k = \frac{(b - a)\dfrac{\pi}{4}\cosh(z_k)}{\cosh^2\left[\dfrac{\pi}{2}\,\text{senh}(z_k)\right]} \tag{4.7.16}$$

En la ecuación (4.7.13), cabría preguntarse cuán grande debiera ser la N, lo cual se puede contestar examinando el denominador de la ecuación (4.7.16). Cuando z_k crece, tiende a

$$\cosh^2\left[\frac{\pi}{2}\,\underline{\text{s}}\text{enh}(z_k)\right] \longrightarrow \frac{1}{4}\exp\left[\frac{\pi}{2}\exp(z_k)\right] \tag{4.7.17}$$

lo cual crece de una forma doblemente exponencial y provoca un desbordamiento. Con variables de precisión simple en la IBM PC y en una computadora VAX esto ocurre cuando

$$\frac{1}{4}\exp\left[\frac{\pi}{2}\exp(z_k)\right] = 2 \times 10^{38} \tag{4.7.18}$$

o, en forma equivalente, si z_k es mayor que 4.0, aproximadamente. Este criterio determina el máximo N que puede utilizarse. En la IBM PC es

$$Nh < 4 \quad \text{(IBM PC)} \tag{4.7.19}$$

En una mainframe de IBM, el máximo valor de punto flotante en FORTRAN-77 es 7.5×10^{75} por lo que el criterio para N es

$$Nh < 4.7 \quad \text{(mainframe de IBM)} \tag{4.7.20}$$

Otro problema es el error de redondeo en la ecuación (4.7.15) que aparece cuando el término con la tangente hiperbólica se acerca mucho a —1 o 1. Para evitar esto, primero nos percatamos que el término con la tangente hiperbólica se puede escribir como

$$\tanh(p) = \left(s - \frac{1}{s}\right)\Big/\left(s + \frac{1}{s}\right) \tag{4.7.21}$$

donde $s = \exp(p)$. Por medio de la ecuación (4.7.21), la ecuación (4.7.15) se escribe en la forma

$$x_k = \left(bs + \frac{a}{s}\right)\Big/\left(s + \frac{1}{s}\right) \tag{4.7.22}$$

donde

$$s = \exp\left[\frac{\pi}{2}\,\mathrm{senh}\,(z_k)\right]$$

Puesto que la ecuación (4.7.22) no tiene la operación de sustracción, se eliminan los errores serios de redondeo.

El PROGRAMA 4-5 lleva a cabo la integración por medio de la transformación exponencial doble. Para los límites infinitos $(-\infty, \infty)$, se utiliza en forma directa la regla del trapecio en el PROGRAMA 4-1 en vez del PROGRAMA 4-5.

Ejemplo 4.7

La longitud de la curva dada por $y = g(x)$, $a < x < b$, es

$$l = \int_a^b \sqrt{1 + (g'(x))^2}\,dx$$

Calcule la longitud del arco parabólico $y^2 = 4x$ en $0 < x < 2$.

(Solución)

Puesto que $g(x) = 2\sqrt{x}$, su derivada es $g'(x) = 1/\sqrt{x}$. La integral es

$$l = \int_0^2 \sqrt{1 + \frac{1}{x}}\,dx$$

El integrando en la ecuación anterior es singular en $x = 0$.

El cálculo se lleva a cabo utilizando la versión en BASIC del PROGRAMA 4-5 en una IBM PC. En el programa, los límites de integración en la coordenada transformada se hacen iguales a $z = -4$ y $z = 4$. Los resultados obtenidos son:

N	l
10	3.600710
20	3.595706
30	3.595706

donde N es el número de intervalos utilizados en la regla extendida del trapecio.

RESUMEN DE ESTA SECCIÓN

a) La regla extendida del trapecio es la óptima para integrar una función analítica en $(-\infty, \infty)$.

b) La integración de una función con una singularidad se puede llevar a cabo mediante la regla extendida del trapecio con la transformación exponencial doble.

4.8 INTEGRACION NUMERICA EN UN DOMINIO BIDIMENSIONAL

Consideremos un dominio como el que se muestra en la figura 4.5, en el que las fronteras izquierda y derecha son segmentos de recta verticales y las fronteras inferior y superior están dadas por curvas $y = d(x)$ y $y = c(x)$, respectivamente. La doble integral en el dominio se escribe como

$$I = \int_a^b \left[\int_{c(x)}^{d(x)} f(x, y)\, dy \right] dx \tag{4.8.1}$$

Sin embargo, los problemas de dobles integrales no siempre se pueden escribir en la forma de la ecuación (4.8.1). A menudo tienen formas distintas como

$$I = \int_a^b \int_{c(x)}^{d(x)} f(x, y)\, dy\, dx$$

$$I = \int_a^b dx \int_{c(x)}^{d(x)} dy\, f(x, y)$$

o incluso

$$I = \iint_A f(x, y)\, dx\, dy$$

donde A representa el dominio. En cualquiera de estos casos, el problema debe reescribirse en la forma de la ecuación (4.8.1) antes de proseguir con la integración numérica. Se deben intercambiar x y y en caso necesario.

El principio general de la integración numérica de la ecuación anterior es reducirla a una combinación de problemas unidimensionales. Si definimos

$$G(x) \equiv \int_{c(x)}^{d(x)} f(x, y)\, dy \tag{4.8.2}$$

entonces, la ecuación (4.8.1) queda

$$I = \int_a^b G(x)\, dx \tag{4.8.3}$$

a la cual se puede aplicar cualquiera de las fórmulas de cuadratura numéricas descrita anteriormente y se expresa en la forma:

$$I = \sum_{i=0}^{N} w_i G(x_i) \tag{4.8.4}$$

donde w_i son los pesos y los x_i son los puntos de la cuadratura particular. Los valores numéricos de $G(x_i)$ también se evalúan en forma numérica. Si hacemos $x = x_i$, la ecuación (4.8.2) queda

$$G(x_i) \equiv \int_{c(x_i)}^{d(x_i)} f(x_i, y)\, dy \tag{4.8.5}$$

que es un problema unidimensional ya que la única variable del integrando es y. La ecuación (4.8.5) se puede evaluar mediante alguna de las cuadraturas numéricas.

Para ejemplificar, aplicamos la regla extendida del trapecio al problema de integración doble:

$$I = \int_a^b \left[\int_{c(x)}^{d(x)} f(x, y)\, dy \right] dx \tag{4.8.6}$$

El rango de integración $[a, b]$ se divide en N intervalos con igual separación, con un tamaño del intervalo dado por $h_x = (b - a)/N$ (véase la figura 4.5, en donde se supone $N = 4$). Los puntos de la retícula se denotarán como $x_0, x_1, x_2, \ldots, x_N$. Al aplicar la regla del trapecio en el eje x, obtenemos

$$I = \frac{h_x}{2} \left[\int_{c(x_0)}^{d(x_0)} f(x_0, y)\, dy + 2 \int_{c(x_1)}^{d(x_1)} f(x_1, y)\, dy \right.$$
$$\left. + 2 \int_{c(x_2)}^{d(x_2)} f(x_2, y)\, dy + \cdots + \int_{c(x_N)}^{d(x_N)} f(x_N, y)\, dy \right] \tag{4.8.7}$$

La ecuación (4.8.7) se puede escribir en forma más compacta como

$$I = (h_x/2)[G(x_0) + 2G(x_1) + 2G(x_2) + \cdots + G(x_N)] \tag{4.8.8}$$

donde

$$G(x_i) = \int_{c(x_i)}^{d(x_i)} f(x_i, y)\, dy \tag{4.8.9}$$

Al evaluar la ecuación (4.8.9), el dominio de integración $[c(x_i), d(x_i)]$ se divide en N intervalos con un tamaño de

$$h_y = \frac{1}{N} [d(x_i) - c(x_i)]$$

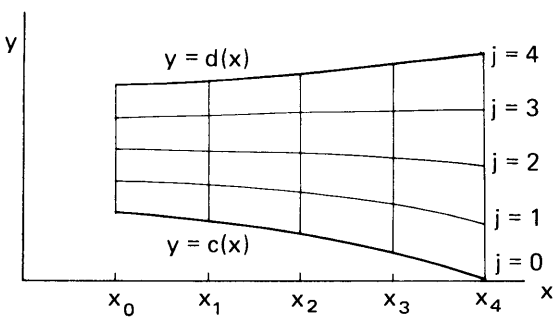


Figura 4.5 Dominio bidimensional para la doble integración

Los valores y de los puntos de la retícula se denotan como $y_{i,0}, y_{i,1}, y_{i,2}, \ldots, y_{i,N}$. Entonces, la integración mediante la regla extendida del trapecio da como resultado

$$G(x_i) = \int_{c(x_i)}^{d(x_i)} f(x_i, y)\,dy$$

$$= \frac{h_y}{2}\left[f(x_i, y_{i,0}) + 2f(x_i, y_{i,1}) + 2f(x_i, y_{i,2}) + \cdots + f(x_i, y_{i,N})\right] \quad (4.8.10)$$

La regla del trapecio utilizada en las integrales anteriores puede remplazarse por cualquier otro método de integración numérica, incluyendo las cuadraturas de Gauss y las reglas de Simpson [Press/Flannery/Teukolsky/Vetterling, pág. 126].

Ejemplo 4.8

Evalúe la siguiente integral doble

$$I = \int_a^b \left[\int_{c(x)}^{d(x)} \operatorname{sen}(x + y)\,dy\right] dx$$

mediante la regla de 1/3 de Simpson. Los límites de integración son

$$a = 1, \quad b = 3$$

$$c(x) = \ln(x)$$

$$d(x) = 3 + \exp(x/5)$$

(Solución)

Para la regla de 1/3 de Simpson, los puntos de la retícula en el eje x son

$$x_0 = 1, \quad x_1 = 2, \quad x_2 = 3$$

En la figura E4.8 aparecen el dominio de integración y los puntos de la retícula. Al aplicar la regla de 1/3 de Simpson a la primera integral obtenemos

$$I = \frac{h_x}{3}\left[G(x_0) + 4G(x_1) + G(x_2)\right]$$

donde $h_x = (b - a)/2 = 1$ y

$$G(x_i) = \int_{\ln(x_i)}^{3+\exp(x_i/5)} \operatorname{sen}(x_i + y)\,dy$$

o más explícitamente

$$I = \int_a^b \left[\int_{c(x)}^{d(x)} \operatorname{sen}(x + y)\,dy\right] dx$$

$$= \frac{h_x}{3}\left[\int_{\ln(1)}^{3+\exp(1/5)} \operatorname{sen}(1 + y)\,dy + 4\int_{\ln(2)}^{3+\exp(2/5)} \operatorname{sen}(2 + y)\,dy + \int_{\ln(3)}^{3+\exp(3/5)} \operatorname{sen}(3 + y)\,dy\right]$$

$$= \frac{h_x}{3}\left[\int_0^{4.2214} \operatorname{sen}(1 + y)\,dy + 4\int_{0.6931}^{4.4918} \operatorname{sen}(2 + y)\,dy + \int_{1.0986}^{4.8221} \operatorname{sen}(3 + y)\,dy\right]$$

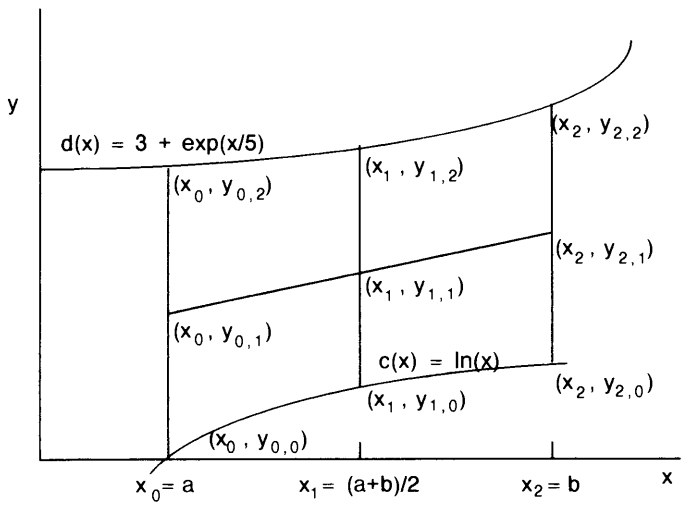


Figura E4.8 Retícula para la integración doble

Con la regla de 1/3 de Simpson, la primera integral del renglón anterior queda

$$\int_0^{4.2214} \text{sen}(1 + y)\, dy$$

$$= \frac{2.11070}{3}[\text{sen}(1 + 0) + 4\,\text{sen}(1 + 2.11070) + \text{sen}(1 + 4.2214)]$$

$$= \frac{2.11070}{3}[0.84147 + (4)(0.03088) + (-0.87322)]$$

$$= 0.064581$$

Cálculos análogos dan por resultado

$$\int_{0.6931}^{4.4918} \text{sen}(2 + y)\, dx = -2.1086$$

$$\int_{1.0986}^{4.8221} \text{sen}(3 + y)\, dx = -0.67454$$

Así, el valor final de la integral doble es

$$I = \frac{1}{3}[0.064581 + (4)(-2.1086) - 0.67454]$$

$$= -3.0148$$

Resumen de Esta Sección

a) La doble integración numérica es la doble aplicación de un método de integración numérica para las integrales de una sola variable, una vez para la dirección y y otra vez para la dirección x.

b) Cualquier método de integración numérica para integrales de una sola variable se puede aplicar a las integrales dobles.

PROGRAMAS

PROGRAMA 4-1 Reglas extendidas del trapecio y de Simpson

A) Explicaciones

El PROGRAMA 4-1 integra una función analítica, ya sea mediante la regla extendida del trapecio o mediante la regla extendida de Simpson, según la elección del usuario. Antes de correr el programa, el usuario debe definir el integrando en el subprograma FUNC. El usuario puede dar como entrada la elección de un método de integración, los límites de integración y el número de intervalos, en forma interactiva desde el teclado. Si el número de intervalos para la regla de Simpson es impar, se utiliza la regla de 3/8 para los primeros tres intervalos de la retícula y después se utiliza la regla extendida de 1/3 para el resto del dominio.

Cuando se corre el programa, la computadora envía un mensaje para recordar al usuario que la función a integrar debe definirse en el subprograma FUNC, después le pregunta el método que usará para integrar; si es el de SIMPSON la entrada debe ser 1. A continuación, el programa pregunta los valores de N, A y B.

B) Variables

ISIMP: especificación del método
ISIMP = 0 (Regla del trapecio)
ISIMP = 1 (Regla de Simpson)
A, B: límite inferior y superior de integración, respectivamente
N: número de intervalos en la retícula
H: espaciamiento, H = (B — A)/N
W: valores de los pesos en las fórmulas de integración
S, SS: integral
II: último punto de la retícula para la regla de 3/8

C) Listado

```
C-----CSL/F4-1.FOR     REGLAS DEL TRAPECIO Y DE SIMPSON
      COMMON A,B,H
      CHARACTER SIMP*6
      PRINT *
      PRINT *,'CSL/F4-1     REGLAS DEL TRAPECIO Y DE SIMPSON'
      PRINT *
      PRINT *,'LA FUNCION A INTEGRAR SE DEBE CODIFICAR EN
      PRINT *,'EL SUBPROGRAMA LLAMADO FUNC
      PRINT *
10    PRINT *,'OPRIMA O PARA EL TRAPECIO, 1 PARA SIMPSON
      READ *, ISIMP
      PRINT *,'¿NUMERO DE INTERVALOS?
      READ *,N
135   IF (N .GT.0.AND. ISIMP.EQ.0) GOTO 140
      IF (ISIMP.EQ.1.AND.N.GT.1) GOTO 140
```

```
            PRINT *,'LA ENTRADA NO ES VALIDA, REPITA '
            GO TO 10
140     PRINT *,'¿LIMITE INFERIOR DE INTEGRACION?     '
        READ *,A
150     PRINT *,'LIMITE SUPERIOR DE INTEGRACION ? '
        READ *,B
160     H=(B-A)/N
        IF (ISIMP.EQ.0) THEN
           CALL  TRAPZ(S,N)          !-- Se eligió la regla del trapecio.
           GOTO 200
           ELSE
           CALL  SIMPS(S,N)          !-- Se eligió la regla de Simpson.
        END IF
200     PRINT *,'-----------------------------------------'
210     PRINT *,'RESULTADO FINAL=',S
220     PRINT *,'-----------------------------------------'
        PRINT *
        PRINT*
        PRINT*,'OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
        READ *,K
        IF(K.EQ.1) GOTO 10
        PRINT*
        END
C*****************************
        SUBROUTINE TRAPZ(S,N)     !-- Regla del trapecio
        COMMON A,B,H
        S=0
        DO 10 I=0,N
           X=A+I*H
           W=2
           IF(I.EQ.0 .OR. I.EQ.N) W = 1
           S=S+W*FUNC(X)
c           PRINT *,I,X,H,FUNC(X),W
10      CONTINUE
        S=S*H/2
        RETURN
        END
C*****************************
        SUBROUTINE SIMPS(SS,N)
        COMMON A,B,H
        S=0
        SS=0
        IF (N/2*2.EQ.N) THEN
           LS = 0
           GOTO 35
        END IF
           LS=3
           DO 30 I=0,3               !  Regla de 3/8 de Simpson si N es impar
             X=A+H*I
             W=3
             IF (I.EQ.0 .OR. I.EQ.3) W=1
             SS=SS+W*FUNC(X)
30         CONTINUE
        SS=SS*H*3/8
        IF (N.EQ.3) RETURN
35         DO 40 I=0, N-LS           !  Regla de 1/3 de Simpson
             X=A+H*(I+LS)
             W=2
             IF (INT(I/2)*2+1.EQ.I) W=4
             IF (I.EQ.0 .OR. I.EQ.N-LS)   W=1
             S=S+W*FUNC(X)
```

```
40        CONTINUE
      SS=SS+S*H/3
      RETURN
      END
C*******************
      FUNCTION FUNC(X)     !-- Evalúa la función a integrar
      FUNC = (1 + (X/2)**2)**2*3.14159
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F4-1      REGLAS DEL TRAPECIO Y DE SIMPSON
LA FUNCION A INTEGRAR SE DEBE CODIFICAR EN EL SUBPROGRAMA LLAMADO FUNC
OPRIMA 0 PARA EL TRAPECIO, 1 PARA SIMPSON
0
¿NUMERO DE INTERVALOS?
10
¿LIMITE INFERIOR DE INTEGRACION?
0
¿LIMITE SUPERIOR DE INTEGRACION?
2
-----------------------------------------
RESULTADO FINAL    11.77047
-----------------------------------------
OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR'
OPRIMA 0 PARA EL TRAPECIO, 1 PARA SIMPSON
1
¿NUMERO DE DATOS?
5
LIMITE INFERIOR DE INTEGRACION ?
0
LIMITE SUPERIOR DE INTEGRACION ?
2
-----------------------------------------
RESULTADO FINAL    11.73095
-----------------------------------------
OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR
```

PROGRAMA 4-2 Fórmulas cerradas de Newton-Cotes

A) Explicaciones

El PROGRAMA 4-2 lleva a cabo la integración numérica utilizando las fórmulas cerradas de Newton-Cotes, mientras que el PROGRAMA 4-3 utiliza las fórmulas abiertas.

Antes de correr el programa, el usuario debe definir la función a integrar en el subprograma FUN, donde aparece F = SIN (X) como ejemplo. Las instrucciones DATA contienen los valores de W, Q y R copiados de la tabla 4.2. El programa pregunta por el orden del método y los límites de integración.

B) Variables

N: orden de la fórmula de Newton-Cotes, $2 \leq N \leq 10$

A y B: límites inferior y superior de la integral

W(J, N): J-ésimo factor de peso en la fórmula de orden N (tablas 4.2 y 4.3)

Q, R: numerador y denominador de $\alpha = Q/R$ (tablas 4.2 y 4.4)

H: espaciamiento, (B — A)/N

I: integral (respuesta final)

C) Listado

```
C-----CSL/F4-2.FOR        FORMULA CERRADA DE NEWTON-COTES
      DIMENSION W(0:20,10)
      PRINT *
      PRINT *,'CSL/F4-2     FORMULA CERRADA DE NEWTON COTES '
      PRINT *
      DATA (W(I,1),I=0,3)/1,1,1,2/
      DATA (W(I,2),I=0,4)/1,4,1,1,3/
      DATA (W(I,3),I=0,5)/1,3,3,1,3,8/
      DATA (W(I,4),I=0,6)/7,32,12,32,7,2,45/
      DATA (W(I,5),I=0,7)/19,75,50,50,75,19,5,288/
      DATA (W(I,6),I=0,8)/41,216,27,272,27,216,41,1,140/
      DATA (W(I,7),I=0,7)/751,3577,1323,2989,2989,1323,3577,751/
      DATA (W(I,7),I=8,9)/7,17280/
      DATA (W(I,8),I=0,8)/989,5888,-928, 10946 ,-4540, 10946 ,-928,5888,989/
      DATA (W(I,8),I=9,10)/4,14175/
      DATA (W(I,9),I=0,7)/2857,15741,1080,19344,5788,5788,19344,1080/
      DATA (W(I,9),I=8,11)/15741,2857,9,89600/
      DATA (W(I,10),I=0,5)/16067, 106300,--48525 ,272400,-260550, 427368 /
      DATA (W(I,10),I=6,10)/-260550,272400,-48525,106300,16067/
      DATA (W(I,10),I=11,12)/5,299376/
1     PRINT *,'¿NUMERO DE DATOS?                (2 - 10) '
      READ *,K
      PRINT *,'A  (LIMITE INFERIOR DE INTEGRACION) ?'
      READ *,A
      PRINT *,'B  (LIMITE SUPERIOR DE INTEGRACION) ? '
      READ *,B
      PRINT *
      N=K-1
      Q=W(N+1,N)
      R=W(N+2,N)
      PRINT 10,Q,R
10    FORMAT(' Q=',1PE14.6, '     R=', 1PE14.6)
      PRINT *
      AL=Q/R                 ! alfa
      H=(B-A)/N              ! Intervalo de la retícula
      PRINT*,'-------------------------------------------------------- '
      PRINT*,'     N          X                F(X)                W          '
      PRINT*,'-------------------------------------------------------- '
220   ANS=0          ! Inicialización de la fórmula de Newton-Cotes
      DO 240 J=0, N                          !--J es el índice de los puntos de la retícula
         X=A+J*H
         F=FUN(X)
         PRINT 250, J,X,F,W(J,N)
         ANS=ANS+F*W(J,N)                !-- Fórmula de Newton-Cotes
240   CONTINUE
250    FORMAT(1X,I5,1P2E15.6, F13.4)
      ANS=ANS*H*AL
      PRINT *
```

```
PRINT*,'-------------------------------------------------- '
PRINT*,'      RESULTADO FINAL       I=', ANS
PRINT*,'-------------------------------------------------- '
PRINT*
PRINT*,' OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
READ *,K
IF(K.EQ.1) GOTO 1
PRINT*
END
C***********************
      FUNCTION FUN(X)    ! - - Evalúa la función a integrar
      FUN=SIN(X)
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F4-2        FORMULA CERRADA DE NEWTON-COTES

¿ NUMERO DE DATOS ? (2 - 10)
6
A (LIMITE INFERIOR DE INTEGRACION) ?
0
B (LIMITE SUPERIOR DE INTEGRACION) ?
2
Q=  5.000000E+00     R=  2.880000E+02
-----------------------------------------------------
    N        X              F(X)              W
-----------------------------------------------------
    0   0.000000E+00   0.000000E+00     190.0000
    1   4.000000E-01   3.894183E-01     750.0000
    2   8.000000E-01   7.173561E-01     500.0000
    3   1.200000E+00   9.320391E-01     500.0000
    4   1.600000E+00   9.995736E-01     750.0000
    5   2.000000E+00   9.092974E-01     190.0000
-----------------------------------------------------
    RESULTADO FINAL        I=   1.416117
-----------------------------------------------------
OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR
```

PROGRAMA 4-3 Fórmulas abiertas de Newton-Cotes

A) Explicaciones

El PROGRAMA 4-3 desarrolla la integración numérica utilizando las fórmulas abiertas de Newton-Cotes. La estructura del PROGRAMA 4-3 es muy similar a la del PROGRAMA 4-2. Véase CSL para los detalles.

PROGRAMA 4-4 Cuadratura de Gauss

A) Explicaciones

El PROGRAMA 4-4 es para cuadratura de Gauss y contiene la tabla de cuadratura de Gauss.

Antes de ejecutar el programa, el usuario debe definir el integrando en el subprograma FUN. Al ejecutarse el programa pide al usuario que proporcione los valores de N, A y B por medio del teclado.

B) Variables

N: número de puntos utilizados en la retícula (el orden de la fórmula menos uno)
A: límite inferior de integración
B: límite superior de integración
XA(I): coordenada x de los puntos de la retícula
W(I): pesos de integración
F: integrando
XI: resultado calculado

C) Listado

```
C-----CSL/F4-4.FOR        CUADRATURA DE GAUSS
      DIMENSION W(0:10),XA(0:10)
      PRINT *
      PRINT *,'CSL/F4-4        CUADRATURA DE GAUSS
      PRINT *
17    PRINT *,'LOS ORDENES DE CUADRATURAS DISPONIBLES SON: N=2,3,4,5,6,8,10'
      PRINT *
      PRINT *,'N ?'
      READ *,N
      PRINT *
      IF (N .LT. 0)    STOP
      IF ((N.EQ.7) .OR. (N.EQ.9))   PRINT *,'N = 7 Y 9 NO ESTAN DISPONIBLES
      IF ((N.EQ.7) .OR. (N.EQ.9))   GOTO 17
      PRINT *,'¿EL LIMITE INFERIOR A?'
      READ *,A
      PRINT *,'¿EL LIMITE SUPERIOR B?'
      READ *,B
      IF (N .NE.2) GOTO 45
      XA(2)= 0.5773502691
      W(2)=1
      GOTO 200
45    IF (N .NE.3) GOTO 65
      XA(2)= 0
      XA(3)= 0.7745966692
      W(2)=0.8888888888
      W(3)=0.5555555555
      GOTO 200
65    IF (N.NE.4) GOTO 85
      XA(3)=0.3399810435
      XA(4)=0.8611363115
      W(3)=0.6521451548
      W(4)=0.3478548451
      GOTO 200
85    IF (N.NE.5) GOTO 115
      XA(3)=0
```

```
      XA(4)=0.5384693101
      XA(5)=0.9061798459
      W(3)=0.5688888888
      W(4)=0.4786286704
      W(5)=0.2369268850
      GOTO 200
115   IF ( N.NE.6) GOTO 135
      XA(4)=0.2386191860
      XA(5)=0.6612093864
      XA(6)=0.9324695142
      W(4)=0.4679139345
      W(5)=0.3607615730
      W(6)=0.1713244923
      GOTO 200
135   IF (N.NE.8) GOTO 175
      XA(5)=0.1834346424
      XA(6)=0.5255324099
      XA(7)=0.7966664774
      XA(8)=0.9602898564
      W(5)=0.3626837833
      W(6)=0.3137066458
      W(7)=0.2223810344
      W(8)=0.1012285362
      GOTO 200
175   IF (N.NE.10) GOTO 17
      XA(6)=0.1488743389
      XA(7)=0.4333953941
      XA(8)=0.6794095682
      XA(9)= 0.8650633666
      XA(10)=0.9739065285
      W(6)=0.2955242247
      W(7)=0.2692667193
      W(8)=0.2190863625
      W(9)=0.1494513491
      W(10)=0.0666713443
200   PI=3.1415927
      DO J=1,INT(N/2)
         W(J)=W(N+1-J)
         XA(J)=-XA(N+1-J)
      END DO
      PRINT *
      PRINT*,'-------------------------------------------------------'
      PRINT*,'     N          X              F(X)                W      '
      PRINT*,'-------------------------------------------------------'
      DO J=1,N
         XA(J)=(XA(J)*(B-A)+A+B)/2          ! Puntos de Gauss
      END DO
300   XI=0            ! Inicialización de la fórmula de cuadratura de Gauss
      DO J=1,N
         X=XA(J)
         CALL FUN(F,X)                      ! Cálculo de los valores de la función
         XI=XI+F*W(J)                         ! Fórmula de cuadratura de Gauss
         PRINT 20, J,X,F,W(J)
20       FORMAT(1X,I5,3F15.8)
      END DO
      XI=XI*(B-A)/2
325   PRINT *
      PRINT *,'---------------------------------------'
      PRINT *,'RESULTADO FINAL  I =',XI
340   PRINT *,'---------------------------------------'
```

```
350     PRINT *
        PRINT*
        PRINT*, 'PARA CONTINUAR, OPRIMA 1'
        READ *,K
        IF(K.EQ.1) GOTO 17
        PRINT*
        END
C*************************
        SUBROUTINE FUN(F,X)    ! Evalúa la función a integrar
        F=SIN(X)
        RETURN
        END
```

D) Ejemplo de salida

```
CSL/F4-4        CUADRATURA DE GAUSS

LOS ORDENES DE CUADRATURA DISPONIBLES SON:   N=2,3,4,5,6,8,10

N ?
6
¿EL LIMITE INFERIOR A?
0
¿EL LIMITE SUPERIOR B?
2
-----------------------------------------------------
    N          X              F(X)              W
-----------------------------------------------------
    1      0.06753051     0.06747920     0.17132449
    2      0.33879060     0.33234668     0.36076158
    3      0.76138079     0.68992162     0.46791393
    4      1.23861921     0.94533461     0.46791393
    5      1.66120946     0.99591553     0.36076158
    6      1.93246949     0.93530613     0.17132449
---------------------------------------
RESULTADO FINAL  I =    1.416147
---------------------------------------
PARA CONTINUAR, OPRIMA 1
```

PROGRAMA 4-5 Integración de una función singular

A) Explicaciones

El PROGRAMA 4-5 integra mediante la regla extendida del trapecio, una función que es singular en uno o ambos límites de integración, después de aplicar la transformación exponencial doble. La función a integrar se define en el subprograma FUN.

B) Variables

A: límite inferior de integración de x

B: límite superior de integración de x

DXDZ: dx/dz

H: tamaño del intervalo en la coordenada z

H: número de intervalos en la coordenada z

SS: resultado de la integral

C) Listado

```
C-----CSL/F4-5.FOR      INTEGRACION DE UNA FUNCION SINGULAR
C
C                 REGLA DEL TRAPECIO CON
C                 TRANSFORMACION EXPONENCIAL DOBLE
      DOUBLEPRECISION SS,P,W,Z,EXZ,HCOS,HSIN,S,PAI,DXDZ,X,FUN
      PRINT *
      PRINT *,'CSL/F4-5   INTEGRACIOIN DE UNA FUNCION SINGULAR MEDIANTE
      PRINT *,'           LA TRANSFORMACION EXPONENCIAL DOBLE '
      PRINT *
      PRINT *,'     (LA FUNCION ESTA DEFINIDA EN EL SUBPROGRAMA "FUN")'
1     PRINT*
      PRINT*,'¿NUMERO TOTAL DE INTERVALOS, N?'
      READ *,N
      PRINT*,'¿LIMITE INFERIOR DE INTEGRACION, A?'
      READ*,A
      PRINT*,'¿LIMITE SUPERIOR DE INTEGRACION, B?'
      READ*,B
      N=N/2                 ! La mitad de los puntos en la retícula
      H=4.0/N               ! determinan a h; el numerador puede ser más grande
C                           ! si los exponentes de la computadora así lo permiten
      PAI=3.1415927
      SS=0
      DO K=-N,N
         Z=H*K
         EXZ=DEXP(Z)
         HCOS=(EXZ+1.0/EXZ)/2.
         HSIN=(EXZ-1.0/EXZ)/2.
         S=DEXP(PAI*0.5*HSIN)
         X=(B*S+A/S)/(S+1.0/S)
         IF (X.NE.A.AND.X.NE.B) THEN
            P=PAI/2.0*HSIN
            W=DEXP(P)
            DXDZ=(B-A)*PAI/2.0*HCOS/((W+1.0/W)/2.0)**2.0/2.0
            SS=SS + H*FUN(X)*DXDZ
         END IF
      END DO
      PRINT*,'-----------------------------------------'
      PRINT *,'RESULTADO FINAL  I= ',SS
      PRINT*,'-----------------------------------------'
      PRINT*
      PRINT*
      PRINT*,'OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
      READ *,K
      IF(K.EQ.1) GOTO 1
      PRINT*
      END
C*******************
      FUNCTION FUN(X)   ! Evalúa la función a integrar
      DOBLE PRECISION X, FUN
      FUN= DSQRT(1.0+1/X)
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F4-5   INTEGRACION DE UNA FUNCION SINGULAR MEDIANTE
           LA TRANSFORMACION EXPONENCIAL DOBLE

¿NUMERO TOTAL DE INTERVALOS, N?
10
¿LIMITE INFERIOR DE INTEGRACION, A?
0
¿LIMITE SUPERIOR DE INTEGRACION, B?
1
-----------------------------------------
RESULTADO FINAL  I=     2.297836480119168
-----------------------------------------
OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
```

PROGRAMA 4-6 Integración doble

A) Explicaciones

El PROGRAMA 4-6 calcula una integral doble por medio de la regla de Simpson. El usuario debe definir tres funciones en el programa antes de ejecutarlo: el integrando en el subprograma FUNC3, la curva límite inferior en FUNC1 y la curva límite superior en FUNC2. Al ejecutarse, el programa pregunta el número de intervalos que se utilizarán en las direcciones x y y. También pregunta por A y B.

B) Variables

A: límite inferior de integración de x
B: límite superior de integración de x
C: curva límite inferior
D: curva límite superior
F: integrando
HX, HY: intervalos de la retícula en las direcciones x y y, respectivamente. (HY depende del valor de x)
W: pesos de integración en la regla de Simpson
S: resultado de la integral en la dirección de y
T: resultado total de la integral

C) Listado

```
C-----CSL/F4-6.FOR     INTEGRACION DOBLE MEDIANTE LA REGLA DE SIMPSON
      PRINT *
      PRINT *,'CSL/F4-6     INTEGRACION DOBLE MEDIANTE LA REGLA DE SIMPSON'
1     PRINT *
      PRINT *,'¿NUMERO DE INTERVALOS EN LA DIRECCION X?'
```

```
      READ *,M
      PRINT *, ¿NUMERO DE INTERVALOS EN LA DIRECCION Y?'
      READ *,N
      IF (M/2*2.NE.M .OR. N/2*2.NE.N) THEN
         PRINT *, EL NUMERO DE INTERVALOS DEBE SER PAR. REPITA.'
         GO TO 1
      END IF
      PRINT *, '¿LIMITE INFERIOR EN EL EJE X, A? '
      READ *,A
      PRINT *, ' ¿LIMITE SUPERIOR EN EL EJE B?'
      READ *,B
      PRINT *
      PRINT *,'  PUNTO EN        INTEGRAL EN LA              PESO'
      PRINT *,'   EL EJE X       DIRECCION DE Y
      PRINT *,'    I                 I~Y^                      W'
      HX=(B-A)/M
      T=0
      DO I=0, M
         X=A+I*HX
         CALL   FUNCT1(C,X)          ! Encuentra el límite inferior de los valores de y
         CALL   FUNCT2(D,X)          ! Encuentra el límite superior de los valores de y
         HY=(D-C)/N                  ! Tamaño del intervalo en la dirección de y
         S=0
         DO J=0,N
            Y=C+J*HY                 ! Valor en y de los puntos de la retícula
            CALL   FUNCT3(F,X,Y)     ! Encuentra el valor de f
            W=4
            IF (INT(J/2)*2.EQ.J) W=2
            IF((J.EQ.0).OR.(J.EQ.N)) W=1
            S=S+W*F                  ! Integración en la dirección de y
         END DO
         S=S*HY/3
         W=4
         IF (INT(I/2)*2 .EQ.I) W=2
         IF((I.EQ.0).OR.(I.EQ.M)) W=1
         PRINT 330,I,S,W
330      FORMAT(3X,I2,7X,1PE14.7,6X, 0PF10.4)
         T=T+W*S                     !-- Integración en la dirección de x
      END DO
      T=T*HX/3                       !-- Resultado de la integración
      PRINT *
      PRINT *,'-----------------------------------------------'
      PRINT *,' RESULTADO FINAL = ',T
      PRINT *,'-----------------------------------------------'
      PRINT *
      PRINT*
      PRINT*,'OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR '
      READ *,K
      IF(K.EQ.1) GOTO 1
      PRINT*
      END
C*****************************
      SUBROUTINE FUNCT1(C,X)     ! Curva límite inferior
200   C= LOG(X)
      RETURN
      END
C*****************************
      SUBROUTINE FUNCT2(D,X)     ! Curva límite superior
210   D=3+EXP(X/5)
      RETURN
      END
```

```
C*****************************
      SUBROUTINE FUNCT3(F,X,Y)  ! Evalúa la función a integrar
220   F=SIN(X+Y)
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F4-6     INTEGRACION DOBLE MEDIANTE LA REGLA DE SIMPSON

¿NUMERO DE INTERVALOS EN LA DIRECCION X?
6
¿NUMERO DE INTERVALOS EN LA DIRECCION Y?
6
¿LIMITE INFERIOR EN EL EJE X, A?
1
¿LIMITE SUPERIOR EN EL EJE X, B?
3

 PUNTO EN      INTEGRAL EN LA        PESO
 EL EJE X      DIRECCION DE Y
   I              I~Y^                 W
   0         5.3062823E-02          1.0000
   1        -8.5075313E-01          4.0000
   2        -1.5474440E+00          2.0000
   3        -1.8811973E+00          4.0000
   4        -1.8000411E+00          2.0000
   5        -1.3400741E+00          4.0000
   6        -6.0834056E-01          1.0000
-----------------------------------------------
RESULTADO FINAL I =   -2.615372
-----------------------------------------------
OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
```

PROBLEMAS

4.1) Evalúe las siguientes integrales utilizando la regla extendida del trapecio con intervalos de $N = 2$, 4 y 8 (también 16 y 32 si se utiliza un programa de computadora).

$3x^3 + 5x - 1$	$[0, 1]$
$x^3 - 2x^2 + x + 2$	$[0, 3]$
$x^4 + x^3 - x^2 + x + 3$	$[0, 1]$
$\tan(x)$	$[0, \pi/4]$
e^x	$[0, 1]$
$1/(2 + x)$	$[0, 1]$

4.2) Calcule la integral

$$I = \int_0^{\pi/2} \operatorname{sen}(x)\,dx$$

mediante la regla extendida del trapecio con $N = 2, 4, 8, 25$ y 100 intervalos. Evalúe después el error de los resultados numéricos comparándolos con sus valores exactos.

4.3) A continuación se da una tabla de valores

x	$f(x)$
0.0	0
0.1	2.1220
0.2	3.0244
0.3	3.2568
0.4	3.1399
0.5	2.8579
0.6	2.5140
0.7	2.1639
0.8	1.8358

Evalúe la integral

$$\int_0^{0.8} f(x)\,dx$$

por la regla extendida del trapecio con $h = 0.4$, $h = 0.2$ y $h = 0.1$.

4.4) Aplique la integración de Romberg a los resultados de la regla del trapecio con $h = 0.1$ y $h = 0.2$ del problema 4.3 para hacer una mejor estimación de la integral.

4.5) La siguiente es una tabla de valores:

i	x_i	$f(x_i)$
1	0	0.9162
2	0.25	0.8109
3	0.5	0.6931
4	0.75	0.5596
5	1.0	0.4055

a) Calcule

$$I = \int_0^1 f(x)\,dx$$

utilizando la regla extendida del trapecio con $h = 0.25$ y $h = 0.5$.

b) Aplique la integración de Romberg a los resultados del inciso a) para estimar un valor más preciso de I.

4.6) Repita el problema del ejemplo 4.1 utilizando la regla extendida de 1/3 de Simpson con $N = 2, 4, 8, 26, 50$ y 100.

4.7) Repita el problema 4.1 utilizando la regla de Simpson con $N = 4, 8$ y 16.

4.8) Obtenga la regla de 1/3 de Simpson integrando el polinomio de interpolación de Newton hacia adelante ajustado en x_0, $x_0 + h$ y $x_0 + 2h$.

4.9) Demuestre que la integración de Romberg basada en I_{2h} e I_h de la regla extendida del trapecio es idéntica al resultado de la regla de Simpson, utilizando h como el tamaño del intervalo.

4.10) Evalúe la integral de las siguientes funciones en el intervalo indicado, utilizando la regla de Simpson con $N = 2, 4, 8, 16$ y 32:

a) $y = \dfrac{1}{2 + \cos(x)}$ $\qquad [0, \pi]$

b) $y = \dfrac{\log(1+x)}{x}$ $\quad [1, 2]$

c) $y = \dfrac{1}{1 + \operatorname{sen}^2(x)}$ $\quad [0, \pi/2]$

4.11) Evalúe la integral de las siguientes funciones en el intervalo indicado, utilizando la regla de Simpson con $N = 2, 4, 8, 16$ y 32.

a) $y = x \exp(2x)$ $\quad [0, 1]$

b) $y = x^{-x}$ $\quad [0, 1]$

c) $y = \exp(2x) \operatorname{sen}^2(x)$ $\quad [0, 2\pi]$

4.12) Repita el problema del ejemplo 4.1, utilizando la regla extendida de Simpson con $N = 3, 7$ y 11 intervalos.

4.13) Suponga que usted es un arquitecto y planea utilizar un gran arco de forma parabólica dado por

$$y = 0.1x(30 - x) \text{ metros}$$

donde y es la altura desde el piso y x está en metros. Calcule la longitud total del arco utilizando la regla extendida de Simpson. (Divida el dominio desde $x = 0$ hasta $x = 30$ metros en 10 intervalos de la misma longitud). La longitud total del arco está dada por

$$L = \int_0^{30} \sqrt{1 + (dy/dx)^2}\, dx$$

4.14) Un automóvil con masa $M = 5400$ kg se mueve a una velocidad de 30 m/seg. El motor se apaga súbitamente a los $t = 0$ seg. Suponga que la ecuación de movimiento después de $t = 0$ está dada por

$$5400v \frac{dv}{dx} = -8.276v^2 - 2000$$

donde $v = v(t)$ es la velocidad (m/seg) del automóvil al tiempo t. El lado izquierdo representa $Mv(dv/dx)$. El primer término del lado derecho es la fuerza aerodinámica y el segundo término es la resistencia de las llantas al rodaje. Calcule la distancia que recorre el auto hasta que la velocidad se reduce a 15 m/seg. (*Sugerencia*: la ecuación de movimiento se puede integrar como

$$\int_{15}^{30} \frac{5400v\, dv}{8.276v^2 + 2000} = \int dx = x$$

Evalúe la ecuación anterior mediante la regla de Simpson.)

4.15) Si $f(x)$ es un polinomio de orden menor o igual que N, la fórmula cerrada de Newton-Cotes de orden N (que utilizan N + 1 puntos) es exacta. Explique la razón.

4.16) La longitud de una curva definida por $x = \theta(t)$, $y = \psi(t)$, $a < t < b$, está dada por

$$s = \int_a^b ([\theta'(t)]^2 + [\psi(t)]^2)^{1/2}\, dt$$

Use las cuadraturas de Gauss con $N = 2, 4$ y 6 para encontrar la longitud de la cicloide definida por

$$x = 3[t - \operatorname{sen}(t)], \quad y = 2 - 2\cos(t), \quad 0 < t < 2\pi$$

4.17) Si $f(x)$ es un polinomio de orden menor o igual que $2N-1$, la cuadratura de Gauss con N puntos de Legendre es exacta. Explique por qué.

4.18) Evalúe la siguiente integral mediante la cuadratura de Gauss de $N = 4$ y $N = 6$.

$$I = \int_0^1 \frac{\ln(1+x)}{x}\,dx$$

4.19) Evalúe la siguiente integral impropia en forma tan exacta como sea posible, utilizando la regla extendida del trapecio.

$$\int_{-\infty}^{\infty} \frac{\exp(-x^2)}{(1+x^2)}\,dx$$

4.20) Evalúe las siguientes integrales impropias, lo más exacto que sea posible, utilizando la regla extendida del trapecio con la transformación exponencial doble.

a) $\displaystyle\int_0^1 \frac{\tan(x)}{x^{0.7}}\,dx$

b) $\displaystyle\int_0^1 \frac{\exp(x)}{\sqrt{1-x^2}}\,dx$

4.21) Repita el problema 4.18 con el PROGRAMA 4-5.

4.22) Calcule las siguientes integrales con la cuadratura de Gauss de $N = 6$:

a) $\displaystyle\int_0^{\pi} \frac{1}{2+\cos(x)}\,dx$

b) $\displaystyle\int_1^2 \frac{\ln(1+x)}{x}\,dx$

c) $\displaystyle\int_0^1 x\exp(2x)\,dx$

d) $\displaystyle\int_0^1 x^{-x}\,dx$

4.23) Calcule la siguiente integral utilizando la regla extendida del trapecio en cada dirección:

$$\int_1^2 dx \int_0^1 dy\,\mathrm{sen}(x+y)$$

(Use sólo dos intervalos en cada dirección; la función seno está en radianes.)

4.24) Evalúe la siguiente integral mediante la regla de 1/3 de Simpson:

$$I = \int_0^1 \int_0^x \sqrt{x+y}\,dy\,dx$$

4.25) El área de un círculo unitario es π. La precisión de un esquema numérico para la integración doble se puede examinar mediante el siguiente problema:

$$I = \iint_D dy\,dx$$

donde D significa que la integral se toma en el interior de

$$x^2 + y^2 \leqslant 2x$$

el cual es un círculo unitario. Realice la evaluación numérica de la doble integral anterior mediante la regla extendida de Simpson en ambas direcciones, con 2×2, 4×4, 8×8, 16×16, 32×32 y 64×64 intervalos.

4.26) Utilice la regla extendida de Simpson con 10 intervalos en cada dirección para evaluar la integral doble

$$I = \int_0^{\pi} \int_0^{\operatorname{sen}(x)} \exp(-x^2 - y^2)\, dy\, dx$$

4.27) Evalúe la siguiente integral doble mediante la regla de 1/3 de Simpson:

$$I = \int_1^2 \int_0^{2-0.5x} \sqrt{x + y}\, dy\, dx$$

4.28) Repita el problema del ejemplo 4.8 utilizando la cuadratura de Gauss con $N = 3$.

BIBLIOGRAFIA

Abramowitz, M., y I. A. Stegun, editores, *Handbook of Mathematical Functions*, National Bureau of Standards, 1970.

Carnahan, B., H. A. Luther y J. O. Wilkes, *Applied Numerical Methods,* Wiley, 1969.

Ferziger, J. H., *Numerical Methods for Engineering Application*, Wiley-Interscience, 1981.

Froeberg, C. E., *Numerical Mathematics - Theory and Computer Applications*, Benjamin/Cummings, 1985.

Gerald, C. F. y P. O. Wheatley, *Applied Numerical Analysis*, 4a. edición, Addison-Wesley, 1989.

James, M. L., G. M. Smith y J. C. Wolford, *Applied Numerical Methods for Digital Computation*, 3a. edición, Harper & Row, 1985.

King, J. T., *Introduction to Numerical Computation*, McGraw-Hill, 1984.

Mori, M., Quadrature Formulas Obtained by Variable Transformation and DE-rule, *J. Comp. Appl. Math.*, Vol. 12-13, pág. 119-130, 1980.

Mori, M. y R. Piessens, editores, *Numerical Quadrature*, North-Holland, 1987.

Press, W. P., B. P. Flannery, S. A. Teukolsky y W. T. Vetterling, *Numerical Recipes*, Cambridge University Press, 1986.

Stoer, J. y R. Burlish, *Introduction to Numerical Analysis*, Springer-Verlag, 1980.

Takahashi, H. y M. Mori. "Double Exponential Formulas for Numerical Integration", *Publ. RIMS*, Kyoto University, Vol. 9, Núm. 3, 1974.

5
Diferenciación numérica

5.1 INTRODUCCION

La diferenciación numérica, o aproximación por diferencias, se utiliza para evaluar las derivadas de una función por medio de sus valores dados en los puntos de una retícula. Las aproximaciones por diferencias son importantes en la solución de ecuaciones diferenciales ordinarias y parciales.

Para ilustrar la diferenciación numérica, consideremos una función $f(x)$ como la que se muestra en la figura 5.1. Supongamos que se desea evaluar la primera derivada de $f(x)$ en $x = x_0$. Si se conocen los valores de f en $x_0 - h$, x_0 y $x_0 + h$, donde h es el tamaño del intervalo entre dos puntos consecutivos en el eje x, entonces se puede aproximar $f'(x_0)$ mediante el gradiente de la interpolación lineal A, B o C mostradas en la figura 5.1. Estas tres aproximaciones se llaman respectivamente las aproximaciones por diferencias *hacia adelante*, *hacia atrás y central*. Sus fórmulas matemáticas son como sigue:

a) Aproximación que utiliza A (aproximación por diferencias hacia adelante)

$$f'(x_0) \simeq \frac{f(x_0 + h) - f(x_0)}{h} \tag{5.1.1}$$

b) Aproximación que utiliza B (aproximación por diferencias hacia atrás)

$$f'(x_0) \simeq \frac{f(x_0) - f(x_0 - h)}{h} \tag{5.1.2}$$

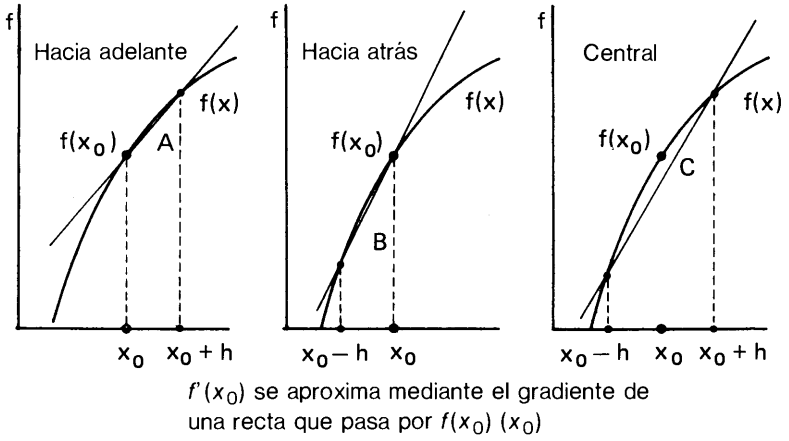


Figura 5.1 Explicación gráfica de las aproximaciones por diferencias de $f'(x_0)$.

c) Aproximación que utiliza C (aproximación por diferencias central)

$$f'(x_0) \simeq \frac{f(x_0 + h) - f(x_0 - h)}{2h} \tag{5.1.3}$$

Existen tres tipos de enfoques para obtener aproximaciones por diferencias. El primero se basa en el desarrollo de Taylor de la función alrededor de un punto de la retícula, el segundo utiliza los operadores de diferencia y el tercero deriva los polinomios de interpolación. En la tabla 5.1 se resumen las ventajas y desventajas de cada enfoque. El PROGRAMA 5-1 genera las fórmulas de aproximación por diferencias (véase la sección 5.3 para más detalles).

5.2 USO DEL DESARROLLO DE TAYLOR

Cuando una función se representa numéricamente en puntos discretos, ésta se aproxima mediante la interpolación. Así, la integración numérica de la función se calcula

Tabla 5.1 Breve resumen de los tres métodos para obtener fórmulas de diferenciación numérica

Método de obtención	Ventajas	Desventajas
Desarrollo de Taylor	Los términos del error se obtienen en forma explícita. Se puede aplicar a retículas no uniformes.	Sólo se puede obtener una fórmula a la vez.
Operador de diferencias	Bastante similaridad entre las derivadas y las aproximaciones por diferencias.	Necesita el desarrollo de Taylor para analizar el error.
Derivación de polinomios de interpolación	Se pueden obtener, en forma sistemática, muchas fórmulas de aproximación por diferencias.	Difícil de aplicar en retículas no uniformes.

integrando la fórmula de interpolación, como se explicó en el capítulo 2. De la misma forma, se pueden obtener fórmulas de diferenciación numérica al diferenciar las fórmulas de interpolación.

Comenzaremos obteniendo fórmulas mediante el desarrollo de Taylor, ya que es equivalente a la diferenciación de una interpolación y conduce exactamente a los mismos resultados. En esta sección se explica, de manera adecuada, la obtención de la aproximación por diferencias utilizando el desarrollo de Taylor. A continuación, en la siguiente sección se explica un punto de vista más genérico. Entre la bibliografía para el uso de los desarrollos de Taylor están [Hornbeck y James, Smith y Wolford].

Para una derivada de orden p, el mínimo número de datos necesario para obtener una aproximación por diferencias es $p + 1$. Por ejemplo, una aproximación por diferencias para la primera derivada de una función necesita al menos dos puntos.

Empecemos a deducir la aproximación por diferencias para $f_i' = f'(x_i)$ utilizando $f_i = f(x_i)$ y $f_{i+1} = f(x_{i+1})$. Los valores de f en todos los puntos distintos de x_i se desarrollan en una serie de Taylor. El desarrollo de Taylor de f_{i+1} alrededor de x_i es

$$f_{i+1} = f_i + hf_i' + \frac{h^2}{2} f_i'' + \frac{h^3}{6} f_i''' + \frac{h^4}{24} f_i'''' + \cdots \tag{5.2.1}$$

Al despejar f'_i en la ecuación (5.2.1) se obtiene

$$f_i' = \frac{f_{i+1} - f_i}{h} - \frac{1}{2} hf_i'' - \frac{1}{6} h^2 f_i''' - \cdots \tag{5.2.2}$$

Si truncamos después del primer término, la ecuación (5.2.2) es la aproximación por diferencias hacia adelante que ya se conocía como la ecuación (5.1.1). Los términos truncados conforman el error de truncamiento. Este se puede representar por medio del coeficiente principal $(1 - (h/2)f_i''$ en este caso) debido a que los demás términos se anulan más rápido que éste cuando h decrece. La aproximación por diferencias hacia adelante se expresa, incluyendo el efecto del error por truncamiento como sigue:

$$f_i' = \frac{f_{i+1} - f_i}{h} + 0(h) \tag{5.2.3}$$

donde

$$0(h) = -\frac{1}{2} hf_i''$$

El término $O(h)$ indica que el error es aproximadamente proporcional al intervalo h de la retícula. El error también es proporcional a la segunda derivada f''.

La aproximación por diferencias hacia atrás de la primera derivada, utilizando f_{i-1} y $_i$ se obtiene de manera similar. El desarrollo de Taylor de f_{i-1} es

$$f_{i-1} = f_i - hf_i' + \frac{h^2}{2} f_i'' - \frac{h^3}{6} f_i''' + \frac{h^4}{24} f_i'''' - \cdots \tag{5.2.4}$$

Al despejar f'_i, se obtiene la aproximación por diferencias hacia atrás como

$$f'_i = \frac{f_i - f_{i-1}}{h} + 0(h) \tag{5.2.5}$$

donde

$$0(h) = \frac{1}{2}hf''_i$$

La aproximación por diferencias centrales utilizando f_{i+1} y f_{i-1} se puede obtener mediante los desarrollos de Taylor de f_{i+1} y f_{i-1} ya dados en las ecuaciones (5.2.1) y (5.2.4), respectivamente. Si restamos la ecuación (5.2.4) de la (5.2.1), obtenemos

$$f_{i+1} - f_{i-1} = 2hf'_i + \frac{1}{3}h^3f'''_i + \cdots \tag{5.2.6}$$

donde el término f''_i se elimina en forma automática. Al despejar de aquí f_i tenemos

$$f'_i = \frac{f_{i+1} - f_{i-1}}{2h} - \frac{1}{6}h^2f'''_i + \cdots \tag{5.2.7}$$

La aproximación por diferencias centrales se expresa como

$$f'_i = \frac{f_{i+1} - f_{i-1}}{2h} + 0(h^2) \tag{5.2.8}$$

donde

$$0(h^2) = -\frac{1}{6}h^2f'''_i$$

Es importante observar que, debido a la cancelación del término f'', el error de la aproximación por diferencias centrales es proporcional a h^2 en vez de h. Al decrecer h, el error decrece más rápido que en las otras dos aproximaciones.

Como se explicó antes, una aproximación por diferencias de $f^{(p)}_i$ necesita al menos $p + 1$ puntos. Si se utilizan más datos, se puede obtener una aproximación por diferencias más exacta. Con puntos dados, una ecuación por diferencias con la máxima exactitud es tal que el término del error es el del máximo orden posible.

Para ilustrar el significado de esto, obtendremos una aproximación por diferencias de f_i utilizando f_{i+1} y f_{i+2}. Puesto que el número mínimo de datos necesarios para f' es dos, tenemos un dato más que el mínimo. Los desarrollos de f_{i+1} y

f_{i+2} se escriben como

$$f_{i+1} = f_i + hf'_i + \frac{h^2}{2} f''_i + \frac{h^3}{6} f'''_i + \frac{h^4}{24} f''''_i + \cdots \tag{5.2.9}$$

$$f_{i+2} = f_i + 2hf'_i + 4\frac{h^2}{2} f''_i + 8\frac{h^3}{6} f'''_i + 16\frac{h^4}{24} f''''_i + \cdots \tag{5.2.10}$$

con estas dos ecuaciones es posible eliminar los términos de la segunda derivada. Por esto, el término principal de los errores de truncamiento es el de la derivada de tercer orden. Al restar la ecuación (5.2.11) de cuatro veces la ecuación (5.2.9) obtendremos

$$4f_{i+1} - f_{i+2} = 3f_i + 2hf'_i - \frac{2}{3}h^3 f'''_i + \cdots \tag{5.2.11}$$

Al despejar f'_i de la ecuación (5.2.11) se tiene que

$$f'_i = \frac{-f_{i+2} + 4f_{i+1} - 3f_i}{2h} + 0(h^2) \tag{5.2.12}$$

donde el término del error está dado por

$$0(h^2) = \frac{1}{3}h^2 f'''_i$$

La ecuación (5.2.12) se llama *aproximación por diferencias hacia adelante con tres puntos para* f'_i y el error es del mismo orden que el de la aproximación por diferencias centrales.

Análogamente, la *aproximación por diferencias hacia atrás con tres puntos* se puede obtener utilizando f_i, f_{i-1}, y f_{i-2} como sigue:

$$f'_i = \frac{3f_i - 4f_{i-1} + f_{i-2}}{2h} + 0(h^2) \tag{5.2.13}$$

donde

$$0(h^2) = \frac{1}{3}h^2 f'''_i$$

Ejemplo 5.1

Calcule la primera derivada de tan (x) en $x = 1$ mediante las cinco aproximaciones por diferencias obtenidas en esta sección, utilizando $h = 0.1$, 0.05 y 0.02. Evalúe después el porcentaje de error de cada aproximación comparándolo con el valor exacto.

(Solución)

Sustituimos $f_i = f(1 + ih) = \tan(1 + ih)$ en las ecuaciones (5.2.5), (5.2.3), (5.2.8), (5.2.13) y (5.2.12) y obtenemos los siguientes resultados:

	h=0.1	h=0.05	h=0.02
[tan(1)-tan(1-h)]/h	2.9724	3.1805	3.3224
	(13.2)[a]	(7.1)	(3.0)
[tan(1+h)-tan(1)]/h	4.0735	3.7181	3.5361
	(-18.9)	(-8.5)	(-3.2)
[tan(1+h)-tan(1-h)]/2h	3.5230	3.4493	3.4293
	(-2.8)	(-0.69)	(-0.11)
[3 tan(1)-4 tan(1-h)+tan(1-2h)]/2h	3.3061	3.3885	3.4186
	(3.5)	(1.08)	(0.20)
[-tan(1+2h)+4 tan(1+h)-3 tan(1)]/2h	3.0733	3.3627	3.4170
	(10.3)	(1.83)	(0.25)

[a] Porcentaje del error

Conviene observar que los errores de las dos primeras aproximaciones decrecen en proporción con h, mientras que los errores de las últimas tres aproximaciones decrecen en proporción con h^2. Es claro que la razón de reducción del error se vuelve más rápida cuando el orden de precisión es mayor.

A continuación obtendremos aproximaciones por diferencias para la segunda derivada. El principio básico para obtener una aproximación por diferencias de segundo orden es eliminar la primera derivada de los desarrollos de Taylor y, de ser posible, tantos términos de orden superior a 2 como sea posible.

Por ejemplo, obtenemos una aproximación por diferencias de f_i'' utilizando f_{i+1}, f_i y f_{i-1}. Los desarrollos de Taylor de f_{i+1} y f_{i-1} están dados por las ecuaciones (5.2.1) y (5.2.4). Al sumar los dos desarrollos, obtenemos

$$f_{i+1} + f_{i-1} = 2f_i + h^2 f_i'' + \frac{1}{12}h^4 f_i'''' + \cdots$$

Restando $2f_i$ de ambos lados tenemos

$$f_{i+1} - 2f_i + f_{i-1} = h^2 f_i'' + \frac{1}{12}h^4 f_i'''' + \cdots$$

Entonces, truncamos después del término f'' y reescribimos para obtener

$$f_i'' = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + 0(h^2) \tag{5.2.14}$$

La ecuación (5.2.14) se llama *aproximación por diferencias centrales de f''* y el error se representa como

$$0(h^2) = -\frac{1}{12}h^2 f_i''''$$

Se puede obtener otra aproximación por diferencias de f_i'' utilizando f_i, f_{i-1} y f_{i-2} (puesto que $p = 2$, el número mínimo de datos necesarios es 3). Al restar dos veces el desarrollo de Taylor de f_{i-1} del de f_{i-2} se tiene

$$f_{i-2} - 2f_{i-1} = -f_i + h^2 f_i'' - h^3 f_i''' + \cdots$$

Al despejar f_i'' de la ecuación anterior se tiene

$$f_i'' = \frac{f_{i-2} - 2f_{i-1} + f_i}{h^2} + 0(h) \qquad 0(h) = hf_i''' \tag{5.2.15}$$

La ecuación (5.2.15) se llama *aproximación por diferencias hacia atrás de f_i''.*

Las aproximaciones por diferencias para las derivadas de orden superior se pueden obtener mediante combinaciones lineales adecuadas de los desarrollos de Taylor. Las derivadas se vuelven cada vez más complicadas al aumentar el número de puntos o el orden de las derivadas. De hecho, las aproximaciones por diferencias que aparecen en los libros tienen errores frecuentes, particularmente en los términos del error para las aproximaciones por diferencias de orden superior. Por esta razón, en la siguiente sección se describen un algoritmo más sistemático y un programa de computadora basado en el (PROGRAMA 5-1). Por medio de este programa, la verificación de las aproximaciones por diferencias es muy sencilla.

Hasta este momento, hemos supuesto que los puntos de la retícula tienen separación uniforme. Sin embargo, las aproximaciones por diferencias en retículas con separación no uniforme se pueden obtener mediante los desarrollos de Taylor.

Las aproximaciones por diferencias que se utilizan con frecuencia se enumeran en la tabla 5.2.

Tabla 5.2 Aproximaciones por diferencias[a]

Primera derivada

a) Aproximaciones por diferencias hacia adelante:

$$f_i' = \frac{f_{i+1} - f_i}{h} + 0(h), \qquad 0(h) = -\frac{1}{2}hf_i''$$

$$f_i' = \frac{-f_{i+2} + 4f_{i+1} - 3f_i}{2h} + 0(h^2), \qquad 0(h^2) = \frac{1}{3}h^2 f_i'''$$

$$f_i' = \frac{2f_{i+3} - 9f_{i+2} + 18f_{i+1} - 11f_i}{6h} + 0(h^3), \qquad 0(h^3) = -\frac{1}{4}h^3 f_i''''$$

(Continúa)

Tabla 5.2 (Continúa)

b) Aproximaciones por diferencias hacia atrás:

$$f'_i = \frac{f_i - f_{i-1}}{h} + O(h), \qquad O(h) = \frac{1}{2}hf''_i$$

$$f'_i = \frac{3f_i - 4f_{i-1} + f_{i-2}}{2h} + O(h^2), \qquad O(h^2) = \frac{1}{3}h^2f'''_i$$

$$f'_i = \frac{11f_i - 18f_{i-1} + 9f_{i-2} - 2f_{i-3}}{6h} + O(h^3), \quad O(h^3) = \frac{1}{4}h^3f''''_i$$

c) Aproximación por diferencias centrales:

$$f'_i = \frac{f_{i+1} - f_{i-1}}{2h} + O(h^2), \qquad O(h^2) = -\frac{1}{6}h^2f'''_i$$

$$f'_i = \frac{-f_{i+2} + 8f_{i+1} - 8f_{i-1} + f_{i-2}}{12h} + O(h^4), \quad O(h^4) = \frac{1}{30}h^4f^{(v)}_i$$

Segunda derivada

d) Aproximaciones por diferencias hacia adelante:

$$f''_i = \frac{f_{i+2} - 2f_{i+1} + f_i}{h^2} + O(h), \qquad O(h) = -hf'''_i$$

$$f''_i = \frac{-f_{i+3} + 4f_{i+2} - 5f_{i+1} + 2f_i}{h^2} + O(h^2), \quad O(h^2) = \frac{11}{12}h^2f''''_i$$

e) Aproximaciones por diferencias hacia atrás:

$$f''_i = \frac{f_i - 2f_{i-1} + f_{i-2}}{h^2} + O(h), \qquad O(h) = hf'''_i$$

$$f''_i = \frac{2f_i - 5f_{i-1} + 4f_{i-2} - f_{i-3}}{h^2} + O(h^2), \quad O(h^2) = \frac{11}{12}h^2f''''_i$$

f) Aproximaciones por diferencias centrales:

$$f''_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + O(h^2), \qquad O(h^2) = -\frac{1}{12}h^2f''''_i$$

$$f''_i = \frac{-f_{i+2} + 16f_{i+1} - 30f_i + 16f_{i-1} - f_{i-2}}{h^2} + O(h^4), \quad O(h^4) = \frac{1}{90}h^4f^{(vi)}_i$$

Tercera derivada

g) Aproximaciones por diferencia hacia adelante.

$$f'''_i = \frac{f_{i+3} - 3f_{i+2} + 3f_{i+1} - f_i}{h^3} + O(h), \quad O(h) = -\frac{3}{2}hf''''_i$$

h) Aproximaciones por diferencias hacia atrás:

$$f'''_i = \frac{f_i - 3f_{i-1} + 3f_{i-2} - f_{i-3}}{h^3} + O(h), \quad O(h) = \frac{3}{2}hf''''_i$$

i) Aproximación por diferencias centrales:

$$f'''_i = \frac{f_{i+2} - 2f_{i+1} + 2f_{i-1} - f_{i-2}}{2h^3} + O(h^2), \quad O(h^2) = -\frac{1}{4}h^2f^{(v)}_i$$

[a] Las aproximaciones por diferencias que aparecen en esta tabla se generaron mediante el PROGRAMA 5-1.

Resumen de Esta Sección

a) Una aproximación por diferencias para $f_i^{(p)}$ requiere de al menos $p + 1$ puntos.
b) La aproximación por diferencias se obtiene desarrollando f_j en una serie de Taylor alrededor de x_i.
c) Las derivadas de orden menor que p deben eliminarse. Esto es posible con un mínimo de $p + 1$ puntos.
d) El término del error es el término truncado de menor orden.

5.3 ALGORITMO GENERICO PARA OBTENER UNA APROXIMACION POR DIFERENCIAS

El objetivo de esta sección es describir un algoritmo genérico para obtener una aproximación por diferencias de una derivada de orden dado, utilizando un conjunto específico de puntos en una retícula. El algoritmo se implanta entonces como PROGRAMA 5-1.

Supongamos que el número total de puntos en la retícula es L y que los puntos de la retícula están numerados como $i = \alpha, \beta, \ldots, \lambda$. Supongamos que $L \geqslant p + 1$, donde p es el orden de la derivada por aproximar. Las abscisas de los puntos de la retícula son $x_i = \alpha h, \beta h, \ldots, \lambda h$ con $i = \alpha, \beta, \ldots, \lambda$.

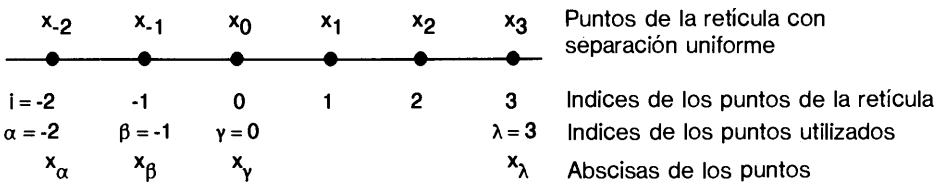


Figura 5.2 Ilustración de los puntos de la retícula utilizados para la aproximación por diferencias

La aproximación por diferencias de la p-ésima derivada de $f(x)$, utilizando estos puntos de la retícula, se puede escribir en la forma:

$$f_0^{(p)} = \frac{a_\alpha f_\alpha + a_\beta f_\beta + \cdots + a_\lambda f_\lambda}{h^p} + E \tag{5.3.1}$$

donde a_α hasta a_λ son L coeficientes indeterminados; $f_\alpha = f(x_\alpha)$, $f_\beta = f(x_\beta), \ldots$ son las coordenadas que se usarán y E es el error, que se escribe como

$$E = c_1 h^{L-p} f^{(L)} + c_2 h^{L-p+1} f^{(L+1)} \tag{5.3.2}$$

La esencia del algoritmo es sustituir los desarrollos de Taylor de f_i en la ecuación (5.3.1) y calcular los coeficientes indeterminados de forma que el término del error se minimice o, en forma equivalente, que el orden de E sea el máximo orden posible.

Para simplificar la explicación posterior, supongamos que $p = 1$, $L = 3$, $\alpha = 0$, $\beta = 1$ y $\gamma = 2$. Entonces, la ecuación (5.3.1) se escribe como

$$f'_0 = \frac{a_0 f_0 + a_1 f_1 + a_2 f_2}{h} + E \tag{5.3.3}$$

donde a_0, a_1 y a_2 son tres coeficientes indeterminados y $x_0 = 0$, $x_1 = h$ y $x_2 = 2h$ son los puntos de la retícula que se utilizarán. Sustituimos los desarrollos de Taylor de f_1 y f_2 alrededor de $x = 0$ en la ecuación (5.3.3) para obtener

$$f'_0 = \frac{a_0 f_0}{h} + \frac{a_1}{h}\left[f_0 + hf'_0 + \frac{h^2}{2} f''_0 + \frac{h^3}{6} f'''_0 + \cdots\right]$$
$$+ \frac{a_2}{h}\left[f_0 + 2hf'_0 + \frac{4h^2}{2} f''_0 + \frac{8h^3}{6} f'''_0 + \cdots\right] + E$$

o, reagrupando términos,

$$f'_0 = f_0[a_0 + a_1 + a_2]\frac{1}{h} + f'_0[0 + a_1 + 2a_2] + f''_0[0 + a_1 + 4a_2]\frac{h}{2}$$
$$+ f'''_0[0 + a_1 + 8a_2]\frac{h^2}{6} + f''''_0[0 + a_1 + 16a_2]\frac{h^3}{24} + \cdots + E \tag{5.3.4}$$

La ecuación (5.3.4) tiene tres coeficientes indeterminados, los cuales se pueden definir mediante tres condiciones. Para minimizar el error de la ecuación (5.3.4), hacemos los coeficientes de f_0, f'_0 y f''_0 iguales a 0, 1 y 0 respectivamente:

$$\begin{aligned} a_0 + a_1 + a_2 &= 0 \\ 0 + a_1 + 2a_2 &= 1 \\ 0 + a_1 + 4a_2 &= 0 \end{aligned} \tag{5.3.5}$$

Al resolver las ecuaciones anteriores, vemos que los valores de los tres coeficientes indeterminados son $a_0 = -\frac{3}{2}$, $a_1 = 2$ y $a_2 = -\frac{1}{2}$.

Los términos no nulos de orden superior de la ecuación (5.3.4) constituyen el error; a saber,

$$E = -f'''_0[0 + a_1 + 8a_2]\frac{h^2}{6} - f''''_0[0 + a_1 + 16a_2]\frac{h^3}{24} + \cdots \tag{5.3.6}$$

Al comparar la ecuación (5.3.6) con la ecuación (5.3.2), tenemos que c_1 y c_2 de la ecuación (5.3.2) están dados por

$$c_1 = -\frac{1}{6}(a_1 + 8a_2)$$

$$c_2 = -\frac{1}{24}(a_1 + 16a_2)$$

de lo que se obtiene, sustituyendo $a_1 = 2$ y $a_2 = -\frac{1}{2}$,

$$c_1 = -\frac{1}{6}\left(2 - \frac{8}{2}\right) = \frac{1}{3}$$

$$c_2 = -\frac{1}{24}\left(2 - \frac{16}{2}\right) = \frac{1}{4}$$

Puesto que el primer término de la ecuación (5.3.2) no es nulo, ignoramos el segundo término y escribimos el término del error como

$$E = \frac{1}{3}h^2 f_0''' \tag{5.3.7}$$

Si, por otro lado, el primer término de la ecuación (5.3.6) fuera igual a cero, el segundo término representaría el error.

El resultado final de estos cálculos es

$$f_0' = \frac{1}{h}\left[-\frac{3}{2}f_0 + 2f_1 - \frac{1}{2}f_2\right] + E$$

o, en forma equivalente

$$f_0' = \frac{-3f_0 + 4f_1 - f_2}{2h} + E \tag{5.3.8}$$

donde

$$E = \frac{1}{3}h^2 f_0'''$$

En términos más generales, con L datos, podemos definir L coeficientes indeterminados de la ecuación (5.3.1) fijando correctamente los primeros L términos del desarrollo de Taylor de la ecuación (5.3.1). Así, el término del error es proporcional al $(L + 1)$-ésimo término o, en forma equivalente, a la L-ésima derivada si su coeficiente no se anula. Si es igual a cero, el término del error es del orden inmediato superior.

Este algoritmo funciona incluso cuando los índices α, β, ... no son enteros. Esto quiere decir que la aproximación por diferencias en una retícula con separación no uniforme se puede obtener mediante el mismo algoritmo.

Como se describió antes, mediante el PROGRAMA 5-1 se puede llevar a cabo este algoritmo.

Resumen de Esta Sección. El algoritmo genérico descrito en esta sección es esencialmente el mismo algoritmo analizado en la sección anterior. Sin embargo, su formulación general hace posible desarrollar un programa de computadora.

5.4 USO DE LOS OPERADORES DE DIFERENCIAS

Definimos a continuación tres operadores de diferencias [Ralston; Isaacson/Keller]:

a) Operador de diferencias hacia adelante: Δ

$$\Delta f_i = f_{i+1} - f_i \tag{5.4.1}$$

b) Operador de diferencias hacia atrás: ∇

$$\nabla f_i = f_i - f_{i-1} \tag{5.4.2}$$

c) Operador diferencial central: δ

$$\delta f_i = f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}$$

o bien

$$\delta f_{i+\frac{1}{2}} = f_{i+1} - f_i \tag{5.4.3}$$

donde

$$f_{i+\frac{1}{2}} = f\left(x_i + \frac{h}{2}\right).$$

Los operadores de diferencias de orden superior se pueden escribir como potencias de los operadores de diferencias anteriores: por ejemplo, Δ^n, ∇^n y δ^n son operadores de diferencias de orden n. Se pueden obtener otros operadores de diferencias de orden n al aplicar ∇ y Δ en la forma $\nabla^{n-m}\Delta^m$ donde $1 \leqslant m \leqslant n$. En el caso $n = 2$, los operadores de diferencias dan como resultado

$$\Delta^2 f_i = \Delta(f_{i+1} - f_i) = f_{i+2} - 2f_{i+1} + f_i \tag{5.4.4a}$$

$$\nabla^2 f_i = \nabla(f_i - f_{i-1}) = f_i - 2f_{i-1} + f_{i-2} \tag{5.4.4b}$$

$$\delta^2 f_i = \delta(f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}) = f_{i+1} - 2f_i + f_{i-1} \tag{5.4.4c}$$

$$\Delta\nabla f_i = \Delta(f_i - f_{i-1}) = f_{i+1} - 2f_i + f_{i-1} \tag{5.4.4d}$$

$$\nabla\Delta f_i = \nabla(f_{i+1} - f_i) = f_{i+1} - 2f_i + f_{i-1} \tag{5.4.4e}$$

Conviene observar que en las ecuaciones anteriores las últimas tres diferencias son idénticas. De hecho, podemos escribir la relación de identidad:

$$\delta^2 = \Delta\nabla = \nabla\Delta \tag{5.4.5}$$

Si n es par y $m = n/2$, $\nabla^{n-m}\Delta^m$ es igual a $\nabla^m\Delta^m$, que es el n-ésimo operador de diferencias centrales.

Las aproximaciones por diferencias se obtienen al aproximar los operadores diferenciales mediante los operadores de diferencias. Por ejemplo, el operador diferencial ordinario de primer orden se puede aproximar de tres formas diferentes:

$$\begin{aligned} \frac{d}{dx} &\simeq \frac{\Delta}{\Delta x} \\ \frac{d}{dx} &\simeq \frac{\nabla}{\nabla x} \\ \frac{d}{dx} &\simeq \frac{\delta}{\delta x} \end{aligned} \tag{5.4.6}$$

Si aplicamos las aproximaciones de la ecuación (5.4.6) a una función $f(x)$, se obtienen las aproximaciones por diferencias de f_i'. Si utilizamos $\frac{\Delta}{\Delta x}x$ se obtiene la aproximación por diferencias hacia adelante:

$$\left[\frac{d}{dx}f(x)\right]_{x_i} \simeq \frac{\Delta}{\Delta x}f_i = \frac{f_{i+1} - f_i}{h} \tag{5.4.6}$$

donde Δx en el denominador se interpreta como $\Delta x_i = x_{i+1} - x_i = h$. Análogamente, si utilizamos $\frac{\nabla}{\nabla x}x$, obtenemos

$$\left[\frac{d}{dx}f(x)\right]_{x_i} \simeq \frac{\nabla}{\nabla x}f_i = \frac{f_i - f_{i-1}}{h} \tag{5.4.7}$$

La aproximación por diferencias centrales de f_i' se obtiene al aplicar $\frac{\delta}{\delta x}$ con base en la retícula con separación $2h$, es decir,

$$\left[\frac{d}{dx}f(x)\right]_{x_i} \simeq \frac{\delta}{\delta x}f_i = \frac{f_{i+1} - f_{i-1}}{2h} \tag{5.4.8}$$

donde δx en el denominador se interpreta como $\delta x_i = x_{i+1} - x_{i-1} = 2h$. La aproximación por diferencias centrales también se puede obtener tomando el promedio aritmético de las aproximaciones por diferencias hacia adelante y hacia atrás como

$$\left[\frac{d}{dx}f(x)\right]_{x_i} \simeq \frac{1}{2}\left[\frac{\Delta}{\Delta x} + \frac{\nabla}{\nabla x}\right]f_i = \frac{f_{i+1} - f_{i-1}}{2h} \tag{5.4.9}$$

Las aproximaciones para el operador diferencial de segundo orden se pueden escribir por medio de la doble aplicación de la aproximación de primer orden:

$$\begin{aligned}
\frac{d^2}{dx^2} &= \frac{\Delta^2}{\Delta x^2} \\
\frac{d^2}{dx^2} &= \frac{\nabla^2}{\nabla x^2} \\
\frac{d^2}{dx^2} &= \frac{\nabla}{\nabla x}\left(\frac{\Delta}{\Delta x}\right) = \frac{\Delta}{\Delta x}\left(\frac{\nabla}{\nabla x}\right) \\
\frac{d^2}{dx^2} &= \frac{\delta^2}{\delta x^2}
\end{aligned} \tag{5.4.10}$$

Resumen de Esta Sección

a) Se presentaron tres operadores de diferencias básicos.
b) Las aproximaciones de diferencias se obtienen aproximando los operadores diferenciales mediante los operadores de diferencias.
c) Al combinar los tres operadores de diferencias, se pueden obtener varias aproximaciones por diferencias.

5.5 USO DE LA DIFERENCIACION DE LOS POLINOMIOS DE INTERPOLACION DE NEWTON

Estos polinomios, tanto del tipo hacia adelante como hacia atrás, son útiles para obtener aproximaciones por diferencias; sin embargo, sólo consideraremos el caso del tipo hacia adelante. Mediante el uso de los polinomios de interpolación de Newton, se obtienen varias aproximaciones por diferencias en forma sistemática [Carnahan/Luther/Wilkes; Cheney/Kincaid].

La fórmula de interpolación de Newton hacia adelante ajustada a $N + 1$ puntos se escribe como

$$\begin{aligned}
g(x) = g(x_k + sh) &= \sum_{n=0}^{N} \binom{s}{n} \Delta^n f_k \\
&= f_k + s\Delta f_k + \frac{1}{2}s(s-1)\Delta^2 f_k + \frac{1}{6}s(s-1)(s-2)\Delta^3 f_k \\
&\quad + \frac{1}{24}s(s-1)(s-2)(s-3)\Delta^4 f_k + \cdots + \binom{s}{N}\Delta^N f_k
\end{aligned} \tag{5.5.1}$$

donde

$$s = \frac{x - x_k}{h}$$

y $f_k, f_{k+1}, \ldots, f_{k+N}$ son los valores de la función en $x_k, x_{k+1}, \ldots, x_{k+N}$, respectivamente.

Como se describió en el capítulo 2 la fórmula de interpolación de Newton hacia adelante ajustada a $N + 1$ datos es un polinomio de orden N. Sus derivadas aproximan a las derivadas de $f(x)$. La exactitud de las aproximaciones depende tanto de N como del punto dentro del rango de interpolación en donde se obtiene la derivada. Puesto que la exactitud de una interpolación de Newton es mejor en el centro del dominio de interpolación, también la exactitud de la aproximación por diferencias es mejor en el centro.

Para explicar cómo se obtienen las aproximaciones por diferencias, sea $N = 2$, con lo que la ecuación (5.5.1) se escribe como

$$g(x) = f_k + s\Delta f_k + \frac{1}{2}s(s-1)\Delta^2 f_k \tag{5.5.2}$$

Al derivar una vez se tiene

$$g'(x) = \frac{1}{h}\left[\Delta f_k + \frac{1}{2}(2s-1)\Delta^2 f_k\right] \tag{5.5.3}$$

Para $s = 0$, 1 y 2 se tiene, respectivamente,

$$g'(x_k) = \frac{1}{2h}[2\Delta f_k - \Delta^2 f_k] = \frac{1}{2h}[-f_{k+2} + 4f_{k+1} - 3f_k]$$

$$g'(x_{k+1}) = \frac{1}{2h}[2\Delta f_k + \Delta^2 f_k] = \frac{1}{2h}[f_{k+2} - f_k]$$

$$g'(x_{k+2}) = \frac{1}{2h}[2\Delta f_k + 3\Delta^2 f_k] = \frac{1}{2h}[3f_{k+2} - 4f_{k+1} + f_k]$$

Las ecuaciones anteriores son la aproximación por diferencias hacia adelante en el k-ésimo punto de la retícula, la aproximación por diferencias centrales en el punto $k + 1$ y la aproximación por diferencias hacia atrás en el punto $k + 2$, respectivamente. Si remplazamos k en la primera, segunda y tercera ecuación por i, $i - 1$, $i - 2$, respectivamente, obtenemos

$$g'(x_i) = \frac{1}{2h}[2\Delta f_i - \Delta^2 f_i] = \frac{1}{2h}[-f_{i+2} + 4f_{i+1} - 3f_i] \tag{5.5.4}$$

$$g'(x_i) = \frac{1}{2h}[2\Delta f_{i-1} + \Delta^2 f_{i-1}] = \frac{1}{2h}[f_{i+1} - f_{i-1}] \tag{5.5.5}$$

$$g'(x_i) = \frac{1}{2h}[2\Delta f_{i-2} + 3\Delta^2 f_{i-2}] = \frac{1}{2h}[3f_i - 4f_{i-1} + f_{i-2}] \tag{5.5.6}$$

Estas son las ecuaciones ya presentadas en las ecuaciones (5.2.12), (5.2.8) y (5.2.13).

El error de un polinomio de interpolación de Newton hacia adelante se representa mediante el término que se añadiría si la interpolación se ajustara a un punto

más de la retícula (véase la subsección 2.4.2). Con esta regla, evaluamos el error de las aproximaciones por diferencias en las ecuaciones (5.5.4), (5.5.5) y (5.5.6). Si en la ecuación (5.5.1), N aumenta de $N = 2$ a $N = 3$, el término adicional en la ecuación (5.5.2) es

$$\frac{1}{6}s(s-1)(s-2)\Delta^3 f_k \tag{5.5.7}$$

Su primera derivada con respecto de x es

$$\frac{1}{6h}[3s^2 - 6s + 2]\Delta^3 f_k \tag{5.5.8}$$

de lo cual se obtiene

$$\begin{aligned} &\frac{1}{3h}\Delta^3 f_k, \text{ para } s = 0 \\ &-\frac{1}{6h}\Delta^3 f_k, \text{ para } s = 1 \\ &\frac{1}{3h}\Delta^3 f_k, \text{ para } s = 2 \end{aligned} \tag{5.5.9}$$

Observemos que la N-ésima derivada de la interpolación de Newton hacia adelante de orden N es

$$\frac{d^N}{dx^N}g(x) = \frac{1}{h^N}\Delta^N f_i \tag{5.5.10}$$

La ecuación es una aproximación de la N-ésima derivada de $f(x)$ en todo el rango de interpolación. Por lo tanto, podemos escribir

$$\Delta^N f_i \simeq h^N f^{(N)}(x) \tag{5.5.11}$$

donde $f^{(n)}$ denota la N-ésima derivada de $f(x)$. Utilizamos la ecuación (5.5.11), con lo que la ecuación (5.5.9) es, aproximadamente,

$$\frac{1}{3}h^2 f_k''', \text{ para } s = 0 \tag{5.5.12}$$

$$-\frac{1}{6}h^2 f_k''', \text{ para } s = 1 \tag{5.5.13}$$

$$\frac{1}{3}h^2 f_k''', \text{ para } s = 2 \tag{5.5.14}$$

Cada una de las expresiones anteriores representa los errores de las ecuaciones (5.5.4), (5.5.5) y (5.5.6), respectivamente y coinciden con los errores obtenidos en la sección 5.2 al utilizar los desarrollos de Taylor.

En general, una aproximación por diferencias de orden p se obtiene diferenciando un polinomio de interpolación de Newton hacia adelante de orden mayor o igual que p. Una aproximación por diferencias de gran precisión se obtiene aumentando el orden del polinomio de interpolación de Newton hacia adelante. Por lo tanto, una aproximación por diferencias más precisa requiere más puntos en la retícula. La precisión de una aproximación por diferencias es mejor en el centro del rango de interpolación. La aproximación por diferencias centrales se puede considerar como la derivada de la interpolación de Newton en el centro del rango de interpolación. Por otro lado, las aproximaciones por diferencias hacia adelante y hacia atrás son las derivadas del polinomio de interpolación de Newton en las orillas del rango de interpolación. Por consiguiente, la aproximación por diferencias centrales siempre es más exacta que las aproximaciones por diferencias hacia adelante, hacia atrás o cualquier otra que se refiera a un extremo y que utilicen el mismo polinomio de interpolación de Newton.

Resumen de Esta Sección

a) Las aproximaciones por diferencias se pueden deducir derivando un polinomio de interpolación, por ejemplo, el polinomio de interpolación de Newton hacia adelante.
b) El término del error en la aproximación por diferencias se obtiene utilizando el término adicional que surge con el uso de un punto adicional de los datos.
c) Las aproximaciones por diferencias que se construyen aplicando la fórmula de interpolación son consistentes con las que se obtienen por medio de los desarrollos de Taylor.
d) La precisión de una fórmula de interpolación que utiliza puntos de la retícula separados de manera uniforme es más grande en el centro del dominio de interpolación. En consecuencia, la aproximación por diferencias es más exacta si se usa la derivada de una fórmula de interpolación en el centro del dominio. Esto explica por qué la aproximación por diferencias centrales siempre es más precisa que las aproximaciones por diferencias hacia adelante y hacia atrás que usan el mismo número de puntos.

5.6 APROXIMACION DE DERIVADAS PARCIALES POR DIFERENCIAS

Las aproximaciones por diferencias para el caso de las derivadas parciales de funciones multidimensionales son esencialmente iguales al caso de la diferenciación numérica de las funciones unidimensionales.

Consideremos una función bidimensional $f(x, y)$. La aproximación por diferencias de la derivada parcial

$$f_x = \frac{\partial}{\partial x} f(x, y) \quad \text{en } x = x_0 \quad \text{y} \quad y = y_0 \tag{5.6.1}$$

se puede deducir fijando y en y_0 y considerando a $f(x, y_0)$ como una función unidimensional. Por lo tanto, las aproximaciones por diferencias hacia adelante, hacia atrás y centrales para la derivada parcial anterior se pueden escribir respectivamente como

$$f_x \simeq \frac{f(x_0 + \Delta x, y_0) - f(x_0, y_0)}{\Delta x} \tag{5.6.2a}$$

$$f_x \simeq \frac{f(x_0 + \Delta x, y_0) - f(x_0 - \Delta x, y_0)}{2\Delta x} \tag{5.6.2b}$$

$$f_x \simeq \frac{f(x_0, y_0) - f(x_0 - \Delta x, y_0)}{\Delta x} \tag{5.6.2c}$$

Las aproximaciones por diferencias centrales para las segundas derivadas parciales de $f(x, y)$ en x_0 y y_0 son

$$f_{xx} = \frac{\partial^2}{\partial x^2} f \simeq \frac{f(x_0 + \Delta x, y_0) - 2f(x_0, y_0) + f(x_0 - \Delta x, y_0)}{\Delta x^2} \tag{5.6.3a}$$

$$f_{yy} = \frac{\partial^2}{\partial y^2} f \simeq \frac{f(x_0, y_0 + \Delta y) - 2f(x_0, y_0) + f(x_0, y_0 - \Delta y)}{\Delta y^2} \tag{5.6.3b}$$

$$f_{xy} = \frac{\partial\partial}{\partial x \partial y} f \simeq \frac{f(x_0 + \Delta x, y_0 + \Delta y) - f(x_0 + \Delta x, y_0 - \Delta y)}{4\Delta x \Delta y} + \frac{-f(x_0 - \Delta x, y_0 + \Delta y) + f(x_0 - \Delta x, y_0 - \Delta y)}{4\Delta x \Delta y} \tag{5.6.3c}$$

Ejemplo 5.2

La tabla siguiente muestra los valores de una función bidimensional $f(x, y)$:

y	$x = 1.0$	1.5	2.0	2.5	3.0
1.0	1.63	2.05	2.50	2.98	3.49
1.5	1.98	2.51	3.08	3.69	4.33
2.0	2.28	2.91	3.61	4.37	5.17
2.5	2.64	3.25	4.08	5.00	5.98
3.0	2.65	3.50	4.48	5.57	6.76

a) Utilice las aproximaciones por diferencias centrales para evaluar las siguientes derivadas parciales:

$$f_x(2, 2),\ f_y(2, 2),\ f_{yy}(2, 2) \quad \text{y} \quad f_{xy}(2, 2)$$

b) Use la aproximación por diferencias hacia adelante de tres puntos para evaluar las siguientes derivadas parciales:

$$f_x(2, 2),\ f_y(2, 2)$$

(Solución)

Al emplear la definición $\Delta x = \Delta y = h = 0.5$, se hacen los cálculos como sigue:

a) $$f_x(2, 2) = \frac{f(2 + h, 2) - f(2 - h, 2)}{2h} = \frac{4.37 - 2.91}{(2)(0.5)} = 1.46$$

$$f_y(2, 2) = \frac{f(2, 2 + h) - f(2, 2 - h)}{2h} = \frac{4.08 - 3.08}{(2)(0.5)} = 1.00$$

$$f_{yy}(2, 2) = \frac{f(2, 2 + h) - 2f(2, 2) + f(2, 2 - h)}{h^2}$$

$$= \frac{4.08 - 2(3.61) + 3.08}{(0.5)^2} = -0.24$$

$$f_{xy}(2, 2) = \frac{f(2 + h, 2 + h) - f(2 + h, 2 - h) - f(2 - h, 2 + h) + f(2 - h, 2 - h)}{(2h)^2}$$

$$= \frac{5.0 - 3.69 - 3.25 + 2.51}{[2(0.5)]^2} = 0.57$$

b) $$f_x(2, 2) = \frac{-f(2 + 2h, 2) + 4f(2 + h, 2) - 3f(2, 2)}{2h}$$

$$= \frac{-(5.17) + 4(4.37) - 3(3.61)}{(2)(0.5)} = 1.48$$

$$f_y(2, 2) = \frac{-f(2, 2 + 2h) + 4f(2, 2 + h) - 3f(2, 2)}{2h}$$

$$= \frac{-(4.48) + 4(4.08) - 3(3.61)}{(2)(0.5)} = 1.01$$

Resumen de Esta Sección. Las aproximaciones por diferencias para las derivadas parciales son esencialmente iguales a las derivadas ordinarias. Por lo tanto, todas las aproximaciones por diferencias desarrolladas para el caso de las derivadas ordinarias se aplican a las derivadas parciales.

PROGRAMAS

PROGRAMA 5-1 Cálculo de aproximaciones por diferencias

A) Explicaciones

El programa encuentra la aproximación por diferencias para la derivada del orden deseado utilizando los puntos de la retícula especificados por el usuario.

El programa pregunta por: 1) el número de puntos en la retícula que se usarán en la fórmula para la aproximación por diferencias (se puede utilizar un máximo de 10 puntos); 2) los índices de los puntos de la retícula, y 3) el orden de la derivada que se aproxima. El programa supondrá que el intervalo de separación es h, sin que el usuario especifique su valor numérico.

Para especificar la aproximación por diferencias deseada, denotamos el número de puntos en la retícula por L, el orden de la derivada por p y las abscisas de los puntos de la retícula como $x_\alpha, x_\beta, \ldots, x_\lambda$, donde $x_\alpha = \alpha h$, $x_\beta = \beta h, \ldots, x_\lambda = \lambda h$.

El algoritmo funciona incluso cuando $\alpha, \beta, \ldots, \lambda$ no son enteros sino que representan cualquier valor decimal positivo o negativo. Las coordenadas de los puntos en la retícula se hacen $x_i = ih$, $i = \alpha, \beta, \ldots, \lambda$ y la derivada del orden especificado se evalúa en $x = 0$. Por ejemplo, si se va a evaluar $f''(0)$ utilizando puntos en $x = -2$, 0.5 y 1.5, entonces hacemos $L = 3$, α —2, $\beta = 0.5$ y $\gamma = 1.5$ con $h = 1$. con $h = 1$.

La salida del programa está dada en la forma de los coeficientes de la ecuación (5.3.1); es decir, $a_\alpha, a_\beta, \ldots, a_\lambda$, y c_1 y c_2 de la ecuación (5.3.2). Como se vio en la sección 5.3, el segundo término de la ecuación (5.3.2) se debe ignorar si $c_1 = 0$.

Los coeficientes de las ecuaciones lineales se guardan en el arreglo A(K, L) al igual que en B(K, L). El primero se utiliza para la solución, mientras que el segundo se reserva para su uso posterior. Las ecuaciones lineales se resuelven mediante la subrutina de la eliminación gaussiana. En el capítulo 6 se explicarán más detalles del esquema de eliminación gaussiana. Al regresar de la subrutina, la solución de las ecuaciones lineales se guarda en A(K, KM + 1), K = 1, 2, . . . , KM. Los coeficientes de la aproximación por diferencias son números decimales en primera instancia. Para expresarlos en forma racional con coeficientes enteros en el numerador y denominador, se hacen algunos cálculos adicionales.

B) Variables

KM: número de puntos en la retícula que se utilizará en la aproximación por diferencias (L)

EL(K): índice de los puntos de la retícula para el K-ésimo punto contando desde la izquierda (valores de $\alpha, \beta, \gamma, \ldots$)

DR: orden de la derivada que se aproximará (p)

A(K, L): coeficientes de la ecuación lineal [véase la ecuación (5.3.5)]

C(K): coeficientes del K-ésimo valor de la función en el numerador de la aproximación por diferencias para K $\leqslant$ KM: coeficiente del término del error para K > KM

F: recíproco del denominador de la aproximación por diferencias

C) Listado

```
C-----CSL/F5-1      CALCULO DE APROXIMACIONES POR DIFERENCIAS
      COMMON N, A(10,11), EL(10), B(10,11),C(10) , CF(11)
      MT=6
      PRINT *, 'CSL/F5-1     CALCULO DE APROXIMACIONES POR DIFERENCIAS'
      PRINT *
90    PRINT *, '¿NUMERO DE PUNTOS?'
      READ*, KM
92    IF (KM.GT.2.OR.KM.LE.10) GOTO 95
      PRINT *, ' ENTRADA NO VALIDA. POR FAVOR REPITA LA ENTRADA'
      GO TO 90
95    PRINT *, ' ¿NUMERO DE PUNTOS EN LA RETICULA? ',
     1 '   (OPRIMA ENTER DESPUES DE CADA NUMERO)'
```

```
          DO  K=1 , KM
             print 98, k
             READ *,  EL(K)
98           FORMAT( '¿INDICE DEL PUNTO?,',I2,'?')
          END DO
103     PRINT *, '  DE EL ORDEN DE LA DERIVADA '
105     READ *, KDR
106     Z=1.0
        DO I =1,KDR
           Z=Z*FLOAT(I)
        END DO
110     DO 130  K=1, KM+2
           DO L=1, KM
             IF(K.EQ.1) A(K,L)=1.0
             IF(K.GT.1) A(K,L)=EL(L)**(K-1)      ! Preparación de los coeficientes de la matriz
             B(K,L)=A(K,L)                       ! Almacenamiento de los mismos en B (1, J)
           END DO
130     CONTINUE
135     FF=1
           DO K=1, KM
             A(K,KM+1)=0
             IF (K-1.EQ.KDR)   A(K,KM+1)=Z       ! Término no homogéneo
           END DO
140     N=KM
160     PRINT *
        KMP2=KM+2
        CALL GAUSS
170     DO 190 K=1, KM+2
           C(K)=0.0
             DO L=1, KM
               C(K)=C(K)+B(K,L)*A(L,KM+1)        ! Coeficientes del término del error
             END DO
190     CONTINUE
191     F=1000.0
        DO 194  K=1, KM
           IF( A(K,KM+1).EQ.0)  GOTO 194
           IF( ABS(A(K,KM+1)).LT.0.0001)    GOTO 194
           U=ABS(A(K,KM+1))
           IF (U .LT.F) F=U
194     CONTINUE
C                                   !-- Coeficientes de la fórmula de diferencias
           DO K=1, KM
             CF(K)=A(K,KM+1)/F
           END DO
198     print 197
197     FORMAT(' ESQUEMA DE DIFERENCIAS '
        DO 210 K=1,KM
           FINV=1.0/F
           print 7002, CF(K),FINV,KDR,EL(K)
7002    FORMAT(1X,'+[', F10.5,'/(',F8.5,' H**',I1,')] F(',F6.3,'H)')
210     CONTINUE
        print *
        print 7005
7005    FORMAT(' TERMINO DEL ERROR '
217     DO K=1, KM+2
           IF (ABS(C(K)).LT.0.00000001) C(K)=0
        END DO
        DD=1.0
        DO K=1,KM
           DD=DD*FLOAT(K)
        END DO
```

```
C
      DO K=KM+1, KM+2
         CM=-C(K)
         CPDD=-C(K)/DD
         KM1=K-1
         NH=KM1-KDR
         IF(K.EQ.KM+1.AND.CM.NE.0)
     &                  print 7020, CM,DD, NH, KM1     ! Imprime los términos del error
         IF(K.EQ.KM+2) print 7020, CM,DD, NH, KM1
         DD=DD*FLOAT(K)
      END DO
7020  FORMAT( 5X,'(', F10.5,'/',F10.5,')H**',I1,2X,'F^(',  I1, ')')
      print 7030
7030  FORMAT( /'---------------------------------------------')
      PRINT *
      PRINT *,' OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
      READ *,KK
      IF (KK.EQ.1) GO TO 90
      PRINT *
      END
C*********************
      SUBROUTINE GAUSS        !-- Solución de ecuaciones simultáneas
      COMMON N, A(10,11), EL(10), B(10,11),C(10)
      NM=N-1
      N1=N+1
      DO 1085 I=1,NM
        IPV=I
        I1=I+1
          DO J=I1, N
            IF(ABS(A(IPV,I)).LT.ABS(A(J,I))) IPV=J
          END DO
      IF (IPV.EQ.I) GOTO 1060
        DO JC=1,N1
          TM=A(I,JC)
          A(I,JC)=A(IPV,JC)
          A(IPV,JC)=TM
        END DO
1060    DO 1080 JR=I1, N                    !  Comienza la eliminación hacia adelante
          IF (A(JR,I).EQ.0) GOTO 1080
          IF(A(I,I).EQ.0) PRINT *,I,JR,(A(I,JJJ),JJJ=1,3)
          R=A(JR,I)/A(I,I)
          N1=N+1
1075        DO KC=I1, N1
              A(JR,KC)=A(JR,KC)-R*A(I,KC)
            END DO
1080    CONTINUE
1085  CONTINUE
1090  IF (A(N,N).EQ.0)  GOTO 1200
1100  A(N,N+1)=A(N,N+1)/A(N,N)              !  Comienza la sustitución hacia atrás
1110  DO  1130  NV=NM, 1, -1
        VA=A(NV,N+1)
        NV1=NV+1
          DO K=NV1, N
            VA=VA-A(NV,K)*A(K,N+1)
          END DO
        A(NV,N+1)=VA/A(NV,NV)
1130  CONTINUE
      RETURN
1200  print 1210
1210  FORMAT( '  LA MATRIZ ES SINGULAR ')
      STOP
      END
```

D) Ejemplo de salida

```
CSL/F5-1     CALCULO DE APROXIMACIONES POR DIFERENCIAS

¿NUMERO DE PUNTOS?
3
 ¿NUMERO DE PUNTOS EN LA RETICULA? (OPRIMA ENTER DESPUES DE CADA NUMERO)
 ¿INDICE DEL PUNTO? 1?
0
¿INDICE DEL PUNTO,  2?
1
¿INDICE DEL PUNTO,  3?
2
  DE EL ORDEN DE LA DERIVADA
1

ESQUEMA DE DIFERENCIAS
+[   -3.00000/( 2.00000 H**1)] F( 0.000H)
+[    4.00000/( 2.00000 H**1)] F( 1.000H)
+[   -1.00000/( 2.00000 H**1)] F( 2.000H)

TERMINO DE ERROR
     (    2.00000/    6.00000)H**2   F^(3)
     (    6.00000/   24.00000)H**3   F^(4)
-----------------------------------------------
OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
¿NUMERO DE PUNTOS?
5
  ¿NUMERO DE PUNTOS EN LA RETICULA?    (OPRIMA ENTER DESPUES DE CADA NUMERO)
¿INDICE DEL PUNTO?, 1?
-2
¿INDICE DEL PUNTO?, 2?
-1
¿INDICE DEL PUNTO?, 3?
0
¿INDICE DEL PUNTO?, 4?
1
¿INDICE DEL PUNTO?, 5?
2
  DE EL ORDEN DE LA DERIVADA
2

ESQUEMA DE DIFERENCIAS
+[   -1.00000/(12.00000 H**2)] F(-2.000H)
+[   16.00000/(12.00000 H**2)] F(-1.000H)
+[  -30.00000/(12.00000 H**2)] F( 0.000H)
+[   16.00000/(12.00000 H**2)] F( 1.000H)
+[   -1.00000/(12.00000 H**2)] F( 2.000H)

TERMINO DE ERROR
     (    0.00000/  120.00000)H**3   F^(5)
     (    8.00000/  720.00000)H**4   F^(6)
-----------------------------------------------
OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR
```

PROBLEMAS

5.1) Evalúe la primera derivada de $y(x) = \text{sen}\,(x)$ para $x = 1$ y $h = 0.001, 0.005, 0.01, 0.05, 0.1$ y 0.5 mediante los tres esquemas diferentes:

a) $y'(1) = [y(1 + h) - y(1)]/h$

b) $y'(1) = [y(1) - y(1 - h)]/h$

c) $y'(1) = [y(1 + h/2) - y(1 - h/2)]/h$

Evalúe los errores comparándolos con los valores exactos.

5.2) Calcule $d\sqrt{x}/dx$ en $x = 1$ utilizando las aproximaciones por diferencias hacia adelante, hacia atrás y centrales con $h = 0.1$, 0.05 y 0.025. Evalúe el error de cada resultado **a)** comparándolo con el valor exacto y **b)** utilizando los términos del error que se muestra en la tabla 5.2; es decir, $\frac{1}{2}hf''$, $\frac{1}{2}hf''$ y $-\frac{1}{6}h^2f''$, respectivamente.

5.3) Obtenga una aproximación por diferencias y el término del error para f_i utilizando: **i)** f_{i+1} y f_{i+2}; **ii)** f_{i-1}, f_i y f_{i+2}, y **iii)** f_{i-2} y f_{i+2}. Suponga que los puntos de la retícula tiene una separación uniforme.

5.4) Obtenga una aproximación por diferencias y el término del error para f_i'' utilizando f_i, f_{i-1} y f_{i-2} (aproximación por diferencias hacia atrás de tres puntos para f_i'').

5.5) Repita el problema 5.2 con las aproximaciones hacia adelante y hacia atrás exactas de segundo orden:

a) $y'(1) = [-y(1 + 2h) + 4y(1 + h) - 3y(1)]/2h$

b) $y'(1) = [3y(1) - 4y(1 - h) + y(1 - 2h)]/2h$

y evalúe los errores comparando con el valor exacto de $y'(1)$.

5.6) Calcule la primera derivada $y'(1)$, donde $y(x) + \text{sen}(x)$ mediante las aproximaciones por diferencias hacia adelante y hacia atrás exactas de segundo orden que se utilizaron en el problema 5.5 para $h = 0.001$, 0.005, 0.01, 0.1 y 0.5. Evalúe también el error de cada aproximación numérica, comparándolo con el valor real. Si se observa un crecimiento del error al reducir h, explique la razón.

5.7) Considere una varilla uniforme de 1 metro de longitud apoyada en dos extremos; el momento del doblamento está dado por la siguiente fórmula:

$$y'' = M(x)/EI$$

donde $y(x)$ es la deflección, $M(x)$ es el momento de doblamiento y EI es la rigidez en la unión. Calcule el momento de doblamiento en cada punto de la retícula —incluyendo los dos extremos— suponiendo que la distribución de la deflección tiene los siguientes valores:

i	x_i	y_i
0	0.0 (m)	0.0 (cm)
1	0.2	7.78
2	0.4	10.68
3	0.6	8.37
4	0.8	3.97
5	1.0	0.0

Suponga que $EI = 1.2 \times 10$ Nm^2. Utilice la aproximación por diferencias centrales para los puntos de la retícula distintos de los extremos. Para éstos, utilice la aproximación por diferencias hacia adelante o hacia atrás utilizando cuatro puntos.

5.8) Evalúe la segunda derivada de tan (x) en $x = 1$ por la fórmula de diferencias centrales utilizando $h = 0.1$, 0.05 y 0.02. Determine el error comparándolo con el valor real y muestre que el error es proporcional a h^2.

5.9) La distribución de la velocidad de un fluido cerca de una superficie plana está dada por la siguiente tabla:

i	y_i (m)	u_i (m/s)
0	0.0	0.0
1	0.002	0.006180
2	0.004	0.011756
3	0.006	0.016180
4	0.008	0.019021

La ley de Newton para la tensión superficial está dada por

$$\tau = \mu \frac{d}{dy} u$$

donde μ es la viscosidad que suponemos vale 0.001 Ns/m^2. Calcule la tensión superficial en $y = 0$ mediante aproximación por diferencias utilizando los siguientes puntos: i) $i = 0$ y 1; ii) $i = 0$, 1 y 2.

5.10) a) Conociendo el término del error para

$$f'_i \simeq \frac{f_i - f_{i-1}}{h}$$

estime el término del error para

$$f'_i \simeq \frac{f_i - f_{i-2}}{2h}$$

b) La precisión de una aproximación por diferencias se puede mejorar mediante una combinación lineal de dos aproximaciones por diferencias, de forma que los errores de truncamiento de orden menor en las dos aproximaciones se cancelen. Determine α de la siguiente aproximación de forma que se optimice la precisión:

$$f'_i \simeq \alpha \frac{f_i - f_{i-1}}{h} + (1 - \alpha) \frac{f_i - f_{i-2}}{2h}$$

5.11) Determine α de

$$f''_i \simeq \alpha \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + (1 - \alpha) \frac{f_{i+2} - 2f_i + f_{i-2}}{(2h)^2}$$

para que la precisión se maximice. Sugerencia: elimine el término principal del error en

$$f''_i \simeq \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}$$

5.12) Obtenga las aproximaciones por diferencias más precisas para f'_i y f''_i utilizando $f_{i-2}, f_{i-1}, f_i, f_{i+1}$ y f_{i+2}. Suponga que el espaciamiento es constante.

5.13) Aplique los desarrollos de Taylor para obtener las aproximaciones por diferencias de f'_i y f''_i utilizando f_i, f_{i+1}, f_{i+2} y f_{i+3} con la más alta precisión posible. Suponga que el espaciamiento es constante.

5.14) A continuación se da una tabla de valores

x	f
−0.1	4.157
0	4.020
0.2	4.441

a) Obtenga la mejor aproximación por diferencias para calcular $f'(0)$ con los datos proporcionados.

b) ¿Cuál es el término del error para la aproximación por diferencias?

c) Calcule $f'(0)$ mediante la fórmula obtenida.

5.15) Evalúe el error de truncamiento de la siguiente fórmula de diferencias:

$$f'_i \simeq \frac{-f_{i+3} + 9f_{i+1} - 8f_i}{6h}$$

5.16) A continuación se proporcionan dos aproximaciones por diferencias para la cuarta derivada:

$$f''''_i = \frac{f_{i+4} - 4f_{i+3} + 6f_{i+2} - 4f_{i+1} + f_i}{h^4} + 0(h)$$

$$f''''_i = \frac{f_{i+2} - 4f_{i+1} + 6f_i - 4f_{i-1} + f_{i-2}}{h^4} + 0(h^2)$$

Utilice el desarrollo de Taylor para encontrar los términos del error.

5.17) Demuestre las ecuaciones siguientes:

$$\nabla^3 f_i = \Delta^3 f_{i-3}$$
$$\Delta^4 f_i = \nabla^4 f_{i+4}$$
$$\Delta^3 \nabla f_i = \Delta^4 f_{i-1}$$
$$\delta^2 f_i = \Delta^2 f_{i-1}$$
$$\Delta^n \nabla^m f_i = \Delta^{n+m} f_{i-m}$$

5.18) Encuentre m si $\nabla^5 f_i = \Delta^4 \nabla f_{i+m}$.

5.19) Escriba explícitamente las aproximaciones por diferencias siguientes, en términos de f_i y estime el orden del error en cada uno:

a) $f''_i = \Delta^2 f_i / h^2$

b) $f''_i = \nabla^2 f_i / h^2$

c) $f''_i = \nabla\Delta\, f_i / h^2$

5.20) Formule las siguientes aproximaciones por diferencias, en términos de f_i y obtenga los términos del error:

a) $f'''_i = \Delta^3 f_i / h^3$

b) $f'''_i = \Delta^2 \nabla f_i / h^3$

c) $f'''_i = \Delta \nabla^2 f_i / h^3$

d) $f'''_i = \nabla^3 f_i / h^3$

e) $f'''_i = \frac{1}{2}[\Delta^2 \nabla f_i / h^3 + \Delta \nabla^2 f_i / h^3]$

5.21) Obtenga la aproximación por diferencias centrales de f_i'', utilizando la interpolación de Newton hacia adelante de orden cuatro (ajustada a cinco puntos) y evalúe el término del error mediante la fórmula de interpolación de Newton del orden inmediato superior.

5.22) Muestre que la primera derivada de la ecuación (5.4.1) está dada por

$$g'(x) = \frac{1}{h}\left[\Delta f_k + \frac{1}{2}(2s-1)\Delta^2 f_k + \frac{1}{6}(3s^2 - 6s + 2)\Delta^3 f_k \right.$$
$$\left. + \frac{1}{24}(4s^3 - 18s^2 + 22s - 6)\Delta^4 f_k + \cdots + \frac{d}{ds}\binom{s}{N}\Delta^N f_k\right]$$

donde $x = x_i + sh$.

5.23) Exprese las siguientes fórmulas mediante el operador de desplazamiento (véase el apéndice c).

a) $\frac{1}{2}(\nabla^2\Delta + \Delta\nabla^2)f_i$

b) $\nabla^2\Delta^2 f_i$

5.24) Obtenga aproximaciones por diferencias para f_i'' y f_i''' mediante las diferencias del problema anterior y evalúe el orden del error utilizando el desarrollo de Taylor.

5.25) Las siguientes son aproximaciones por diferencias de $y^{(iv)}$:

$$\nabla^4 y_i/h^4, \quad \nabla^3\Delta y_i/h^4, \quad \nabla^2\Delta^2 y_i/h^4, \quad \nabla\Delta^3 y_i/h^4, \quad \Delta^4 y_i/h^4$$

a) Escriba explícitamente las aproximaciones por diferencias, en términos de f_i.

b) Evalúe el término del error para cada aproximación utilizando los polinomios de interpolación de Newton hacia adelante. (Véase la sección 5.4 para evaluar los errores de las fórmulas hacia atrás mediante los polinomios de interpolación de Newton hacia adelante.)

5.26) La distribución de la velocidad de un fluido cerca de una superficie plana está dada por

i	y_i (m)	u_i (m/s)
0	0.0	0.0
1	0.001	0.4171
2	0.003	0.9080
3	0.006	1.6180

donde y es la distancia a la superficie y u es la velocidad. Si el flujo es laminar y $\mu = 0.001$ Ns/m^2, calcule la tensión superficial en $y = 0$ utilizando los siguientes datos:

a) $i = 0$ y 1

b) $i = 0, 1,$ y 2

(*Sugerencia*: véase el problema 5.9 para la ley de viscosidad de Newton).

5.27) Obtenga las aproximaciones por diferencias de f_0', derivando las fórmulas de interpolación de Newton hacia adelante, ajustadas a los siguientes datos y evalúe los términos del error para cada fórmula de aproximación:

a) f_0, f_1

b) f_0, f_1, f_2

c) f_0, f_1, f_2, f_3

5.28) Obtenga las aproximaciones por diferencias de f_i' y f_i'' utilizando f_i, f_{i+1}, f_{i+2} y f_{i+3} que tengan la mayor precisión posible. Suponga que la separación es constante.

5.29) Repita el problema 5.27) utilizando los siguientes datos

a) f_{-1}, f_0, f_1

b) f_{-2}, f_{-1}, f_0

c) $f_{-2}, f_{-1}, f_0, f_1, f_2$

5.30) Demuestre por inducción las siguientes relaciones:

a) $\Delta^n y(x) = \sum_{k=0}^{n} (-1)^k \frac{n!}{k!(n-k)!} y(x + nh - kh)$

b) $\nabla^n y(x) = \sum_{k=0}^{n} (-1)^k \frac{n!}{k!(n-k)!} y(x - kh)$

c) $\delta^{2n} y(x) = \sum_{k=0}^{2n} (-1)^k \frac{2n!}{k!(2n-k)!} y(x + 2nh - kh)$

5.31) Deduzca la aproximación por diferencias hacia adelante para la primera derivada que tenga una precisión de tercer orden (el error es proporcional a f''') en una retícula con separación uniforme, utilizando el polinomio de interpolación de Newton hacia adelante.

5.32) Obtenga la aproximación por diferencias de $f''(x_i)$, utilizando los tres puntos que se muestran abajo:

Figura P5.32

5.33) Obtenga la aproximación por diferencias de $f'''(x_i)$ utilizando los cuatro puntos de la retícula que se muestran abajo:

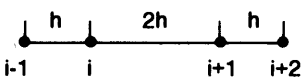


Figura P5.33

5.34) La tabla de valores de $f(x, y)$ es:

y	$x = 0.0$	0.5	1.0	1.5	2.0
0.0	0.0775	0.1573	0.2412	0.3309	0.4274
0.5	0.1528	0.3104	0.4767	0.6552	0.8478
1.0	0.2235	0.4547	0.7002	0.9653	1.2533
1.5	0.2866	0.5846	0.9040	1.2525	1.6348

a) Evalúe $(\partial / \partial y) f$ en $x = 1.0$ y $y = 0$ utilizando la aproximación por diferencias hacia adelante con un error del orden de h^2, donde $h = 0.5$.

b) Evalúe $(\partial^2 / \partial x^2)$ en $x = 1.0$ y $y = 1.0$, utilizando la aproximación por diferencias centrales con un error del orden de h^2, donde $h = 0.5$.

c) Evalúe $(\partial^2/\partial x \partial y)f$ en $x = 0$ y $y = 0$, utilizando la aproximación por diferencias hacia adelante, con un error del orden de h^2, donde $h = 0.5$.

BIBLIOGRAFIA

Carnahan, B., H. A. Luther y J. O. Wilkes, *Applied Numerical Methods*, Wiley, 1969.

Cheney, W. y D. Kincaid, *Numerical Mathematics and Computing*, Brooks/Cole, 1985.

Isaacson, E. y H. B. Keller, *Analysis of Numerical Methods*, Wiley, 1966.

Hornbeck, R. W., *Numerical Methods*, Quantum, 1975.

James, M. L., G. M. Smith y J. C. Wolford, *Applied Numerical Methods for Digital Computations*, 3a. edición, Harper & Row, 1985.

Ralston, A., *A First Course in Numerical Analysis*, McGraw-Hill, 1965.

6
Algebra lineal numérica

6.1 INTRODUCCION

El objetivo principal de este capítulo es el estudio de los métodos computacionales básicos para resolver conjuntos no homogéneos de ecuaciones lineales. El álgebra lineal es fundamental, tanto para el análisis científico como para los métodos numéricos, que no podríamos hacer mucho sin tener un conocimiento básico de ella.

Los primeros temas de este capítulo son las eliminaciones de Gauss y Gauss-Jordan para resolver ecuaciones lineales. Se describen sin la notación vectorial o matricial. No obstante, se introduce a continuación de esto un mínimo de notaciones vectoriales/matriciales y sus reglas básicas. Después se analizan tres temas interrelacionados: la inversión de una matriz, la descomposición LU y el determinante. Concluimos con el estudio de la solución de m ecuaciones con n incógnitas.

Tabla 6.1 Comparación de los tres métodos para las ecuaciones lineales

Método	Ventajas	Desventajas
Eliminación de Gauss	El algoritmo de solución más básico.	Solución de un único conjunto de ecuaciones lineales a la vez.
Eliminación de Gauss-Jordan	La base para calcular la inversa; puede resolver conjuntos múltiples de ecuaciones.	Menos eficiente para un único conjunto de ecuaciones.
Descomposición LU	Eficaz si un conjunto de ecuaciones lineales se resuelve varias veces con distintos términos no homogéneos (por ejemplo, en el método de la potencia inversa).	Menos eficiente y más complicado que la eliminación de Gauss si sólo se usa una vez.

Los temas referentes a los conjuntos de ecuaciones lineales homogéneas se dejan pendientes para el próximo capítulo.

En la tabla 6.1 se resumen las ventajas y desventajas del uso de las eliminaciones de Gauss y Gauss-Jordan, así como la descomposición LU.

6.2 ELIMINACIONES DE GAUSS Y GAUSS-JORDAN PARA PROBLEMAS IDEALES SENCILLOS

La eliminación de Gauss es el método que se utiliza en forma más amplia para resolver un conjunto de ecuaciones lineales. En esta sección, estudiaremos la eliminación de Gauss y su variante, la eliminación de Gauss-Jordan sin pivoteo. El pivoteo en la eliminación de Gauss se analizará en la sección siguiente.

Un conjunto de N ecuaciones es de la forma

$$\begin{aligned} a_{1,1}x_1 + a_{1,2}x_2 + a_{1,3}x_3 + \cdots + a_{1,N}x_N &= y_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + a_{2,3}x_3 + \cdots + a_{2,N}x_N &= y_2 \\ &\vdots \\ a_{N,1}x_1 + a_{N,2}x_2 + a_{N,3}x_3 + \cdots + a_{N,N}x_N &= y_N \end{aligned} \tag{6.2.1}$$

donde los $a_{i,j}$ son coeficientes, los x_i son las incógnitas y los y_i *son términos conocidos llamados términos libres o independientes*. En este caso, el número de incógnitas es igual al número de ecuaciones, que es la forma más usual de un conjunto de ecuaciones lineales. Si estos números son distintos, pueden existir las soluciones, pero esto debe estudiarse con más cuidado. Los problemas en donde el número de ecuaciones es distinto del número de incógnitas se reservan para las secciones 6.10.

Cuando al menos uno de los términos libres de la ecuación (6.2.1) es distinto de cero, se dice que el conjunto es no homogéneo. La eliminación de Gauss se aplica sólo al caso de los conjuntos no homogéneos de ecuaciones. No siempre puede ser fácil la solución de un conjunto de ecuaciones lineales, debido al hecho de que quizá no tenga una solución única. Aunque tuviera una solución única, la solución calculada puede ser inexacta en el caso de un problema mal condicionado.

Sin embargo, para simplificar la exposición, consideraremos un problema ideal en el que el conjunto de ecuaciones tiene una solución única y no aparece ninguna dificultad en el proceso de solución. La eliminación de Gauss consiste en: a) la eliminación hacia adelante, y b) la sustitución hacia atrás. La eliminación hacia adelante se lleva a cabo de la manera siguiente.

La primera ecuación se multiplica por $a_{2,1}/a_{1,1}$ y se le resta a la segunda ecuación para eliminar el primer término de la segunda; de la misma forma, el primer término de las ecuaciones restantes, $i > 2$, se elimina restando la primera ecuación multiplicada por $a_{i,1}/a_{1,1}$. Así, las ecuaciones se deberían ver así:

$$\begin{aligned} a_{1,1}x_1 + a_{1,2}x_2 + a_{1,3}x_3 + \cdots + a_{1,N}x_N &= y_1 \\ a'_{2,2}x_2 + a'_{2,3}x_3 + \cdots + a'_{2,N}x_N &= y'_2 \\ &\vdots \\ a'_{N,2}x_2 + a'_{N,3}x_3 + \cdots + a'_{N,N}x_N &= y'_N \end{aligned} \tag{6.2.2}$$

donde

$$a'_{i,j} = a_{i,j} - (a_{i,1}/a_{1,1})a_{1,j}$$

Conviene observar que la primera ecuación no ha cambiado.

En seguida, el segundo término de cada una de las ecuaciones, desde la tercera hasta la última, $i > 2$, se elimina restando la segunda ecuación multiplicada por $a'_{i,2}/a'_{2,2}$. Después de terminar este paso, se eliminan los terceros términos de las demás ecuaciones, de la cuarta a la última. Al finalizar este proceso de eliminación hacia adelante, el conjunto de ecuaciones se verá de la forma siguiente:

$$\begin{aligned} a_{1,1}x_1 + a_{1,2}x_2 + a_{1,3}x_3 + \cdots + a_{1,N}x_N &= y_1 \\ a'_{2,2}x_2 + a'_{2,3}x_3 + \cdots + a'_{2,N}x_N &= y'_2 \\ a''_{3,3}x_3 + \cdots + a''_{3,N}x_N &= y''_3 \\ &\vdots \\ a^{(N-1)}_{N,N}x_N &= y^{(N-1)}_N \end{aligned} \tag{6.2.3}$$

Los términos principales de cada una de las ecuaciones anteriores reciben el nombre de *pivotes*. Se podría normalizar cada una de las ecuaciones, dividiendo entre el coeficiente principal, pero esto no se utiliza en la eliminación de Gauss; la razón fundamental es que la normalización de las ecuaciones aumenta el tiempo total de cálculo.

El procedimiento de sustitución hacia atrás comienza con la última ecuación. Se obtiene la solución de x_N en la última ecuación:

$$x_N = y^{(N-1)}_N / a^{(N-1)}_{N,N}$$

Sucesivamente,

$$\begin{aligned} x_{N-1} &= [y^{(N-2)}_{N-1} - a^{(N-1)}_{N-1,N}x_N]/a^{(N-2)}_{N-1,N-1} \\ &\vdots \\ x_1 &= \left[y_1 - \sum_{j=2}^{N} a_{1,j}x_j\right] \Big/ a_{1,1} \end{aligned} \tag{6.2.4}$$

Con esto se completa la eliminación de Gauss.

La eliminación de Gauss se puede realizar escribiendo sólo los coeficientes y los lados derechos en una forma de arreglo. De hecho, esto es precisamente lo que hace un programa de computadora. Incluso para los cálculos a mano, es más conveniente utilizar el arreglo que escribir todas las ecuaciones. La expresión en forma de arreglo de la ecuación (6.2.1) es

$$\begin{matrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,N-1} & a_{1,N} & y_1 \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots & a_{2,N-1} & a_{2,N} & y_2 \\ \vdots & & & & & & \\ a_{N,1} & a_{N,2} & a_{N,3} & \cdots & a_{N,N-1} & a_{N,N} & y_N \end{matrix} \tag{6.2.5}$$

Todas las etapas intermedias de la eliminación hacia adelante se escriben en forma de arreglo. El arreglo después de la eliminación hacia adelante queda como

$$\begin{array}{ccccccc} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,N-1} & a_{1,N} & y_1 \\ 0 & a'_{2,2} & a'_{2,3} & \cdots & a'_{2,N-1} & a'_{2,N} & y'_2 \\ 0 & 0 & a''_{3,3} & \cdots & a''_{3,N-1} & a''_{3,N} & y''_3 \\ \vdots & & & & & & \\ 0 & 0 & 0 & \cdots & a^{(N-2)}_{N-1,N-1} & a^{(N-2)}_{N-1,N} & y^{(N-2)}_{N-1} \\ 0 & 0 & 0 & \cdots & 0 & a^{(N-1)}_{N,N} & y^{(N-1)}_{N} \end{array} \tag{6.2.6}$$

Ejemplo 6.1

Resuelva las siguientes ecuaciones lineales en forma de arreglo:

$$\begin{aligned} 2x_1 + x_2 - 3x_3 &= -1 \\ -x_1 + 3x_2 + 2x_3 &= 12 \\ 3x_1 + x_2 - 3x_3 &= 0 \end{aligned} \tag{A}$$

(Solución)

La expresión en arreglo de las ecuaciones es

$$\begin{array}{rrrr} 2 & 1 & -3 & -1 \\ -1 & 3 & 2 & 12 \\ 3 & 1 & -3 & 0 \end{array} \tag{B}$$

En el arreglo B, las primeras tres columnas son los coeficientes de la ecuación (A) y la última columna representa los términos libres.

Para comenzar la eliminación hacia adelante, se resta el primer renglón multiplicado por − 1/2 del segundo renglón. El primer renglón, multiplicado por 3/2 se le resta al tercero. El arreglo queda como

$$\begin{array}{rrrr} 2 & 1 & -3 & -1 \\ 0 & 7/2 & 1/2 & 23/2 \\ 0 & -1/2 & 3/2 & 3/2 \end{array} \tag{C}$$

Continuamos con la eliminación hacia adelante y restamos el segundo renglón multiplicado por − 1/7 del tercer renglón:

$$\begin{array}{rrrr} 2 & 1 & -3 & -1 \\ 0 & 7/2 & 1/2 & 23/2 \\ 0 & 0 & 11/7 & 22/7 \end{array} \tag{D}$$

Esto concluye la eliminación hacia adelante.

La sustitución hacia atrás comienza con el último renglón. Interpretamos el último renglón como

$$(11/7)x_3 = 22/7$$

y obtenemos

$$x_3 = 2$$

Análogamente,

$$x_2 = 3$$

y

$$x_1 = 1$$

La eliminación de Gauss-Jordan es una variante de la eliminación de Gauss y comparte con ésta el proceso de eliminación hacia adelante, pero difiere en el proceso hacia atrás, el cual recibe el nombre de *eliminación hacia atrás.*

Partimos de la ecuación (6.2.6), la eliminación hacia atrás convierte en 1 a los coeficientes en la posición de pivoteo y elimina los demás. Primero se divide el último renglón entre $a^{(N-1)}_{N,N}$ para obtener

$$0 \quad 0 \quad 0 \quad \cdots \quad 1 \quad \bar{y}_N \tag{6.2.7}$$

donde

$\bar{y}_N = y_N^{(N-1)}/a_{N,N}^{(N-1)}$

Los N-ésimos coeficientes de cada renglón, excepto el último se eliminan restando el último renglón —ecuación (6.2.7)— multiplicado por el N-ésimo coeficiente al i-ésimo renglón:

$$\begin{array}{cccccc} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,N-1} & 0 \quad \bar{y}_1 \\ 0 & a'_{2,2} & a'_{2,3} & \cdots & a'_{2,N-1} & 0 \quad \bar{y}_2 \\ 0 & 0 & a''_{3,3} & \cdots & a''_{3,N-1} & 0 \quad \bar{y}_3 \\ \vdots & & & & & \\ 0 & 0 & 0 & \cdots & a^{(N-2)}_{N-1,N-1} & 0 \quad \bar{y}_{N-1} \\ 0 & 0 & 0 & \cdots & 0 & 1 \quad \bar{y}_N \end{array} \tag{6.2.8}$$

donde

$\bar{y}_i = y_i^{(i-1)} - a_{i,N}^{(i-1)}\bar{y}_N$

La ecuación (6.2.8) tiene la misma configuración de la ecuación (6.2.6) excepto por el último renglón y la N-ésima columna. Por lo tanto, el $(N-1)$-ésimo renglón se puede normalizar y se puede eliminar la $(N-1)$-ésima columna, siguiendo un procedimiento análogo. Dividimos el $(N-1)$-ésimo renglón entre $a^{(N-2)}_{N-1,N-1}$.

Entonces, los $(N-1)$-ésimos coeficientes de todos los renglones arriba del $(N-1)$-ésimo

renglón se eliminan restando el $(N-1)$-ésimo renglón multiplicado por el $(N-1)$-ésimo coeficiente al renglón que se eliminará:

$$\begin{matrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & 0 & 0 & \bar{y}'_1 \\ 0 & a'_{2,2} & a'_{2,3} & \cdots & 0 & 0 & \bar{y}'_2 \\ 0 & 0 & a''_{3,3} & \cdots & 0 & 0 & \bar{y}'_3 \\ \vdots & & & & & & \\ 0 & 0 & 0 & \cdots & 1 & 0 & \bar{y}'_{N-1} \\ 0 & 0 & 0 & \cdots & 0 & 1 & \bar{y}_N \end{matrix} \tag{6.2.9}$$

Al repetir el proceso de eliminación, el arreglo queda finalmente

$$\begin{matrix} 1 & 0 & 0 & \cdots & 0 & 0 & \bar{y}_1^{(N-1)} \\ 0 & 1 & 0 & \cdots & 0 & 0 & \bar{y}_2^{(N-2)} \\ 0 & 0 & 1 & \cdots & 0 & 0 & \bar{y}_3^{(N-3)} \\ \vdots & & & & & & \\ 0 & 0 & 0 & \cdots & 1 & 0 & \bar{y}'_{N-1} \\ 0 & 0 & 0 & \cdots & 0 & 1 & \bar{y}_N \end{matrix} \tag{6.2.10}$$

Esta es la conclusión de la eliminación hacia atrás. Deben observarse dos aspectos de la ecuación (6.2.10). En primer lugar, todos los coeficientes son iguales a cero, excepto los pivotes, que valen 1. En segundo lugar, puesto que la ecuación (6.2.10) es una forma de arreglo de un conjunto de ecuaciones, cada renglón se interpreta como

$$x_i = \bar{y}_i^{(N-i)}$$

es decir, la columna de la extrema derecha es la solución final.

Ejemplo 6.2

Resuelva el mismo problema del ejemplo 6.1 mediante la eliminación de Gauss-Jordan.

(Solución)

La eliminación hacia adelante de la eliminación de Gauss-Jordan es idéntica a la eliminación de Gauss, por lo que comenzaremos con la eliminación hacia atrás, partiendo de la ecuación (D) del ejemplo 6.1.

El tercer renglón de la ecuación (D) en el ejemplo 6.1 se divide entre 11/7. Se resta el tercer renglón, multiplicado por 1/2 del segundo renglón y se resta el tercer renglón multiplicado por -3 del primero:

$$\begin{matrix} 2 & 1 & 0 & 5 \\ 0 & 7 & 0 & 21 \\ 0 & 0 & 1 & 2 \end{matrix} \tag{E}$$

El segundo renglón se divide entre 7:

$$\begin{matrix} 2 & 1 & 0 & 5 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 2 \end{matrix} \qquad \text{(F)}$$

Se resta la segunda ecuación, multiplicada por 1, del primer renglón:

$$\begin{matrix} 2 & 0 & 0 & 2 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 2 \end{matrix} \qquad \text{(G)}$$

Finalmente, se divide el primer renglón entre 2 para completar la solución:

$$\begin{matrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 2 \end{matrix} \qquad \text{(H)}$$

Podemos ver que la última columna es la solución [compare con la ecuación (6.3.4)] y que las primeras tres columnas son iguales a cero excepto por la unidad en cada posición diagonal. El proceso explicado mediante este arreglo se puede extender a un conjunto de ecuaciones lineales de cualquier tamaño.

Al utilizar las eliminaciones de Gauss y Gauss-Jordan en los cálculos a mano, es útil escribir los productos de un renglón y una constante, como se muestra en el siguiente ejemplo.

Ejemplo 6.3

Resuelva las siguientes ecuaciones mediante la eliminación de Gauss con un cálculo manual:

$$\begin{aligned} -0.04x_1 + 0.04x_2 + 0.12x_3 &= 3 \\ 0.56x_1 - 1.56x_2 + 0.32x_3 &= 1 \\ -0.24x_1 + 1.24x_2 - 0.28x_3 &= 0 \end{aligned}$$

(Solución)

El arreglo del problema es

$$\begin{matrix} -0.04 & 0.04 & 0.12 & 3 \\ 0.56 & -1.56 & 0.32 & 1 \\ -0.24 & 1.24 & -0.28 & 0 \end{matrix}$$

La eliminación hacia adelante es como sigue:

renglón 1	−0.04	0.04	0.12	3	
renglón 2	0.56	−1.56	0.32	1	
renglón 3	−0.24	1.24	−0.28	0	
renglón 1 por 0.56/(0.04) = 14 :	−0.56	0.56	1.68	42	(A)
renglón 1 por 0.24/(0.04) = 6 :	−0.24	0.24	0.72	18	(B)

Sumamos (A) al renglón 2, restamos (B) del renglón 3 para obtener

renglón 1	−0.04	0.04	0.12	3
renglón 2	0	−1	2	43
renglón 3	0	+1	−1	−18

El segundo coeficiente del tercer renglón se elimina sumando el renglón 2 con el renglón 3:

renglón 1	−0.04	0.04	0.12	3
renglón 2	0	−1	2	43
renglón 3	0	0	1	25

Las sustituciones hacia atrás de la eliminación de Gauss son directas:

$$x_3 = 25/(1) = 25$$
$$x_2 = [43 - (2)(25)]/(-1) = 7$$
$$x_1 = [3 - (0.12)(25) - (0.04)(7)]/(-0.04) = 7$$

Comentario: siempre que sean necesarios los múltiplos de un renglón en los cálculos a mano, no dude en escribirlos como se muestra en (A) y (B).

En la eliminación de Gauss-Jordan, no es necesario separar las eliminaciones hacia adelante y hacia atrás. Esto es posible ya que un pivote se puede usar para eliminar a la vez no sólo los coeficientes debajo de él sino también los de arriba. Si se sigue este enfoque, la forma de los coeficientes es diagonal cuando se termina la eliminación para el último pivote.

Resumen de Esta Sección

a) La eliminación de Gauss consiste en la eliminación hacia adelante y la sustitución hacia atrás. La primera se lleva a cabo utilizando un arreglo formado por los coeficientes y los términos libres.
b) La eliminación hacia adelante de la eliminación de Gauss-Jordan es idéntica a la de la eliminación de Gauss. Sin embargo, la eliminación de Gauss-Jordan utiliza la eliminación hacia atrás en vez de la sustitución hacia atrás.

6.3 PIVOTEO Y ELIMINACION CANONICA DE GAUSS

En la sección 6.2, la eliminación de Gauss se aplica a un problema ideal sencillo con coeficientes no nulos. Sin embargo, el método no funciona si el primer coeficiente del primer renglón es igual a cero o si un coeficiente de la diagonal se anula en el proceso de solución, ya que se usan como denominadores en la eliminación hacia adelante.*

* En la ecuación (6.2.1), los coeficientes de la diagonal —o pivotes— son los coeficientes de x_1 en la primera ecuación, el de x_2 en la segunda y el de x_3 en la tercera ecuación. En general, el coeficiente de x_n en la n-ésima ecuación es el coeficiente en la diagonal.

El pivoteo se usa para cambiar el orden secuencial de las ecuaciones con dos propósitos: primero, para evitar que los coeficientes de la diagonal se anulen, y segundo, para hacer que cada coeficiente de la diagonal tenga magnitud mayor que cualquiera de los coeficientes por debajo de él. Las ecuaciones no se afectan matemáticamente por cambios en el orden secuencial, pero el cambio de orden hace posible el cálculo cuando el coeficiente de la diagonal se anula. Aun cuando todos los coeficientes de la diagonal fueran nulos, los cambios de orden aumentan la exactitud de los cálculos.

El pivoteo que se explica en el resto de esta sección es más adecuado para los programas de computadora que para los cálculos a mano, ya que éste tiende a incrementar sustancialmente la cantidad de esfuerzo. Por lo tanto, en los cálculos rápidos a mano de problemas que supuestamente se comportan bien y que se encuentran en algunas situaciones como las de los exámenes, podría evitarse que los estudiantes hicieran el pivoteo, excepto en el caso en que los coeficientes de la diagonal se anularan (a menos que el maestro indique otra cosa).

Para explicar el pivoteo, consideremos el arreglo

$$\begin{matrix} 0 & 10 & 1 & 2 \\ 1 & 3 & -1 & 6 \\ 2 & 4 & 1 & 5 \end{matrix} \tag{6.3.1}$$

No se puede eliminar el primer número de los renglones segundo y tercero debido a que el primer número del primer renglón es igual a cero. En nuestro primer pivoteo, se intercambian el primer y último renglones. Algunos podrían intercambiarse el primero y el segundo renglón, en vez de esto, se lleva el tercer renglón a la parte de arriba debido a que el primer número del tercer renglón tiene el mayor módulo (valor absoluto) de la primera columna. El hecho de llevar el número más grande de la columna a la posición diagonal tiene la ventaja de reducir el error de redondeo. Después de este pivoteo, el arreglo queda como

$$\begin{matrix} 2 & 4 & 1 & 5 \\ 1 & 3 & -1 & 6 \\ 0 & 10 & 1 & 2 \end{matrix} \tag{6.3.2}$$

A continuación, eliminamos el primer número del segundo renglón, restando a éste el primero, multiplicado por 1/2:

$$\begin{matrix} 2 & 4 & 1 & 5 \\ 0 & 1 & -3/2 & 7/2 \\ 0 & 10 & 1 & 2 \end{matrix} \tag{6.3.3}$$

El primer número del tercer renglón ya es igual a cero, por lo que procedemos a eliminar el segundo número, 10, del tercer renglón. Sin embargo, este número es mayor que el segundo número del segundo renglón (coeficiente de la diagonal). En general, como ya se ha mencionado, no es recomendable eliminar un número mayor

en magnitud que el término de la diagonal. Por lo tanto, intercambiamos el orden de los renglones segundo y tercero:

$$\begin{matrix} 2 & 4 & 1 & 5 \\ 0 & 10 & 1 & 2 \\ 0 & 1 & -3/2 & 7/2 \end{matrix} \tag{6.3.4}$$

Después eliminamos el segundo número del tercer renglón, con lo que el arreglo (6.3.4) se transforma en

$$\begin{matrix} 2 & 4 & 1 & 5 \\ 0 & 10 & 1 & 2 \\ 0 & 0 & -16/5 & 33/5 \end{matrix} \tag{6.3.5}$$

Los datos 2, 10 y $-16/5$ se llaman *coeficientes de la diagonal* o *pivotes*.

Las sustituciones hacia atrás dan como resultado

$$x_3 = -2.0625$$
$$x_2 = (2 - x_3)/10 = 0.4062$$
$$x_1 = (5 - 4x_2 - x_3) = 2.7187$$

Las eliminaciones de Gauss-Jordan también producen

$$\begin{matrix} 1 & 0 & 0 & 2.7187 \\ 0 & 1 & 0 & 0.4062 \\ 0 & 0 & 1 & -2.0625 \end{matrix}$$

Para simplificar la explicación, el ejemplo de problema considerado aquí es del tipo bien condicionado o bien planteado, por lo que la precisión no se ve afectada por el pivoteo.)

Si a pesar del pivoteo, es inevitable un elemento nulo en la diagonal, esto indica que el problema es de los que no tienen solución única. Si así ocurre, detenemos el esfuerzo del cálculo.

Sin embargo, el pivoteo no puede sortear todas las dificultades relacionadas con la solución de las ecuaciones lineales. Si los resultados siguen sin ser precisos aun con el pivoteo, se debe utilizar la doble precisión. En general, las ecuaciones lineales asociadas con problemas de ecuaciones diferenciales con valores en la frontera están bien condicionados y pocas veces tienen problemas con la precisión simple. Por otro lado, las ecuaciones lineales asociadas con el ajuste de curvas con base en los mínimos cuadrados, a menudo están mal condicionados, por lo que requieren una alta precisión. Los problemas mal condicionados se analizarán con más detalle en la sección 6.9.

El PROGRAMA 6-1 ejecuta la eliminación de Gauss con pivoteo.

El ejemplo 6.4 muestra el efecto del pivoteo en un conjunto típico de ecuaciones lineales.

Ejemplo 6.4

La solución exacta del siguiente problema en forma de arreglo es aquella en donde todas las soluciones valen 1, debido a que los términos libres son la suma de los coeficientes del mismo renglón:

1.334E−4	4.123E+1	7.912E+2	−1.544E+3	−711.5698662
1.777	2.367E−5	2.070E+1	−9.035E+1	−67.87297633
9.188	0	−1.015E+1	1.988E−4	−0.961801200
1.002E+2	1.442E+4	−7.014E+2	5.321	13824.12100

a) Resuelva las ecuaciones sin povoteo y después con pivoteo, usando la precisión simple.

b) Repita el problema resolviéndolo con doble precisión.

(Solución)

La solución con pivoteo se puede obtener ejecutando el PROGRAMA 6-1. La solución sin pivoteo también se obtiene con el mismo programa, pero quitando unas cuantas líneas que se refieren al pivoteo. Los resultados son los siguientes:

a) Precisión simple:

i	Sin pivoteo[a] x_i	Con pivoteo[a] x_i
1	0.95506	0.99998
2	1.00816	1
3	0.96741	1
4	0.98352	1

Los resultados en precisión simple sin pivoteo son muy desalentadores, pero el pivoteo mejora la exactitud en forma significativa.

b) Doble precisión:

i	Sin pivoteo[a] x_i	Con pivoteo[a] x_i
1	0.9999 9999 9861 473	1.0000 0000 0000 002
2	1.0000 0000 0000 784	1.0000 0000 0000 000
3	0.9999 9999 9984 678	1.0000 0000 0000 000
4	0.9999 9999 9921 696	1.0000 0000 0000 000

[a] Estos cálculos se llevaron a cabo en una VAX, cuya precisión es casi la misma que la de la PC de IBM y las *mainframe* de IBM. La precisión simple de CDC y Cray es aproximadamente del doble de VAX, IBM PC y de la mainframe de IBM. Por lo tanto, si se utilizan CDC o Cray con precisión simple en este problema, los resultados serán equivalentes a los que se muestran aquí con doble precisión. Véase el capítulo 1 para una comparación de varias computadoras.

La doble precisión mejora la exactitud, incluso sin pivoteo. Pero con el pivoteo, aquella aumenta todavía más.

Resumen de Esta Sección

a) El pivoteo tiene dos finalidades: una, superar la dificultad que presentan los coeficientes nulos en la diagonal; la otra, hacer decrecer los errores de redondeo.
b) El pivoteo se puede evitar para los cálculos rápidos a mano, excepto cuando los coeficientes de la diagonal se anulan (siempre y cuando el problema sea un ejercicio bien planteado).
c) Para la mejor precisión, se recomienda el uso de la doble precisión y pivoteo.

6.4 PROBLEMAS SIN SOLUCION UNICA

No siempre es posible resolver un conjunto de ecuaciones lineales en forma numérica. Los siguientes conjuntos de ecuaciones lineales son tres ejemplos sencillos pero importantes.

a) $$\begin{aligned} -x + y &= 1 \\ -2x + 2y &= 2 \end{aligned}$$

b) $$\begin{aligned} -x + y &= 1 \\ -x + y &= 0 \end{aligned}$$

c) $$\begin{aligned} -x + y &= 1 \\ x + 2y &= -2 \\ 2x - y &= 0 \end{aligned}$$

Las ecuaciones de cada conjunto se grafican en la figura 6.1.

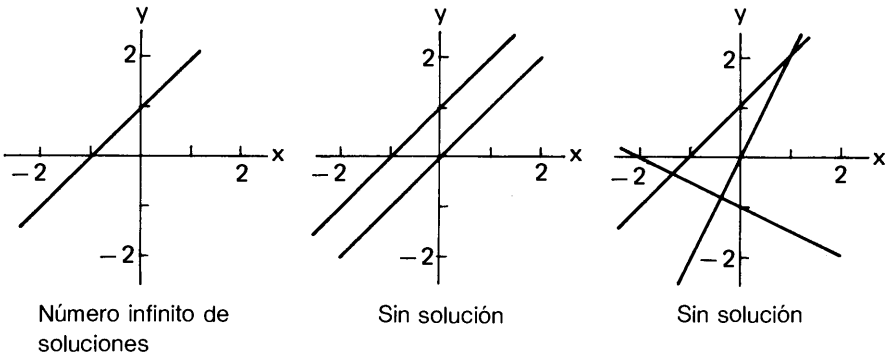


Figura 6.1 Gráfica de tres conjuntos de ecuaciones lineales

En el conjunto a) la segunda ecuación es el doble de la primera ecuación, por lo que matemáticamente son idénticas. Cualquier punto (x, y) que satisfaga una de las ecuaciones, también es solución de la otra. Por lo tanto, el número de soluciones es infinito. En otras palabras, no existe una solución única. Si una ecuación es múltiplo de otra, o se puede obtener sumando o restando otras ecuaciones, se dice que esa ecuación es *linealmente dependiente* de las otras. Si ninguna de las ecuaciones es li-

nealmente dependiente, se dice que todas las ecuaciones son *linealmente independientes.*

En el conjunto b), las dos ecuaciones son rectas paralelas que no se intersecan, por lo que no existe solución. Tal sistema recibe el nombre de *sistema inconsistente.* Un conjunto de ecuaciones es inconsistente si el lado izquierdo de al menos una de las ecuaciones se puede eliminar totalmente (sumando o restando otras ecuaciones), mientras que el lado derecho permanece distinto de cero.

En el tercer conjunto, existen tres ecuaciones independientes con dos incógnitas. Como se puede ver en la figura 6.1 c), las tres ecuaciones no se pueden satisfacer simultáneamente.

Un caso como el de c) no puede ocurrir si el número de ecuaciones es igual al número de incógnitas. En este caso, sólo puede ocurrir la falta de independencia lineal —como en a)— o la inconsistencia, como en b). Si el número de ecuaciones es mayor de dos, la carencia de independencia lineal o la inconsistencia son menos obvias. Sin embargo, un programa que ejecute la eliminación de Gauss y que intente llevarla a cabo en uno de tales conjuntos se detiene a mitad de los cálculos debido a un error aritmético, como un desbordamiento o una división entre cero. De hecho, si un conjunto de ecuaciones es inconsistente o linealmente dependiente, uno de los renglones de coeficientes en el arreglo (sin incluir el último número correspondiente al término del lado derecho) se anula durante la eliminación hacia adelante. En el PROGRAMA 6-1, se detectan tales casos y el programa se detiene después de imprimir "*la matriz es singular*".

Resumen de Esta Sección

Las condiciones necesarias para la existencia de una solución única son las siguientes:

a) El número de ecuaciones debe ser igual al número de incógnitas.
b) Cada ecuación es linealmente independiente; o en forma equivalente, ninguna ecuación se puede eliminar sumando o restando otras ecuaciones.

6.5 MATRICES Y VECTORES

Esta sección presenta las operaciones con matrices y vectores.

Una *matriz* es un arreglo rectangular de números, tal como los ya vistos en la sección anterior. Cuando el arreglo es cuadrado, se llama *matriz cuadrada.* Las siguientes son matrices cuadradas:

$$A = \begin{bmatrix} 2 & 1 & 0 \\ -1 & 3 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 3 & -1 & 1 \\ 2 & 1 & 3 \\ -2 & 4 & 5 \end{bmatrix}$$

A menudo, las matrices se escriben en forma simbólica como sigue:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{6.5.1}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \tag{6.5.2}$$

Note que en las ecuaciones (6.5.1) y (6.5.2), el primer subíndice de una matriz cambia en la dirección vertical y el segundo en la dirección horizontal. Las ecuaciones (6.5.1) y (6.5.2) se abrevian a menudo como

$$A = [a_{ij}] \quad \text{y} \quad B = [b_{ij}]$$

respectivamente.

Si una matriz es rectangular con m renglones y n columnas, decimos que es una *matriz de* $m \times n$. Por ejemplo,

$$A = \begin{bmatrix} 5 & 9 & 2 \\ 3 & 0 & 4 \end{bmatrix}$$

es una matriz de 2×3.

Las matrices tienen cuatro operaciones básicas análogas a las de los números: suma, resta, multiplicación y división. De éstas, las dos primeras son directas, pero las últimas dos son un poco más complejas. La suma, resta y multiplicación se definen como sigue:

Suma

$$A + B = C$$

donde $C = [c_{ij}]$ es una matriz en la que cada elemento está dado por

$$c_{ij} = a_{ij} + b_{ij}$$

Resta

$$A - B = C$$

donde $C = [c_{ij}]$, con

$$c_{ij} = a_{ij} - b_{ij}$$

Multiplicación

$$AB = C$$

donde $C = [c_{ij}]$ con

$$c_{ij} = \sum_{k=1}^{N} a_{ik} b_{kj}$$

Como se puede ver fácilmente, en general el producto AB no es igual a BA. Si $AB = BA$, se dice que las matrices A y B *conmutan*. Si A y B son matrices rectangulares, el producto sólo existe si A es una matriz de $m \times n$ y B es una matriz de $n \times k$ (el número de columnas de A es igual al número de renglones de B).

La división de una matriz entre otra se define de la manera siguiente:

División

$$B^{-1}A = C$$

donde A se divide entre B y B_{-1} es la llamada inversa de B. La división es equivalente a

$$A = BC$$

La división es mucho más restrictiva que las demás operaciones puesto que B^{-1} sólo puede existir si B es una matriz cuadrada. En la sección 6.6 se dan más detalles de las matrices inversas.

Un vector columna es un arreglo de columna de números o variables, por ejemplo:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Si un vector columna está formado por N números (o elementos), se dice que el orden del vector es N. Un vector columna también se puede pensar como una matriz de $N \times 1$. Un vector renglón es un arreglo en renglón de números, por ejemplo:

$$a = [a_1, a_2, a_3, a_4]$$

Un vector renglón se considera como una matriz de $1 \times N$. Cuando se usa la palabra "vector" sin especificar si es "columna" o "renglón", usualmente quiere decir un vector columna. Puesto que los vectores son casos especiales de matrices, todas las reglas de las matrices se aplican a los vectores.

La suma de vectores se define como

$$x + y = z$$

donde x, y y z son vectores del mismo orden; los i-ésimos elementos de los vectores guardan la relación

$$x_i + y_i = z_i$$

La resta efectuada de un vector a otro es

$$x - y = z$$

donde

$$x_i - y_i = z_i$$

La multiplicación de una matriz y un vector se define como

$$Ax = y$$

donde x y y son vectores y A es una matriz. En la ecuación anterior, las y_i se escriben en forma explícita como

$$y_i = \sum_{k=1}^{N} a_{ik} x_k$$

Ejemplo 6.5

Se definen las siguientes matrices cuadradas y vectores

$$A = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 1 & 2 \\ 4 & 1 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 7 & 3 & 1 \\ 2 & 3 & 5 \\ 8 & 1 & 6 \end{bmatrix}$$

$$x = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix}, \quad y = \begin{bmatrix} 3 \\ 9 \\ 4 \end{bmatrix}$$

Calcule $A + B$, $B - A$, AB, BA, $x + y$, $x - y$ y Ax.

(Solución)

Los cálculos se muestran a continuación:

$$A + B = \begin{bmatrix} 1+7 & 2+3 & 4+1 \\ 3+2 & 1+3 & 2+5 \\ 4+8 & 1+1 & 3+6 \end{bmatrix} = \begin{bmatrix} 8 & 5 & 5 \\ 5 & 4 & 7 \\ 12 & 2 & 9 \end{bmatrix}$$

$$A - B = \begin{bmatrix} 1-7 & 2-3 & 4-1 \\ 3-2 & 1-3 & 2-5 \\ 4-8 & 1-1 & 3-6 \end{bmatrix} = \begin{bmatrix} -6 & -1 & 3 \\ 1 & -2 & -3 \\ -4 & 0 & -3 \end{bmatrix}$$

$$AB = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 1 & 2 \\ 4 & 1 & 3 \end{bmatrix} \begin{bmatrix} 7 & 3 & 1 \\ 2 & 3 & 5 \\ 8 & 1 & 6 \end{bmatrix}$$

$$= \begin{bmatrix} 1\times7+2\times2+4\times8 & 1\times3+2\times3+4\times1 & 1\times1+2\times5+4\times6 \\ 3\times7+1\times2+2\times8 & 3\times3+1\times3+2\times1 & 3\times1+1\times5+2\times6 \\ 4\times7+1\times2+3\times8 & 4\times3+1\times3+3\times1 & 4\times1+1\times5+3\times6 \end{bmatrix}$$

$$= \begin{bmatrix} 43 & 13 & 35 \\ 39 & 14 & 20 \\ 54 & 18 & 27 \end{bmatrix}$$

$$BA = \begin{bmatrix} 7 & 3 & 1 \\ 2 & 3 & 5 \\ 8 & 1 & 6 \end{bmatrix} \begin{bmatrix} 1 & 2 & 4 \\ 3 & 1 & 2 \\ 4 & 1 & 3 \end{bmatrix} = \begin{bmatrix} 20 & 18 & 37 \\ 31 & 12 & 29 \\ 35 & 23 & 52 \end{bmatrix}$$

$$x + y = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix} + \begin{bmatrix} 3 \\ 9 \\ 4 \end{bmatrix} = \begin{bmatrix} 4 \\ 13 \\ 6 \end{bmatrix}$$

$$x + y = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix} - \begin{bmatrix} 3 \\ 9 \\ 4 \end{bmatrix} = \begin{bmatrix} -2 \\ -5 \\ -2 \end{bmatrix}$$

$$AX = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 1 & 2 \\ 4 & 1 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix} = \begin{bmatrix} 1\times1+2\times4+4\times2 \\ 3\times1+1\times4+2\times2 \\ 4\times1+1\times4+3\times2 \end{bmatrix} = \begin{bmatrix} 17 \\ 11 \\ 11 \end{bmatrix}$$

Comentario: observe que $AB \neq BA$.

Ejemplo 6.6

Calcule los siguientes productos:

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 5 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 & 7 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix}$$

$$\begin{bmatrix} 8 & 1 & 3 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 8 & 1 & 3 \\ 1 & 5 & 2 \end{bmatrix}$$

(Solución)

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 5 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \times 5 + 2 \times 1 \\ 4 \times 5 + 3 \times 1 \\ 0 \times 5 + 2 \times 1 \end{bmatrix}$$

$$[2 \quad 1 \quad 7] \begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix} = [2 \times 1 + 1 \times 4 + 7 \times 0 \quad 2 \times 2 + 1 \times 3 + 7 \times 2] = [6 \quad 21]$$

$$\begin{bmatrix} 8 & 1 & 3 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 8 \times 1 + 1 \times 4 + 3 \times 0 & 8 \times 2 + 1 \times 3 + 3 \times 2 \\ 1 \times 1 + 5 \times 4 + 2 \times 0 & 1 \times 2 + 5 \times 3 + 2 \times 2 \end{bmatrix}$$

$$= \begin{bmatrix} 12 & 25 \\ 21 & 21 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 8 & 1 & 3 \\ 1 & 5 & 2 \end{bmatrix} = \begin{bmatrix} 1 \times 8 + 2 \times 1 & 1 \times 1 + 2 \times 5 & 1 \times 3 + 2 \times 2 \\ 4 \times 8 + 3 \times 1 & 4 \times 1 + 3 \times 5 & 4 \times 3 + 3 \times 2 \\ 0 \times 8 + 2 \times 1 & 0 \times 1 + 2 \times 5 & 0 \times 3 + 2 \times 2 \end{bmatrix}$$

$$= \begin{bmatrix} 10 & 11 & 7 \\ 35 & 19 & 18 \\ 2 & 10 & 4 \end{bmatrix}$$

A continuación se definen algunas matrices y vectores especiales:

Matriz nula. Todos los elementos de la matriz nula son iguales a cero:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Matriz identidad. Todos los elementos son iguales a cero excepto los elementos de la diagonal, que son iguales a uno. Una matriz identidad se denota por I, es decir,

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matriz transpuesta. Para una matriz definida por $A = [a_{ij}]$, su transpuesta se define como $A^t = [a_{ij}]$ (se intercambian i y j). Por ejemplo:

$$A = \begin{bmatrix} 2 & 3 \\ 0 & 5 \end{bmatrix} \text{entonces} A^t = \begin{bmatrix} 2 & 0 \\ 3 & 5 \end{bmatrix}$$

$$B = \begin{bmatrix} 5 & 0 \\ 2 & 7 \\ 1 & 2 \end{bmatrix} \text{entonces} B^t = \begin{bmatrix} 5 & 2 & 1 \\ 0 & 7 & 2 \end{bmatrix}$$

Matriz inversa. La inversa de una matriz cuadrada A se escribe como A^{-1} y satisface $AA^{-1} = A^{-1}A = I$. La explicación de cómo se calcula A^{-1} se pospone hasta la sección 6.6.

Matriz ortogonal. Una matriz que tiene columnas ortonormales. Satisface

$$Q^tQ = I, \quad QQ^t = I, \quad \text{y} \quad Q^t = Q^{-1}$$

Vector nulo. Todos los elementos del vector nulo son iguales a cero:

$$x = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Vectores unitarios. Todos los elementos son iguales a cero excepto uno que vale 1:

$$u = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad v = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad w = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Vector transpuesto. Si un vector está dado por

$$v = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

entonces su transpuesto se escribe como v^t y se define como

$$v^t = [x_1 \quad x_2 \quad x_3]$$

El transpuesto de un vector columna es un vector renglón.

Resumen de Esta Sección

a) Un vector columna es una matriz con una única columna, un vector renglón es una matriz con un único renglón y se puede expresar como el transpuesto de un vector columna.

b) Se puede sumar o restar dos matrices con el mismo número de columnas y renglones.
c) Una matriz B puede multiplicar a la izquierda (premultiplicarse) a otra matriz A si el número de columnas de A es igual al número de renglones de B.
d) Si $BA = I$ o $AB = I$, donde I es una matriz identidad, entonces $B = A^{-1}$.

6.6 INVERSION DE UNA MATRIZ

Se puede calcular la inversa de una matriz aplicando la eliminación de Gauss-Jordan. Consideremos una ecuación lineal escrita en notación matricial:

$$Ax = y \tag{6.6.1}$$

donde A es una matriz cuadrada. Supongamos que el pivoteo no es necesario; al multiplicar a la izquierda (premultiplicar) la ecuación (6.6.1) por una matriz cuadrada G se obtiene

$$GAx = Gy \tag{6.6.2}$$

Si escogiéramos a G como la inversa de A, es decir, A^{-1}, la ecuación (6.6.2) se reduciría a

$$x = A^{-1}y \tag{6.6.3}$$

que es la solución. En otras palabras, la eliminación de Gauss-Jordan es equivalente a la multiplicación a la izquierda por $G = A^{-1}$.

Por lo tanto, si aplicamos las mismas operaciones que en la eliminación de Gauss-Jordan a la matriz identidad (es decir, multiplicar los renglones por los mismos números utilizados en la eliminación de Gauss-Jordan y restar los renglones en la misma forma), entonces la matriz identidad debe transformarse en A^{-1}. Esto se puede escribir de manera simbólica como

$$GI = A^{-1} \tag{6.6.4}$$

Para calcular A^{-1}, escribimos A e I es una forma aumentada de arreglo

$$\begin{matrix} a_{1,1} & a_{1,2} & a_{1,3} & 1 & 0 & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & 0 & 1 & 0 \\ a_{3,1} & a_{3,2} & a_{3,3} & 0 & 0 & 1 \end{matrix} \tag{6.6.5}$$

Entonces seguimos la eliminación de Gauss-Jordan exactamente en la misma forma que al resolver un conjunto de ecuaciones lineales. Cuando la mitad izquierda de la matriz aumentada se reduce a una matriz identidad, la mitad derecha se convierte en A^{-1}.

Ejemplo 6.7

Calcule la inversa de la matriz

$$A = \begin{bmatrix} 2 & 1 & -3 \\ -1 & 3 & 2 \\ 3 & 1 & -3 \end{bmatrix} \qquad \text{(A)}$$

(Solución)

Escribimos a A e I en un arreglo:

$$\begin{matrix} 2 & 1 & -3 & 1 & 0 & 0 \\ -1 & 3 & 2 & 0 & 1 & 0 \\ 3 & 1 & -3 & 0 & 0 & 1 \end{matrix}$$

El procedimiento de eliminación que se muestra a continuación es esencialmente el mismo que la eliminación de Gauss-Jordan descrito en la sección 6.2.

La eliminación hacia adelante se hace como sigue. Se resta el primer renglón —multiplicado por —1/2— al segundo renglón; luego, se resta el primer renglón —multiplicado por 3/2— al tercero:

$$\begin{matrix} 2 & 1 & -3 & 1 & 0 & 0 \\ 0 & 3.5 & 0.5 & 0.5 & 1 & 0 \\ 0 & -0.5 & 1.5 & -1.5 & 0 & 1 \end{matrix}$$

Se resta el segundo renglón, multiplicado por $-0.5/3.5 = -1/7$ al tercero:

$$\begin{matrix} 2 & 1 & -3 & 1 & 0 & 0 \\ 0 & 3.5 & 0.5 & 0.5 & 1 & 0 \\ 0 & 0 & 1.5714 & -1.4285 & 0.14285 & 1 \end{matrix} \qquad \text{(B)}$$

Se continúa de la siguiente forma con la eliminación hacia atrás: el último renglón se divide entre 1.5714:

$$\begin{matrix} 2 & 1 & -3 & 1 & 0 & 0 \\ 0 & 3.5 & 0.5 & 0.5 & 1 & 0 \\ 0 & 0 & 1 & -0.90909 & 0.090909 & 0.63636 \end{matrix}$$

El segundo renglón se divide entre 3.5 y luego se resta del último renglón, multiplicado por 0.5/3.5:

$$\begin{matrix} 2 & 1 & -3 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0.27272 & 0.27272 & -0.09090 \\ 0 & 0 & 1 & -0.90909 & 0.09090 & 0.63636 \end{matrix}$$

El primer renglón se divide entre 2 y se resta —después de multiplicarlo por 1/2— al segundo renglón; el primer renglón también se resta al último, una vez que se haya multiplicado por 3/2.

$$\begin{matrix} 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0.27272 & 0.27272 & -0.09090 \\ 0 & 0 & 1 & -0.90909 & 0.09090 & 0.63636 \end{matrix}$$

Las últimas tres columnas del arreglo aumentado anterior constituyen la inversa de la matriz A. Esto se puede verificar multiplicando A a la izquierda o a la derecha por A^{-1} de la manera siguiente:

$$\begin{bmatrix} 2 & 1 & -3 \\ -1 & 3 & 2 \\ 3 & 1 & -3 \end{bmatrix}\begin{bmatrix} -1 & 0 & 1 \\ 0.27272 & 0.27272 & -0.09090 \\ -0.90909 & 0.09090 & 0.63636 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} -1 & 0 & 1 \\ 0.27272 & 0.27272 & -0.09090 \\ -0.90909 & 0.09090 & 0.63636 \end{bmatrix}\begin{bmatrix} 2 & 1 & -3 \\ -1 & 3 & 2 \\ 3 & 1 & -3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Aunque no utilizamos el pivoteo en la anterior explicación de la inversión de matrices, éste es necesario puesto que el esquema de inversión es, en esencia, una eliminación de Gauss. Por fortuna, la matriz inversa no se ve afectada por un cambio en el orden secuencial de las ecuaciones. La primera columna de A^{-1} es la solución de Ax col [1, 0, 0] y la segunda y tercera columnas son las soluciones de Ax = col [0, 1, 0] y Ax = col [0, 0, 1], respectivamente. El orden secuencial de los elementos en x no se ve afectado por mezclar el orden de las ecuaciones. Así, A^{-1} no se altera por el pivoteo.

Al calcular la inversa de una matriz mediante un cálculo a mano, se sugiere escribir los valores de un renglón al multiplicarlos por una constante, como se muestra en el siguiente ejemplo.

Ejemplo 6.8

Obtenga la inversa de la siguiente matriz mediante cálculos a mano con pivoteo:

$$\begin{matrix} -0.04 & 0.04 & 0.12 \\ 0.56 & -1.56 & 0.32 \\ -0.24 & 1.24 & -0.28 \end{matrix}$$

(Solución)

El arreglo aumentado es

$$\begin{matrix} -0.04 & 0.04 & 0.12 & 1 & 0 & 0 \\ 0.56 & -1.56 & 0.32 & 0 & 1 & 0 \\ -0.24 & 1.24 & -0.28 & 0 & 0 & 1 \end{matrix}$$

Se hace un primer pivoteo debido a que el elemento superior a la extrema izquierda es menor que el elemento inmediato inferior:

renglón 1	0.56	−1.56	0.32	0	1	0
renglón 2	−0.04	0.04	0.12	1	0	0
renglón 3	−0.24	1.24	−0.28	0	0	1

La eliminación hacia adelante se hace como sigue:

renglón 1 por 0.04/0.56	0.04	−0.114285	0.022857	0	0.071428	0	(A)
renglón 1 por 0.24/0.56	0.24	−0.668571	0.137142	0	0.428571	0	(B)

Sumamos (A) al primer renglón y (B) al tercero:

renglón 1	0.56	−1.56	0.32	0	1	0
renglón 2	0	−0.071428	0.142857	1	0.071428	0
renglón 3	0	0.571428	−0.142857	0	0.428571	1

Se intercambian el segundo y tercer renglón para el pivoteo:

renglón 1	0.56	−1.56	0.32	0	1	0
renglón 2	0	0.571428	−0.142857	0	0.428571	1
renglón 3	0	−0.071428	0.142857	1	0.071428	0

El segundo renglón multiplicado por 0.071428/0.571428 = 0.125 es:

	0	0.071428	−0.017857	0	0.053571	0.125(C)

Se suma (c) al renglón 3:

	0	0	0.124993	1	0.124993	0.125

Esto concluye con la eliminación hacia adelante. El arreglo queda:

renglón 1	0.56	−1.56	0.32	0	1	0
renglón 2	0	0.571428	−0.142857	0	0.428571	1
renglón 3	0	0	0.124993	1	0.124993	0.125

Para comenzar la eliminación hacia atrás, dividimos la última columna entre 0.124993:

renglón 1	0.56	−1.56	0.32	0	1	0
renglón 2	0	0.571428	−0.142857	0	0.428571	1
renglón 3	0	0	1	8	1	1

Sumamos, el tercer renglón (que se ha multiplicado por 0.142857) con el segundo y dividimos el resultado entre 0.571428:

renglón 1	0.56	−1.56	0.32	0	1	0
renglón 2	0	1	0	2	1	2
renglón 3	0	0	1	8	1	1

Sumamos el segundo renglón —multiplicado por 1.56— y restamos el tercer renglón (multiplicado por 0.32) al primero:

renglón 1	1	0	0	1	4	5
renglón 2	0	1	0	2	1	2
renglón 3	0	0	1	8	1	1

Así, la inversa es

$$\begin{bmatrix} 1 & 4 & 5 \\ 2 & 1 & 2 \\ 8 & 1 & 1 \end{bmatrix}$$

Hasta aquí hemos utilizado la eliminación de Gauss-Jordan para calcular la inversa de una matriz, pero también se puede hacer mediante la eliminación de Gauss. La razón es la siguiente: la eliminación de Gauss-Jordan de la matriz aumentada —para el caso de una matriz de orden N— se puede separar en soluciones de N conjuntos de ecuaciones lineales con la misma matriz de coeficientes A. El primer conjunto tiene el término no homogéneo igual a la primera columna de la matriz identidad, el segundo conjunto tiene el término no homogéneo igual a la segunda columna de la matriz identidad, etc. La solución del primer conjunto se convierte en la primera columna de la matriz inversa, la solución del segundo conjunto es la segunda columna, etc. En los cálculos reales, no hay que calcular cada conjunto por separado, sino en forma simultánea debido a que el procedimiento de cálculo es el mismo para todos los conjuntos. La cantidad de cálculos con la eliminación de Gauss es menor que con la eliminación de Gauss-Jordan, puesto que en la primera no hay que reducir el lado izquierdo de la matriz aumentada a una matriz identidad. Por esta razón, el PROGRAMA 6-2 utiliza la eliminación de Gauss en vez de la eliminación de Gauss-Jordan.

Resumen de Esta Sección

a) La inversa de una matriz se puede calcular aplicando la eliminación de Gauss-Jordan al arreglo aumentado que está formado por la matriz que se invertirá y la matriz identidad.
b) Una vez formado el arreglo aumentado, el pivoteo posterior no afecta el resultado de la eliminación de Gauss-Jordan.
c) La inversa de una matriz se puede calcular también mediante la eliminación de Gauss.

6.7 DESCOMPOSICION *LU*

El esquema de descomposición LU es una transformación de una matriz A como producto de dos matrices,

$$A = LU$$

donde L es una matriz triangular inferior y U es una matriz triangular superior. Cuando uno debe resolver varios conjuntos de ecuaciones lineales en los que todas las matrices de coeficientes son iguales pero los términos no homogéneos (lado derecho) son distintos, la solución de las ecuaciones utilizando la descomposición LU tiende a ser más eficiente que la eliminación de Gauss.

La descomposición LU para una matriz de 3×3 se ilustra de la manera siguiente:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{2,1} & 1 & 0 \\ l_{3,1} & l_{3,2} & 1 \end{bmatrix} \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} \\ 0 & u_{2,2} & u_{2,3} \\ 0 & 0 & u_{3,3} \end{bmatrix} \tag{6.7.1}$$

Conviene observar que los elementos de la diagonal de L valen 1.

Para evaluar $u_{i,j}$ y $l_{i,j}$ en la ecuación (6.7.1) sin pivoteo, primero multiplicamos el primer renglón de L por cada columna de U y comparamos el resultado con el primer renglón de A. Tenemos entonces que el primer renglón de U es idéntico al de A:

$$u_{1,j} = a_{1,j}, \quad j = 1 \text{ a } 3 \tag{6.7.2}$$

Multiplicamos el segundo y tercer renglones de L por la primera columna de U respectivamente, y lo comparamos con el lado izquierdo para obtener

$$a_{2,1} = l_{2,1}u_{1,1}, \quad a_{3,1} = l_{3,1}u_{1,1}$$

o en forma equivalente

$$l_{2,1} = a_{2,1}/u_{1,1}, \quad l_{3,1} = a_{3,1}/u_{1,1} \tag{6.7.3}$$

Multiplicamos el segundo renglón de L por la segunda y tercera columnas de U y las comparamos con el lado izquierdo para obtener

$$a_{2,2} = l_{2,1}u_{1,2} + u_{2,2}, \quad a_{2,3} = l_{2,1}u_{1,3} + u_{2,3}$$

o bien

$$u_{2,2} = a_{2,2} - l_{2,1}u_{1,2}, \quad u_{2,3} = a_{2,3} - l_{2,1}u_{1,3} \tag{6.7.4}$$

Multiplicamos el tercer renglón de L por la segunda columna de U y tenemos

$$a_{3,2} = l_{3,1}u_{1,2} + l_{3,2}u_{2,2}$$

o, en forma equivalente,

$$l_{3,2} = [a_{3,2} - l_{3,1}u_{1,2}]/u_{2,2} \tag{6.7.5}$$

Finalmente, $l_{3,3}$ se obtiene multiplicando la última columna de U por el último renglón de L y lo igualamos a $a_{3,3}$ como sigue

$$l_{3,1}u_{1,3} + l_{3,2}u_{2,3} + u_{3,3} = a_{3,3}$$

o bien

$$u_{3,3} = a_{3,3} - l_{3,1}u_{1,3} - l_{3,2}u_{2,3} \tag{6.7.6}$$

Ejemplo 6.9

Descomponga la siguiente matriz en matrices L y U:

$$A = \begin{bmatrix} 2 & 1 & -3 \\ -1 & 3 & 2 \\ 3 & 1 & -3 \end{bmatrix}$$

(Solución)

Seguimos el procedimiento de las ecuaciones (6.7.2) a la (6.7.6) para obtener:

$$u_{1,1} = 2, \quad u_{1,2} = 1, \quad u_{1,3} = -3$$
$$l_{2,1} = -0.5, \quad l_{3,1} = 1.5$$
$$u_{2,2} = 3 - (-0.5)(1) = 3.5$$
$$u_{2,3} = 2 - (-0.5)(-3) = 0.5$$
$$l_{3,2} = [1 - (1.5)(1)]/3.5 = -0.142857$$
$$u_{3,3} = -3 - (1.5)(-3) - (-0.142857)(-0.5) = 1.57142$$

Entonces,

$$L = \begin{bmatrix} 1 & 0 & 0 \\ -0.5 & 1 & 0 \\ 1.5 & -0.1428 & 1 \end{bmatrix} \quad U = \begin{bmatrix} 2 & 1 & -3 \\ 0 & 3.5 & 0.5 \\ 0 & 0 & 1.5714 \end{bmatrix}.$$

Los resultados anteriores se pueden verificar volviendo a sustituir en la ecuación (6.7.1).

El esquema general de la descomposición LU para una matriz de orden N es el siguiente:

a) El primer renglón de U, $u_{i,j}$ para $j = 1$ hasta N, se obtiene por medio de

$$u_{1,j} = a_{1,j}, \quad j = 1 \text{ hasta } N \tag{6.7.7}$$

b) La primera columna de L, l_{i1} para $i = 2$ hasta N, se obtiene por medio de

$$l_{i,1} = a_{i,1}/u_{1,1}, \quad i = 2 \text{ hasta } N \tag{6.7.8}$$

c) El segundo renglón de U se obtiene como

$$u_{2,j} = a_{2,j} - l_{2,1}u_{1,j}, \quad j = 2 \text{ hasta } N \tag{6.7.9}$$

d) La segunda columna de L se obtiene mediante

$$l_{i,2} = [a_{i,2} - l_{i,1}u_{1,2}]/u_{2,2}, \quad i = 3 \text{ hasta } N \tag{6.7.10}$$

e) El n-ésimo renglón de u se obtiene de

$$u_{n,j} = a_{n,j} - \sum_{k=1}^{n-1} l_{n,k}u_{k,j}, \quad j = n \text{ hasta } N \tag{6.7.11}$$

f) La n-ésima columna de L se obtiene de

$$l_{i,n} = [a_{i,n} - \sum_{k=1}^{n-1} l_{i,k}u_{k,n}]/u_{n,n}, \quad i = n + 1 \text{ hasta } N \tag{6.7.12}$$

En el proceso anterior, no se calculan los elementos de la diagonal de L, es decir, $l_{i,i}$, puesto que todos valen 1.

Como se habrá observado, los elementos de la parte triangular superior de L son iguales a cero. También los elementos de la parte triangular inferior de la matriz U se anulan. Por lo tanto, los elementos de L y U se pueden guardar en un arreglo con el fin de ahorrar espacio en la memoria. Por ejemplo, las matrices L y U de la ecuación (6.7.1) se pueden combinar en un arreglo como

$$\begin{matrix} u_{1,1} & u_{1,2} & u_{1,3} \\ l_{2,1} & u_{2,2} & u_{2,3} \\ l_{3,1} & l_{3,2} & u_{3,3} \end{matrix}$$

En este arreglo, los elementos de la diagonal de L no se guardan porque valen 1. Para reducir aún más el uso del espacio en la memoria, los resultados de la factorización se escriben encima del espacio de memoria de A. Esto es posible debido a que cada elemento $a_{i,j}$ de A se utiliza sólo una vez para calcular $l_{i,j}$ o $u_{i,j}$ en toda la factorización. Por lo tanto, al utilizar $a_{i,j}$, su espacio de memoria se puede utilizar para guardar $l_{i,j}$ o $u_{i,j}$.

Ahora estudiamos las formas para resolver un conjunto de ecuaciones lineales. La ecuación $Ax = y$ se puede escribir como

$$LUx = y \tag{6.7.13}$$

donde $LU = A$. La ecuación (6.7.13) se resuelve como sigue. Sea

$$Ux = z \tag{6.7.14}$$

La ecuación (6.7.13) queda

$$Lz = y \tag{6.7.15}$$

La solución de la ecuación (6.7.15) para z es fácil, debido a la forma triangular de L. Una vez que se conoce z, se resuelve la ecuación (6.7.14) en términos de x.

En el caso de una matriz de 3×3, por ejemplo, podemos escribir la ecuación (6.7.15) como

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{2,1} & 1 & 0 \\ l_{3,1} & l_{3,2} & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \tag{6.7.16}$$

La solución se calcula en forma recursiva como

$$\begin{aligned} z_1 &= y_1 \\ z_2 &= [y_2 - z_1 l_{2,1}] \\ z_3 &= [y_3 - z_1 l_{3,1} - z_2 l_{3,2}] \end{aligned} \tag{6.7.17}$$

Escribimos la ecuación (6.7.14) en forma más explícita

$$\begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} \\ 0 & u_{2,2} & u_{2,3} \\ 0 & 0 & u_{3,3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$$

y la solución es

$$\begin{aligned} x_3 &= \frac{z_3}{u_{3,3}} \\ x_2 &= \frac{z_2 - u_{2,3}x_3}{u_{2,2}} \\ x_1 &= \frac{z_1 - u_{1,2}x_2 - u_{1,3}x_3}{u_{1,1}} \end{aligned}$$

Para una matriz de orden N, las eliminaciones hacia adelante y hacia atrás se resumen de la forma siguiente.

a) Paso de eliminación hacia adelante:

$$\begin{aligned} z_1 &= y_1 \\ z_i &= y_i - \left[\sum_{j=1}^{i-1} l_{i,j} z_j\right], \quad i = 2, 3, \ldots, N \end{aligned}$$

b) Paso de la eliminación hacia atrás:

$$x_N = \frac{z_N}{u_{N,N}}$$

$$x_i = \frac{\left[z_i - \sum_{j=i+1}^{N} u_{i,j}x_j\right]}{u_{i,i}}, \quad i = N-1, N-2, \ldots, 3, 2, 1$$

Hasta este punto de la sección, no hemos utilizado el pivoteo con el fin de hacer más sencilla la explicación. Sin embargo, el pivoteo es importante, por la misma razón que en la eliminación de Gauss. Debemos recordar que el pivoteo en la eliminación de Gauss es equivalente a mezclar las ecuaciones en el conjunto. En forma matricial, quiere decir que los renglones de coeficientes se mezclan entre sí junto con el término del lado derecho. Esto indica que el pivoteo se puede aplicar a la descomposición LU siempre que la forma de mezclar se aplique a los términos de ambos lados en la misma forma. Al hacer el pivoteo en la descomposición LU, se registran los cambios en el orden de los renglones. Después se aplica el mismo reordenamiento a los términos del lado derecho, antes de comenzar a resolver de acuerdo con los pasos a) y b) señalados arriba.

Resumen de Esta Sección

a) Cualquier matriz no singular se puede descomponer en la forma LU.
b) Si un conjunto de ecuaciones lineales debe resolverse en forma repetida con distintos términos no homogéneos, es recomendable la descomposición LU.
c) La matriz U es idéntica al arreglo de coeficientes que aparece en la eliminación de Gauss cuando finaliza la eliminación hacia adelante.
d) La descomposición LU también es útil al evaluar el determinante, como se verá en la siguiente sección.

6.8 DETERMINANTES

Ya hemos tenido contacto con los determinantes, pero hemos pospuesto el análisis detallado hasta ahora.

El determinante es un número importante asociado con toda matriz cuadrada. De hecho, un conjunto no homogéneo de ecuaciones lineales no tiene una solución única, a menos que el determinante de la matriz de coeficientes sea distinto de cero. Por otro lado, un conjunto homogéneo de ecuaciones lineales tiene más de una solución sólo cuando el determinante es igual a cero. Hay muchas ocasiones en las que es necesario evaluar el determinante de una matriz. (Véase el capítulo 7 para ver ejemplos.)

El determinante de una matriz A de orden N se denota por det (A) y se define como

$$\det(A) = \sum (\pm) a_{i1}a_{j2}a_{k3}, \ldots, a_{rN} \tag{6.8.1}$$

donde la suma se hace sobre todas las permutaciones de los primeros subíndices de a, y ($\pm$) toma el signo más si la permutación es par y menos si la permutación es impar.*

Para una matriz de 2×2, el determinante de A se calcula como

$$\det(A) = \det \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = a_{11}a_{22} - a_{12}a_{21} \tag{6.8.2}$$

Para una matriz de 3×3, el determinante es

$$\begin{aligned} \det(A) &= \det \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \\ &= a_{11}a_{22}a_{33} + a_{21}a_{32}a_{13} + a_{31}a_{12}a_{23} \\ &\quad - a_{11}a_{32}a_{23} - a_{21}a_{12}a_{33} - a_{31}a_{22}a_{13} \end{aligned} \tag{6.8.3}$$

Se puede memorizar la regla para calcular el determinante de una matriz de 3×3 como la *regla del espagueti*. En la figura 6.2, cada una de las tres líneas continuas une tres números. Los productos a lo largo de las tres líneas continuas tienen signos positivos en la ecuación (6.8.3). Los productos a lo largo de las líneas punteadas tienen signos negativos en la ecuación (6.8.3). Esta regla no se puede extender a una matriz de orden mayor o igual que 4×4.

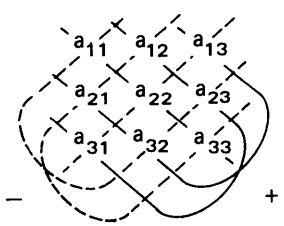


Figura 6.2 Regla del espagueti para calcular el determinante de una matriz de 3×3.

Si el orden de la matriz es mayor que 3, el cálculo directo del determinante por medio de la ecuación (6.8.2) no es práctico debido a que el número de cálculos aumenta muy rápidamente. De hecho, una matriz de orden N tiene $N!$ permutaciones, por lo que el determinante de una matriz de 5×5, por ejemplo, tiene 120 términos, cada uno de los cuales necesita cuatro multiplicaciones. El determinante de una matriz de 10×10 tiene más de 2×10^8 términos, cada uno de los cuales requiere nueve multiplicaciones.

* La secuencia del primer subíndice es $(i, j, k, \ldots, r)$ y se llama *permutación*. Una permutación es impar o par si $(i, j, k, \ldots, r)$ se obtiene al intercambiar el orden de cualesquiera dos números consecutivos en $(1, 2, 3, \ldots, N)$ un número impar o par de veces, respectivamente. Por ejemplo, $(3, 2, 1, 4, \ldots, N)$ se obtiene mediante intercambios de los primeros tres números: $123 \rightarrow 213 \rightarrow 231 \rightarrow 321$ (es decir, tres veces). Así, la permutación $(3, 2, 1, 4, \ldots, N)$ es impar. Sin embargo, se puede ver que los intercambios entre dos números no tienen que hacerse entre dos números consecutivos sino entre cualquier pareja de números. En este ejemplo, $(3, 2, 1, 4, \ldots, N)$ se obtiene al intercambiar 1 y 3 en $(1, 2, 3, \ldots, N)$. El número de intercambios es uno, por lo que la permutación $(3, 2, 1, 4, \ldots, N)$ es impar.

Una forma práctica de calcular el determinante es utilizar el proceso de eliminación hacia adelante en la eliminación de Gauss o, en forma alternativa, la descomposición LU descrita en la sección 6. Primero haremos notar dos importantes reglas de los determinates:

Regla 1: det (BC) = det (B) det (C)

lo que significa que el determinante de un producto de matrices es el producto de los determinantes de todas las matrices.

Regla 2: det (M) = el producto de todos los elementos de la diagonal de M, si M es una matriz triangular superior e inferior.

Por ejemplo, si todos los elementos de la diagonal de una matriz triangular valen 1, el determinante también es unitario.

Si no se utiliza el pivoteo, el cálculo del determinante mediante la descomposición LU es directo. Según la regla 1, el determinante se puede escribir como

$$\det(A) = \det(LU) = \det(L)\det(U) = \det(U) \tag{6.8.4}$$

donde det (L) = 1 porque L es una matriz triangular inferior y todos los elementos de la diagonal valen 1. El det (U) es el producto de todos los elementos de la diagonal de U, que es igual a det (A):

$$\det(A) = \prod_{i=1}^{N} u_{ii} \tag{6.8.5}$$

Cuando se utiliza el pivoteo en la descomposición LU, su efecto debe tomarse en cuenta. Primero, debemos reconocer que la descomposición LU con pivoteo es equivalente a realizar dos procesos separados: 1) transformar A en A' llevando a cabo todos los cambios de renglones, y 2) descomponer en seguida A' en LU sin pivoteo. El primer paso se puede expresar como

$$A' = PA, \text{ o en forma equivalente } A = P^{-1}A' \tag{6.8.6}$$

donde P es una *matriz de permutación* y representa la operación de pivoteo. El segundo proceso es

$$A' = LU \tag{6.8.7}$$

Por lo tanto, L y U están relacionados con A en la forma

$$A = P^{-1}LU \tag{6.8.8}$$

El determinante de A se puede escribir entonces como

$$\det(A) = \det(P^{-1}) \det(L) \det(U)$$

o, en forma equivalente,

$$\det(A) = \gamma \det(U) \tag{6.8.9}$$

donde $\det(L) = 1$ y $\gamma = \det(P^{-1})$ es igual a -1 o $+1$ dependiendo de si el número de pivoteo es impar o par, respectivamente.

Este algoritmo de cálculo del determinante se incorpora en el PROGRAMA 6-3.

Ejemplo 6.10

Calcule el determinante de la matriz del ejemplo 6.9.

(Solución)

Por medio del resultado de la descomposición LU en el ejemplo 6.9.

$$\det(L) = 1 \text{ y } \det(U) = (2)(3.5)(1.5714) = 11$$

Puesto que no se utilizó ningún pivoteo, $\gamma = 1$. Así, el determinante es

$$\det(A) = \gamma \det(U) = 11$$

También se puede calcular el determinante de una matriz durante el proceso de eliminación de Gauss. Esto se debe a que cuando se termina la eliminación hacia adelante, la matriz original se ha transformado en la matriz U de la descomposición LU. Por lo tanto, el determinante se puede calcular tomando el producto de todos los términos de la diagonal y multiplicando después por 1 o -1, según sea par o impar el número de operaciones de pivoteo realizadas, respectivamente. Este es el algoritmo que se implanta en el PROGRAMA 6-1 para calcular el determinante.

Ejemplo 6.11

a) Calcule el determinante de la matriz de coeficientes en la ecuación (6.3.1).
b) Calcule el determinante de la matriz del ejemplo 6.7.

(Solución)

a) La matriz de coeficientes para la ecuación (6.3.1) es

$$A = \begin{bmatrix} 0 & 10 & 1 \\ 1 & 3 & -1 \\ 2 & 4 & 1 \end{bmatrix}$$

De acuerdo con la ecuación (6.3.5), la matriz después de terminar la eliminación hacia adelante queda

$$\begin{bmatrix} 2 & 4 & 1 \\ 0 & 10 & 1 \\ 0 & 0 & -16/5 \end{bmatrix}$$

donde se aplicaron dos pivoteos.

Por lo tanto,

$$\det(A) = (-1)^2(2)(10)(-16/5) = 64$$

b) La matriz se define como [véase la ecuación (A) del ejemplo 6.7]

$$A = \begin{bmatrix} 2 & 1 & -3 \\ -1 & 3 & 2 \\ 3 & 1 & -3 \end{bmatrix}$$

Después de la eliminación hacia adelante, la matriz triangular superior es [véase la ecuación (B) del ejemplo 6.7]

$$\begin{bmatrix} 2 & 1 & -3 \\ 0 & 3.5 & 0.5 \\ 0 & 0 & 1.5714 \end{bmatrix}$$

No se usó ningún pivoteo. Por lo tanto, obtenemos

$$\det(A) = (2)(3.5)(1.5714) = 11$$

Resumen de Esta Sección

a) El determinante se puede calcular fácilmente para las matrices de 2×2 y 3×3 mediante cálculos a mano.
b) Para matrices más grandes, se usa la eliminación de Gauss o la descomposición LU.
c) Las dos reglas analizadas en esta sección, con frecuencia, son muy útiles para evaluar el determinante.

6.9 PROBLEMAS MAL CONDICIONADOS

Los problemas sin solución única se pueden identificar con relativa facilidad. De cualquier forma, si uno de esos problemas se trata de resolver, la computadora se detendrá.

Sin embargo, ciertos problemas tienen solución, aunque sus soluciones se vuelven muy imprecisas debido a severos errores de redondeo. Los problemas de este tipo se llaman *problemas mal condicionados*.

La matriz A de un problema mal condicionado tiene las siguientes características:

a) Un ligero cambio de los coeficientes (o elementos de la matriz) provoca cambios significativos en la solución.
b) Los elementos de la diagonal de la matriz de coeficientes tienden a ser menores que los elementos que no pertenecen a la diagonal.

c) El det (A) det (A^{-1}) difiere en forma significativa de 1.
d) El resultado de $(A^{-1})^{-1}$ es muy distinto de A.
e) AA^{-1} difiere en grado sumo de la matriz identidad.
f) $A^{-1}(A^{-1})^{-1}$ difiere más de la matriz identidad que lo que difiere AA^{-1}.

El pivoteo analizado antes mejora la exactitud de la solución si el problema está más o menos mal condicionado, pero en el caso de los problemas mal condicionados, el solo uso del pivoteo no salva la exactitud. El mejor remedio es aumentar la precisión del cálculo (véase el ejemplo 6.4 y el capítulo 1, donde se analiza la precisión).

Ejemplo 6.12

Una matriz de Hilbert [Morris] se define como

$$A = [a_{i,j}]$$

donde

$$a_{i,j} = \frac{1}{i+j-1}$$

de la cual se sabe que está mal condicionada, incluso para un orden pequeño. Calcule a) A^{-1} $(A^{-1})^{-1}$ y b) det $(A$ det (A^{-1}) para la matriz de Hilbert de 4×4. Use precisión simple.

(Solución)

La matriz de Hilbert de 4×4 es

$$\begin{bmatrix} 1 & 1/2 & 1/3 & 1/4 \\ 1/2 & 1/3 & 1/4 & 1/5 \\ 1/3 & 1/4 & 1/5 & 1/6 \\ 1/4 & 1/5 & 1/6 & 1/7 \end{bmatrix}$$

Se obtienen los siguientes resultados, utilizando la precisión simple en una VAX 8550: a) $A^{-1}(A^{-1})^{-1} =$

$$\begin{bmatrix} 1.0001183 & -0.0014343 & 0.0032959 & -0.0021362 \\ -0.0000019 & 1.0000000 & -0.0001221 & 0.0000610 \\ 0.0000000 & 0.0000000 & 0.9999390 & 0.0000305 \\ 0.0000000 & -0.0000305 & 0.0000610 & 0.9999390 \end{bmatrix}$$

b) det (A) det $(A^{-1}) = (1.6534499E-07)(6047924.) = 0.99999393$

El producto de los determinantes se desvía de la unidad al aumentar el orden de la matriz. Sin embargo, la desviación de A^{-1} $(A^{-1})^{-1}$ de la matriz identidad detecta las matrices mal condicionadas en forma más clara que el producto de los determinantes.

Resumen de Esta Sección

a) El hecho de que la matriz de coeficientes de un conjunto de ecuaciones lineales esté mal condicionado o no, no se puede ver fácilmente examinando la solución de las ecuaciones lineales.
b) Entre los métodos para examinar las matrices mal condicionadas se incluyen el cálculo de A^{-1} $(A^{-1})^{-1}$ y det (A) det (A^{-1}).

6.10 SOLUCION DE N ECUACIONES CON M INCOGNITAS

En las secciones anteriores estudiamos cómo calcular la solución única de $Ax = y$ utilizando la eliminación de Gauss u otros métodos. La condición necesaria para la existencia de una única solución es que A sea una matriz cuadrada y det $(A) \neq 0$. Si det $(A) = 0$, decimos que la matriz es singular y dejamos de buscar la solución. Sin embargo, esto no se debe a que no haya solución, sino a que no existe solución única [Strang]. Si det $(A) = 0$, al menos una de las ecuaciones es linealmente dependiente y puede eliminarse. Después de la eliminación, el número de ecuaciones se vuelve menor que el número de incógnitas.

En general, el número de ecuaciones, n, puede ser menor que el número de incógnitas, m. La ecuación de un problema de este tipo se puede escribir en la forma $Ax = y$, donde la matriz A ya no es cuadrada sino rectangular. Para $n < m$, el número de soluciones es infinito, si el sistema es homogéneo, pero los valores numéricos de una solución no pueden determinarse de manera única. En esta sección examinaremos las soluciones no únicas de n ecuaciones con m incógnitas, donde $n < m$.

Como un ejemplo de ecuaciones lineales de $n < m$, consideremos

$$x + y = 1 \tag{6.10.1}$$

donde $n =$ y $m = 2$. La solución se puede escribir como

$$x = 1 - y$$

o

$$y = 1 - x$$

En la primera de estas ecuaciones, la y del lado derecho es una *variable libre* y la x del lado izquierdo es una *variable básica*. En la segunda ecuación, x es una variable libre y y es una variable básica. Cualquiera que sea la forma elegida de la solución, la solución para la variable básica está dada en términos de la variable libre. En caso de que el número de ecuaciones sea insuficiente, a) la solución está dada en la forma de una ecuación, en vez de forma numérica, y b) el número de soluciones es infinito debido a que los parámetros libres pueden tomar cualquier valor.

Si tenemos n ecuaciones linealmente independientes para m incógnitas y $n < m$, podemos encontrar n variables básicas y $m - n$ variables libres. Si ponemos las

variables básicas del lado izquierdo de las ecuaciones y todas las variables libres del lado derecho, en el conjunto de n ecuaciones se pueden despejar las variables libres en términos de las variables básicas.

El único requerimiento para elegir las variables básicas es que las n ecuaciones de las variables básicas formen un conjunto no singular. Como resultado, la elección de variables básicas no siempre es única. El hecho de saber cuáles de las variables pueden ser básicas no es evidente a primera vista. Sin embargo, hay un método para encontrarlas por medio de la eliminación de Gauss (o Gauss-Jordan) diseñado para las matrices de $n \times m$, el cual se explica a continuación.

Aunque en el párrafo anterior supusimos que n de las ecuaciones dadas son linealmente independientes, eliminaremos esta restricción en este momento. Esto se debe a que, mediante la eliminación de Gauss, se eliminarán en forma automática las ecuaciones dependientes, por lo que las ecuaciones restantes serán linealmente independientes.

Consideremos el siguiente sistema

$$\begin{aligned} -1u + 2v + 2w + x - 2y &= 2 \\ 3u - 6v - w + 5x - 4y &= 1 \\ 2u - 4v - 1.5w + 2x - y &= -0.5 \end{aligned} \tag{6.10.2}$$

donde $n = 3$ y $m = 5$. Para simplificar la exposición, reescribimos las ecuaciones anteriores en forma de arreglo aumentado:

u	v	w	x	y	LD
−1	2	2	1	−2	2
3	−6	−1	5	−4	1
2	−4	−1.5	2	−1	−0.5

El primer objetivo en este conjunto es el pivoteo, debido a que el coeficiente de u en el segundo renglón es mayor que el del primero:

u	v	w	x	y	LD
3	−6	−1	5	−4	1
−1	2	2	1	−2	2
2	−4	−1.5	2	−1	−0.5

Ahora, el coeficiente de u en los renglones segundo y tercero (multiplicados por $-1/3$ y $2/3$ respectivamente), se eliminan restándoles el primer renglón:

u	v	w	x	y	LD
3	−6	−1	5	−4	1
0	0	1.666667	2.666667	−3.333333	2.333333
0	0	−0.833333	−1.333333	1.666667	−1.166667

En el arreglo anterior, los coeficientes de v en el segundo y tercer renglones se anulan en forma automática. Si estuviéramos trabajando con la eliminación de Gauss,

como se describió en la sección 6.2, terminaríamos aquí el proceso. Sin embargo, pasamos al tercer renglón.

Consideramos el coeficiente de w en el segundo renglón como pivote y eliminamos el coeficiente de w en el tercer renglón restándole el segundo renglón (previamente multiplicado por 0.833333/1.666667)

u	v	w	x	y	LD
3	−6	−1	5	−4	1
0	0	1.666667	2.666667	−3.333333	2.333333
0	0	0	0	0	0

En este arreglo, se ha anulado completamente el último renglón, lo que indica que la tercera ecuación no era independiente. Ahora, consideramos que el número de ecuaciones es $n = 2$ y $m = 5$.

Las ecuaciones representadadas por el arreglo anterior se escriben en forma explícita como

$$\begin{aligned} 3u - 6v - \quad 1w + \quad 5x - \quad 4y &= 1 \\ 1.666667w + 2.666667x - 3.333333y &= 2.333333 \end{aligned} \tag{6.10.3}$$

Los coeficientes de los términos principales después de la eliminación hacia adelante (como 3 en el primer renglón y 1.666667 en el segundo renglón) son pivotes. Las variables correspondientes, u y w, se llaman *variables básicas*. Las demás son *variables libres*. Movemos todas las variables libres del lado derecho, con lo que obtenemos

$$\begin{aligned} 3u - \quad w &= 1 + 6v - \quad 5x + \quad 4y \\ 1.666667w &= 2.333333 \quad - 2.666667x + 3.333333y \end{aligned} \tag{6.10.4}$$

Es claro que la matriz de coeficientes

$$\begin{bmatrix} 3 & -1 \\ 0 & 1.666667 \end{bmatrix}$$

es no singular, por lo que se puede obtener la solución de las variables básicas en términos de las variables libres. Si aplicamos la sustitución hacia atrás, la solución final es

$$\begin{aligned} u &= 0.8 + 2v - 2.2x + 2y \\ w &= 1.4 \quad - 1.6x + 2y \end{aligned} \tag{6.10.5}$$

Al desarrollar un programa para resolver las ecuaciones de $m \times n$, la eliminación de Gauss-Jordan tiende a tener más ventajas que la eliminación de Gauss. Para ilustrar la aplicación de la eliminación de Gauss-Jordan, consideremos otro problema:

u	v	w	x	y	LD
2	3	1	4	1	6
2	3	1	1	−1	1
4	6	−1	1	2	5

Después del pivoteo, el arreglo queda

$$\begin{array}{rrrrrr} 4 & 6 & -1 & 1 & 2 & 5 \\ 2 & 3 & 1 & 1 & -1 & 1 \\ 2 & 3 & 1 & 4 & 1 & 6 \end{array}$$

El primer renglón se normaliza dividiendo entre 4 y después se eliminan los primeros coeficientes de los demás renglones:

$$\begin{array}{rrrrrr} 1 & 1.5 & -0.25 & 0.25 & 0.5 & 1.25 \\ 0 & 0 & 1.5 & 0.5 & -2 & -1.5 \\ 0 & 0 & 1.5 & 3.5 & 0 & 3.5 \end{array}$$

donde los segundos coeficientes en los renglones segundo y tercero se eliminan automáticamente. El segundo renglón se normaliza dividiendo entre 1.5. Se eliminan en seguida los terceros coeficientes del primer y tercer renglones, restándole a éstos un múltiplo del segundo renglón:

$$\begin{array}{rrrrrr} 1 & 1.5 & 0 & 0.333333 & 0.166667 & 1 \\ 0 & 0 & 1 & 0.333333 & -1.333333 & -1 \\ 0 & 0 & 0 & 3 & 2 & 5 \end{array}$$

El tercer renglón se normaliza dividiendo entre 3. Se eliminan a continuación los cuartos coeficientes del primer y segundo renglones, restándoles un múltiplo del tercer renglón:

$$\begin{array}{rrrrrr} 1 & 1.5 & 0 & 0 & -0.055556 & 0.444444 \\ 0 & 0 & 1 & 0 & -1.555556 & -1.555556 \\ 0 & 0 & 0 & 1 & 0.666667 & 1.666667 \end{array}$$

En el arreglo anterior, el coeficiente de cada variable básica es 1 y es el único coeficiente no nulo de cada columna. La ecuación obtenida se escribe en forma explícita como

$$\begin{aligned} 1u + 1.5v + 0w + 0x - 0.055556y &= 0.444444 \\ 1w + 0x - 1.555556y &= -1.555556 \\ 1x + 0.666667y &= 1.666667 \end{aligned} \tag{6.10.6}$$

o, después de mover las variables libres del lado derecho,

$$\begin{aligned} u &= 0.444444 - 1.5v + 0.055556y \\ w &= -1.555556 \qquad\quad + 1.555556y \\ x &= 1.666667 \qquad\quad - 0.666667y \end{aligned} \tag{6.10.7}$$

De manera más general, consideremos n ecuaciones con m incógnitas:

$$\begin{aligned} a_{1,1}x_1 + a_{1,2}x_2 + \cdots a_{1,m}x_m &= y_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots a_{2,m}x_m &= y_2 \\ &\vdots \\ a_{n,1}x_1 + a_{n,2}x_2 + \cdots a_{n,m}x_m &= y_n \end{aligned} \tag{6.10.8}$$

donde suponemos que $n \leqslant m$, incluyendo el caso de $n = m$. Las ecuaciones se expresan en la forma de arreglo aumentado:

$$\begin{array}{cccc} a_{1,1} & a_{1,2} & \cdots & a_{1,m} : y_1 \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} : y_2 \\ & & \cdots & \\ a_{n,1} & a_{n,2} & \cdots & a_{n,m} : y_n \end{array} \tag{6.10.9}$$

La aplicación de la eliminación de Gauss-Jordan al arreglo anterior llevará a una forma como la siguiente:

$$\begin{array}{cccccccc} 1 & x & 0 & 0 & x & x & : & x' \\ 0 & 0 & 1 & 0 & x & x & : & x' \\ 0 & 0 & 0 & 1 & x & x & : & x' \\ 0 & 0 & 0 & 0 & 0 & 0 & : & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & : & 0 \end{array} \tag{6.10.10}$$

donde x denota valores no nulos, los pivotes valen 1 y los símbolos "x" corresponden a los coeficientes de las variables libres. En los procesos de eliminación de Gauss-Jordan, los renglones se intercambian (pivoteo) en caso necesario. Los renglones nulos representan a las ecuaciones linealmente dependientes que se han eliminado. Es fácil reescribir este arreglo en forma de ecuación, como se mostró antes. El esquema de solución está implantado en el PROGRAMA 6-4.

El algoritmo de solución que se ha explicado es universal, ya que, además de calcular la solución de un sistema de $n \times m$:

a) Se puede aplicar para encontrar la solución única de un sistema $n \times n$.
b) Encuentra la solución, incluso cuando la matriz de coeficientes sea singular.
c) Se puede utilizar para calcular la inversa de una matriz cuadrada. Para encontrar la inversa de una matriz de $n \times n$, consideramos un arreglo aumentado de $n \times (2n + 1)$, en el que las primeras n columnas son de la matriz cuadrada, las siguientes n forman una matriz identidad y la última columna —correspondiente a los términos no homogéneos— se hacen iguales a cero.

Resumen de Esta Sección

a) Cuando el rango de las ecuaciones lineales es mayor que el número de incógnitas, las soluciones para las variables básicas se dan en términos de las variables libres.

b) Los algoritmos de solución para los sistemas de $n \times m$ son universales y sirven para resolver dichos sistemas, así como para encontrar la inversa de una matriz.

PROGRAMAS

Programa 6-1 Eliminación de Gauss

A) Explicaciones

El PROGRAMA 6-1 resuelve un conjunto de ecuaciones lineales por el método de eliminación de Gauss con pivoteo, de acuerdo con el algoritmo explicado en la sección 6.3.

Este programa sólo utiliza un arreglo de variables, $A(I, J)$. Los enunciados DATA definen a N y $A(I, J)$ en la forma aumentada, incluyendo los términos del lado derecho. La eliminación de Gauss se lleva a cabo en la subrutina GAUSS. En dicha subrutina, se calcula primero el épsilon de la máquina y después se utiliza para hacer cero los números pequeños que surgen debido a errores de truncamiento de restas. La eliminación hacia adelante se lleva a cabo en el ciclo DO que se cierra en S-1010. El pivoteo se hace en el ciclo DO antes de S-1045. El ciclo DO antes de S-1200 es la sustitución hacia atrás. Este programa también calcula el determinante de la matriz (véase la sección 6.8 para el determinante).

B) Variables

$A(I, J)$: arreglo aumentado.
N: orden de la matriz
EPS: épsilon de la máquina

C) Listado

```
C-----CSL/F6-1.FOR      ELIMINACION DE GAUSS
      DIMENSION A(10,11)
      PRINT *
      PRINT *,'CSL/F6-1      ELIMINACION DE GAUSS '
      DATA  N/3/                                  !-- N es el orden de la matriz
      DATA  (A(1,J),J=1,4)/ 0,-1, 2, 0/ !-- inicializa los elementos de la matriz
      DATA  (A(2,J),J=1,4)/-2, 2,-1, 0/ !-- inicializa los elementos de la matriz
      DATA  (A(3,J),J=1,4)/-2, 4, 3, 1/ !-- inicializa los elementos de la matriz
      PRINT *
      PRINT *,' MATRIZ AUMENTADA '
      PRINT *
      DO I=1,N
          PRINT 61,(A(I,J),J=1,4)
61        FORMAT(1X,1P6E12.4)
      END DO
      PRINT *
      CALL  GAUSS(N,A)
```

```
65      PRINT *
68      PRINT *,' SOLUCION '
69      PRINT *,'----------------------------------------'
        PRINT *,'          I          X(I)'
70      PRINT *,'----------------------------------------'
          DO I=1,N
72          FORMAT(5X,I5, 1PE16.6)
            PRINT 72,I,A(I,N+1)
          END DO
75      PRINT *,'----------------------------------------'
80      PRINT *
        STOP
        END
C*************************
        SUBROUTINE GAUSS(N,A)                    ! Eliminación de Gauss
        INTEGER PV                               ! Indice de pivoteo
        DIMENSION A(10,11)
        EPS=1.0                                  ! Se calcula el épsilon de la máquina
10      IF (1.0+EPS.GT.1.0) THEN
           EPS=EPS/2.0
           GOTO 10
        END IF
        EPS=EPS*2.
        PRINT *,'              EPSILON DE LA MAQUINA =',EPS
        EPS2=EPS*2                               !
1005    DET= 1                                   ! Inicialización del determinante
        DO 1010 I=1,N-1
           PV=I
           DO J=I+1,N
              IF (ABS(A(PV,I))   .LT.   ABS(A(J,I))) PV=J
           END DO
           IF (PV.EQ.I) GOTO 1050
           DO JC=1,N+1
              TM=A(I,JC)
              A(I,JC)=A(PV,JC)
              A(PV,JC)=TM
           END DO
1045       DET=-1*DET     ! Cada vez que se realice un pivoteo, cambia el signo de DET
1050       IF (A(I,I).EQ.0) GOTO 1200 ! Una matriz singular si A(I, I) = O.
           DO JR=I+1, N                ! Eliminación por debajo de la diagonal.
              IF (A(JR,I).NE.0) THEN
                 R=A(JR,I)/A(I,I)
                 DO KC=I+1,N+1
                    TEMP=A(JR,KC)
                    A(JR,KC)=A(JR,KC)-R*A(I,KC)
                    IF (ABS(A(JR,KC)).LT.EPS2*TEMP) A(JR,KC)=0.0
C                         Si el resultado de la resta es menor que
C                         el doble del épsilon de la máquina por el valor
C                         original, se cambia su valor a cero.
                 END DO
              END IF
1060       END DO
1010    CONTINUE
        DO I=1,N
           DET=DET*A(I,I)                  !  Se calcula el determinante.
        END DO
        PRINT *
        PRINT *,' DETERMINANTE = ',DET
        PRINT *
        IF (A(N,N).EQ.0) GOTO 1200
        A(N,N+1)=A(N,N+1)/A(N,N)
```

```
      DO NV=N-1,1,-1                    ! Comienza la sustitución hacia atrás.
         VA=A(NV,N+1)
         DO K=NV+1,N
            VA=VA-A(NV,K)*A(K,N+1)
         END DO
         A(NV,N+1)=VA/A(NV,NV)
      END DO
      RETURN
1200  PRINT *,'LA MATRIZ ES SINGULAR'
      STOP
      END
```

D) Ejemplo de salida

```
CSL/F6-1      ELIMINACION DE GAUSS

MATRIZ AUMENTADA
  0.0000E+00 -1.0000E+00  2.0000E+00  0.0000E+00
 -2.0000E+00  2.0000E+00 -1.0000E+00  0.0000E+00
 -2.0000E+00  4.0000E+00  3.0000E+00  1.0000E+00
            EPSILON DE LA MAQUINA=  5.9604645E-08
                 DETERMINANTE =    -16.00000
SOLUCION
-----------------------------------------
        I          X(I)
-----------------------------------------
        1     1.875000E-01
        2     2.500000E-01
        3     1.250000E-01
-----------------------------------------
```

PROGRAMA 6-2 Inversión de una matriz

A) Explicaciones

Este programa obtiene la inversa de una matriz por medio de la eliminación de Gauss, como se explicó al final de la sección 6.6. Los primeros enunciados DATA definen el valor de N, que por el momento es 3, con el fin de ejemplificar. El arreglo aumentado también se especifica en un enunciado DATA. La inversión de la matriz se lleva a cabo en la subrutina, la cual es esencialmente igual a la eliminación de Gauss del PROGRAMA 6-1, excepto que se añade la parte aumentada y que la sustitución hacia atrás se aplica a las N columnas del lado derecho en la matriz aumentada. Cuando se concluye la subrutina, la matriz inversa está incluida en las últimas N columnas del arreglo aumentado A. Después de regresar al programa principal, se imprime la matriz inversa.

B) Variables

EPS: épsilon de la máquina

N: orden de la matriz

A(I, J): arreglo aumentado: las primeras N columnas son la matriz dada y las últimas N columnas son inicialmente la matriz identidad

C) Listado

```
C-----CSL/F6-2.FOR        INVERSION DE UNA MATRIZ
      DIMENSION A(0:10,0:20)
      PRINT *
      PRINT *,'CSL/F6-2     INVERSION DE UNA MATRIZ '
      DATA N/3/
      DATA (A(1,J),J=1,6)/2, 1,-3, 1, 0, 0/
      DATA (A(2,J),J=1,6)/-1, 3, 2, 0, 1, 0/
      DATA (A(3,J),J=1,6)/3, 1,-3, 0, 0, 1/
      PRINT *
      PRINT *,'MATRIZ ORIGINAL '
      PRINT *
      DO I=1, N
         PRINT 20,(A(I,J),J=1,3)
      END DO
20    FORMAT(1X,1P5E12.5)
      PRINT *
      CALL  GAUSS(N,A)
      PRINT *
      PRINT *,' MATRIZ INVERSA '
      PRINT
      DO I=1,N
         PRINT 20,(A(I,J),J=N+1,N*2)
      END DO
      PRINT *
      STOP
      END
C***************************
      SUBROUTINE GAUSS(N,A)          !  Subrutina para la eliminación de Gauss
      DIMENSION  A(0:10,0:20)
      INTEGER PV
      EPS=1.0
10    IF(1.0+EPS.GT.1.0) THEN
        EPS=EPS/2.0
        GOTO 10
      END IF
      EPS=EPS*2
      PRINT *,' EPSILON DE LA MAQUINA =', EPS
      EPS2=EPS*2
      DET=1.0                            !  Inicialización del determinante
      DO 1010 I=1,N-1
         PV=I
         DO J=I+1,N
           IF (ABS(A(PV,I))  .LT.  ABS(A(J,I))) PV=J
         END DO
         IF (PV.NE.I ) THEN
           DO JC=1,N*2
             TM=A(I,JC)
             A(I,JC)=A(PV,JC)
             A(PV,JC)=TM
           END DO
           DET=-DET
         END IF
         IF (A(I,I).EQ.0) GOTO  1200
C   ELIMINACION POR DEBAJO DE LA DIAGONAL ---
         DO JR=I+1,N
           IF (A(JR,I).NE.0) THEN
              R=A(JR,I)/A(I,I)
              DO KC=I+1,N*2
                 TEMP=A(JR,KC)
```

```
                A(JR,KC)=A(JR,KC)-R*A(I,KC)
                IF(ABS(A(JR,KC)).LT.EPS2*TEMP) A(JR,KC)=0.0
             END DO
          END IF
       END DO
1010   CONTINUE
       DO I=1,N
          DET=DET*A(I,I)
       END DO
       PRINT *
       PRINT *, DETERMINANTE=',DET
       PRINT *
C   SUSTITUCION HACIA ATRAS
       IF (A(N,N).EQ.0) GOTO 1200
       DO 1100 M=N+1,N*2
          A(N,M)=A(N,M)/A(N,N)
          DO NV=N-1,1,-1
             VA=A(NV,M)
             DO K=NV+1,N
                VA=VA-A(NV,K)*A(K,M)
             END DO
             A(NV,M)=VA/A(NV,NV)
          END DO
1100   CONTINUE
       RETURN
1200   PRINT *,'LA MATRIZ ES SINGULAR'
       END
```

D) Ejemplo de salida

```
CSL/F6-2          INVERSION DE UNA MATRIZ

MATRIZ ORIGINAL
 2.00000E+00 1.00000E+00-3.00000E+00
-1.00000E+00 3.00000E+00 2.00000E+00
 3.00000E+00 1.00000E+00-3.00000E+00
          EPSILON DE LA MAQUINA =  5.9604645E-08
                DETERMINANTE    = 11.00000

MATRIZ INVERSA
-1.00000E+00-9.93411E-09 1.00000E+00
 2.72727E-01 2.72727E-01-9.09091E-02
-9.09091E-01 9.09091E-02 6.36364E-01
```

PROGRAMA 6-3 Descomposición LU

A) Explicaciones

Este programa descompone una matriz A en la forma LU con pivoteo. En el programa, las matrices L y U se guardan en la forma combinada y se expresan mediante el arreglo $L(I, J)$; vea el párrafo posterior a la ecuación (6.7.12) para una mayor explicación. Sin embargo, para simplificar la demostración, la matriz original A y L se expresan en forma aparte. (Cualquiera de estos arreglos se puede eliminar, como se verá después.)

Este programa también calcula el determinante de la matriz original con el algoritmo explicado en la sección 6.8.

Para reducir las necesidades de espacio de memoria, se puede eliminar fácilmente uno de los arreglos A o L. Por ejemplo, si sólo se va a usar $A(I, J)$, se puede borrar "$L(30, 30)$" de los enunciados de dimensión y remplazar cada uno de los "$L(I, J)$" por "$A(I, J)$" en todo el programa.

B) Variables

N: orden de la matriz

A(I, J): elementos de la matriz A

EL(I, J): matriz triangular inferior L y matriz triangular superior U en la forma combinada

IP(I): permutación debida al pivoteo

IPC: número de pivoteos realizados

DE: determinante de la matriz original

C) Listado

```
C-----CSL/F6-3.FOR      DESCOMPOSICION LU CON PIVOTEO
      DIMENSION A(20,20),EL(20,20),IP(20)
      REAL*4 L
5     PRINT *
      PRINT *,'CSL/F6-3      DESCOMPOSICION LU '
      PRINT *
      DATA N/3/
      DATA (A(1,J),J=1,3)/ 2, 1 ,-3/
      DATA (A(2,J),J=1,3)/-1, 3,  2/
      DATA (A(3,J),J=1,3)/ 3, 1, -3/
      PRINT *,' N = ',N
      PRINT *,' MATRIZ ORIGINAL----------------------------'
      DO I=1,N
         PRINT 10,(A(I,J),J=1,N)
      END DO
10    FORMAT(1X, 1P6E12.5)
      PRINT *
      PRINT *,' SI SE DESEA EL PIVOTEO, OPRIMA 1; DE LO CONTRARIO OPRIMA 0 '
      READ *,NP
C-- INICIALIZACION DEL PIVOTEO Y LA MATRIZ EL
      IPC=1
      DO I=1,N
         IP(I)=I
         DO J=1,N
            EL(I,J)=0
         END DO
      END DO
      J=1
      IF(NP.EQ.1) CALL  PIVOT(N,A,EL,J,IP,IPC)
C  PRIMER RENGLON
      DO J=1,N
         EL(1,J)=A(1,J)
      END DO
```

```
C   PRIMERA COLUMNA
      DO I=2,N
         EL(I,1)=A(I,1)/EL(1,1)
      END
      DO 80 M=2,N
         IF(NP.EQ.1) CALL  PIVOT(N,A,EL,M,IP,IPC)
C                                                          ! M-ESIMO RENGLON
         DO J=M,N
           S=0
             DO K=1,M-1
               S=S+EL(M,K)*EL(K,J)
             END DO
           EL(M,J)=A(M,J)-S
         END DO
C                                                          ! M-ESIMA COLUMNA
         DO I=M+1,N
           S=0
             DO K=1,M-1
               S=S+EL(I,K)*EL(K,M)
             END DO
           EL(I,M)=(A(I,M)-S)/EL(M,M)
         END DO
80    CONTINUE
      PRINT *,' PERMUTACION '
          PRINT 361,(IP(I),I=1,N)
361   FORMAT(1X,10I3)
368   PRINT *
      PRINT *,' MATRICES LU EN FORMA COMPACTA '
        DO I=1,N
          PRINT 10,(EL(I,M),M=1,N)
        END DO

      PRINT *
 C---CALCULO DEL DETERMINANTE
      DE=1
      DO I=1,N
         DE=DE*EL(I,I)
      END DO
      IF (IPC.EQ.INT(IPC/2)*2)  DE=-DE
      PRINT *
      PRINT *,(' DETERMINANTE = '),DE
395   PRINT *
      PRINT *,'---------------------------------------------'
      PRINT *
      PRINT*
      END
C*********************************
      SUBROUTINE PIVOT(N,A,EL,J,IP,IPC)
      DIMENSION A(20,20),EL(20,20),IP(20)
405   T=0
      DO 420 K=J,N
         IF (ABS(A(K,J)) .LE.T) GOTO 420
         JJ=K
         T=ABS(A(K,J))
420   CONTINUE
      IF (JJ.EQ.J) RETURN
425   IPC=IPC+1
      DO 430  M=1,N
         T=A(J,M)
         A(J,M)=A(JJ,M)
```

```
          A(JJ,M)=T
          T=EL(J,M)
          EL(J,M)=EL(JJ,M)
          EL(JJ,M)=T
430     CONTINUE
        IT=IP(J)
        IP(J)=IP(JJ)
        IP(JJ)=IT
        PRINT *,' NUMERO DE PIVOTEOS    =',IPC
        RETURN
        END
```

D) Ejemplo de salida

```
CSL/F6-3      DESCOMPOSICION LU

MATRIZ ORIGINAL
 2.00000E+00 1.00000E+00-3.00000E+00
-1.00000E+00 3.00000E+00 2.00000E+00
 3.00000E+00 1.00000E+00-3.00000E+00

SI SE DESEA EL PIVOTEO, OPRIMA 1; DE LO CONTRARIO OPRIMA 0
1
NUMERO DE PIVOTEOS    = 2
PERMUTACION
  3  2  1

MATRICES LU EN FORMA COMPACTA
 3.00000E+00 1.00000E+00-3.00000E+00
-3.33333E-01 3.33333E+00 1.00000E+00
 6.66667E-01 1.00000E-01-1.10000E+00
                        DETERMINANTE  = 11.00000
```

PROGRAMA 6-4 M Ecuaciones con N incógnitas

A) Explicaciones

Este programa reduce un conjunto de m ecuaciones con n incógnitas ($m \leqslant n$) a la forma reducida, como se ilustra en la ecuación (6.10.10). Se utiliza la eliminación de Gauss-Jordan. Si el sistema es inconsistente, el programa se detiene y da un mensaje que indica la inconsistencia. Si $m = n$ y existe una única solución, el arreglo de coeficientes se convierte en una matriz identidad. Si $m < n$, los coeficientes de las variables básicas seleccionadas por el programa se convierten en uno, por lo que la solución se puede escribir con facilidad en términos de las variables libres a partir del arreglo reducido.

Para hacer el programa más sencillo y compacto, las eliminaciones de coeficientes se realizan sin separar las eliminaciones hacia adelante y hacia atrás. Esto quiere decir que cuando se haya un coeficiente pivote no nulo, se eliminan todos los coeficientes de la misma columna —arriba y abajo de los pivotes— antes de pasar al siguiente pivote. Se realiza el pivoteo antes de la eliminación. Véase la sección 6.10 para la interpretación y el uso de los resultados obtenidos.

B) Variables

A(I, J): coeficientes
DET: determinante
RANK: rango del conjunto
EPS: épsilon de la máquina

C) Listado

```
C-----CSL/F6-4.FOR        N ECUACIONES CON M INCOGNITAS
      INTEGER PV, RANK
      COMMON A(10,20),M,N
      DATA N,M/3,5/
      DATA (A(1,J),J=1,6)/2,  3,  1,  4,  1,  6/
      DATA (A(2,J),J=1,6)/2,  3,  1,  1, -1,  1/
      DATA (A(3,J),J=1,6)/4,  6, -1,  1,  2,  5/
      PRINT*
      PRINT*,'CSL/F6-4       N ECUACIONES CON M INCOGNITAS '
      PRINT*
C                                          !    Epsilon de la máquina
      EPS=1.0
      DO 10 L=1,100
          IF (EPS+1.LE.1) GOTO 15
          EPS=EPS/2
10    CONTINUE
15    EPS=EPS*2
      PRINT*,' EPSILON DE LA MAQUINA =', EPS
      EPS2=EPS*2
C                                          !    Impresión de la entrada
      PRINT*, ' N(NUMERO DE ECUACIONES)=',N
      PRINT*, ' M(NUMERO DE INCOGNITAS)=',M
      PRINT*
      PRINT *,' MATRIZ AUMENTADA (ULTIMA COLUMNA; TERMINOS NO HOMOGENEOS) '
      CALL LIST
50    CONTINUE
      PRINT*
C-- COMIENZA EL ESQUEMA DE GAUSS-JORDAN
      DET=1.0
      I=0
      DO 500 K=1,N
         PV=K
90       I=I+1
         IF(I.GT.M) GOTO 600
         DO 100 J=K+1,N
            IF (ABS(A(PV,I)).LT.ABS(A(J,I))) PV=J
100      CONTINUE
         IF (PV.EQ.K.AND. A(K,I).EQ.0) GOTO 90
         IF (PV.EQ.K) GOTO 340
300      DO 320 JC=I,M+1                   !  Pivoteo
               TM=A(K,JC)
               A(K,JC)=A(PV,JC)
               A(PV,JC)=TM
320      CONTINUE
      DET=-DET
340   Z=A(K,I)                             !  Comienza la eliminación
      DET=DET*Z
      RANK=K
```

```
      DO 330 J=I,M+1
         A(K,J)=A(K,J)/Z
330   CONTINUE
         DO 400 JR=1,N
            IF(JR.EQ.K) GOTO 400
            IF (A(JR,I).EQ.0) GOTO 400
            R=A(JR,I)
            DO 390 KC=I,M+1
               A(JR,KC)=A(JR,KC)-R*A(K,KC)
               IF (ABS(A(JR,KC)/A(K,KC)).LT.EPS2) A(JR,KC)=0
390         CONTINUE
400      CONTINUE
500   CONTINUE
      DO 520 J=I,M
         IF (A(N,J).EQ.0) GOTO 520
         DO 510 JR=J,M+1
            A(N,JR)=A(N,JR)/A(N,J)
510      CONTINUE
         A(N,J)=1.0
         GOTO 600
520   CONTINUE
C                                          !  Verificación de la consistencia
600   IF (RANK.EQ.K) GOTO 640
      KONSIS=0
      DO 620 J=RANK+1,N
         IF (A(J,M+1).NE.0) KONSIS=KONSIS+1
620   CONTINUE
640   PRINT*,' RANGO DE LA MATRIZ    =', RANK
      PRINT*,' DETERMINANTE          =',DET
      PRINT*
      PRINT*,' MATRIZ REDUCIDA :'
      CALL LIST
      PRINT*
      IF (KONSIS .NE.0 ) THEN
          PRINT*,' CUIDADO: LA MATRIZ NO ES CONSISTENTE :'.
          ENDIF
      END
C*******************
      SUBROUTINE LIST                      !  Subrutina para imprimir la matriz
      COMMON A(10,20),M,N
      WRITE(6,*)
      DO 10 I=1,N
      WRITE(6,30)   (A(I,J),J=1,M+1)
10    CONTINUE
30    FORMAT(1P8E12.4)
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F6-4      N ECUACIONES CON M INCOGNITAS

 EPSILON DE LA MAQUINA  =5.9604645E-08
 N( NUMERO DE ECUACIONES )=      3
 M( NUMERO DE INCOGNITAS )=      5

 MATRIZ AUMENTADA (ULTIMA COLUMNA: TERMINOS NO HOMOGENEOS)
 2.000000    3.000000    1.000000    4.000000    1.000000    6.000000
 2.000000    3.000000    1.000000    1.000000   -1.000000    1.000000
 4.000000    6.000000   -1.000000    1.000000    2.000000    5.000000
```

```
RANGO DE LA MATRIZ     =    3
DETERMINANTE           =   -18.00000

MATRIZ REDUCIDA :
1.000000    1.500000    0.000000    0.000000   -0.055556    0.444444
0.000000    0.000000    1.000000    0.000000   -1.555556   -1.555556
0.000000    0.000000    0.000000    1.000000    0.666667    1.666667
```

PROBLEMAS

6.1) Resuelva con una calculadora el siguiente conjunto de ecuaciones, por medio de la eliminación de Gauss, en forma de arreglo, (sin usar pivoteo).

a)
$$\begin{aligned} 2x_1 + x_2 - 3x_3 &= -1 \\ -x_1 + 3x_2 + 2x_3 &= 12 \\ 3x_1 + x_2 - 3x_3 &= 0 \end{aligned}$$

b)
$$\begin{aligned} 0.1x_1 - 0.6x_2 + x_3 &= 0 \\ -2x_1 + 8x_2 + 0.3x_3 &= 1 \\ x + 6x_2 + 4x_3 &= 2 \end{aligned}$$

6.2) Resuelva los siguientes conjuntos de ecuaciones por medio de la eliminación de Gauss-Jordan:

a)
$$\begin{aligned} 4x + y - z &= 9 \\ 3x + 2y - 6z &= -2 \\ x - 5y + 3z &= 1 \end{aligned}$$

b)
$$\begin{aligned} x - y &= 0 \\ -x + 2y - z &= 1 \\ - y + 1.1z &= 0 \end{aligned}$$

6.3) Repita el problema 6.1) con pivoteo en la forma de arreglo, utilizando una calculadora.

6.4) Resuelva el siguiente conjunto de ecuaciones sin pivoteo y después con pivoteo:

$$\begin{aligned} 6.122x + 1500.5y &= 1506.622 \\ 2000x + 3y &= 2003 \end{aligned}$$

Redondee los números después de la sexta cifra significativa.

6.5) Resuelva las siguientes ecuaciones mediante la eliminación de Gauss sin pivoteo y después con pivoteo. Para simular el efecto del redondeo, trunque cada número después de la cuarta cifra significativa.

$$\begin{aligned} 1.001x + 1.5y &= 0 \\ 2x + 3y &= 1 \end{aligned}$$

6.6) Los siguientes conjuntos de ecuaciones lineales tienen coeficientes comunes pero distintos términos del lado izquierdo:

a)
$$\begin{aligned} x + y + z &= 1 \\ 2x - y + 3z &= 4 \\ 3x + 2y - 2z &= -2 \end{aligned}$$

b)
$$\begin{aligned} x + y + z &= -2 \\ 2x - y + 3z &= 5 \\ 3x + 2y - 2z &= 1 \end{aligned}$$

c) $$\begin{aligned} x + y + z &= 2 \\ 2x - y + 3z &= -1 \\ 3x + 2y - 2z &= 4 \end{aligned}$$

Los coeficientes y los tres conjuntos de términos del lado derecho se pueden combinar en un arreglo

$$\begin{matrix} 1 & 1 & 1 & 1 & -2 & 2 \\ 2 & -1 & 3 & 4 & 5 & -1 \\ 3 & 2 & -2 & -2 & 1 & 4 \end{matrix}$$

Si aplicamos el esquema de Gauss-Jordan a este arreglo y reducimos las tres primeras columnas a la forma de la matriz identidad, las soluciones para los tres problemas se obtienen en forma automática en las columnas cuarta, quinta y sexta al terminar la eliminación. Calcule la solución de esta forma.

6.7) Calcule $C \equiv A + B$, $D \equiv A - B$, $E \equiv AB$, donde

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 3 & 0 & 2 \end{bmatrix}$$

$$B = \begin{bmatrix} 4 & 1 & 2 \\ 3 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

6.8) Calcule $B^t A^t$ y $(AB)^t$ utilizando las definiciones del problema anterior y muestre que los resultados son idénticos.

6.9) Calcule $E = AB$, donde

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 3 & 0 & 2 \end{bmatrix}$$

$$B = \begin{bmatrix} 3 \\ 5 \\ 1 \end{bmatrix}$$

6.10) Calcule $D = A + A'$, $E = A - A'$, $F = AB$, $G = BA$, y $H = BC$ donde

$$A = \begin{bmatrix} 1 & 2 & 3 & 1 \\ 0 & 1 & 4 & 2 \\ 3 & 0 & 2 & 3 \end{bmatrix} \quad A' = \begin{bmatrix} 2 & 3 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 2 & 1 & 5 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 4 & 1 & 2 \\ 3 & 2 & 1 \\ 0 & 1 & 2 \\ 3 & 1 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 7 \\ 1 \\ 4 \end{bmatrix}$$

6.11) Calcule $E = B + CD$, donde

$$A = \begin{bmatrix} 3 & 2 & 1 \\ 0 & 4 & 3 \\ 0 & 0 & 6 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 2 \\ -1 & 1 & 0 \\ 0 & 3 & 2 \end{bmatrix} \quad D = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 5 & 2 & 7 \end{bmatrix}$$

6.12) Calcule la inversa de

$$A = \begin{bmatrix} 7 & 1 \\ 4 & 5 \end{bmatrix}$$

6.13) Calcule la inversa de

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 4 & 5 \\ 2 & 1 & 2 \\ 8 & 1 & 1 \end{bmatrix}$$

6.14) Encuentre la inversa de la siguiente matriz:

$$\begin{bmatrix} 3 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

6.15) Encuentre la inversa de la siguiente matriz usando el pivoteo:

$$\begin{bmatrix} 0 & 5 & 1 \\ -1 & 6 & 3 \\ 3 & -9 & 5 \end{bmatrix}$$

6.16) Descomponga las siguientes matrices en matrices L y U mediante una calculadora y verifique después la descomposición calculando el producto LU.

a) $\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$

b) $\begin{bmatrix} 2 & -1 & 0 \\ -3 & 4 & -1 \\ 0 & -1 & 2 \end{bmatrix}$

6.17) Resuelva las siguientes ecuaciones utilizando la descomposición LU.

a) $\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

b) $\begin{bmatrix} 2 & -1 & 1 \\ -3 & 4 & -1 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$

6.18) Encuentre el determinante de las siguientes matrices:

$$A = \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix}$$

$$B = \begin{bmatrix} 3 & 2 \\ 1 & 3 \end{bmatrix}$$

$$C = \begin{bmatrix} 4 & -1 & 2 \\ 1 & 2 & -3 \\ 0 & 3 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} -1 & 1 & 2 & -3 \\ 2 & -1 & 3 & 2 \\ 0 & 2 & 4 & 1 \\ 5 & 1 & 1 & -1 \end{bmatrix}$$

6.19) Calcule el determinante de

$$A = \begin{bmatrix} 8 & 1 & 3 & 2 \\ 2 & 9 & -1 & -2 \\ 1 & 3 & 2 & -1 \\ 1 & 0 & 6 & 4 \end{bmatrix}$$

que se puede descomponer como el producto de

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0.25 & 1 & 0 & 0 \\ 0.125 & 0.328 & 1 & 0 \\ 0.125 & -0.0143 & 2.545 & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} 8 & 1 & 3 & 2 \\ 0 & 8.75 & -1.75 & -2.5 \\ 0 & 0 & 2.2 & -0.4285 \\ 0 & 0 & 0 & 4.8052 \end{bmatrix}$$

6.20) Evalúe el determinante de A^{-1} donde

$$A = BCD$$

y

$$A = \begin{bmatrix} 3 & 2 & 1 \\ 0 & 4 & 3 \\ 0 & 0 & 6 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 2 \\ -1 & 1 & 0 \\ 0 & 3 & 2 \end{bmatrix} \quad D = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 5 & 2 & 7 \end{bmatrix}$$

6.21) Evalúe el determinante de la transpuesta de las matrices del problema anterior y muestre que el determinante de A es igual al determinante de A^t.

6.22) La matriz A es la matriz de Hilbert de 5×5 dada por

$$A = [a_{i,j}] \quad \text{donde} \quad a_{i,j} = \frac{1}{i+j-1}$$

Calcule (a) A^{-1}, (b) $A^{-1}A$, (c) $(A^{-1})^{-1}A^{-1}$

6.23) Desarrolle el determinante de la siguiente matriz en forma de un polinomio:

$$A = \begin{bmatrix} 2-s & 4 & 6 \\ 1 & -1-9 & 5 \\ 2 & 0 & 1-s \end{bmatrix}$$

6.24) Encuentre la solución general de

$$\begin{aligned} 4x + y - z &= 9 \\ 3x + 2y - 6z &= -2 \end{aligned}$$

6.25) Encuentre las variables básicas y libres de las siguientes ecuaciones. Encuentre después la solución general.

$$\begin{bmatrix} 1 & 3 & 3 & 2 \\ 2 & 6 & 9 & 5 \\ -1 & 3 & 3 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \\ 5 \end{bmatrix}$$

BIBLIOGRAFIA

Dogarra, J. J., J. R. Bunch, C. B. Moler y G. W. Stewart, *LINPACK User's Guide*, SIAM, 1979.

Forsythe, G. E. y C. B. Moler, *Computer Solution of Linear Algebra Systems*, Prentice-Hall, 1967.

Forsythe, G. E., M. A. Malcolm y C. B. Moler, *Computer Methods for Mathematical Computations*, Prentice-Hall, 1977.

Jennings, A., Matriz *Computations for Engineers and Scientists*, Wiley, 1977.

Lang, S., *Linear Algebra*, Springer-Verlag, 1987.

Morris, J. L., *Computational methods in Elementary Numerical Analysis*, Wiley, 1983.

Strang, G., *Linear Algebra and its Applications*, 2a. edición, Academic Press, 1980.

7
Cálculo de valores propios de una matriz

7.1 INTRODUCCION

Las ecuaciones lineales homogéneas se asocian con frecuencia a sistemas que presentan oscilaciones armónicas sin que haya fuerzas externas. En esta categoría están la vibración de una cuerda, una membrana u otros sistemas estructurales. Al estudiar la naturaleza o estabilidad dinámica de tales sistemas, es necesario resolver ecuaciones homogéneas (en particular, determinar los valores característicos de las ecuaciones homogéneas).

Las ecuaciones lineales homogéneas también son importantes en diversos análisis matemáticos. Por ejemplo, al resolver un sistema de ecuaciones diferenciales ordinarias es necesario determinar un conjunto de ecuaciones lineales homogéneas. Otro ejemplo es el hecho de que los métodos de solución numérica en ecuaciones diferenciales parciales están relacionados con los valores característicos de las ecuaciones lineales homogéneas.

Al principio de la sección 6.2 señalamos que un conjunto de ecuaciones lineales que posea al menos un valor distinto de cero en el lado derecho es un conjunto no homogéneo. Todas las ecuaciones lineales examinadas en el capítulo 6 son no homogéneas. Por otro lado, cuando el lado derecho de cada ecuación es igual a cero, el conjunto recibe el nombre de conjunto homogéneo. Por ejemplo,

$$\begin{aligned} 3x - 2y + z &= 0 \\ x + y + 2z &= 0 \\ 4x - y + 3z &= 0 \end{aligned} \tag{7.1.1}$$

La solución de un conjunto homogéneo es muy distinta del caso de las ecuaciones lineales no homogéneas. Para explicar la razón de esto, supongamos que existe una solución de la ecuación (7.1.1) y que se puede escribir como $x = a, y = b, z = c$. Entonces, $x = ka, y = kb, z = kc$, donde k es una constante arbitraria, también satisface la ecuación (7.1.1). Esto significa que podemos fijar una incógnita en un valor arbitrario, digamos $x = \beta$, y resolver el sistema en términos del resto de las incógnitas.

Sin embargo, si fijamos x en cierto valor arbitrario β, la ecuación (7.1.1) se transforma en

$$\begin{aligned} -2y + z &= -3\beta \\ +y + 2z &= -\beta \\ -y + 3z &= -4\beta \end{aligned} \tag{7.1.2}$$

Aquí tenemos tres ecuaciones con dos incógnitas. Si en una combinación diferente de dos ecuaciones se obtiene una solución diferente, no existe la solución del sistema en su totalidad. El conjunto de ecuaciones tiene una solución sólo si una de las tres ecuaciones es idéntica a otra, o bien es una combinación lineal de las demás (es decir, cuando se puede eliminar una ecuación sumando o restando múltiplos de las otras ecuaciones).

Si al menos una de las ecuaciones en la ecuación (7.1.2) es linealmene dependiente, el determinante de la matriz de coeficientes de la ecuación (7.1.1) se anula. Por lo tanto, la condición necesaria para que exista la solución de un conjunto homogéneo de ecuaciones lineales es que su determinante sea igual a cero. (La situación es la opuesta al caso del conjunto no homogéneo de ecuaciones, puesto que un conjunto no homogéneo de ecuaciones lineales tiene una solución única sólo si el determinante es distinto de cero.)

Para la ecuación (7.1.1), su determinante resulta ser igual a cero:

$$\det(A) = \det \begin{bmatrix} 3 & -2 & 1 \\ 1 & 1 & 2 \\ 4 & -1 & 3 \end{bmatrix} = 0 \tag{7.1.3}$$

y la solución se puede escribir como

$$\begin{aligned} x &= \beta \\ y &= 2\beta \\ z &= -\beta \end{aligned}$$

o, en forma equivalente,

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \beta \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix} \tag{7.1.4}$$

donde β es una constante arbitraria.

Una forma estándar de un conjunto de ecuaciones homogéneas (por ejemplo, con tres incógnitas) es la siguiente:

$$\begin{aligned}(a_{11} - \lambda)x_1 + a_{12}x_2 + a_{13}x_3 &= 0\\ a_{21}x_1 + (a_{22} - \lambda)x_2 + a_{23}x_3 &= 0\\ a_{31}x_1 + a_{32}x_2 + (a_{33} - \lambda)x_3 &= 0\end{aligned} \tag{7.1.5a}$$

donde λ es un *valor característico*, o *valor propio*. La ecuación (7.1.5a) se puede escribir en la forma equivalente

$$Ax = \lambda x \tag{7.1.5b}$$

donde

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

La ecuación (7.1.5a) o (7.1.5b) tiene una solución no trivial sólo en el caso en que el valor característico cumpla que

$$\det \begin{bmatrix} a_{11} - \lambda & a_{12} & a_{13} \\ a_{21} & a_{22} - \lambda & a_{23} \\ a_{31} & a_{32} & a_{33} - \lambda \end{bmatrix} = 0 \tag{7.1.6a}$$

o, de forma más compacta,

$$f(\lambda) \equiv \det (A - \lambda I) = 0 \tag{7.1.6b}$$

La función $f(\lambda)$ es una función característica y es un polinomio con respecto de λ. El orden de este polinomio es igual al orden de la matriz. Los valores propios son las raíces de la ecuación característica. La ecuación (7.1.5b) recibe el nombre de *problema de valores propios de una matriz*.

Una vez que se obtienen las soluciones de la ecuación característica, se puede calcular la solución de la ecuación homogénea para cada valor propio. Dicha solución recibe el nombre de *vector propio*.

Otra forma de los problemas de valores propios está dada por

$$Ax = \lambda Bx \tag{7.1.7}$$

donde A y B son matrices. La ecuación característica para la ecuación (7.1.7) se escribe como

$$f(\lambda) \equiv \det (A - \lambda B) = 0$$

Ejemplo 7.1

Consideremos un sistema vertical formado por masas y resortes. Las notaciones de la figura son:

$k_{0,1}$, $k_{1,2}$, $k_{2,3}$ y $k_{3,4}$: constantes de los resortes
m_i, $i = 1, 2, 3$: masas
y_i: desplazamiento de la masa i desde la posición estática

Al suponer que no hay fricción, las ecuaciones diferenciales para los desplazamientos de las masas son

$$\begin{aligned} m_1 \frac{d^2}{dt^2} y_1(t) &= -(k_{01} + k_{12})y_1 + k_{12}y_2 \\ m_2 \frac{d^2}{dt^2} y_2(t) &= k_{12}y_1 - (k_{12} + k_{23})y_2 + k_{23}y_3 \\ m_3 \frac{d^2}{dt^2} y_3(t) &= k_{23}y_2 - (k_{23} + k_{34})y_3 \end{aligned} \tag{A}$$

Obtenga el problema de valores propios asociado con una oscilación armónica (véase la figura E7.1). Suponga que todas las masas son idénticas y que $m_1 = m_2 = m_3 = m$.

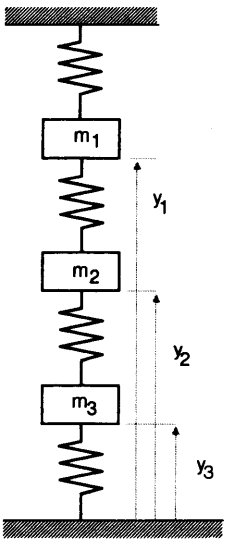


Figura E7.1 Sistema vertical de masas y resortes

(Solución)

Para una oscilación armónica, la solución se puede escribir como

$$y_i = \exp(j\omega t)f_i, \quad i = 1, 2, 3 \tag{B}$$

donde ω es una velocidad angular no determinada, f_i son las incógnitas y $j = \sqrt{-1}$. Al sustituir la ecuación (B) en la ecuación (A) obtenemos

$$\begin{aligned} -\omega^2 f_1 &= -(1/m)(k_{01} + k_{12})f_1 + (1/m)k_{12}f_2 \\ -\omega^2 f_2 &= (1/m)k_{12}f_1 - (1/m)(k_{12} + k_{23})f_2 + (1/m)k_{23}f_3 \\ -\omega^2 f_3 &= (1/m)k_{23}f_2 - (1/m)(k_{23} + k_{34})f_3 \end{aligned} \tag{C}$$

donde las ecuaciones se dividen entre m. En notación matricial, la ecuación (C) se escribe como

$$Af - \lambda f = 0 \tag{D}$$

donde

$$\lambda = \omega^2 \tag{E}$$

$$A = \begin{bmatrix} (k_{01} + k_{12})/m, & -k_{12}/m, & 0 \\ -k_{12}/m, & (k_{12} + k_{23})/m, & -k_{23}/m \\ 0 & -k_{23}/m, & (k_{23} + k_{34})/m \end{bmatrix} \tag{F}$$

y

$$f = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \tag{G}$$

Al calcular los valores propios de una matriz debemos tomar en cuenta lo siguiente:

a) Todos los valores propios de una matriz simétrica son reales.
(Si todos los valores propios de una matriz simétrica son positivos, se dice que la matriz es *positiva definida*.)

Tabla 7.1 Métodos numéricos para el cálculo de valores propios

Método	Resultado	Real/complejo	Forma[a]	Comentarios
Método de interpolación	Polinomio	*R*, *C*	*A*, *B*	Interpolación de Newton hacia adelante (sólo para matrices pequeñas)
Método de potencias/ método de potencias inversas/método de potencias inversas con desplazamiento	Valores propios	*R*	*A*	Se calcula sólo un valor propio a la vez
Matriz de Householder/ tridiagonal	Valores propios	*R*	*A*	*Sólo para matrices simétricas*
Iteración de Householder/*QR*	Valores propios	*R*, *C*	*A*	*Matrices no simétricas*

[a] *A*: det $(A - \lambda I)$
B: det $(A - \lambda B)$

b) Una matriz no simétrica con entradas reales puede tener valores propios complejos, los cuales aparecen como parejas de complejos conjugados.

En el resto de este capítulo, nos centraremos en los métodos numéricos básicos para el cálculo de valores propios. Los métodos de solución que se analizan en este capítulo se resumen en la tabla 7.1. La bibliografía general para el cálculo de valores propios aparece al final de este capítulo.

7.2 METODO DE INTERPOLACION

Comenzaremos con el método de interpolación [Faddeeva], que es un algoritmo primitivo, pero fácil de comprender. En este enfoque, la función característica se reduce a una serie de potencias con respecto de λ. Luego, se determinan las raíces de la serie de potencias mediante el método de Bairstow (descrito en el capítulo 3).

El procedimiento de reducción consta, en este caso, de dos etapas:

a) Transformar la función característica en un polinomio de Newton hacia adelante.
b) Convertir el polinomio de Newton hacia adelante en una serie de potencias.

Para una matriz de orden N, la función característica es un polinomio de orden N. Como tal, si se construye una tabla de valores de $f(\lambda)$ mediante $N + 1$ valores de 2 con separación uniforme, entonces $f(\lambda)$ se puede expresar mediante el polinomio de interpolación de Newton hacia adelante de orden N (véase el capítulo 2):

$$f(\lambda) = g(s) = \sum_{n=0}^{N} \binom{s}{n} \Delta^n f_0 \tag{7.2.1}$$

con

$$f_i = f(\lambda_i), \quad i = 0, 1, 2, \ldots, N$$
$$s = (\lambda - \lambda_0)/\Delta\lambda$$

donde λ_i son valores de λ con separación uniforme, $\lambda_i = \lambda_{i-1} + \Delta\lambda$. Los valores de $f_i = f(\lambda_i)$, $i = 0, 1, 2, \ldots, N$ se evalúan mediante el cálculo directo de determinante de $(A - \lambda_i I)$ (véase la sección 6.8). Aunque el incremento $\Delta\lambda$ es arbitrario, los valores demasiado pequeños o demasiado grandes pueden provocar errores de redondeo al calcular la tabla de diferencias. Puesto que λ_0 también es arbitrario, lo igualamos a cero, con lo que s se transforma en

$$s = \lambda/\Delta\lambda$$

El coeficiente binomial $\binom{s}{n}$ se puede expresar como

$$\binom{s}{n} = \frac{s(s-1)(s-2)\cdots(s-n)}{n!}$$

$$= \sum_{i=1}^{n} c_{n,i} s^i, \quad n \geqslant 1 \tag{7.2.2}$$

donde los $c_{n,i}$ se llaman *coeficientes de Markov*. En la tabla 7.2 aparecen algunos de sus valores. Sustituimos la ecuación (7.2.2) en la ecuación (7.2.1) y reagrupamos términos para obtener

$$g(\lambda) = f_0 + \sum_{n=1}^{N} \sum_{i=1}^{n} c_{n,i} s^i \Delta^n f_0$$

$$= f_0 + \sum_{i=1}^{N} \left(\sum_{n=i}^{N} c_{n,i} \Delta^n f_0 \right) s^i$$

$$= f_0 + \sum_{i=1}^{N} b_i s^i \tag{7.2.3}$$

donde

$$b_i = \sum_{n=i}^{N} c_{n,i} \Delta^n f_0 \tag{7.2.4}$$

Así, si usamos $s = \lambda/\Delta\lambda$, la ecuación (7.2.3) se puede reescribir en términos de λ :

$$g(\lambda) = f_0 + \sum_{i=1}^{N} b_i \left(\frac{\lambda}{\Delta\lambda} \right)^i \tag{7.2.5}$$

Esta es la forma de la ecuación característica como serie de potencias que se desea. El PROGRAMA 7-1 transforma la ecuación característica en una serie de potencias.

Tabla 7.2 Coeficientes de Markov, $c_{n,i}$

	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$
$n=1$	1					
$n=2$	−0.5	0.5				
$n=3$	0.33333	−0.5	0.16666			
$n=4$	−0.25	0.45833	−0.25	0.04167		
$n=5$	0.2	−0.41667	0.29167	−0.08333	0.00833	
$n=6$	−0.166666	0.38056	−0.31250	0.11806	−0.02083	0.00139

Este método también se puede aplicar a la ecuación característica en la forma det $(A - \lambda B) = 0$.

Ejemplo 7.2

Determinar la serie de potencias de la siguiente ecuación característica mediante un polinomio de interpolación de Newton:

$$f(\lambda) = \det \begin{bmatrix} 3-\lambda & 4 & -2 \\ 3 & -1-\lambda & 1 \\ 2 & 0 & 5-\lambda \end{bmatrix}$$

Calcule después los valores característicos mediante el método de Bairstow.

(Solución)

Se calculan los valores de $f(\lambda)$ para cuatro valores distintos de λ, a saber, 0, 0.5, 1.0, 1.5, evaluando directamente el determinante:

$$\lambda = 0 \quad : \text{determinante} = -71$$
$$\lambda = 0.5: \text{determinante} = -68.875$$
$$\lambda = 1.0: \text{determinante} = -64$$
$$\lambda = 1.5: \text{determinante} = -57.125$$

La tabla de diferencias hacia adelante es

i	λ_i	f_i	Δf_i	$\Delta^2 f_i$	$\Delta^3 f_i$
0	0	−71	2.125	2.75	−0.75
1	0.5	−68.875	4.875	2.0	
2	1.0	−64	6.875		
3	1.5	−57.125			

Utilizamos las diferencias en todo el primer renglón; la fórmula de interpolación de Newton hacia adelante se escribe como

$$g(\lambda) = -71 + 2.125s + \frac{2.75}{2}s(s-1) - \frac{0.75}{6}s(s-1)(s-2)$$

donde $s = \lambda/0.5$. Usamos los coeficientes de Markov para transformar esta ecuación en

$$g(\lambda) = -71 + [(1)(2.125) + (-0.5)(2.75) + (0.333333)(-0.75)]s$$
$$+ [(0.5)(2.75) + (-0.5)(-0.75)]s^2$$
$$+ [(0.166666)(-0.75)]s^3$$

Sustituimos $s = \lambda/0.5$ y reagrupamos los términos para obtener

$$g(\lambda) = -71 + \lambda + 7\lambda^2 - \lambda^3$$

Se calculan, mediante el PROGRAMA 3-7, las raíces de la ecuación anterior, las cuales son

$$4.875 \pm 1.431i, \quad -2.750$$

Resumen de Esta Sección

a) La función característica se transforma en una fórmula de interpolación de Newton, la cual a su vez se reescribe como una serie de potencias mediante los coeficientes de Markov.
b) Las raíces de la serie de potencias se calculan mediante el método de Bairstow.

7.3 METODO DE HOUSEHOLDER PARA UNA MATRIZ SIMETRICA

Dada una matriz simétrica, ésta se puede transformar en una matriz tridiagonal mediante el método de Householder, el cual consiste en una serie de transformaciones de similaridad. Los valores propios de una matriz tridiagonal se pueden calcular siguiendo el método de bisección.

En el resto de esta sección, analizaremos las dos etapas del método de Householder/bisección (la transformación de Householder y el método de bisección) para determinar los valores propios de una matriz tridiagonal simétrica.

7.3.1 Transformación de una matriz simétrica en una matriz tridiagonal

La matriz original A se denota ahora como $A^{(1)}$:

$$A^{(1)} = A = \begin{bmatrix} x & x & x & \cdot & x \\ x & x & x & \cdot & \cdot \\ x & x & x & \cdot & \cdot \\ x & x & x & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ x & x & x & \cdot & x \end{bmatrix} \tag{7.3.1}$$

El primer paso para reducir $A^{(1)}$ a una matriz tridiagonal es transformarla a la siguiente forma, que se denota como $A^{(2)}$:

$$A^{(2)} = \begin{bmatrix} x & x & 0 & \cdot & 0 \\ x & x & x & \cdot & x \\ 0 & x & x & \cdot & x \\ 0 & x & x & \cdot & x \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & x & x & \cdot & x \end{bmatrix} \tag{7.3.2}$$

La transformación se lleva a cabo multiplicando por la izquierda y por la derecha a A por una matriz de transformación P:

$$A^{(2)} = PA^{(1)}P \tag{7.3.3}$$

En la ecuación anterior, P se define mediante

$$P = I - \frac{uu^T}{h} \tag{7.3.4}$$

con

$$u = \text{col}\,(0,\ a_{2,1} + G,\ a_{3,1},\ a_{4,1},\ \ldots,\ a_{N,1}) \tag{7.3.5}$$

donde $a_{i,j}$ son los elementos de $A^{(1)}$ y

$$G = \left[\sum_{i=2}^{N} (a_{i,1})^2\right]^{1/2} \text{signo}\,(a_{2,1}) \tag{7.3.6}$$

$$h = G^2 + Ga_{2,1} \tag{7.3.7}$$

Donde signo $(a) = +1$ si $a \geqslant 0$, o bien signo $(a) = -1$ si $a < 0$. La matriz de transformación P tiene las siguientes propiedades:

$$\begin{aligned} P &= P^{-1} \\ P^T &= P \\ PP &= I \end{aligned} \tag{7.3.8}$$

La matriz A de orden N se reduce a una matriz tridiagonal repitiendo las transformaciones de este tipo $N - 2$ veces. La matriz $A^{(m)}$ tiene una submatriz principal de orden m en la forma tridiagonal en la esquina superior izquierda. En general, la transformación de $A^{(m)}$ en $A^{(m+1)}$ se escribe como

$$A^{(m+1)} = PA^{(m)}P \tag{7.3.9}$$

donde

$$P = I - \frac{uu^T}{h} \tag{7.3.10}$$

$$u = \text{col}\,(0, 0, 0, \ldots, 0,\ a_{m+1,m} + G,\ a_{m+2,m},\ \ldots,\ a_{N,m}) \tag{7.3.11}$$

donde $a_{i,j}$ son elementos de $A^{(m)}$ y

$$G = \left[\sum_{i=m+1}^{N} (a_{i,m})^2\right]^{1/2} \text{signo}\,(a_{m+1,m}) \tag{7.3.12}$$

$$h = G^2 + Ga_{m+1,m} \tag{7.3.13}$$

La matriz de transformación P de cada paso satisface la ecuación (7.3.8).

7.3.2 Valores propios de una matriz tridiagonal

La matriz tridiagonal denotada como M es simétrica y se escribe como

$$M = \begin{bmatrix} a_1, & b_1 & \cdot & \cdot & \cdot \\ b_1, & a_2, & b_2 & \cdot & \cdot \\ \cdot & b_2, & a_3, & b_3 & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & b_{N-1}, & a_N \end{bmatrix} \tag{7.3.14}$$

Definimos una sucesión de polinomios como

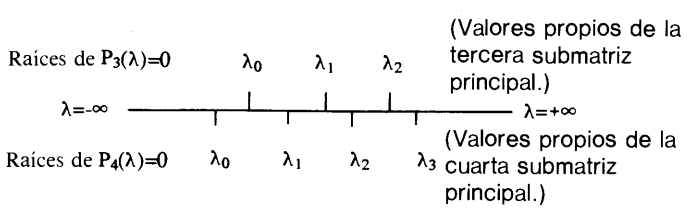
$$\begin{aligned} p_0(\lambda) &= 1 \\ p_1(\lambda) &= a_1 - \lambda \\ p_2(\lambda) &= [a_2 - \lambda]p_1(\lambda) - [b_1]^2 \\ &\vdots \\ p_i(\lambda) &= [a_i - \lambda]p_{i-1}(\lambda) - [b_{i-1}]^2 p_{i-2}(\lambda) \\ &\vdots \end{aligned} \tag{7.3.15}$$

Conviene observar que p_i en la ecuación (7.3.15) es el determinante de la submatriz principal de M, que incluye desde el primer elemento sobre la diagonal hasta el i-ésimo elemento sobre la misma. Para el caso de una matriz tridiagonal de orden N, la ecuación característica que se debe resolver es $p_N(\lambda) = 0$.

La sucesión de polinomios definida en la ecuación (7.3.15) tiene una propiedad importante: una raíz de $p_k(\lambda)$ siempre separa a una pareja de raíces consecutivas de $p_{k+1}(\lambda)$. En otras palabras, cada raíz de $p_{k+1}(\lambda)$ está entre dos raíces consecutivas de $p_k(\lambda)$, excepto las raíces máxima y mínima del primer polinomio. La figura 7.1 ilustra la relación entre las raíces de polinomios consecutivos. Debido a esta propiedad, se pueden calcular las raíces de $p_{k+1}(\lambda)$ mediante el método de bisección.

El PROGRAMA 7-2 transforma una matriz simétrica en una matriz tridiagonal y calcula después todos los valores propios mediante el método de bisección.

Figura 7.1 Relación entre las raíces de $p_k(\lambda) = 0$



Ejemplo 7.3

Calcule los valores propios de la siguiente matriz mediante el método de Householder/bisección:

$$A^{(1)} = A = \begin{bmatrix} 3 & 1 & 4 \\ 1 & 7 & 2 \\ 4 & 2 & 0 \end{bmatrix}$$

(Solución)

Puesto que $N = 3$, sólo se requiere una transformación. Los elementos de u dados en la ecuación (7.3.11) son los siguientes:

$$\begin{aligned} u_1 &= 0 \\ G &= \sqrt{1^2 + 4^2}\,\text{signo}(1) = 4.1231 \\ a_{2,1} &= 1 \\ h &= (4.1231)^2 + (4.1231)(1) = 21.1231 \\ u_2 &= a_{2,1} + \sqrt{17} = 1 + 4.1231 = 5.1231 \\ u_3 &= a_{3,1} = 4 \end{aligned}$$

La ecuación (7.3.11) queda entonces como

$$u = \text{col}\,[0, 5.123106, 4]$$

Sustituimos u y h en la ecuación (7.3.14) para obtener la matriz transformada:

$$\begin{aligned} A^{(2)} &= \left(I - \frac{uu^T}{h}\right) A^{(1)} \left(I - \frac{uu^T}{h}\right) \\ &= \begin{bmatrix} 3.0000 & -4.1231 & 0.0 \\ -4.1231 & 1.3529 & 3.4118 \\ 0.0 & 3.4118 & 5.6471 \end{bmatrix} \end{aligned}$$

El valor propio de la prmera submatriz principal es 3.000, como se puede ver inmediatamente. La segunda submatriz principal es

$$\begin{bmatrix} 3.0000 & -4.1231 \\ -4.1231 & 1.3529 \end{bmatrix}$$

Sus valores propios —calculados mediante el método de bisección— son —2.0280 y 6.3810. (Note que los dos valores propios están separados por el valor propio de la primera submatriz principal, 3.000.) Los valores propios de la matriz tridiagonal completa son —2.8941, 4.3861 y 8.508.

Resumen de Esta Sección

a) El método de Householder es una serie de transformaciones de similaridad que cambian una matriz simétrica a una forma tridiagonal simétrica.

b) Los valores propios de la matriz tridiagonal se obtienen mediante el método de bisección.

7.4 METODOS DE POTENCIAS

Existen dos razones por las que los métodos de potencias son importantes. La primera es que éstos son un medio sencillo para el cálculo de los valores propios. La segunda es que están relacionados estrechamente con la iteración QR que se analiza en la sección siguiente.

Los métodos de potencias tienen tres versiones. La primera es el método de potencia *regular*, que se basa en la potencia de la matriz y determina el máximo valor propio mediante iteraciones. La segunda es el método de potencias *inversas*, que se basa en la potencia inversa de la matriz y encuentra el mínimo valor propio. La tercera es el método de potencias inversas *con desplazamiento*. En el resto de esta sección, llamaremos al primero simplemente *método de potencias* y al segundo como *método de potencias inversas*.

Método de potencias. Consideremos una matriz A de $N \times N$. Los valores y vectores propios satisfacen la ecuación

$$Au_i = \lambda_i u_i \tag{7.4.1}$$

donde λ_i es el i-ésimo valor propio y u_i es el i-ésimo vector propio. Si A es una matriz simétrica, entonces todos los valores propios son reales. Si A no es simétrica, algunos valores propios pueden ser complejos. Supondremos que el máximo valor propio es real y aislado (es decir, no es un valor propio doble) y que los valores propios están numerados en orden creciente,

$$|\lambda_1| \leqslant |\lambda_2| \cdots \leqslant |\lambda_{N-1}| < |\lambda_N| \tag{7.4.2}$$

El método de potencias comienza con una estimación inicial del vector propio, $u^{(0)}$, que puede ser cualquier vector no nulo. La primera aproximación iterativa es

$$u^{(1)} = Au^{(0)}$$

y las iteraciones subsecuentes son

$$u^{(k+1)} = \frac{1}{\lambda^{(k)}} Au^{(k)} \tag{7.4.3}$$

con

$$\lambda^{(k)} = \frac{(u^{(k)}, u^{(k)})}{\left(u^{(k)}, \dfrac{1}{\lambda^{(k-1)}} u^{(k-1)}\right)} \tag{7.4.4}$$

donde (a, b) denota el producto escalar de dos vectores a y b, y k es el número de la iteración. Al continuar el proceso de iteración, $\lambda_i^{(k)}$ converge al máximo valor propio y $u^{(k)}$ converge al vector propio correspondiente.

Ejemplo 7.4

Determine el máximo valor propio y el vector propio correspondiente a la siguiente matriz, utilizando el método de potencias:

$$A = \begin{bmatrix} 3 & 1 & 4 \\ 1 & 7 & 2 \\ 4 & 2 & 0 \end{bmatrix}$$

(Solución)

Damos una estimación inicial de

$$u^{(0)} = \text{col}\,(1\ 1\ 1)$$

En la tabla 7.3 se muestra la solución iterativa después de cada ciclo de iteración.

Tabla 7.3 Solución iterativa

k	$\lambda^{(k)}$	Tres elementos de $u^{(k)}$		
1	8.333333	0.9600000	1.200000	0.7200000
2	8.453462	0.8233314	1.277583	0.7381592
3	8.492605	0.7889469	1.323826	0.6886570
4	8.503810	0.7579303	1.344461	0.6824518
5	8.506876	0.7462269	1.355852	0.6724730
10	8.507998	0.7319254	1.366952	0.6655880
15	8.508000	0.7314215	1.367357	0.6653084
18	8.508000	0.7314049	1.367370	0.6652992
19	8.508000	0.7314036	1.367371	0.6652986
20	8.508000	0.7314029	1.367372	0.6652982

El valor propio converge después de 15 pasos de iteración, aproximadamente. La convergencia del vector propio es más lenta que la del valor propio.

Ahora analizaremos la razón por la que el método de potencias converge. El vector inicial se puede desarrollar en términos de los vectores propios de A,

$$u^{(0)} = \sum_{i=1}^{N} a_i u_i \tag{7.4.5}$$

donde a_i son los coeficientes del desarrollo y u_i es el i-ésimo vector propio de A. Sustituimos la ecuación (7.4.5) en la ecuación (7.4.3) para obtener

$$u^{(k)} = \frac{(\lambda_N)^k}{\lambda^{(1)} \cdots \lambda^{(k-1)}} \left[\left(\frac{\lambda_1}{\lambda_N} \right)^k u_1 + \left(\frac{\lambda_2}{\lambda_N} \right)^k u_2 + \cdots u_N \right) \Bigg] \tag{7.4.6}$$

Al crecer k, todos los términos de los paréntesis cuadrados de la ecuación (7.4.6) se anulan, excepto u_N. Si $u^{(k)}$ converge a u_N, es fácil ver que la ecuación (7.4.4) converge a λ_N.

Antes de terminar con la explicación del método de potencias, debemos señalar que la ecuación (7.4.3) se puede escribir como

$$u^{(k+1)} = \frac{1}{\prod_{l=1}^{k} \lambda^{(l)}} A^{k+1} u^{(0)} \tag{7.4.7}$$

Por lo tanto, la esencia del método de potencias es multiplicar la estimación inicial por una potencia de A. El factor $1/\lambda^{(l)}$ normaliza los vectores de la iteración. Si no se normalizaran, la magnitud de los vectores podría aumentar o disminuir en forma no acotada y causar desbordamientos.

Método de potencias inversas. Este método es idéntico al anterior, excepto por el hecho de que utiliza la inversa A^{-1} en vez de A. Puesto que los valores propios de A^{-1} son los recíprocos de A, el método de potencias aplicado a A^{-1} calculará el mínimo valor propio de A. Por supuesto, debemos suponer que este mínimo valor propio es real y aislado:

$$|\lambda_1| < |\lambda_2| \cdots \leqslant |\lambda_{N-1}| \leqslant |\lambda_N| \tag{7.4.8}$$

En caso contrario, el método no funciona.

El primer paso de iteración es

$$Au^{(1)} = u^{(0)} \tag{7.4.9}$$

o, en forma equivalente,

$$u^{(1)} = A^{-1} u^{(0)} \tag{7.4.10}$$

donde $u^{(0)}$ es un vector no nulo, dado como estimación inicial. Los siguientes pasos de iteración son

$$Au^{(k+1)} = \lambda^{(k)} u^{(k)} \tag{7.4.11}$$

con

$$\lambda^{(k)} = \frac{(u^{(k)}, \lambda^{(k-1)} u^{(k-1)})}{(u^{(k)}, u^{(k)})} \tag{7.4.12}$$

Las ecuaciones (7.4.10) y (7.4.11) se pueden evaluar en forma directa, utilizando A^{-1} cuando la matriz es pequeña. Sin embargo, si la matriz A es grande y poco densa (o esparcida), se utiliza la eliminación de Gauss o la descomposición LU en cada ciclo de iteración, en vez de guardar A^{-1}.

Es fácil explicar por qué converge el método de potencias inversas. Puesto que el vector inicial se puede desarrollar como en la sección (7.4.5), al sustituir ésta en la ecuación (7.4.11) obtenemos

$$u^{(k)} = \frac{\lambda^{(1)} \cdots \lambda^{(k-1)}}{(\lambda_1)^k} \left[u_1 + \left(\frac{\lambda_1}{\lambda_2}\right)^k u_2 + \cdots + \left(\frac{\lambda_1}{\lambda_N}\right)^k u_N \right]$$

Puesto que λ_1 es el mínimo valor propio, todos los términos de los paréntesis cuadrados de la ecuación anterior tienden a cero al crecer k, excepto por u_1.

Método de potencias inversas con desplazamiento. Este método también se conoce como el método de Wielandt [Wachspress] y puede determinar cualquier vector y valor propio siempre que éste sea real y aislado. La esencia del método es calcular el valor propio de una matriz desplazada dada por

$$A' = A - \alpha I$$

El valor propio de A' está desplazado de los valores propios de A por α, es decir,

$$\lambda_i' = \lambda_i - \alpha$$

Por lo tanto, si se aplica el método de potencias inversas de A', el vector de iteración converge al valor propio λ_i' que es más cercano a cero. Así, se calcula el valor propio de A que se desee haciendo α igual a una estimación inicial de dicho valor. Este método se aplica en las secciones 10.7 y 10.8.

7.5 ITERACION *QR*

Es una secuencia iterativa de transformaciones de similaridad. Cada paso de la iteración consiste en la descomposición de la matriz en la forma QR y en la transformación de similaridad. Si denotamos a la matriz inicial por $A_0 = A$, donde A es la matriz original de la cual deseamos calcular los valores. La matriz A_0 se descompone en

$$A_0 = Q_0 R_0 \tag{7.5.1}$$

donde Q_0 es una matriz ortonormal y R_0 es una matriz triangular superior. La transformación de similaridad se escribe como

$$A_1 = Q_0^{-1} A_0 Q_0$$

o, en forma equivalente,

$$A_1 = R_0 Q_0 \tag{7.5.2}$$

Los siguientes pasos son esencialmente idénticos y se escriben como

$$\begin{aligned} A_k &= Q_k R_k \\ A_{k+1} &= R_k Q_k \end{aligned} \tag{7.5.3}$$

Para mejorar la eficiencia de los cálculos, se hacen dos modificaciones a las ecuaciones de (7.5.3). La primera es que el proceso de iteración se cambia a

$$\begin{aligned} A_k - \alpha_k I &= Q_k R_k \\ A_{k+1} &= R_k Q_k + \alpha_k I \end{aligned} \tag{7.5.4}$$

Este cambio recibe el nombre de *desplazamiento*, debido a que al restar $\alpha_k I$ a A_k se desplazan los valores propios del lado derecho una longitud igual a α_k; lo mismo ocurre con los valores propios de $R_k Q_k$. Si sumamos $\alpha_k I$ a la segunda ecuación, desplazamos de nuevo los valores propios de A_{k+1} hacia los valores originales. Sin embargo, los desplazamientos aceleran la convergencia de los valores propios cercanos a α_k. La segunda modificación se refiere al hecho de que, en vez de aplicar la descomposición a la matriz original, primero se transforma ésta a la forma de Hessenberg. Cuando A_0 está en la forma de Hessenberg, las demás A_k también tienen la misma forma. Esta descomposición se puede llevar a cabo mediante el algoritmo de Gram-Schmit, pero se utiliza un proceso más eficiente. Véase [Morris] para la explicación acerca de la convergencia de la iteración QR. Para la programación, se recomienda también [Martin, Peters y Wilkinson y Shoup].

Si se aplica el método de Householder al caso de una matriz no simétrica, ésta se reduce a la forma

$$\begin{bmatrix} x & x & x & x & \cdots & x \\ x & x & x & x & & x \\ 0 & x & x & x & & x \\ 0 & 0 & x & x & & x \\ 0 & 0 & 0 & x & & x \\ \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix} \tag{7.5.5}$$

que se llama una forma superior de Hessenberg. El algoritmo de la transformación de Householder analizado en la sección anterior funciona sin modificaciones.

La iteración QR transforma cualquier matriz real con forma de Hessenberg a una matriz triangular superior por bloques, como

$$\begin{bmatrix} D & x & x & x & x & x & x \\ 0 & D & x & x & x & x & x \\ 0 & 0 & D & D & x & x & x \\ 0 & 0 & D & D & x & x & x \\ 0 & 0 & 0 & 0 & D & D & x \\ 0 & 0 & 0 & 0 & D & D & x \\ 0 & 0 & 0 & 0 & 0 & 0 & D \end{bmatrix} \tag{7.5.6}$$

donde D y x representan elementos no nulos: se puede pensar en D como perteneciente a un bloque diagonal de 1×1 o de 2×2 (submatrices cuadradas). La matriz completa tiene forma de matriz triangular superior por bloques.

Los valores propios de una matriz triangular superior por bloques son iguales a los de la matriz diagonal por bloques, la cual se obtiene haciendo iguales a cero todos los elementos por arriba de los bloques de la diagonal:

$$\begin{bmatrix} D & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & D & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & D & D & 0 & 0 & 0 \\ 0 & 0 & D & D & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & D & D & 0 \\ 0 & 0 & 0 & 0 & D & D & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & D \end{bmatrix} \tag{7.5.7}$$

Los valores propios de una matriz diagonal por bloques son iguales a los de los bloques diagonales. Los elementos individuales sobre la diagonal (como los indicados por D en la primera, segunda y última posiciones) son valores propios en sí mismos. Cada submatriz diagonal de 2×2 tiene una pareja de valores propios reales o complejos conjugados. Los valores propios de los bloques de 2×2 se calculan como las raíces de un polinomio cuadrático.

El PROGRAMA 7-3 contiene la transformación de Householder y la iteración QR, por lo que encuentra al mismo tiempo todos los valores propios de una matriz no simétrica.

Ejemplo 7.5

Determine los valores propios de la siguiente matriz mediante la iteración de Householder/QR:

$$\begin{bmatrix} 5.3 & 2.3 & 4.6 & 2.7 & 1.6 & 2.2 \\ 2.4 & 7.8 & 5.7 & 8.4 & 3.4 & 4.2 \\ 3.4 & 5.6 & 2.4 & 1.7 & 7.4 & 3.9 \\ 8.3 & 7.5 & 9.2 & 6.1 & 5.2 & 7.9 \\ 4.3 & 5.9 & 7.2 & 2.6 & 4.9 & 0.8 \\ 0.9 & 2.7 & 4.9 & 4.8 & 6.7 & 4.8 \end{bmatrix}$$

(Solución)

La transformación de Householder reduce la matriz anterior a

$$\begin{bmatrix} 5.3000\text{E}+00 & -5.1043\text{E}+00 & 2.4620\text{E}+00 & 2.5047\text{E}+00 & -1.4576\text{E}+00 & -7.9177\text{E}-01 \\ -1.0272\text{E}+01 & 1.9500\text{E}+01 & -1.2706\text{E}+01 & -3.5993\text{E}+00 & 3.9338\text{E}+00 & 4.9378\text{E}+00 \\ 0.0 & -1.1336\text{E}-01 & 4.3307\text{E}+01 & -4.1250\text{E}+02 & -1.1056\text{E}+00 & -8.6185\text{E}-01 \\ 0.0 & 0.0 & 1.9893\text{E}+00 & -3.8121\text{E}+00 & -5.4740\text{E}+01 & 3.2432\text{E}+01 \\ 0.0 & 0.0 & 0.0 & 2.4664\text{E}+00 & 3.0625\text{E}+00 & 3.9123\text{E}+00 \\ 0.0 & 0.0 & 0.0 & 0.0 & -3.0033\text{E}+00 & 2.9188\text{E}+00 \end{bmatrix}$$

La transformación QR transforma ahora esta matriz en

$$\begin{bmatrix} 28.3953 & -1.0980 & -4.5150 & 4.5051 & -4.6998 & 2.4398 \\ 0.0 & -2.7794 & -0.4306 & 1.9669 & 2.4996 & 0.0536 \\ 0.0 & 2.1144 & -3.3709 & -4.5749 & -0.3379 & 0.9949 \\ 0.0 & 0.0 & 0.0 & 3.1160 & 0.7233 & 0.1572 \\ 0.0 & 0.0 & 0.0 & 0.0 & 3.3789 & -3.7079 \\ 0.0 & 0.0 & 0.0 & 0.0 & 3.3854 & 2.5600 \end{bmatrix}$$

En la matriz anterior se tienen dos bloques diagonales de 2×2:

$$\begin{bmatrix} -2.7794 & -0.4306 \\ 2.1144 & -3.3709 \end{bmatrix} \text{ y } \begin{bmatrix} 3.3789 & -3.7079 \\ 3.3854 & 2.56 \end{bmatrix}$$

Los valores propios de estas submatrices son $-3.0751 \pm 0.9142j$ y $2.9694 \pm 3.5193j$ = donde $j\sqrt{-1}$. Los demás elementos sobre la diagonal son valores propios. Por lo tanto, los valores propios de la matriz dada son

$$28.3953,\ -3.0751 \pm 0.9142j,\ 3.1160,\ 2.9694 \pm 3.5193j$$

Una aplicación importante de la iteración QR es el cálculo de raíces de un polinomio en forma de serie de potencias.

La siguiente forma de una matriz se llama *matriz de Frobenius*:

$$P = \begin{bmatrix} -p_{N-1} & -p_{N-2} & -p_{N-3} & \cdots & -p_0 \\ 1 & 0 & 0 & & \\ 0 & 1 & 0 & & \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ 0 & & & 1 & 0 \end{bmatrix} \tag{7.5.8}$$

Su función característica se escribe como

$$f(\lambda) = \det \begin{bmatrix} -p_{N-1} - \lambda & -p_{N-2} & -p_{N-3} & \cdots & -p_0 \\ 1 & -\lambda & 0 & & \\ 0 & 1 & -\lambda & & \\ \cdot & \cdot & \cdot & \cdots & \\ 0 & & & 1 & -\lambda \end{bmatrix} \tag{7.5.9}$$

El desarrollo de la ecuación (7.5.9) en serie de potencias da como resultado

$$\begin{aligned} f(\lambda) &= |P - \lambda I| \\ &= (-1)^N[\lambda^N + p_{N-1}\lambda^{N-1} + \cdots + p_0] \end{aligned} \tag{7.5.10}$$

Por otro lado, cualquier ecuación polinomial

$$a_N x^N + \cdots + a_2 x^2 + a_1 x + a_0 = 0 \tag{7.5.11}$$

se puede expresar en la forma de la ecuación (7.5.8) con las definiciones

$$\begin{aligned} p_0 &= a_0/a_N \\ p_1 &= a_1/a_N \\ p_2 &= a_2/a_N \\ &\vdots \\ p_{N-1} &= a_{N-1}/a_N \end{aligned} \tag{7.5.12}$$

La matriz de Frobenius tiene la forma de Hessenberg. Por lo tanto, se pueden calcular sus valores propios mediante la iteración QR. La matriz de Frobenius se puede utilizar directamente como entrada para un programa, como el PROGRAMA 7-3.

Ejemplo 7.6

Calcule las raices de la siguiente ecuación polinomial mediante la iteración QR:

$$f(x) = x^5 - 0.2x^4 + 7x^3 + x^2 - 3.5x + 2 = 0$$

(Solución)

Pensemos en $f(x)$ como la función característica para una matriz de Frobenius:

$$M = \begin{bmatrix} 0.2 & -7 & -1 & 3.5 & -2 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Utilizamos el PROGRAMA 7-3 para determinar los valores propios. Los resultados son

$$-0.9085, \quad 0.1403 \pm 2.7314i, \quad 0.4139 \pm 0.3506i$$

El lector puede verificar estos resultados mediante el método de Bairstow, PROGRAMA 3-7.

Resumen de Esta Sección

a) La iteración QR transforma una matriz con forma de Hessenberg en la forma triangular superior por bloques, la cual consta a lo más de bloques de 2×2.

b) Se encuentra una pareja de valores propios, reales o complejos conjugados, para cada bloque de 2×2 en la posición diagonal. Los elementos en la diagonal que no forman parte de un bloque de 2×2 son valores propios reales.

c) Una aplicación de la iteración QR es el cálculo de raíces de un polinomio. Esto es posible debido a que un polinomio se puede transformar en una matriz de Frobenius, la cual tiene la forma de Hessenberg, por lo que la iteración QR se puede aplicar de manera directa. Los valores propios de la matriz de Frobenius son las raíces del polinomio.

PROGRAMAS

PROGRAMA 7-1 Método de interpolación

A) Explicaciones

Este programa transforma la función característica de una matriz en una serie de potencias mediante el método de interpolación.

Antes de correr el programa, hay que definir en los enunciados DATA el orden de la matriz, N, así como los elementos de la matriz. Cuando se ejecuta el programa, se le da a la computadora en forma interactiva, el tamaño λ del intervalo. Después de esto, se calculan los N + 1 valores del determinante mediante la subrutina DETMNT y se almacenan en el arreglo FF(JJ). A continuación se calcula la tabla de diferencias y se almacena en DF(I, J). Los coeficientes del polinomio se calculan mediante los coeficientes de Markov, los que a su vez se calculan internamente cuando se necesitan. La precisión de los coeficientes del polinomio puede verse afectada por la entrada para el intervalo de lambda.

B) Variables

A(I, J): elementos de la matriz
N: orden de la matriz
DE: incremento de λ
RA(I): valores discretos de λ para los que se calcula $f(\lambda)$
DF(I, J): tabla de diferencias
CC(I): coeficiente de la potencia
MV(L): coeficiente de Markov

C) Listado

```
C-----CSL/F7-1.FOR        METODO DE INTERPOLACION (FORTRAN)
C
      DIMENSION A(20,20),FF(0:20),RA(0:20),DF(0:20,0:20), B(20,20)
      DIMENSION CC(0:20)
      REAL MV(0:20)
      PRINT *
      PRINT *,'CSL/F7-1        METODO DE INTERPOLACION (FORTRAN)
      PRINT *
C-----DEFINICION DE LA MATRIZ
      DATA N/4/                                  ! Defina el orden de la matriz
      DATA (B(1,J),J=1,4)/4,3,2,1/
      DATA (B(2,J),J=1,4)/3,3,2,1/
      DATA (B(3,J),J=1,4)/2,2,2,1/
      DATA (B(4,J),J=1,4)/1,1,1,1/
      PRINT *,'ORDEN DE LA MATRIZ =',N
      PRINT *,' MATRIZ '
             DO I=1,N
               PRINT 56, (B(I,J),J=1,N)
             END DO
```

```
1       PRINT *
        PRINT *,'DELTA LAMBDA ?'
        READ *,DE                ! Incremento de lambda para la tabla de diferencias
        DO JJ=0,N
           RA(JJ)=JJ*DE
           PRINT *
           PRINT *,'LAMBDA=   ',RA(JJ)
             DO J=1,N
               DO I=1,N
                  A(I,J)=B(I,J)
               END DO
               A(J,J)=A(J,J)-RA(JJ)
             END DO
           CALL DETMNT(N,A,S)
           FF(JJ)=S
        END DO
56      FORMAT(1X, 1P6E12.5)
C-- La siguiente parte calcula la tabla de diferencias hacia adelante
        DO I=0,N
           DF(I,0)=FF(I)         ! Inicialización de la tabla de diferencias
        END DO
        M=N
        DO J=1,N
           M=M-1
           DO I=0,M
              DF(I,J)=DF(I+1,J-1)-DF(I,J-1)          ! Tabla de diferencias
           END DO
        END DO
        PRINT *
        PRINT *,' TABLA DE DIFERENCIAS PARA LOS DETERMINANTES '
        DO I=0,N
            WRITE (*,'(F8.4, 1P6E11.3)') RA(I), (DF(I,J),J=0,N-I)
        END DO
C-- La siguiente rutina es para calcular los coeficientes de las potencias por medio de los coeficientes de Markov
        PRINT *
205     PRINT *,' COEFICIENTES DE MARKOV '
        DO I=0,N
           CC(I)=0
           MV(I)=0
        END DO
        MV(1)=1               ! Se inicializan los coeficientes de Markov
        CC(0)=DF(0,0)
        CC(1)=DF(0,1)
        DO K=2,N
           DO L=K,1,-1
             MV(L)=(MV(L-1)-(K-1)*MV(L))/K
             CC(L)=CC(L)+MV(L)*DF(0,K)        ! Coeficientes de Markov
           END DO
           PRINT 223, (MV(L),L=1,K)
223        FORMAT(1X, 6F12.7)
        END DO
        PRINT *
        PRINT *,'--- RESULTADO FINAL ---'
        PRINT *,' POTENCIA,   COEFICIENTES'
        DO I=0,N
           CC(I)=CC(I)/DE**I
           WRITE (6,'(2X,I3,5X,F10.4)')I, CC(I)
        END DO
        PRINT *
460     PRINT *,'-------------------------------------------------------'
```

```
500   PRINT *
      PRINT*
      PRINT*, ' OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR '
      READ *,K
      IF(K.EQ.1) GOTO 1
      PRINT*
      END
C*****************************
      SUBROUTINE DETMNT(N,A,DET)    ! Cálculo del determinante
      DIMENSION A(20,20)
      INTEGER PV,PC
      PC=0                          ! Inicialización del contador de pivoteo
      DO I=1,N-1                    ! Comienza la eliminación hacia atrás
         PV=I
         DO J=I+1, N
            IF (ABS(A(PV,I)) .LT. ABS(A(J,I))) PV=J
         END DO
         IF (PV.NE.I) THEN
           DO JC=1,N
              TM=A(I,JC)
              A(I,JC)=A(PV,JC)
              A(PV,JC)=TM
           END DO
           PC=PC+1
         END IF
         DO JR=I+1,N
           IF (A(JR,I).NE.0) THEN
              R=A(JR,I)/A(I,I)
              DO KC=I+1,N
                 A(JR,KC)=A(JR,KC)-R*A(I,KC)
              END DO
           END IF
         END DO
      END DO                                ! Fin de la eliminación hacia atrás
      IF (A(N,N).EQ.0) GOTO 1200
      DET=1                                 ! Inicialización del determinante
         DO I=1,N
            DET=DET*A(I,I)
         END DO
      IF (PC .NE. INT(PC/2)*2) DET=-DET
      PRINT *,' DETERMINANTE =      ',DET
      PRINT *,'NUMERO DE PIVOTEOS= ',PC
      RETURN
1200  PRINT *,' LA MATRIZ ES SINGULAR '
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F7-1        METODO DE INTERPOLACION (FORTRAN)

ORDEN DE LA MATRIZ =              4
MATRIZ
 4.00000E+00 3.00000E+00 2.00000E+00 1.00000E+00
 3.00000E+00 3.00000E+00 2.00000E+00 1.00000E+00
 2.00000E+00 2.00000E+00 2.00000E+00 1.00000E+00
 1.00000E+00 1.00000E+00 1.00000E+00 1.00000E+00
```

```
DELTA LAMBDA ?
0.5

LAMBDA=      0.0000000E+00
 DETERMINANTE =          1.000000
 NUMERO DE PIVOTEOS =            0

LAMBDA=      0.5000000
 DETERMINANTE =        6.2499966E-02
 NUMERO DE PIVOTEOS=             1

LAMBDA=      1.000000
 DETERMINANTE =       -8.9406981E-08
 NUMERO DE PIVOTEOS=             0

LAMBDA=      1.500000
 DETERMINANTE =        -4.437500
 NUMERO DE PIVOTEOS=             1

LAMBDA=      2.000000
 DETERMINANTE =        -17.00000
 NUMERO DE PIVOTEOS=             1

TABLA DE DIFERENCIAS PARA DETERMINANTES
 0.0000  1.000E+00 -9.375E-01  8.750E-01 -5.250E+00  1.500E+00
 0.5000  6.250E-02 -6.250E-02 -4.375E+00 -3.750E+00
 1.0000 -8.941E-08 -4.438E+00 -8.125E+00
 1.5000 -4.438E+00 -1.256E+01
 2.0000 -1.700E+01

COEFICIENTES DE MARKOV
  -0.5000000   0.5000000
   0.3333333  -0.5000000   0.1666667
  -0.2500000   0.4583333  -0.2500000   0.0416667

--- RESULTADO FINAL ---
POTENCIA,    COEFICIENTES
   0           1.0000
   1          -7.0000
   2          15.0000
   3         -10.0000
   4           1.0000
```

PROGRAMA 7-2 Householder/bisección

A) Explicaciones

Este programa calcula los valores propios de una matriz real simétrica transformando la matriz en una matriz tridiagonal simétrica y calculando en seguida los valores propios de ésta mediante el método de bisección.

El orden y los elementos de la matriz se especifican en los enunciados DATA. La reducción de la matriz propuesta a la forma tridiagonal se lleva a cabo en el programa principal por el esquema de Householder. Los enunciados en este proceso se pueden comparar fácilmente con las ecuaciones en el texto. Aquí, la matriz tridiagonal es un caso especial de la matriz de Hessenberg. Después de esto, se calculan los valores propios de la matriz tridiagonal. El determinante de una matriz tridiago-

nal se calcula utilizando la ecuación (7.3.15) en la subrutina BISEC. Los resultados finales se imprimen.

B) Variables

N: orden de la matriz
A(I, J): elementos de la matriz
S: s
SSR: $\sqrt{s}$
UAU: $s = u^T A u$
T(I, J): memoria temporal de matrices
EI(L, J): J-ésimo valor propio de la L-ésima matriz tridiagonal principal
XL: cota inferior para las estimaciones del valor propio
XH: cota superior para las estimaciones del valor propio

C) Listado

```
C-----CSL/F7-2.FOR      HOUSEHOLDER/TRIDIAGONAL'
      DIMENSION A(0:10,0:10),U(0:10),T(0:10,0:10)
      DIMENSION G(0:10),EI(0:10,0:10)
      COMMON L
3     PRINT *
      PRINT *,'CSL/F7-2          HOUSEHOLDER/TRIDIAGONAL'
      PRINT *,'                SOLO PARA MATRICES SIMETRICAS ='
      PRINT *
C-----------------------------------------------------------------------
C---ESTE PROGRAMA CALCULA LOS VALORES PROPIOS DE UNA MATRIZ SIMETRICA EN DOS
C   ETAPAS; ETAPA 1:
C              REDUCCION DE UNA MATRIZ A UNA FORMA TRIDIAGONAL MEDIANTE EL
C              ESQUEMA DE HOUSEHOLDER
C          ETAPA 2:
C              CALCULO DE LOS VALORES PROPIOS DE LA MATRIZ TRIDIAGONAL MEDIANTE EL
C              ESQUEMA DE BISECCION COMBINADO CON LA INTERPOLACION LINEAL
C-----------------------------------------------------------------------
20    N=4                                  ! Define el orden de la matriz
      DATA (A(1,J),J=1,4)/4, 3, 2, 1/      ! Matriz de N x N
      DATA (A(2,J),J=1,4)/3, 3, 2, 1/
      DATA (A(3,J),J=1,4)/2, 2, 2, 1/
      DATA (A(4,J),J=1,4)/1 ,1, 1, 1/
      PRINT *,'-------- REDUCCION DE UNA MATRIZ A LA FORMA DE HESSENBERG
      PRINT *,'         O TRIDIAGONAL MEDIANTE EL ESQUEMA DE HOUSEHOLDER'
      PRINT *
      PRINT *,' MATRIZ ORIGINAL (DEBE SER SIMETRICA) '
      PRINT *
      DO I=1,N
         PRINT 52,(A(I,J),J=1,N)
52       FORMAT(1X, 1P6E12.5)
      END DO
      DO IR=1,N-2                  ! Comienza el esquema de Householder.
        S=0
         DO I=1,N
```

```
            U(I)=0
            IF (I .GT.  IR+1) U(I)=A(I,IR)
            IF (I .GT.  IR)  S=S+A(I,IR)*A(I,IR)
         END DO
         W=1
         IF (A(IR+1,IR) .LT. 0) W=-1
         SSR=SQRT(S)
         PRINT *
         H=S+ABS(A(IR+1,IR))*SSR
         U(IR+1)=A(IR+1,IR)+SSR*W
         UAU=0
         DO I=1,N
            DO J=1,N
               UAU=UAU+U(I)*A(I,J)*U(J)
               IF ((I .LE. IR).AND.(J .LE. IR)) THEN
               T(I,J)=A(I,J)
               GOTO 710
               ENDIF
               IF ((J.EQ.IR).AND.(I .GE. IR+2))  THEN
                  T(I,J)=0
                  GOTO 710
               ENDIF
               B23=0
               DO K=1,N
                  B23=B23-(U(I)*A(K,J)+A(I,K)*U(J))*U(K)
               END DO
               T(I,J)=A(I,J)+B23/H
710         END DO
         END DO
         UAU=UAU/H/H
         DO I=1,N
            DO J=1,N
               A(I,J)=T(I,J)+UAU*U(I)*U(J)
               IF (ABS(A(I,J)) .LT. .000001) A(I,J)=0
            END DO
         END DO
      END DO
      PRINT *,' MATRIZ DE HESSENBERG O TRIDIAGONAL '
      PRINT *
      DO I=1,N
         PRINT 52,(A(I,J),J=1,4)        ! Imprime la matriz de Hessenberg
      END DO
      PRINT *
      PRINT *,'PARA CONTINUAR, OPRIMA 1 Y LA TECLA ENTER '
      READ*,DUMM
      PRINT *
      KM=N
      DO L=1,KM
         IF (L.EQ.1) THEN
            EI(1,1)=A(1,1)
         ELSE
            DO J=1,L
               XL=EI(L-1,J-1)
               XH=EI(L-1,J)
               KM=J
               CALL BISEC(G,A,XL,XH,XM)
               EI(L,J)=XM
            END DO
         END IF
         EI(L,0)=-99
```

```
          EI(L,L+1)=99
          IF (L.NE.N) THEN
             PRINT *
             PRINT 2130, L,L
2130         FORMAT(' VALORES PROPIOS DE LA ',I2,' X',I2,
     1               '  SUBMATRIZ PRINCIPAL   '          )
          ELSE
          PRINT *
          PRINT *,' RESULTADOS FINALES (VALORES PROPIOS DE LA MATRIZ COMPLETA) '
          PRINT *,'-----------------------------------------------------------'
          END IF
          PRINT 52,(EI(L,I),I=1,L)
       END DO
       PRINT *,'-----------------------------------------------------------'
       PRINT *
       PRINT *
       STOP
       END
C*************************************
       SUBROUTINE BISEC(G,A,XL,XH,XM)     ! Bisección
       DIMENSION A(0:10,0:10), G(0:10)    ! Calcula las raíces de un determinante
       COMMON L
       KA=0
       CALL  DETERM(G,A,XL,YL)
       CALL  DETERM(G,A,XH,YH)
80     KA=KA+1
       IF (KA .GT. 99) RETURN
       DX=XH-XL
       IF (DX  .LT.  .0000001) RETURN
       IF (DX .GT. 1) THEN
          XM=(XL+XH)/2                    ! Esquema de bisección
          CALL  DETERM(G,A,XM,YM)
          GOTO 30
       END IF
       XB=XM
       XM=(XL*YH-XH*YL)/(YH-YL)           ! Esquema de interpolación lineal
       CALL  DETERM(G,A,XM,YM)
       IF (ABS(XB-XM) .LT. .000001) RETURN
30     IF (YL*YM .LT. 0) THEN
          XH=XM
          YH=YM
          GOTO 80
       ENDIF
       XL=XM
       YL=YM
       GOTO 80
       END
C*************************************
       SUBROUTINE DETERM(G,A,X,SL)  ! Calcula el determinante de una matriz tridiagonal
       DIMENSION A(0:10,0:10),G(0:10)
       COMMON L
       G(0)=1
       IF (L.EQ.1)  RETURN
       G(1)=A(1,1)-X
       IF (L.EQ.1) RETURN
       DO K=2,L
         G(K)=(A(K,K)-X)*G(K-1)-A(K,K-1)*A(K,K-1)*G(K-2)
       END DO
       SL=G(L)
       RETURN
       END
```

D) Ejemplo de salida

```
CSL/F7-2          HOUSEHOLDER/TRIDIAGONAL
                (SOLO PARA MATRICES SIMETRICAS)
-------- REDUCCION DE UNA MATRIZ A LA FORMA DE HESSENBERG
         O TRIDIAGONAL MEDIANTE EL ESQUEMA DE HOUSEHOLDER

MATRIZ ORIGINAL (DEBE SER SIMETRICA)
 4.00000E+00 3.00000E+00 2.00000E+00 1.00000E+00
 3.00000E+00 3.00000E+00 2.00000E+00 1.00000E+00
 2.00000E+00 2.00000E+00 2.00000E+00 1.00000E+00
 1.00000E+00 1.00000E+00 1.00000E+00 1.00000E+00

 MATRIZ DE HESSENBERG O TRIDIAGONAL
 4.00000E+00-3.74166E+00 0.00000E+00 0.00000E+00
-3.74166E+00 5.00000E+00 4.62911E-01 0.00000E+00
 0.00000E+00 4.62911E-01 6.66666E-01-8.90870E-02
 0.00000E+00 0.00000E+00-8.90869E-02 3.33333E-01

VALORES PROPIOS DE LA SUBMATRIZ PRINCIPAL DE 1 × 1
 4.00000E+00

VALORES PROPIOS DE LA SUBMATRIZ PRINCIPAL DE 2 × 2
 7.25082E-01 8.27492E+00

VALORES PROPIOS DE LA SUBMATRIZ PRINCIPAL DE 3 × 3
 3.81085E-01 9.94722E-01 8.29086E+00

RESULTADOS FINALES (VALORES PROPIOS DE LA MATRIZ COMPLETA)
-----------------------------------------------------
 2.83250E-01 4.26021E-01 9.99996E-01 8.29086E+00
-----------------------------------------------------
```

PROGRAMA 7-3 Iteración *QR*

A) Explicaciones

El PROGRAMA 7-3 transforma una matriz dada (tanto simétrica como no simétrica) a la forma de Hessenberg mediante el método de Householder y luego encuentra todos los valores propios mediante la iteración *QR*.

La matriz se define en los enunciados DATA. El resto del programa se divide en dos partes. La primera es el método de Householder para reducir una matriz propuesta a la forma de Hessenberg; es idéntica al PROGRAMA 7-2. La segunda parte es la iteración *QR*. Los valores propios se imprimen como resultados finales.

B) Variables

N: orden de la matriz
A(I, J): elementos de la matriz
S: s
SSR: $\sqrt{s}$

UAU: $u^T Au$

RL(K): parte real del K-ésimo valor propio

IT: contador de las iteraciones

IM(K): parte imaginaria del K-ésimo valor propio

C) Listado

```
C-----CSL/F7-3.FOR     ITERACION DE HOUSEHOLDER/QR
C
      DIMENSION A(0:10,0:10),U(0:10),T(0:10,0:10)
      DIMENSION F(0:10),RL(0:10)
      REAL IM(0:10),MA
C       * Si se desea utilizar la doble precisión, escriba C en la columna 1 de las tres líneas anteriores
C       * y quite la C de la primera columna de las tres líneas siguientes
c       DOUBLE PRECISION A(0:10,0:10),U(0:10),T(0:10,0:10)
c       DOUBLE PRECISION F(0:10),RL(0:10),P,Q,R,S,W,X,Y,Z
c       REAL*16 IM(0:10),MA
      CHARACTER*7 G(0:10)
      PRINT *
      PRINT *,'CSL/F7-3        ITERACION DE HOUSEHOLDER/QR '
      PRINT *
C----- Definición de la matriz
      DATA  N/6/                ! Define el orden de la matriz.
      DATA  (A(1,J),J=1,6)/5.3, 2.3, 4.6, 2.7, 1.6, 2.2/
      DATA  (A(2,J),J=1,6)/2.4, 7.8, 5.7, 8.4, 3.4, 4.2/
      DATA  (A(3,J),J=1,6)/3.4, 5.6, 2.4, 1.7, 7.4, 3.9/
      DATA  (A(4,J),J=1,6)/8.3, 7.5, 9.2, 6.1, 5.2, 7.9/
      DATA  (A(5,J),J=1,6)/4.3, 5.9, 7.2, 2.6, 4.9, 0.8/
      DATA  (A(6,J),J=1,6)/0.9, 2.7, 4.9, 4.8, 6.7, 4.8/
      PRINT *,' ETAPA 1 - - REDUCCION DE UNA MATRIZ A LA FORMA DE HESSENBERG '
      PRINT *,'             O TRIDIAGONAL MEDIANTE EL ESQUEMA DE HOUSEHOLDER '
      PRINT *
      PRINT *,' MATRIZ ORIGINAL '
      PRINT *
      R=1
      DO I=1,N
         PRINT 192,(A(I,J),J=1,6)
      END DO
192   FORMAT(1X, 1P7E11.4)
      PRINT *
      DO 220 IR=1,N-2
        S=0
        DO I=1,N
          U(I)=0
          IF (I .GT.  IR+1) U(I)=A(I,IR)
          IF (I .GT.  IR)  S=S+A(I,IR)*A(I,IR)
        END DO
        W=1
        IF (A(IR+1,IR) .LT. 0) W=-1
        SSR=SQRT(S)
        H=S+ABS(A(IR+1,IR))*SSR
        U(IR+1)=A(IR+1,IR)+SSR*W
        UAU=0
        DO 275 I=1,N
          DO 280 J=1,N
```

```
          UAU=UAU+U(I)*A(I,J)*U(J)
            IF (I .LE. IR .AND. J .LE. IR) THEN
              T(I,J)=A(I,J)
              GOTO 280
              ENDIF
            IF (J.EQ.IR .AND. I .GE. IR+2) THEN
              T(I,J)=0
              GOTO 280
              ENDIF
            B23=0
            DO K=1,N
              B23=B23-(U(I)*A(K,J)+A(I,K)*U(J))*U(K)
            END DO
          T(I,J)=A(I,J)+B23/H
280       CONTINUE
275     CONTINUE
        UAU=UAU/H/H
        DO I=1,N
           DO J=1,N
             A(I,J)=T(I,J)+UAU*U(I)*U(J)
             IF (ABS(A(I,J)) .LT. .000001) A(I,J)=0
           END DO
        END DO
220   CONTINUE
      PRINT *,' MATRIZ DE HESSENBERG O TRIDIAGONAL '
      PRINT *
      DO I=1,N
         PRINT 390,(A(I,J),J=1,6)
      END DO
390   FORMAT(1X, 1P7E11.4)
      PRINT*
      PRINT*,'PARA CONTINUAR OPRIMA CUALQUIER NUMERO Y LA TECLA ENTER. '
      READ*, DUMMY
      PRINT *
      PRINT *
      PRINT *,' ETAPA 2 - - ITERACION QR PARA ENCONTRAR LOS VALORES PROPIOS '
      PRINT *
      MA=1.0                          ! Se calcula el épsilon de la máquina.
441   IF (1+MA.GT.1) THEN
         MA=MA/2
         GOTO 441
      END IF
      MA=(MA*2)**2                    ! Cuadrado del épsilon de la máquina.
      PRINT *, ' CRITERIO DE CONVERGENCIA =', MA
      NN=N
435   IF (NN.EQ.0) GOTO 765
      IT=0
      NA=NN-1
445   DO L=NN,2,-1
       IF(ABS(A(L,L-1)).LE.MA*(ABS(A(L-1,L-1))+ABS(A(L,L)))) GOTO 470
      END DO
      L=1
470   X=A(NN,NN)
      IF (L.EQ.NN) GOTO 705
      Y=A(NA,NA)
      R=A(NN,NA)*A(NA,NN)
      IF (NA.EQ.L) GOTO 720
      IF (IT.EQ.30) GOTO 760
      IF (IT.EQ.10 .OR. IT.EQ.20) GOTO 505
```

```
      S=X+Y
      Y=X*Y-R
      GOTO 510
505   Y=ABS(A(NN,NA)) + ABS(A(NA,NN-2))
      S=1.5*Y
      Y=Y*Y
510   IT=IT+1
      PRINT 513, IT,NN
513   FORMAT('ITR. NO.=',I3,'       NN=',I3)
      DO M=NN-2,L,-1
        X=A(M,M)
        R=A(M+1,M)
        Z=A(M+1,M+1)
        P=X*(X-S)+Y+R*A(M,M+1)
        Q=R*(X+Z-S)
        R=R*A(M+2,M+1)
        W=ABS(P)+ABS(Q)+ABS(R)
        P=P/W
        Q=Q/W
        R=R/W
        IF (M.EQ.L) GOTO 560
        HH=ABS(A(M,M-1))*(ABS(Q)+ABS(R))
        IF (HH.LT.MA*ABS(P)*(ABS(A(M-1,M-1))+ABS(X)+ABS(Z))) GOTO 560
      END DO
560   DO I=M+2,NN
        A(I,I-2)=0
      END DO
      DO I=M+3,NN
        A(I,I-3)=0
      END DO
      DO 575 K=M,NA
         IF (K.NE.M) THEN
           P=A(K,K-1)
           Q=A(K+1,K-1)
           R=A(K+2,K-1)
           IF (NA.EQ.K) R=0
           X=ABS(P)+ABS(Q)+ABS(R)
           IF (X.EQ.0) GOTO 575
           P=P/X
           Q=Q/X
           R=R/X
           END IF
         S=SQRT(P*P+Q*Q+R*R)
         IF (P .LT. 0) S=-S
         IF (K .NE. M) A(K,K-1)=-S*X
         IF (L .NE. M) A(K,K-1)=-A(K,K-1)
         P=P+S
         X=P/S
         Y=Q/S
         Z=R/S
         Q=Q/P
         R=R/P
         DO J=K,NN
           P=A(K,J)+Q*A(K+1,J)
           IF (NA.NE.K) THEN
             P=P+R*A(K+2,J)
             A(K+2,J)=A(K+2,J)-P*Z
             END IF
           A(K+1,J)=A(K+1,J)-P*Y
           A(K,J)=A(K,J)-P*X
```

```
          END DO
          J=NN
          IF (K+3 .LT. NN) J=K+3
          DO I=L,J
            P=X*A(I,K)+Y*A(I,K+1)
            IF (NA.NE.K) THEN
              P=P+Z*A(I,K+2)
              A(I,K+2)=A(I,K+2)-P*R
              END IF
            A(I,K+1)=A(I,K+1)-P*Q
            A(I,K)=A(I,K)-P
          END DO
575     CONTINUE
        GOTO 445
C--------- RAIZ SIMPLE
705     RL(NN)=X
        IM(NN)=0
        NN=NN-1
        GOTO 435
C--------- PAR DE RAICES
720     P=(Y-X)/2
        Q=P*P+R
        Y=SQRT(ABS(Q))
        IF (Q .LT. 0) GOTO 755
C----------- PAR REAL
730     IF (P .LT. 0) Y=-Y
        Y=P+Y
        RL(NN-1)=X+Y
        RL(NN)=X-R/Y
740     IM(NN-1)=0
        IM(NN)=0
        NN=NN-2
        GOTO 435

C----------- PAR COMPLEJO
755     RL(NN-1)=X+P
        RL(NN)=X+P
        IM(NN-1)=Y
        IM(NN)=-Y
        NN=NN-2
        GOTO 435
760     PRINT *,' SE HA EXCEDIDO EL LIMITE DE ITERACIONES '
765     PRINT *
        PRINT *,' VALORES PROPIOS
        PRINT *,'-----------------------------------------------'
        PRINT *,'     NO.       PARTE REAL        PARTE IMAGINARIA '
        PRINT *,'-----------------------------------------------'
        DO I=1,N
          PRINT 820,I,RL(I),IM(I)
        END DO
        PRINT *,'-----------------------------------------------'
820     FORMAT(1X,I5,2X,1P2E16.8,' I')
        PRINT *
        PRINT *,' MATRIZ REDUCIDA '
          DO I=1,N
           PRINT 1200,(A(I,J),J=1,N)
          END DO
1200    FORMAT(1X, 1P7E11.4)
        STOP
        END
```

D) Ejemplo de salida

```
CSL/F7-3        ITERACION DE HOUSEHOLDER/QR

ETAPA 1 - - REDUCCION DE UNA MATRIZ A LA FORMA DE HESSENBERG
            O TRIDIAGONAL MEDIANTE EL ESQUEMA DE HOUSEHOLDER

MATRIZ ORIGINAL
 5.3000E+00 2.3000E+00 4.6000E+00 2.7000E+00 1.6000E+00 2.2000E+00
 2.4000E+00 7.8000E+00 5.7000E+00 8.4000E+00 3.4000E+00 4.2000E+00
 3.4000E+00 5.6000E+00 2.4000E+00 1.7000E+00 7.4000E+00 3.9000E+00
 8.3000E+00 7.5000E+00 9.2000E+00 6.1000E+00 5.2000E+00 7.9000E+00
 4.3000E+00 5.9000E+00 7.2000E+00 2.6000E+00 4.9000E+00 8.0000E-01
 9.0000E-01 2.7000E+00 4.9000E+00 4.8000E+00 6.7000E+00 4.8000E+00

MATRIZ DE HESSENBERG O TRIDIAGONAL
 5.3000E+00-5.1043E+00 2.4620E+00 2.5047E+00-1.4576E+00-7.9177E-01
-1.0272E+01 1.9500E+01-1.2706E+01-3.5993E+00 3.9338E+00 4.9378E+00
 0.0000E+00-1.1336E+01 4.3307E+00-4.1246E-02-1.1056E+00-8.6185E-01
 0.0000E+00 0.0000E+00 1.9893E+00-3.8121E+00-5.4741E-01 3.2432E-01
 0.0000E+00 0.0000E+00 0.0000E+00 2.4664E+00 3.0625E+00 3.9123E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00-3.0033E+00 2.9188E+00

ETAPA 2 - - ITERACION QR PARA ENCONTRAR LOS VALORES PROPIOS
    ( CRITERIO DE CONVERGENCIA =   3.5527137E-15)
ITR. NO.=  1        NN=  6
ITR. NO.=  2        NN=  6
ITR. NO.=  3        NN=  6
ITR. NO.=  4        NN=  6
ITR. NO.=  1        NN=  4
ITR. NO.=  2        NN=  4
ITR. NO.=  3        NN=  4
ITR. NO.=  4        NN=  4
ITR. NO.=  1        NN=  3

VALORES PROPIOS
-----------------------------------------------
    NO.   PARTE REAL          PARTE IMAGINARIA
-----------------------------------------------
     1     2.83953304E+01  0.00000000E+00 I
     2    -3.07517910E+00  9.14228916E-01 I
     3    -3.07517910E+00 -9.14228916E-01 I
     4     3.11612034E+00  0.00000000E+00 I
     5     2.96946931E+00  3.51930857E+00 I
     6     2.96946931E+00 -3.51930857E+00 I
-----------------------------------------------

MATRIZ REDUCIDA
 2.8395E+01-1.9324E+00-4.2270E+00-4.5041E+00 4.6254E-01-5.2750E+00
-4.9089E-19-2.7794E+00-4.3060E-01-4.2567E+00 8.3568E-01 2.3792E+00
 5.4210E-20 2.1442E+00-3.3709E+00 2.5847E+00 9.7257E-01-2.3789E-01
 7.0197E-25-2.1792E-13 1.4095E-18 3.1161E+00 4.0594E-01 6.1544E-01
 0.0000E+00-1.9961E-12 1.5384E-13-3.4276E-16 2.5655E+00 3.7212E+00
 0.0000E+00 0.0000E+00 6.8632E-12 1.2490E-16-3.3722E+00 3.3735E+00
```

(*Los pequeños residuos debajo de la diagonal deben considerarse como cero.*)

PROBLEMAS

7.1) Transforme la siguiente función característica en un polinomio de Newton hacia adelante evaluando $f(\lambda)$ para $\lambda = -1$, 0 y 1:

$$f(\lambda) = \det \begin{bmatrix} 2-\lambda & -2 \\ 1 & 3-\lambda \end{bmatrix}$$

Después de esto, reduzca el polinomio de interpolación de Newton a una serie de potencias utilizando los coeficientes de Markov. Compare sus resultados con la serie de potencias que se obtiene desarrollando el determinante de manera directa.

7.2) Transforme la siguiente ecuación característica en una serie de potencias utilizando el polinomio de interpolación de Newton:

$$f(\lambda) = \det (A - \lambda I)$$

donde

$$A = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

(*Sugerencia*: evalúe $f(\lambda)$ para $\lambda = 0$, 1, 2 y 3.)

7.3) Repita el problema 7.2) para las siguientes matrices:

a) $\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$

b) $\begin{bmatrix} 2 & -1 & 0 \\ -1.5 & 2 & -0.5 \\ 0 & -1 & 1 \end{bmatrix}$

7.4) Calcule los valores propios de las siguientes matrices mediante el método de interpolación:

a) $\begin{bmatrix} 5 & 2 & 7 \\ 3 & 1 & 5 \\ 2 & 6 & 2 \end{bmatrix}$

b) $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 4 \end{bmatrix}$

7.5) Transforme la ecuación característica de la matriz dada por

$$\begin{bmatrix} 5.300000 & 2.300000 & 4.600000 & 2.700000 \\ 2.400000 & 7.800000 & 5.700000 & 8.399999 \\ 3.400000 & 5.600000 & 2.400000 & 1.700000 \\ 8.300000 & 7.500000 & 9.200000 & 6.100000 \end{bmatrix}$$

en una serie de potencias mediante el método de interpolación, luego encuentre los valores propios mediante el esquema de Bairstow.

7.6) Calcule los valores propios de la siguiente matriz simétrica mediante el esquema de Householder/tridiagonal:

$$\begin{bmatrix} 3.0000000 & 2.0000000 & 1.0000000 \\ 2.0000000 & 2.0000000 & 1.0000000 \\ 1.0000000 & 1.0000000 & 1.0000000 \end{bmatrix}$$

7.7) Determine los valores propios de la siguiente matriz simétrica mediante el esquema de Householder/tridiagonal:

$$M = \begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 3 & 2 & 1 \\ 2 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

7.8) Transforme las siguientes matrices a la forma de Hessenberg utilizando una transformación de similaridad; haga los cálculos a mano.

(a) $\begin{bmatrix} 3 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 4 \end{bmatrix}$

(b) $\begin{bmatrix} 4 & 1 & 2 \\ 3 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$

7.9) Determine los valores propios de la siguiente matriz utilizando el esquema de Householder/tridiagonal:

$$\begin{bmatrix} 5.3000000 & 2.3000000 & 4.6000000 & 2.7000000 \\ 2.3000000 & 7.8000000 & 5.7000000 & 8.3999990 \\ 4.6000000 & 5.7000000 & 2.4000000 & 1.7000000 \\ 2.7000000 & 8.3999990 & 1.7000000 & 6.1000000 \end{bmatrix}$$

7.10) Calcule los valores propios de la siguiente matriz utilizando el esquema de Householder/QR:

$$\begin{bmatrix} 5.300000 & 2.300000 & 4.600000 & 2.700000 \\ 2.400000 & 7.800000 & 5.700000 & 8.399999 \\ 3.400000 & 5.600000 & 2.400000 & 1.700000 \\ 8.300000 & 7.500000 & 9.200000 & 6.100000 \end{bmatrix}$$

7.11) Encuentre los valores propios de la matriz dada en el problema 7.9) mediante el esquema de Householder/QR.

7.12) Calcule las raíces de los siguientes polinomios mediante el esquema QR:

a) $y = 1.1 + 4x + 3x^2 + x^3$

b) $y = -2.202 - 5.301x - 3x^2 + 1.1x^3 + x^4$

(*Sugerencia*: transforme el problema del polinomio a un problema equivalente de valores propios de una matriz utilizando la matriz de Frobenius y aplique entonces el esquema QR.)

7.13) Al aumentar la multiplicidad de un valor propio, la precisión del esquema QR se vuelve más pobre. Esto debe tenerse en cuenta al aplicar dicho esquema para determinar las raíces de un polinomio. (Esta situación es muy similar a la del esquema de Bairstow; véase el problema 3.30). El siguiente polinomio tiene a $x = 1$ como una raíz de multiplicidad 6:

$$x^6 - 6x^5 + 15x^4 - 20x^3 + 15x^2 - 6x + 1 = 0$$

Intente encontrar todas las raíces mediante el esquema de QR o de Householder/QR.

BIBLIOGRAFIA

Cowell, W. R., editor, *Sources and Development of Mathematical Software*, Prentice-Hall, 1984.

Faddeeva, V. N., *Computational Methods of Linear Algebra*, Dover, 1959.

Garbow, B. S., J. M. Boyle, J. J. Dongarra y C. B. Moler, *Matrix Eigensystem Routines*—EISPACK Guide Extension, *Lecture Notes in Computer Science*, Vol. 51, Springer-Verlag, 1977.

IMSL Library Reference Manual, IMSL, Inc., 7500 Bellaire Boulvard, Houston, Texas 77036.

Jennings, A., *Matrix Computations for Engineers and Scientists*, Wiley, 1977.

Lawson, C. R. H., D. Kincaid y F. Krogh, *Matrix Eigensystem Routines*—EISPACK Guide Extension, *Lecture Notes in Computer Science*, Vol. 51, Springer-Verlag, 1977.

Martin, R. S., P. Peters y J. H. Wilkinson, "*The QR Algorithm for Real Hessenberg Matrices*". Numerische Math., Vol. 14, 1970.

Morris, J.L., *Computational Methods in Elementary Numerical Analysis*, Wiley, 1983.

Shoup, E., *Applied Numerical Methods for the Micro-Computer*, Prentice-Hall, 1984.

Smith, B. T., J. M. Boyle, J. J. Dongarra, B. S. Garbow, Y. Ikebe, V. C. Klema y C. B. Moler, Matrix Eigensystems Routines—EISPACK Guide, *Lecture Notes in Computer Science*, Vol. 6, Springer-Verlag, 1976.

Stewart, G. W., *Introduction to Matrix Computations*, Academic Press, 1974.

Wachpress, E. L., *Iterative Solution of Elliptic Systems*, Prentice-Hall, 1966.

8
Ajuste de curvas

8.1 INTRODUCCION

Los datos que se obtienen mediante mediciones fluctúan, esto se debe a errores aleatorios del sistema de medición aplicado al comportamiento intrínsecamente estocástico del sistema en observación. Cualquiera que sea la razón, es frecuente que surja la necesidad de ajustar una función a los datos de una medición. Por ejemplo, un investigador podría intentar desarrollar una fórmula empírica para el sistema en observación, o bien un economista desearía ajustar una curva a una tendencia económica actual para poder predecir el futuro.

Si el número de datos es igual al orden de un polinomio más uno, podemos ajustar con exactitud dicho polinomio a los datos (ésta es la interpolación polinomial descrita en el capítulo 2). Sin embargo, al hacer el ajuste de una función a los datos de una medición, deben utilizarse un número de datos mucho mayor que el orden del polinomio. De hecho, mientras más datos se utilicen, mejor será la precisión de la curva ajustada.

Por tanto, ¿cómo podemos ajustar una función a los puntos dados? Lo mejor que podemos hacer es considerar una función con pocos parámetros libres y determinarlos de forma que la desviación de la función con respecto a los datos sea mínima. Dicha minimización de la desviación se obtiene mediante el método de mínimos cuadrados.

8.2 REGRESION LINEAL

Supongamos que deseamos encontrar una función lineal que se ajuste a los datos de la tabla 8.1, con una desviación mínima. La función lineal determinada de esta manera se llama una recta de regresión.

Tabla 8.1 Un conjunto de datos observados

i	x	y
1	0.1	0.61
2	0.4	0.92
3	0.5	0.99
4	0.7	1.52
5	0.7	1.47
6	0.9	2.03

La función lineal se expresa aquí como

$$g(x) = a + bx \tag{8.2.1}$$

donde a y b son constantes por determinar. La desviación de la recta con respecto a cada dato se define como

$$r_i = y_i - g(x_i) = y_i - (a + bx_i), \quad i = 1, 2, \ldots, L \tag{8.2.2}$$

donde L es el número total de datos —seis en este ejemplo— y a y b son constantes por determinar.

El cuadrado total de las desviaciones está dado por

$$R = \sum_{i=1}^{L} (r_i)^2 = \sum_{i=1}^{L} (y_i - a - bx_i)^2 \tag{8.2.3}$$

Debido a que a y b son parámetros arbitrarios, se determinan de forma que minimicen a R. El mínimo de R se obtiene si las derivadas parciales de R con respecto a a y b se anulan:

$$\begin{aligned} \frac{\partial R}{\partial a} &= -2 \sum_{i=1}^{L} (y_i - a - bx_i) = 0 \\ \frac{\partial R}{\partial b} &= -2 \sum_{i=1}^{L} x_i(y_i - a - bx_i) = 0 \end{aligned} \tag{8.2.4}$$

que, después de dividir entre -2, se puede reescribir como

$$\begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} \tag{8.2.5}$$

donde

$$A_{1,1} = L$$
$$A_{1,2} = \sum x_i$$
$$Z_1 = \sum y_i$$
$$A_{2,1} = \sum x_i$$
$$A_{2,2} = \sum (x_i)^2$$
$$Z_2 = \sum x_i y_i$$

En las ecuaciones anteriores, la suma es sobre i, desde 1 hasta L. Observe que $A_{2,1}$ es igual a $A_{1,2}$. La solución de la ecuación (8.2.5) se escribe como

$$a = \frac{A_{2,2}Z_1 - A_{1,2}Z_2}{d}$$
$$b = \frac{A_{1,1}Z_2 - A_{2,1}Z_1}{d} \tag{8.2.6}$$

donde

$$d = A_{1,1}A_{2,2} - A_{1,2}A_{2,1}$$

El PROGRAMA 8-1 de este capítulo lleva a cabo la regresión lineal. Puede graficar los puntos dados y la línea de regresión en la pantalla.

Ejemplo 8.1

Calcule la línea de regresión para los datos de la tabla 8.1.

(Solución)

Calculamos los coeficientes de la ecuación (8.2.5), como en la tabla 8.2.

Tabla 8.2

Propósito	A12, A21	Z1	A22	Z2
i	x_i	y_i	x_i^2	$x_i y_i$
1	.1	.61	.01	.061
2	.4	.92	.16	.368
3	.5	.99	.25	.495
4	.7	1.52	.49	1.064
5	.7	1.47	.49	1.029
6	.9	2.03	.81	1.827
Total	3.3	7.54	2.21	4.844

De los resultados de esa tabla obtenemos

$$A11 = L = 6, \quad A12 = 3.3, \quad Z1 = 7.54$$

$$A21 = 3.3, \quad A22 = 2.21, \quad Z2 = 4.844$$

Así, la ecuación (8.2.5) queda

$$\begin{bmatrix} 6 & 3.3 \\ 3.3 & 2.21 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 7.54 \\ 4.844 \end{bmatrix}$$

La solución es

$$a = 0.2862, \quad b = 1.7645$$

Así, la recta de regresión es

$$g(x) = 0.2862 + 1.7645x$$

que se grafica en la figura E8.1 con los puntos dados.

Ahora evaluamos la desviación de la línea ajustada en la tabla 8.3.

Tabla 8.3

i	$x(i)$	$y(i)$	$g = ax + b$	Desviación
1	0.1	0.61	0.4626161	0.14738
2	0.4	0.92	0.9919831	−0.07198
3	0.5	0.99	1.168439	−0.17844
4	0.7	1.52	1.52135	−0.00135
5	0.7	1.47	1.52135	−0.05135
6	0.9	2.03	1.874261	0.15574

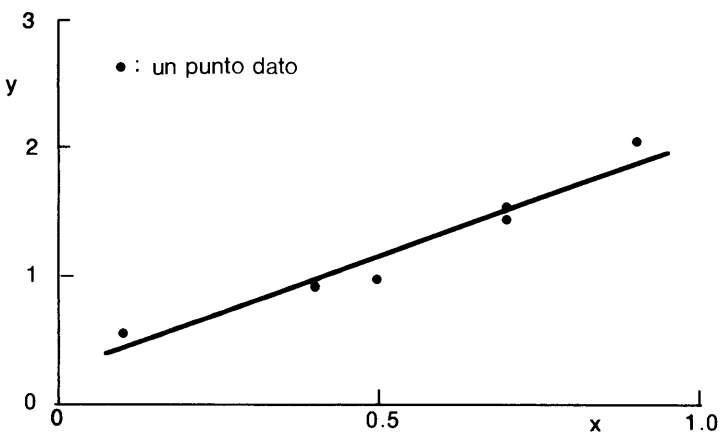


Figura E8.1 Recta ajustada a los datos

Resumen de Esta Sección

a) La finalidad de la regresión lineal es la de ajustar una función lineal a puntos dados mediante el método de mínimos cuadrados.
b) El PROGRAMA 8-1 se puede utilizar para una regresión lineal.

8.3 AJUSTE DE CURVAS CON UN POLINOMIO DE ORDEN SUPERIOR

La regresión lineal explicada en la sección anterior funciona bien si los datos de la medición son lineales intrínsecamente, o si el rango de las abscisas es pequeño. Sin embargo, para otros casos, se pueden obtener mejores resultados ajustando un polinomio de orden superior al conjunto de datos.

El principio de los mínimos cuadrados se puede extender para ajustar un polinomio de cualquier orden a los datos de una medición. Primero se escribe un polinomio de orden N como

$$g(x) = a_0 + a_1 x + a_2 x^2 + \cdots a_N x^N \tag{8.3.1}$$

La desviación de la curva de los puntos dados es

$$r_i = y_i - g(x_i), \quad i = 1, 2, \ldots, L \tag{8.3.2}$$

donde L es el número de puntos dados. El total de los cuadrados de la desviación es el siguiente:

$$R = \sum_{i=1}^{L} (r_i)^2 \tag{8.3.3}$$

Hacemos iguales a cero las derivadas parciales de R con respecto a los coeficientes del polinomio para minimizar a R:

$$\frac{\partial R}{\partial a_n} = 0, \quad n = 0, 1, 2, \ldots, N \tag{8.3.4}$$

o, en forma equivalente,

$$\sum_{n=0}^{N} \left[\sum_{i=1}^{L} x_i^{n+k} \right] a_n = \sum_{i=1}^{L} x_i^k y_i \text{ para } k = 0, 1, 2, \ldots, N \tag{8.3.5}$$

que se puede escribir en forma más explícita como:

$$\begin{bmatrix} L & \sum x_i & \sum x_i^2 & \cdots & \sum x_i^N \\ \sum x_i & \sum x_i^2 & \sum x_i^3 & \cdots & \sum x_i^{N+1} \\ & & & \cdots & \\ \sum x_i^N & \sum x_i^{N+1} & \sum x_i^{N+2} & \cdots & \sum x_i^{2N} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \\ a_N \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \\ \\ \sum x_i^N y_i \end{bmatrix}$$

Los coeficientes a_n, $n = 0, 1, 2, \ldots, N$ se determinan resolviendo la ecuación (8.3.5) en forma simultánea (mediante eliminación de Gauss).

Ejemplo 8.2

Ajuste un polinomio cuadrático a los datos de la tabla 8.1.

(Solución)

La ecuación para los coeficientes, en notación matricial, es

$$\begin{bmatrix} 6.0000 & 3.3000 & 2.2100 \\ 3.3000 & 2.2100 & 1.6050 \\ 2.2100 & 1.6050 & 1.2245 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 7.5400 \\ 4.8440 \\ 3.5102 \end{bmatrix}$$

Los coeficientes de las potencias son:

Potencia n	Coeficiente a_n
0	0.587114
1	0.059102
2	1.729537

Por lo que el polinomio cuadrático determinado es

$$y = 0.587114 + 0.059102x + 1.729537x^2$$

En la tabla 8.4 se muestra la evaluación del error.

Tabla 8.4

i	$x(i)$	$y(i)$	Polinomio	Desviación
1	0.1	0.61	0.6103198	−0.00032
2	0.4	0.92	0.8874811	0.03252
3	0.5	0.99	1.049050	−0.05905
4	0.7	1.52	1.475959	0.04404
5	0.7	1.47	1.475959	−0.00596
6	0.9	2.03	2.041231	−0.01123

Las ecuaciones lineales que surgen en el ajuste de curvas son con frecuencia mal condicionadas cuando los coeficientes de las ecuaciones lineales se convierten en una mezcla de números muy grandes con cifras muy pequeñas. Esta diseminación se intensifica cuando aumentan tanto el rango de valores de x en los puntos dados y el orden del polinomio. Por lo tanto, es recomendable utilizar la doble precisión para resolver las ecuaciones lineales (véase el PROGRAMA 8-1).

Resumen de Esta Sección

a) El ajuste de polinomios es una extensión del ajuste de rectas y se basa en el método de mínimos cuadrados.

b) El número de puntos ajustados es generalmente mucho más grande que el orden del polinomio.

c) Con frecuencia, la ecuación lineal asociada con el ajuste de polinomios es demasiado vulnerable a los errores de redondeo. Se recomienda el uso de la doble precisión.

8.4 AJUSTE DE CURVAS MEDIANTE UNA COMBINACION LINEAL DE FUNCIONES CONOCIDAS

Al ajustar una función a puntos dados, se puede utilizar una combinación lineal de cualesquiera funciones conocidas, en vez de emplear polinomios.

La curva ajustada a los puntos dados se puede escribir en este caso como

$$\begin{aligned} g(x) &= a_1 f_1(x) + a_2 f_2(x) + a_3 f_3(x) + \cdots + a_N f_N(x) \\ &= \sum_{n=1}^{N} a_n f_n(x) \end{aligned} \tag{8.4.1}$$

donde $f_1, f_2, \ldots$ son funciones prescritas, $a_1, a_2, \ldots$ son coeficientes indeterminados y N es el número total de funciones prescritas.

La desviación de la curva con respecto de cada punto dado se define como

$$r_i = y_i - \sum_{n=1}^{N} a_n f_n(x_i), \; i = 1, 2, \ldots, L \tag{8.4.2}$$

donde L es el número total de puntos dados. El total de los cuadrados de las desviaciones es

$$\begin{aligned} R &= \sum_{i=1}^{L} [r_i]^2 \\ &= \sum_{i=1}^{L} \left[y_i - \sum_{n=1}^{N} a_n f_n(x_i) \right]^2 \end{aligned} \tag{8.4.3}$$

Al igualar a cero las derivadas parciales de R con respecto a los coeficientes indeterminados, obtenemos

$$\frac{\partial R}{\partial a_n} = 0, \quad n = 1, 2, \ldots, N \tag{8.4.4}$$

o, de manera equivalente,

$$\sum_{m=1}^{N} \left[\sum_{i=1}^{L} f_m(x_i) f_n(x_i) \right] a_m = \sum_{i=1}^{L} y_i f_n(x_i) \quad \text{para } n = 1 \text{ a } N \tag{8.4.5}$$

donde se dividió la ecuación entre 2. La ecuación (8.4.5) tiene N ecuaciones con N incógnitas. Así, las ecuaciones se pueden resolver mediante la eliminación de Gauss.

Ejemplo 8.3

Determine los coeficientes de la función

$$g(x) = a_1 + a_2x + a_3\,\mathrm{sen}(x) + a_4\exp(x)$$

ajustada a los datos de la tabla siguiente:

x	y
0.1	0.61
0.4	0.92
0.5	0.99
0.7	1.52
0.7	1.47
0.9	2.03

(Solución)

En forma matricial, la ecuación (8.4.5) es

$$\begin{bmatrix} 6.0000E+00 & 3.3000E+00 & 3.0404E+00 & 1.0733E+01 \\ 3.3000E+00 & 2.2100E+00 & 2.0124E+00 & 6.5645E+00 \\ 3.0404E+00 & 2.0124E+00 & 1.8351E+00 & 6.0030E+00 \\ 1.0733E+01 & 6.5645E+00 & 6.0030E+00 & 2.0325E+01 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 7.5400E+00 \\ 4.8440E+00 \\ 4.4102E+00 \\ 1.4693E+01 \end{bmatrix}$$

Los coeficientes determinados son los siguientes:

Función n	Coeficiente a_n
1	−5.265785
2	−19.706939
3	13.496765
4	5.882096

La evaluación del error se muestra en la tabla 8.5.

Tabla 8.5

i	$x(i)$	$y(i)$	Curva ajustada	Desviación
1	0.1	0.61	0.6117	−0.0017
2	0.4	0.92	0.8824	0.0376
3	0.5	0.99	1.0494	−0.0594
4	0.7	1.52	1.4793	0.0407
5	0.7	1.47	1.4793	−0.0093
6	0.9	2.03	2.0380	−0.0080

Resumen de Esta Sección

a) El ajuste de curvas, incluyendo el ajuste de rectas y polinomios, se basa en el método de mínimos cuadrados.

b) Las funciones que aparecen en una combinación lineal se pueden elegir, ya sea por experiencia o por razones teóricas.

PROGRAMA

PROGRAMA 8-1 Curvas por mínimos cuadrados

A) Explicaciones

Este programa incluye la regresión lineal (sección 8.2), el ajuste de polinomios (sección 8.3) y las combinaciones lineales de funciones (sección 8.4) en un programa. El programa analiza las desviaciones de la curva ajustada y la gráfica junto con los puntos dados. El polinomio puede ser de cualquier orden menor que 11. Si el orden es 1, el programa lleva a cabo el análisis de regresión lineal.

Al ejecutar el programa, la computadora pregunta el tipo de ajuste de curvas deseado, ya sea el ajuste de polinomios o una combinación lineal de funciones. Se presiona el 0 para elegir la primera opción y 1 para la segunda. Sin embargo, si se elige la combinación lineal, las funciones a utilizar deben estar definidas en la subrutina *FUN*. El ajuste de una recta es un caso especial de ajuste de polinomios.

Para aumentar la precisión de los cálculos, se utiliza la doble precisión para las variables clave. El conjunto de datos que se ajustará mediante un polinomio se define en los enunciados DATA. Las ecuaciones lineales se resuelven mediante la subrutina *GAUSS*. Después de determinar la curva, se calculan sus desviaciones y se imprimen.

B) Variables

LP:	1 para ajuste polinomial y 2 para combinaciones lineales
NORD:	orden del polinomio a ajustar (en el caso de que LP = 1)
IN:	número de puntos dados
A(I, J):	elementos de la matriz de coeficientes
A(I, M + 1):	$(i - 1)$-ésimo coeficiente del polinomio, a_{i-1}
GG(I):	valor del polinomio ajustado en el punto dado t_i
X(I), Y(I):	datos

C) Listado

```
C-----CSL/F8-1.FOR     AJUSTE DE CURVAS CON POLINOMIOS O
C                      COMBINACIONES LINEALES MEDIANTE MINIMOS CUADRADOS
10     DIMENSION X(0:100), Y(0:100),A(0:10,0:10),GG(0:100)
```

```
      DOUBLEPRECISION A
      PRINT *
      PRINT *,'CSL/F8-1       AJUSTE DE CURVAS MEDIANTE MINIMOS CUADRADOS '
      PRINT *
      DATA IN/6/                    ! Número de puntos dados.
      DATA (X(J),J=1,6)/0.1,  0.4,  0.5,  0.7,  0.7,  0.9/
      DATA (Y(J),J=1,6)/0.61, 0.92, 0.99, 1.52, 1.47, 2.03/
      PRINT *,'  OPRIMA 0 PARA AJUSTE POLINOMIAL   '
      PRINT *,'          1 PARA COMBINACION LINEAL   '
      READ *, LP
      IF (LP.EQ.1) GOTO 200
C----------------------                                   ! Ajuste polinomial
      PRINT *,'  DE EL ORDEN DEL POLINOMIO  '
      READ *, NORD
      M=NORD+1
      DO K=1,M                  ! Inicialización de la matriz
         DO J=1,M+1
            A(K,J)=0.0
         END DO
      END DO
      DO K=1,M                  ! Desarrollo de la matriz para el ajuste polinomial
         DO I=1,IN              ! IN es el número de puntos dados
            DO J=1,M
               JJ=K-1+J-1
               YY=1.0
               IF(JJ.NE.0) YY=X(I)**JJ
               A(K,J)=A(K,J)+YY
            END DO
            JEX=K-1
            YY=1.0
            IF(JEX.NE.0) YY=X(I)**JEX
            A(K,M+1)=A(K,M+1)+Y(I)*YY
         END DO
      END DO
      GOTO 220
C----------------------                                   ! Combinación lineal
200   PRINT *,'  DE EL NUMERO DE FUNCIONES EN LA COMBINACION LINEAL   '
      PRINT *,'     (POR EL MOMENTO SOLO SE DISPONE DE CUATRO FUNCIONES)'
      READ *,M          ! M es el número de funciones en la combinación lineal
      DO I=1,IN
         DO K=1,M
            FK=FUN(K,X(I))
            DO J=1,M
               A(K,J)=A(K,J)+FK*FUN(J,X(I))
            END DO
            A(K,M+1)=A(K,M+1)+Y(I)*FK
         END DO
       END DO
C-------------------------                                ! Solución
220    DO I=1,M
          PRINT 50, (A(I,J),J=1,M+1)
       END DO
50     FORMAT(1X,1P7E11.3)
       N=M
          CALL  GAUSS(M,A)
      PRINT *
      PRINT *,'  DETERMINACION DE COEFICIENTES   '
      PRINT *,'----------------------------------------'
      IF(LP.EQ.1) THEN
          PRINT *,'    FUNCION              COEFICIENTE
```

```
      ELSE
           PRINT *,'    POTENCIA            COEFICIENTE '
      END IF
      PRINT *,'----------------------------------------'
      DO I=1,M
         IF(LP.EQ.1) WRITE (6,595)  I,  A(I,M+1)
         IF(LP.NE.1) WRITE (6,595)  I-1,A(I,M+1)
      END DO
595   FORMAT(2X,I4,8X,F12.6)
      PRINT *,'----------------------------------------'
      PRINT *
      PRINT *
      DO I=1,IN
         GG(I)=0.0
         DO K=1,M
            IF(LP.EQ.1) GG(I)=GG(I)+A(K,M+1)*FUN(K,X(I))
            IF(LP.NE.1) GG(I)=GG(I)+A(K,M+1)*X(I)**(K-1)
         END DO
      END DO
      PRINT *,' EVALUACION DEL ERROR '
      PRINT *,'-------------------------------------------------------'
      PRINT *,'   I       X(I)       Y(I)       POLINOMIO       DESVIACION '
      PRINT *,'-------------------------------------------------------'
      DO I=1,IN
         WRITE (6,435)I,X(I),Y(I),GG(I),Y(I)-GG(I)
      END DO
435   FORMAT(I4,3X,F8.2,2X,F10.5,3X,F12.7,2X,F11.6)
      PRINT *,'-------------------------------------------------------'
      PRINT *
      STOP
      END
C************************** ELIMINACION DE GAUSS
      SUBROUTINE GAUSS(N,A)
      DOUBLEPRECISION A,TM,VA,R
      DIMENSION A(0:10,0:10)
      DO I=1, N-1
         IPV=I
         DO J=I+1,N
            IF (ABS(A(IPV,I))  .LT.  ABS(A(J,I))) IPV=J
         END DO
         IF (IPV.NE.I) THEN
            DO JC=1,N+1
               TM=A(I,JC)
               A(I,JC)=A(IPV,JC)
               A(IPV,JC)=TM
            END DO
         END IF
         DO JR=I+1,N
           IF (A(JR,I).NE.0) THEN
             IF (A(I,I).EQ.0.0) GOTO 840
             R=A(JR,I)/A(I,I)
             DO KC=I+1,N+1
                A(JR,KC)=A(JR,KC)-R*A(I,KC)
             END DO
           END IF
         END DO
      END DO
C-- SUSTITUCION HACIA ATRAS
      IF (A(N,N).EQ.0) GOTO 840
      A(N,N+1)=A(N,N+1)/A(N,N)
```

```
      DO NV=N-1,1,-1
         VA=A(NV,N+1)
         DO K=NV+1,N
            VA=VA-A(NV,K)*A(K,N+1)
         END DO
         A(NV,N+1)=VA/A(NV,NV)
      END DO
      RETURN
840   PRINT *,'LA MATRIZ ES SINGULAR'
      RETURN
      END
C*********************  SUBPROGRAMA QUE DEFINE LAS FUNCIONES
      FUNCTION FUN(K,X)
      IF (K.EQ.1) FUN=1                    !  Primera función
      IF (K.EQ.2) FUN=X                    !  Segunda función
      IF (K.EQ.3) FUN=SIN(X)        !  etc.
      IF (K.EQ.4) FUN=EXP(X)
      RETURN
      END
```

D) Ejemplo de salida

(Corrida 1)

```
CSL/F8-1          AJUSTE DE CURVAS MEDIANTE MINIMOS CUADRADOS

 OPRIMA 0 PARA AJUSTE POLINOMIAL
        1 PARA COMBINACION LINEAL
0
 DE EL ORDEN DEL POLINOMIO
2
  6.000E+00  3.300E+00  2.210E+00  7.540E+00
  3.300E+00  2.210E+00  1.605E+00  4.844E+00
  2.210E+00  1.605E+00  1.224E+00  3.510E+00

  DETERMINACION DE COEFICIENTES
-----------------------------------------
  POTENCIA            COEFICIENTE
-----------------------------------------
     0              0.587116
     1              0.059094
     2              1.729546
-----------------------------------------
EVALUACION DEL ERROR
-------------------------------------------------------
   I       X(I)       Y(I)        POLINOMIO       DESVIACION
-------------------------------------------------------
  1        0.10      0.61000       0.6103207      -0.000321
  2        0.40      0.92000       0.8874806       0.032519
  3        0.50      0.99000       1.0490491      -0.059049
  4        0.70      1.52000       1.4759588       0.044041
  5        0.70      1.47000       1.4759588      -0.005959
  6        0.90      2.03000       2.0412321      -0.011232
-------------------------------------------------------
```

(Corrida 2)

```
CSL/F8-1          AJUSTE DE CURVAS MEDIANTE MINIMOS CUADRADOS

 OPRIMA 0 PARA AJUSTE POLINOMIAL
        1 PARA COMBINACION LINEAL
```

```
1
 DE EL NUMERO DE FUNCIONES EN LA COMBINACION LINEAL
   (POR EL MOMENTO SOLO SE DISPONE DE CUATRO FUNCIONES)
3
  6.000E+00   3.300E+00   3.040E+00   7.540E+00
  3.300E+00   2.210E+00   2.012E+00   4.844E+00
  3.040E+00   2.012E+00   1.835E+00   4.410E+00

 DETERMINACION DE COEFICIENTES
----------------------------------------
   POTENCIA          COEFICIENTE
----------------------------------------
    1              0.522520
    2              7.903179
    3             -7.129104
----------------------------------------
EVALUACION DEL ERROR
-------------------------------------------------------
   I       X(I)       Y(I)       POLINOMIO       DESVIACION
-------------------------------------------------------
  1       0.10      0.61000      0.6011153       0.008885
  2       0.40      0.92000      0.9075877       0.012412
  3       0.50      0.99000      1.0562348      -0.066235
  4       0.70      1.52000      1.4620502       0.057950
  5       0.70      1.47000      1.4620502       0.007950
  6       0.90      2.03000      2.0509617      -0.020962
-------------------------------------------------------
```

PROBLEMAS

8.1) Determine una función lineal ajustada a los siguientes puntos mediante el método de mínimos cuadrados. (Primero trabaje con una calculadora y luego verifique la respuesta con el PROGRAMA 8-1.)

i	x_i	y_i
1	1.0	2.0
2	1.5	3.2
3	2.0	4.1
4	2.5	4.9
5	3.0	5.9

8.2) Obtenga una función lineal ajustada a los siguientes puntos mediante el método de mínimos cuadrados. (Responda la pregunta utilizando el PROGRAMA 8-1).

i	x_i	y_i
1	0.1	9.9
2	0.2	9.2
3	0.3	8.4
4	0.4	6.6
5	0.5	5.9
6	0.6	5.0
7	0.7	4.1
8	0.8	3.1
9	0.9	1.9
10	1.0	1.1

8.3) Ajuste un polinomio cuadrático al siguiente conjunto de datos.

x	y
0	1
1	0
2	0
3	2

8.4 a) Ajuste un polinomio cuadrático al siguiente conjunto de datos:

i	x_i	y_i
1	0	0
2	1	2.3
3	2	4.2
4	3	5.7
5	4	6.5
6	5	6.9
7	6	6.8

b) Evalúe las desviaciones del polinomio con respecto a los datos.

8.5) Repita el problema anterior con polinomios de primer y tercer orden.

8.6) Ajuste polinomios de orden 1, 2 y 3 a los siguientes datos y compare las desviaciones de los tres polinomios:

$x(i)$	$y(i)$
0	0
0.002	0.618
0.004	1.1756
0.006	1.618
0.008	1.9021

8.7) Ajuste una función cuadrática a los siguientes datos y grafique la curva ajustada junto con los puntos dados:

i	x_i	y_i
1	0	0
2	0.2	7.78
3	0.4	10.68
4	0.6	8.37
5	0.8	3.97
6	1	0

8.8) Ajuste un polinomio cúbico a los datos del problema anterior.

8.9) Ajuste

$$g(x) = a_0 + a_1 x + a_2 \operatorname{sen}(\pi x) + a_3 \operatorname{sen}(2\pi x)$$

a la tabla siguiente:

i	$x(i)$	$y(i)$
1	0.1	0
2	0.2	2.1220
3	0.3	3.0244
4	0.4	3.2568
5	0.5	3.1399
6	0.6	2.8579
7	0.7	2.5140
8	0.8	2.1639
9	0.9	1.8358

BIBLIOGRAFIA

Daniel, A., y F. S. Wood, *Fitting Equations to Data*, Wiley-Interscience, 1971.

Dongarra, J. J., J. R. Bunch, C. B. Moler y G. W. Stewart, *LINPACK User's Guide*, SIAM, 1979.

Forsythe, G. E., M. A. Malcolm y C. B. Moler, *Computer Methods for Mathematical Computations*, Prentice-Hall, 1977.

Jennings, A., *Matrix Computations for Engineers and Scientists*, Wiley, 1977.

Robinson, E. A., *Least Squares Regression Analysis in Terms of Linear Algebra*, Goose Pond Press, 1981.

9

Problemas de ecuaciones diferenciales ordinarias con valor o condición inicial

9.1 INTRODUCCION

Los problemas de ecuaciones diferenciales ordinarias (EDO) se clasifican en problemas con condiciones iniciales y problemas con condiciones en la frontera. Muchos de los problemas con condiciones iniciales dependen del tiempo; en ellos, las condiciones para la solución están dadas en el tiempo inicial. Los métodos numéricos para los problemas con condiciones iniciales difieren en forma significativa de los que se utilizan para los problemas con condiciones en la frontera. Por lo tanto, este capítulo examina los métodos de solución numérica sólo para el primer tipo, mientras que el capítulo 10 describe los métodos numéricos para el segundo tipo.

El problema con condiciones iniciales de una EDO de primer orden se puede escribir en la forma

$$y'(t) = f(y, t), y(0) = y_0 \tag{9.1.1}$$

donde $f(y, t)$ es una función de y en tanto que t y la segunda ecuación es una condición inicial. En la ecuación (9.1.1), la primera derivada de y está dada como una función conocida de y y t y queremos calcular la función incógnita y integrando numéricamente $f(y, t)$. Si f fuera independiente de y, el cálculo sería simplemente una de las integraciones directas analizadas en el capítulo 4. Sin embargo, el hecho de que f sea una función de la función desconocida y hace que la integración sea distinta.

La condición inicial siempre es parte de la definición del problema, debido a que la solución de un problema con condiciones iniciales sólo se puede determinar de manera única si dicha condición inicial está dada.

A continuación se muestran más ejemplos de problemas con condiciones iniciales de EDO de primer orden:

a) $y'(t) = 3y + 5, \quad y(0) = 1$

b) $y'(t) = ty + 1, \quad y(0) = 0$

c) $y'(t) = -\dfrac{1}{1 + y^2}, \quad y(0) = 1$

e) $y' = z, z' = -y, \quad y(0) = 1, z(0) = 0$

Véase [Rieder y Busby] para ejemplos en ingeniería.

Los métodos numéricos para las EDO calculan la solución en los puntos $t_n = t_{n-1} + h$, donde h es el tamaño del paso (o intervalo de tiempo).

En este capítulo se analizan tres tipos de métodos de integración numérica para problemas con condiciones iniciales: el de Euler, de Runge-Kutta y el predictor-corrector. Los aspectos principales de ellos se resumen en la tabla 9.1.

Tabla 9.1 Resumen de los métodos para los problemas de EDO con condición inicial

Nombre de los métodos	Fórmula relevante	Error Local	Error Global	Otras características[a]
Ecuaciones no rígidas:				
Métodos de Euler				
hacia adelante	Diferencias hacia adelante	$0(h^2)$	$0(h)$	AI, CF
modificado	Regla del trapecio	$0(h^3)$	$0(h^2)$	AI, CF, NL
hacia atrás	Diferencias hacia atrás	$0(h^2)$	$0(h)$	AI, CF, NL
Runge-Kutta				
de segundo orden	Regla del trapecio	$0(h^3)$	$0(h^2)$	AI, CF
de tercer orden	Regla de 1/3 de Simpson	$0(h^4)$	$0(h^3)$	AI, CF
de cuarto orden	Regla de 1/3 o 3/8 de Simpson	$0(h^5)$	$0(h^4)$	AI, CF
Predictor-corrector				
de segundo orden	(idéntico al de Runge-Kutta de segundo orden)			AI, CF
de tercer orden	Newton hacia atrás	$0(h^4)$	$0(h^3)$	NA, CD
de cuarto orden	Newton hacia atrás	$0(h^5)$	$0(h^4)$	NA, CD
Ecuaciones rígidas:				
Métodos implícitos	Diferencias hacia atrás; método de Gear			AI/NA
Transformación exponencial	Transformación exponencial			AI

[a]NA: Sin capacidad de autoinicialización.
AI: Capacidad de autoinicialización.
CF: El tamaño del intervalo se puede cambiar con facilidad a mitad de la solución.
CD: El tamaño del intervalo se cambia con dificultad.
NL: En cada paso, podría requerirse la solución de ecuaciones no lineales.

Una vez que aprendamos los métodos numéricos para resolver ecuaciones diferenciales de primer orden, éstos se pueden aplicar con facilidad a las EDO de orden superior, ya que se pueden descomponer en un conjunto de ecuaciones diferenciales de primer orden. Por ejemplo, consideremos

$$y'''' + ay''' + by'' + cy' + ey = g \tag{9.1.2}$$

donde a, b, c, e y g son constantes o funciones conocidas de t. Las condiciones iniciales están dadas por

$$y(0) = y_0, \quad y'(0) = y'_0,$$
$$y''(0) = y''_0, \quad y'''(0) = y'''_0$$

y donde y_0, y'_0, y''_0 y y'''_0 son valores preescritos. Si definimos u, v y w como

$$u = y', \qquad v = y'', \qquad w = y'''$$

la ecuación (9.1.2) se puede escribir como

$$w' + aw + bv + cu + ey = g \tag{9.1.3}$$

Así, la ecuación (9.1.2) es equivalente al siguiente conjunto de cuatro EDO de primer orden:

$$\begin{aligned} y' &= u, & y(0) &= y_0 \\ u' &= v, & u(0) &= y'_0 \\ v' &= w, & v(0) &= y''_0 \\ w' &= g - aw - bv - cu - ey, & w(0) &= y'''_0 \end{aligned}$$

Podemos entonces aplicar los métodos numéricos de solución de EDO de primer orden al conjunto anterior.

También se pueden aplicar métodos numéricos a las ecuaciones integro-diferenciales. Por ejemplo, consideremos la ecuación dada por

$$y'' + ay + \int_0^t y(s)\,ds = g, \quad y(0) = y_0, \quad y'(0) = y'_0 \tag{9.1.4}$$

Definimos u y v como

$$u = y', \qquad v = \int_0^t y(s)\,ds$$

La ecuación (9.14) queda

$$\begin{aligned} u' &= -ay - v + g, & u(0) &= y'_0 \\ v' &= y, & v(0) &= 0 \\ y' &= u, & y(0) &= y_0 \end{aligned} \tag{9.1.5}$$

El anterior conjunto de EDO de primer orden se puede resolver mediante un método numérico.

9.2 METODOS DE EULER

Comenzamos nuestro estudio con los métodos de Euler, los cuales son adecuados para una programación rápida debido a su sencillez. Debemos señalar que, cuando el sistema de ecuaciones es cada vez más complicado, se utilizan con más frecuencia los métodos de Euler. De hecho, una gran parte de los métodos numéricos para las ecuaciones diferenciales parciales parabólicas e hiperbólicas —que son mucho más complicadas que las ecuaciones diferenciales ordinarias— se basan en los métodos de Euler y no en los métodos de Runge-Kutta o predictor-corrector.

Los métodos de Euler tienen tres versiones: a) Euler hacia adelante, b) Euler modificado y c) Euler hacia atrás.

9.2.1 Método de Euler hacia adelante

El método de Euler hacia adelante para la ecuación $y' = f(y, t)$ se obtiene reescribiendo la aproximación por diferencias hacia adelante,

$$\frac{y_{n+1} - y_n}{h} \simeq y'_n \tag{9.2.2}$$

como

$$y_{n+1} = y_n + hf(y_n, t_n) \tag{9.2.3}$$

donde se usa $y'_n = f(y_n, t_n)$. Mediante la ecuación (9.2.3), se calcula y_n en forma recursiva como

$$\begin{aligned} y_1 &= y_0 + hy'_0 = y_0 + hf(y_0, t_0) \\ y_2 &= y_1 + hf(y_1, t_1) \\ y_3 &= y_2 + hf(y_2, t_2) \\ &\vdots \\ y_n &= y_{n-1} + hf(y_{n-1}, t_{n-1}) \end{aligned}$$

Ejemplo 9.1

a) Resuelva $y' = -20y + 7\exp(-0.5t)$, $y(0) = 5$, por medio del método de Euler hacia adelante con $h = 0.01$ para $0 < t \leqslant 0.02$. Haga el cálculo a mano.

b) Repita lo mismo para $h = 0.01$, 0.001 y 0.0001 en una computadora, para $0 \leqslant t \leqslant 0.09$. Evalúe lcs errores de los tres cálculos mediante la comparación con la solución analítica dada por

$$y = 5e^{-20t} + (7/19.5)(e^{-0.5t} - e^{-20t})$$

(Solución)

a) Los primeros cálculos con $h = 0.01$ son los siguientes:

$$t_0 = 0, \qquad y_0 = y(0) = 5$$

$$t_1 = 0.01, \quad y_1 = y_0 + hy_0' = 5 + (0.01)(-20(5) + 7 \exp(0))$$
$$= 4.07$$

$$t_2 = 0.02, \quad y_2 = y_1 + hy_1' = 4.07 + (0.01)(-20(4.07) + 7 \exp(-0.005))$$
$$= 3.32565$$

$$\vdots$$

$$t_n = nh, \qquad y_n = y_{n-1} + hy_{n-1}'$$

b) Los resultados computacionales para los valores seleccionados de t, con tres valores de intervalos de tiempo (espaciamiento de la retícula) se muestran en la tabla 9.2.

Tabla 9.2 Método de Euler hacia adelante

t	$h = 0.01$	$h = 0.001$	$h = 0.0001$
0.01	4.07000 (8.693)	4.14924 (0.769)	4.15617 (0.076)[a]
0.02	3.32565 (14.072)	3.45379 (1.259)	3.46513 (0.124)
0.03	2.72982 (17.085)	2.88524 (1.544)	2.89915 (0.153)
0.04	2.25282 (18.440)	2.42037 (1.684)	2.43554 (0.167)
0.05	1.87087 (18.658)	2.04023 (1.722)	2.05574 (0.171)
0.06	1.56497 (18.125)	1.72932 (1.690)	1.74454 (0.168)
0.07	1.31990 (17.119)	1.47496 (1.613)	1.48949 (0.160)
0.08	1.12352 (15.839)	1.26683 (1.507)	1.28041 (0.150)
0.09	0.96607 (14.427)	1.09646 (1.387)	1.10895 (0.138)
0.10	0.83977 (12.979)	0.95696 (1.261)	0.96831 (0.126)

[a] (error) × 100

Comentarios: la exactitud del método de Euler hacia adelante aumenta al disminuir el intervalo de tiempo h. En efecto, las magnitudes de los errores son aproximadamente proporcionales a h. Sin embargo, una reducción mayor de h, sin el uso de la doble precisión, tiene sus desventajas, puesto que aumenta el error numérico debido al redondeo (véase el capítulo 1).

Aunque el método de Euler hacia adelante es muy sencillo, debe utilizarse cuidadosamente para evitar dos tipos de errores. El primer tipo lo forman los errores de truncamiento, como en el caso del ejemplo 9.1. El segundo tipo lo constituye la posible inestabilidad, que aparece cuando la constante del tiempo es negativa (la solución tiende a cero si no hay término fuente), a menos de que el intervalo de tiempo h sea suficientemente pequeño. Una ecuación característica con solución decreciente es $y' = -\alpha y$, $y(0) = y_0 > 0$, donde $\alpha > 0$. La solución exacta es $y = y_0 e(-\alpha t)$. El método de Euler hacia adelante para este problema es

$$y_{n+1} = (1 - \alpha h)y_n$$

Si $\alpha h < 1$, la solución numérica es decreciente y positiva; pero si $\alpha h > 1$, el signo de la solución es alternante. Además, si $\alpha h > 2$, la magnitud de la solución aumenta en cada paso y la solución oscila. Esto se conoce como *inestabilidad*.

El método de Euler hacia adelante es aplicable a un conjunto de EDO de primer orden. Consideremos un conjunto de EDO de primer orden dado por:

$$\begin{aligned} y' &= f(y, z, t), \quad y(0) = y_0 \\ z' &= g(y, z, t), \quad z(0) = z_0 \end{aligned} \tag{9.2.4}$$

El método de Euler hacia adelante para la ecuación (9.2.4) se escribe como

$$\begin{aligned} y_{n+1} &= y_n + hy'_n = y_n + hf(y_n, z_n, t_n) \\ z_{n+1} &= z_n + hz'_n = z_n + hg(y_n, z_n, t_n) \end{aligned} \tag{9.2.5}$$

Como se mencionó antes, una ecuación diferencial ordinaria de orden superior se puede descomponer en un sistema simultáneo de ecuaciones diferenciales de primer orden.

Ejemplo 9.2

Por medio del método de Euler hacia adelante con $h = 0.5$, determine los valores de $y(1)$ y $y'(1)$ para

$$y''(t) - 0.05y'(t) + 0.15y(t) = 0, \quad y'(0) = 0, \quad y(0) = 1$$

(Solución)

Sea $y' = z$: entonces la EDO de segundo orden es

$$\begin{aligned} y' &= z, & y(0) &= 1 \\ z' &= 0.05z - 0.15y, & z(0) &= 0 \end{aligned}$$

Denotaremos $y_n = y(nh)$ y $z_n = z(nh)$. Las condiciones iniciales son $y_0 = y(0) = 1$ y z_0 y $y'(0) = 0$. Por medio del método de Euler hacia adelante, se obtienen y y z en $n = 1$ y $n = 2$:

$t = 0.5$:

$$\begin{aligned} y'_0 &= z_0 = 0 \\ z'_0 &= 0.05z_0 - 0.15y_0 = -0.15 \\ y_1 &= y_0 + hy'_0 = 1 + (0.5)(0) = 1 \\ z_1 &= z_0 + hz'_0 = 0 + (0.5)(-0.15) = -0.075 \end{aligned}$$

$t = 1$:

$$\begin{aligned} y'_1 &= z_1 = -0.075 \\ z'_1 &= 0.05z_1 - 0.15y_1 = (0.05)(-0.075) - (0.15)(1) = -0.15375 \\ y_2 &= y_1 + hy'_1 = 1 + (0.5)(-0.075) = 0.96250 \\ z_2 &= z_1 + hz'_1 = -0.075 + (0.5)(-0.15375) = -0.15187 \end{aligned}$$

Por lo tanto

$$\begin{aligned} y(1) &= y_2 = 0.96250 \\ y'(1) &= z(1) = z_2 = -0.15187 \end{aligned}$$

Ejemplo 9.3

Resuelva el siguiente conjunto de EDO mediante el método de Euler hacia adelante, utilizando $h = 0.005\pi$ y $h = 0.0005\pi$:

$$\begin{aligned} y' &= z, \quad & y(0) &= 1 \\ z' &= -y, \quad & z(0) &= 0 \end{aligned} \tag{A}$$

(Solución)

Los cálculos para los primeros pasos, con $h = 0.0005\pi$ son:

$$\begin{aligned}
t_0 = 0: \quad & y_0 = 1 \\
& z_0 = 0 \\
t_1 = 0.0005\pi: \quad & y_1 = y_0 + hz_0 = 1 + (0.0005\pi)(0) = 1.0 \\
& z_1 = z_0 - hy_0 = 0 - (0.0005\pi)(1) = -0.00157 \\
t_2 = 0.001\pi: \quad & y_2 = y_1 + hz_1 = 1 + (0.0005\pi)(-0.00157) = 0.99999 \\
& z_2 = z_1 - hy_1 = -0.00157 - (0.0005\pi)(1) = -0.00314
\end{aligned}$$

En la tabla 9.3 se comparan los resultados de estos cálculos para valores seleccionados de t con los de la solución exacta, $y = \cos(t)$ y $z = -\text{sen}(t)$.

Tabla 9.3

	Exacto		$h = 0.005\pi$		$h = 0.0005\pi$	
t	$y = \cos(t)$	$z = -\text{sen}(t)$	y	z	y	z
0.5π	0	−1	1.32E − 4	−1.01241	2.62E − 6	−1.00123
π	−1	0	−1.02497	−2.67E − 4	−1.00247	−5.25E − 6
1.5π	0	1	−4.01E − 4	1.03770	−7.88E − 6	1.00371
2π	1	0	1.05058	5.48E − 4	1.00495	1.05E − 5
3π	−1	0	−1.07682	−8.43E − 4	−1.00743	−1.58E − 5
6π	1	0	1.15954	1.82E − 3	1.01491	3.19E − 5
8π	1	0	1.21819	2.54E − 3	1.01994	4.27E − 5

En esta tabla se puede observar que el error crece al incrementar t y es proporcional a h. (Véanse los valores de y para $t = \pi$, 2π, 3π, 6π y 8π; los valores de z no siguen este patrón, puesto que cuando z es cercano a cero, los errores de z se ven afectados en forma significativa por el desfasamiento.)

9.2.3 Método de Euler modificado

El método de Euler modificado tiene dos motivaciones. La primera es que es más preciso que el anterior. La segunda es que es más estable.

Este método se obtiene al aplicar la regla del trapecio para integrar $y' = f(y, t)$:

$$y_{n+1} = y_n + \frac{h}{2}\left[f(y_{n+1}, t_{n+1}) + f(y_n, t_n)\right] \tag{9.2.6}$$

Si f es lineal en y, la ecuación (9.2.6) se puede resolver fácilmente en términos de y_{n+1}. Por ejemplo, si la EDO está dada por

$$y' = ay + \cos(t)$$

La ecuación (9.2.6) queda

$$y_{n+1} = y_n + \frac{h}{2}[ay_{n+1} + \cos(t_{n+1}) + ay_n + \cos(t_n)]$$

Por lo tanto, al despejar y_{n+1} se obtiene

$$y_{n+1} = \frac{1 + ah/2}{1 - ah/2} y_n + \frac{h/2}{1 - ah/2}[\cos(t_{n+1}) + \cos(t_n)] \tag{9.2.7}$$

Si f es una función no lineal de y, la ecuación (9.2.6) es una función no lineal de y_{n+1}, por lo que se requiere un algoritmo para resolver ecuaciones no lineales. Uno de los métodos ampliamente utilizados para resolver ecuaciones no lineales es el de la sustitución sucesiva (sección 3.6):

$$y_{n+1}^{(k)} - y_n = \frac{h}{2}[f(y_{n+1}^{(k-1)}, t_{n+1}) + f(y_n, t_n)] \tag{9.2.8}$$

donde $y_{n+1}^{(k)}$ es la k-ésima aproximación iterativa de y_{n+1}, y $y_{n+1}^{(0)}$ es una estimación inicial de y_{n+1}. Esta iteración se detiene si $|y_{n+1}^{(k)} - y_{n+1}^{(k-1)}|$ es menor que una cierta tolerancia prestablecida. La estimación inicial es igual a y_n. Entonces, el primer paso de iteración es idéntico al método de Euler hacia adelante. En el caso de que sólo se utilice un paso más de iteración, el esquema se convierte en el método de Runge-Kutta de segundo orden o, en forma equivalente, en el método predictor-corrector de Euler. Pero en el método de Euler modificado, la iteración sigue hasta satisfacer la tolerancia.

El ejemplo 9.4 muestra una aplicación del método de Euler modificado al caso de una EDO no lineal de primer orden.

Ejemplo 9.4

Resuelva la siguiente ecuación mediante el método de Euler modificado, con $h = 0.1$:

$$y' = -y^{1.5} + 1, \quad y(0) = 10$$

con $0 \leqslant t \leqslant 1$. Imprima los resultados hasta $t = 1$.

(Solución)

El método de Euler modificado es

$$y_{n+1} = y_n + \frac{h}{2}[-(y_{n+1})^{1.5} - (y_n)^{1.5} + 2] \tag{A}$$

Para $n = 0$:

$$y_1 = y_0 + \frac{h}{2}[-(y_1)^{1.5} - (y_0)^{1.5} + 2]$$

y_0 es la mejor estimación del lado derecho para y_1. Sustituimos $y_1 \simeq y_0$ del lado derecho y obtenemos

$$y_1 \simeq 10 + \frac{0.1}{2}[-(10)^{1.5} - (10)^{1.5} + 2] = 6.93772$$

Incorporamos este valor de y_1 en la ecuación (A) y tenemos

$$y_1 \simeq 10 + \frac{0.1}{2}[-(6.93772)^{1.5} - (10)^{1.5} + 2] = 7.60517$$

Repetimos la sustitución unas veces más y obtenemos

$$y_1 \simeq 10 + \frac{0.1}{2}[-(7.60517)^{1.5} - (10)^{1.5} + 2] = 7.47020$$

$$y_1 \simeq 10 + \frac{0.1}{2}[-(7.47020)^{1.5} - (10)^{1.5} + 2] = 7.49799$$

$$y_1 \simeq 10 + \frac{0.1}{2}[-(7.49799)^{1.5} - (10)^{1.5+2}] = 7.49229$$

$$\vdots$$

$$y_1 = 10 + \frac{0.1}{2}[-(7.49326)^{1.5} - (10)^{1.5} + 2] = 7.49326$$

Los resultados calculados para diez pasos son:

t	y
0.0	10.0
0.1	7.4932
0.2	5.8586
0.3	4.7345
0.4	3.9298
0.5	3.3357
0.6	2.8859
0.7	2.5386
0.8	2.2658
0.9	2.0487
1.0	1.8738

¿Por qué es mejor la precisión del método modificado con respecto al método de Euler hacia adelante? Para dar una explicación analítica a esta pregunta, consideremos una ecuación de prueba, $y' = \alpha y$. Entonces, la ecuación 9.2.6 para este problema es

$$y_{n+1} = y_n + \frac{\alpha h}{2}(y_{n+1} + y_n) \tag{9.2.9}$$

o bien, al despejar y_{n+1},

$$y_{n+1} = \left(1 + \frac{\alpha h}{2}\right)\left(1 - \frac{\alpha h}{2}\right)^{-1} y_n$$

Desarrollamos los coeficientes de esa ecuación y tenemos

$$y_{n+1} = \left(1 + \alpha h + \frac{1}{2}(\alpha h)^2 + \frac{1}{4}(\alpha h)^3 + \cdots\right) y_n$$

Al hacer la comparación de este desarrollo con la serie de Taylor del valor exacto, $y(t_{n+1}) = \exp(\alpha h) y_n$, tenemos que la ecuación (9.2.9) es exacta hasta el término de segundo orden. Así, el método de Euler modificado es un método (preciso) de segundo orden. Por otro lado, un análisis similar del método de Euler hacia adelante muestra que tiene una precisión de primer orden.

El error local (generado en cada paso) del método de Euler hacia adelante es proporcional a h^2, mientras que el error global es proporcional a h; en tanto que el error local del método de Euler modificado es proporcional a h^3 y su error global es proporcional a h^2. El orden del error del método de Euler hacia atrás es igual al del método de Euler hacia adelante.

Al aplicar el método de Euler modificado a un conjunto de EDO, la ecuación debe resolverse en forma simultánea o "implícita". Sin embargo, la ventaja de la solución implícita consiste en el hecho de que el método es más estable que el método de Euler hacia adelante, por lo que permite un mayor intervalo de tiempo.

9.2.4 Método de Euler hacia atrás

Este método se basa en aproximación por diferencias hacia atrás y se escribe como

$$y_{n+1} = y_n + hf(y_{n+1}, t_{n+1}) \tag{9.2.10}$$

La precisión de este método es la misma que la del de Euler hacia adelante. Además, si f es una función no lineal de y, debe utilizarse un esquema iterativo en cada paso, justo como en el método de Euler modificado. Sin embargo, las ventajas son: a) el método es estable para los problemas rígidos, y b) la solución es positiva si la solución exacta es positiva. Véanse aplicaciones del método de Euler hacia atrás en la sección 9.5 y el capítulo 12.

Resumen de Esta Sección

a) El método de Euler hacia adelante utiliza la aproximación por diferencias hacia adelante. Su error en un intervalo es proporcional a h^2, mientras que su error global es proporcional a h. El método de Euler hacia adelante podría ser inestable si la EDO tiene una constante de tiempo con signo negativo, a menos que se utilice una h pequeña.

b) El método de Euler modificado utiliza la regla del trapecio. Si la EDO no es lineal, se requiere un método iterativo para cada intervalo. Su error en un intervalo es proporcional a h^3, mientras que su error global lo es a h^2.

c) El método de Euler hacia atrás utiliza la aproximación por diferencias hacia atrás. Sus errores son análogos a los del método de Euler hacia adelante. El método es estable, por lo que se puede utilizar para resolver problemas rígidos que son difíciles de resolver por otros medios.

9.3 METODOS DE RUNGE-KUTTA

Una desventaja fundamental de los métodos de Euler consiste en que los órdenes de precisión son bajos. Esta desventaja tiene dos facetas. Para mantener una alta precisión se necesita una h pequeña, lo que aumenta el tiempo de cálculo y provoca errores de redondeo.

En los métodos de Runge-Kutta, el orden de precisión aumenta al utilizar puntos intermedios en cada intervalo. Una mayor precisión implica además que los errores decrecen más rápido al reducir h, en comparación con los métodos con precisión baja.

Consideremos una ecuación diferencial ordinaria

$$y' = f(y, t), \quad y(0) = y_0 \tag{9.3.1}$$

Para calcular y_{n+1} en $t_{n+1} = t_n + h$, dado un valor de y_n, integramos la ecuación (9.3.1) en el intervalo $[t_n, t_{n+1}]$:

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f(y, t)\, dt \tag{9.3.2}$$

Los métodos de Runge-Kutta se obtienen al aplicar un método de integración numérica al integral del lado derecho de la ecuación (9.3.2) [Fox/Mayers]. En el resto de esta sección analizaremos los métodos de Runge-Kutta de segundo, tercero y cuarto orden.

9.3.1 Método de Runge-Kutta de segundo orden

Aplicamos la regla del trapecio al lado derecho de la ecuación (9.3.2):

$$\int_{t_n}^{t_{n+1}} f(y, t)\, dt \simeq \frac{1}{2} h[f(y_n, t_n) + f(y_{n+1}, t_{n+1})] \tag{9.3.3}$$

En esta ecuación, y_{n+1} es una incógnita, por lo que aproximamos el segundo término mediante $f(\bar{y}_{n+1}, t_{n+1})$, donde $\bar{y}_{n+1}$ es la primera estimación de y_{n+1} obtenida mediante el método de Euler hacia adelante. Este esquema se conoce como el método de Runge-Kutta de segundo orden y se resume como

$$\bar{y}_{n+1} = y_n + hf(y_n, t_n)$$

$$y_{n+1} = y_n + \frac{h}{2}[f(y_n, t_n) + f(\bar{y}_{n+1}, t_{n+1})]$$

Una forma canónica de lo anterior es

$$\begin{aligned} k_1 &= hf(y_n, t_n) \\ k_2 &= hf(y_n + k_1, t_{n+1}) \\ y_{n+1} &= y_n + \frac{1}{2}[k_1 + k_2] \end{aligned} \tag{9.3.4}$$

El método de Runge-Kutta de segundo orden es idéntico al método predictor-corrector de Euler, que es el método más simple de este tipo (véase la sección 9.4). También es equivalente al método modificado de Euler, con únicamente dos pasos de iteración.

Ejemplo 9.5

El circuito que se muestra en la figura E9.5 tiene una autoinductancia de $L = 50$H, una resistencia de $R = 20$ ohms y una fuente de voltaje de $V = 10$ vols. Si el interruptor se cierra en el instante $t = 0$, la corriente $I(t)$ satisface la ecuación

$$L\frac{d}{dt}I(t) + RI(t) = E, \quad I(0) = 0 \tag{A}$$

Determine el valor de la corriente para $0 < t \leqslant 10$ segundos, mediante el método de Runge-Kutta de segundo orden, con $h = 0.1$.

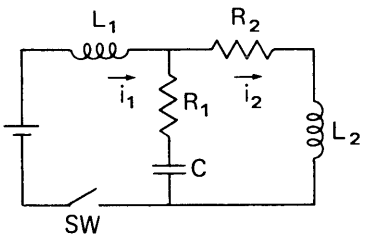


Figura E9.5 Un circuito eléctrico

(Solución)

En primer lugar, reescribimos la ecuación (A) como

$$\frac{d}{dt}I = -\frac{R}{L}I + \frac{E}{L} \equiv f(I, t)$$

Podemos desarrollar entonces el método de Runge-Kutta de segundo orden como

$$k_1 = h\left[-\frac{R}{L}I_n + \frac{E}{L}\right]$$

$$k_2 = h\left[-\frac{R}{L}(I_n + k_1) + \frac{E}{L}\right]$$

$$I_{n+1} = I_n + \frac{1}{2}(k_1 + k_2)$$

Los cálculos de los primeros dos pasos son los siguientes:

$$n = 0\ (t = 0.1):\quad k_1 = 0.1[(-0.4)(0) + 0.2] = 0.02$$

$$k_2 = 0.1[(-0.4)(0 + 0.02) + 0.2] = 0.0192$$

$$I_1 = I_0 + \frac{1}{2}(k_1 + k_2) = 0 + \frac{1}{2}(0.02 + 0.0192) = 0.0196$$

$$n = 1\ (t = 0.2):\quad k_1 = 0.1[(-0.4)(0.0196) + 0.2] = 0.019216$$

$$k_2 = 0.1[(-0.4)(0.0196 + 0.019216) + 0.2] = 0.018447$$

$$I_2 = I_1 + \frac{1}{2}(k_1 + k_2)$$

$$= 0.0196 + \frac{1}{2}(0.019216 + 0.018447) = 0.038431$$

El resultado final de los cálculos (en múltiplos de 10 pasos) es:

t (seg)	I (amp)
0	0
1	0.1648
2	0.2752
3	0.3493
4	0.3990
5	0.4332
6	0.4546
7	0.4695
8	0.4796
9	0.4863
10	0.4908
(∞)	(0.5000)

Podemos analizar la precisión del método de Runge-Kutta de segundo orden mediante la ecuación de prueba $y' = \alpha y$, como se describe al final de la sección 9.2. Sin embargo, para ser más formales, consideremos una forma genérica, $y' = f(y, x)$. Primero desarrollamos el valor exacto de y_{n+1} en una serie de Taylor:

$$y_{n+1} = y_n + hf + \frac{h^2}{2}[f_t + f_y f] + \frac{h^3}{6}[f_{tt} + 2f_{ty}f + f_{yy}f^2 + f_t f_y + f_y^2 f] + 0(h^4) \tag{9.3.5}$$

en donde todas las derivadas de y se expresan en términos de f y de sus derivadas parciales evaluadas en t_n.

Ahora desarrollamos la segunda ecuación de (9.3.4) en serie de Taylor:

$$y_{n+1} = y_n + hf + \frac{h^2}{2}[f_t + f_y f] + \frac{h^3}{4}[f_{tt} + 2f_{ty}f + f_{yy}f^2] + 0(h^4) \tag{9.3.6}$$

Al comparar las ecuaciones (9.3.6) y (9.3.5) podemos observar que (9.3.4) tiene una exactitud del orden de h^2 y una discrepancia (que es el error generado en cada paso) proporcional a h^3. Conviene hacer notar que el método de Runge-Kutta de segundo orden es idéntico al método modificado de Euler [ecuación (9.2.8)] con dos pasos de iteración. Sin embargo, el orden de precisión de ambos métodos es idéntico, aunque el segundo necesite una convergencia iterativa. Esto es un indicador de que la iteración del método modificado de Euler sólo mejora un poco la precisión. (De hecho, el uso del método de Runge-Kutta de segundo orden con una h más pequeña mejora la precisión de una manera más eficaz que si se usara el método modificado de Euler con una convergencia iterativa estricta.) Podríamos haber realizado el análisis anterior más fácilmente con la ecuación de prueba $y' = \alpha y$, pero esta vía la dejamos como ejercicio.

Es fácil aplicar el método de Runge-Kutta de segundo orden a una ecuación diferencial ordinaria de orden superior. A manera de ejemplo, consideremos la ecuación diferencial de segundo orden:

$$y''(t) + ay'(t) + by(t) = q(t), \quad y(0) = 1, y'(0) = 0 \tag{9.3.7}$$

donde a y b son los coeficientes y $q(t)$ es una función conocida, al igual que las condiciones iniciales. Definimos

$$z(t) = y'(t) \tag{9.3.8}$$

La ecuación (9.3.7) se reduce a unas ecuaciones diferenciales simultáneas de primer orden:

$$\begin{aligned} y' &= f(y, z, t) \equiv z, & y(0) &= 1 \\ z' &= g(y, z, t) \equiv -az - by + q, & z(0) &= 0 \end{aligned} \tag{9.3.9}$$

Podemos escribir como sigue el método de Runge-Kutta de segundo orden para estas ecuaciones:

$$\begin{aligned} k_1 &= hf(y_n, z_n, t_n) = hz_n \\ l_1 &= hg(y_n, z_n, t_n) = h(-az_n - by_n + q_n) \\ k_2 &= hf(y_n + k_1, z_n + l_1, t_{n+1}) = h(z_n + l_1) \\ l_2 &= hg(y_n + k_1, z_n + l_1, t_{n+1}) = h(-a(z_n + l_1) - b(y_n + k_1) + q_{n+1}) \\ y_{n+1} &= y_n + \frac{1}{2}(k_1 + k_2) \\ z_{n+1} &= z_n + \frac{1}{2}(l_1 + l_2) \end{aligned} \tag{9.3.10}$$

Ejemplo 9.6

Cierto material de forma cúbica, con una masa de $M = 0.5$ kg se pone en el extremo inferior de un resorte sin masa. El extremo superior se fija a una estructura en reposo. El cubo recibe una resistencia de $R = -B\,dy/dt$ del aire,

donde B es una constante de amortiguamiento (véase la figura E9.6). La ecuación de movimiento es

$$M\frac{d^2}{dt^2}y + B\frac{d}{dt}y + ky = 0, \quad y(0) = 1,\ y'(0) = 0 \tag{A}$$

donde y es el desplazamiento desde la posición estática, $k = 100$ kg/seg^2 es la constante del resorte y $B = 10$ kg/seg.

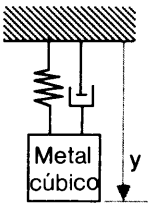


Figura E9.6 Un sistema de masa-resorte

a) Calcule a mano $y(t)$, para $0 < t < 0.05$ mediante el método de Runge-Kutta de segundo orden y $h = 0.025$.

b) Calcule $y(t)$, para $0 < t < 10$ segundos con el método de Runge-Kutta de segundo orden y $h = 0.001$.

c) Repita el cálculo con $B = 0$.

(Solución)

Podemos escribir la ecuación (A) como

$$\begin{aligned} y' &= z \equiv f(y, z, t), & y(0) &= 1 \\ z' &= -\frac{B}{M}z - \frac{k}{M}y \equiv g(y, z, t), & z(0) &= 0 \end{aligned} \tag{B}$$

Sean $a = B/M = 20$, $b = k/M = 200$ y $g = 0$; así, el método de Runge-Kutta de segundo orden para la ecuación (A) toma la forma de la ecuación (9.3.9).

a) Para $n = 1$: $t = 0.025$

$$k_1 = hf(y_0, z_0, t_0) = hz_0 = 0.025(0) = 0$$

$$l_1 = hg(y_0, z_0, t_0) = h(-20z_0 - 200y_0) = 0.025(-20(0) - 200(1)) = -5$$

$$k_2 = hf(y_0 + k_1, z_0 + l_1, t_0) = h(z_0 + l_1) = 0.025(0 - 5) = -0.125$$

$$\begin{aligned} l_2 = hg(y_0 + k_1, z_0 + l_1, t_1) &= h[-20(z_0 + l_1) - 200(y_0 + k_1)] \\ &= 0.025[-20(0 - 5) - 200(1 + 0)] = -2.5 \end{aligned}$$

$$y_1 = y_0 + \frac{1}{2}(0 - 0.125) = 0.9375$$

$$z_1 = z_0 + \frac{1}{2}(-5 - 2.5) = -3.75$$

Para $n = 2$: $t = 0.05$

$$k_1 = hf(y_1, z_1, t_1) = hz_1 = 0.025(-3.75) = -0.09375$$

$$\begin{aligned} l_1 = hg(y_1, z_1, t_1) &= h(-20z_1 - 200y_1) \\ &= 0.025[-20(-3.75) - 200(0.9375)] = -2.8125 \end{aligned}$$

$$k_2 = hf(y_1 + k_1, z_1 + l_1, t_1) = h(z_1 + l_1)$$
$$= 0.025(-3.75 - 2.8125) = 0.1640625$$
$$l_2 = hg(y_1 + k_1, z_1 + l_1, t_1) = h[-20(z_1 + l_1) - 200(y_1 + k_1)]$$
$$= 0.025[-20(-3.75 - 2.8125) - 200(0.9375 - 0.093750)]$$
$$= -0.9375$$
$$y_2 = y_1 + \frac{1}{2}(-0.09375 - 0.1640625) = 0.80859$$
$$z_2 = z_1 + \frac{1}{2}(-2.8125 - 0.9375) = -5.625$$

b) y c) En esta parte de los cálculos utilizamos el PROGRAMA 9-1. Abajo se muestran los resultados computacionales después de cada 50 pasos hasta 0.75 segundos:

t (seg)	b) y (metros) ($B = 10$)	c) y (metros) ($B = 0$)
0	1.000	1.000
0.05	0.823	0.760
0.1	0.508	0.155
0.15	0.238	−0.523
0.2	0.066	−0.951
0.25	−0.016	−0.923
0.3	−0.042	−0.45
0.35	−0.038	0.235
0.4	−0.025	0.810
0.45	−0.013	0.996
0.5	−0.004	0.705
0.55	0.000	0.075
0.6	0.001	−0.590
0.65	0.001	−0.973
0.7	0.001	−0.889
0.75	0.000	−0.378

9.3.2 Método de Runge-Kutta de tercer orden

Un método de Runge-Kutta más preciso que el anterior es resultado de un esquema de integración numérica de orden superior para el segundo término de la ecuación (9.3.2). Con la regla de 1/3 de Simpson, la ecuación (9.3.2) es:

$$y_{n+1} = y_n + \frac{h}{6}[f(y_n, t_n) + 4f(\bar{y}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) + f(\bar{y}_{n+1}, t_{n+1})] \tag{9.3.11}$$

donde $\bar{y}_{n+1}$ y $\bar{y}_{n+\frac{1}{2}}$ son estimaciones, puesto que no conocemos $y_{n+\frac{1}{2}}$ y y_{n+1}.

Obtenemos la estimación $\bar{y}_{n+\frac{1}{2}}$ mediante el método de Euler hacia adelante:

$$\bar{y}_{n+\frac{1}{2}} = y_n + \frac{h}{2}f(y_n, t_n) \tag{9.3.12}$$

La estimación $\bar{y}_{n+1}$ es

$$\bar{y}_{n+1} = y_n + hf(y_n, t_n)$$

o bien

$$\bar{y}_{n+1} = y_n + hf(\bar{y}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}})$$

o una combinación lineal de ambas

$$\bar{y}_{n+1} = y_n + h[\theta f(y_n, t_n) + (1 - \theta)f(\bar{y}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}})] \tag{9.3.13}$$

Donde θ es un parámetro que hay que determinar de forma que maximice la precisión del método numérico. Con la ecuación (9.3.13), el esquema global tiene la forma siguiente:

$$\begin{aligned} k_1 &= hf(y_n, t_n) \\ k_2 &= hf\left(y_n + \frac{1}{2}k_1, t_n + \frac{h}{2}\right) \\ k_3 &= hf(y_n + \theta k_1 + (1 - \theta)k_2, t_n + h) \\ y_{n+1} &= y_n + \frac{1}{6}(k_1 + 4k_2 + k_3) \end{aligned} \tag{9.3.14}$$

Para optimizar θ, desarrollamos k_1, k_2 y k_3 en serie de Taylor:

$$k_1 = hf \tag{9.3.15a}$$

$$k_2 = hf + \frac{1}{2}h^2(f_t + f_y f) + \frac{1}{8}h^3(f_{tt} + 2f_{ty}f + f_{yy}f^2) \tag{9.3.15b}$$

$$\begin{aligned} k_3 = hf &+ h^2(f_t + f_y f) + \frac{1}{2}h^3[f_{tt} + 2f_{ty}f \\ &+ f_{yy}f^2 + (1 - \theta)(f_t + f_y f)f_y] \end{aligned} \tag{9.3.15c}$$

donde f y sus derivadas se evalúan en t_n. Sustituimos la ecuación (9.3.15) en la ecuación (9.3.14) y comparándola con la ecuación (9.3.5), determinamos que $\theta = -1$ es el óptimo, puesto que en este caso la ecuación (9.3.14) coincide con la ecuación (9.3.5) hasta el término de tercer orden.

El desarrollo anterior es más fácil de entender si se aplica a la ecuación de prueba $y' = \alpha y$.

En resumen, el método de Runge-Kutta con una precisión de tercer orden se escribe como

$$\begin{aligned} k_1 &= hf(y_n, t_n) \\ k_2 &= hf\left(y_n + \frac{1}{2}k_1, t_n + \frac{h}{2}\right) \\ k_3 &= hf(y_n - k_1 + 2k_2, t_n + h) \\ y_{n+1} &= y_n + \frac{1}{6}(k_1 + 4k_2 + k_3) \end{aligned} \tag{9.3.16}$$

9.3.3 Método de Runge-Kutta de cuarto orden

El método de Runge-Kutta de cuarto orden se obtiene de una manera análoga a la del tercer orden, excepto que se utiliza un paso intermedio adicional para evaluar la derivada. Podemos escoger de varias formas el esquema de integración numérica que utilizaremos en la ecuación (9.3.2). El método de Runge-Kutta de cuarto orden tiene una precisión hasta el término de cuarto orden del desarrollo de Taylor, por lo que el error local es proporcional a h^5.

Las siguientes dos versiones del método de Runge-Kutta de cuarto orden son las de uso más popular. La primera se basa en la regla de 1/3 de Simpson y se escribe como

$$\begin{aligned} k_1 &= hf(y_n, t_n) \\ k_2 &= hf\left(y_n + \frac{k_1}{2}, t_n + \frac{h}{2}\right) \\ k_3 &= hf\left(y_n + \frac{k_2}{2}, t_n + \frac{h}{2}\right) \\ k_4 &= hf(y_n + k_3, t_n + h) \\ y_{n+1} &= y_n + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4] \end{aligned} \tag{9.3.17}$$

La segunda versión se basa en la regla de 3/8 de Simpson y se expresa como

$$\begin{aligned} k_1 &= hf(y_n, t_n) \\ k_2 &= hf\left(y_n + \frac{k_1}{3}, t_n + \frac{h}{3}\right) \\ k_3 &= hf\left(y_n + \frac{k_1}{3} + \frac{k_2}{3}, t_n + \frac{2h}{3}\right) \\ k_4 &= hf(y_n + k_1 - k_2 + k_3, t_n + h) \\ y_{n+1} &= y_n + \frac{1}{8}[k_1 + 3k_2 + 3k_3 + k_4] \end{aligned} \tag{9.3.18}$$

Ejemplo 9.7

Calcule $y(1)$ resolviendo

$$y' = -1/(1 + y^2), \quad y(0) = 1$$

por medio del método de Runge-Kutta de cuarto orden, con $h = 1$.

(Solución)

Hacemos

$$f(y, t) = -\frac{1}{1 + y^2}$$

y $y_0 = 1$ y $t_0 = 0$. Puesto que sólo tenemos un intervalo, los cálculos son:

$$k_1 = hf(y_0, t_0) = -\frac{1}{(1 + 1)} = -\frac{1}{2}$$

$$k_2 = hf\left(y_0 + \frac{k_1}{2}, t_0 + \frac{h}{2}\right) = -\frac{1}{(1 + (0.75)^2)} = -0.64$$

$$k_3 = hf\left(y_0 + \frac{k_2}{2}, t_0 + \frac{h}{2}\right) = -\frac{1}{(1 + (0.68)^2)} = -0.6838$$

$$k_4 = hf(y_0 + k_3, t_0 + h) = -\frac{1}{(1 + (0.3161)^2)} = -0.9091$$

$$y_1 = y_0 + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4]$$

$$= 1 + \frac{1}{6}[-0.5 - 2(0.64) - 2(0.6838) - 0.9091] = 0.3238$$

Ejemplo 9.8

Resuelva

$$y' = ty + 1, \quad y(0) = 0$$

mediante el método de Runge-Kutta de cuarto orden —ecuación (9.3.17)— con $h = 0.2$, 0.1 y 0.05, respectivamente; evalúe el error para cada h en $t = 1, 2, 3, 4$ y 5.

(Solución)

Los cálculos de este ejemplo se realizaron con el PROGRAMA 9-2. Los resultados son los siguientes:

	$h = 0.2$		$h = 0.1$		$h = 0.05$	
t	y	e.p.	y	e.p.	y	e.p.[a]
1	1.41067	(0.00)	1.41069	(0.00)	1.41068	(0.00)
2	8.83839	(0.01)	8.83937	(0.00)	8.83943	(0.00)
3	112.394	(0.11)	112.506	(0.01)	112.514	(0.00)
4	3716.42	(0.52)	3734.23	(0.04)	3735.72	(0.00)
5	330549.	(1.71)	335798.	(0.15)	336273.	(0.01)

[a] e.p.: error porcentual.

Al comparar estos resultados con los del método de Euler, tenemos que el error del método de Runger-Kutta de cuarto orden, con $h = 0.1$, es comparable con el error del método modificado de Euler para $h = 0.01$. Además, el método de Runge-Kutta de cuarto orden con $h = 0.2$ es comparable con el método de Euler hacia adelante para $h = 0.001$.

La aplicación de método de Runge-Kutta de cuarto orden a un conjunto de ecuaciones diferenciales ordinarias es análoga a la aplicación del método de segundo orden. Con el fin de simplificar la explicación, consideremos un conjunto de dos ecuaciones:

$$\begin{aligned} y' &= f(y, z, t) \\ z' &= g(y, z, t) \end{aligned} \tag{9.3.19}$$

El método de Runge-Kutta de cuarto orden para este conjunto es

$$\begin{aligned} k_1 &= hf(y_n, z_n, t_n) \\ l_1 &= hg(y_n, z_n, t_n) \\ k_2 &= hf\left(y_n + \frac{k_1}{2}, z_n + \frac{l_1}{2}, t_n + \frac{h}{2}\right) \\ l_2 &= hg\left(y_n + \frac{k_1}{2}, z_n + \frac{l_1}{2}, t_n + \frac{h}{2}\right) \\ k_3 &= hf\left(y_n + \frac{k_2}{2}, z_n + \frac{l_2}{2}, t_n + \frac{h}{2}\right) \\ l_3 &= hg\left(y_n + \frac{k_2}{2}, z_n + \frac{l_2}{2}, t_n + \frac{h}{2}\right) \\ k_4 &= hf(y_n + k_3, z_n + l_3, t_n + h) \\ l_4 &= hg(y_n + k_3, z_n + l_3, t_n + h) \end{aligned} \tag{9.3.20}$$

$$y_{n+1} = y_n + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4] \tag{9.3.21}$$

$$z_{n+1} = z_n + \frac{1}{6}[l_1 + 2l_2 + 2l_3 + l_4] \tag{9.3.22}$$

Incluso cuando el número de ecuaciones en un conjunto es mayor que dos, el método de Runge-Kutta de cuarto orden es esencialmente el mismo. En el PROGRAMA 9-3 se da un programa para resolver un conjunto de ecuaciones con el método de Runge-Kutta de cuarto orden.

Ejemplo 9.9

Repita el problema del ejemplo 9.3 con el método de Runge-Kutta de cuarto orden con $h = 0.2\,\pi$ y $h = 0.05\,\pi$.

(Solución)

Utilizamos el PROGRAMA 9-3 para obtener los siguientes resultados:

	Valor exacto		$h = 0.2\pi$		$h = 0.05\pi$	
t	$y = \cos(t)$	$z = -\text{sen}(t)$	y	z	y	z
0.5π	0	−1	1.23E−4	−0.99997	1.32E−6	−0.99999
π	−1	0	−0.99993	−2.48E−4	−0.99999	−2.65E−6
1.5π	0	1	−3.72E−4	0.99990	−3.96E−6	0.99999
2π	1	0	0.99987	4.95E−4	0.99999	5.29E−6
3π	−1	0	−0.99989	−7.43E−4	−0.99999	−7.94E−6
6π	1	0	0.99960	1.49E−3	0.99999	1.57E−5
8π	1	0	0.99947	1.98E−3	0.99999	2.11E−5

La comparación de estos valores con los resultados de la solución de Euler hacia adelante en el ejemplo 9.3 muestra que la precisión del método de Runge-Kutta de cuarto orden (incluso con $h = 0.2\pi$) es significativamente mejor que el método de Euler hacia adelante para $h = 0.01\pi$.

9.3.4 Error, estabilidad y optimización del intervalo de la retícula

Los métodos de Runge-Kutta están sujetos a dos tipos de errores: el error de truncamiento y el de inestabilidad. Como ya se ha analizado, el error de truncamiento se debe a la discrepancia entre el desarrollo de Taylor del método numérico y el de la solución exacta. El tamaño del error decrece al aumentar el orden del método. Por otro lado, la inestabilidad es un efecto acumulado del error local, de forma que el error de la solución crece sin límite al avanzar los intervalos de tiempo.

Para analizar la estabilidad de un método de Runge-Kutta consideremos la ecuación de prueba

$$y' = \alpha y \tag{9.3.23}$$

donde $\alpha < 0$. Para y_n dada, el valor exacto de y_{n+1} está dada en forma analítica por por

$$y_{n+1} = \exp(\alpha h) y_n \tag{9.3.24}$$

Conviene observar que $|y_{n+1}|$ decrece cuando n (o el tiempo) aumenta, ya que $\alpha < 0$.

La solución numérica de la ecuación (9.3.23) mediante el método de Runge-Kutta de cuarto orden es
becomes

$$\begin{aligned}
k_1 &= \alpha h y_n \\
k_2 &= \alpha h\left(y_n + \frac{k_1}{2}\right) = \alpha h\left(1 + \frac{1}{2}\alpha h\right) y_n \\
k_3 &= \alpha h\left(y_n + \frac{k_2}{2}\right) = \alpha h\left(1 + \frac{1}{2}\alpha h\left(1 + \frac{1}{2}\alpha h\right)\right) y_n \\
k_4 &= \alpha h(y_n + k_3) = \alpha h\left(1 + \alpha h\left(1 + \frac{1}{2}\alpha h\left(1 + \frac{1}{2}\alpha h\right)\right)\right) y_n
\end{aligned} \tag{9.3.25}$$

$$y_{n+1} = \left[1 + \alpha h + \frac{1}{2}(\alpha h)^2 + \frac{1}{6}(\alpha h)^3 + \frac{1}{24}(\alpha h)^4\right] y_n \tag{9.3.26}$$

La ecuación (9.3.26) es igual a los primeros cinco términos del desarrollo de Taylor para el lado derecho de la ecuación (9.3.24) alrededor de t_n. El factor

$$\gamma = 1 + \alpha h + \frac{1}{2}(\alpha h)^2 + \frac{1}{6}(\alpha h)^3 + \frac{1}{24}(\alpha h)^4 \tag{9.3.27}$$

de la ecuación (9.3.26) aproxima a exp (αh) de la ecuación (9.3.24), por lo que en esta aproximación se originan tanto el error de truncamiento como la inestabilidad de la ecuación (9.3.26).

En la figura 9.1 se grafican juntos la ecuación (9.3.27) y exp (αh), para poderlas comparar. La figura indica que si $\alpha < 0$ y el módulo (valor absoluto) de αh aumenta, crece la desviación de γ con respecto de exp (αh), por lo que se incrementa el error del método de Runge-Kutta. En particular, cuando $\alpha h \leq -2.785$, el método se vuelve inestable, debido a que el módulo de la solución numérica crece a cada paso, mientras que el módulo de la solución verdadera decrece en cada paso por un factor de exp (αh).

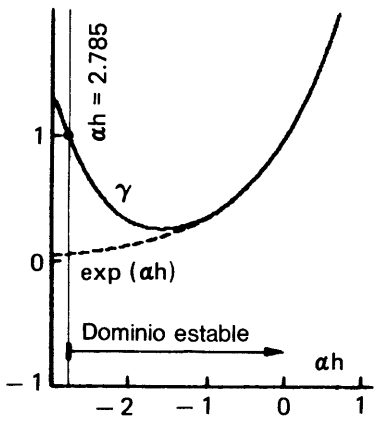


Figura 9.1 Dominio de estabilidad

En las aplicaciones prácticas del método de Runge-Kutta, es posible determinar el tamaño óptimo de un intervalo de la retícula de la manera siguiente. A manera de ejemplo, supongamos que deseamos mantener menor que ξ. El error local del método de Runge-Kutta de tercer orden. El error local de este método para un intervalo de prueba h es proporcional a h^4, por lo que expresamos el error en la forma

$$E_h = Bh^4 \tag{9.3.28}$$

donde B es una constante que depende del problema dado. Si aplicamos el mismo método de Runge-Kutta en dos pasos y con $h/2$ como intervalo de tiempo, el error resulta ser proporcional a $2(h/2)^4$, donde el factor de 2 se debe a la acumulación del error en dos etapas. Así, se tiene que

$$2E_{h/2} = 2B\left(\frac{h}{2}\right)^4 = \frac{1}{8}Bh^4 \tag{9.3.29}$$

Restamos la ecuación (9.3.29) a la ecuación (9.3.28), con lo que obtenemos

$$E_h - 2E_{h/2} = Bh^4 - \frac{1}{8}Bh^4 = \frac{7}{8}Bh^4 \tag{9.3.30}$$

Podemos evaluar el lado izquierdo de la ecuación anterior mediante un experimento numérico (es decir, ejecutamos el esquema dos veces, partiendo del mismo valor inicial). En la primera ejecución, sólo se avanza un intervalo, utilizando un valor de prueba para h. Denotamos el resultado de este cálculo como $[y_1]_h$. En la segunda ejecución, $[y_2]_{h/2}$ se calcula en dos intervalos de tiempo, con $h/2$ como intervalo. Usamos los resultados de estos dos cálculos y evaluamos el lado izquierdo de la ecuación (9.3.30) como sigue

$$E_h - 2E_{h/2} = [y_1]_h - [y_2]_{h/2} \tag{9.3.31}$$

Sustituimos la ecuación (9.3.31) en la ecuación (9.3.30) y despejamos a B.

$$B = \frac{8}{7}([y_1]_h - [y_2]_{h/2})/h^4 \tag{9.3.32}$$

Una vez determinada B, podemos encontrar el h máximo (u óptimo) que satisface el criterio de $E_h \leq \xi$ sustituyendo $E_h = \xi$ en la ecuación (9.3.28) y despejando h:

$$h = \left(\frac{\xi}{B}\right)^{0.25} \tag{9.3.33}$$

La teoría que hemos descrito recuerda la integración de Romberg, explicada en la sección 3.2.

Ejemplo 9.10

Suponga que el método de Runge-Kutta de cuarto orden se aplica a

$$y' = -\frac{y}{1+t^2}, \quad y(0) = 1$$

determine el tamaño óptimo de intervalo que satisfaga $E_h \leqslant 0.00001$.

(Solución)

Para el caso de método de Runge-Kutta de cuarto orden, el error local se expresa como

$$E_h = Bh^5 \tag{A}$$

El punto de vista es muy parecido al de las ecuaciones (9.3.28) a (9.3.33), excepto que el orden del error es cinco. El error acumulado en dos pasos, con $h/2$, es $2E_{h/2} = 2B(h/2)^5$. Evaluamos en forma numérica la diferencia entre los errores de un paso y dos pasos, $E_h - 2E_{h/2}$,

$$2E_h - 2E_{h/2} = [y_1]_h - [y_2]_{h/2}. \tag{B}$$

En la ecuación (B), $[y_1]_h$ es el resultado del método de Runge-Kutta de cuarto orden con sólo un paso y h; mientras que $[y_2]_{h/2}$ es el resultado del mismo método con dos pasos y $h/2$. Sustituimos (A) en (B) y despejamos B.

$$B = \frac{16}{15}([y_1]_h - [y_1]_{h/2})/h^5 \qquad \text{(C)}$$

En realidad, desarrollamos el método de Runge-Kutta de cuarto orden con sólo un paso y $h = 1$, partiendo de la condición inicial dada. Entonces lo ejecutamos para dos pasos con $h/2 = 1/2$. Los resultados son

$$[y_1]_1 = 0.4566667 \quad \text{(un intervalo únicamente)}$$
$$[y_2]_{1/2} = 0.4559973 \quad \text{(dos intervalos)}$$

De la ecuación (C), obtenemos el valor de B

$$B = \frac{16}{15}(0.4566667 - 0.4559973)/(1)^5 = 6.3 \times 10^{-4} \qquad \text{(D)}$$

Sustituimos esto en la ecuación (A), con lo que el error local para cualquier h es

$$E_h = 6.3 \times 10^{-4} h^5$$

El máximo h que satisface el criterio dado, $E_h < 0.00001$, es

$$h = (0.00001/6.3 \times 10^{-4})^{1/5} = 0.44 \qquad \text{(E)}$$

Resumen de Esta Sección

a) Los métodos de Runge-Kutta se obtienen al integrar la EDO de primer orden con métodos numéricos. El método de Runge-Kutta de segundo orden es idéntico al método modificado de Euler con dos ciclos de iteración y al método predictor-corrector de segundo orden.
b) Una EDO de orden superior se puede resolver mediante un método de Runge-Kutta, después de transformarla a un conjunto de EDO de primer orden.
c) Los métodos de Runge-Kutta se vuelven inestables si α es negativa y αh excede un cierto criterio.
d) Se puede calcular el error local de un método de Runge-Kutta ejecutándolo dos veces: la primera con un intervalo y un valor de h y la segunda vez con dos intervalos y $h/2$.

9.4 METODOS PREDICTOR-CORRECTOR

9.4.1 Método predictor-corrector de Adams de tercer orden

Un método predictor-corrector consta de un paso predictor y un paso corrector en cada intervalo. El predictor estima la solución para el nuevo punto y el corrector mejora su precisión. Los métodos de predictor-corrector utilizan la solución de los puntos anteriores, en lugar de utilizar puntos intermedios en cada intervalo.

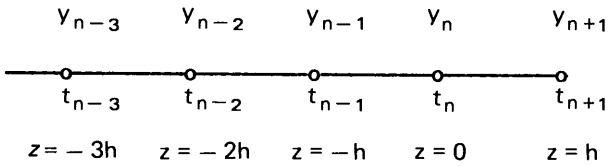


Figura 9.2 Puntos de la retícula utilizados en los métodos predictor-corrector

Para explicar los métodos, consideremos un intervalo de tiempo dividido de manera uniforme y supongamos que hemos calculado la solución hasta el tiempo n, por lo que es posible utilizar los valores de y y y' en los tiempos anteriores para calcular y_{n+1}.

Las fórmulas predictoras y correctoras se obtienen al sustituir una aproximación polinomial adecuada de $y'(t)$ en la ecuación (9.3.2). El miembro más primitivo de los métodos predictor-corrector es el de segundo orden, que es idéntico al método de Runge-Kutta de segundo orden.

Obtengamos un predictor de tercer orden al aproximar $y' = f(y, t)$ con un polinomio de interpolación cuadrática, ajustado a f'_n, y'_{n-1} y y'_{n-2}:

$$y'(z) = \frac{1}{2h^2}\left[(z+h)(z+2h)y'_n - 2z(z+2h)y'_{n-1} + z(z+h)y'_{n-2}\right] + E(z) \qquad (9.4.1)$$

donde z es una coordenada local dada por

$$z = t - t_n$$

y $E(z)$ es el error (véase la sección 2.3). La ecuación (9.4.1) es la interpolación de Lagrange ajustada a los valores y'_n, y'_{n-1} y y'_{n-2}. El error del polinomio es

$$E(z) = \frac{1}{3!} z(z+h)(z+2h)y^{(\text{iv})}(\xi), \quad t_{n-2} \leqslant \xi \leqslant t_{n+1} \qquad (9.4.2)$$

En esta ecuación, la derivada del término del error es de cuarto orden, puesto que se ha ajustado un polinomio cuadrático a y'.

La ecuación (9.3.2) se puede reescribir en términos de la coordenada local $z = t - t_n$ como

$$y_{n+1} = y_n + \int_0^h y'(z)\,dz \qquad (9.4.3)$$

Sustituimos la ecuación (9.4.1) en la ecuación (9.4.3) para obtener

$$y_{n+1} = y_n + \frac{h}{12}(23y'_n - 16y'_{n-1} + 5y'_{n-2}) + 0(h^4) \qquad (9.4.4)$$

La ecuación (9.4.4) recibe el nombre de fórmula predictora de tercer orden de Adams-Bashforth. El error de la ecuación (9.4.4) se atribuye a la ecuación (9.4.2), el

cual se evalúa al integrar ésta en $[0, h]$:

$$0(h^4) = \frac{3}{8}h^4 y^{(iv)}(\xi), \quad t_{n-2} \leqslant \xi \leqslant t_{n+1}$$

En la deducción de la ecuación (9.4.4), hay que observar que se usó la ecuación (9.4.1) como extrapolación. Como señalamos en la sección 2.9, la extrapolación es menos precisa que la interpolación (véanse la sección 2.9 y el apéndice A). Por lo tanto, sólo utilizamos la ecuación (9.4.4) como un predictor y la escribimos como

$$\bar{y}_{n+1} = y_n + \frac{h}{12}(23y'_n - 16y'_{n-1} + 5y'_{n-2}) + 0(h^4) \tag{9.4.5}$$

donde la barra superior indica un predictor.

Para obtener una fórmula correctora, se necesita un valor predicho de y'_{n+1}, denotado por $\bar{y}'_{n+1}$, el cual se calcula sustituyendo $\bar{y}_{n+1}$ en $y'(t) = f(y, t)$:

$$\bar{y}'_{n+1} = f(\bar{y}_{n+1}, t_{n+1})$$

El polinomio cuadrático ajustado a $\bar{y}'_{n+1}$, y'_n y y'_{n-1} se escribe como

$$y'(z) = \frac{1}{2h^2}[z(z+h)\bar{y}'_{n+1} - 2(z-h)(z+h)y'_n + z(z-h)y'_{n-1}] + E(z) \tag{9.4.6}$$

donde z es la coordenada local definida después de la ecuación (9.4.1). El error de esta ecuación es

$$E(z) = \frac{1}{3!}(z-h)z(z+h)y^{(iv)}(\xi), \quad t_{n-1} < \xi < t_{n+1}$$

Sustituimos la ecuación (9.4.6) en la ecuación (9.4.3) para obtener la fórmula correctora

$$y_{n+1} = y_n + \frac{h}{12}(5\bar{y}'_{n+1} + 8y'_n - y'_{n-1}) + 0(h^4) \tag{9.4.7}$$

El error es

$$0(h^4) = -\frac{1}{24}h^4 y^{(iv)}(\xi), \quad t_{n-1} < \xi < t_{n+1}$$

La ecuación (9.4.7) es la *fórmula correctora de Adams-Moulton de tercer orden*. El conjunto de ecuaciones (9.4.5) y (9.4.7) se llama *método predictor-corrector de Adams de tercer orden*.

Como hemos visto en el análisis anterior, podemos obtener muchas fórmulas al cambiar la elección de los polinomios de extrapolación e interpolación.

En el análisis de los métodos predictor-corrector, hemos supuesto que se dispone de las soluciones para los puntos anteriores. Como ya se explicó, el método predictor-corrector de tercer orden necesita tres valores previos de y. Por lo tanto, para comenzar con el método, se necesitan las soluciones para $n = 0$, $n = 1$ y $n = 2$; la primera está dada por una condición inicial, pero las otras deben obtenerse por otros medios, como por ejemplo un método de Runge-Kutta.

Ejemplo 9.11

Repita el problema del ejemplo 9.8, con el método predictor-corrector de Adams de tercer orden y $h = 0.1, 0.01$.

(Solución)

Puesto que los métodos predictor-corrector no pueden autoinicializarse, utilizamos el método de Runge-Kutta de cuarto orden para obtener la solución en los primeros dos intervalos. Para este cálculo, modificamos el PROGRAMA 9-3 e incorporamos el método predictor-corrector, por lo que calculamos y_1 y y_2 mediante el método de Runge-Kutta; en tanto que el resto lo calculamos mediante el método predictor-corrector. El programa utilizado es el PROGRAMA 9-4. Los resultados computacionales son:

	$h = 0.1$		$h = 0.01$	
t	y	(e.p.)[a]	y	(e.p.)[a]
1	1.41091	(−0.01)	1.41069	(0.0000)
2	8.84404	(−0.05)	8.83943	(−0.0001)
3	112.644	(−0.12)	112.514	(−0.0004)
4	3740.07	(−0.11)	3736.00	(0.0004)
5	335593	(0.22)	336344.	(0.0009)

[a] e.p.: error porcentual.

9.4.2 Método predictor-corrector de Adams de cuarto orden

Podemos escribir el polinomio de interpolación de Newton hacia atrás [véase la ecuación (2.4.14)] de y' en los puntos n, $n - 1$, $n - 2, \ldots, n - m$, de la manera siguiente:

$$g_m(t) = \sum_{k=0}^{m} (-1)^k \binom{s + k - 1}{k} \Delta^k y'_{n-k} \tag{9.4.8}$$

donde

$$s = \frac{t - t_n}{h}$$

Sustituimos la ecuación (9.4.8) en la ecuación (9.3.2) para obtener así la fórmula predictora de Adams-Bashfort de orden $m + 1$:

$$\bar{y}_{n+1} = y_n + h[b_0 y'_n + b_1 \Delta y'_{n-1} + \cdots + b_m \Delta^m y'_{n-m}] \tag{9.4.9}$$

donde

$$b_k = \int_0^1 \binom{s+k-1}{k} ds \tag{9.4.10}$$

Los primeros b_k son

$$b_0 = 1$$

$$b_1 = \frac{1}{2}$$

$$b_2 = \frac{5}{12}$$

$$b_3 = \frac{3}{8}$$

$$b_4 = \frac{251}{720}$$

Por ejemplo, si hacemos $m = 2$ en la ecuación (9.4.9), obtenemos el predictor de tercer orden dado por (9.4.4). Si seguimos el mismo procedimiento para el caso $m = 3$, obtenemos la fórmula predictora de cuarto orden:

$$\bar{y}_{n+1} = y_n + \frac{9h}{24}(55y'_n - 59y'_{n-1} + 37y'_{n-2} - 9y'_{n-3}) + 0(h^5) \tag{9.4.11}$$

donde

$$0(h^5) = \frac{251}{720} h^5 y^{(v)}(\xi), \quad t_{n-3} < \xi < t_{n+1}$$

Podemos obtener las fórmulas correctoras mediante el polinomio ajustado a y' en los puntos de la retícula, $n + 1, n, n - 1, \ldots, n - m + 1$. La fórmula de interpolación de Newton hacia atrás es, en este caso (véase la sección 2.5):

$$g(t) \simeq \sum_{k=0}^{m} \binom{s+k-2}{k} \Delta^k y'_{n+1-k} \tag{9.4.12}$$

Al sustituir esta ecuación en (9.3.2) resulta la fórmula predictora de Adams-Moulton:

$$y_{n+1} = y_n + h[c_0 y'_{n+1} + c_1 \Delta y'_n + \cdots + c_m \Delta^m y'_{n-m}] \tag{9.4.13}$$

donde

$$c_k = \int_0^1 \binom{s+k-2}{k} ds$$

Los primeros valores c_k son:

$$c_0 = 1$$

$$c_1 = -\frac{1}{2}$$

$$c_2 = -\frac{1}{12}$$

$$c_3 = -\frac{1}{24}$$

$$c_4 = -\frac{19}{720}$$

Si hacemos $m = 3$ en (9.4.13), obtenemos la fórmula correctora de Adams-Moulton de cuarto orden:

$$y_{n+1} = y_n + \frac{h}{24}(9\bar{y}'_{n+1} + 19y'_n - 5y'_{n-1} + y'_{n-2}) + 0(h^5) \tag{9.4.14}$$

donde $y'_n = f(y_n, t_n)$ y

$$0(h^5) = -\frac{19}{720}h^5 y^{(v)}(\xi), \quad t_{n-2} \leqslant \xi \leqslant t_{n+1}$$

El conjunto de ecuaciones (9.4.11) y (9.4.14) recibe el nombre de *método predictor corrector de cuarto orden de Adams*.

9.4.3 Ventajas y desventajas de los métodos predictor-corrector

Una de sus ventajas es la eficiencia computacional: utilizan la información de pasos anteriores. De hecho, la función $f(y, t)$ se evalúa sólo dos veces en cada paso, independientemente del orden del método predictor-corrector; en tanto que el método de Runge-Kutta de cuarto orden hace la evaluación de $f(y, t)$ cuatro veces en cada intervalo. Otra ventaja consiste en que se puede detectar el error local de cada paso mediante un pequeño esfuerzo computacional extra. En la subsección 9.4.4 analizamos la técnica para detectar dicho error. Por otro lado, tiene algunas desventajas:

a) No se puede inicializar por sí mismo, debido a que utiliza puntos anteriores. Se puede utilizar otro método, como el de Runge-Kutta, hasta que se conozcan las soluciones de un número suficiente de puntos.

b) Debido al uso de puntos anteriores, no es fácil cambiar el tamaño del intervalo a la mitad del proceso de solución. Aunque se pueden obtener fórmulas predicto-correctoras para el caso de puntos con espaciamiento no uniforme, los coeficientes de las fórmulas cambian en cada intervalo, por lo que la programación se torna complicada.
c) Este método no se puede utilizar si y' es discontinua. Este caso puede ocurrir cuando uno de los coeficientes de la ecuación diferencial cambia de manera discontinua a mitad del intervalo.

Sin embargo, es posible resolver las últimas dos dificultades de la siguiente forma: puesto que el programa predictor-corrector debe contener un método que se inicialice por sí mismo —tal como el de Runge-Kutta— se pueden volver a retomar los cálculos cuando sea necesario cambiar el tamaño del intervalo o si y' se torna discontinua.

9.4.4 Análisis del error local y la inestabilidad de los métodos predictor-corrector

Una de las ventajas del método predictor-corrector es la facilidad para evaluar con facilidad el error local, si se registra la diferencia entre el predictor y el corrector en cada paso. Como ejemplo de dicho análisis, consideremos el método predictor-corrector de Adams de tercer orden. Las ecuaciones (9.4.4) y (9.4.7) indican que, en el caso en que $y_n, y_{n-1}, y_{n-2}, \ldots$ sean exactas, los valores del predictor-corrector son

$$\bar{y}_{n+1} = y_{n+1\ \text{exacto}} - \frac{3}{8}h^4 y^{(\text{iv})}(\xi) \tag{9.4.15}$$

$$y_{n+1} = y_{n+1,\ \text{exacto}} + \frac{1}{24}h^4 y^{(\text{iv})}(\xi) \tag{9.4.16}$$

Si suponemos además que los valores de la cuarta derivada en las ecuaciones (9.4.15) y (9.4.16) son idénticos, al restar la ecuación (9.4.16) de (9.4.15) obtenemos

$$\bar{y}_{n+1} - y_{n+1} = -\frac{10}{24}h^4 y^{(\text{iv})}(\xi) \tag{9.4.17}$$

Volvemos a sustituir la ecuación (9.4.17) en la ecuación (9.4.16), con lo que resulta

$$y_{n+1,\ \text{exacto}} - y_{n+1} = \frac{1}{10}(\bar{y}_{n+1} - y_{n+1}) \tag{9.4.18}$$

El lado derecho de la ecuación (9.4.18) es el error local del corrector. El cálculo es sencillo debido a que dicho error queda expresado en términos de la diferencia entre el predictor y el corrector. Si se utiliza este algoritmo en cada intervalo, es posible hacer un seguimiento automático en un programa del error local del método.

Analicemos ahora la estabilidad de un método predictor-corrector, considerando nuevamente el método predictor-corrector de Adams de tercer orden dado por las ecuaciones (9.4.5) y (9.4.7). Supongamos que aplicamos el método a la ecuación de prueba, dada por

$$y' = \alpha y \tag{9.4.19}$$

Sustituimos la ecuación (9.4.19) en las ecuaciones (9.4.4) y (9.4.7), con lo que obtenemos

$$\bar{y}_{n+1} = y_n + \frac{\alpha h}{12}(23y_n - 16y_{n-1} + 5y_{n-2})$$

$$y_{n+1} = y_n + \frac{\alpha h}{12}(5\bar{y}_{n+1} + 8y_n - y_{n-1})$$

Eliminamos y_{n+1} en las ecuaciones anteriores y reagrupamos los términos, de lo que resulta

$$y_{n+1} = -a_2 y_n - a_1 y_{n-1} - a_0 y_{n-2} \tag{9.4.20}$$

donde

$$\begin{aligned} a_2 &= -(1 + 13b + 115b^2) \\ a_1 &= b + 80b^2 \\ a_0 &= -25b^2 \\ b &= \frac{\alpha h}{12} \end{aligned}$$

Podemos pensar en la ecuación (9.4.20) como un problema de una ecuación de diferencias con condición inicial, cuya solución analítica se puede obtener de manera análoga a la de una ecuación diferencial ordinaria lineal de tercer orden. De hecho, la solución analítica de la ecuación (9.4.20) tiene la forma

$$y_n = c\gamma^n \tag{9.4.21}$$

donde γ es un valor característico y c es una constante. Sustituimos la ecuación (9.4.21) en la ecuación (9.4.20) para obtener la ecuación característica:

$$\gamma^3 + a_2\gamma^2 + a_1\gamma + a_0 = 0 \tag{9.4.22}$$

La ecuación (9.4.22) es una ecuación polinomial de tercer orden, por lo que tiene tres raíces, de las cuales dos pueden ser complejas. Denotamos las tres raíces por

$$\gamma_1, \quad \gamma_2, \quad \text{y} \quad \gamma_3$$

Puesto que cada γ_1, γ_2 y γ_3 satisface la ecuación (9.4.20), cualquier combinación lineal de estas soluciones también es solución de (9.4.20). La solución general de esta

ecuación se puede escribir entonces como

$$y_n = c_1(\gamma_1)^n + c_2(\gamma_2)^n + c_3(\gamma_3)^n \tag{9.4.23}$$

donde c_1, c_2 y c_3 quedan determinados al dar los valores iniciales de y_0, y_1 y y_2 (conviene recordar que el método predictor-corrector de tercer orden necesita tres valores de inicialización).

La solución exacta del problema original, ecuación (9.4.19), está dada por

$$y_n = y(0) \exp(\alpha nh) \tag{9.4.24}$$

donde $y(0)$ es la condición inicial de $y(t)$. ¿Cómo se relaciona cada uno de los términos de la ecuación (9.4.23) con la ecuación (9.4.24)? La respuesta es que un término de la ecuación (9.4.23) es una aproximación de la ecuación (9.4.24), pero los otros dos son irrelevantes para la solución exacta y forman parte del error del esquema. Supongamos que el primer término es la aproximación, mientras que los otros dos forman el error. La inestabilidad del método se relaciona entonces con los dos últimos términos. Si éstos se anulan cuando n crece, no existe la inestabilidad. Si la magnitud de estos términos se vuelve mayor que la unidad, surge un comportamiento errático de la solución numérica. Esta es la inestabilidad y aparece cuando

$$|\gamma_2| > 1 \quad \text{o} \quad |\gamma_3| > 1 \quad \text{o ambos}$$

Al aplicar el método predictor-corrector a la ecuación (9.4.19), tanto α como h afectan la inestabilidad. Sin embargo, ya que ambas variables aparecen siempre como un producto [véase la ecuación (9.4.20)], podemos considerar a αh como un único parámetro. En la tabla 9.4 se muestran las raíces de la ecuación (9.4.22) para distintos valores de αh.

Tabla 9.4 Valores característicos del método predictor-corrector de tercer orden, aplicado a $y'(t) = \alpha y(t)$

αh	$\exp(\alpha h)$	γ_1	Porcentaje del error	γ_2, γ_3	$\lvert\gamma_2\rvert, \lvert\gamma_3\rvert$
0.1	1.1051	1.1051	0	$0.006 \pm 0.039j$	0.040
0.2	1.2214	1.2214	0	$0.014 \pm 0.074j$	0.075
0.5	1.6487	1.6477	0.06	$0.047 \pm 0.155j$	0.162
1.0	2.7183	2.6668	1.90	$0.108 \pm 0.231j$	0.255
1.5	4.4816	4.1105	8.3	$0.155 \pm 0.266j$	0.308
2.0	7.3891	5.9811	23.	$0.190 \pm 0.283j$	0.341
2.5	12.1825	8.2705	32.	$0.215 \pm 0.292j$	0.362
−0.1	0.9048	0.9048	0	$-0.003 \pm 0.043j$	0.043
−0.2	0.8187	0.8189	0.02	$-0.002 \pm 0.092j$	0.092
−0.3	0.7408	0.7416	0.1	$-0.003 \pm 0.145j$	0.145
−0.4	0.6703	0.6732	0.43	$-0.011 \pm 0.203j$	0.203
−0.5	0.6065	0.6147	1.35	$0.022 \pm 0.265j$	0.266
−1.0	0.3678	0.4824	31.2	$0.116 \pm 0.588j$	0.600
−1.5	0.2231	0.4944	121.	$0.338 \pm 0.821j$	0.889
−2.0	0.1353	0.5650	419.	$0.731 \pm 0.833j$	1.109

$j = \sqrt{-1}$

En la tabla 9.4, γ_1 es la raíz relevante para la solución exacta; es decir, es una aproximación para exp (αh). Las otras dos γ son raíces irrelevantes. La última columna muestra la magnitud de la segunda y tercera raíces. Se ve que cuando $\alpha h > 0$ (es decir, $\alpha > 0$), las magnitudes de γ_2 y γ_3 siempre son menores que γ_1. Por lo tanto, si n crece, la magnitud del segundo y tercer términos disminuye con respecto al primero. Así, no hay inestabilidad en el caso $\alpha > 0$.

En la segunda parte de la tabla 9.4, donde $\alpha < 0$, la raíz relevante γ_1 siempre es menor que uno y decrece cuando αh es más negativa. Si la magnitud de αh es muy pequeña, las raíces irrelevantes son menores que la raíz relevante, pero la magnitud de aquéllas sigue aumentando y excede a la magnitud de ésta antes de que αh alcance el valor —1. La magnitud de la raíz irrelevante es mayor que la unidad aproximadamente cuando $\alpha h = -1.8$. Cuando esto ocurre, el segundo y tercer términos de la ecuación (9.4.20) muestran un comportamiento errático. Es decir, mientras que el primer término tiende a cero, los otros divergen de manera oscilatoria. Por lo tanto, existe una inestabilidad del método predictor-corrector de segundo orden en el caso $\alpha h < -1.8$.

En la tabla 9.4 también aparece una información importante acerca de la precisión del método predictor-corrector de Adams de tercer orden. Como se analizó anteriormente, la primera raíz γ_1 de la tabla 9.4 tiende a exp (αh). La discrepancia entre γ_1 y exp (αh) mide directamente el error local. La tabla muestra que, cuando $\alpha > 0$, el porcentaje del error es pequeño hasta que αh alcanza el valor 0.5. Si $\alpha < 0$, el porcentaje de error aumenta rápidamente al crecer αh ; para $\alpha h = -0.5$, que todavía está lejos del dominio de inestabilidad, es significativo el porcentaje del error.

Resumen de Esta Sección

a) Un método predictor-corrector consta de un predictor y un corrector.
b) Los predictores de los métodos predictor-corrector de Adams reciben el nombre de predictores de Adams-Bashforth. Se obtienen al integrar una extrapolación polinomial de y' para los puntos anteriores.
c) Los correctores de los métodos predictor-corrector de Adams reciben el nombre de correctores de Adams-Moulton y se obtienen al integrar una interpolación polinomial de y' para los puntos anteriores más $\bar{y}'$ (el valor predicho para el nuevo punto).
d) El método predictor-corrector de segundo orden es idéntico al método de Runge-Kutta de segundo orden.
e) Los métodos predictor-corrector de tercero y cuarto orden no pueden inicializarse por sí mismos. Sin embargo, una vez inicializados, su eficiencia computacional es mayor que la del método de Runge-Kutta. La verificación del error en cada intervalo es más fácil que en el caso del método de Runge-Kutta.

9.5 MAS APLICACIONES

Mostraremos en esta sección cinco aplicaciones de los métodos numéricos para problemas con condiciones iniciales. Aunque en toda la sección utilizamos el méto-

do de Runge-Kutta de cuarto orden, éste puede remplazarse por cualquiera de los demás métodos analizados en este capítulo.

Ejemplo 9.12

Una pieza metálica con una masa de 0.1 kg y 200° C (o 473° K) se coloca en cierto momento dentro de un cuarto con una temperatura de 25° C, en donde está sujeta al enfriamiento por convección natural y la transferencia de calor por radiación. Bajo la hipótesis de que la distribución de temperatura es uniforme en el metal, la ecuación de la temperatura se puede escribir como

$$\frac{dT}{dt} = \frac{A}{\rho c v}[\varepsilon\sigma(297^4 - T^4) + h_c(297 - T)], \quad T(0) = 473 \qquad \text{(A)}$$

donde T es la temperatura en grados Kelvin y las constantes son

$\rho = 300$ kg/m^3	(densidad del metal)
$v = 0.001$ m^3	(volumen del metal)
$A = 0.25$ m^2	(área de la superficie del metal)
$c = 900$ J/kgK	(calor específico del metal)
$h_c = 30$ J/m^{2}K	(coeficiente de transferencia de calor)
$\varepsilon = 0.8$	(emisividad del metal)
$\sigma = 5.67 \times 10^{-8}$ w/m^{2}K^4	(constante de Stefan-Boltzmann)

(Solución)

Podemos resolver este problema modificando el PROGRAMA 9-2, el cual utiliza el método de Runge-Kutta de cuarto orden. A continuación se muestran las temperaturas calculadas mediante este método para varios valores de t y $h = 1$.

t (seg)	T (° K)
0	473
10	418.0
20	381.7
30	356.9
60	318.8
120	300.0
180	297.4

Ejemplo 9.13

La corriente eléctrica del circuito que aparece en la figura E9.13a satisface la ecuación integro-diferencial

$$L\frac{di}{dt} + Ri + \frac{1}{C}\int_0^t i(t')\,dt' + \frac{1}{C}q(0) = E(t), \quad t > 0 \qquad \text{(A)}$$

donde el circuito se cierra en el instante $t = 0$; $i = i(t)$ es la corriente (amp); R es una resistencia (ohm); L, C y E están dadas por

$$L = 200 \text{ henry}$$
$$C = 0.001 \text{ faradio}$$
$$E(t) = 1 \text{ voltio para } t > 0$$

las condiciones iniciales son $q(0) = 0$ (carga inicial del capacitor) e $i(0) = 0$. Calcular la corriente para $0 \leqslant t \leqslant 5$ seg después de cerrar el circuito ($t = 0$), con los siguientes valores de R:

a) $R = 0$ ohm
b) $R = 50$ ohm
c) $R = 100$ ohm
d) $R = 300$ ohm

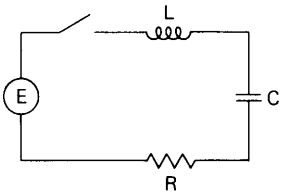


Figura E9.13a Circuito eléctrico

(Solución)

En primer lugar, definimos

$$q(t) = \int_0^t i(t')\,dt' \tag{B}$$

Derivamos (B) para obtener

$$\frac{d}{dt}q(t) = i(t), \quad q(0) = 0 \tag{C}$$

Sustituimos la ecuación (B) en la ecuación (A) y reescribimos

$$\frac{d}{dt}i(t) = -\frac{R}{L}i(t) - \frac{1}{LC}q(t) + \frac{1}{LC}q(0) + \frac{E(t)}{L}, \quad i(0) = 0 \tag{D}$$

Así, transformamos la ecuación (A) en un conjunto de dos EDO de primer orden, las ecuaciones (C) y (D). Modificamos en dos sentidos el PROGRAMA 9-4 para este problema (véase la nota más adelante). En la figura E9.13b se muestra de manera gráfica el resultado del cálculo.

Nota: a) para llevar a cabo los cálculos de los cuatro casos en una sola ejecución, incorporamos cuatro parejas acopladas de EDO de primer orden, siendo la primera pareja correspondiente al primer caso, la segunda pareja al segundo caso, etc. Esto es posible debido a que no todas las ecuaciones del programa deben acoplarse matemáticamente. b) Añadimos una rutina de graficación, por lo que los cuatro casos tienen una salida gráfica.

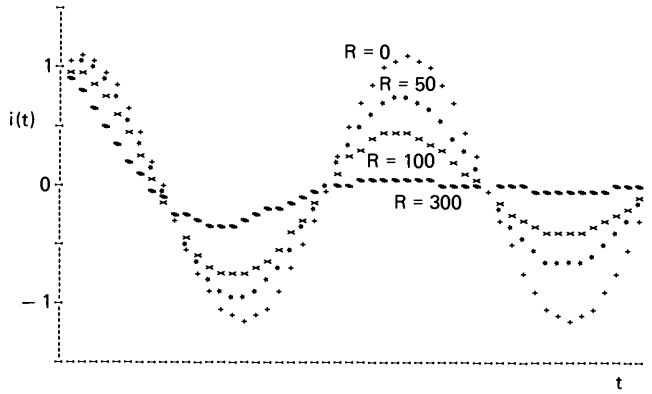


Figura E9.13b Gráfica de los resultados obtenidos

Ejemplo 9.14

En la figura de abajo se muestra un sistema de tres masas. Los desplazamientos de estas tres masas satisfacen las ecuaciones dadas por

$$\begin{aligned} M_1y_1'' + B_1y_1' + K_1y_1 - B_1y_2' - K_2y_2 &= F_1(t) \\ -B_1y_1' - K_1y_1 + M_2y_2'' + B_1y_2' + (K_1 + K_2)y_2 - K_2y_3 &= 0 \\ -K_2y_2 + M_3y_3'' + B_2y'_3 + (K_2 + K_3)y_3 &= F_3(t) \end{aligned} \qquad \text{(A)}$$

Las constantes y condiciones iniciales son

$K_1 = K_2 = K_3 = 1$	(constantes de los resortes, kgm/s^2)
$M_1 = M_2 = M_3 = 1$	(masa, kg)
$F_1(t) = 1,\ F_3(t) = 0$	(fuerza, Newton)
$B_1 = B_2 = 0.1$	(coeficientes de amortiguamiento, kg/s)
$y_1(0) = y_1'(0) = y_2(0) = y_2'(0) = y_3(0) = y_3'(0) = 0$	(condiciones iniciales)

Resuelva las ecuaciones anteriores mediante el método de Runge-Kutta de cuarto orden, para $0 \leqslant t \leqslant 30$ seg y $h = 0.1$

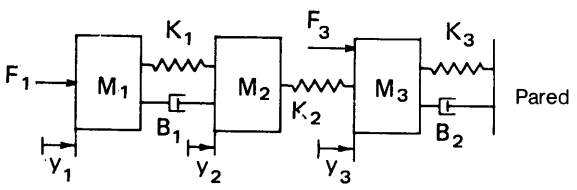


Figura E9.14a Sistema de masas-resortes

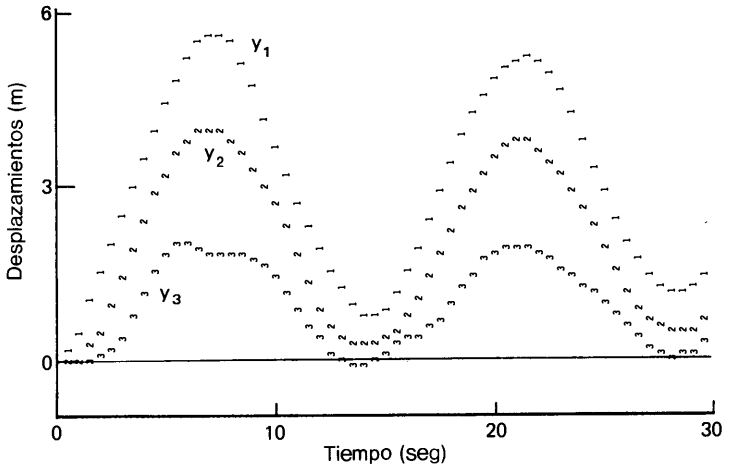


Figura E9.14b Resultado de los cálculos

(Solución)

Definimos

$$y_4 = y_1', \quad y_5 = y_2', \quad \text{y} \quad y_6 = y_3' \tag{B}$$

La ecuación (A) se escribe como un conjunto de seis EDO de primer orden, de la manera siguiente:

$$y_1' = y_4 \tag{C1}$$

$$y_2' = y_5 \tag{C2}$$

$$y_3' = y_6 \tag{C3}$$

$$y_4' = \frac{1}{M_1}[-B_1y_4 - K_1y_1 + B_1y_5 + K_2y_2 + F_1] \tag{C4}$$

$$y_5' = \frac{1}{M_2}[B_1y_4 + K_1y_1 - B_1y_5 - (K_1 + K_2)y_2 + K_2y_3] \tag{C5}$$

$$y_6' = \frac{1}{M_3}[K_2y_2 - B_2y_6 - (K_2 + K_3)y_3 + F_3] \tag{C6}$$

Resolvemos estas ecuaciones modificando el PROGRAMA 9-3. En la figura E9.14b se muestran los resultados computacionales.

Ejemplo 9.15

Una varilla de 1.0 m de longitud, colocada en un vacío, se calienta mediante una corriente eléctrica aplicada a la misma. La temperatura en los extremos se fija en 273° K. El calor se disipa de la superficie mediante la transferencia de

calor por radiación hacia el ambiente, cuya temperatura es 273°K. Con las siguientes constantes, determinar la distribución de temperatura en la dirección del eje.

$k = 60$ W/mK	(conductividad térmica)
$Q = 50$ W/m	(tasa de generación de calor por unidad de longitud de la barra)
$\sigma = 5.67 \times 10^{-8}$ W/m^{2}K^4	(constante de Stefan-Boltzmann)
$A = 0.0001$ m^2	(área de la sección transversal)
$P = 0.01$ m	(perímetro de la varilla)

(Solución)

La ecuación de conducción del calor en la dirección del eje x es

$$-Ak\frac{d^2}{dx^2}T + P\sigma(T^4 - 273^4) = Q \quad 0 < x < 1.0 \tag{A}$$

con las condiciones en la frontera dadas por

$$T(0) = T(1.0) = 273 \text{ K}$$

donde T es la temperatura en grados Kelvin.

Este problema es un problema con condiciones en la frontera (especificadas en $x = 0$ y $x = 1$), pero se puede resolver como un problema de condición inicial sobre la base de prueba y error. Definimos y_1 y y_2 como

$$y_1(x) = T(x)$$
$$y_2(x) = T'(x)$$

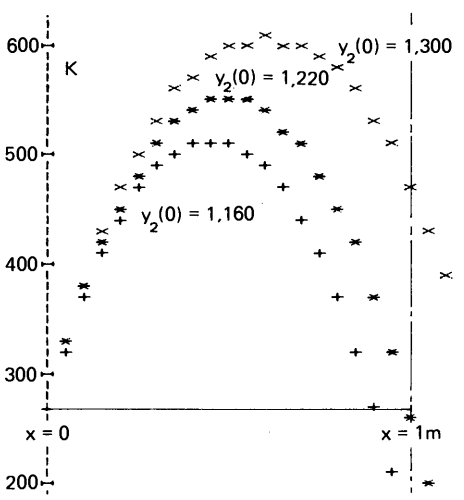


Figura E9.15 Resultados obtenidos mediante el método de disparo

La ecuación (A) se puede reescribir como un conjunto de dos EDO de primer orden como

$$y_1' = y_2 \qquad \text{(B)}$$

$$y_2' = \frac{P}{Ak}\sigma(y^4 - 273^4) - \frac{Q}{kA}$$

Sólo se obtiene una condición inicial, $y_1(0) = 273$, a partir de las condiciones en la frontera ($y_2(0)$ no se conoce). Por ello, resolvemos la ecuación (A) con valores de prueba para $y_2(0)$, hasta satisfacer la condición en la frontera para el extremo derecho, $y_1(1) = 273$. Este enfoque se llama *método de disparo* [Rieder/Busby].

Para este ejemplo, utilizamos el PROGRAMA 9-3 con ciertas modificaciones. Los resultados se grafican directamente en una impresora y se muestran en la figura E9.15. Se puede ver que $y_2(0) = 1160$ es demasiado pequeño como estimación inicial, mientras que $y_2(0) = 1300$ es muy grande. Algún valor de $y_2(0)$ entre estos dos dará el resultado óptimo. Después de unas cuantas pruebas, determinamos que $y_2(0) = 1220$ satisface de manera casi exacta la condición correcta en la frontera.

Ejemplo 9.16

La temperatura de una barra de hierro de 55 cm de longitud perfectamente aislada es inicialmente de 200° C. En cierto instante, se reduce la temperatura del extremo izquierdo y en $t = 0$ seg es 0° C. Calcule la distribución de la temperatura cada 100 seg hasta alcanzar los 1000 seg. Las constantes son

$k = 80.2$ W/mK	(conductividad térmica)
$\rho = 7870$ kg/m^3	(densidad)
$c = 447$ kJ/kg° K	(unidad de calor específico)

(Solución)

En primer lugar, dividimos la varilla en once volúmenes de control, según se muestra en la figura E9.16a. Si denotamos la temperatura promedio del i-ésimo volumen de control mediante $T_1(t)$, la ecuación del balance de calor para el i-ésimo volumen de control es

$$\rho c \Delta x A(dT_i/dt) = (q_{i-1} - q_i)A \qquad \text{(A)}$$

En la ecuación (A), q_1 es el flujo de calor (tasa de conducción de la transferencia del calor por unidad de área de la sección transversal) en la frontera de los volúmenes de control i e $i + 1$, dados por

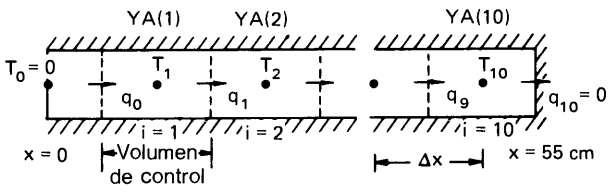


Figura E9.16a Una barra aislada

PROPORCIONE EL INTERVALO DE IMPRESION O GRAFICACION (I.I.)
100
PROPORCIONE EL NUMERO DE PASOS EN CADA I.I. DE X
5
PROPORCIONE EL MAXIMO VALOR DE X PARA DETENER EL CALCULO
1005

H= 20	T_0	T_1	T_2	.....							T_{10}
t= 100 (seg)	0	109	170	192	198	200	200	200	200	200	200
200	0	81	141	175	191	197	199	200	200	200	200
300	0	67	122	160	182	193	197	199	200	200	200
400	0	58	108	146	172	186	194	198	199	200	200
500	0	52	99	136	163	180	190	195	198	199	200
600	0	48	91	127	155	173	186	193	196	198	199
700	0	44	85	120	147	167	181	190	195	197	198
800	0	41	80	114	141	162	176	186	192	196	197
900	0	39	76	108	135	156	172	183	190	194	196
1000	0	37	72	104	130	152	168	179	187	192	194

Figura E9.16b Resultado de los cálculos

de

$$q_i = -\frac{k}{\Delta x}(T_{i+1} - T_i) \text{ para } i = 0, 1, 2, \ldots, 9 \tag{B}$$

y

$$q_{10} = 0 \tag{C}$$

Sustituimos (B) en la ecuación (A) y reagrupamos:

$$\frac{dT_i}{dt} = \frac{k}{\rho c \Delta x^2}(T_{i-1} - 2T_i + T_{i+1}) \tag{D-1}$$

para $i = 1, 2, 3\ldots, 9$ y

$$\frac{dT_{10}}{dt} = \frac{k}{\rho c \Delta x^2}(T_9 - T_{10}) \tag{D-2}$$

Podemos considerar la ecuación (D) como un conjunto de EDO de primer orden y resolverla mediante uno de los métodos de Runge-Kutta. El conjunto de ecuaciones se resolvió utilizando el PROGRAMA 9-3 con ciertas modificaciones. En la figura E9.16b se muestran los resultados obtenidos.

Notas:

a) Podemos ver la ecuación (D) como una aproximación mediante semidiferencias de la ecuación de conducción de calor (ecuación diferencial parcial parabólica)

$$\frac{\partial}{\partial x} k \frac{\partial}{\partial x} T(x, t) = \rho c \frac{\partial T(x, t)}{\partial t}$$

con condición inicial $T(x, 0) = 200°$ C y condiciones en la frontera $T(0, t) = T(55, t) = 0$.

b) Esta técnica de solución para una ecuación diferencial parcial que emplea un método numérico para las EDO, recibe el nombre de *método de líneas*.

c) Se puede hacer un cambio menor para implantar la conductividad térmica dependiente del espacio y el tiempo; es decir, recalcular k para cada frontera de los volúmenes de control en cada intervalo de tiempo.

d) El estudio del autor indica que los cálculos con $h = 50$ seg concuerdan con los que se obtienen utilizando $h = 1$ seg, pero el esquema de solución se vuelve inestable con $h = 100$ seg.

9.6 EDO RIGIDAS

9.6.1 Por qué las ecuaciones rígidas son difíciles

La rigidez se refiere a una pequeña constante del tiempo en una EDO. Por ejemplo, sea

$$y' = -\alpha y + s(t), \quad y(0) = y_0 \tag{9.6.1}$$

donde $\alpha > 0$. Si $s = 0$, la solución de esta ecuación es

$$y(t) = y_0 e^{-\alpha t} \tag{9.6.2a}$$

y si $s(t) \neq 0$, entonces

$$y(t) = y_0 e^{-\alpha t} + e^{-\alpha t} \int_0^t s(\xi) e^{\alpha \xi}\, d\xi \tag{9.6.2b}$$

La respuesta del sistema a la condición inicial, así como a los cambios de $s(t)$ queda caracterizada por $1/|\alpha|$, que recibe el nombre de *constante del tiempo*.

Es difícil, y a veces imposible, resolver un problema rígido mediante un método estándar de Runge-Kutta o predictor-corrector. Por ejemplo, si utilizamos el método de Runge-Kutta de cuarto orden para las ecuaciones (9.6.2a y b), el cálculo es inestable a menos que $h < 2.785/|\alpha|$ (véase la subsección 9.3.5). Al hacerse más pequeña la constante del tiempo, se debe utilizar un intervalo de tiempo cada vez más pequeño. Por ejemplo, si $\alpha = -100000$ seg^{-1}, h debe ser menor que $2.785/100000 = 0.000002785$ seg para que se conserve la estabilidad. Los métodos predictores-correctores analizados anteriormente están sujetos a restricciones parecidas.

Cuando hay que calcular las transiciones muy rápidas de un sistema, es comprensible la necesidad de pequeños intervalos de tiempo. Por otro lado, si $s(t)$ es una función constante o con variación lenta, la solución cambia muy lentamente, por lo que, de manera natural, desearíamos utilizar intervalos mayores. Sin embargo, los mismos intervalos pequeños de tiempo son necesarios para garantizar la estabilidad de la solución numérica, sin importar qué tan lento sea el cambio real de la solución.

La rigidez es particularmente crítica para un conjunto de EDO [Gear (1971); Gear (1979); Hall/Watt; Fertziger; Kuo]. Si el conjunto de ecuaciones contiene úni-

camente una ecuación rígida, la estabilidad del método numérico está determinada por la menor constante del tiempo de la ecuación más rígida.* Por ejemplo, en el caso de dos ecuaciones,

$$\begin{aligned} y' &= -y + z + 3 \\ z &= -10^7 z + y \end{aligned} \tag{9.6.3}$$

donde la segunda ecuación tiene una constante del tiempo mucho menor que la de la primera.

Se han propuesto varios métodos numéricos que permitan mayores intervalos de tiempo; entre éstos están el método implícito de Runge-Kutta y el método racional de Runge-Kutta. Examinamos estos dos métodos en el resto de esta sección.

9.6.2 Métodos implícitos

Para simplificar el análisis, consideremos un conjunto de dos EDO:

$$\begin{aligned} \frac{d}{dt} y &= f(y, z, t) \\ \frac{d}{dt} z &= g(y, z, t) \end{aligned} \tag{9.6.4}$$

Utilizamos la aproximación por diferencias hacia atrás en el lado izquierdo, con lo que podemos escribir un esquema implícito como

$$\begin{aligned} y_{n+1} - y_n &= hf(y_{n+1}, z_{n+1}, t_{n+1}) \equiv hf_{n+1} \\ z_{n+1} - z_n &= hg(y_{n+1}, z_{n+1}, t_{n+1}) \equiv hg_{n+1} \end{aligned} \tag{9.6.5}$$

donde los términos f y g del lado derecho tienen incógnitas, y_{n+1} y z_{n+1}.

Si f y g son funciones no lineales, no podemos resolver la ecuación (9.6.5) en forma exacta. Sin embargo, la solución iterativa explicada en la subsección 9.2.3 se puede aplicar con facilidad [Hall/Watt]. Exhortamos al lector a que intente esta vía. Desgraciadamente, no es eficiente en términos computacionales para un sistema grande EDO. Un punto de vista más eficiente es linealizar las ecuaciones mediante desarrollos de Taylor [Kubicek; Constantinides]. El desarrollo de Taylor de $f_{k,n+1}$ en torno de t_n es

$$\begin{aligned} f_{n+1} &= f_n + f_y \Delta y + f_z \Delta z + f_t h \\ g_{n+1} &= g_n + g_y \Delta y + g_z \Delta z + g_t h \end{aligned} \tag{9.6.6}$$

* En términos más estrictos, la constante del tiempo es un valor propio del sistema, por lo que no es un valor asociado con ninguna de las ecuaciones individuales del conjunto. Sin embargo, si una de las ecuaciones es mucho más rígida que las demás, esta ecuación determina la mínima constante del tiempo y tiene poca influencia de las demás.

donde

$$\Delta y = y_{n+1} - y_n, \quad \Delta z = z_{n+1} - z_n \tag{9.6.7}$$

Sustituimos la ecuación (9.6.6) en la ecuación (9.6.5); utilizamos además la ecuación (9.6.7) para obtener

$$\begin{bmatrix} 1 - hf_y & -hf_z \\ -hg_y & 1 - hg_z \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} hf_n + h^2 f_t \\ hg_n + h^2 g_t \end{bmatrix} \tag{9.6.8a}$$

o, en forma más compacta,

$$(I - hJ)\Delta\bar{y} = VLD \tag{9.6.8b}$$

donde

VLD = vector del lado derecho de la ecuación (9.6.8a)
J = la matriz jacobiana definida por

$$J = \begin{bmatrix} f_y & f_z \\ g_y & g_z \end{bmatrix}$$

I = matriz identidad
$\Delta\bar{y}$ = col $(\Delta y, \Delta z)$

Resolvemos la ecuación (9.6.8) mediante la eliminación de Gauss. El método implícito es incondicionalmente estable, a menos que los efectos no lineales provoquen la inestabilidad.

El método se ha extendido a un conjunto grande de EDO simultáneas. Los métodos de Gear [Gear, 1971], disponibles en la biblioteca NAG [NAG], utilizan aproximaciones por diferencias hacia atrás de orden superior con un espaciamiento variable en la retícula.

9.6.3 Método exponencial

Varios investigadores han propuesto y utilizado la transformación exponencial y el ajuste exponencial para resolver EDO rígidas. En esta subsección daremos sólo una breve introducción de las ideas básicas de los métodos exponenciales.

Para explicar el principio en el que se basan, consideremos una sola EDO de primer orden:

$$y' = f(y, t) \tag{9.6.9}$$

donde, para hacer el análisis más sencillo, suponemos que f no incluye a t en forma explícita.

Al sumar cy a ambos lados de la ecuación (9.6.9), obtenemos

$$y' + cy = f(y, t) + cy \tag{9.6.10}$$

donde c es una constante. Si usamos e^{-ct} como factor integrante, la ecuación (9.6.10) se integra en el intervalo $[t_n, t_{n+1}]$ de la manera siguiente:

$$y(t_{n+1}) = y_n e^{-ch} + \int_0^h [f(y(t_n + \xi), t_n + \xi) + cy(t_n + \xi)]e^{c(\xi - h)}\, d\xi \tag{9.6.11}$$

donde $t_{n+1} = t_n + h$. La ecuación (9.6.11) es exacta, sin importar la elección de c.

Se pueden obtener varios esquemas numéricos distintos si introducimos una aproximación para $f + cy$ en el integrando. Sin embargo, la precisión de una integración aproximada se ve afectada entonces por el valor de c. Para determinar un valor apropiado de c, escribimos y como

$$y(t) = y_n + \delta y(t) \tag{9.6.12}$$

Sustituimos la ecuación (9.6.12) en la ecuación (9.6.9), de lo que resulta

$$\begin{aligned} \delta y' &= f(y_n + \delta y) \\ &= f_n + (f_y)_n \delta y + 0(\delta y^2) \end{aligned} \tag{9.6.13}$$

Ignoramos el término del error de segundo orden, con lo que podemos escribir la ecuación (9.6.13) en la siguiente forma equivalente:

$$y' - (f_y)_n y = f_n - (f_y)_n y_n \tag{9.6.14}$$

que es una aproximación linealizada de la ecuación (9.6.9) en torno de $t = t_n$. Si hacemos c igual a

$$c = -(f_y)_n \tag{9.6.15}$$

en la ecuación (9.6.10), entonces ésta queda idéntica a la de la ecuación (9.6.14). Utilizamos

$$y'' = f' = f_y y' = f_y f \tag{9.6.16}$$

Podemos expresar la ecuación (9.6.15) también como

$$c = -(f'/f)_n \tag{9.6.17}$$

Obtenemos un esquema numérico explícito aproximando los términos de los paréntesis cuadrados de la ecuación (9.6.11) por

$$[f(y, t_n + \xi) + cy(t_n + \xi)] \simeq f_n + cy_n \tag{9.6.18}$$

Puesto que el lado derecho de la ecuación (9.6.18) es constante, la ecuación (9.6.11) se reduce a

$$\begin{aligned} y_{n+1} &= y_n e^{ch} + (1/c)(1 - e^{-ch})[f_n + cy_n] \\ &= y_n + hf_n\left[\frac{1 - e^{-ch}}{ch}\right] \end{aligned} \tag{9.6.19}$$

que se conoce como el *método ajustado en forma exponencial* [Bui; Oran; Hetric; Fergason/Hansen]. Este método no sólo es incondicionalmente estable, sino que también garantiza que la solución es positiva si se espera que la solución sea positiva.

Los errores de la ecuación (9.6.19) provienen de la aproximación de la ecuación (9.6.18). En el resto de esta subsección desarrollamos un método más preciso mediante un procedimiento iterativo. Con base en la ecuación (9.6.19), podemos plantear el siguiente predictor de $y(t)$ para $t_n < t < t_{n+1}$

$$\bar{y}(t) = y_n + \left[\frac{1 - e^{-c\xi}}{c}\right]f_n, \quad \xi = t - t_n \tag{9.6.20}$$

y para t_{n+1},

$$\bar{y}_{n+1} = y_n + \left[\frac{1 - e^{-ch}}{c}\right]f_n$$

Sustituimos la ecuación (9.6.20) en (9.6.11), para obtener

$$y_{n+1} = \bar{y}_n + \int_{\xi=0}^{h} [f(\bar{y}(t_n + \xi), t_n + \xi) - f_n + cy(t_n + \xi) - cy_n]e^{c(\xi - h)}\, d\xi \tag{9.6.21}$$

El segundo término de la ecuación (9.6.21) es una corrección de la ecuación (9.6.20) y se puede evaluar de las siguientes formas:

a) Mediante la integración analítica, cuando sea posible.
b) Con la aproximación de los términos en los paréntesis cuadrados mediante una interpolación lineal.
c) Al usar la integración con la regla del trapecio.

El enfoque a) no es fácil, a menos que f sea una función sencilla, por lo que no lo tomaremos en cuenta. Para examinar b), la interpolación lineal de la parte entre paréntesis se escribe como

$$[f(\bar{y}(t_n + \xi)) - f_n + cy(t_n + \xi) - cy_n] \cong B\xi \tag{9.6.22}$$

donde

$$B = \frac{f_{n+1} - f_n + c(y_{n+1} - y_n)}{h}$$

Al sustituir la ecuación (9.6.22) en la ecuación (9.6.21), el corrector es

$$y_{n+1} = \bar{y}_{n+1} + \frac{Bh^2}{ch}\left(\frac{1 - e^{-ch}}{ch - 1}\right) \tag{9.6.23}$$

Si utilizamos la regla del trapecio, el corrector queda

$$y_{n+1} = \bar{y}_{n+1} + \frac{Bh^2}{2} \tag{9.6.24}$$

lo que coincide con la ecuación (9.6.23) en el límite de $ch \to 0$.

El segundo término de (9.6.23) o (9.6.24) es una corrección de la ecuación (9.6.19). Para calcularlo, primero evaluamos la ecuación (9.6.19) y después el segundo término.

La extensión del método exponencial a un conjunto de ecuaciones no lineales es directa, con un procedimiento esencialmente idéntico al de una única ecuación. Una vez obtenidos los predictores de todas las variables, se evalúa entonces el segundo término de la ecuación (9.6.23) o (9.6.24).

Resumen de Esta Sección

a) Una EDO es rígida si la constante del tiempo es pequeña y $f'/f < 0$ (si existe el término no homogéneo, la solución tiende a cero). Si se utiliza un método numérico estándar, como los de Runge-Kutta o predictores-correctores, se necesita un pequeño intervalo de tiempo aun cuando la solución vaya cambiando en forma lenta.

b) Para atenuar la dificultad de las EDO rígidas, se presentan dos métodos, el implícito y el exponencial.

PROGRAMAS

PROGRAMA 9-1 Método de Runge-Kutta de segundo orden

A) Explicaciones

Este programa calcula la solución de la EDO de segundo orden que aparece en el ejemplo 9.6 mediante el método de Runge-Kutta de segundo orden.

Las constantes dadas en el ejemplo 9.6, se definen en las instrucciones DATA. Las condiciones iniciales están dadas en YB y ZB. Los resultados obtenidos se imprimen después de cada 50 pasos.

B) Variables

Y, Z: y y z para un nuevo paso
YB, ZB: y y z del paso anterior
BM, KM: a y b en la ecuación (9.3.10)

C) Listado

```
C     CSL/F9-1.FOR      ESQUEMA DE RUNGE-KUTTA DE SEGUNDO ORDEN
C                       (SOLUCION DEL PROBLEMA DEL EJEMPLO 9.6 INCISO b)
      REAL*8 M,K,K1,K2,L1,L2,KM
      PRINT *,'CSL/F9-1    ESQUEMA DE RUNGE-KUTTA DE SEGUNDO ORDEN
      DATA T,    K,     M,    B,     Z,    Y,    H
     &     /0.0,100.0, 0.5, 10.0, 0.0, 1.0, 0.001/
      PRINT *,'      T              Y                Z'
      PRINT 1,T,Y,Z
1     FORMAT( F10.5,   1P2E13.6)
      KM=K/M
      BM=B/M
      DO N=1,20
        DO KOUNT=1,50
           T=T+H
           K1=H*Z
           L1=-H*(BM*Z + KM*Y)
           K2=H*(Z+L1)
           L2=-H*(BM*(Z+L1) + KM*(Y+K1))
           Y=Y+(K1+K2)/2
           Z=Z+(L1+L2)/2
        END DO
          PRINT 1,T,Y,Z
      END DO
      END
```

D) Ejemplo de salida

```
CSL/F9-1    ESQUEMA DE RUNGE-KUTTA DE SEGUNDO ORDEN
     T              Y              Z
  0.00000 1.000000E+00 0.000000E+00
  0.05000 8.230488E-01-5.815448E+00
  0.10000 5.083122E-01-6.190855E+00
  0.15000 2.383530E-01-4.451182E+00
  0.20000 6.674805E-02-2.461108E+00
  0.25000-1.662533E-02-9.825372E-01
  0.30000-4.225293E-02-1.406029E-01
  0.35000-3.886459E-02 2.117637E-01
  0.40000-2.582995E-02 2.771569E-01
  0.45000-1.320036E-02 2.171473E-01
  0.50000-4.550497E-03 1.292080E-01
  0.55000 1.172962E-05 5.766744E-02
  0.60000 1.686464E-03 1.385871E-02
  0.65000 1.791016E-03-6.460617E-03
  0.69999 1.286237E-03-1.197583E-02
  0.74999 7.104116E-04-1.037225E-02
  0.79999 2.831070E-04-6.636304E-03
  0.84999 4.004550E-05-3.249089E-03
  0.89999-6.151515E-05-1.017550E-03
  0.94999-8.021756E-05 1.119957E-04
```

PROGRAMA 9-2 Esquema de Runge-Kutta de cuarto orden

A) Explicaciones

Es un programa de Runge-Kutta de cuarto orden para resolver una ecuación diferencial de primer orden. Antes de ejecutar el programa, el usuario debe definir la ecuación diferencial ordinaria a resolver, en el subprograma FUN. Cuando se ejecuta el programa, se le pregunta al usuario el número de pasos, I, en el intervalo de impresión de t, denotado por TD. Entonces, el intervalo del tiempo se hace igual a h = TD/I. También se pregunta al usuario el máximo t en el que debe evaluarse la solución.

La condición inicial para y en $t = 0$ se plantea en el programa. Los valores de k_j, $j = 1, 2, 3$ y 4 se calculan mediante la subrutina. El cálculo se detiene cuando se excede XL (el valor máximo de t especificado en la entrada).

B) Variables

H: intervalo de tiempo, h
F: $f(y, t)$ de la ecuación (9.3.1)
K1, K2, K3 y K4: k_1, k_2, k_3 y k_4, respectivamente
Y: y
YA: y en el subprograma que define a la ecuación diferencial
X: t
XA: t de la ecuación diferencial en el subprograma
XL: valor máximo de t
TD: intervalo de impresión de t (la solución se imprime después de cada incremento de t por TD).

C) Listado

```
C-----CSL/F9-2.FOR      ESQUEMA DE RUNGE-KUTTA DE CUARTO ORDEN
C                       (VEASE EL EJEMPLO 9.8)
      REAL K1,K2,K3,K4
      PRINT *
      PRINT*,'CSL/F9-2   ESQUEMA DE RUNGE-KUTTA DE CUARTO ORDEN'
1     PRINT *
      PRINT *, ' ¿INTERVALO DE IMPRESION DE T? '
         READ *, XPR
      PRINT *, '¿NUMERO DE PASOS EN UN INTERVALO DE IMPRESION? '
         READ *, I
      PRINT *,' ¿T MAXIMO?'
         READ *, XL
      Y=0                        ! Aquí se fija el valor inicial de la solución.
      H=XPR/I                    ! H es el intervalo del tiempo
      PRINT *, 'H=', H
      XP=0                       ! Se inicializa el tiempo
      HH=H/2
      PRINT *
```

```
      PRINT *,'-------------------------------------'
      PRINT *,'        T                   Y'
      PRINT *,'-------------------------------------'
      PRINT 82, XP,Y
82    FORMAT( 1X,F10.6, 7X,1PE15.6)
30    DO J=1,I      ! Avanza I pasos en cada intervalo de impresión
         XB=XP
         XP=XP+H
         YN=Y
         XM=XB+HH
         K1=H*FUN(XB,YN)
         K2=H*FUN(YN+K1/2,XM)
         K3=H*FUN(YN+K2/2,XM)
         K4=H*FUN(YN+K3,XP)
         Y=YN + (K1+K2*2+K3*2+K4)/6
      END DO
      PRINT 82, XP,Y
      IF (XP.LE.XL) GO TO 30
      PRINT *
      PRINT *,' SE HA EXCEDIDO EL LIMITE DE X '
      PRINT *
200   PRINT*
      PRINT*,' OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
      READ *,K
      IF(K.EQ.1) GOTO 1
      PRINT*
      END
C*****************************
      FUNCTION FUN(X,Y)
      FUN = X*Y+1
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F9-2   ESQUEMA DE RUNGE-KUTTA DE CUARTO ORDEN

¿INTERVALO DE IMPRESION DE T?
1
NUMERO DE PASOS EN UN INTERVALO DE IMPRESION
10
¿T MAXIMO?
11
H=  0.1000000

-------------------------------------
       T                  Y
-------------------------------------
  0.000000          0.000000E+00
  1.000000          1.410686E+00
  2.000000          8.839370E+00
  2.999999          1.125059E+02
  3.999998          3.734233E+03
  4.999998          3.357973E+05
  5.999997          8.194363E+07
  6.999996          5.419219E+10
  7.999995          9.693210E+13
  8.999998          4.676607E+17
 10.000002          6.064899E+21
 11.000006          2.105243E+26
```

PROGRAMA 9-3 Método de Runge-Kutta de cuarto orden para un sistema de EDO

A) Explicaciones

Este programa se diseñó para resolver un conjunto de cualquier número de ecuaciones diferenciales ordinarias de primer orden, aunque los datos muestra son los del ejemplo 9.5.

En el subprograma FUNCT se define el conjunto de ecuaciones diferenciales ordinarias de primer orden a resolver. En el programa principal se definen el número de ecuaciones IM, así como los IM valores de las condiciones iniciales. Para correr el programa con un nuevo problema, el usuario debe cambiar las ecuaciones en FUNCT, el valor de IM y las condiciones iniciales. La estructura del programa es esencialmente idéntica a la del PROGRAMA 9-2, pero se calcula cada paso intermedio en un ciclo DO para el número IM de ecuaciones.

B) Variables

Y(1): y

Y(2): z

Y(I): I-ésima incógnita

YN(I): y_n para $I = 1$ y z_n para $I = 2$, etc.

YA(I): $y_n + k_1/2$ o $y_n + k_2/2$ o $y_n + k_3$ para $I = 1$;
$z_n + l_1/2$ o $z_n + l_2/2$ o $z_n + l_3$ para $I = 2$;

K(J, 1), $J = 1, 2, 3, 4$: k_1, k_2, k_3, k_4

K(J, 2), $J = 1, 2, 3, 4$: l_1, l_2, l_3, l_4

K(J, M), $J = 1, 2, 3, 4$: similar a lo anterior para la M-ésima ecuación diferencial

IM: número de ecuaciones en el conjunto

NS: número de intervalos de tiempo en un intervalo de impresión, TD

XP: límite máximo de t

TD: intervalo de impresión para t

C) Listado

```
C-----CSL/F9-3.FOR      ESQUEMA DE RUNGE-KUTTA DE CUARTO ORDEN
C                       PARA UN CONJUNTO DE ECUACIONES
C                       (VEASE EL EJEMPLO 9.9)
      DIMENSION YA(0:10), YN(0:10), EK(0:4,0:10),Y(0:10)
      PRINT *
      PRINT *,'CSL/F9-3          ESQUEMA DE RUNGE-KUTTA DE CUARTO ORDEN '
      PRINT *,'                  PARA UN CONJUNTO DE ECUACIONES '
      IM=2          ! Número de ecuaciones
      Y(1) = 1      ! Condición inicial para y - 1 en t=0
      Y(2) = 0      ! Condición inicial para y - 2 en y = 0.
1     PRINT *
      PRINT *,' ¿INTERVALO DE IMPRESION DE T?
      READ *,PI
      PRINT *,' ¿NUMERO DE PASOS EN UN INTERVALO DE IMPRESION DE T?'
      READ *,NS
```

```
      PRINT *,' ¿T MAXIMO PARA DETENER LOS CALCULOS? '
      READ *,XL
      H= PI/NS
      PRINT *,' H= ',H
      XP=0
      HH=H/2
      PRINT *
      LI = 0                    ! Inicialización del número de línea
      PRINT*,' LINEA   T                Y(1),                Y(2), .....'
      WRITE (*,98) LI,XP, (Y(I),I=1,IM)
28    LI=LI+1
      DO N=1,NS
         XB=XP                  ! Tiempo anterior
         XP=XP+H                ! Tiempo nuevo
         XM=XB+HH               ! Tiempo en el punto medio
         J=1                    ! Esta parte calcula k~1.
         DO I=1,IM
            YA(I)=Y(I)
         END DO
         XA=XB
         CALL  FUNCT(EK,J,YA,H)
         J=2                    ! Esta parte calcula k~2.
         DO I=1,IM
            YA(I)=Y(I)+EK(1,I)/2
         END DO
         XA=XM
         CALL  FUNCT(EK,J,YA,H)
         J=3                    ! Esta parte calcula k~3.
         DO I=1,IM
            YA(I)=Y(I)+EK(2,I)/2
         END DO
         XA=XM
           CALL  FUNCT(EK,J,YA,H)
         J=4                    ! Esta parte calcula k~4.
         DO I=1,IM
            YA(I)=Y(I)+EK(3,I)
         END DO
         XA=XP
         CALL  FUNCT(EK,J,YA,H)
           DO I=1,IM            ! Esquema de Runge-Kutta de 4o. orden
             Y(I)=Y(I)+(EK(1,I)+EK(2,I)*2+EK(3,I)*2+EK(4,I))/6
           END DO

      END DO
      WRITE (*,98) LI,XP, (Y(I),I=1,IM)
98    FORMAT(1X, I2, F10.6, 2X, 1P4E16.6/(15X,1P4E16.6))
      IF (XP .LT. XL) GOTO 28
200   PRINT*
      PRINT*,' OPRIMA 1 PARA CONTINUAR O 0 PARA TERMINAR '
      READ *,K
      IF(K.EQ.1) GOTO 1
      PRINT*
      END
C*****************************************
      SUBROUTINE FUNCT(EK,J,YA,H)          ! DEFINE UN CONJUNTO DE ECUACIONES
      DIMENSION EK(0:4,0:10),YA(0:10)
      EK(J,1)=YA(2)*H
      EK(J,2)=-YA(1)*H
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F9-3          ESQUEMA DE RUNGE-KUTTA DE CUARTO ORDEN
                  PARA UN CONJUNTO DE ECUACIONES

¿INTERVALO DE IMPRESION DE T?
0.5
¿NUMERO DE PASOS DE UN INTERVALO DE IMPRESION DE T?
2
¿T MAXIMO PARA DETENER LOS CALCULOS?
5.1
 H=    0.2500000

LINEA   T            Y(1),            Y(2), .....
 0   0.000000     1.000000E+00     0.000000E+00
 1   0.500000     8.775873E-01    -4.794100E-01
 2   1.000000     5.403255E-01    -8.414482E-01
 3   1.500000     7.078408E-02    -9.974816E-01
 4   2.000000    -4.160835E-01    -9.093117E-01
 5   2.500000    -8.010826E-01    -5.985258E-01
 6   3.000000    -9.899591E-01    -1.412116E-01
 7   3.500000    -9.364737E-01     3.506708E-01
 8   4.000000    -6.537223E-01     7.566990E-01
 9   4.500000    -2.109293E-01     9.774704E-01
10   5.000000     2.835002E-01     9.589372E-01
11   5.500000     7.085202E-01     7.056382E-01
```

PROGRAMA 9-4 Método predictor-corrector de tercer orden

A) Explicaciones

Este programa resuelve una ecuación diferencial ordinaria de primer orden mediante el método predictor-corrector de tercer orden. Se utiliza el método de Runge-Kutta de cuarto orden para la inicialización.

En el subprograma FUNC se define la ecuación que se resolverá. La estructura del programa es muy parecida a la del PROGRAMA 9-2, excepto por lo siguiente: en los primeros dos intervalos, se utiliza el método de Runge-Kutta de cuarto orden; a partir del tercer intervalo, se ignora el esquema de Runge-Kutta pero se utiliza el esquema predictor-corrector de tercer orden.

B) Variables

Y: función desconocida y

XA, YA: t y y, respectivamente

X: t

I: número de pasos en un intervalo de impresión

N: índice para contar los pasos

H: intervalo de tiempo (H = 1/I)

F: f de $y' = f(y, t)$

XL: límite máximo de t

TD: intervalo de tiempo para la impresión

C) Listado

```
C-----CSL/B9-4.FOR     METODO PREDICTOR-CORRECTOR DE TERCER ORDEN
      REAL*8 K1,K2,K3,K4
      PRINT *,'CSL/B9-4     METODO PREDICTOR-CORRECTOR DE TERCER ORDEN '
      PRINT *, '¿INTERVALO DE TIEMPO PARA IMPRIMIR LA SOLUCION? '
      READ *, TD
      PRINT *, '¿NUMERO DE PASOS EN UN INTERVALO DE IMPRESION?'
      READ *, I
      PRINT *,' ¿LIMITE DE T? '
      READ *, TMAX
      H=TD/I                          !   Intervalo de tiempo
      HH=H/2
      PRINT *, ' Tamaño del paso =', h
      Y=0             !   Condición inicial de la solución
      YB=0
      TN=0
      FC=0
      FB=0
      G=H/12
      PRINT *
      PRINT *,       '  Solución  '
      PRINT *,       '        t                    y'
      CALL FUNC(FA,0.0, Y)
      N=0
30    DO J=1,I                  !   En este ciclo se avanzan I pasos del tiempo
         N=N+1                     !   Conteo de los pasos del tiempo
         FD=FC
         FC=FB
         FB=FA
         TB=TN
         TN=TB+H
         TM=TB+HH
      IF (N.LE.2) THEN          !   Runge-Kutta de 4o. orden
          CALL FUNC(F,TB,Y)
         K1=H*F
          CALL FUNC(F,TM,Y+K1/2)
         K2=H*F
          CALL FUNC(F,TM,Y+K2/2)
         K3=H*F
          CALL FUNC(F,TN,Y+K3)
         K4=H*F
         Y=Y + (K1 + K2*2 + K3*2 + K4)/6
      ELSE                  !   Método predictor-corrector de tercer orden
         YP=Y+G*(23*FB-16*FC+5*FD)                          ! predictor
            CALL FUNC(FP,TN,YP)
         Y =Y+G*(5*FP+8*FB-FC)                              ! corrector
      END IF
            CALL FUNC(FA,TN,Y)
      END DO
      PRINT *, TN, Y
      IF (TN.GT.TMAX) STOP
      GO TO 30
      END
```

```
C*************************************
      SUBROUTINE FUNC(F,T,Y)        ! Define la ecuación diferencial
110   F=T*Y + 1
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/B9-4     METODO PREDICTOR-CORRECTOR DE TERCER ORDEN
¿INTERVALO DE TIEMPO PARA IMPRIMIR LA SOLUCION?
1
¿NUMERO DE PASOS EN UN INTERVALO DE IMPRESION?
10
¿LIMITE DE T?
5
Tamaño del paso = 0.1000000

 Solución
        t                y
  1.000000         1.410910
  2.000000         8.844146
  2.999999         112.6442
  3.999998         3740.071
  4.999998         335593.3
```

PROBLEMAS

9.1) Resuelva los siguientes problemas en $0 \leqslant t \leqslant 5$ mediante el método de Euler hacia adelante y $h = 0.5$, haciendo las operaciones a mano. Repita lo anterior con $h = 0.01$ y una computadora (escriba usted mismo un programa breve). Evalúe los errores por comparación con los valores exactos que se dan a continuación:

a) $y' + ty = 1, \qquad y(0) = 1$

b) $y' + 3y = e^{-t}, \qquad y(0) = 1$

c) $y' = (t^2 - y), \qquad y(0) = 0.5$

d) $y' + y|y| = 0, \qquad y(0) = 1$

e) $y' + |y|^{1/2} = \text{sen}\,(t), \quad y(0) = 1$

Solución exacta

	Caso *a*	*b*[a]	*c*	*d*	*e*
t	*y*	*y*	*y*	*y*	*y*
0	1.0000	1.0000	0.5000	1.0000	1.0000
1	1.3313	0.2088	0.4482	0.5000	0.6147
2	0.7753	0.06890	1.7969	0.3333	0.7458
3	0.4043	2.4955*E*–2	4.9253	0.2500	0.4993
4	0.2707	9.1610*E*–3	9.9725	0.2000	−0.2714
5	0.2092	3.3692*E*–3	16.980	0.1666	−2.2495

[a] *Sugerencia:* la solución de b) podría oscilar con $h = 0.5$, pero de todas formas se le exhorta a realizarlo.

9.2) Resuelva

$$y''(t) - 0.05y'(t) + 0.15y(t) = 0, \quad y'(0) = 0, \quad y(0) = 1$$

y determine los valores de $y(1)$ y $y(2)$ mediante el método de Euler hacia adelante con $h = 0.5$.

9.3) Resuelva los siguientes problemas en $0 \leqslant t \leqslant 5$ mediante el método de Euler hacia adelante, con $h = 0.1$ y $h = 0.01$ (escriba su propio programa). Evalúe los errores usando las siguientes soluciones exactas:

a) $y'' + 8y = 0, y(0) = 1, y'(0) = 0$

b) $y'' - 0.01(y')^2 + 2y = \operatorname{sen}(t), y(0) = 0, y'(0) = 1$

c) $y'' + 2ty' + ty = 0, y(0) = 1, y'(0) = 0$

d) $(e^t + y)y'' = t, y(0) = 1, y'(0) = 0$

Solución exacta

	Caso *a*	*b*	*c*	*d*
t	*y*	*y*	*y*	*y*
0	1.0	0.0000	1.0000	1.0000
1	−0.9514	0.8450	0.8773	1.0629
2	0.8102	0.9135	0.5372	1.3653
3	−0.5902	0.1412	0.3042	1.8926
4	0.3128	−0.7540	0.1763	2.5589
5	−0.0050	−0.9589	0.1035	3.2978

9.4) Resuelva las ecuaciones siguientes para $0 < t < 5$ con el método modificado de Euler:

$$4y' = -3y + 7z + 2t, \quad y(0) = 1$$
$$7z' = -2y + 8z, \quad z(0) = 0$$

Utilice $h = 0.01$ y $h = 0.001$.

9.5) Un depósito cónico contiene agua hasta 0.5 m de altura a partir del fondo. El depósito tiene un orificio, en el fondo, de 0.02 m de radio. El radio del depósito está dado por $r = 0.25y$, donde r es el radio y y es la altura medida desde el fondo. La velocidad del agua que pasa por el orificio está dada por $v^2 = 2gy$, donde $g = 9.8$ m/seg^2. Por medio del método de Euler hacia adelante ($h = 0.001$ seg), calcule cuántos minutos se tardará en vaciar el depósito.

9.6) En la figura P9.6 se muestra un circuito, el cual tiene una autoinductancia de $l = 100$ henrys, una resistencia de $R = 2$ ohms y una fuente de voltaje de CD de 10 voltios. Si el circuito se cierra en el instante $t = 0$, la corriente $I(t)$ cambia según la fórmula

$$L\frac{d}{dt}I(t) + RI(t) = E, \quad I(0) = 0$$

a) Determine la corriente I en $t = 1, 2, 3, 4$ y 5 seg mediante el método de Euler hacia adelante y $h = 0.01$.

b) Evalúe el error, comparando la solución numérica con la solución analítica, dada por $I(t) = (E/R)(1 - e^{-Rt/L})$.

c) Analice el efecto de h, repitiendo los cálculos anteriores pero con $h = 0.1$.

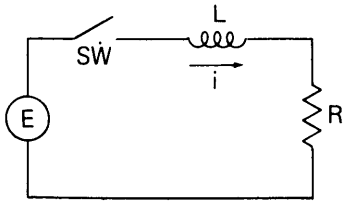


Figura P9.6 Circuito eléctrico

9.7) Un tubo con forma de U y 0.05 m de radio se llena con agua, pero con una división de forma que el nivel del agua en la parte vertical de la izquierda es 0.2 m más alto que el de la parte vertical derecha. En el instante $t = 0$ se retira la división. El nivel del agua de la parte izquierda $y_A A$, medido desde el plano intermedio entre las dos superficies, satisface la ecuación

$$Ly_A'' = -2gy_A$$

donde L es la longitud total del agua en el tubo —que se supone mide 1m— mientras que $g = 9.8$ m/seg^2. Si se desprecia la fricción en el tubo, calcule el nivel del agua por medio del método de Euler hacia adelante para $0 < t < 10$ seg y determine cuándo alcanza y_A su máximo y su mínimo. Utilice $h = 0.001$.

9.8) Repita el problema anterior, suponiendo la existencia de fricción en el tubo de forma que la ecuación de movimiento es

$$Ly_A'' = -2gy_A - \beta y_A'$$

donde $\beta = 0.8$ m/seg. Use $h = 0.001$.

9.9) La densidad numérica (número de átomos por cm^3) del yodo-135 (radioisótopo) satisface la ecuación

$$\frac{d}{dt} N_i(t) = -\lambda_i N_i(t)$$

donde $N(t)$ es la densidad numérica del yodo-135 y λ_i es su constante de decaimiento, igual a 0.1044 hrs^{-1}. Si $N_1(0) = 10^5$ átomos/cm^3 en el instante $t = 0$, calcule $N_i(t)$ en $t = 1$ hora mediante el método modificado de Euler. Haga h igual a 0.05 de hora.

9.10) El producto del decaimiento del yodo-135 (véase el problema anterior) es el xenón-135; también es radiactivo. Su constante de decaimiento es $\lambda_x = 0.0753$ hrs^{-1}. La densidad numérica del xenón satisface la ecuación

$$\frac{d}{dt} N_x(t) = -\lambda_x N_x(t) + \lambda_i N_i(t)$$

donde N_x es la densidad numérica del xenón y N_i es la densidad numérica del yodo, definida en el problema anterior. Suponga que $N_x(0) = 0$ y desarrolle un programa para calcular N_i y N_x con base en el método modificado de Euler. (Puesto que las ecuaciones diferenciales son lineales, utilice las soluciones que más se aproximen a cada intervalo de tiempo.) Imprima la solución para cada 5 horas y hasta alcanzar las 50 horas. Use $h = 0.1$ hora.

9.11) Calcule $y(1)$ resolviendo la ecuación siguiente mediante el método de Runge-Kutta de segundo orden, con $h = 0.5$:

$$y' = -\frac{y}{t + y^2}, \quad y(0) = 1$$

9.12) Calcule $y(2)$ para la ecuación siguiente por medio del método de Runge-Kutta de segundo orden, con $h = 1$:

$$y'' + 0.2y' + 0.003y \operatorname{sen}(t) = 0, \quad y(0) = 0, y'(0) = 1$$

9.13) Determine el valor de $y(1)$ resolviendo

$$y'' - 0.05y' + 0.15y = 0, \quad y(0) = 1, \quad y'(0) = 0$$

Utilice el método de Runge-Kutta de segundo orden, con $h = 0.5$.

9.14) Resuelva la siguiente ecuación diferencial:

$$2y'' + (y')^2 + y = 0, \quad y(0) = 0, y'(0) = 1$$

mediante el método de Runge-Kutta de segundo orden, con $h = 0.5$ y evalúe $y(1)$ y $y'(1)$.

9.15) Un problema de una ecuación diferencial ordinaria con condición inicial está dado por

$$y''' = -y, \quad y(0) = 1$$
$$y'(0) = y''(0) = 0$$

Calcule $y(0.4)$ por medio del método de Runge-Kutta de segundo orden, con $h = 0.2$.

9.16 a) Un tanque de 50 galones de agua contiene sal con una concentración de 10 onzas/galón. Con el fin de diluir el contenido de sal, se suministra agua pura a razón de 2 galones/minuto. Si el depósito tiene una mezcla uniforme y la misma cantidad de agua que entra sale del depósito cada minuto, la concentración de sal satisface

$$y_1'(t) = -\frac{2}{50}y_1, \quad y_1(0) = 10$$

donde $y_1(t)$ es la concentración de sal en onzas/galón y t es el tiempo en minutos. Utilice el método de Runge-Kutta de segundo orden con $h = 1$ minuto para determinar cuánto tiempo debe transcurrir para que la concentración de la sal sea 1/10 de su valor inicial.

b) El agua que sale del tanque entra a otro tanque de 20 galones, en el cual también se vierte agua pura a razón de 3 galones/minuto y se mezcla bien. La concentración de la sal en el segundo tanque satisface

$$y_2'(t) = -\frac{3}{20}y_2(t) + \frac{2}{20}y_1(t), \quad y_2(0) = 0$$

donde $y_1(t)$ es la concentración de sal del tanque de 50 galones del problema anterior. Utilice el método de Runge-Kutta de segundo orden para determinar cuándo alcanza su máximo la concentración de sal en el tanque de 20 galones. Suponga que el segundo tanque tiene agua pura en el instante $t = 0$.

9.17) Repita el problema 9.12 con el método de Runge-Kutta de tercer orden.

9.18) Se dispara un proyectil al aire, con un ángulo de 45° con respecto del suelo a $u = v = 150$ m/seg, donde u y v son las velocidades horizontal y vertical, respectivamente. Las ecuaciones de movimiento están dadas por

$$\begin{aligned} u' &= -cVu, & u(0) &= 150 \text{ m/seg} \\ v' &= -g - cVv, & v(0) &= 150 \text{ m/seg} \end{aligned} \tag{A}$$

donde u y v son funciones del tiempo, $u = u(t)$ y $v = v(t)$, y

$$V = \sqrt{u^2 + v^2}$$
$$c = 0.005 \quad \text{(coeficiente de arrastre)}$$
$$g = 9.8 \text{ m/seg}^2 \text{ (gravedad)}$$

Las ecuaciones de movimiento se pueden resolver mediante alguno de los métodos de Runge-Kutta. La trayectoria del proyectil se puede determinar al integrar

$$x' = u \quad \text{y} \quad y' = v$$

o bien

$$x = \int_0^t u(t')\,dt' \qquad y = \int_0^t v(t')\,dt' \tag{B}$$

El siguiente programa resuelve la ecuación (A) y evalúa la ecuación (B), mediante el método de Runge-Kutta de segundo orden:

```
C   RUNGE-KUTTA DE SEGUNDO ORDEN PARA EL PROBLEMA DEL PROYECTIL

      DATA UB,VB,H,C,T,X,Y/150.0, 150.0, 0.1, 0.005, 0.0, 0.0, 0.0/
      real K1,K2,L1,L2
      PRINT*,
     % ' TIEMPO          U              V              X              Y'
      PRINT 20,T,UB,VB,X,Y
20    FORMAT( 5F12.6)

      DO N=1, 200
         T=T+H
         VEL1=SQRT(UB**2+VB**2)                 ! Velocidad absoluta en t (n)
         K1= -C*VEL1*UB*H
         L1 = (-9.8 - C*VEL1*VB)*H
         VEL2=SQRT((UB+K1)**2+(VB+L1)**2)       ! Velocidad absoluta en t (n+1)
         K2= -C*VEL2*(UB+K1)*H
         L2 = (-9.8 - C*VEL2*(VB+L1))*H
         U=UB + (K1+K2)/2
         V=VB + (L1+L2)/2
         X = X + 0.5*(U+UB)*H
         Y = Y + 0.5*(V+VB)*H
         UB=U
         VB=V
         PRINT 20,T,U,V,X,Y
         IF(Y.LT.0) STOP
      END DO
      END
```

a) Corra el programa y grafique la trayectoria del proyectil.

b) Reescriba el programa con el método de Runge-Kutta de tercer orden.

9.19) Calcule $y(1)$, resolviendo la ecuación siguiente mediante el método de Runge-Kutta de cuarto orden, con $h = 1$:

$$y' = -\frac{y}{t + y^2}, \quad y(0) = 1 \text{ para } t = 0$$

9.20) A continuación se muestra la solución de $y' = -1(1 + y^2)$ mediante el método de Runge-Kutta de segundo orden, con dos valores distintos de h.

t	h = 0.1 y	h = 0.2 y
0.0	1.0000000	1.0000000
0.1	0.9487188	
0.2	0.894672	0.8947514
0.3	0.8375606	
0.4	0.7770516	0.7772616
0.5	0.7127807	
0.6	0.6443626	0.6447898
0.7	0.5714135	
0.8	0.4935937	0.4943817
0.9	0.4106803	
1.0	0.3226759	0.3240404

a) Estime el error local con $h = 0.1$.

b) Estime un valor más preciso de $y(1)$.

9.21) Calcule a mano la solución de

$$y'(t) = -\frac{1}{1 + y^2}, \quad y(0) = 1$$

para $t = 1$ y $t = 2$ por medio del método de Runge-Kutta de cuarto orden, con $h = 0.5$ y $h = 1$.

9.22) Repita el problema 9.1 mediante el método de Runge-Kutta de cuarto orden, con $h = 0.1$.

9.23) Para la ecuación dada por

$$y' = 3y + \exp(1 - t), \quad y(0) = 1$$

calcule el intervalo óptimo de tiempo para el método de segundo orden de Runge-Kutta, de forma que satisfaga la condición para el error local, $E(h) < 0.0001$. (Ejecute el método de Runge-Kutta de segundo orden para un intervalo con un valor de h y vúelvalo a ejecutar para dos intervalos con $h/2$.)

9.24) Repita el problema (9.23) mediante el método de Runge-Kutta de cuarto orden.

9.25) Repita el análisis de las ecuaciones (9.3.23) a (9.3.27) y obtenga la ecuación correspondiente a la ecuación (9.3.23) para el método de Runge-Kutta de tercer orden.

9.26) Si se aplica el método de Runge-Kutta de tercer orden a $y' = -\alpha y$, determine el rango de h en donde el método sea inestable.

9.27) La temperatura inicial de la pieza metálica del ejemplo 9.12 es ahora 25°C. Dicha pieza se calienta internamente de forma eléctrica a razón de $q = 3000$ W. La ecuación de la temperatura es

$$\frac{dT}{dt} = \frac{1}{\rho c v}\left[q - \epsilon\sigma A(T^4 - 298^4) - h_c A(T - 298)\right], \quad T(0) = 298$$

Calcule la temperatura hasta t = minutos, e imprima los resultados para cada 0.5 min mediante el método de Runge-Kutta de cuarto orden, con $h = 0.1$ min. (Utilice las constantes dadas en el ejemplo 9.12.)

9.28) El movimiento del sistema de masas que se muestra en la figura P9.28 está dado por

$$y'' + 2\zeta\omega y' + \omega^2 y = F(t)/M$$

donde

$\omega = (k/M)^{1/2}$ (frecuencia natural sin amortiguamiento, s^{-1})
$\zeta = c/(2M\omega) = 0.5$ (factor de amortiguamiento)
$k = 3.2$ (constante del resorte, kg/s^2)
$M = 5$ (masa, kg)
$F(t) = 0$ (fuerza, Newtons)

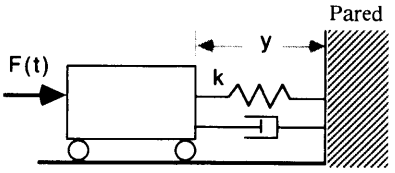


Figura P9.28 Sistema masa-resorte

Si $F(t)$ es una función escalonada de magnitud $F_0 = 1$ kg y cuya duración es 1 seg, determine el movimiento de la masa para $0 < t < 10$ seg por medio del método de Runge-Kutta de cuarto orden.

9.29) Determine la respuesta y carga dinámica del sistema amortiguado del problema anterior sujeto a un pulso de fuerza triangular

$$\begin{aligned} F(t) &= 2F_0 t, & 0 \leqslant t \leqslant 1 \text{ seg} \\ &= 2F_0(1-t), & 1 \leqslant t \leqslant 2 \text{ seg} \\ &= 0, & t > 2 \text{ seg} \end{aligned}$$

donde $F_0 = 1$ Kg (fuerza). Utilice el método de Runge-Kutta de cuarto orden.

9.30) La ecuación diferencial del circuito que aparece en la figura P9.30 es

$$L_1 \frac{d}{dt} i_1 + R_A(i_1 - i_2) + \frac{1}{C}\int_0^t (i_1(t') - i_2(t'))\,dt' = e(t)$$

$$-\frac{1}{C}\int_0^t (i_1(t') - i_2(t'))\,dt' - R_A(i_1 - i_2) + R_B i_2 + L_2 \frac{d}{dt} i_2 = 0$$

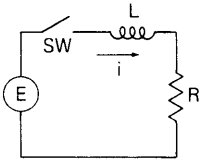


Figura P9.30 Circuito eléctrico

Las condiciones iniciales son

$$i_1(0) = i_2(0) = 0,$$

y $e(t) = 1$. Mediante el método de Runge-Kutta de cuarto orden con $h = 0.1$ seg, determine i_1 e i_2 para $0 < t < 10$ seg.

9.31) El problema del ejemplo 9.14 tiene una simetría geométrica con respecto de $x = 0.5$. Por lo tanto, podemos reenunciar el problema en la siguiente forma equivalente:

$$Ak\frac{d^2}{dx^2}T + P\sigma(T^4 - 273^4) = Q, \quad 0.5 < x < 1$$

$$T'(0.5) = 0$$

$$T(1) = 0$$

Resuelva las ecuaciones anteriores mediante el método del disparo y el método de Runge-Kutta de cuarto orden. (*Sugerencia*: cambie $T(0.5)$ por prueba y error hasta satisfacer $T(1) = 0$.)

9.32) Repita a), b) y c) del problema 9.1 con el método predictor-corrector de Adams de tercer orden con $h = 0.5$.

9.33) Repita a), b) y c) del problema 9.1 con el método predictor-corrector de Adams de cuarto orden con $h = 1.0$.

9.34) Escriba en forma explícita los predictores de Adams-Bashfort de orden 2, 3, 4 y 5.

9.35) Escriba en forma explícita los correctores de Adams-Moulton de orden 2, 3, 4 y 5.

9.36) Resuelva el problema del ejemplo 9.11 mediante el esquema predictor-corrector de Adams de cuarto orden. (*Sugerencia*: modifique el PROGRAMA 9-4 de forma que los primeros tres pasos se calculen mediante el método de Runge-Kutta de cuarto orden y el resto se calcule mediante el método predictor-corrector de cuarto orden.)

9.37) El método predictor-corrector de Euler es

$$\bar{y}_{n+1} = y_n + hy'_n$$

$$y_{n+1} = y_n + \frac{1}{2}h(\bar{y}'_{n+1} + y'_n)$$

Demuestre que si aplicamos este método a la ecuación $y'(t) = \alpha y(t)$ con $\alpha < 0$, la inestabilidad se presenta cuando $\alpha h < -2$.

9.38) El método predictor-corrector de Milne está dado por

$$\bar{y}_{n+1} = y_{n-1} = \frac{4}{3}h(2y'_n - y'_{n-1} + 2y'_{n-2}) \quad \text{predictor}$$

$$y_{n+1} = y_{n-1} + \frac{1}{3}h(\bar{y}'_{n+1} + 4y'_n + y'_{n-1}) \quad \text{corrector}$$

Muestre que este método se torna inestable para $y' = \alpha y$ si $\alpha < 0$ y

$$-\infty < \alpha h < -0.8 \quad \text{o} \quad -0.3 < \alpha h < 0$$

(*Sugerencia*: calcule las raíces de la ecuación característica, como en la sección 9.4.4.)

BIBLIOGRAFIA

Bui, T. D., A. K. Oppenheim y D. T. Pratt, "Recent advances in methods for numerical solution of O. D. E. initial value problems", *J. Comp. Math.* vol. 11, págs. 283-296, 1984.

Constantinides, A., *Applied Numerical Methods with Personal Computers*, McGraw-Hill, 1987.

Creese, T. M. y R. M. Haralick, *Differential Equations for Engineers*, McGraw-Hill, 1978.

Ferziger, J. H., *Numerical Methods for Engineering Application*, Wiley-Interscience, 1981.

Fox, L. y D. F. Mayers, *Computing Methods for Scientists and Engineers*, Oxford, University Press, 1968.

Furgason, D. R. y K. F. Hansen, "Solution of the space dependent reactor kinetics equations in three dimensions", *Nucl. Sci. Eng.*, vol. 51, págs. 189-205, 1973.

Gear, C. W., *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, 1971.

Gear, C. W., "A User's View of Solving Stiff Ordinary Differential Equations", *SIAM Review*, vol. 21, 1979.

Habib, I. S., *Engineering Analysis Methods*, Lexington Books, 1975.

Hall, G. y J. M. Watt, *Modern Numerical Methods for Ordinary Differential Equations*, Clarendon Press, 1976.

Hetric, D. L., *Dynamics of Nuclear Reactors*, University of Chicago Press, 1971.

Kubicek, M. y V. Hlavacek, *Numerical Solution of Nonlinear Boundary Value Problems with Applications*, Prentice-Hall, 1983.

Kuo, K. K., *Principles of Combustion*, Wiley-Interscience, 1986.

Lapidus, L. y J. H. Seinfeld, *Numerical Solution of Ordinary Differential Equations*, Academic Press, 1971.

NAG Fortran Library (1987), Algorithm Group Inc., 1101 31st, Suite 100, Downers Grove, IL 60515-1263.

Oran, E. S. y J. P. Boris, *Numerical Simulation of Reactive Flow*, Elsevier, 1987.

Rieder, W. G. y H. R. Busby, *Introductory Engineering Modeling*, Wiley, 1986.

10
Problemas de ecuaciones diferenciales con valores en la frontera

10.1 INTRODUCCION

En los problemas de ecuaciones diferenciales ordinarias unidimensionales con valores en la frontera, se pide que la solución satisfaga las condiciones de frontera en ambos extremos del dominio unidimensional. La definición de las condiciones en la frontera es parte fundamental de los problemas de este tipo. Por ejemplo, consideremos una varilla delgada de metal con longitud H, tal que sus extremos estén conectados a distintas fuentes de calor. Si el calor sale de la superficie de la varilla únicamente mediante la transferencia de calor por medio de convección, podemos escribir la ecuación de la temperatura como

$$-A \frac{d}{dx} k(x) \frac{d}{dx} T(x) + h_c P T(x) = h_c P T_\infty + AS(x) \tag{10.1.1}$$

donde $T(x)$ es la temperatura del punto que se encuentra a una distancia x del extremo izquierdo, A es el área constante de una sección transversal de la varilla, k es la conductividad térmica, P el perímetro de la varilla, h_c el coeficiente de transferencia de calor por convección, T_∞ es la temperatura neta del aire y S es la fuente de calor. Las condiciones en la frontera son

$$\begin{aligned} T(0) &= T_L \\ T(H) &= T_R \end{aligned} \tag{10.1.2}$$

donde T_I y T_D son las temperaturas del cuerpo dadas en los extremos izquierdo y derecho, respectivamente.

Si $\overline{T}$ se define como

$$\overline{T} = T - T_\infty$$

podemos escribir la ecuación (10.1.1) como

$$-\frac{d}{dx} k(x) \frac{d}{dx} \overline{T}(x) + \frac{h_c P}{A} \overline{T}(x) = S(x) \tag{10.1.3}$$

en la cual hemos dividido ambos miembros entre A. El primer término representa la difusión del calor, el segundo es la pérdida de calor en el aire por medio de la convección y el lado derecho es la fuente de calor.

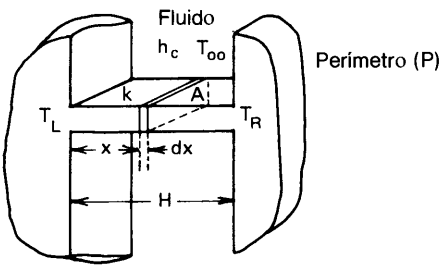


Figura 10.1 Una lámina conectada a dos fuentes de calor

Otro ejemplo de una EDO como forma similar es la ecuación de difusión de neutrones dada por

$$-\frac{d}{dx} D(x) \frac{d}{dx} \psi(x) + \Sigma_a \psi(x) = S(x) \tag{10.1.4}$$

donde ψ es el flujo de neutrones, D es el coeficiente de difusión y S es la fuente de neutrones. El primer término indica la difusión de neutrones, el segundo la pérdida por absorción y el lado derecho es la fuente de neutrones.

En el resto de este capítulo estudiaremos la ecuación

$$-\frac{d}{dx} p(x) \frac{d}{dx} \phi(x) + q(x)\phi(x) = S(x) \tag{10.1.5}$$

o ecuaciones análogas en coordenadas cilíndricas o esféricas. El primer término es el de difusión; el segundo es el de la pérdida y el lado derecho es el término de la fuente, independientemente de la situación física particular en cuestión.

Debemos hacer énfasis en que la ecuación (10.1.5) es una ley de la conservación de la difusión. Al integrar la ecuación (10.1.5) en $[a, b]$, tenemos que

$$Z(b) - Z(a) + \int_a^b q(x)\phi(x)\, dx = \int_a^b S(x)\, dx \tag{10.1.6}$$

donde

$$Z(x) = -p(x)\frac{d}{dx}\phi(x)$$

es el flujo de calor en x si se toma en cuenta la conducción de calor, o bien la corriente de los neutrones en el caso de la difusión de neutrones. En todo caso, los términos primero y segundo de la ecuación (10.1.6) son, respectivamente, el flujo hacia adentro y el flujo hacia afuera de la propiedad representada por ϕ, el tercer término es la pérdida total en $[a, b]$ y el lado derecho es la fuente total en $[a, b]$. Así, (10.1.6) representa la conservación de la propiedad en el intervalo $[a, b]$.

Si la ecuación (10.1.1) fuera un problema con condición inicial, sólo se deben especificar dos condiciones de frontera en una de las fronteras, así la solución numérica podría pasar de ese extremo hacia el otro, utilizando un método numérico (por ejemplo, el método de Runge-Kutta de cuarto orden). Aunque podríamos utilizar los métodos de solución de los problemas con condiciones iniciales para este otro tipo de problemas, tal como lo mostramos en el capítulo 9, ellos funcionan sobre la base de prueba y error (conocido como el método de disparo, véase el ejemplo 9.15). Una ventaja de este método es que se puede utilizar con facilidad un programa ya existente para problemas con condiciones iniciales. Sin embargo, es frecuente que el método falle, ya que puede enfrentarse a la inestabilidad numérica. Además, si el número de condiciones en los extremos es mayor que dos, es difícil aplicarlo [Hall/Watt].

Una vía más general para resolver los problemas con valores en la frontera consta de: a) la obtención de ecuaciones en diferencias y b) la resolución de dichas ecuaciones en forma simultánea.

En este capítulo, estudiaremos en primer término la obtención de aproximaciones por diferencias para problemas con valores en la frontera y su solución simultánea. Después analizaremos la aplicación de los métodos tanto a problemas con valores en la frontera como a problemas de valores propios. El estudio de los métodos numéricos para problemas unidimensionales con valores en la frontera nos ayudará a comprender los métodos de solución para las ecuaciones diferenciales parciales. En la tabla 10.1 aparece un breve resumen de los métodos.

10.2 PROBLEMAS CON VALORES EN LA FRONTERA PARA VARILLAS Y LAMINAS

En esta sección obtendremos ecuaciones con diferencias finitas para las ecuaciones diferenciales ordinarias de segundo orden con valores en la frontera.

Para explicar el principio del método, consideremos la ecuación

$$-\phi''(x) + q\phi(x) = S(x) \qquad (10.2.1)$$

$$0 < x < H$$

con condiciones en la frontera

$$\begin{aligned} \phi'(0) &= 0 \quad \text{(condición en la frontera izquierda)} \\ \phi(H) &= \phi_R \quad \text{(condición en la frontera derecha)} \end{aligned} \qquad (10.2.2)$$

Tabla 10.1 Resumen de los métodos para los problemas unidimensionales con valores en la frontera

Tipo de problemas y método de solución	Ventajas	Desventajas
Problemas no homogéneos		
Método de disparo	Se puede utilizar un programa ya existente para problemas con condiciones iniciales.	Se hace con el método de prueba y error. Se aplica a una clase limitada de problemas. La solución puede ser inestable.
Método de diferencias finitas que utiliza la solución tridiagonal	No hay problemas de inestabilidad. No se utiliza el método de prueba y error para problemas lineales. Se puede aplicar a los problemas no lineales con iteración.	Se debe desarrollar un programa para cada problema particular.
Problemas de valores propios		
Método matricial (véase el capítulo 7)	Se calculan todos los valores propios de una sola vez.	No se puede aplicar si el tamaño de la matriz es grande.
Método iterativo		
Método de la potencia inversa	Sencillez.	Solamente para el valor propio fundamental.
Método de la potencia inversa con desplazamiento	Igual de sencillo que el método de la potencia. Se puede calcular cualquier valor propio.	Debe usarse el método de prueba y error para evaluar el valor propio; sólo para valores propios reales.

donde q es un coeficiente constante. Si dividimos el dominio en N intervalos de igual longitud, obtenemos una retícula como la de la figura 10.2, donde los intervalos miden $h = H/N$.

Aplicamos la aproximación por diferencias centrales [véase el inciso *f*) de la tabla 5.2] al primer término de la ecuación (10.2.1), con lo que obtenemos la ecuación en diferencias para la i-ésima retícula:

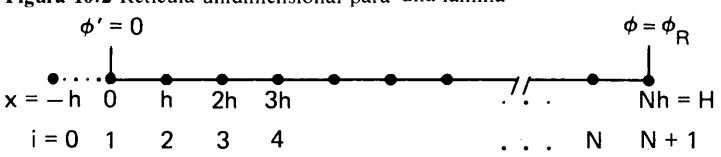
$$\frac{-\phi_{i-1} + 2\phi_i - \phi_{i+1}}{h^2} + q\phi_i = S_i \tag{10.2.3}$$

donde $\phi_i = \phi(x_i)$, $S_i = S(x_i)$ y se supone que q es constante. Multiplicamos la ecuación anterior por h^2:

$$-\phi_{i-1} + (2 + w)\phi_i - \phi_{i+1} = h^2 S_i \tag{10.2.4}$$

donde $W = h^2 q$. Esta ecuación se puede aplicar a todos los puntos de la retícula, excepto cuando $i = 1$ e $i = N + 1$.

Figura 10.2 Retícula unidimensional para una lámina



La condición de la frontera izquierda, dada por la ecuación (10.2.2) es equivalente a una condición simétrica en la frontera llamada *condición adiabática en la frontera* en el caso de la transferencia de calor. Si se considera un punto hipotético de la rejilla $i = 0$ localizado en $x = -h$ la ecuación (10.2.4) en el caso $i = 1$ es

$$-\phi_0 + (2 + w)\phi_1 - \phi_2 = h^2 S_1$$

En esta ecuación, podemos hacer $\phi_0 = \phi_2$ debido a la simetría. Al dividir la ecuación resultante entre 2 se tiene

$$\left(1 + \frac{w}{2}\right)\phi_1 - \phi_2 = \frac{1}{2}h^2 S_1 \tag{10.2.5}$$

Como $\phi_{N+1} = \phi(H) = \phi_R$ en la frontera derecha, la ecuación (10.2.4) con $i = N$ es

$$-\phi_{N-1} + (2 + w)\phi_N = h^2 S_N + \phi_R \tag{10.2.6}$$

donde los términos conocidos se pasaron del lado derecho.

El conjunto de ecuaciones (10.2.4), (10.2.5) y (10.2.6) se escriben en forma conjunta como

$$\begin{aligned}
(1 + w/2)\phi_1 - \phi_2 &= h^2 S_1/2 \\
-\phi_1 + (2 + w)\phi_2 - \phi_3 &= h^2 S_2 \\
-\phi_2 + (2 + w)\phi_3 - \phi_4 &= h^2 S_3 \\
&\vdots \\
-\phi_{N-1} + (2 + w)\phi_N &= h^2 S_N + \phi_R
\end{aligned} \tag{10.2.7a}$$

o en la forma matricial equivalente

$$\begin{bmatrix} 1 + w/2 & -1 & & & & \\ -1 & 2 + w & -1 & & & \\ & -1 & 2 + w & -1 & & \\ & & & \ddots & & \\ & & & & -1 & 2 + w \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_N \end{bmatrix} = \begin{bmatrix} h^2 S_1/2 \\ h^2 S_2 \\ h^2 S_3 \\ \vdots \\ h^2 S_N + \phi_R \end{bmatrix} \tag{10.2.7b}$$

Todos los elementos de la matriz de la ecuación (10.2.7b) son cero, excepto los de las tres diagonales. Esta forma especial recibe el nombre de *matriz tridiagonal* y aparece muy frecuentemente en los métodos numéricos para problemas con valores en la frontera. Llamaremos a la ecuación (10.2.7a) o (10.2.7b) una *ecuación tridiagonal*, la cual se resuelve mediante el método tridiagonal que se analiza en la siguiente sección.

Tabla 10.2 Tres tipos de condiciones en la frontera

Tipo	Explicación	Ejemplos
Condición en la frontera con un valor fijo (tipo de Dirichlet)	Se da un valor de la función solución.	$\phi(0) = 0$, $\phi(0) = 1$
Condición en la frontera para la derivada (tipo de Neumann)	Se da la derivada de la solución.	$\phi'(0) = 0$, $\phi'(0) = 1$
Condición en la frontera de tipo mixto (tipo mixto)	Se relaciona un valor de la función con la derivada.	$\phi'(0) + \alpha\phi(0) = \beta$

Las condiciones en la frontera se clasifican en tres tipos, como se muestra en la tabla 10.2. Analicemos la implantación de una condición mixta. Supongamos que las dos condiciones de frontera de la ecuación (10.2.1) son del tipo mixto, es decir,

$$-\phi'(0) + f_I\phi(0) = g_I \tag{10.2.8}$$

$$\phi'(H) + f_D\phi(H) = g_D \tag{10.2.9}$$

donde f_I, f_D, g_I y g_D son constantes. Consideremos la retícula que se muestra en la figura 10.3 (igual a la de la figura 10.2, excepto que el último punto tiene el número N en vez de $N + 1$).

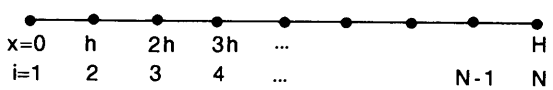


Figura 10.3 Retícula unidimensional

La ecuación en diferencias (10.2.4) no se altera para $i = 2$ hasta $N - 1$, pero debemos revisar las ecuaciones para $i = 1$ y N debido a que las condiciones en la frontera han cambiado. En primer lugar, analicemos la frontera izquierda. Utilizamos la aproximación por diferencias hacia adelante con base en un intervalo de longitud $h/2$ de la ecuación (10.2.1) en $x = 0$ y tenemos que

$$-\frac{\phi'\left(\frac{h}{2}\right) - \phi'(0)}{\frac{h}{2}} + q\phi_1 = S_1$$

Donde podemos sustituir ϕ' $(h/2)$ con la aproximación por diferencias centrales.

$$\phi'\left(\frac{h}{2}\right) = \frac{1}{h}(\phi_2 - \phi_1)$$

y $\phi'(0)$ se elimina mediante la ecuación (10.2.8). Así, obtenemos

$$-\frac{\frac{1}{h}(\phi_2 - \phi_1) + g_L - f_I \phi_1}{\frac{h}{2}} + q\phi_1 = S_1$$

o, en forma equivalente,

$$\left(1 + \frac{w}{2} + hf_I\right)\phi_1 - \phi_2 = \frac{1}{2}h^2 S_1 + hg_I \tag{10.2.10}$$

donde $w = qh^2$, y los términos conocidos están agrupados del lado derecho.

La ecuación en diferencias de la frontera derecha se obtiene mediante un procedimiento similar:

$$-\phi_{N-1} + \left(1 + \frac{w}{2} + hf_D\right)\phi_N = \frac{1}{2}h^2 S_N + hg_D \tag{10.2.11}$$

El conjunto de ecuaciones (10.2.10), (10.2.4) y (10.2.11) forma un conjunto tridiagonal.

Ejemplo 10.1

Determinar las ecuaciones en diferencias para el siguiente problema con valores en la frontera:

$$-2y''(x) + y(x) = \exp(-0.2x) \tag{A}$$

con condiciones en la frontera

$$y(0) = 1$$
$$y'(10) = -y(10)$$

Suponga que los intervalos de la retícula tienen una longitud unitaria.

(Solución)

En la figura E10.1 se muestra la retícula. Las ecuaciones en diferencias para $i = 1$ hasta 9 son las siguientes:

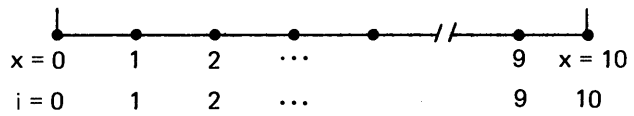


Figura E10.1

$$2(-y_{i-1} + 2y_i - y_{i+1}) + y_i = \exp(-0.2i) \tag{B}$$

donde se usa que $x_i = i$.

Para $i = 1$, sustituimos la condición en la frontera $y_0 = y(0) = 1$ en las ecuaciones anteriores y tenemos que

$$5y_1 - 2y_2 = \exp(-0.2) + 2 \tag{C}$$

Para $i = 10$, primero aproximamos la ecuación (A) por

$$-\frac{2[y'(10) - y'(9.5)]}{\frac{1}{2}} + y(10) = \exp(-2) \qquad \text{(D)}$$

Por medio de la aproximación por diferencias centrales, el término $y'(9.5)$ es

$$y'(9.5) = [y(10) - y(9)]/1 \qquad \text{(E)}$$

Sustituimos la ecuación (E) y la condición en la frontera derecha $y'(10) = -y(10)$ en la ecuación (D), de lo que resulta

$$-2y_9 + 4.5y_{10} = 0.5 \exp(-2) \qquad \text{(F)}$$

En resumen, las ecuaciones en diferencias son

$$\begin{aligned} 5y_1 - 2y_2 &= \exp(-0.2) + 2 \\ -2y_{i-1} + 5y_i - 2y_{i+1} &= \exp(-0.2x_i), \text{ para } i = 2 \text{ a } 9 \\ -2y_9 + 4.5y_{10} &= 0.5 \exp(-2) \end{aligned} \qquad \text{(G)}$$

donde $x_i = i$. El PROGRAMA 10-1 resuelve las ecuaciones anteriores, de la (A) a la (G).

Resumen de Esta Sección

a) Se presentan los métodos numéricos básicos para los problemas de ecuaciones diferenciales de segundo orden con valores en la frontera. Se dan dos condiciones de frontera, una en el extremo izquierdo y otra en el extremo derecho del dominio.
b) Se aproxima el término de la segunda derivada mediante la aproximación por diferencias centrales.
c) El conjunto de ecuaciones en diferencias para cada problema es una ecuación tridiagonal en forma matricial.

10.3 ALGORITMO DE SOLUCION POR MEDIO DE SISTEMAS TRIDIAGONALES

Escribimos la ecuación tridiagonal obtenida en la sección 10.2 de la forma siguiente:

$$\begin{bmatrix} B_1 & C_1 & & & & & \\ A_2 & B_2 & C_2 & & & & \\ & A_3 & B_3 & C_3 & & & \\ & & & \ddots & & & \\ & & & A_i & B_i & C_i & \\ & & & & & \ddots & \\ & & & & & A_n & B_n \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_i \\ \vdots \\ \phi_N \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \vdots \\ D_i \\ \vdots \\ D_N \end{bmatrix} \qquad (10.3.1)$$

El algoritmo de solución para esta ecuación recibe el nombre de *solución tridiagonal* (una variante de la eliminación de Gauss) dada a continuación:

a) Se inicializan dos nuevas variables:

$$B'_1 = B_1 \quad \text{y} \quad D'_1 = D_1$$

b) Se calculan en forma recursiva las siguientes ecuaciones, en orden creciente de i hasta llegar a N:

$$R = A_i/B'_{i-1}$$

$$B'_i = B_i - RC_{i-1}$$

$$D'_i = D_i - RD'_{i-1} \text{ para } i = 2, 3, \ldots, N.$$

c) Se calcula la solución para la última incógnita:

$$\phi_N = D'_N/B'_N$$

d) Se calcula la ecuación siguiente en orden decreciente de i:

$$\phi_i = (D'_i - C_i\phi_{i+1})/B'_i, \quad i = N-1, N-2, \ldots, 1$$

En un programa de computadora no es necesario distinguir las variables con apóstrofe B'_i y D'_i de B_i y D_i puesto que las primeras se almacenan en los mismos espacios de memoria que las últimas. Así, el paso a) no es necesario en la programación verdadera.

La solución tridiagonal se emplea del PROGRAMA 10-1 al PROGRAMA 10-4. La siguiente es una subrutina en FORTRAN que lleva a cabo la solución tridiagonal:

```
      SUBROUTINE TRIDG(A,B,C,D,N)
      DIMENSION A(1),B(1),C(1),D(1)
      DO 10 I=2,N
      R=A(I)/B(I-1)
      B(I)=B(I)-R*C(I-1)
      D(I)=D(I)-R*D(I-1)
10    CONTINUE
      D(N)=D(N)/B(N)
      DO 20 I=N-1,1,-1
      D(I)=(D(I)-C(I)*D(I+1))/B(I)
20    CONTINUE
      RETURN
      END
```

Al concluir los cálculos de la subrutina, la solución está almacenada en el arreglo D(I).

Resumen de Esta Sección. La solución tridiagonal es el método numérico más básico que se emplea para resolver problemas de ecuaciones diferenciales ordinarias con valores en la frontera.

10.4 COEFICIENTES VARIABLES Y RETICULA CON ESPACIAMIENTO NO UNIFORME EN LA GEOMETRIA LAMINAR

En muchos problemas se tiene que los coeficientes de la ecuación diferencial dependen del espacio y se utiliza una retícula no uniforme. Estos casos aparecen, por ejemplo, cuando la geometría está formada por varios materiales.

La ecuación diferencial ordinaria de segundo orden para la geometría laminar con coeficientes variables es

$$-(p(x)\phi'(x))' + q(x)\phi(x) = S(x) \tag{10.4.1}$$

con las condiciones de frontera dadas por las ecuaciones (10.2.8) y (10.2.9). Denotaremos la longitud del intervalo de x_i a x_{i+1} por h_i. Supondremos que p, q y S de cada intervalo son constantes denotadas por p_i, q_i y S_i, respectivamente, como se muestra en la figura 10.4.

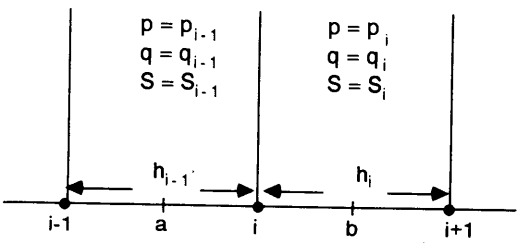


Figura 10.4 Constantes en los intervalos de la retícula

El método de integración es una forma natural para obtener ecuaciones en diferencias con coeficientes constantes por partes. En este método, integramos la ecuación (10.4.1) desde a hasta b (véase la figura 10.4):

$$-\int_a^b (p(x)\phi'(x))'\,dx + \int_a^b q(x)\phi(x)\,dx = \int_a^b S(x)\,dx \tag{10.4.2}$$

donde $a = x_i - h_{i-1}/2$ y $b = x_i + h_i/2$ (que son los puntos medios entre $i-1$ e i y entre i e $i+1$, respectivamente.

El primer término de la ecuación (10.4.2) es

$$-\int_a^b (p\phi')'\,dx = -(p\phi')_{i+\frac{1}{2}} + (p\phi')_{i-\frac{1}{2}}$$

Aproximamos las derivadas del lado derecho mediante las diferencias centrales:

$$(p\phi')_{i-\frac{1}{2}} = p_{i-1}(\phi_i - \phi_{i-1})/h_{i-1}$$
$$(p\phi')_{i+\frac{1}{2}} = p_i(\phi_{i+1} - \phi_i)/h_i$$

donde $p(x) = p_i$ para $x_i < x < x_{i+1}$. Así, el lado derecho de la ecuación (10.4.2) queda

$$-\int_a^b (p\phi')'\,dx = -\frac{p_{i-1}}{h_{i-1}}\phi_{i-1} + \left(\frac{p_{i-1}}{h_{i-1}} + \frac{p_i}{h_i}\right)\phi_i - \frac{p_i}{h_i}\phi_{i+1} \tag{10.4.3}$$

El segundo término de la ecuación (10.4.2) se transforma en

$$\int_a^b q(x)\phi(x)\,dx = \frac{1}{2}(q_{i-1}h_{i-1} + q_ih_i)\phi_i \tag{10.4.4}$$

donde aproximamos el factor $\phi(x)$ del integrando por ϕ_i. El lado derecho de la ecuación (10.4.1) queda como

$$\int_a^b S(x)\,dx = \frac{1}{2}(S_{i-1}h_{i-1} + S_ih_i) \tag{10.4.5}$$

Sustituimos las ecuaciones (10.4.3), (10.4.4) y (10.4.5) en la ecuación (10.4.2) para obtener

$$\begin{aligned} &-\frac{p_{i-1}}{h_{i-1}}\phi_{i-1} + \left(\frac{p_{i-1}}{h_{i-1}} + \frac{p_i}{h_i}\right)\phi_i - \frac{p_i}{h_i}\phi_{i+1} + \frac{1}{2}(q_{i-1}h_{i-1} + q_ih_i)\phi_i \\ &= \frac{1}{2}(S_{i-1}h_{i-1} + S_ih_i) \end{aligned} \tag{10.4.6}$$

Podemos escribir entonces la ecuación (10.4.6) en la forma

$$A_i\phi_{i-1} + B_i\phi_i + C_i\phi_{i+1} = D_i \tag{10.4.7}$$

donde

$$A_i = -\frac{p_{i-1}}{h_{i-1}}$$

$$B_i = \frac{p_{i-1}}{h_{i-1}} + \frac{p_i}{h_i} + \frac{1}{2}(q_{i-1}h_{i-1} + q_ih_i)$$

$$C_i = -\frac{p_i}{h_i}$$

$$D_i = \frac{1}{2}(S_{i-1}h_{i-1} + S_ih_i)$$

Si suponemos que las condiciones en la frontera están dadas por las ecuaciones (10.2.8) y (10.2.9), podemos deducir las ecuaciones en diferencias para los puntos frontera izquierdo y derecho integrando la ecuación (10.4.1). Consideremos el caso del punto frontera izquierdo; entonces, a y b de la ecuación (10.4.2) son

$a = x_1$ (el punto frontera izquierdo)

$b = x_1 + h_1/2$ (el punto medio entre x_1 y x_2)

Entonces el primer término de la ecuación (10.4.2) queda como

$$-\int_a^b (p\phi')'\,dx = -(p\phi')_{1+\frac{1}{2}} + (p\phi')_1 \tag{10.4.8}$$

Aproximamos el primer término del lado derecho mediante las diferencias centrales:

$$-(p\phi')_{1+\frac{1}{2}} = \frac{-p_1(\phi_2 - \phi_1)}{h_1} \tag{10.4.9}$$

El ϕ' del segundo término del lado derecho de la ecuación (10.4.8) se elimina mediante la ecuación (10.2.8). Así, la ecuación (10.4.8) se transforma en

$$-\int_a^b (p\phi')'\,dx = -\frac{p_1}{h_1}(\phi_2 - \phi_1) + p_1(-g_I + f_I\phi_1) \tag{10.4.10}$$

El segundo término y el lado derecho de la ecuación (10.4.2) quedan respectivamente como sigue:

$$\int_a^b q(x)\phi(x)\,dx = \frac{1}{2}q_1h_1\phi_1 \tag{10.4.11}$$

$$\int_a^b S(x)\,dx = \frac{1}{2}S_1h_1 \tag{10.4.12}$$

Sustituimos las ecuaciones (10.4.10) a (10.4.12) en la ecuación (10.4.2) para obtener

$$\left(\frac{p_1}{h_1} + p_1f_I + \frac{1}{2}q_1h_1\right)\phi_1 - \frac{p_1}{h_1}\phi_2 = \frac{1}{2}S_1h_1 + p_1g_L \tag{10.4.13}$$

o, en forma más compacta,

$$B_1\phi_1 + C_1\phi_2 = D_1 \tag{10.4.14}$$

La ecuación en diferencias para el punto frontera de la derecha se deduce de manera análoga y se escribe como

$$A_N\phi_{N-1} + B_N\phi_N = D_N \tag{10.4.15}$$

El conjunto de ecuaciones en diferencias obtenido de esta manera —es decir, las ecuaciones (10.4.14), (10.4.7) y (10.4.15)— tienen exactamente la forma de la ecuación (10.3.1).

Las ecuaciones en diferencias determinadas en esta sección satisfacen la ley de conservación. Antes de terminar con esta sección, mencionaremos la conservación de la propiedad en cuestión por parte de la ecuación (10.4.7) y la positividad de la solución.

Si definimos

$$Z_i = -\frac{p_i}{h_i}(\phi_{i+1} - \phi_i) \tag{10.4.16}$$

$$\overline{B}_i = \frac{1}{2}(q_{i-1}h_{i-1} + q_i h_i)\phi_i \tag{10.4.17}$$

podemos escribir la ecuación (10.4.7) como

$$Z_i - Z_{i-1} + \overline{B}_i = D_i \tag{10.4.18}$$

donde D_i se definió después de la ecuación (10.4.7). Si sumamos la ecuación (10.4.18) para $i = k, k + 1, \ldots, m$ obtenemos

$$Z_m - Z_{k-1} + \sum_{i=k}^{m} \overline{B}_i = \sum_{i=k}^{m} D_i \tag{10.4.19}$$

Esta ecuación satisface la conservación de la propiedad, representada por ϕ en $x_{k-\frac{1}{2}} < x < x_{m+\frac{1}{2}}$, puesto que el primer y segundo términos son los flujos hacia adentro y hacia afuera de la propiedad en cuestión, el tercero es la pérdida total y el lado derecho es la fuente total. Esta afirmación es cierta para cualquier elección de k y m.

Cuando el conjunto de ecuaciones satisface la ley de la conservación, se dice que las ecuaciones en diferencias están en la *forma de conservación*. Si estas ecuaciones se deducen de alguna otra forma, podrían no satisfacer la ley de la conservación.

Si se cumple que: a) se imponen condiciones en la frontera con significado físico y b) el coeficiente del término de pérdida no es negativo, entonces la matriz de coeficientes de las ecuaciones en diferencias dadas en la forma conservativa [y escritas como en la ecuación (10.3.1)] satisface las condiciones siguientes:

a) La matriz de coeficientes es simétrica.
b) Todos los coeficientes sobre la diagonal son positivos.
c) Todos los coeficientes fuera de la diagonal y que no sean nulos son negativos.
d) Los coeficientes de cada renglón satisfacen

$$b_i \geqslant -a_i - c_i$$

y de forma que en al menos uno de los renglones hay una desigualdad estricta.
e) Todos los a_i, b_i y c_i son distintos de cero (a_1 y c_N no existen).

Se puede demostrar que la inversa de la matriz que satisface las cinco condiciones anteriores es una matriz positiva; es decir, todos los elementos de la matriz inversa

son positivos. Esto implica que si $S_i \geqslant 0$ y existe al menos una i para la que se da la desigualdad estricta, la solución es positiva en todas partes.

Resumen de Esta Sección

a) Se considera que los coeficientes de la ecuación diferencial ordinaria dependen del tiempo, pero son constantes en cada intervalo de la retícula.
b) Las ecuaciones en diferencias forman una ecuación tridiagonal.
c) Las ecuaciones en diferencias que hemos obtenido en esta sección están en la forma conservativa.

10.5 PROBLEMAS CON VALORES EN LA FRONTERA PARA CILINDROS Y ESFERAS

La forma de obtener ecuaciones en diferencias para el caso de las ecuaciones diferenciales ordinarias de segundo orden en geometrías cilíndricas y esféricas es muy similar a la forma analizada en la sección 10.4. De nuevo, las ecuaciones en diferencias tendrán la forma de la ecuación (10.3.1), la cual se puede resolver por medio de la solución tridiagonal descrita en la sección 10.3. Por lo tanto, veremos cómo se obtienen dichas ecuaciones en diferencias.

Podemos escribir la ecuación diferencial ordinaria de segundo orden para el caso de geometrías cilíndricas y esféricas mediante una sola ecuación:

$$-\frac{1}{r^m}\frac{d}{dr}p(r)r^m\frac{d}{dr}\phi(r) + q(r)\phi(r) = S(r) \tag{10.5.1}$$

donde

$$m = 1 \text{ para el caso del cilindro}$$
$$m = 2 \text{ para el caso de la esfera}$$

Observemos también que si $m = 0$, la ecuación (10.5.1) representa la geometría laminar.

Si consideramos el caso en que los coeficientes dependan del tiempo y que exista una retícula no uniforme —como en la sección anterior— el método de integración es la forma más natural para obtener una aproximación por diferencias; es decir, debemos integrar la ecuación sobre una celda correspondiente a un volumen cilíndrico o esférico, según sea el caso de la geometría. (Por supuesto, es posible la diferenciación de la ecuación (10.5.1) siempre que q y S se promedien en forma adecuada.

Ahora mostraremos cómo se obtienen las ecuaciones en diferencias para el caso del cilindro ($m = 1$), utilizando las notaciones de h, p, q y S definidas en la figura 10.4 y suponiendo que p, q y S son constantes entre dos puntos consecutivos. Multiplicamos la ecuación (10.5.1) por r e integramos desde $a = r_{i-1/2}$ hasta $b = r_{i+1/2}$, los cuales son los puntos medios de $[r_{i-1}, r_i]$ y $[r_i, r_{i+1}]$, respectivamente:

$$-\int_a^b \left[\frac{d}{dr}rp(r)\frac{d}{dr}\phi(r)\right]dr + \int_a^b q(r)\phi(r)r\,dr = \int_a^b S(r)r\,dr \tag{10.5.2}$$

Aquí $r\,dr$ representa un elemento infinitesimal de volumen, dividido entre $2\pi L$ (L es la altura del cilindro circular). El primer término de la ecuación (10.5.2) se transforma en

$$p_{i-1}r_{i-\frac{1}{2}}\left[\frac{d}{dr}\phi(r)\right]_{i-\frac{1}{2}} - p_i r_{i+\frac{1}{2}}\left[\frac{d}{dr}\phi(r)\right]_{i+\frac{1}{2}}$$

y luego, utilizamos las aproximaciones por diferencias para las derivadas, con lo que tenemos

$$p_{i-1}r_{i-\frac{1}{2}}(\phi_i - \phi_{i-1})/h_{i-1} - p_i r_{i+\frac{1}{2}}(\phi_{i+1} - \phi_i)/h_i \tag{10.5.3}$$

El primer término, multiplicado por $2\pi L$, es el flujo total de la propiedad a través de la superficie cilíndrica en $a = r_{i-1/4}$ y el segundo es su análogo para $b = r_{i+1/2}$.

Podemos aproximar el segundo término de la ecuación (10.5.2) por

$$\int_a^b q(r)\phi(r)r\,dr = [v_l q_{i-1} + v_r q_i]\phi_i \tag{10.5.4}$$

y representa la pérdida total de la propiedad física en $[r_{i-1/2}, r_{i+1/2}]$, donde

$$v_l = \frac{1}{2}\left[r_i^2 - \left(r_i - \frac{h_{i-1}}{2}\right)^2\right] = \frac{h_{i-1}}{2}\left(r_i - \frac{h_{i-1}}{4}\right) \tag{10.5.5}$$

$$v_r = \frac{1}{2}\left[\left(r_i + \frac{h_i}{2}\right)^2 - r_i^2\right] = \frac{h_i}{2}\left(r_i + \frac{h_i}{4}\right) \tag{10.5.6}$$

Conviene observar que v_l por $2\pi L$ es el volumen de la celda cilíndrica entre $r = r_{i-1/2}$ y $r = r_i$, mientras que v_r es su análogo entre r_i y $r_{i+1/2}$. De manera análoga, podemos aproximar al tercer término de la ecuación (10.5.2) por

$$\int_a^b S(r)r\,dr = v_l S_{i-1} + v_r S_i \tag{10.5.7}$$

Si agrupamos todos los términos, la aproximación por diferencias de la ecuación (10.5.1) toma la forma tridiagonal. Aunque hemos descrito el caso cilíndrico con detalle, el análisis para el caso de la geometría esférica es esencialmente el mismo.

Las ecuaciones en diferencias que hemos obtenido en esta sección se encuentran en la forma conservativa. La matriz de coeficientes para el caso del cilindro tiene exactamente las mismas propiedades matemáticas del caso laminar (véase la sección 10.4), por lo que posee una matriz inversa positiva.

Se puede perder la conservación de la propiedad si las ecuaciones en diferencias se obtienen de forma distinta. Veamos un ejemplo de esto en la ecuación (10.5.1); para simplificar el análisis, supongamos que $m = 1$ y que p, q y s son constantes. Reescribimos el primer término para poner la ecuación (10.5.2) en la forma

$$-p\frac{d^2}{dr^2}\phi(r) - \frac{p}{r}\frac{d}{dr}\phi + q\phi(r) = S \tag{10.5.8}$$

Al obtener la diferenciación directa de la ecuación (10.5.8) en una retícula uniforme tenemos que

$$p\frac{-\phi_{i-1}+2\phi_i-\phi_{i+1}}{h^2}-p\frac{\phi_{i+1}-\phi_{i-1}}{2hr_i}+q\phi_i=S \tag{10.5.9}$$

Esta ecuación en diferencias viola la ley de la conservación, aunque se utiliza con frecuencia.

En general, y de ser posible, debemos tratar de evitar la forma no conservativa de las ecuaciones en diferencias, puesto que éstas son menos exactas que las aproximaciones por diferencias conservativas. De hecho, la precisión de la solución para la ecuación (10.5.9) es cada vez más pobre hacia el centro de la coordenada, donde r se anula [Smith].

Resumen de Esta Sección

a) Las ecuaciones en diferencias para las geometrías cilíndrica y esférica se obtienen al integrar la ecuación diferencial en el espacio.

b) Las ecuaciones en diferencias para las geometrías cilíndrica y esférica se escriben en la forma de una ecuación tridiagonal.

10.6 PROBLEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS NO LINEALES CON VALORES EN LA FRONTERA

Una ecuación diferencial ordinaria es no lineal si la incógnita aparece en forma no lineal, o bien si su(s) coeficiente(s) dependen de la solución. Por ejemplo, la ecuación de conducción del calor en un tubo refrigerante se vuelve no lineal si existe transferencia de calor por radiación hacia afuera de la superficie. La ecuación de difusión para una sustancia química es no lineal si tiene un término de pérdida en el que el coeficiente depende de la densidad de la sustancia. En un reactor nuclear, las propiedades de los materiales se ven afectadas en forma significativa por la cantidad de neutrones, aunque esto ocurre en forma indirecta si el nivel de potencia es alto, por lo que la ecuación que determina el flujo de neutrones se torna no lineal. Véanse [Kubicek y Hlavacek], así como [Nishida, Miura y Fujii] para numerosos ejemplos de problemas no lineales con valores en la frontera.

Los métodos de solución para los problemas no lineales con valores en la frontera requieren de la aplicación iterativa de un método de solución para los problemas lineales. Analizaremos dos métodos generales, considerando una ecuación no lineal de difusión dada por

$$\begin{gathered}-\phi''+0.01\phi^2=\exp(-x),\quad 0<x<H\\ \phi(0)=\phi(H)=0\end{gathered} \tag{10.6.1}$$

Antes de pasar a los algoritmos de solución numérica, debemos observar ciertos aspectos particulares de los problemas no lineales con valores en la frontera. En primer lugar, a diferencia de los problemas lineales, no se garantiza la existencia de la solución. En segundo lugar, un problema no lineal con valores en la frontera puede tener más de una solución. De hecho, mediante un algoritmo iterativo se podrían obtener distintas soluciones con distintas estimaciones iniciales. Por lo tanto, al obtener una solución numérica hay que investigar si tiene significado físico.

Sustitución Sucesiva Reescribimos la ecuación (10.6.1) como

$$-\phi'' + \alpha(x)\phi(x) = \exp(-x) \tag{10.6.2}$$

donde

$$\alpha(x) = 0.01\phi(x)$$

El método que analizaremos es una extensión del método de sustitución sucesiva descrito en el capítulo 3 y se desarrolla como sigue:

a) Se hace una estimación de $\alpha(x)$; por ejemplo, $\alpha(x) = 0.01$.
b) Se resuelve la ecuación (10.6.2) en forma numérica, como un problema lineal con valores en la frontera (puesto que α permanece fijo, la ecuación es lineal).
c) Se modifica el valor de $\alpha(x) = 0.01\,\phi(x)$ con el valor actualizado de $\phi(x)$ obtenido en b).
d) Se repiten los dos pasos anteriores hasta que el término $\phi\,(x)$ coincida en dos soluciones consecutivas dentro de una tolerancia fija de antemano.

Método de Newton. Supongamos que disponemos de una estimación de $\phi\,(x)$, la cual denotaremos por $\psi\,(x)$. En este caso, podemos expresar la solución exacta como

$$\phi(x) = \psi(x) + \delta\psi(x) \tag{10.6.3}$$

donde $\delta\psi(x)$ es una corrección de la estimación. Si sustituimos la ecuación (10.6.3) en la ecuación (10.6.1), obtenemos

$$-\delta\psi'' + (0.01)[2\psi\delta\psi + (\delta\psi)^2] = \psi'' - 0.01\psi^2 + \exp(-x) \tag{10.6.4}$$

Si ignoramos el término de segundo orden $(\delta\psi)^2$, tenemos que

$$-\delta\psi'' + 0.02\psi\delta\psi = \psi'' - 0.01\psi^2 + \exp(-x) \tag{10.6.5}$$

la cual se puede resolver como un problema lineal con valores en la frontera. Obtenemos entonces $\psi(x) + \delta\psi(x)$ como solución aproximada de la ecuación (10.6.1). Podríamos mejorar aún más la solución, utilizando el resultado más reciente como una nueva estimación. Este procedimiento es una extensión del método de Newton descrito en el capítulo 3.

Ejemplo 10.2

Determine, con base en el método de Newton, las linealizaciones de las ecuaciones en diferencias para la ecuación (10.6.1), en el dominio $0 < x < 2$ y con las condiciones en la frontera dadas por $\phi(0) = \phi(2) = 0$ con 10 intervalos en la retícula. Resolver las ecuaciones.

(Solución)

La forma linealizada de la ecuación (10.6.1) está dada por la ecuación (10.6.5). Con el espaciamiento $h = 2/10 = 0.2$, escribimos las ecuaciones en diferencias para el caso de la ecuación (10.6.5):

$$-\delta\psi_{i-1} + 2\delta\psi_i - \delta\psi_{i+1} + 0.02h^2\psi_i\delta\psi_i = \psi_{i-1} - 2\psi_i + \psi_{i+1} - 0.01h^2\psi_i^2 + h^2\exp(-ih), \quad i = 1, 2, \ldots, 9$$

donde $i = 0$ cuando $x = 0$; la ecuación está multiplicada por h^2. Podemos escribir esta ecuación en la forma de (10.3.1) si definimos

$$A_i = -1$$
$$B_i = 2 + 0.02h^2\psi_i$$
$$C_i = -1$$
$$D_i = \psi_{i-1} - 2\psi_i + \psi_{i+1} - 0.01h^2\psi_i^2 + h^2\exp(-ih)$$

Damos comienzo a la iteración de Newton haciendo una estimación de $\psi_i = 0$ para todos los puntos de la reticula. Después resolvemos las ecuaciones en diferencias para $i = 1, 2, 3, \ldots, 9$ mediante la solución tridiagonal. En la tabla 10.3 se da la solución iterativa para los primeros cinco puntos de la reticula.

Tabla 10.3

Número de iteración	Puntos de la reticula				
	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
1	0.0850	0.1406	0.1720	0.1837	0.1792
2	0.0935	0.1546	0.1891	0.2019	0.1970
3	0.0943	0.1560	0.1908	0.2038	0.1988
4	0.0944	0.1561	0.1910	0.2040	0.1990

Resumen de Esta Sección

a) Las ecuaciones diferenciales no lineales se resuelven en forma iterativa mediante la sustitución sucesiva o el método de Newton.

b) En cada paso de la iteración para la sustitución sucesiva o el método de Newton, se resuelven las ecuaciones en diferencias mediante la solución tridiagonal.

10.7 PROBLEMAS DE VALORES PROPIOS EN ECUACIONES DIFERENCIALES ORDINARIAS

Un problema con valores en la frontera se puede convertir también en un problema de valores propios, siempre que: a) el término de la fuente (o no homogéneo) de la ecuación diferencial sea igual a cero, y b) las dos condiciones en la frontera sean de la forma $\phi = 0$ o bien $\phi' = \gamma\psi$, donde γ es una constante. Este tipo de condiciones recibe el nombre de *condiciones homogéneas en la frontera*. Podemos normalizar las soluciones de un problema de valores propios mediante una constante arbitraria. Los problemas de valores propios de las ecuaciones diferenciales ordinarias tienen una relación muy cercana con los problemas de valores propios para matrices, pues-

to que un conjunto de ecuaciones en diferencias finitas para las primeras se transforma en un conjunto de ecuaciones para las segundas.

Existen muchos ejemplos de problemas de valores propios, entre los que se encuentran: la vibración armónica de varillas, cuerdas y vigas; el doblamiento de una viga debido a una fuerza longitudinal (véase problema 10.23); así como la distribución del flujo de neutrones en la fase crítica de un reactor nuclear [Nakamura] (véase también el problema 10.21).

Hay dos tipos de métodos de solución de las ecuaciones en diferencias para los problemas de valores propios. El primero se relaciona con los métodos directos de solución de problemas de valores propios para matrices, como se analizó en el capítulo 7. El segundo es de tipo iterativo.

Para explicar los métodos numéricos que resuelven los problemas de valores propios, consideremos primero un ejemplo: la vibración longitudinal de una varilla. La ecuación que determina dicha vibración es muy similar a las ecuaciones de vibración torsional de una varilla (o resorte) y de vibración transversal de una cuerda. Por lo tanto, si desarrollamos un programa de computadora para una de ellas, éste se puede aplicar a los otros dos tipos cambiando las definiciones de las variables y constantes.

La ecuación de onda para la vibración longitudinal de una varilla elástica con sección transversal variable es

$$\frac{\partial}{\partial x}\left(EA(x)\frac{\partial u}{\partial x}\right) = w(x)\frac{\partial^2 u}{\partial t^2} \tag{10.7.1}$$

donde $u = u(x, t)$ es el desplazamiento de la varilla en x y t, E es el módulo de elasticidad, $A(x)$ es el área de la sección transversal de la varilla y $w(x)$ es la masa de la varilla por unidad de longitud. Supondremos que H es la longitud de la varilla y que sus dos extremos están fijos.

Cuando la varilla tiene una oscilación armónica sostenida, podemos escribir la solución de (10.7.1) en la forma

$$u(x, t) = \operatorname{sen}(2\pi\nu t + \omega_0)f(x) \tag{10.7.2}$$

donde ν es la frecuencia de vibración, $f(x)$ es el modo espacial de oscilación y w_0 es la fase. Determinamos la ecuación para $f(x)$ sustituyendo (10.7.2) en (10.7.1) y dividiendo entre sen $(2\pi\nu t + \omega_0)$:

$$\frac{d}{dx}\left[EA(x)\frac{d}{dx}f(x)\right] = -(2\pi\nu)^2 w(x)f(x)$$

donde las derivadas parciales se cambian por las derivadas ordinarias, puesto que la ecuación ya no contiene a t. Esta ecuación se puede escribir en la siguiente forma compacta:

$$-[p(x)f'(x)]' = \lambda v(x)f(x) \tag{10.7.3}$$

donde $p(x) = EA(x)$, $\lambda = \nu^2$ y $v(x) = (2\pi)^2 w(x)$. Puesto que ambos extremos están fijos, las condiciones en la frontera son

$$f(0) = f(H) = 0 \tag{10.7.4}$$

Obtendremos ahora las ecuaciones en diferencias, suponiendo que el número total de puntos en la retícula es igual a $N + 2$ (incluyendo los extremos) y que el intervalo entre ellos es $h = H/(N + 1)$. Si nos fijamos en los tres puntos de la retícula que se muestran en la figura 10.5, la aproximación por diferencias para el lado izquierdo de la ecuación (10.7.3) es

$$\begin{aligned}(p(x)f')' &= \frac{p(b)f'(b) - p(a)f'(a)}{h} \\ &= \frac{p(b)\dfrac{f_{i+1} - f_i}{h} - p(a)\dfrac{f_i - f_{i-1}}{h}}{h} \\ &= \frac{p(a)f_{i-1} - (p(a) + p(b))f_i + p(b)f_{i+1}}{h^2} \qquad (10.7.5)\end{aligned}$$

en donde el lado derecho del primer renglón es una aproximación por diferencias centrales de $(p(x)f')'$, el segundo renglón se obtiene al aplicar las aproximaciones por diferencias centrales a f' y el tercero al reagrupar el segundo renglón.

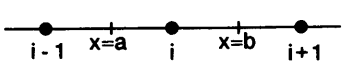


Figura 10.5 Configuración de la retícula en torno del punto i (a y b son los puntos medios)

Podemos aproximar el lado derecho de la ecuación (10.7.3) por

$$v(x)f(x) = v(x_i)f_i \qquad (10.7.6a)$$

o bien

$$v(x)f(x) = \frac{v(a) + v(b)}{2} f_i \qquad (10.7.6b)$$

Al sustituir las ecuaciones (10.7.5) y (10.7.6a) en la ecuación (10.7.3), la ecuación en diferencias se transforma en

$$-p(a)f_{i-1} + [p(a) + p(b)]f_i - p(b)f_{i+1} = \lambda v(x_i)h^2 f_i \qquad (10.7.7)$$

Si utilizamos la ecuación (10.7.6b) en vez de (10.7.6a), el lado derecho de la ecuación (10.7.7) cambia de manera acorde.

La deducción de ecuaciones en diferencias para otras coordenadas unidimensionales y otros problemas físicos es esencialmente la misma. En general, los problemas de valores propios de las ecuaciones diferenciales ordinarias se pueden escribir como

$$-\frac{1}{x^m}(x^m p(x)\phi')' + q(x)\phi(x) = \lambda v(x)\phi(x) \qquad (10.7.8)$$

donde $p(x)$, $q(x)$ y $v(x)$ son coeficientes ya determinados y m es un parámetro geométrico dado por $m = 0$ para el plano, $m = 1$ para el cilindro y $m = 2$ para la esfera. Si $q = 0$ y $m = 0$, la ecuación (10.7.8) se reduce a la forma de la ecuación (10.7.3). Si $m > 0$, primero hay que multiplicar dicha ecuación por x^m antes de calcular las aproximaciones por diferencias, como en la sección 10.5.

Cuando los coeficientes de la ecuación (10.7.8) son reales, todos los valores propios son reales. En particular, si $q \geqslant 0$, todos los valores propios son positivos (positivo definido). La función propia correspondiente al mínimo valor propio no se anula entre los extremos. La segunda función propia tiene una raíz, la tercera dos, etc., como lo ilustra la figura 10.6.

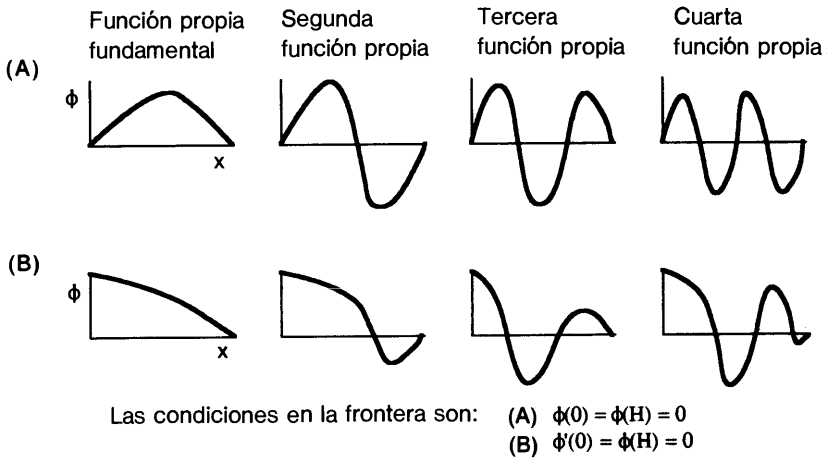


Figura 10.6 Distribución de las funciones propias.

Las ecuaciones en diferencias para la ecuación (10.7.8) se pueden escribir en forma tridiagonal:

$$\begin{aligned} B_1 f_1 + C_1 f_2 &= \lambda G_1 f_1 \\ A_2 f_1 + B_2 f_2 + C_2 f_3 &= \lambda G_2 f_2 \\ A_i f_{i-1} + B_i f_i + C_i f_{i+1} &= \lambda G_i f_i \\ &\vdots \\ A_N f_{N-1} + B_N f_N &= \lambda G_N f_N \end{aligned} \tag{10.7.9}$$

Podemos resolver la ecuación (10.7.9) ya sea en forma directa, mediante el método de valores propios de una matriz descrito en el capítulo 7, o en forma iterativa, como se explica en el resto de esta sección. Cuando se elija un método, hay que tomar en cuenta los aspectos siguientes:

a) El método matricial puede determinar todos los valores propios, incluyendo los valores propios complejos. Pero si el número de puntos en la retícula aumenta, el tiempo de cómputo se prolonga.

b) La precisión de la ecuación en diferencias se va perdiendo a medida que es mayor el número de oscilaciones espaciales (o, en forma equivalente, cuando el valor propio se hace más grande). Esto se debe a que el error de truncamiento de las ecuaciones en diferencias aumenta con rapidez al crecer el número de nodos de la oscilación espacial.

c) Si los coeficientes de la ecuación tridiagonal son simétricos, todos los valores propios son reales.

d) Por lo general, en la mayoría de los problemas científicos se buscan el valor propio más pequeño y los más próximos a éste, así como sus correspondientes funciones propias. Los métodos iterativos son adecuados para este tipo de problemas.

Método de la Potencia Inversa. Este es un método iterativo para calcular el valor propio más cercano a cero, así como su función propia correspondiente. Cuando el valor propio mínimo es real y positivo, este método sirve para determinar el valor propio fundamental y su función propia.

El método de la potencia inversa para la ecuación (10.7.9) es

$$A_i f_{i-1}^{(t)} + B_i f_i^{(t)} + C_i f_{i+1}^{(t)} = \lambda^{(t-1)} G_i f_i^{(t-1)} \qquad (10.7.10)$$
$$i = 1, 2, \ldots, N$$

donde t es el número de iteración y $f_0 = f_{N+1} = 0$. La solución se calcula de la manera siguiente:

Paso 1: se hacen las estimaciones iniciales de $f_i^{(0)}$ para toda i, las cuales pueden ser sin excepción iguales a cero, salvo para un punto de la retícula.

Paso 2: la estimación inicial $\lambda^{(0)}$ de λ se hace igual a la unidad.

Paso 3: se calcula el valor de $f_i^{(1)}$ en la ecuación (10.7.10) mediante la solución tridiagonal.

Paso 4: las ecuaciones siguientes determinan la nueva estimación de λ :

$$\lambda^{(1)} = \lambda^{(0)} \frac{\sum_i G_i f_i^{(0)} f_i^{(1)}}{\sum_i G_i [f_i^{(1)}]^2} \qquad (10.7.11)$$

Paso 5: se sustituyen $\lambda^{(1)}$ y $f_i^{(1)}$ en el lado derecho de (10.8.10), en la que se determina el valor de $f_i^{(2)}$ mediante la solución tridiagonal.

Paso 6: se repite la operación similar al paso 5, incrementando el ciclo de iteración t. El cálculo de λ después de cada ciclo es

$$\lambda^{(t)} = \lambda^{(t-1)} \frac{\sum_i G_i f_i^{(t-1)} f_i^{(t)}}{\sum_i G_i [f_i^{(t)}]^2} \qquad (10.7.12)$$

Paso 7: la iteración se detiene cuando se satisface el criterio de convergencia

$$|\lambda^{(t-1)}/\lambda^{(t)} - 1| < \varepsilon$$

donde ε es un criterio de convergencia determinado de antemano.

Con el fin de simplificar los cálculos, podríamos sustituir la ecuación (10.7.12) por

$$\lambda^{(t)} = \lambda^{(t-1)} \frac{\sum_i G_i f_i^{(t-1)}}{\sum_i G_i f_i^{(t)}} \tag{10.7.13}$$

o bien

$$\lambda^{(t)} = \lambda^{(t-1)} \frac{\sum_i f_i^{(t-1)}}{\sum_i f_i^{(t)}} \tag{10.7.14}$$

Sin embargo, las tasas de convergencia del valor propio con estas dos ecuaciones son más lentas que con la ecuación (10.7.12).

Si $q > 0$, la tasa de convergencia decrece rápidamente al aumentar el valor de q, o bien al disminuir el tamaño del dominio. Para los problemas unidimensionales de valores propios, se puede mejorar la convergencia iterativa lenta mediante el método de la potencia inversa con desplazamiento, el cual se explica más adelante. Por otro lado, si $q = 0$ en la ecuación (10.7.8), la tasa de convergencia del método de la potencia inversa no se ve afectado por el número de puntos en la retícula.

Método de la Potencia Inversa con Desplazamiento. Este método (véase también la sección 7.3) se obtiene mediante una ligera modificación al método anterior, pero acelera en forma significativa la convergencia iterativa hacia el valor propio y la función propia fundamentales. También puede determinar valores y funciones propias reales mayores.

Hacemos el valor propio de la ecuación (10.7.9) igual a

$$\lambda = \lambda_e + \delta\lambda \tag{10.7.15}$$

donde λ_e es una estimación del valor propio a determinar y $\delta\lambda$ es la corrección. Así, podemos escribir la ecuación (10.7.9) como

$$\begin{aligned}
(2-\lambda_e)f_1 - f_2 &= \delta\lambda f_1 \\
-f_1 + (2-\lambda_e)f_2 - f_3 &= \delta\lambda f_2 \\
-f_2 + (2-\lambda_e)f_3 - f_4 &= \delta\lambda f_3 \\
&\vdots \\
-f_{N-1} + (2-\lambda_e)f_N &= \delta\lambda f_N
\end{aligned} \tag{10.7.16}$$

donde, para simplificar este ejemplo, hemos supuesto que los coeficientes son $A = C = -1$, $B = 2$ y $D = 1$. Resolvemos la ecuación (10.7.16) mediante el método de la potencia inversa, considerando a $\delta\lambda$ como una definición alternativa del valor propio. Una vez que se ha determinado $\delta\lambda$, se calcula el verdadero valor propio, sustituyendo $\delta\lambda$ en la ecuación (10.7.15). La convergencia es más rápida si λ_e es más cercano al verdadero valor propio. Incluso con una estimación pobre de λ_e, la convergencia mejora de manera significativa. Cuando no se dispone de una estimación del valor propio fundamental, ésta se puede obtener después de unos cuantos pasos de iteración con $\lambda_e = 0$.

Sin embargo, λ_e no debe ser muy cercano a otro valor propio. En tal caso, sería grave el error de truncamiento en la solución tridiagonal y la solución numérica no sería confiable (la matriz de coeficientes estaría mal condicionada). De hecho, si λ_e fuera igual a un valor propio, el lado izquierdo de (10.7.16) sería singular y la solución tridiagonal se detendría debido a un error aritmético, como un desbordamiento o una división entre cero. Véase el PROGRAMA 10-4.

Ejemplo 10.3

Utilice el PROGRAMA 10-4 para determinar los tres primeros valores y funciones propias.

(Solución)

Determinaremos primero el valor propio y la función propia fundamental de la ecuación (10.7.10) mediante el PROGRAMA 10-4. Puesto que no existe una estimación del primer valor propio, hacemos $\lambda_e = 0$. Las aproximaciones sucesivas de este valor propio son:

Número de iteración	
1	0.0819672
2	0.0810257
3	0.0810142
4	0.0810140
El valor final de λ es 0.0810140	

La función propia para el primer valor propio es:

i-ésimo punto de la reticula	f_i
1	0.28469
2	0.54628
3	0.76359
4	0.91901
5	1.00000
6	1.00000
7	0.91901
8	0.76359
9	0.54628
10	0.28469

Si ponemos $\lambda_e = 0.2$ como primera estimación del segundo valor propio, el valor propio converge a 0.081014, que sigue siendo el valor propio funda-

mental. Si la segunda estimación es $\lambda_e = 0.5$, obtenemos $\lambda = 0.690274$, para el cual la función propia cambia de signo dos veces. Por lo tanto, este valor es el tercer valor propio. El segundo debe estar entre el primero y el tercero. Así, hacemos $\lambda_e = 0.4$, de lo que resulta $\lambda = 0.317493$, que es el segundo valor propio, debido a que la función propia sólo cambia de signo una vez.

La distribución de la segunda y tercera funciones propias es como sigue:

i	Segunda función propia ($\lambda = 0.317493$) f_i	Tercera función propia ($\lambda = 0.690274$) f_i
1	−0.54560	0.76386
2	−0.91809	0.99999
3	−0.99923	0.54498
4	−0.76319	−0.28743
5	−0.28475	−0.92291
6	0.28428	−0.92291
7	0.76326	−0.28743
8	0.99999	0.54498
9	0.91918	0.99999
10	0.54638	0.76386

Resumen de Esta Sección

a) La ecuación diferencial y las condiciones en la frontera en un problema de valores propios son ambas homogéneas.
b) Si el mínimo valor propio es real y positivo, éste se puede calcular mediante el método de la potencia inversa y el método de la potencia inversa con desplazamiento.

10.8 ANALISIS DE CONVERGENCIA DE LOS METODOS ITERATIVOS

En esta sección explicaremos por qué convergen los métodos de la potencia inversa y de la potencia inversa con desplazamiento. Véase el capítulo 2 de Nakamura (1986) para la convergencia iterativa de problemas en ingeniería nuclear.

Si $q(x) \geqslant 0$ y $v(x) > 0$ en la ecuación (10.7.8), la ecuación en diferencias dada por (10.7.9) tiene las siguientes propiedades:

a) Existen N valores propios reales, positivos y distintos, donde N es el número de incógnitas en la ecuación (10.7.9).
b) La función propia fundamental (asociada con el valor propio mínimo) sólo se anula en los extremos.
c) La segunda función propia tiene una raíz en el dominio, mientras que la n-ésima función propia tiene $n - 1$ raíces dentro del dominio.

Para el análisis posterior, escribimos la ecuación (10.7.9) en forma compacta:

$$Mf = \lambda Gf \tag{10.8.1}$$

donde M y G son las matrices tridiagonal y diagonal, respectivamente y f representa un vector propio (función propia en forma vectorial).

Suponemos que los N valores propios distintos de la ecuación (10.8.1) están ordenados de forma que

$$0 < \lambda_0 < \lambda_1 < \cdots < \lambda_{N-1}$$

donde λ_0 es el valor propio fundamental. Si denotamos el vector propio correspondiente a λ_n por u_u, podemos escribir la ecuación (10.8.1) como

$$Mu_n = \lambda_n G u_n, \quad n = 0, 1, \ldots, N-1 \tag{10.8.2}$$

En el caso en que M sea una matriz simétrica y todos los elementos de la diagonal de G sean distintos de cero, los valores propios tienen las siguientes propiedades:

a) Dos valores propios distintos son ortogonales entre sí:

$$(u_m)^T G u_n = 0 \quad \text{si } n \neq m \text{ (relación de ortogonalidad)} \tag{10.8.3}$$

b) Puesto que todos los vectores propios son independientes, cualquier vector de orden N se puede expresar como combinación lineal de los vectores propios:

$$z = \sum_{n=0}^{N-1} a_n u_n \quad \text{(compleción)} \tag{10.8.4}$$

donde z es un vector cualquiera de orden N y a_n es un coeficiente.

Ahora escribimos el método de la potencia inversa dado por la ecuación (10.7.10) como

$$Mf^{(t)} = \lambda^{(t-1)} G f^{(t-1)} \tag{10.8.5}$$

Si utilizamos la propiedad de la ecuación (10.8.4), podemos desarrollar la estimación inicial $f^{(0)}$ mediante los vectores propios, de la manera siguiente:

$$f^{(0)} = \sum_{n=0}^{N-1} c_n u_n \tag{10.8.6}$$

donde c_n es un coeficiente que se puede determinar gracias a la relación de ortogonalidad (10.8.3). Hacemos la estimación inicial $\lambda^{(0)}$ igual a uno. Podemos escribir la solución de (10.8.5) para $f^{(1)}$ como

$$f^{(1)} = \sum_{n=0}^{N-1} c_n \left(\frac{1}{\lambda_n}\right) u_n \tag{10.8.7}$$

donde usamos $\lambda^{(0)} = 1$ en la ecuación (10.8.5), para $t = 1$. Podemos verificar la ecuación (10.8.7) al sustituir (10.8.6) en el lado derecho de (10.8.5) con $t = 1$, utilizando (10.8.2).

Para el segundo ciclo de iteración, sustituimos la ecuación (10.8.7) en el lado derecho de (10.8.5) con $t = 2$ y determinamos el valor de $f^{(2)}$. Podemos escribir el resultado como

$$f^{(2)} = \lambda^{(1)} \sum_{n=0}^{N-1} c_n \left(\frac{1}{\lambda_n}\right)^2 u_n \tag{10.8.8}$$

La solución iterativa, después de t ciclos de iteración, es

$$f^{(t)} = \lambda^{(1)}\lambda^{(2)} \cdots \lambda^{(t-1)} \sum_n c_n \left(\frac{1}{\lambda_n}\right)^t u_n \tag{10.8.9}$$

o, en forma equivalente,

$$f^{(t)} = \frac{\lambda^{(1)}\lambda^{(2)} \cdots \lambda^{(t-1)}}{(\lambda_0)^t} \left[c_0 u_0 + c_1 \left(\frac{\lambda_0}{\lambda_1}\right)^t u_1 + c_2 \left(\frac{\lambda_0}{\lambda_2}\right)^t u_2 + \cdots \right] \tag{10.8.10}$$

Puesto que λ_0 es el valor propio mínimo, todos los términos con el factor $(\lambda_0/\lambda_n)^t$ se anulan al tender t a infinito. Así, la ecuación (10.8.10) tenderá a

$$f^{(t)} \longrightarrow Cu_0$$

donde C es una constante. La convergencia está determinada por la razón definida por $\sigma \equiv \lambda_0/\lambda_1$, llamada *razón de dominancia*, que es la razón en la que el coeficiente de u_1 en la ecuación (10.8.10) decrece en un ciclo de iteración.

Explicaremos ahora la convergencia del valor propio. Para simplificar la exposición, primero reescribimos la ecuación (10.7.11) con vectores y matrices como

$$\lambda^{(t)} = \lambda^{(t-1)} \frac{(f^{(t-1)})^T G f^{(t)}}{(f^{(t)})^T G f^{(t)}} \tag{10.8.11}$$

Utilizamos la ecuación (10.8.10) para transformar el numerador de (10.8.11):

$$(f^{(t-1)})^T G f^{(t)} = \frac{[\lambda^{(1)}\lambda^{(2)} \cdots \lambda^{(t-2)}]^2 \lambda^{(t-1)}}{(\lambda_0)^{2t-1}} \left[c_0^2 b_0 + c_1^2 \left(\frac{\lambda_0}{\lambda_1}\right)^{2t-1} b_1 + \cdots \right] \tag{10.8.12}$$

donde

$$b_n = (u_n)^T G u_n$$

Además, hemos utilizado las relaciones de ortogonalidad (10.8.3). De manera análoga, el denominador se puede escribir como

$$(f^{(t)})^T G f^{(t)} = \frac{[\lambda^{(1)}\lambda^{(2)} \cdots \lambda^{(t-1)}]^2}{(\lambda_0)^{2t}} \left[c_0^2 b_0 + c_1^2 \left(\frac{\lambda_0}{\lambda_1}\right)^{2t} b_1 + \cdots \right] \tag{10.8.13}$$

Sustituimos las ecuaciones (10.8.12) y (10.8.13) en la ecuación (10.8.11) y reagrupamos los términos:

$$\lambda^{(t)} = \lambda_0 \frac{1 + k_1 \left(\frac{\lambda_0}{\lambda_1}\right)^{2t-1} + k_2 \left(\frac{\lambda_0}{\lambda_2}\right)^{2t-1} + \cdots}{1 + k_1 \left(\frac{\lambda_0}{\lambda_1}\right)^{2t} + k_2 \left(\frac{\lambda_0}{\lambda_2}\right)^{2t} + \cdots} \tag{10.8.14}$$

donde

$$k_m = \left(\frac{c_m}{c_0}\right)^2 \frac{b_m}{b_0}$$

La ecuación anterior tiende a λ_0 cuando t crece, puesto que $\lambda_0/\lambda_1 < 1$.

Conviene observar también que la razón definida por

$$\frac{\lambda^{(t)} - \lambda^{(t-1)}}{\lambda^{(t-1)} - \lambda^{(t-2)}} \tag{10.8.15}$$

tiende al cuadrado de la razón de dominancia cuando t crece. La demostración se deja como ejercicio para el lector (véase el problema 10.25).

El radio de dominancia eficaz para el método de la potencia inversa con desplazamiento es

$$\frac{\lambda_0 - \lambda_e}{\lambda_1 - \lambda_e} \tag{10.8.16}$$

Cuando $\lambda_e \simeq \lambda_0$, el valor absoluto de la razón que aparece en la ecuación (10.8.16), es significativamente menor que λ_0/λ_1. Esto explica por qué el método de la potencia inversa con desplazamiento converge mucho más rápido que el método de la potencia inversa. De manera similar, se explica la convergencia del método con desplazamiento a los valores propios mayores.

Resumen de Esta Sección

a) La convergencia del método de la potencia inversa se explica al considerar el problema como positivo definido (todos los valores propios son reales y positivos).

b) Podemos desarrollar una estimación inicial en términos de los vectores propios. Con el método de la potencia inversa, la magnitud de cada vector propio disminuye según la potencia inversa del valor propio correspondiente.

c) La razón de convergencia del método de la potencia inversa queda determinada por la razón de dominancia.

d) El método de la potencia inversa con desplazamiento es una modificación del método de la potencia inversa, pero su razón de dominancia real es significativamente menor que la del método de la potencia inversa.

10.9 DOBLAMIENTO Y VIBRACION DE UNA VIGA

Los problemas de ecuaciones diferenciales ordinarias con valores en la frontera de orden superior a dos se pueden resolver de manera análoga. Como ejemplo, mostraremos algunos métodos numéricos para el doblamiento y vibración de una viga como un problema de una ecuación diferencial ordinaria de cuarto orden, con valores en la frontera. Primero analizaremos al doblamiento de una viga bajo una carga distribuida $P(x)$, que es un problema no homogéneo con valores en la frontera.

Si la carga distribuida es $P(x)$, podemos escribir la ecuación para el desplazamiento de la viga como

$$\frac{d^2}{dx^2}\left(EI\frac{d^2y}{dx^2}\right) = P(x) \tag{10.9.1}$$

donde E es el módulo de elasticidad e I es el momento de inercia de la sección transversal, y es el desplazamiento y P es la carga, como se muestra en la figura 10.7. El producto EI recibe el nombre de *rigidez a la flexión*.

Calcularemos las aproximaciones en diferencias finitas de la ecuación (10.9.1) para una viga no uniforme que está detenida en el extremo izquierdo pero queda libre en el extremo derecho, como en la figura 10.7.

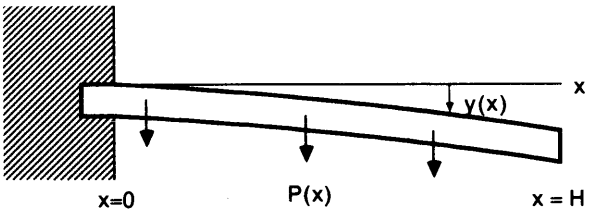


Figura 10.7 Una viga sujeta en uno de los extremos

Las condiciones en la frontera para este problema son

$$y(0) = 0 \quad \text{y} \quad y'(0) = 0 \text{ para la frontera izquierda}$$

$$M = y''(H) = 0 \quad \text{y} \quad V = y'''(H) = 0 \text{ para la frontera derecha}$$

donde M y V son los momentos de doblamiento y esfuerzo respectivamente, en la frontera derecha.

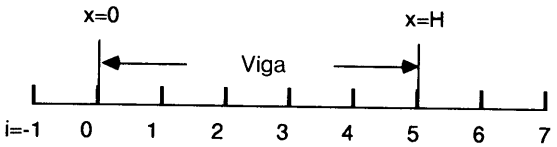


Figura 10.8 Una retícula para la viga ($i = -1$, 6 y 7 son los puntos hipotéticos en la retícula)

Para obtener las ecuaciones en diferencias, consideramos la retícula que se muestra en la figura 10.8. Aquí, los puntos de la retícula en $i = -1$, 6 y 7 son hipotéticos. Supondremos que la retícula tiene un espaciamiento uniforme. Escribimos entonces (10.9.1) como

$$EI_i y_i'''' + 2EI_i' y_i''' + EI_i'' y_i'' = P(x) \tag{10.9.2}$$

Las derivadas y'''', y''' y y'' se evalúan numéricamente mediante las aproximaciones por diferencias centrales:

$$\begin{aligned} y'''' &= (y_{i-2} - 4y_{i-1} + 6y_i - 4y_{i+1} + y_{i+2})/h^4 \\ y''' &= (-y_{i-2} + 2y_{i-1} - 2y_{i+1} + y_{i+2})/2h^3 \\ y'' &= (y_{i-1} - 2y_i + y_{i+1})/h^2 \end{aligned} \tag{10.9.3}$$

Podemos calcular los términos I'_i e I''_i mediante las aproximaciones por diferencias finitas como sigue: para i desde 1 hasta N — 1,

$$\begin{aligned} I_i' &= \frac{I_{i+1} - I_{i-1}}{2h} \\ I_i'' &= \frac{I_{i+1} - 2I_i + I_{i-1}}{h^2} \end{aligned} \tag{10.9.4}$$

donde utilizamos la aproximación por diferencias centrales; para N,

$$\begin{aligned} I_N' &= \frac{3I_N - 4I_{N-1} + I_{N-2}}{(2h)} \\ I_N'' &= \frac{-2I_N + 5I_{N-1} - 4I_{N-2} + I_{N-3}}{h^2} \end{aligned} \tag{10.9.4}$$

en donde usamos la aproximación por diferencias hacia atrás.

Al sustituir las ecuaciones (10.9.3) a (10.9.5) en la ecuación (10.9.2), las ecuaciones en diferencias son:

$$a_i y_{i-2} + b_i y_{i-1} + c_i y_i + d_i y_{i+1} + e_i y_{i+2} = f_i \quad i = 1, 2, \ldots, N \tag{10.9.6}$$

donde

$$\begin{aligned} a_i &= \quad EI_i/h^4 - \quad EI_i'/h^3 \\ b_i &= -4EI_i/h^4 + 2EI_i'/h^3 + \quad EI_i''/h^2 \end{aligned}$$

$$
\begin{aligned}
c_i &= 6EI_i/h^4 + 2EI''_i/h^2 \\
d_i &= -4EI_i/h^4 - 2EI'_i/h^3 + EI''_i/h^2 \\
e_i &= EI_i/h^4 + EI'_i/h^3 \\
f_i &= P(x_i)
\end{aligned}
$$

Si $i = 1$, la ecuación (10.9.6) tiene la forma

$$a_1 y_{-1} + b_1 y_0 + c_1 y_1 + d_1 y_2 + e_1 y_3 = f_1 \tag{10.9.7}$$

que contiene un punto hipotético $i = -1$ fuera del dominio. En (10.9.7), hacemos $y_0 = 0$, ya que $y(0) = 0$ es una condición en la frontera. La segunda condición en la frontera para el extremo izquierdo, $y'(0) = 0$, se puede aproximar mediante $(y_1 - y_{-1})/2h = 0$, por lo que podemos hacer $y_{-1} = y_1$. Al sustituir estas relaciones en (10.9.7), se tiene que

$$(a_1 + c_1)y_1 + d_1 y_2 + e_1 y_3 = f_1 \tag{10.9.8}$$

Para trabajar con las condiciones en la frontera derecha, escribimos las aproximaciones por diferencias de dichas condiciones:

$$
\begin{aligned}
y''' &= \frac{-y_{N-2} + 2y_{N-1} - 2y_{N+1} + y_{N+2}}{2h^3} = 0 \\
y'' &= \frac{y_{N-1} - 2y_N + y_{N+1}}{2h^2} = 0
\end{aligned}
\tag{10.9.9}
$$

Podemos considerar las ecuaciones (10.9.9) como miembros de un conjunto de ecuaciones simultáneas.

Así, el conjunto de ecuaciones a resolver toma la siguiente forma matricial:

$$
\begin{bmatrix}
a_1 + c_1, & d_1, & e_1 & & & & \\
b_2, & c_2, & d_2, & e_2 & & & \\
a_3, & b_3, & c_3, & d_3, & e_3 & & \\
 & a_4, & b_4, & c_4, & d_4, & e_4 & \\
 & & a_5, & b_5, & c_5, & d_5, & e_5 \\
 & & -1, & 2, & 0, & -2, & 1 \\
 & & & 1, & -2, & 1, & 0
\end{bmatrix}
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix}
=
\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ 0 \\ 0 \end{bmatrix}
\tag{10.9.10}
$$

Esta ecuación se resuelve mediante la eliminación de Gauss o la descomposición LU. Véase el PROGRAMA 10-5.

Consideremos ahora que existe una vibración armónica de la misma viga [Thomson]. Para una oscilación armónica, la ecuación (10.9.1) se reemplaza por

$$\frac{d^2}{dx^2}\left[EI\frac{d^2}{dx^2}\bar{y}(x,t)\right] = -w(x)\frac{d^2}{dt^2}\bar{y}(x,t) \tag{10.9.11}$$

donde $\bar{y}$ es el desplazamiento en x al tiempo t y $w(x)$ es la masa de la viga por unidad de longitud en el punto x. Buscamos la solución que tenga la forma

$$\bar{y}(x, t) = y(x) \exp(j\omega t) \tag{10.9.12}$$

donde $y(x)$ es el modo armónico y ω es la velocidad angular de la oscilación (2π veces la frecuencia) y $j = \sqrt{-1}$. Al sustituir la ecuación (10.9.12) en (10.9.11) y dividir entre exp $(j\omega t)$ obtenemos

$$\frac{d^2}{dx^2}\left[EI\frac{d^2}{dx^2}y(x)\right] = \lambda w(x)y(x) \tag{10.9.13}$$

donde $\lambda = \omega^2$ se considera un valor propio.

El problema de valores propios analizado aquí se implanta en el PROGRAMA 10-6.

Resumen de Esta Sección

a) Se analizan los métodos numéricos para resolver problemas de doblamiento y vibración de una viga.
b) Se resuelven las ecuaciones en diferencias para el problema de doblamiento mediante la descomposición LU (puesto que las ecuaciones diferenciales no son de segundo orden y las ecuaciones en diferencias ya no son tridiagonales).
c) El problema de la vibración es un problema de valores propios. Sus ecuaciones en diferencias son esencialmente iguales a las del problema de doblamiento, excepto que en este caso son homogéneas y tienen valores propios. Se utiliza el método de la potencia inversa con desplazamiento para calcular no sólo el modo fundamental sino también los modos mayores.

PROGRAMAS

PROGRAMA 10-1 Solución de problemas lineales con valores en la frontera

A) Explicaciones

El PROGRAMA 10-1 resuelve la ecuación (G) del ejemplo 10.1. El número total de incógnitas y los coeficientes de la ecuación tridiagonal se definen de acuerdo con dicho ejemplo.

El esquema de solución tridiagonal se implanta en la subrutina TDRG. Al regresar de la subrutina, la solución del esquema tridiagonal se almacena en la variable D(I). Se define D(0) sólo para fines de impresión.

B) Variables

A(I), B(I), C(I), D(I): A_i, B_i, C_i, D_i de la ecuación (10.3.1), respectivamente.
N: número total de incógnitas.

C) Listado

```
C------CSL/F10-1.FOR     PROBLEMA DE EDO LINEAL CON VALORES EN LA FRONTERA
       DIMENSION A(20),B(20),C(20),D(20)
       PRINT *
       PRINT *,'CSL/F10-1    PROBLEMA DE EDO LINEAL CON VALORES EN LA FRONTERA'

       N=10                          ! Número de puntos en la retícula
       DO 140 I=1, N
          X=I
          A(I)=-2                    ! Coeficientes tridiagonales
          B(I)=5
          C(I)=-2
          D(I)=EXP(-0.2*X)           ! Término fuente (no homogéneo)
140    CONTINUE
       D(1)=D(1)+2                   ! Ajuste con la condición en la frontera
       D(N)=D(N)*0.5
       B(N)=4.5
       CALL TRDG(A,B,C,D,N)
       D(0)=1                        ! Definición de D(0) para fines de impresión
       PRINT*,'   PUNTO DE LA RETICULA SOLUCION '
       DO 175 I=0, N
          PRINT 171, I,D(I)          ! Impresión de la solución
175    CONTINUE
171    FORMAT(2X, I5, 7X,F15.6)
       PRINT *
       STOP
       END
C*********************************
      SUBROUTINE TRDG(A,B,C,D,N)     ! Subrutina tridiagonal
      DIMENSION A(1),B(1),C(1),D(1)
      DO 220 I=2, N
         R=A(I)/B(I-1)
         B(I)=B(I)-R*C(I-1)
         D(I)=D(I)-R*D(I-1)
220   CONTINUE
      D(N)=D(N)/B(N)
      DO 240 I=N-1, 1 ,-1
         D(I)=(D(I)-C(I)*D(I+1))/B(I)
240   CONTINUE
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F10-1      PROBLEMA DE EDO LINEAL CON VALORES EN LA FRONTERA
PUNTO DE LA RETICULA    SOLUCION
      0                 1.000000
      1                 0.846449
      2                 0.706756
      3                 0.585282
      4                 0.482043
      5                 0.395162
      6                 0.321921
      7                 0.259044
      8                 0.202390
      9                 0.145983
     10                 0.079919
```

PROGRAMA 10-2 Solución de problemas no lineales con valores en la frontera

A) Explicaciones

El PROGRAMA 10-2 resuelve el problema del ejemplo 10.2. Las constantes, parámetros y la estimación inicial de ψ_i se definen al principio. La iteración comienza en S-150. La ecuación tridiagonal se resuelve en la subrutina TDRG. Al regresar de la subrutina, D(I) contiene a $\delta\psi_i$.NT cuenta el número de puntos en la retícula que no satisfacen el criterio de convergencia. Si la cuenta es cero, el programa comienza la impresión final. Si el contador es mayor que cero, la iteración continúa. En S-250, la estimación se actualiza, sumando $\delta\psi_i$ veces ω al ψ_i anterior. Para mejorar la estabilidad de la solución iterativa, el valor de $\delta\psi_i$ se decrementa en un factor de ω, que recibe el nombre de parámetro de *baja relajación* y cumple $0 < \omega < 1$. La estimación actualizada se imprime en S-160.

B) Variables

N: número de variables
EP: criterio de convergencia
H: espaciamiento de la retícula
PS(I): ψ_i
A(I), B(I), C(I): coeficientes de la ecuación tridiagonal
D(I): término no homogéneo de la ecuación tridiagonal; también es $\delta\psi_i$ al terminar de obtener la solución
NT: número de puntos en la retícula que no satisfacen el criterio de convergencia
K: contador de las interaciones
ω: parámetro de baja relajación $0 < \omega < 1$

C) Listado

```
C-----CSL/F10-2.FOR     PROBLEMA DE EDO NO LINEAL CON VALORES EN LA FRONTERA
      COMMON N, A(30),B(30),C(30),D(30),PS(20)
      PRINT *
      PRINT *,'CSL/F10-2.FOR PROBLEMA DE EDO NO LINEAL CON VALORES EN LA FRONTERA'
130   N=9
      H=0.2
      H2=H*H
      K=0
      EP=0.0001
      W=0.9
      DO 131 I=1, N
      PS(I)=0
131   CONTINUE
150   K=K+1
      DO I=1, N
         A(I)=-1
         B(I)=2+0.02*H2*PS(I)
         C(I)=-1
         D(I)=PS(I-1)-2*PS(I)+PS(I+1)-0.01*H2*PS(I)**2+EXP(-0.2*I)*H2
```

```
      END DO
      CALL TRDG(A,B,C,D,N)
190   NT=0
      DO I=1, N
        IF( ABS(D(I)) .GT.  EP) NT=NT+1
      END DO
200   PRINT *,' IT.NO.=', K
      PRINT 215 ,NT
215   FORMAT(' NUMERO DE PUNTOS QUE NO SATISFACEN EL CRITERIO=', I2)
      IF (NT.EQ.0) GOTO 270
240   PRINT *,'     I       PSI(I)        DEL PSI(I)'
      DO I=1, N
         PS(I)=D(I)*W +PS(I)
         PRINT 257, I, PS(I), D(I)
      END DO
      WRITE (*, 3000)  (PS(I),I=1,5)
3000  FORMAT(1X,5F8.4)
257   FORMAT(3X,I2, 3X, 1P2E13.4)
      GOTO 150
270   PRINT 275
275   FORMAT( /' --- RESULTADO FINAL ---'/'--- PUNTO DE LA RETICULA SOLUCION')
280   PS(0)=0
290   DO 295 I=0, N
      PRINT 296, I,PS(I)
295   CONTINUE
296   FORMAT(6X, I2, F12.5)
      END
C*******************
C     Favor de copiar la subrutina TRDG del programa 10-1.
```

D) Ejemplo de salida

```
CSL/F10-2.FOR   PROBLEMA DE EDO NO LINEAL CON VALORES EN LA FRONTERA

 IT.NO.=            1
NUMERO DE PUNTOS QUE NO SATISFACEN EL CRITERIO = 9
    I       PSI(I)        DEL PSI(I)
   1       8.5039E-02    9.4487E-02
   2       1.4060E-01    1.5623E-01
   3       1.7204E-01    1.9115E-01
   4       1.8371E-01    2.0412E-01
   5       1.7921E-01    1.9912E-01
   6       1.6147E-01    1.7941E-01
   7       1.3288E-01    1.4764E-01
   8       9.5416E-02    1.0602E-01
   9       5.0683E-02    5.6315E-02
  0.0850  0.1406  0.1720  0.1837  0.1792
 IT.NO.=            2
NUMERO DE PUNTOS QUE NO SATISFACEN EL CRITERIO = 9
    I       PSI(I)        DEL PSI(I)
   1       9.3500E-02    9.4013E-03
   2       1.5458E-01    1.5531E-02
   3       1.8913E-01    1.8990E-02
   4       2.0195E-01    2.0267E-02
   5       1.9700E-01    1.9764E-02
   6       1.7749E-01    1.7805E-02
   7       1.4607E-01    1.4653E-02
   8       1.0489E-01    1.0524E-02
   9       5.5716E-02    5.5921E-03
  0.0935  0.1546  0.1891  0.2020  0.1970
```

(La salida de la tercera iteración se omite.)

```
 IT.NO.=            4
NUMERO DE PUNTOS QUE NO SATISFACEN EL CRITERIO= 7
    I        PSI(I)          DEL PSI(I)
   1       9.4430E-02    9.3965E-05
   2       1.5612E-01    1.5522E-04
   3       1.9101E-01    1.8976E-04
   4       2.0396E-01    2.0252E-04
   5       1.9895E-01    1.9748E-04
   6       1.7925E-01    1.7790E-04
   7       1.4752E-01    1.4641E-04
   8       1.0593E-01    1.0516E-04
   9       5.6270E-02    5.5879E-05
  0.0944  0.1561  0.1910  0.2040  0.1990
NUMERO DE PUNTOS QUE NO SATISFACEN EL CRITERIO= 0

-------- RESULTADO FINAL -------
 PUNTO DE
LA RETICULA   SOLUCION
      0       0.00000
      1       0.09443
      2       0.15612
      3       0.19101
      4       0.20396
      5       0.19895
      6       0.17925
      7       0.14752
      8       0.10593
      9       0.05627
```

PROGRAMA 10-3 Método de la potencia inversa

A) Explicaciones

El PROGRAMA 10-3 resuelve un problema de valores propios mediante el método de la potencia inversa. Supondremos, para mayor facilidad, que en la ecuación (10.7.10).

$$A_i = -1, B_i = 2, C_i = -1 \text{ y } G_i = 1$$

para todos los puntos de la retícula. Supondremos que las condiciones en la frontera son $f_0 = f_{N+1} = 0$. Así, el sistema queda

$$\begin{aligned} 2f_1 - f_2 &= \lambda f_1 \\ -f_1 + 2f_2 - f_3 &= \lambda f_2 \\ -f_2 + 2f_3 - f_4 &= \lambda f_3 \\ &\vdots \\ -f_{N-1} + 2f_N &= \lambda f_N \end{aligned}$$

Las constantes, parámetros e iteraciones, F(I), se inicializan al principio. La estimación del valor propio se inicializa en uno. En el ciclo de iteración, FB(I) se hace igual a F(I) veces el valor propio estimado y se almacena para calcular la siguiente estimación del valor propio. Los coeficientes de la ecuación tridiagonal se hacen

iguales a los valores almacenados en cada ciclo, debido a que A(I), B(I), C(I) y D(I) cambian en la subrutina tridiagonal. Al regresar de dicha subrutina, la solución se guarda en el arreglo D(I). El valor anterior del valor propio se almacena en EB. Si no se cumple con el criterio de convergencia, el programa pasa al siguiente ciclo de iteración. Si se excede el límite del número de iteraciones o se cumple con el criterio de convergencia, la solución se normaliza e imprime.

B) Variables

A(I), B(I), C(I): coeficientes de las ecuaciones tridiagonales
D(I): G_i
AS(I), BS(I), CS(I), DS(I): almacenan a A(I), B(I), C(I) y D(I)
K: contador de iteraciones
N: número de incógnitas
EI: valor propio
EB: valor propio del ciclo anterior de iteración
IT: límite de los pasos de iteración
EP: criterio de convergencia
F(I): f_i
FB(I): valor anterior de f_i
S, SB: denominador y numerador, respectivamente, de la ecuación explicada en la sección 6

C) Listado

```
C       CSL/F10-3.FOR      METODO DE LA POTENCIA INVERSA
        DIMENSION A(20),B(20),C(20),D(20),AS(20),BS(20),CS(20),DS(20)
       &   ,F(20),FB(20)
        PRINT *
        PRINT *,'CSP/F10-3      METODO DE LA POTENCIA INVERSA'
140     K=0
        N=10
        EI=1
        S=1
        IT=30
        EP=0.0001
150     DO I=1, N
           AS(I)=-1.0
           BS(I)=2.0
           CS(I)=-1.0
           F(I)=1.0
           DS(I)=1.0
        END DO
170     PRINT *, '    IT. NO.      VALOR PROPIO
180     K=K+1
        DO I=1, N
           FB(I)=F(I)*EI
        END DO
```

```
      DO I=1, N
         A(I)=AS(I)
         B(I)=BS(I)
         C(I)=CS(I)
         D(I)=DS(I)*FB(I)
      END DO
      CALL TRDG(A,B,C,D,N)
220   SB=0
      S=0
      DO I=1, N
         F(I)=D(I)
         S=S+F(I)**2
         SB=SB+F(I)*FB(I)
      END DO
230   EB=EI
      EI=SB/S
      PRINT 245, K,EI
245   FORMAT('     ',I5, F10.6)
250   IF( ABS(1.0-EI/EB).LE.EP) GOTO 310
260   IF (K.LE.IT) GOTO 180
      PRINT *, ' SE HA EXCEDIDO EL LIMITE DE ITERACIONES'
C          NORMALIZACION DE LA SOLUCION
310   Z=0
      DO I=1, N
         IF (ABS(Z).LE.ABS(F(I))) Z=F(I)
      END DO
      DO I=1, N
         F(I)=F(I)/Z
      END DO
C-------------------------------------
350   EIGEN=EI
      PRINT 355,EIGEN
355   FORMAT('VALOR PROPIO=',F10.6)
360   PRINT *, '  I          F(I)'
      DO I=1, N
         PRINT 370, I,F(I)
      END DO
370   FORMAT(2X,I2,F14.6)
      PRINT *, ' ---------------------------------------'
390   PRINT*,' OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR'
      READ *,KSTOP
      IF (KSTOP.EQ.1) GOTO 140
      END
C*******************************
C     Favor de copiar la subrutina TRDG del programa 10-1.
```

D) Ejemplo de salida

```
CSP/F10-3      METODO DE LA POTENCIA INVERSA
   IT. NO.      VALOR PROPIO
        1  0.081967
        2  0.081026
        3  0.081014
        4  0.081014
VALOR PROPIO= 0.081014
  I          F(I)
  1       0.284692
  2       0.546290
  3       0.763594
```

```
 4        0.919018
 5        1.000000
 6        1.000000
 7        0.919018
 8        0.763594
 9        0.546290
10        0.284692
-------------------------------------
OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR
```

PROGRAMA 10-4 Método de la potencia inversa con desplazamiento

A) Explicaciones

El PROGRAMA 10-4 resuelve el mismo problema que el PROGRAMA 10-3 por medio del método de la potencia inversa con desplazamiento. Las variables y estructura del programa del primero son muy similares a las del segundo.

Sin embargo, hay dos cambios principales de la programación en el PROGRAMA 10-3. El primero es que el programa lee λ_e como entrada mediante el teclado. Después de obtener la solución iterativa para el valor de λ_e dado, el programa pide el siguiente λ_e para repetir la solución del mismo problema. El segundo cambio es que BS(I) se define de acuerdo con la ecuación (10.7.13).

Para determinar un valor propio y una función propia mayores por medio del método de la potencia inversa con desplazamiento, se hace un procedimiento de prueba y error, puesto que no existen las estimaciones de dichos valores propios. El método es el siguiente:

a) Se hace una cierta estimación de λ_e.
b) Se ejecuta el esquema de la potencia inversa con desplazamiento.
c) Se determina cuál de los valores propios y funciones propias se obtiene, contando el número de nodos en la función propia. Si la solución cambia de signo J veces en el dominio, entonces la función obtenida es la $J + 1$.
d) Si se desea una función propia menor a la obtenida en el paso b), entonces se prueba con una estimación menor de λ_e y se repiten los pasos b) y c). Si se desea una función propia mayor, se intenta con una estimación mayor de λ_e y se hace lo mismo.

B) Variables

Las mismas del PROGRAMA 10-3 excepto

ES: estimación del valor propio, λ_e
EI: $\delta\lambda$
EB: Valor anterior de $\delta\lambda$

C) Listado

```
C       CSL/F10-4.FOR      METODO DE LA POTENCIA INVERSA CON DESPLAZAMIENTO
        DIMENSION A(20),B(20),C(20),D(20),AS(20),BS(20),CS(20),DS(20)
       &    ,F(20),FB(20)
        PRINT *
        PRINT *,'CSP/F10-4  METODO DE LA POTENCIA INVERSA CON DESPLAZAMIENTO
140     K=0
        N=10
        EI=1
        S=1
        IT=30
        EP=0.0001
        PRINT *, '¿CUAL ES EL LAMBDA ESTIMADO?'
        READ *,ES
150     DO I=1, N
           AS(I)=-1.0
           BS(I)=2.0-ES
           CS(I)=-1.0
           F(I)=1.0
           DS(I)=1.0
        END DO
170     PRINT *, ' IT. NO.     DEL-LAMB'
180     K=K+1
        DO I=1, N
           FB(I)=F(I)*EI
        END DO
        DO I=1, N
           A(I)=AS(I)
           B(I)=BS(I)
           C(I)=CS(I)
           D(I)=DS(I)*FB(I)
        END DO
        CALL TRDG(A,B,C,D,N)
220     SB=0
        S=0
        DO I=1, N
           F(I)=D(I)
           S=S+F(I)**2
           SB=SB+F(I)*FB(I)
        END DO
230     EB=EI
        EI=SB/S
        PRINT 245, K,EI
245     FORMAT('     ',I5, F10.6)
250     IF( ABS(1.0-EI/EB).LE.EP) GOTO 310
260     IF (K.LE.IT) GOTO 180
        PRINT *, ' SE HA EXCEDIDO EL LIMITE DE LAS ITERACIONES'
C           NORMALIZACION DE LA SOLUCION
310     Z=0
        DO I=1, N
           IF (ABS(Z).LE.ABS(F(I))) Z=F(I)
        END DO
        DO I=1, N
           F(I)=F(I)/Z
        END DO
C-------------------------------------
350     EIGEN=EI+ES
        PRINT 355,EIGEN
```

```
355    FORMAT(' VALOR PROPIO=',F10.6)
360    PRINT *, '  I          F(I)'
       DO I=1, N
          PRINT 370, I,F(I)
       END DO
370    FORMAT(1X,I2,F14.6)
       PRINT *, ' ----------------------------------------'
390    PRINT*,' OPRIMA 1 PARA CONTINUAR, O 0 PARA TERMINAR '
       READ *,KSTOP
       IF (KSTOP.EQ.1) GOTO 140
       END
C*******************************
C      Favor de copiar la subrutina TRDG del programa 10-1.
```

D) Ejemplo de salida

```
CSP/F10-4      METODO DE LA POTENCIA INVERSA CON DESPLAZAMIENTO

 ¿CUAL ES EL LAMBDA ESTIMADO?
1.1
   IT. NO. DEL-LAMB
        1 -0.719871
        2 -0.488669
        3 -0.410787
        4 -0.403040
        5 -0.405550
        6 -0.407667
        7 -0.408513
        8 -0.400037
        9 -0.213611
       10  0.050254
       11  0.068609
       12  0.069154
       13  0.069169
       14  0.069170
VALOR PROPIO= 1.169170
  I          F(I)
 1       0.918816
 2       0.763320
 3      -0.284706
 4      -0.999906
 5      -0.546023
 6       0.546326
 7       1.000000
 8       0.284527
 9      -0.763650
10      -0.919069
---------------------------------------
¿CUAL ES EL LAMBDA ESTIMADO?
3.5
  IT. NO. DEL-LAMB
        1 -1.930510
        2  0.138717
        3  0.180737
        4  0.182382
        5  0.182498
        6  0.182506
```

```
VALOR PROPIO= 3.682506
  I          F(I)
 1        0.547176
 2       -0.919911
 3        1.000000
 4       -0.762820
 5        0.284214
 6        0.284216
 7       -0.762821
 8        0.999999
 9       -0.919910
10        0.547175
```

PROGRAMA 10-5 Doblamiento de una viga

A) Explicaciones

El PROGRAMA 10-5 resuelve la ecuación (10.9.10) mediante el esquema de descomposición LU.

Los datos para K, H y E se definen en las instrucciones DATA. Los valores de I' e I'' se calculan por medio de las aproximaciones por diferencias centrales para los puntos de la retícula, excepto para la frontera derecha, en la que se utiliza la aproximación por diferencias hacia atrás.

La matriz A se descompone en L y U mediante el algoritmo de la descomposición LU explicado en la sección 6.7 (véase también el PROGRAMA 6-3). Las matrices L y U se guardan en forma compacta en el arreglo L(I, J) cuando la subrutina concluye.

Al regresar de la subrutina, se comienza con la solución de la ecuación (10.9.10), utilizando las matrices L y U en los procesos de sustitución hacia adelante y hacia atrás.

B) Variables

A: matriz de coeficientes de la ecuación (10.9.10)

L: resultado de la descomposición LU en forma compacta (véase la sección 6.7)

P(I): carga vertical de la viga en el punto i de la retícula

FF(I): deflección de la viga en i

AA(I), BB(I), CC(I), DD(I), EE(I): a_i, b_i, c_i, d_i y e_i de la ecuación (10.9.10)

IM(J): momento de inercia de la sección transversal

LL(J): carga

K: número de puntos en la retícula sobre la viga, sin incluir los puntos hipotéticos

H: espaciamiento en la retícula

E: módulo de elasticidad

N: orden de la matriz, K + 2

I1(J), I2(J): I' e I'' en el punto J de la retícula, respectivamente

C) Listado

```
C     CSL/F10-5.FOR        DOBLAMIENTO DE UNA VIGA
      COMMON A(30,30),EL(30,30),P(30),N
      DIMENSION IM(0:10),I1(10),I2(10)
      DIMENSION AA(10),BB(10),CC(10),DD(10),FF(10),LL(10), EE(10)
      REAL IM,LL
      DATA K,H,E/ 5, 2, 150/
      DATA (IM(J),J=0,5)/1,1,1,1,1,1/
      DATA (LL(J),J=1,5)/2,2,2,2,2  /
      PRINT *,'CSL/F10-5 _DOBLAMIENTO DE UNA VIGA'
      N=K+2
      H2=H*H
      H3=H*H2
      H4=H3*H
      DO J=1, K-1
         I1(J)=(IM(J+1)-IM(J-1))/2/H
         I2(J)=(IM(J+1)-2*IM(J)+IM(J-1))/H/H
      END DO
      I1(K) =(3*IM(K)-4*IM(K-1)+IM(K-2))/2/H
      I2(K)=(2*IM(K)-5*IM(K-1)+4*IM(K-2)-IM(K-3))/H/H
      DO J=1, K
        AA(J)=E*IM(J)/H4-E*I1(J)*H3
        BB(J)=-4*E*IM(J)/H4+2*E*I1(J)/H3+E*I2(J)/H2
        CC(J)=6*E*IM(J)/H4 +2*E*I2(J)/H2
        DD(J)=-4*E*IM(J)/H4-2*E*I1(J)/H3+E*I2(J)/H2
        EE(J)=E*IM(J)/H4-E*I1(J)/H3
      END DO
      A(1,1)=CC(1)+AA(1)                    !  Matriz de coeficientes
      A(1,2)=DD(1)
      A(1,3)=EE(1)
      A(1,N+1)=LL(1)
      DO J=2, K
        IF (J.GT.2) A(J,J-2)=AA(J)
        A(J,J-1)=BB(J)
        A(J,J)=CC(J)
        A(J,J+1)=DD(J)
        A(J,J+2)=EE(J)
        A(J,N+1)=LL(J)
      END DO
      A(N-1,K-2)=-1
      A(N-1,K-1)=2
      A(N-1,K+1)=-2
      A(N-1,K+2)=1
      A(N,K-1)=1
      A(N,K)=-2
      A(N,K+1)=1
      PRINT *
      PRINT *,' Matriz de coeficientes aumentada'
      DO J=1, N
          PRINT 35, (A(J,L),L=1,N+1)
35        FORMAT(1x,1P8E10.3)
      END DO
C
      CALL LU                          !  Descomposición LU
C
      FF(1)=LL(1)/EL(1,1)              !  Sustitución hacia adelante
      DO J=2, N                        !    EL: Matrices L y U en forma compacta
        FF(J)=LL(J)                    !    FF: desplazamiento
        DO I=1, J-1
```

```
          FF(J)=FF(J)-FF(I)*EL(J,I)
        END DO
        FF(J)=FF(J)/EL(J,J)
      END DO
      DO J=N-1,1,-1                    ! Sustitución hacia atrás
        DO I=J+1, N
          FF(J)=FF(J)-EL(J,I)*FF(I)
        END DO
      END DO
      PRINT *
      PRINT *,' Solución'
      PRINT *,' Punto desplazamiento'
      DO J=1, K
        PRINT 200 , J,FF(J)
200     FORMAT(1x,I3, F12.6)
      END DO
      END
C*************************************************************
      SUBROUTINE LU                            ! Descomposición LU
      COMMON A(30,30),EL(30,30),P(30),N
      J=1
      DO I=1, N
        EL(I,1)=A(I,1)
      END DO
      DO J=2, N
        EL(1,J)=A(1,J)/EL(1,1)
      END DO
      DO J=2, N
        DO I=J, N
          S=0
          DO R=1, J-1
            S=S+EL(I,R)*EL(R,J)
          END DO
          EL(I,J)=A(I,J)-S
        END DO
        DO I=J+1, N
          S=0
          DO R=1, J-1
            S=S+EL(J,R)*EL(R,I)
          END DO
          EL(J,I)=(A(J,I)-S)/EL(J,J)
        END DO
      END DO
      PRINT *
      PRINT *,'Matrices LU en forma compacta
      DO I=1, N
          PRINT 20,(EL(I,J),J=1,N)
20        FORMAT(1x,1P8E10.3)
      END DO
      RETURN
      END
```

D) Ejemplo de salida

```
CSL/F10-5   DOBLAMIENTO DE UNA VIGA

Matriz de coeficientes aumentada
 6.563E+01-3.750E+01 9.375E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 2.000E+00
-3.750E+01 5.625E+01-3.750E+01 9.375E+00 0.000E+00 0.000E+00 0.000E+00 2.000E+00
 9.375E+00-3.750E+01 5.625E+01-3.750E+01 9.375E+00 0.000E+00 0.000E+00 2.000E+00
```

```
 0.000E+00 9.375E+00-3.750E+01 5.625E+01-3.750E+01 9.375E+00 0.000E+00 2.000E+00
 0.000E+00 0.000E+00 9.375E+00-3.750E+01 5.625E+01-3.750E+01 9.375E+00 2.000E+00
 0.000E+00 0.000E+00-1.000E+00 2.000E+00 0.000E+00-2.000E+00 1.000E+00 0.000E+00
 0.000E+00 0.000E+00 0.000E+00 1.000E+00-2.000E+00 1.000E+00 0.000E+00 0.000E+00

Matrices LU en forma compacta
 6.563E+01-5.714E-01 1.429E-01 0.000E+00 0.000E+00 0.000E+00 0.000E+00
-3.750E+01 3.482E+01-9.231E-01 2.692E-01 0.000E+00 0.000E+00 0.000E+00
 9.375E+00-3.214E+01 2.524E+01-1.143E+00 3.714E-01 0.000E+00 0.000E+00
 0.000E+00 9.375E+00-2.885E+01 2.076E+01-1.290E+00 4.516E-01 0.000E+00
 0.000E+00 0.000E+00 9.375E+00-2.679E+01 1.821E+01-1.395E+00 5.150E-01
 0.000E+00 0.000E+00-1.000E+00 8.571E-01 1.477E+00-3.256E-01-7.347E-01
 0.000E+00 0.000E+00 0.000E+00 1.000E+00-7.097E-01-4.419E-01 4.082E-02

Solución
Punto Desplazamiento
 1     1.333346
 2     4.373377
 3     8.373421
 4    12.800138
 5    17.333523
```

PROGRAMA 10-6 Vibración de una viga

A) Explicaciones

El PROGRAMA 10-6 resuelve las ecuaciones en diferencias de (10.9.13); se desarrolló al agregar el método de la potencia inversa con desplazamiento al PROGRAMA 10-5.

La primera mitad del programa principal es idéntica a la del PROGRAMA 10-5 excepto por las instrucciones DATA, en las que se añade la definición de los valores de MM(I). Antes de comenzar con la solución iterativa del valor propio y la función propia, la computadora pide una estimación del valor propio. La solución de las ecuaciones lineales en cada iteración se lleva a cabo mediante las sustituciones hacia adelante y hacia atrás con las matrices L y U. El método de prueba y error para calcular el valor propio deseado —y su correspondiente función propia— es el mismo que el del PROGRAMA 10-4. Por supuesto, si la estimación del valor propio es cero, la solución iterativa converge a la solución fundamental.

B) Variables

Las mismas del PROGRAMA 10-5, junto con

EIG: $\delta\lambda$
LE: estimación de λ
EB: valor anterior de $\delta\lambda$
LL(I): valores iterativos de y_i
FF(I): función propia en el i-ésimo punto de la retícula
MM(I): masa por unidad de longitud en el i-ésimo punto de la retícula

C) Listado

```
C     CSL/F10-6.FOR        VIBRACION DE UNA VIGA
      COMMON A(30,30),EL(30,30),P(30),N
      DIMENSION IM(0:10),I1(10),I2(10)
      DIMENSION AA(10),BB(10),CC(10),DD(10),FF(10),LL(10), EE(10)
      REAL IM,LL,LE
      DATA K,H,E/ 5, 2, 150/
      DATA (IM(J),J=0,5)/1,1,1,1,1,1/
      DATA (LL(J),J=1,5)/2,2,2,2,2  /
          REAL MM(10)
          DATA (MM(J),J=1,5)/1,1,1,1,1  /
      PRINT *,'CSL/F10-6  VIBRACION DE UNA VIGA'
      N=K+2
      H2=H*H
      H3=H*H2
      H4=H3*H
      DO J=1, K-1
         I1(J)=(IM(J+1)-IM(J-1))/2/H
         I2(J)=(IM(J+1)-2*IM(J)+IM(J-1))/H/H
      END DO
      I1(K) =(3*IM(K)-4*IM(K-1)+IM(K-2))/2/H
      I2(K)=(2*IM(K)-5*IM(K-1)+4*IM(K-2)-IM(K-3))/H/H
      DO J=1, K
        AA(J)=E*IM(J)/H4-E*I1(J)*H3
        BB(J)=-4*E*IM(J)/H4+2*E*I1(J)/H3+E*I2(J)/H2
        CC(J)=6*E*IM(J)/H4 +2*E*I2(J)/H2
        DD(J)=-4*E*IM(J)/H4-2*E*I1(J)/H3+E*I2(J)/H2
        EE(J)=E*IM(J)/H4-E*I1(J)/H3
      END DO
      A(1,1)=CC(1)+AA(1)
      A(1,2)=DD(1)
      A(1,3)=EE(1)
      A(1,N+1)=LL(1)
      DO J=2, K
        IF (J.GT.2) A(J,J-2)=AA(J)
        A(J,J-1)=BB(J)
        A(J,J)=CC(J)
        A(J,J+1)=DD(J)
        A(J,J+2)=EE(J)
        A(J,N+1)=LL(J)
      END DO
      A(N-1,K-2)=-1
      A(N-1,K-1)=2
      A(N-1,K+1)=-2
      A(N-1,K+2)=1
      A(N,K-1)=1
      A(N,K)=-2
      A(N,K+1)=1
C=========================================
      PRINT *, 'PROPORCIONE UNA ESTIMACION DEL VALOR PROPIO'
      READ *, LE
      DO J=1,K
        A(J,J)=A(J,J)-LE*MM(J)
      END DO
      CALL LU
      PRINT *,'   Iteración              Delta Lambda'
      DO IT=1,20                         !  Comienza la iteración
        EB=EIG
        FF(1)=LL(1)/EL(1,1)              !  Sustitución hacia adelante con LU
        DO J=2,N
```

```
            FF(J)=LL(J)
            DO I=1, J-1
              FF(J) = FF(J)-FF(I)*EL(J,I)
            END DO
            FF(J)=FF(J)/EL(J,J)
          END DO
          DO J=N-1,1,-1                    ! Sustitución hacia atrás
            DO I=J+1,N
              FF(J)=FF(J)-EL(J,I)*FF(I)
            END DO
          END DO
          SL=0                             ! Evaluación del valor propio
          SF=0
          DO J=1,K
            SL=SL+FF(J)*LL(J)
            SF=SF+FF(J)*FF(J)*MM(J)
          END DO
          EIG=SL/SF                        ! Delta Lambda
          DO J=1,K                         ! Se prepara LL para la siguiente iteración
            LL(J)=EIG*FF(J)*MM(J)
          END DO
          PRINT 14, IT, EIG
14        FORMAT (3X,I5,  5X, F12.7)
          IF(ABS(EB-EIG).LT.0.00001) GOTO 500
        END DO                             ! Fin del ciclo de iteración
500     PRINT *
        PRINT *,'VALOR PROPIO=' , EIG+LE
        PRINT *,' FUNCION PROPIA'
        PRINT *,' PUNTO FUNCION PROPIA'
        DO J=1,K
            PRINT 35, J, FF(J)
35          FORMAT(1x,I5,5X, 1PE15.6)
        END DO
        END
C*************************************************************
C    Favor de copiar la subrutina LU del programa 10-5.
```

D) Ejemplo de salida

(Caso 1)

```
CSL/F10-6   VIBRACION DE UNA VIGA
PROPORCIONE UNA ESTIMACION DEL VALOR PROPIO
0.5

  Iteración        Delta Lambda
      1            -0.2861729
      2            -0.3294247
      3            -0.3261672
      4            -0.3263999
      5            -0.3263844
      6            -0.3263853

VALOR PROPIO=   0.1736147
FUNCION PROPIA
PUNTO FUNCION PROPIA
    1          -5.634091E-01
    2          -1.947390E+00
    3          -3.856128E+00
    4          -6.029875E+00
    5          -8.280293E+00
```

(Caso 2)
CSL/F10-6 VIBRACION DE UNA VIGA
PROPORCIONE UNA ESTIMACION DEL VALOR PROPIO
4.0

Iteración	Delta Lambda
1	-0.4298595
2	1.3719629
3	1.2137848
4	1.3733426
5	1.3283689
6	1.3455634
7	1.3397084
8	1.3417829
9	1.3410584
10	1.3413121
11	1.3412225
12	1.3412534
13	1.3412426
14	1.3412471

VALOR PROPIO= 5.341247
FUNCION PROPIA

PUNTO No.	FUNCION PROPIA
1	-1.301901E-01
2	-2.667460E-01
3	-2.298271E-01
4	8.433461E-03
5	3.449624E-01

PROBLEMAS

10.1) Obtenga las ecuaciones en diferencias para $i = 1$ e $i = 10$ en el ejemplo 10.1, suponiendo que las condiciones en la frontera se modifican a $y'(1) = y(1)$ y $y'(10) = 0$.

10.2) Calcule las ecuaciones en diferencias para

$$-(p(x)\phi'(x))' + q(x)\phi(x) = S(x), \quad 0 < x < H$$
$$\phi'(0) = \phi(H) = 0$$

La geometría, retícula y constantes son:

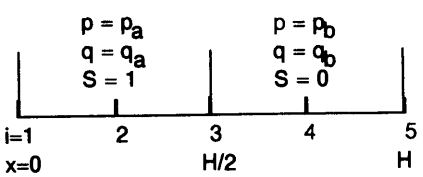


Figura P10.2

Los espacios de la retícula son $h = H/4$ para todos los intervalos.

10.3) Repita el problema anterior, suponiendo que el espaciamiento de la retícula es h_1 para los dos primeros intervalos y h_2 para los dos últimos.

10.4) La ecuación diferencial para un cable flexible de 50 metros de largo, que está fijo en sus dos extremos es

$$y''(x) = -w(x)/T \quad y(0) = y(50) = 0$$

donde x se mide en metros, $y(x)$ es el desplazamiento del cable medido desde el nivel de sus dos extremos (con sentido positivo hacia abajo), T es la componente horizontal de la tensión (5000 kg) y $w(x)$ es la distribución de carga, dada por

$$w(x) = 20[1 + \exp(x/25)] \text{ kg/m}$$

Determine la forma del cable. (Utilice 10 intervalos en la retícula.)

10.5) Consideremos un tubo de enfriamiento con una sección transversal y perímetro variables. Supongamos que la temperatura en cada sección transversal perpendicular al eje es uniforme, con lo que la temperatura en la dirección del eje es la solución de la ecuación

$$-[kA(x)T'(x)]' + P(x)h_cT(x) = P(x)h_cT_\infty$$

donde k es la conductividad térmica, $P(x)$ es el perímetro, $A(x)$ es el área seccional y T_∞ es la temperatura del ambiente. Las condiciones en la frontera son

$$T(0) = 100° \text{ C}$$
$$-kT'(H) = h_c(T(H) - T_\infty)$$

donde H es la longitud del tubo y h_c es el coeficiente de transferencia de calor por convección. Resuelva el problema anterior con las siguientes constantes:

$$h_c = 30 \text{ w/m}^2\text{K}$$
$$H = 0.1 \text{ m}, \quad k = 100 \text{ w/mK}, \quad T_\infty = 20° \text{ C}, \quad \text{y}$$
$$A(x) = (0.005)(0.05 - 0.25x) \text{ m}^2, \quad P(x) = A(x)/0.005 + 0.01 \text{ m}$$

(Utilice una retícula de 10 puntos.)

10.6) La condición en la frontera dada en la forma (10.2.8) es numéricamente equivalente a $\phi(0) = 0$ si hacemos g_l igual a 0 y f_l igual a un valor muy grande, como 10^{10}. ¿Cuáles valores de g_l y f_l hacen que la ecuación (10.2.8) sea equivalente a $\phi(0) = 2$?

10.7) Consideremos una celda cilíndrica de combustible para un reactor nuclear de agua ligera, el cual consiste del combustible y un moderador, como se muestra en la figura 10.7. El flujo térmico de los neutrones en la celda satisface la ecuación de difusión de los neutrones, dada por

$$-\frac{1}{r}\frac{d}{dr}Dr\frac{d}{dr}\phi(r) + \Sigma_a\phi(r) = S(r)$$

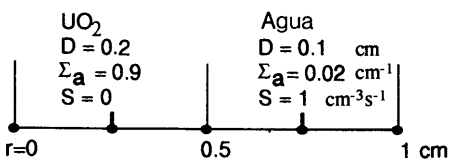


Figura P10.7 Un elemento de combustible

donde D es el coeficiente de difusión, Σ_a es la absorción de la sección transversal y S es la fuente de neutrones. Las constantes de UO_2 y H_2O aparecen en la figura. Las condiciones en la frontera son

$$\phi'(0) = \phi'(1) = 0$$

a) Utilice cinco puntos en la retícula para todo el dominio, con un intervalo constante de 0.25 cm y obtenga las ecuaciones en diferencias para cada uno de dichos puntos.

b) Resuelva las ecuaciones en diferencias obtenidas en a) mediante la solución tridiagonal.

10.8) Una viga de 3.5 m de longitud se encuentra apoyada en dos puntos, uno a 0.5 m y otro a 2.5 m del extremo izquierdo, como se muestra en la figura P10.8. Suponiendo que la varilla no tiene peso, la distribución de la carga en la viga está dada por

$$W(x) = (x - 0.5)\sqrt{1.5 - x} \quad \text{(N/m)}, \qquad \text{si } -0.5 < x < 1.5$$
$$= 0 \qquad\qquad\qquad\qquad\qquad\qquad \text{si } 1.5 < x$$

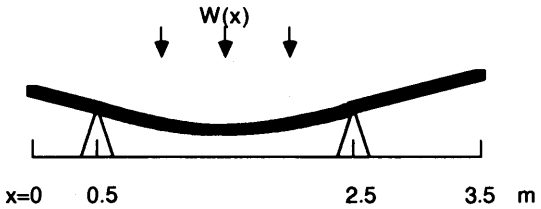


Figura P10.8

Las reacciones de los soportes en a y b están dadas por

$$R_a = \int_{-0.5}^{1.5} (x - x_b)w(x)\,dx/(x_a - x_b), \quad (N)$$
$$R_b = \int_{-0.5}^{1.5} (x - x_a)w(x)\,dx/(x_b - x_a), \quad (N)$$

a) El momento de doblamiento de la viga cumple

$$\frac{d}{dx} M(x) = -w(x) + R_a\delta(x - x_a) + R_b\delta(x - x_b)$$

donde $\delta(x)$ es la función delta y $M(x)$ es la distribución del momento de doblamiento. Las condiciones en la frontera son $M(0) = M(3.5) = 0$. Determine el momento de doblamiento, resolviendo la ecuación diferencial anterior mediante el método de diferencias finitas. (Disponga la retícula de modo que los puntos se encuentren en los soportes. Considere después las reacciones como fuerzas concentradas en los puntos de la retícula situados en los soportes. Las integrales de dichas reacciones se pueden evaluar mediante la regla extendida del trapecio o la regla extendida de 1/3 de Simpson.)

b) La deflección de la viga se relaciona con el momento mediante

$$EI \frac{d}{dx} y(x) = M(x)$$

Determine la deflección entre $x = 0.5$ y $x = 2.5$, resolviendo la ecuación mediante la aproximación por diferencias finitas. Suponga que $EI = 1000$ Nm^2.

10.9) Calcule la suma de todas las ecuaciones en (10.2.7a), suponiendo que $Ak = 1$ y explique cómo se puede obtener directamente el resultado mediante la ecuación (10.2.1).

10.10) Determine la suma de (10.4.7) para $i = j - 1$, $i = j$ e $i = j + 1$ y cancele todos los términos posibles, utilizando las definiciones de A_i, B_i, C_i y D_i. Explique el significado físico de la suma.

10.11) En un material laminar con un espesor de 0.2 cm, el lado izquierdo se encuentra perfectamente aislado, pero la temperatura de la superficie derecha está fija en 0° C. La lámina tiene una fuente de calor distribuido. La ecuación de la temperatura es $-T''(x) = q(x)/k$. Desarrolle un programa para calcular la distribución de la temperatura mediante una retícula de 10 puntos. Si la conductividad térmica es $k = 30$ w/m^{2}K, ejecute el programa con las siguientes distribuciones de la fuente de calor:

a) $q(x) = 200$ kw/m^3

b) $q(x) = 100 \exp(-10x)$ kw/m^3

Compare los resultados con sus correspondientes soluciones analíticas:

a) $T(x) = (10/3)(0.04 - x^2)$

b) $T(x) = 0.033(e^{-2} + 2 - 10x - e^{-10x})$

10.12) Considere la ecuación

$$-\phi''(r) - \phi'(r)/r = S(r), \quad a < r < b$$

para coordenadas cilíndricas, con las siguientes condiciones en la frontera

$$\phi'(a) = 0, \quad \phi'(b) = k(\phi_\infty - \phi(b))$$

a) Utilice una retícula con espaciamiento uniforme para obtener las ecuaciones en diferencias mediante la aproximación por diferencias centrales de 0″ y 0′ y muestre que el conjunto de ecuaciones en diferencias no es conservativa.

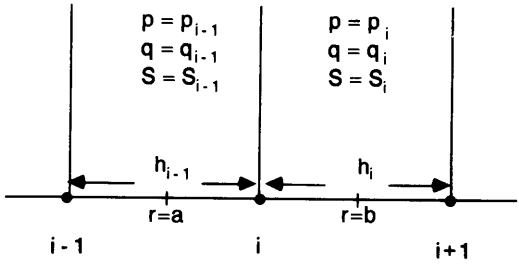
b) Determine las ecuaciones diferenciales en forma conservativa, reescribiendo la ecuación diferencial.

10.13) La ecuación de difusión para el caso de una geometría cilíndrica está dada por

$$-\frac{1}{r}[p(r)r\phi'(r)]' + q(r)\phi(r) = S(r)$$

Si consideramos los tres puntos de la retícula que se muestran en la figura P10.13, podemos obtener ecuaciones en diferencias al integrar la ecuación desde el punto medio entre $i - 1$ e i hasta el punto medio entre i e $i + 1$. Si los coeficientes son constantes (véase la figura P10.13) y el espaciamiento no es uniforme, determine las ecuaciones en diferencias, integrando el volumen entre a y b.

Figura P10.13



10.14) La ecuación para el desplazamiento de una membrana circular sometida a presión constante P es

$$y''(r) + \frac{1}{r} y'(r) = -P/T, \quad 0.2 \text{ m} \leqslant r \leqslant 0.5 \text{ m}$$

donde r es la coordenada radial, y el desplazamiento de la membrana (positivo hacia abajo), T es la tensión (400 kg/m) y $P = 800$ kg/m^2. Las condiciones en la frontera son $y(0.2) = y(0.5) = 0$. Determine el desplazamiento de la membrana, $y(r)$.

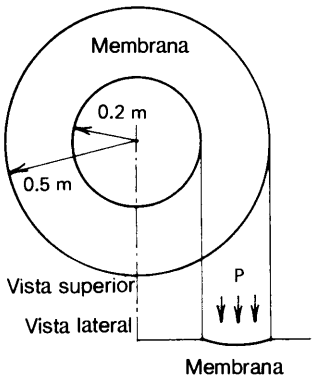


Figura P10.14 Una membrana bajo presión

10.15) Una radiación de rayos gamma se aplica de manera uniforme a la superficie de un material perfectamente esférico de radio 0.05 m. Los rayos penetran el material, el cual los absorbe después. Así, podemos suponer que la distribución de la fuente de calor debida a la radiación es

$$S(r) = 300 \exp [20(r - 0.05)]$$

donde r es el radio en metros y la unidad de S es W/m^3. La superficie de la esfera se expone al aire. El calor comienza a escapar hacia el aire circundante mediante convección, con un coeficiente de transferencia de calor de $h_c = 20$ W/m^{2}K. En el estado estacionario, la distribución de la temperatura es la solución de la ecuación

$$-\frac{1}{r^2}\frac{d}{dr} r^2 k \frac{d}{dr} T(r) = S(r)$$

con las condiciones en la frontera

$$T'(0) = 0$$
$$k' = h_c(T_\infty - T(R))$$

a) Escriba las ecuaciones en diferencias para la temperatura; utilice cuatro intervalos de igual longitud.

b) Resuelva las ecuaciones en diferencias mediante la solución tridiagonal ($T_\infty = 20\sigma$C).

10.16) Uno de los extremos de una lámina rectangular de enfriamiento con longitud $H = 0.1$ m se conecta con una fuente de calor, la cual se encuentra a 500° C. La lámina transfiere calor tanto por radiación como por convección hacia el ambiente, el cual tiene una temperatura de 20° C. Si tanto la lámina como el ambiente son cuerpos negros, la temperatura de

la lámina satisface la ecuación no lineal de difusión:

$$-AkT''(x) + Ph_c(T(x) - T_\infty) + P\sigma(T^4(x) - T_\infty^4) = 0$$

donde

k = 120 W/mk (conductividad térmica)
A = 1.5×10^{-4}m^2 (área de la sección transversal de la lámina)
P = 0.106 m (perímetro de la lámina)
h_c = 100 W/m^{2}K (coeficiente de transferencia de calor por convección)
σ 5.67×10^{-8} W/m^{2}K^4 (constante de Stefan-Boltzmann)
T_∞ = 293 K (temperatura del ambiente)

Las condiciones en la frontera son

$$T(0) = 500 + 273° \text{ K}$$
$$T'(H) = 0$$

donde se supone que el lado derecho del tubo está perfectamente aislado.

a) Obtenga la ecuación en diferencias para la ecuación diferencial anterior, con 10 intervalos de la misma longitud.

b) Resuelva la ecuación en diferencias por medio de la sustitución sucesiva.

c) Repita b) con el método de Newton.

10.17) Resuelva la siguiente ecuación con el método de Newton:

$$-\phi''(x) + [2 + \text{sen}(\phi(x))]\phi(x) = 2, \quad \phi(0) = \phi(2) = 0$$

Utilice 20 intervalos.

10.18) En un reactor químico, la densidad de un material está regida por la fórmula

$$-\phi''(x) + 0.1\phi'(x) = \exp(1 + 0.05\phi^2), \quad 0 < x < 2$$

Las condiciones en la frontera son $\phi(0) = 0$ y $\phi'(2) = 0$. Con 10 intervalos del mismo tamaño, resuelva la ecuación por: a) sustitución sucesiva y b) método de Newton.

10.19) El desplazamiento en la vibración con respecto a un eje de simetría de una membrana circular de radio 0.5 m es la solución de

$$y''(r) + \frac{1}{r}y'(r) = -\lambda y(r) \tag{A}$$

donde λ es el valor propio y las condiciones en la frontera son $y'(0) = 0$ y $y(0.5) = 0$. El significado físico del valor propio de la ecuación anterior es

$$\lambda = \omega^2\rho T = (2\pi\nu)^2\rho T$$

donde ω es la velocidad angular, ν es la frecuencia, ρ la masa por unidad de área de la membrana y T es la tensión.

a) Determine el valor propio fundamental de (A) mediante el método de potencias; use 11 puntos de la retícula incluyendo a los puntos de la frontera.

b) Repita a) con las condiciones en la frontera: $y(0) = y(0.5) = 0$.

10.20) Utilice el método de la potencia inversa con desplazamiento para determinar los tres primeros valores propios de los incisos a) y b) del problema (10.19).

10.21) Considere un reactor nuclear laminar, como el que se muestra en la figura P10.21. De acuerdo con el modelo mono-energético de neutrones, la distribución del flujo de neutrones en un reactor laminar crítico cumple la ecuación de difusión nuclear, dada por

$$-\frac{d}{dx} D \frac{d}{dx} \phi(x) + \Sigma_a \phi(x) = \lambda \Sigma_f \phi(x)$$

donde λ es el recíproco del factor de multiplicación real y además es un valor propio de la ecuación. Desarrolle un programa de cómputo para determinar el valor y la función propios fundamentales de la ecuación anterior, mediante el método de la potencia inversa con desplazamiento. Las condiciones en la frontera son $\phi'(0) = 0$ y $\phi(30) = 0$.

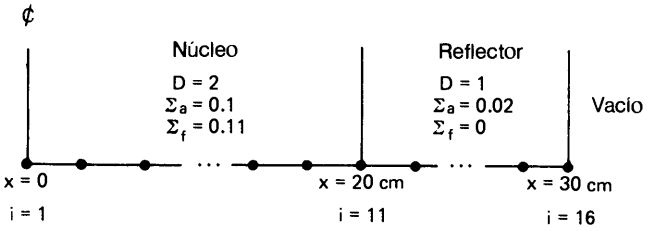


Figura P10.21 Un modelo de reactor nuclear

10.22) La ecuación mono-energética de difusión de neutrones para el caso de un reactor cilíndrico es

$$-\frac{1}{r}\frac{d}{dr} Dr \frac{d}{dr} \phi(r) + \Sigma_a \phi(r) = \lambda \Sigma_f \phi(r)$$

Resuelva esta ecuación, suponiendo que las constantes y dimensiones son las mismas que las del problema 10.21 (el valor x de la figura P10.21 se interpreta como r). Utilice las mismas condiciones en la frontera que usó en el problema 10.21.

10.23) En la figura P10.23 se muestra una viga de 1 m de longitud, la cual está sujeta a una fuerza axial P. La deflección de la varilla es la solución de la ecuación

$$EIy''(x) = M$$

donde E es el módulo de elasticidad. I es el momento de inercia de la sección transversal de la varilla y M es el momento de doblamiento; puesto que este último debe ser igual a $-Py$, la ecuación anterior se transforma en

$$y''(x) = \frac{-P}{EI(x)} y(x), \quad y(0) = y(H) = 0$$

Esta ecuación es un problema de valores propios, puesto que las soluciones sólo existen cuando P toma ciertos valores discretos. Las constantes son

$$I(x) = 6 \times 10^{-5}(2 - 0.1x)\ \text{m}^4$$
$$E = 200 \times 10^9\ \text{Pa}$$
$$H = 1\ \text{m}$$

Calcule el valor propio fundamental P, que corresponde a la deflección ilustrada en la figura P10.23a, así como el siguiente valor propio correspondiente a la deflección que se muestra en la figura 10.23b. (Es usual que el mínimo valor propio sea el de mayor interés, puesto que la configuración de los demás valores propios —como el de la figura 10.23b— es inestable; además, no se puede obtener por medios experimentales, a menos de que se utilicen dispositivos adicionales especiales.) La unidad de P es 1000 Newtons.

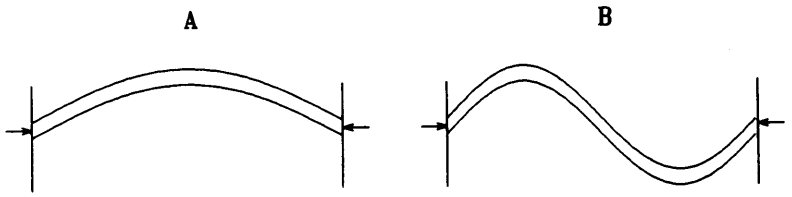


Figura P10.23 Deflección de una viga

10.24) El problema de valores propios del conjunto de ecuaciones en diferencias

$$-\phi_{i-1} + 2\phi_i - \phi_{i+1} = \lambda\phi_i$$

con condiciones en la frontera

$$\phi_0 = \phi_{N+1} = 0$$

tiene N valores propios y sus correspondientes funciones propias. Muestre que éstas se pueden expresar analíticamente como:

$$\phi_i = \operatorname{sen}(\alpha i), \quad \alpha = (\pi m)/(N-1)$$
$$\lambda = 2(1 - \cos(\alpha))$$

donde $m = 1, 2, \ldots, N$.

Sugerencia: sustituya el valor propio y su correspondiente función propia en la ecuación en diferencias y utilice el teorema de la suma de la función seno:

$$\operatorname{sen}(a + b) = \operatorname{sen}(a)\cos(b) + \cos(a)\operatorname{sen}(b)$$
$$\operatorname{sen}(a - b) = \operatorname{sen}(a)\cos(b) - \cos(a)\operatorname{sen}(b)$$

10.25) Se puede estimar el radio de dominancia de la iteración de la potencia inversa dado por la ecuación (10.8.5), si calculamos en cada ciclo de la iteración el número

$$r = \sqrt{\frac{\lambda^{(t)} - \lambda^{(t-1)}}{\lambda^{(t-1)} - \lambda^{(t-2)}}}$$

Demuestre que r converge al radio de dominancia.

10.26) El método de la potencia inversa dado por las ecuaciones (10.7.10) y (10.7.11) se puede modificar:

$$A_i f_{i-1}^{(t)} + B_i f_i^{(t)} + C_i f_{i+1}^{(t)} = y_i^{(t-1)}$$

$$\lambda^{(t)} = \frac{\sum_i y_i^{(t-1)} f_i^{(t)}}{\sum_i G_i [f_i^{(t)}]^2}$$

donde

$$y_i^{(t-1)} = G_i[\theta\lambda^{(t-1)}f_i^{(t-1)} + (1-\theta)\lambda^{(t-2)}f_i^{(t-2)}]$$

y θ es un parámetro de extrapolación que satisface la condición $1 < \theta < 2$. Demuestre que si θ está entre 1 y 2, el esquema anterior converge más rápido que con el método de la potencia inversa.

BIBLIOGRAFIA

Duderstadt, J. J. y L. J. Hamilton, *Nuclear Reactor Analysis*, Wiley, 1976.

Dusinberre, G. M., *Heat Transfer Calculation by Finite Differences*, International Textbook, 1961.

Eckert, E. R. G., *Heat and Mass Transfer*, McGraw-Hill, 1959.

Habib, I. S., *Engineering Analysis Methods*, Lexington Books, 1975.

Hall, G. y J. M. Watt, editores, *Modern Numerical Methods for Ordinary Differential Equations*, Clarendon Press, 1976.

Incorpera, F. P. y D. P. Dewitt, *Fundamentals of Heat Transfer*, Wiley, 1981.

Kreith, F. y W. Z. Black, *Basic Heat Transfer*, Harper & Row, 1980.

Kubicek, M. y V. Hlavacek, *Numerical Solution of Nonlinear Boundary Value Problems with Applications*, Prentice-Hall, 1983.

Nakamura, S., *Computational Methods in Engineering and Science with Applications to Fluid Dynamics and Nuclear Systems*, Krieger, 1986.

Nishida, T., M. Miura y H. Fujii, editores, *Patterns and Waves*, North-Holland, 1986.

Smith, G. D., *Numerical Solution of Partial Differential Equations*, Oxford University Press, 1978.

Thomson, W. T., *Theory of Vibrations*, Prentice-Hall, 1981.

11

Ecuaciones diferenciales parciales elípticas

11.1 INTRODUCCION

Las ecuaciones diferenciales parciales (EDP) de segundo orden se pueden clasificar en tres tipos: 1) parabólicas, 2) elípticas y 3) hiperbólicas [Myint-U/Debnath].

Para distinguir a las ecuaciones diferenciales parciales elípticas de los otros dos tipos, consideremos la siguiente forma general de una EDP de segundo orden en dos variables:

$$A\frac{\partial^2\phi}{\partial x^2} + B\frac{\partial^2\phi}{\partial x\,\partial y} + C\frac{\partial^2\phi}{\partial y^2} + D\frac{\partial\phi}{\partial x} + E\frac{\partial\phi}{\partial y} + F\phi = S \tag{11.1.1}$$

donde x y y son variables independientes y A, B, C, D, E, F y S son funciones dadas de x y y. La ecuación anterior es de alguno de los tres tipos, según las condiciones siguientes:

$$\begin{array}{ll} \text{Parabólica} & \text{si } B^2 - 4AC = 0 \\ \text{Elíptica} & \text{si } B^2 - 4AC < 0 \\ \text{Hiperbólica} & \text{si } B^2 - 4AC > 0 \end{array} \tag{11.1.1a}$$

Las EDP elípticas aparecen en problemas estacionarios de dos y tres dimensiones. Entre los problemas elípticos típicos están la conducción del calor en los sólidos, la difusión de partículas y la vibración de una membrana, entre otros. Estas ecuaciones tienen una relación cercana con las de tipo parabólico. Por ejemplo, al

resolver una EDP parabólica, es frecuente el uso de los métodos numéricos para una EDP elíptica como parte del esquema de solución. Las EDP elípticas se pueden considerar como la contraparte de estado estacionario de las EDP parabólicas. Las ecuaciones de Poisson y Laplace son casos especiales de las EDP elípticas.

El objetivo fundamental de este capítulo es el estudio de los métodos de diferencias finitas para la solución de las EDP elípticas que se puedan escribir en la siguiente forma general:

$$-\nabla p(x, y)\nabla\phi(x, y) + q(x, y)\phi(x, y) = S(x, y) \tag{11.1.2}$$

donde p, q y S son funciones dadas y $q \geqslant 0$. (Si el valor de q es negativo en el problema en cuestión, podría ocurrir que los métodos de solución descritos en este capítulo no fueran aplicables.) Cuando $p = 1$ y $q = 0$ (11.1.2) se transforma en una ecuación de Poisson o de Laplace:

$$\text{Ecuación de Poisson:} \quad -\nabla^2\phi(x, y) = S(x, y)$$

$$\text{Ecuación de Laplace:} \quad -\nabla^2\phi(x, y) = 0$$

Una EDP elíptica puede incluir términos de primer orden, como

$$-\nabla p\, \nabla\phi + u(x, y)\frac{\partial}{\partial x}\phi + v(x, y)\frac{\partial}{\partial y}\phi + q\phi = S \tag{11.1.3}$$

donde u y v son funciones dadas. En dinámica de fluidos, el segundo y tercer términos se llaman *términos advectivos*. Si éstos dominan al primer término, la ecuación tiene un comportamiento más parecido al de una EDP hiperbólica, por lo que deberán aplicarse los métodos para este último caso.

Los métodos de solución numérica para las EDP elípticas se clasifican globalmente en dos categorías: a) métodos de diferencias finitas y b) métodos de elemento finito. Los primeros, tema principal de este capítulo, se obtienen a partir de una retícula rectangular y tienen la gran ventaja de que se disponen de numerosos métodos de solución. La ventaja de los segundos es que se puede determinar la ecuación discreta para casi toda geometría. Por lo tanto, es frecuente elegir el método de elemento finito cuando el problema trata de una geometría complicada. Sin embargo, en años recientes, se han podido resolver problemas geométricamente difíciles mediante el método de diferencias finitas y una transformación de coordenadas [Thompson; Thompson, Wasri y Mastin]. Por una transformación de coordenadas queremos entender la transformación matemática de cierta geometría no rectan-

Tabla 11.1 Breve comparación entre los métodos de diferencias finitas y los métodos de elemento finito

	Ventajas	Desventajas
Métodos de diferencias finitas	Se dispone de numerosos métodos de solución eficientes. Fáciles de vectorizar.	Menos adaptables a una geometría curva que los métodos de elemento finito.
Métodos de elemento finito	Fáciles de adaptar a una geometría curva.	Los algoritmos de solución son limitados y menos eficientes que los métodos de diferencias finitas.

gular en una coordenada rectangular que haga más fáciles los cálculos. Con esta transformación, se puede utilizar el método de diferencias finitas sobre una retícula rectangular en las coordenadas computacionales.

Sin embargo, en el resto de este capítulo nos centraremos en los métodos de diferencias finitas para las EDP elípticas de la forma de la ecuación (11.1.2).

11.2 ECUACIONES EN DIFERENCIAS

Esta sección consta de cuatro subsecciones. Las dos primeras analizan las aproximaciones por diferencias para las geometrías rectangulares y curvas, respectivamente. La tercera hace una descripción de la obtención de ecuaciones en diferencias mediante el método de integración. La cuarta resume las propiedades de las aproximaciones por diferencias.

11.2.1 Aproximaciones por diferencias para las geometrías rectangulares

En esta sección obtendremos las ecuaciones en diferencias finitas para la ecuación de Poisson en las coordenadas cartesianas rectangulares:

$$-\nabla^2\phi(x, y) = S(x, y) \tag{11.2.1}$$

o, en forma equivalente,

$$-\frac{\partial^2\phi(x, y)}{\partial x^2} - \frac{\partial^2\phi(x, y)}{\partial y^2} = S(x, y) \tag{11.2.2}$$

donde $S(x, y)$ es una función dada, la cual recibe el nombre de término no homogéneo (o término fuente).

Para hacer más sencilla la exposición, consideraremos el dominio definido por [véase la figura 11.1a)]

$$0 \leqslant x \leqslant x_{\text{máx}}, \quad 0 \leqslant y \leqslant y_{\text{máx}}$$

Supondremos que las condiciones en la frontera son las siguientes:

$$\begin{aligned}
&\text{frontera izquierda} && \frac{\partial\phi}{\partial x} = 0 && \text{(tipo Neumann)}\\
&\text{frontera derecha} && \phi = 0 && \text{(tipo Dirichlet)}\\
&\text{frontera inferior} && \frac{\partial\phi}{\partial y} = 0 && \\
&\text{frontera superior} && \phi = 0 &&
\end{aligned} \tag{11.2.3}$$

Para obtener las ecuaciones en diferencias finitas, disponemos una retícula con intervalos espaciados de manera uniforme en el dominio rectangular, como lo

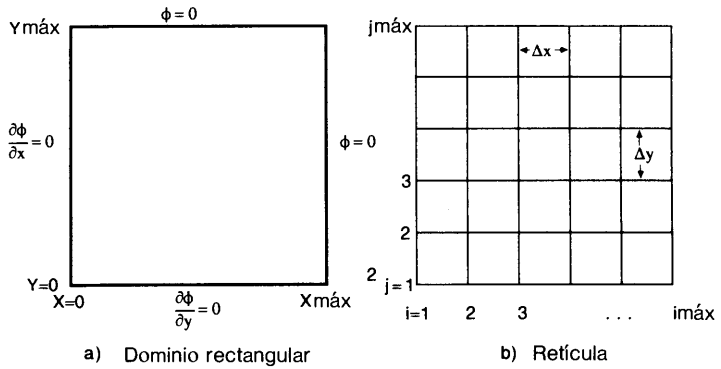


Figura 11.1 Dominio rectangular y una retícula

muestra la figura 11.1b). La longitud de los intervalos en las direcciones de x y y se denotan por ∇x y ∇y, respectivamente. Los puntos de la retícula se numeran por i y j, donde i es el índice de la retícula en la dirección de x y j es su análogo en la dirección de y.

Obtenemos la ecuación en diferencias para un punto (i, j) de la retícula situado dentro de la frontera, si consideramos ese punto y los otros cuatro que lo rodean, como en la figura 11.2a). Si aplicamos la aproximación por diferencias centrales, aproximamos el primer término de la ecuación (11.2.2) por

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{\phi_{i-1,j} - 2\phi_{i,j} + \phi_{i+1,j}}{\Delta x^2} \tag{11.2.4}$$

De manera análoga, la aproximación por diferencias del segundo término es

$$\frac{\partial^2 \phi}{\partial y^2} = \frac{\phi_{i,j-1} - 2\phi_{i,j} + \phi_{i,j+1}}{\Delta y^2} \tag{11.2.5}$$

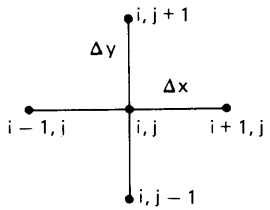
Sustituimos (11.2.4) y (11.2.5) en (11.2.2) para obtener

$$\frac{-\phi_{i-1,j} + 2\phi_{i,j} - \phi_{i+1,j}}{\Delta x^2} + \frac{-\phi_{i,j-1} + 2\phi_{i,j} - \phi_{i,j+1}}{\Delta y^2} = S_{i,j} \tag{11.2.6}$$

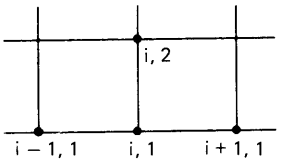
donde $S_{i,j} = S(x_i, y_j)$. La ecuación (11.2.6) se aplica a todos los puntos de la retícula excepto los de la frontera.

Las ecuaciones en diferencias para los puntos de la retícula que se encuentran sobre la frontera requieren un tratamiento especial debido a que: a) el número de puntos vecinos es menor que cuatro, y b) deben tomarse en cuenta las condiciones en la frontera. Sin embargo, para esta geometría en particular, no se necesitan las ecuaciones en diferencias para los puntos de las fronteras derecha y superior, puesto que se conocen los valores de ϕ ($\phi = 0$), a partir de las condiciones en la frontera dadas en (11.2.3).

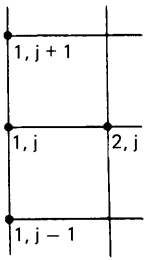
Si consideramos la frontera inferior [véase la figura 11.2b)], podemos obtener la ecuación en diferencias para un punto, $1 < i < i_{\text{máx}}$ y $j = 1$, de la siguiente for-



a) Se utilizan cinco puntos de la retícula en la ecuación en diferencias para un punto interior.



b) Se utilizan cuatro puntos de la retícula en la ecuación en diferencias para un punto (i, 1) que se encuentra en la frontera inferior.



c) Se utilizan cuatro puntos de la retícula en la ecuación en diferencias para un punto (1, j) que se encuentra en la frontera izquierda.

Figura 11.2 Puntos de una retícula que se utilizan en las ecuaciones en diferencias

ma: aproximamos el primer término de (11.2.2) mediante (11.2.4), mientras que el segundo término lo aproximamos por

$$\left(\frac{\partial^2 \phi}{\partial y^2}\right)_{i,1} = \frac{\left(\frac{\partial \phi}{\partial y}\right)_{i,1+\frac{1}{2}} - \left(\frac{\partial \phi}{\partial y}\right)_{i,1}}{\frac{\Delta y}{2}} \tag{11.2.7}$$

Aproximamos el primer término de (11.2.7) mediante la aproximación por diferencias centrales:

$$\left(\frac{\partial \phi}{\partial y}\right)_{i,1+\frac{1}{2}} = \frac{\phi_{i,2} - \phi_{i,1}}{\Delta y} \tag{11.2.8}$$

La condición sobre la frontera inferior dada en (11.2.3) muestra que el segundo término del numerador del lado derecho de (11.2.7) se anula. Por lo tanto, la ecuación (11.2.7) se transforma en

$$\left(\frac{\partial^2 \phi}{\partial y^2}\right)_{i,1} = \frac{2\phi_{i,2} - 2\phi_{i,1}}{\Delta y^2} \tag{11.2.9}$$

Así, al sustituir (11.2.4) y (11.2.9) en la ecuación (11.2.2), se obtiene la ecuación en diferencias para un punto de la frontera inferior:

$$\frac{-\phi_{i-1,1} + 2\phi_{i,1} - \phi_{i+1,1}}{\Delta x^2} + \frac{-2\phi_{i,2} + 2\phi_{i,1}}{\Delta y^2} = S_{i,1} \tag{11.2.10}$$

Para un punto $i = 1$ y $1 < j < j_{\text{máx}}$ en la frontera izquierda [véase la figura 11.2c)], aproximamos el primer término de (11.2.2) por

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{\left(\frac{\partial \phi}{\partial x}\right)_{1+\frac{1}{2},j} - \left(\frac{\partial \phi}{\partial x}\right)_{1,j}}{\frac{\Delta x}{2}}$$

$$= \frac{2\phi_{2,j} - 2\phi_{1,j}}{\Delta x^2} \tag{11.2.11}$$

en donde se utiliza la condición en la frontera izquierda dada por la ecuación (11.2.3), con el fin de eliminar $\partial\phi/\partial x)_{1,j}$, y la aproximación por diferencias centrales se utiliza para el término $(\partial\phi/\partial x)_{1+\frac{1}{2},j}$. Si sustituimos (11.2.11) y (11.2.5) en la ecuación (11.2.2), la ecuación en diferencias es

$$\frac{2\phi_{1,j} - 2\phi_{2,j}}{\Delta x^2} + \frac{-\phi_{1,j-1} + 2\phi_{1,j} - \phi_{1,j+1}}{\Delta y^2} = S_{1,j} \tag{11.2.12}$$

Para el punto de la esquina en $i = j = 1$, aproximamos cada término del lado izquierdo de (11.2.2) por la ecuación (11.2.9) y (11.2.11), respectivamente. Por lo tanto, la ecuación en diferencias se convierte en

$$\frac{2\phi_{1,1} - 2\phi_{2,1}}{\Delta x^2} + \frac{2\phi_{1,1} - 2\phi_{1,2}}{\Delta y^2} = S_{1,1} \tag{11.2.13}$$

Ejemplo 11.1

a) Escriba la aproximación por diferencias de la ecuación de Poisson, para la retícula que se muestra en la figura E11.1.

b) Exprese las ecuaciones en diferencias mediante matrices y vectores.

c) Muestre que la matriz de coeficientes obtenida en b) se puede transformar a una forma simétrica si dividimos o multiplicamos cada renglón por una constante.

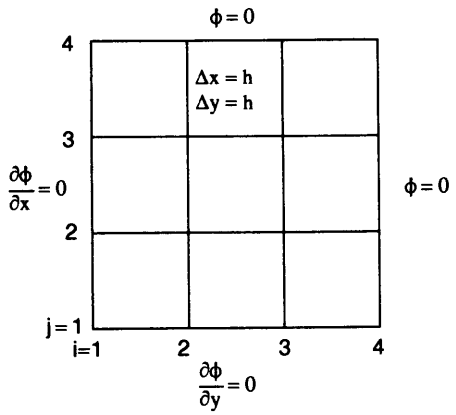


Figura E11.1

(Solución)

a) Puesto que la retícula tiene una separación uniforme en ambas direcciones, las ecuaciones en diferencias se pueden escribir como

$$
\begin{aligned}
4\phi_{1,1} - 2\phi_{2,1} - 2\phi_{1,2} &= h^2 S_{1,1} \\
4\phi_{2,1} - \phi_{1,1} - \phi_{3,1} - 2\phi_{2,2} &= h^2 S_{2,1} \\
4\phi_{3,1} - \phi_{2,1} - 2\phi_{3,2} &= h^2 S_{3,1} \\
4\phi_{1,2} - 2\phi_{2,2} - \phi_{1,1} - \phi_{1,3} &= h^2 S_{1,2} \\
4\phi_{2,2} - \phi_{1,2} - \phi_{3,2} - \phi_{2,1} - \phi_{2,3} &= h^2 S_{2,2} \\
4\phi_{3,2} - \phi_{2,2} - \phi_{3,1} - \phi_{3,3} &= h^2 S_{3,2} \\
4\phi_{1,3} - 2\phi_{2,3} - \phi_{1,2} &= h^2 S_{1,3} \\
4\phi_{2,3} - \phi_{1,3} - \phi_{3,3} - \phi_{2,2} &= h^3 S_{2,3} \\
4\phi_{3,3} - \phi_{3,2} - \phi_{2,3} &= h^2 S_{3,3}
\end{aligned}
\tag{A}
$$

donde se usó que $\phi_{4,1} = \phi_{4,2} = \phi_{4,3} = \phi_{4,4} = \phi_{1,4} = \phi_{2,4} = \phi_{3,4} = 0$

b) En notación matricial, las ecuaciones anteriores dadas en (A) se escriben de la manera siguiente:

$$
\left[\begin{array}{ccc|ccc|ccc}
4 & -2 & 0 & -2 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & 0 & -2 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & 0 & 0 & -2 & 0 & 0 & 0 \\
\hline
-1 & 0 & 0 & 4 & -2 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
\hline
0 & 0 & 0 & -1 & 0 & 0 & 4 & -2 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4
\end{array}\right]
\begin{bmatrix} \phi_{1,1} \\ \phi_{2,1} \\ \phi_{3,1} \\ \phi_{1,2} \\ \phi_{2,2} \\ \phi_{3,2} \\ \phi_{1,3} \\ \phi_{2,3} \\ \phi_{3,3} \end{bmatrix}
=
\begin{bmatrix} h^2 S_{1,1} \\ h^2 S_{2,1} \\ h^2 S_{3,1} \\ h^2 S_{1,2} \\ h^2 S_{2,2} \\ h^2 S_{3,2} \\ h^2 S_{1,3} \\ h^2 S_{2,3} \\ h^2 S_{3,3} \end{bmatrix}
\tag{B}
$$

c) La matriz de coeficientes se puede transformar en una forma simétrica al dividir la primera ecuación entre **4** y dividiendo también las ecuaciones segunda, tercera, cuarta y séptima entre 2:

$$\left[\begin{array}{ccc|ccc|ccc}
1 & -\frac{1}{2} & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\
-\frac{1}{2} & 2 & -\frac{1}{2} & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -\frac{1}{2} & 2 & 0 & 0 & -1 & 0 & 0 & 0 \\
\hline
-\frac{1}{2} & 0 & 0 & 2 & -1 & 0 & -\frac{1}{2} & 0 & 0 \\
0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
\hline
0 & 0 & 0 & -\frac{1}{2} & 0 & 0 & 2 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4
\end{array}\right]
\begin{bmatrix} \phi_{1,1} \\ \phi_{2,1} \\ \phi_{3,1} \\ \phi_{1,2} \\ \phi_{2,2} \\ \phi_{3,2} \\ \phi_{1,3} \\ \phi_{2,3} \\ \phi_{3,3} \end{bmatrix}
=
\begin{bmatrix} \frac{1}{4}h^2S_{1,1} \\ \frac{1}{2}h^2S_{2,1} \\ \frac{1}{2}h^2S_{3,1} \\ \frac{1}{2}h^2S_{1,2} \\ h^2S_{2,2} \\ h^2S_{3,2} \\ \frac{1}{2}h^2S_{1,3} \\ h^2S_{2,3} \\ h^2S_{3,3} \end{bmatrix}
\qquad \text{(C)}$$

Nota: como lo muestra la ecuación (C), la matriz de coeficientes de las ecuaciones en diferencias para el caso de la ecuación de Poisson sobre una retícula rectangular tiene las propiedades siguientes:

a) La matriz es tridiagonal por bloques.
b) Los bloques diagonales son submatrices tridiagonales.
c) Los bloques por fuera de la diagonal, pero adyacentes a los bloques en la diagonal, son submatrices diagonales cuyos elementos sobre la diagonal son negativos.
d) Los demás bloques son submatrices nulas.
e) La matriz completa es simétrica.

El número de elementos nulos crece rápidamente junto con el número total de puntos en la retícula. Si el espacio de memoria de la máquina es limitado, los métodos de solución iterativa son útiles en el caso de las EDP elípticas, puesto que almacenan sólo los elementos no nulos de la matriz, de forma que se necesita mucho menos espacio de memoria que en el método de solución directa.

Las condiciones en la frontera se dan a menudo de la siguiente forma general:

$$\frac{\partial \phi}{\partial n} + \alpha\phi = \beta \quad \text{(tipo mixto)} \tag{11.2.14}$$

donde α y β son constantes y $\partial/\partial n$ es la derivada normal hacia afuera de la frontera. En un dominio rectangular, $\partial/\partial n$ tiene la siguiente interpretación para cada frontera:

$$\begin{aligned}
\frac{\partial}{\partial n} &= -\frac{\partial}{\partial x} \text{ para la frontera izquierda} \\
\frac{\partial}{\partial n} &= \frac{\partial}{\partial y} \text{ para la frontera superior} \\
\frac{\partial}{\partial n} &= \frac{\partial}{\partial x} \text{ para la frontera derecha} \\
\frac{\partial}{\partial n} &= -\frac{\partial}{\partial y} \text{ para la frontera inferior}
\end{aligned} \tag{11.2.15}$$

La implantación de las condiciones en la frontera dadas en la forma de la ecuación (11.2.14) es similar a la de $\partial\phi/\partial x = 0$ o bien la de $\partial\phi/\partial y = 0$. Por ejemplo, si la condición en la frontera superior está dada en la forma de (11.2.14), aproximamos el segundo término de (11.2.2) como

$$\left(\frac{\partial^2\phi}{\partial y^2}\right)_{i,J} = \frac{(\partial\phi/\partial y)_{i,J} - (\partial\phi/\partial y)_{i,J-1/2}}{\Delta y/2}$$

$$= \frac{(-\alpha\phi_{i,J} + \beta) - (\phi_{i,J} - \phi_{i,J-1})/\Delta y}{\Delta y/2}$$

$$= \frac{-2\alpha\Delta y\phi_{i,J} + 2\beta\Delta y - 2\phi_{i,J} + 2\phi_{i,J-1}}{\Delta y^2} \tag{11.2.16}$$

Sustituimos (11.2.16) y (11.2.4) en (11.2.2) para obtener

$$\frac{-\phi_{i-1,J} + 2\phi_{i,J} - \phi_{i+1,J}}{\Delta x^2} + \frac{(2\alpha\Delta y + 2)\phi_{i,J} - 2\phi_{i,J-1}}{\Delta y^2} = S_{i,J} + \frac{2\beta}{\Delta y} \tag{11.2.17}$$

La ecuación (11.2.14) es una forma universal de las condiciones en la frontera, ya que los tres tipos de condiciones (de Dirichlet, Neumann y mixto) se pueden representar de esa manera. Si $\alpha = 0$, se reduce a la condición de frontera de Neumann (con derivada), $\partial\phi/\partial n = \beta$. Por otro lado, si cambiamos β por $\gamma\alpha$ y α crece a infinito, entonces se reduce el tipo de Dirichlet (con un valor fijo), $\phi = \gamma$ (constante).

Aunque no se permite el concepto de infinito en un programa de computadora, se alcanza prácticamente el mismo efecto si α toma un valor muy grande, tal como 10^{10}. La ventaja del uso de esta forma es que, una vez escrito el programa, se puede cambiar con facilidad el tipo de condición en la frontera, simplemente revisando los parámetros α y β para cada frontera.

Ejemplo 11.2

Con la geometría y retícula que aparecen en la figura E11.2, determine las ecuaciones en diferencias para la ecuación de Poisson:

$$-\nabla^2\phi = S \tag{A}$$

Las condiciones en la frontera son

$\dfrac{\partial\phi}{\partial x} = \phi$ para la frontera izquierda

$\dfrac{\partial\phi}{\partial y} = \phi - 2$ para la frontera inferior

$\phi = 5$ para la frontera derecha

$\phi = 7$ para la frontera superior

Los intervalos en la retícula son unitarios en ambas direcciones.

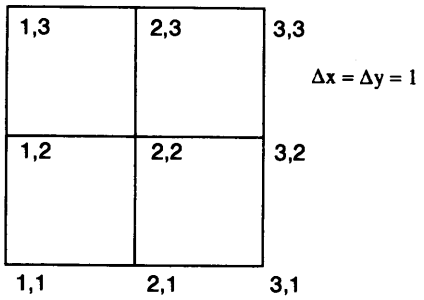


Figura E11.2

(Solución)

Puesto que las condiciones en las fronteras superior e inferior son del tipo de valor fijo, obtenemos las ecuaciones en diferencias sólo para los siguientes cuatro puntos de la retícula: (1, 1), (2, 1), (1, 2) y (2, 2).

Punto (1, 1). Aproximamos la derivada parcial con respecto a x por

$$\left(\frac{\partial^2\phi}{\partial x^2}\right)_{1,1} = \frac{\left(\frac{\partial\phi}{\partial x}\right)_{1+\frac{1}{2},1} - \left(\frac{\partial\phi}{\partial x}\right)_{1,1}}{\frac{1}{2}}$$

$$= \frac{(\phi_{2,1} - \phi_{1,1}) - \phi_{1,1}}{\frac{1}{2}}$$

$$= -4\phi_{1,1} + 2\phi_{2,1} \qquad \text{(B)}$$

en donde utilizamos la condición en la frontera izquierda para eliminar $(\partial\phi/\partial x)_{1,1}$. La derivada parcial con respecto a y se aproxima por

$$\left(\frac{\partial\phi^2}{\partial y^2}\right)_{1,1} = \frac{\left(\frac{\partial\phi}{\partial y}\right)_{1,1+\frac{1}{2}} - \left(\frac{\partial\phi}{\partial y}\right)_{1,1}}{\frac{1}{2}}$$

$$= \frac{(\phi_{1,2} - \phi_{1,1}) - (\phi_{1,1} - 2)}{\frac{1}{2}}$$

$$= -4\phi_{1,1} + 2\phi_{1,2} + 4 \qquad \text{(C)}$$

donde utilizamos la condición en la frontera inferior para eliminar $(\partial\phi/\partial y)_{1,1}$. Sustituimos (B) y (C) en (11.2.2) para obtener

$$8\phi_{1,1} - 2\phi_{1,2} - 2\phi_{2,1} = 4 + S \qquad \text{(D)}$$

Punto (2, 1). Las derivadas parciales se aproximan por

$$\frac{\partial^2\phi}{\partial x^2} = \phi_{1,1} - 2\phi_{2,1} + \phi_{3,1} \qquad \text{(E)}$$

$$\frac{\partial \phi^2}{\partial y^2} = \frac{\left(\frac{\partial \phi^2}{\partial y}\right)_{2,1+\frac{1}{2}} - \left(\frac{\partial \phi}{\partial y}\right)_{2,1}}{\frac{1}{2}}$$

$$= 2\phi_{2,2} - 4\phi_{2,1} + 4 \qquad \text{(F)}$$

Sustituimos las dos ecuaciones anteriores en (11.2.2), con lo que se tiene

$$6\phi_{2,1} - \phi_{1,1} - 2\phi_{2,2} - \phi_{3,1} = 4 + S \qquad \text{(G)}$$

Punto (1, 2). Aproximamos las derivadas parciales mediante

$$\left(\frac{\partial^2 \phi}{\partial x^2}\right)_{1,2} = \frac{\left(\frac{\partial \phi}{\partial x}\right)_{1+\frac{1}{2},2} - \left(\frac{\partial \phi}{\partial x}\right)_{1,2}}{\frac{1}{2}}$$

$$= 2\phi_{2,2} - 4\phi_{1,2} \qquad \text{(H)}$$

$$\left(\frac{\partial \phi^2}{\partial y^2}\right)_{1,2} = \phi_{1,1} - 2\phi_{1,2} + \phi_{1,3} \qquad \text{(I)}$$

Sustituimos lo anterior en (11.2.2):

$$6\phi_{1,2} - \phi_{1,1} - 2\phi_{2,2} - \phi_{1,3} = S \qquad \text{(J)}$$

Punto (2, 2). La ecuación en diferencias es

$$4\phi_{2,2} - \phi_{1,2} - \phi_{3,2} - \phi_{2,1} - \phi_{2,3} = S \qquad \text{(K)}$$

Todo el conjunto de ecuaciones se resume en

$$\begin{aligned} 8\phi_{1,1} - 2\phi_{1,2} - 2\phi_{2,1} &= 4 + S \\ 6\phi_{2,1} - \phi_{1,1} - 2\phi_{2,2} &= 9 + S \\ 6\phi_{1,2} - \phi_{1,1} - 2\phi_{2,2} &= 7 + S \\ 4\phi_{2,2} - \phi_{1,2} - \phi_{2,1} &= 12 + S \end{aligned} \qquad \text{(L)}$$

11.2.2 Geometrías con fronteras curvas

En la sección anterior hemos supuesto que los dominios de las EDP elípticas son rectangulares. Sin embargo, es frecuente que en la práctica las geometrías tengan fronteras irregulares o curvas [Nogotov].

Existen tres puntos de vista principales aplicables a las geometrías no rectangulares:

a) El uso de una retícula rectangular y un ajuste a las ecuaciones en diferencias para los puntos de la retícula cercanos a la frontera.
b) El método del elemento finito.
c) Una transformación matemática de la geometría dada en un dominio rectangular, donde se puedan realizar los cálculos (transformación de coordenadas).

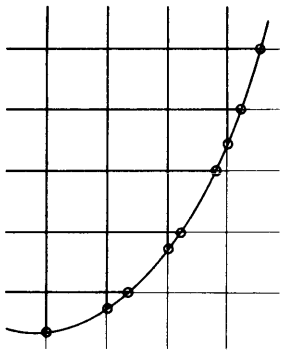


Figura 11.3 Un dominio con fronteras curvas

Para una mayor sencillez, el resto de esta subsección explica el primer punto de vista. Consideremos una frontera curva (como la que se muestra en la figura 11.3) en la que se introduce una retícula rectangular. Para adecuar ésta a la frontera curva, imponemos ciertos puntos especiales de la retícula en las intersecciones de las rectas regulares de la misma con la frontera curva, como se señala en la figura 11.3 mediante círculos pequeños. La ecuación en diferencias para los puntos de la retícula adyacentes a la frontera curva se puede escribir con facilidad. Por ejemplo, si consideramos el caso de la configuración de la retícula que aparece en la figura 11.4, podemos escribir la ecuación en diferencias para (11.2.2) como

$$-\frac{\left(\dfrac{\phi_a - \phi_{i,j}}{\alpha\,\Delta x} - \dfrac{\phi_{i,j} - \phi_{i-1,j}}{\Delta x}\right)}{\dfrac{1}{2}(1+\alpha)\,\Delta x} - \frac{\left(\dfrac{\phi_b - \phi_{i,j}}{\beta\,\Delta y} - \dfrac{\phi_{i,j} - \phi_{i,j+1}}{\Delta y}\right)}{\dfrac{1}{2}(1+\beta)\,\Delta y}$$

$$= \frac{-\phi_{i-1,j} + \left(1 + \dfrac{1}{\alpha}\right)\phi_{i,j} - \dfrac{\phi_a}{\alpha}}{\dfrac{1}{2}(1+\alpha)\,\Delta x^2} + \frac{-\phi_{i,j+1} + \left(1 + \dfrac{1}{\beta}\right)\phi_{i,j} - \dfrac{\phi_b}{\beta}}{\dfrac{1}{2}(1+\beta)\,\Delta x^2} = S_{i,j} \qquad (11.2.18)$$

donde ϕ_a y ϕ_b están dados por las condiciones en la frontera.

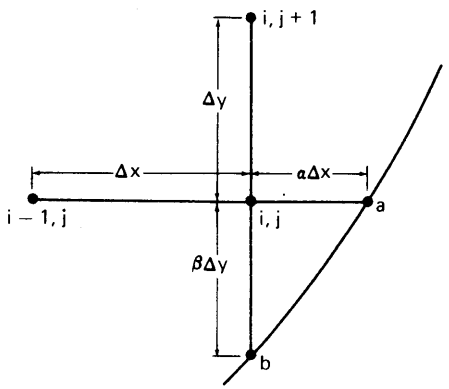


Figura 11.4 Un punto de la retícula adyacente a la frontera curva

11.2.3 Método de integración para la obtención de ecuaciones en diferencias

En esta sección nos restringiremos a dominios rectangulares y estudiaremos un método universal para la obtención de las ecuaciones en diferencias para una EDP elíptica, el cual se basa en la integración de dicha ecuación en el volumen que pertenece a un punto de la retícula. Mediante este método, podemos obtener las ecuaciones en diferencias para casi cualquier situación que incluya: coeficientes variables de la ecuación diferencial parcial elíptica, espaciamiento variable en la retícula y coordenadas cilíndricas y esféricas.

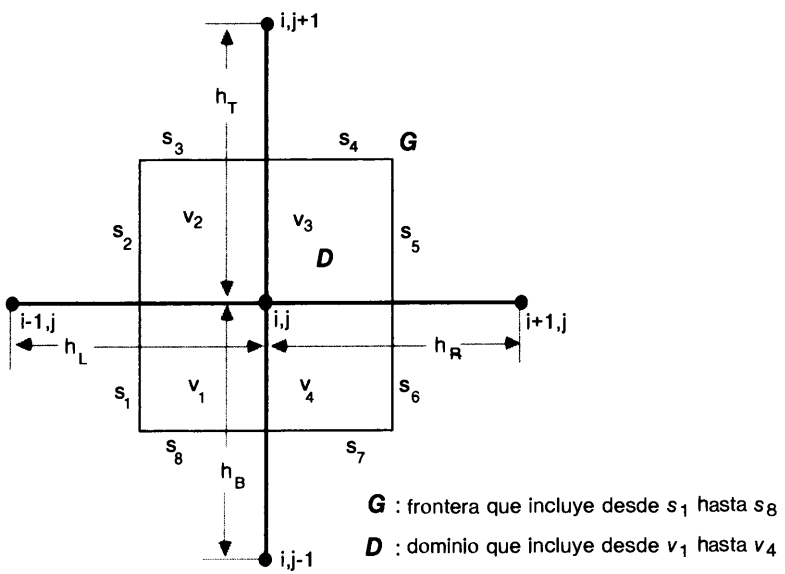
Consideremos la ecuación dada por

$$-\nabla p(x, y)\nabla\phi(x, y) + q(x, y)\phi(x, y) = S(x, y) \tag{11.2.19}$$

Aquí, el operador ∇ puede estar dado en cualquier sistema coordenado, pero supondremos que está en coordenadas rectangulares. Consideremos un sistema reticular rectangular en el que: 1) los espaciamientos cambien de un intervalo al siguiente, 2) p, q y S sean funciones que dependan del tiempo pero que sean constantes en cada uno de los rectángulos cuyas esquinas son cuatro puntos adyacentes de la retícula, por ejemplo, (i, j), $(i-1, j)$, $(i, j-1)$ e $(i-1, j-1)$.

Consideremos ahora un punto (i, j) de la retícula, así como sus cuatro puntos adyacentes, como se muestra en la figura 11.5. La caja rectangular que contiene a dicho punto consta de cuatro lados, cada uno de los cuales pasa por el punto medio del segmento entre el punto (i, j) y uno de los puntos adyacentes de la retícula. Denotamos el rectángulo y su frontera por D y G, respectivamente.

Figura 11.5 Puntos interiores de la retícula con espaciamiento variable (D denota al dominio rectangular que incluye de v_1 hasta v_4 y G es su frontera)



Al integrar (11.2.19) en el dominio D de la figura 11.5 se tiene

$$-\int_G p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds + \iint_D q(x, y)\phi(x, y)\,dx\,dy = \iint_D S(x, y)\,dx\,dy \qquad (11.2.20)$$

en la que hemos utilizado el teorema de Green para el primer término, cuya integral se toma a lo largo de G; además, $\partial/\partial n$ es la derivada normal hacia afuera de la frontera [Nakamura].

Separamos el primer término de (11.2.20) en cuatro partes:

$$\begin{aligned}-\int_G p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds = &-\int_{s_1+s_2} p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds\\ &-\int_{s_3+s_4} p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds\\ &-\int_{s_5+s_6} p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds\\ &-\int_{s_7+s_8} p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds \qquad (11.2.21)\end{aligned}$$

donde los términos s_n denotan las partes de la frontera de D (véase la figura 11.5). Podemos aproximar las derivadas parciales de la ecuación (11.2.21) mediante la aproximación por diferencias en los puntos medios entre dos puntos adyacentes de la retícula. Por ejemplo, la aproximación por diferencias para $\partial\phi/\partial n$ en el primer término del lado derecho de la ecuación (11.2.21) es

$$\frac{\partial\phi}{\partial n} = -\frac{\partial\phi}{\partial x} = -\frac{\phi_{i,j} - \phi_{i-1,j}}{h_L} \qquad (11.2.22)$$

Así, podemos escribir el primer término, después del signo de la igualdad en la ecuación (11.2.21) como

$$-\int_{s_1+s_2} p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds = (s_1p_1 + s_2p_2)\frac{\phi_{i,j} - \phi_{i-1,j}}{h_L} \qquad (11.2.23)$$

Del lado derecho de la ecuación (11.2.23), s_1 y s_2 son las longitudes de las partes de la frontera izquierda de D. En las coordenadas xy, los valores de s_k, $k = 1, 2, \ldots, 8$ son

$$\begin{aligned}&s_1 = \frac{h_B}{2}, \quad s_2 = \frac{h_T}{2}, \quad s_3 = \frac{h_L}{2}, \quad s_4 = \frac{h_R}{2},\\ &s_5 = \frac{h_T}{2}, \quad s_6 = \frac{h_B}{2}, \quad s_7 = \frac{h_R}{2}, \quad s_8 = \frac{h_L}{2}\end{aligned} \qquad (11.2.24)$$

Las demás integrales que aparecen en la ecuación (11.2.21) se pueden aproximar de manera análoga. Por lo tanto, la ecuación (11.2.21) se transforma en

$$\begin{aligned}-\int p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds = {} & (s_1p_1 + s_2p_2)\frac{\phi_{i,j} - \phi_{i-1,j}}{h_L} \\ & + (s_3p_2 + s_4p_3)\frac{\phi_{i,j} - \phi_{i,j+1}}{h_T} \\ & + (s_5p_3 + s_6p_4)\frac{\phi_{i,j} - \phi_{i+1,j}}{h_R} \\ & + (s_7p_4 + s_8p_1)\frac{\phi_{i,j} - \phi_{i,j-1}}{h_B} \qquad (11.2.25)\end{aligned}$$

Aproximamos el lado derecho de la ecuación (11.2.20) y el segundo término del lado izquierdo, respectivamente, por

$$\iint_D q(x, y)\phi(x, y)\,dx\,dy \simeq (v_1q_1 + v_2q_2 + v_3q_3 + v_4q_4)\phi_{i,j} \qquad (11.2.26)$$

$$\iint_D S(x, y)\,dx\,dy \simeq v_1S_1 + v_2S_2 + v_3S_3 + v_4S_4 \qquad (11.2.27)$$

donde

$$v_1 = \frac{h_Lh_B}{4}, \quad v_2 = \frac{h_Lh_T}{4}, \quad v_3 = \frac{h_Rh_T}{4}, \quad v_4 = \frac{h_Rh_B}{4}$$

Agrupamos todos los términos de las ecuaciones (11.2.25) a la (11.2.27), con lo que obtenemos la ecuación en diferencias asociada a los cinco puntos de la retícula:

$$a^C\phi_{i,j} + a^L\phi_{i-1,j} + a^R\phi_{i+1,j} + a^B\phi_{i,j-1} + a^T\phi_{i,j+1} = S_{i,j} \qquad (11.2.28)$$

$$a_L = -\left[\frac{s_1p_1 + s_2p_2}{h_L}\right]_{i,j}$$

$$a_T = -\left[\frac{s_3p_2 + s_4p_3}{h_T}\right]_{i,j}$$

$$a_R = -\left[\frac{s_5p_3 + s_6p_4}{h_R}\right]_{i,j}$$

$$a_B = -\left[\frac{s_7p_4 + s_8p_1}{h_B}\right]_{i,j}$$

$$a_C = [-a_L - a_T - a_R - a_B + v_1q_1 + v_2q_2 + v_3q_3 + v_4q_4]_{i,j}$$

$$S_{i,j} = [v_1S_1 + v_2S_2 + v_3S_3 + v_4S_4]_{i,j}$$

donde los subíndices i y j después de los paréntesis cuadrados indican que los valores se evalúan en el punto (i, j) de la retícula.

El método para obtener las ecuaciones en diferencias para los puntos de la retícula que se encuentran en las fronteras es análogo al procedimiento para los puntos internos. Para ilustrar esto, consideremos un punto en una esquina, como se muestra en la figura 11.6, y supongamos que las condiciones en la frontera tienen la forma:

$$\begin{aligned} \frac{\partial}{\partial y}\phi &= -\alpha_T\phi + \beta_T \quad \text{(condición en la frontera superior)} \\ \frac{\partial}{\partial x}\phi &= -\alpha_R\phi + \beta_R \quad \text{(condición en la frontera derecha)} \end{aligned} \tag{11.2.29}$$

Mediante la ecuación (11.2.20) podemos determinar la ecuación en diferencias para el punto de la esquina, si denotamos el dominio rectangular perteneciente al punto (i, j) de la esquina por D y a la frontera por G (véase la figura 11.6).

Separamos el primer término de la ecuación (11.2.20) en cuatro partes:

$$\begin{aligned} -\int_G p(x, y)\frac{\partial}{\partial n}\phi(x, y)\,ds &= -\int_{s_1} p(x, y)\frac{\partial}{\partial n}\phi(x,y)\,ds \\ &= -\int_{s_3} p(x, y)\frac{\partial}{\partial n}\phi(x,y)\,ds \\ &= -\int_{s_6} p(x, y)\frac{\partial}{\partial n}\phi(x,y)\,ds \\ &= -\int_{s_8} p(x, y)\frac{\partial}{\partial n}\phi(x,y)\,ds \end{aligned} \tag{11.2.30}$$

Las aproximaciones por diferencias para el caso del primer y cuarto términos de la ecuación anterior son análogos a los de un punto interno de la retícula y están dadas por

$$\begin{aligned} -\int_{s_1} p(x,y)\frac{\partial}{\partial n}\phi(x,y)\,ds &= p_1 s_1 \frac{\phi_{i,j} - \phi_{i-1,j}}{h_L} \\ -\int_{s_8} p(x,y)\frac{\partial}{\partial n}\phi(x,y)\,ds &= p_1 s_8 \frac{\phi_{i,j} - \phi_{i,j-1}}{h_B} \end{aligned} \tag{11.2.31}$$

Evaluamos el segundo y tercer términos mediante las condiciones en la frontera:

$$\begin{aligned} -\int_{s_3} p(x, y)\frac{\partial}{\partial n}\phi(x,y)\,ds &= p_1 s_3(\alpha_T\phi_{i,j} - \beta_T) \\ -\int_{s_6} p(x, y)\frac{\partial}{\partial n}\phi(x,y)\,ds &= p_1 s_6(\alpha_R\phi_{i,j} - \beta_R) \end{aligned} \tag{11.2.32}$$

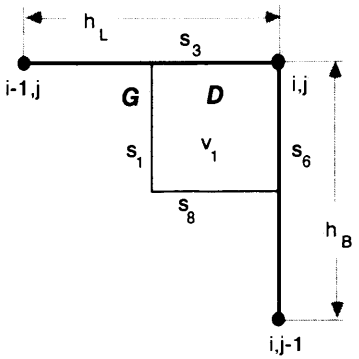


Figura 11.6 Puntos de la retícula en una frontera

Aproximamos el segundo término del lado izquierdo de (11.2.20) por

$$\iint_D q(x, y)\phi(x, y)\,dx\,dy \simeq (v_1 q_1)\phi_{i,j} \tag{11.2.33}$$

El lado derecho de la ecuación (11.2.20) se transforma en

$$\iint_D S(x, y)\,dx\,dy = v_1 S_1$$

Agrupamos todos los términos para obtener

$$a^C\phi_{i,j} + a^L\phi_{i-1,j} + a^B\phi_{i,j-1} = S_{i,j} \tag{11.2.34}$$

$$a_L = -\left[\frac{p_1 s_1}{h_L}\right]_{i,j}$$

$$a_B = -\left[\frac{p_1 s_8}{h_B}\right]_{i,j}$$

$$a_C = [-a_L - a_B + \ + v_1 q_1 + p_1 s_3 \alpha_T + p_1 s_6 \alpha_R]_{i,j}$$

$$S_{i,j} = [v_1 S_1 + p_1 s_3 \beta_T + p_1 s_6 \beta_R]_{i,j}$$

En otros sistemas de coordenadas, se pueden obtener las ecuaciones en diferencias mediante el mismo proceso, si se maneja de manera adecuada la integral de volumen para el sistema de coordenadas dado. Como ejemplo, consideremos la ecuación para el sistema de coordenadas r-z dada por

$$-\left[\frac{1}{r}\frac{\partial}{\partial r} pr \frac{\partial}{\partial r}\phi(r,z) + \frac{\partial}{\partial z}\phi(r,z)\right] + q(r, z)\phi(r, z) = S(r, z) \tag{11.2.35}$$

Como se muestra en la figura 11.7, en un sistema de coordenadas cilíndricas se pueden utilizar un punto (i, j) de la retícula y cuatro puntos adyacentes de la misma, si x y y se intercambian por r y z, respectivamente. Sin embargo, en las coordenadas r-z, el dominio D representa una figura en forma de dona, como en la figura 11.8.

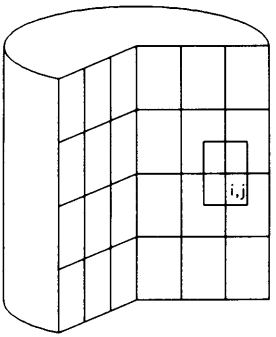


Figura 11.7 Retícula en las coordenadas cilíndricas

Integramos la ecuación (11.2.35) en el dominio D. Esto quiere decir que la integración se lleva a cabo en un volumen en vez de un plano y está dada por

$$2\pi \iint_D \left[-\frac{1}{r}\frac{\partial}{\partial r} pr \frac{\partial}{\partial r} \phi(r, z) - \frac{\partial_2}{\partial z^2} \phi(r,z) + q(r, z)\phi(r, z) \right] r\, dr\, dz$$
$$= 2\pi \iint_D S(r, z) r\, dr\, dz \tag{11.2.36}$$

lo cual se puede escribir, mediante el teorema de Green, como

$$-2\pi \int_G p \frac{\partial}{\partial n} \phi(r,z) r\, ds + 2\pi \iint_D q\phi(r, z) r\, dr\, dz = 2\pi \iint_D S(r, z) r\, dr\, dz \tag{11.2.37}$$

donde la integral del primer término se toma sobre la superficie (o frontera) de D. El resto de la operación es muy similar al caso de la geometría plana antes descrito. La ecuación en diferencias resultante toma exactamente la misma forma de la ecuación (11.2.28) siempre que v y s se interpreten de la manera siguiente; es decir, v y s de la figura 11.5 se interpretan como volúmenes y áreas de la superficie parciales del dominio D. Por ejemplo, v_1 y s_1 se escriben respectivamente como

$$v_1 = \pi(r_i^2 - r_a^2)\frac{h_B}{2}$$

$$s_1 = 2\pi r_a \frac{h_B}{2} = \pi r_a h_B$$

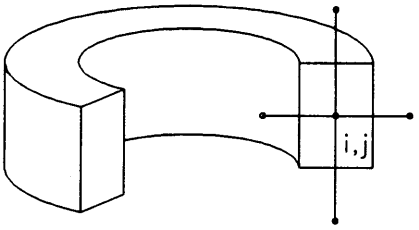


Figura 11.8 Dominio en forma de dona

donde r_a es el radio en el punto medio entre los puntos $(i-1, j)$ e (i, j) de la retícula y r_1 es el radio en (i, j).

11.2.4 Propiedades de las ecuaciones en diferencias

Todas las ecuaciones en diferencias que hemos obtenido en las subsecciones previas se pueden escribir en la forma

$$a^C\phi_{i,j} + a^L\phi_{i-1,j} + a^R\phi_{i+1,j} + a^B\phi_{i,j-1} + a^T\phi_{i,j+1} = S_{i,j} \tag{11.2.38}$$

Esta ecuación se puede utilizar de aquí en adelante para todos los puntos de la retícula, con las interpretaciones siguientes:

a) Aunque por simplicidad se omiten los subíndices i y j de las a, éstas dependen de i y j.
b) Si el punto (i, j) se encuentra fuera del dominio, entonces $\phi_{i,j}$ y sus coeficientes se consideran nulos.
c) Todas las incógnitas están en la lado izquierdo, mientras que los términos conocidos se agrupan en $S_{i,j}$. Por ejemplo, si se conoce el valor de $\phi_{i,j+1}$ a partir de la condición en la frontera, el término se pasa al lado derecho y se suma a $S_{i,j}$.

Con las condiciones apropiadas en la frontera, las ecuaciones en diferencias (11.2.38) de todo el dominio tienen las propiedades siguientes:

a) Los coeficientes de (11.2.38) son negativos o nulos, excepto por a^c.
b) El coeficiente a^c es positivo y recibe el nombre de coeficiente diagonal.
c) La propiedad de simetría.*

$$\begin{aligned} (a^L)_{i,j} &= (a^R)_{i-1,j} \\ (a^B)_{i,j} &= (a^T)_{i,j-1} \end{aligned} \tag{11.2.39}$$

d) El coeficiente a^c es mayor o igual que la suma de los valores absolutos de los demás coeficientes (dominancia diagonal)

$$a^C \geq |a^L| + |a^R| + |a^B| + |a^T| \tag{11.2.40}$$

* Como se ilustró en el ejemplo 1.1, las ecuaciones en diferencias para una EDP elíptica sin términos de derivadas de primer orden se pueden escribir en la forma simétrica, si el conjunto de ecuaciones está en forma conservativa. Debido a esta propiedad de simetría, sólo se requiere almacenar en espacio de memoria tres coeficientes de cada ecuación.

en donde la desigualdad es estricta al menos para un punto de la retícula.* Cuando la desigualdad siempre es estricta, se dice que la ecuación tiene una dominancia diagonal fuerte.

e) Ninguna parte del conjunto de ecuaciones puede resolverse independientemente de las otras (irreductibilidad).

Las cinco propiedades listadas arriba son importantes, puesto que son condiciones suficientes para que un esquema iterativo converja [Varga; Wachspress]. Cuando los coeficientes de las ecuaciones en diferencias que tienen las propiedades anteriores se escriben en forma matricial, la matriz recibe el nombre de *matriz de Stieltjes* o matriz-S.

Resumen de Esta Sección

a) Las ecuaciones en diferencias para una EDP elíptica se pueden obtener mediante la aplicación directa de la aproximación por diferencias o bien mediante el método de integración. El primero es más simple, pero el segundo es más poderoso si los coeficientes de la EDP dependen del espacio y se tiene un espaciamiento variable en la retícula.
b) Las condiciones en la frontera se incorporan a las ecuaciones en diferencias.
c) Las ecuaciones en diferencias en forma matricial son simétricas, pentadiagonales y tienen dominancia diagonal.

11.3 PANORAMA DE LOS METODOS DE SOLUCION PARA LAS ECUACIONES EN DIFERENCIAS ELIPTICAS

Los métodos numéricos que se utilizan para resolver un conjunto de ecuaciones en diferencias se puede clasificar en dos tipos: iterativos y directos. Los primeros se pueden usar en forma universal para problemas de cualquier índole, mientras que los métodos directos sólo son adecuados si se cumple al menos una de las condiciones siguientes:

* En el caso de la ecuación de Poisson o de Laplace, se puede dar la desigualdad sólo en una condición en la frontera. Si existe una término positivo de supresión q como en la ecuación (11.2.26), éste también contribuye a mejorar la desigualdad. El significado físico de una dominancia diagonal fuerte se interpreta como una salida o supresión de la cantidad física (por ejemplo, partículas o calor) representada por la solución. Sin una salida, el sistema físico no tiene un estado estacionario, a menos que el término fuente se anule o el total de la fuente se anule. Si se aplica un método iterativo a un sistema sin salida o sin dominancia diagonal, éste puede no converger. Con frecuencia existen algunas excepciones cuando se resuelve una ecuación de Poisson que surge de un cálculo en dinámica de fluidos. La situación es que:

a) Las ecuaciones en diferencias no tienen dominancia diagonal fuerte, es decir, no hay salida.
b) Sin embargo, si se suman todas las ecuaciones en diferencias, el total de los términos no homogéneos se anula (pues existen términos no homogéneos positivos y negativos).
c) También se anula el total de los términos homogéneos.

Una ecuación de Poisson con estas propiedades tiene una solución, aunque ésta no es única, ya que la solución más una constante arbitraria también es una solución. Si se aplica un método de solución iterativo a dicho problema, éste converge, pero el valor final depende de la estimación inicial o de los parámetros de iteración que se hayan utilizado. A la solución se puede sumar o restar una constante.

a) El número de puntos en la retícula es muy pequeño.
b) Los coeficientes de las ecuaciones en diferencias tienen una forma especial y simple.
c) Se dispone de una computadora con un enorme espacio de memoria.

Para poder explicar las dificultades básicas asociadas con la solución directa de las ecuaciones en diferencias, debemos observar que los elementos de la matriz de coeficientes —la cual representa al conjunto lineal de ecuaciones en diferencias— son casi todos nulos, excepto los ubicados a lo largo de cinco líneas diagonales, lo cual se muestra incluso en un conjunto pequeño de ecuaciones, como las del ejemplo 11.1 (véase también la matriz M de la figura 11.11). Si se piensa utilizar la eliminación de Gauss, todos los coeficientes (incluyendo los nulos) deben guardarse en la memoria de la máquina, de forma que el espacio total puede exceder con facilidad la memoria central disponible. Por ejemplo, el tamaño de una matriz de 20×20 puntos de la retícula se transforma en $400^2 = 160{,}000$.

Sin embargo, al analizar la matriz nos damos cuenta de que ésta tiene una forma de banda diagonal. Por ejemplo, la matriz de coeficientes para una retícula de N por M tiene una banda de $2N + 1$ elementos de ancho. Si aplicamos la eliminación de Gauss, debemos desarrollar un programa de forma tal que sólo almacene $2N + 1$ por M coeficientes, incluyendo los nulos. En este caso, la cantidad necesaria de espacio de memoria es $(2N + 1)(MN)$ en vez de $(NM)^2$.

Si ocurre que tanto los coeficientes de la EDP original como los de la retícula tienen cierta estructura simple, se puede aplicar la transformada rápida de Fourier (TRF) [Nussbaumer] o bien la solución directa rápida (véase la sección 11.8). Por desgracia, la mayoría de los conjuntos de ecuaciones en diferencias no cumplen con estas condiciones. Por otro lado, en los métodos iterativos sólo hay que guardar los coeficientes no nulos. Por lo tanto, incluso un problema con una retícula enorme se puede resolver en forma iterativa con requerimientos mínimos de memoria central.

La tabla 11.2 resume las ventajas y desventajas de los métodos iterativos y directos.

Resumen de Esta Sección

a) Las ecuaciones en diferencias para una EDP elíptica se pueden resolver tanto por métodos iterativos como directos.
b) Los métodos iterativos son de uso más común que los métodos directos. Son sencillos y a menudo constituyen la única forma de resolver las ecuaciones en diferencias.
c) Bajo ciertas condiciones, los métodos directos pueden ser muy eficaces.

11.4 METODOS DE RELAJACION SUCESIVA

Los métodos iterativos se utilizan mucho más ampliamente que los directos en la solución de las ecuaciones en diferencias elípticas, debido a que necesitan mucho menos memoria central y a que son aplicables a casi todas las ecuaciones en diferencias que surgen de las EDP elípticas. Los métodos iterativos son versátiles, se pueden

Tabla 11.2 Resumen de los métodos de solución de diferencias finitas para las ecuaciones diferenciales parciales elípticas

Métodos de solución	Ventajas	Desventajas
Métodos iterativos		
Métodos de relajación, como el método iterativo de Jacobi y el SOR	Programación sencilla. Demostración teórica completa. El parámetro de iteración se optimiza fácilmente.	La tasa de convergencia es más lenta si aumenta el tamaño de la retícula.
Método iterativo extrapolado de Jacobi	La misma eficiencia del SOR en una computadora escalar, pero mucho más rápido en una supercomputadora.	Igual que el anterior.
Método implícito de la dirección alternante (IDA) (véase la sección 11.7)	Más rápido que el SOR para una amplia clase de problemas.	Difícil para optimizar los parámetros de aceleración. Más esfuerzo de programación que el SOR. Para muchos problemas, su tasa de convergencia es más lenta que la del SOR.
Soluciones directas		
Eliminación de Gauss y descomposición LU	Rápido para problemas de tamaño pequeño e intermedio. Robusto.	No se aplica a problemas grandes, debido a que requiere mucho tiempo de cómputo y espacio de memoria.
Solución directa rápida (SDR)	No iterativo. Más rápido que los métodos iterativos.	Limitaciones en la geometría y el número de puntos de la retícula.
Transformada rápida de Fourier (TRF)	Igual que el anterior.	Igual que el anterior.

implantar con esfuerzos de programación relativamente menores y a menudo son la única forma de resolver las ecuaciones en diferencias.

Entre los métodos de relajación sucesiva están el método iterativo de Jacobi, el método de Gauss-Seidel y los métodos de sobrerrelajación sucesiva (SOR). Cada uno de ellos tiene dos versiones: relajación puntual y relajación por líneas.

En esta sección nos centraremos en los tres métodos representativos de iteración puntual: el método iterativo de Jacobi, el de sobrerrelajación sucesiva (SOR; del inglés successive-over-relaxation) y un método iterativo extrapolado de Jacobi (EJ). No analizaremos los métodos de relajación por líneas debido a las limitaciones de espacio. Conviene observar que lo que se gana con estos últimos métodos con respecto a los primeros no es sustancial, excepto para las geometrías rectangulares con un radio de aspecto grande.

Los métodos SOR, en particular el puntual, son métodos iterativos bien conocidos. Aunque el EJ es menos conocido, es casi tan simple de programar que el SOR y tiene exactamente la misma eficiencia computacional en las computadoras escalares. Sin embargo, en una supercomputadora, su eficiencia es varias veces mayor que la del SOR.

11.4.1 Método iterativo de Jacobi

Desde el punto de vista de la eficiencia, el método que se presenta en esta sección es impráctico, si se utiliza aislado. Empero, tiene importancia teórica al analizar los métodos SOR y EJ.

En primer lugar, reescribimos la ecuación (11.2.28) como

$$a^C\phi_{i,j} = S_{i,j} - (a^L\phi_{i-1,j} + a^R\phi_{i+1,j} + a^B\phi_{i,j-1} + a^T\phi_{i,j+1}) \tag{11.4.1}$$

El método iterativo de Jacobi se obtiene a partir de la ecuación anterior sumando el número de iteración t o $t - 1$ como índice superior a cada ϕ y dividiendo entre a^c, de lo que se obtiene

$$\phi_{i,j}^{(t)} = [S_{i,j} - (a^L\phi_{i-1,j}^{(t-1)} + a^R\phi_{i+1,j}^{(t-1)} + a^B\phi_{i,j-1}^{(t-1)} + a^T\phi_{i,j+1}^{(t-1)})]/a^C \tag{11.4.2}$$

Para el primer ciclo de iteración $t = 1$, $\phi^{(0)}{}_{i,j}$ del lado derecho de la ecuación (11.4.2) es una estimación inicial. En cada uno de los ciclos, se evalúa (11.4.2) para todos los puntos de la retícula, excepto en aquéllos para los que están dadas las condiciones en la frontera como valores fijos. El esquema converge con cualquier estimación inicial de la solución, aunque mientras más cercana esté esta estimación al valor exacto, más rápida será la convergencia.

Un examen más detallado de este método revela, sin embargo, que no hay que calcular todos los puntos de la retícula en cada ciclo. Sólo es necesario recorrer la mitad de los puntos (propiedad 2-cíclica).* Supongamos que los puntos de la retícula tienen colores, alternados, escalonados en negro y rojo. Por ejemplo, si el punto (i, j) tiene color rojo, entonces sus vecinos $(i - 1, j)$, $(i + 1, j)$, $(i, j - 1)$, $(i, j + 1)$ tienen color negro. Es fácil ver que los cálculos para los puntos rojos del ciclo t utilizan sólo los puntos negros del ciclo $t - 1$ y viceversa. Si sólo se hacen los cálculos de los puntos rojos en el ciclo t y luego los cálculos de los puntos negros en el ciclo $t + 1$, todos los puntos se actualizan después de dos ciclos de iteración. La iteración puede continuar de esta forma hasta que la solución converja. Este enfoque reduce todo el trabajo computacional a la mitad.

El número de pasos de iteración necesarios para que un método iterativo converja no se conoce de antemano. Por lo tanto, el método continúa hasta que se satisface un criterio de convergencia. Algunos de estos criterios se describen a continuación:

La iteración se detiene si se cumple

$$|\phi_{i,j}^{(t)} - \phi_{i,j}^{(t-2)}| < \varepsilon \tag{11.4.3}$$

para todos los puntos recorridos, donde ε es un criterio preestablecido. En el caso en que las soluciones en puntos distintos varíen por órdenes de magnitud, este criterio se debe modificar a

* Si todos los puntos se recorren en cada ciclo de iteración, se llevan a cabo en forma simultánea dos secuencias independientes de cálculos, las cuales convergen a la misma solución. Estas dos series de cálculos no intercambian la información en cada ciclo de iteración. Esta propiedad de las series duales en el método iterativo de Jacobi se llama *propiedad 2-cíclica* [Varga].

$$\left|1 - \frac{\phi_{i,j}^{(t)}}{\phi_{i,j}^{(t-2)}}\right| < \varepsilon \tag{11.4.4}$$

Así, se prueban los valores muy pequeños o muy grandes de la solución bajo la base de cambios relativos en cada ciclo iterativo.

Las ecuaciones (11.4.3) y (11.4.4) necesitan un enunciado "IF" en el ciclo "DO", que podría no ser recomendable para ciertas computadoras. En tales casos, un punto de vista alternativo sería el remplazar la ecuación (11.4.4) por

$$\frac{\sum_{i,j} |1 - \phi_{i,j}^{(t)}/\phi_{i,j}^{(t-2)}|}{\text{número total de puntos}} < \varepsilon$$

11.4.2 Método de sobrerrelajación sucesiva (SOR)

El método SOR está dado por

$$\phi_{i,j}^{(t)} = \omega[S_{i,j} - (a^L\phi_{i-1,j}^{(t)} + a^R\phi_{i+1,j}^{(t-1)} + a^B\phi_{i,j-1}^{(t)} + a^T\phi_{i,j+1}^{(t-1)})]/a_C + (1-\omega)\phi_{i,j}^{(t-1)} \tag{11.4.5}$$

donde ω es el parámetro de sobrerrelajación, con $1 < \omega < 2$.* Como se puede ver fácilmente, el método SOR se obtiene modificando el método iterativo de Jacobi en dos aspectos. En primer lugar, los índices superiores de $\phi_{i-1,j}$ y $\phi_{i,j-1}$ se cambian de $t-1$ hasta t. Para que esto sea posible, los puntos de la retícula se recorren en orden creciente de i y j, puesto que al hacer esto los términos $\phi_{i-1,j}^{(t)}$ y $\phi_{i,j-1}^{(t)}$ del paso iterativo en curso siempre están disponibles para el cálculo de $\phi_{i,j}^{(t)}$. En segundo lugar, se introduce el parámetro de iteración ω.

En el SOR se recorren todos los puntos de la retícula en cada ciclo.

Si $\omega = 1$, el método recibe el nombre de método de Gauss-Seidel. El óptimo ω está en el lado menor de $1 < \omega < 2$ si el número de puntos en la retícula es muy pequeño, pero tiende a 2 al crecer el número de dichos puntos. En los problemas pequeños, la tasa de convergencia es rápida de manera intrínseca y relativamente insensible al valor elegido de ω por lo que podemos establecer un valor de $\omega = 1.8$ como regla. En la sección 11.5 se analiza con más detalle el efecto de ω. En el PROGRAMA 11-1 se muestra una implantación del SOR puntual. Si la iteración consume más de algunos cientos de ciclos de iteración, es muy recomendable el cálculo de un ω óptimo, como se describe en la sección 11.6.

* La ecuación (11.45) es de sobrerrelajación o de subrelajación, dependiendo de que $1 < \omega < 2$ o de que $0 < \omega < 1$ respectivamente. El segundo tipo no es útil para las EDP lineales elípticas debido a que la tasa de convergencia es más lenta que en el primero. Sin embargo, para una ecuación no lineal en la que se revisen los coeficientes de las ecuaciones en diferencias después de cada iteración (mediante los valores iterados previos) puede aparecer un fenómeno de inestabilidad, a menos que se utilice la subrelajación.

Ejemplo 11.3

Resuelva el conjunto de ecuaciones en diferencias del ejemplo 11.2 mediante el SOR con $\omega = 1.5$.

(Solución)

Obtenemos la solución mediante los siguientes pasos:

a) Movemos todos los términos con signos negativos del lado derecho.

b) Dividimos cada ecuación entre el coeficiente del término del lado izquierdo:

$$\begin{aligned}
\phi_{1,1} &= \frac{1}{8}(2\phi_{1,2} + 2\phi_{2,1} + 4 + S) \\
\phi_{2,1} &= \frac{1}{6}(\phi_{1,1} + 2\phi_{2,2} + 9 + S) \\
\phi_{1,2} &= \frac{1}{6}(\phi_{1,1} + 2\phi_{2,2} + 7 + S) \\
\phi_{2,2} &= \frac{1}{4}(\phi_{1,2} + \phi_{2,1} + 12 + S)
\end{aligned} \tag{A}$$

c) Si denotamos el número de iteración por t, escribimos el SOR como

$$\begin{aligned}
\phi_{1,1}^{(t)} &= \frac{1}{8}\omega(2\phi_{1,2}^{(t-1)} + 2\phi_{2,1}^{(t-1)} + 4 + S) + (1-\omega)\phi_{1,1}^{(t-1)} \\
\phi_{2,1}^{(t)} &= \frac{1}{6}\omega(\phi_{1,1}^{(t)} + 2\phi_{2,2}^{(t-1)} + 9 + S) + (1-\omega)\phi_{2,1}^{(t-1)} \\
\phi_{1,2}^{(t)} &= \frac{1}{6}\omega(\phi_{1,1}^{(t)} + 2\phi_{2,2}^{(t-1)} + 7 + S) + (1-\omega)\phi_{1,2}^{(t-1)} \\
\phi_{2,2}^{(t)} &= \frac{1}{4}\omega(\phi_{1,2}^{(t)} + \phi_{2,1}^{(t)} + 12 + S) + (1-\omega)\phi_{2,2}^{(t-1)}
\end{aligned} \tag{B}$$

Aquí, las ecuaciones se recorren en orden de $\phi_{1,1}$, $\phi_{2,1}$, $\phi_{1,2}$ y $\phi_{2,2}$. Después de calcular el nuevo valor de $\phi_{i,j}^{(t)}$, el valor anterior $\phi_{i,j}^{(t-1)}$ ya no se necesita, por lo que el primero se reescribe sobre el segundo, utilizando el mismo espacio en la memoria. Todas las estimaciones iniciales se hacen iguales a cero:

d) La iteración de la ecuación (B) continúa hasta que todas las $\phi_{i,j}$ convergen.

11.4.3 Método iterativo extrapolado de Jacobi (EJ)

El método iterativo puntual de Jacobi, basado en la propiedad 2-cíclica (que permite recorrer solamente la mitad de los puntos de la retícula en cada ciclo de iteración) se puede acelerar mediante un parámetro de extrapolación de la manera siguiente:

$$\phi_{i,j}^{(t)} = \frac{\theta}{a^C}\left[S_{i,j} - (a^L\phi_{i-1,j}^{(t-1)} + a^R\phi_{i+1,j}^{(t-1)} + a^B\phi_{i,j-1}^{(t-1)} + a^T\phi_{i,j+1}^{(t-1)})\right] + (1-\theta)\phi_{i,j}^{(t-2)} \quad (11.4.6)$$

donde θ es el parámetro de extrapolación, que cumple $1 < \theta < 2$. Conviene observar que el segundo término del lado derecho no causa problemas en la implantación, puesto que cuando sólo se han recorrido la mitad de los puntos, el valor de $\phi_{i,j}^{(t-2)}$ en vez del de $\phi_{i,j}^{(t-1)}$ es el que se encuentra en el espacio de memoria correspondiente al punto (i, j). Este método se conoce también como el *SOR rojo-negro* [Hageman/Young].

En la sección 11.6 se muestra que, para un mismo problema, el óptimo de θ es idéntico al óptimo de ω para el SOR. La tasa de convergencia es la mitad de la tasa de SOR en términos de las veces que se realiza la iteración; pero la cantidad total de trabajo de cómputo es idéntica, puesto que solamente se recorren la mitad de los puntos en cada ciclo de iteración.

En una computadora escalar —como una IBM 370, VAX o IBM PC— la elección entre SOR y EJ es simplemente cuestión de gustos, ya que ambos tienen la misma sencillez y eficiencia. Sin embargo, la verdadera ventaja del EJ se experimenta en una supercomputadora con un procesador vectorial.

Al procesar un ciclo DO, el procesador vectorial vectoriza los cálculos en caso de que no se utilice ningún resultado de los cálculos en el mismo DO, excepto en el mismo ciclo. Por ejemplo, un DO como

```
   DO 20 I=1,10
      F(I)=(F(I-1)+F(I+1))/2
20 CONTINUE
```

no se puede vectorizar, puesto que se vuelve a utilizar el valor de F(I − 1) calculado en el mismo ciclo, pero

```
   DO 20 I=1,10,2
      F(I)=(F(I-1)+F(I+1))/2
20 CONTINUE
```

sí se puede vectorizar. Si un ciclo DO se vectoriza, la rapidez de cómputo se incrementa en forma significativa (usualmente, por un factor aproximado de cinco). El ciclo DO más interno del SOR puntual no se puede vectorizar, mientras que el del EJ sí.

El método EJ es una variación del método semiiterativo cíclico de Chebyshev [Varga]. En este método, se determina θ en cada paso de la iteración mediante el polinomio de Chebyshev y no es constante. Sin embargo, al aumentar el número de iteraciones, tiende a un valor asintótico, que es igual a $\theta_{óptl}$. La tasa global de convergencia de este método es un poco mejor que la del EJ (digamos, un 10% de reducción del total de pasos de iteración).

Resumen de Esta Sección

a) Entre los diferentes métodos iterativos que forman parte de los métodos con relajación, se presentan el método SOR puntual y el método iterativo extrapolado de Jacobi (EJ).
b) Las versiones de los métodos de relajación mediante líneas son sólo un poco más eficientes que las versiones puntuales, excepto para geometrías rectangulares con una razón de aspecto muy grande.
c) El método iterativo de Jacobi, el cual utiliza la propiedad 2-cíclica, tiene la misma eficacia que el método de Gauss-Seidel. Al extrapolarlo, su eficacia se vuelve equivalente a la del SOR.
d) En una supercomputadora con un procesador vectorial, EJ es significativamente más rápido que SOR.

11.5 ANALISIS DE CONVERGENCIA

El objetivo de esta sección es el análisis de las tasas de convergencia del método iterativo de Jacobi del SOR, y del método iterativo extrapolado de Jacobi (EJ). El análisis básico de la convergencia iterativa es importante en la aplicación práctica de los métodos, en particular en lo que se refiere a la eficacia computacional.

Por sencillez, el análisis se lleva a cabo en un problema unidimensional con valores en la frontera, pero el resultado es válido para dos y tres dimensiones. En primer lugar, estudiamos la convergencia del método iterativo de Jacobi, puesto que es la base para el análisis del SOR y del EJ.

Convergencia del Método Iterativo de Jacobi. Consideremos un problema unidimensional con valores en la frontera, dado por

$$-\frac{d^2\phi}{dx^2} = S \qquad \phi(0) = \phi(H) = 0 \tag{11.5.1}$$

Las ecuaciones en diferencias para la ecuación (11.5.1) se escriben como

$$-\phi_{i-1} + 2\phi_i - \phi_{i+1} = h^2 S, \quad i = 1, 2, \ldots, I \qquad \phi_0 = \phi_{I+1} = 0 \tag{11.5.2}$$

donde $h = H/(I + 1)$ es el espaciamiento en la retícula. El método iterativo de Jacobi para (11.5.2) es

$$\phi_i^{(t)} = \frac{1}{2}[h^2 S + \phi_{i-1}^{(t-1)} + \phi_{i+1}^{(t-1)}] \tag{11.5.3}$$

Si denotamos la solución exacta de (11.5.2) por ϕ_i sin el índice superior, podemos expresar a $\phi_i^{(t)}$ como

$$\phi_i^{(t)} = \phi_i - e_i^{(t)} \tag{11.5.4}$$

donde $e_i^{(t)}$ es el error. Así, la ecuación (11.5.3) queda

$$e_i^{(t)} = \frac{1}{2}[e_{i-1}^{(t-1)} + e_{i+1}^{(t-1)}] \tag{11.5.5}$$

El problema de valores propios asociado con la ecuación (11.5.2) se escribe como

$$\begin{aligned} \eta_m \psi_{m,i} &= \frac{1}{2}(\psi_{m,i-1} + \psi_{m,i+1}) \\ \psi_{m,0} &= \psi_{m,i+1} = 0 \end{aligned} \tag{11.5.6}$$

donde η_m es el m-ésimo valor propio y $\psi_{m,i}$ es la función propia correspondiente. Se muestra a continuación que los valores y funciones propios son

$$\psi_{m,i} = \operatorname{sen}(m\alpha i), \quad m = 1, 2, \ldots, I \tag{11.5.7a}$$

$$\eta_m = \cos(m\alpha), \quad m = 1, 2, \ldots, I \tag{11.5.7b}$$

donde el subíndice m denota a la m-ésima solución y $\alpha = \pi/(I + 1)$. Sustituimos (11.5.7a) del lado derecho (LD) de (11.5.6) para obtener

$$\text{LD} = \frac{1}{2}[\operatorname{sen}(m\alpha(i-1)) + \operatorname{sen}(m\alpha(i+1))] = \cos(m\alpha)\operatorname{sen}(m\alpha i) \tag{11.5.8}$$

en donde utilizamos el teorema de la suma para la función seno.* El lado izquierdo de la ecuación (11.5.6) no cambia aunque se sustituyan las ecuaciones (11.5.7a) y (11.5.7b). Esto demuestra que estas ecuaciones son los valores y funciones propias.**

* Teorema de la suma de la función seno:

$$\operatorname{sen}(njB \pm nB) = \operatorname{sen}(njB)\cos(nB) \pm \cos(njB)\operatorname{sen}(nB)$$

** Se podría pensar que las funciones sen $(m\alpha i)$, con m distinta de $1, 2, \ldots, I$ también son funciones propias. Si $m = 0$ o $I + 1$, sen $(m\alpha i)$ se anula para toda i, de modo que las funciones sen $(m\alpha i)$ para estos valores de m son soluciones triviales. Para los demás valores de $m < 0$ y $m > I + 1$, podemos demostrar que sen $(m\alpha i)$ es igual a una constante por sen $(m'\alpha i)$, donde m' es un entero tal que $0 < m' < I + 1$. Para mostrar esto, primero vemos que cualquier m se puede escribir en la forma $m = m' + n(I + 1)$ o bien $m = -m' + (n + 1)(I + 1)$, donde $0 < m' < I + 1$, y n es un entero. Utilizaremos la primera de estas expresiones si n es par y la última si n es impar. Para n par, sen $(m\alpha i)$ se transforma en

$$\operatorname{sen}(m\alpha i) = \operatorname{sen}(n\pi i + m'\alpha i) = \operatorname{sen}(m'\alpha i)$$

donde utilizamos $\alpha(I + 1) = \pi$. Para n impar, sen $(m\alpha i)$ se puede escribir como

$$\operatorname{sen}(m\alpha i) = \operatorname{sen}[(n+1)\pi i - m'\alpha i] = -\operatorname{sen}(m'\alpha i)$$

Así, las funciones sen $(m\alpha i)$ para $m = 1$ a I son las únicas funciones propias independientes.

Las funciones propias de la ecuación (11.5.7a) se pueden utilizar para desarrollar el error después de cada iteración, de forma que el error inicial $e_i^{(0)}$ se escribe como

$$e_i^{(0)} = \sum_{m=1}^{I} A_m \psi_{m,i} \tag{11.5.9}$$

donde A_m es un coeficiente.

Si hacemos $t = 1$ en la ecuación (11.5.5) y sustituimos la ecuación (11.5.9) del lado derecho de (11.5.5), obtenemos

$$e_i^{(1)} = \sum_{m=1}^{I} A_m \eta_m \psi_{m,i} \tag{11.5.10}$$

en donde usamos la ecuación (11.5.6). La ecuación (11.5.10) es el error de la solución iterativa después del primer ciclo de iteración. Si repetimos la sustitución, el error después del t-ésimo ciclo de iteración es

$$e_i^{(t)} = \sum_{m=1}^{I} A_m (\eta_m)^t \psi_{m,i} \tag{11.5.11}$$

Si $|\eta_m| < 1$ para toda m, el error se va desvaneciendo al aumentar el número t de iteraciones. La tasa global de decaimiento del error está determinada por

$$\mu_J = \max_m |\eta_m| \tag{11.5.12}$$

que es el radio espectral del método iterativo de Jacobi. Puesto que el máximo de $|\eta_m|$ se alcanza tanto en $m = 1$ como en $m = I$ [véase la ecuación (11.5.7b)], el radio espectral es igual a

$$\mu_J = \cos(\alpha) = -\cos(I\alpha) \tag{11.5.13}$$

y se puede aproximar por

$$\mu_J \simeq 1 - \frac{1}{2}\alpha^2 = 1 - \frac{1}{2}\left(\frac{\pi}{I+1}\right)^2 \tag{11.5.14}$$

donde $\alpha = \pi/(I + 1)$. La tasa de convergencia de un método iterativo se define como

$$R = -\log_{10} \mu$$

Para el caso del método iterativo de Jacobi en un problema unidimensional, la tasa de convergencia es

$$\begin{aligned} R &= -\log_{10} \mu_J \\ &\simeq -\log_{10}\left[1 - \frac{1}{2}\left(\frac{\pi}{I+1}\right)^2\right] \simeq \frac{1}{2}\left(\frac{\pi}{I+1}\right)^2 \Big/ \ln(10) \end{aligned} \tag{11.5.15}$$

Se observa que la tasa de convergencia del método iterativo de Jacobi es una función que sólo depende del número de puntos en la retícula. Al crecer I, la tasa de convergencia R tiende a cero.

Convergencia del SOR. El análisis de la tasa de convergencia del SOR es más complicado que para el método iterativo de Jacobi, puesto que entre los valores propios hay parejas de complejos conjugados.

Si consideramos el mismo problema unidimensional que utilizamos para el método iterativo de Jacobi, SOR se escribe como

$$\phi_i^{(t)} = \frac{\omega}{2}\left[h^2 S + \phi_{i-1}^{(t)} + \phi_{i+1}^{(t-1)}\right] + (1-\omega)\phi_i^{(t-1)} \tag{11.5.16}$$

En términos del error definido por la ecuación (11.5.4), (11.5.16) se transforma en

$$e_i^{(t)} = \frac{\omega}{2}\left[e_{i-1}^{(t)} + e_{i+1}^{(t-1)}\right] + (1-\omega)e_i^{(t-1)} \tag{11.5.17}$$

El problema de valores propios asociados con la ecuación (11.5.17) es

$$\xi_m\left[v_{m,i} - \frac{\omega}{2}v_{m,i-1}\right] = \frac{\omega}{2}v_{m,i+1} + (1-\omega)v_{m,i} \tag{11.5.18}$$

donde ξ_m es el m-ésimo valor propio, $v_{m,i}$ es la m-ésima función propia y las condiciones en la frontera son

$$v_{m,0} = v_{m,I+1} = 0$$

Podemos desarrollar el error en términos de las funciones propias de la ecuación (11.5.18) de la manera siguiente:

$$e_i^{(t)} = \sum_{m=1}^{I} A_m(\xi_m)^t v_{m,i} \tag{11.5.19}$$

donde los A_m son los coeficientes del desarrollo y dependen del error inicial, $e_i^{(0)}$. Puesto que la tasa de decaimiento del error está determinado por el valor más grande de $|\xi_m|$, debemos evaluar a ξ_m en seguida.

En primer lugar, mostraremos que las funciones propias de la ecuación (11.5.18) están dadas por

$$v_{m,i} = \xi_m^{i/2}\psi_{m,i} \tag{11.5.20}$$

donde $\psi_{m,i}$ es una función propia del método iterativo de Jacobi. Para la m-ésima función propia, la ecuación (11.5.18) se escribe como

$$\xi_m v_{m,i} = \frac{\omega}{2}(\xi_m v_{m,i-1} + v_{m,i+1}) + (1-\omega)v_{m,i} \tag{11.5.21}$$

Si utilizamos la ecuación (11.5.20) y (11.5.6), entonces (11.5.21) se transforma en

$$\xi_m^{(i/2)+1}\,\psi_{m,i} = \omega\xi_m^{(i+1)/2}\eta_m\psi_{m,i} + (1-\omega)\xi_m^{i/2}\psi_{m,i} \tag{11.5.22}$$

Por lo tanto, si dividimos la ecuación (11.5.22) entre $\xi_m^{i/2}\psi_{m,i}$ obtenemos

$$\xi_m = \omega\xi_m^{1/2}\eta_m + (1-\omega) \tag{11.5.23}$$

donde $\eta_m = \cos(m\alpha)$ es el valor propio de Jacobi dado por la ecuación (11.5.7b). La ecuación (11.5.23) es la ecuación característica, la cual debe cumplir ξ_m. Entonces, $\nu_{m,i}$ satisface (11.5.18), es decir, es una función propia.

Puesto que (11.5.23) es una ecuación cuadrática en $\xi_m^{1/2}$, se puede resolver de la manera siguiente:

$$\xi_m^{1/2} = \frac{\omega\eta_m}{2} \pm \sqrt{\frac{\omega^2\eta_m^2}{4} + 1 - \omega} \tag{11.5.24}$$

Al elevar al cuadrado tenemos

$$\xi_m = \frac{\omega^2\eta_m^2}{2} + 1 - \omega \pm \omega\eta_m\sqrt{\frac{\omega^2\eta_m^2}{4} + 1 - \omega} \tag{11.5.25}$$

Esta ecuación relaciona los valores propios de SOR con los correspondientes valores propios de Jacobi.

Podemos utilizar la ecuación (11.5.23) para deducir la relación entre el radio espectral del método iterativo de Jacobi, $\mu_J = \text{máx } |\eta_m|$, y el radio espectral de SOR, $\mu_\omega = \text{máx } |\xi_m|$. Como se puede ver de la ecuación (11.5.7b), el valor máximo de η_m es real y positivo. Se puede mostrar que el valor máximo de ξ_m es real y positivo si $\omega \leqslant \omega_{ópt}$, donde $\omega_{ópt}$ es el valor óptimo de ω (lo cual se explicará posteriormente). Cuando μ_J y μ_ω son reales y positivos, podemos escribir la ecuación (11.5.23) para los radios espectrales como

$$\mu_\omega = \omega\mu_\omega^{1/2}\mu_J + (1-\omega) \tag{11.5.25a}$$

Por otro lado, si $\omega > \omega_{ópt}$, la ecuación (11.5.25a) no se cumple, puesto que el valor propio de SOR correspondiente a $\mu_J = \text{máx } |\eta_m|$ se vuelve complejo. Sin embargo, al incluir ambos casos, el radio espectral de SOR se relaciona con el radio espectral de Jacobi mediante la fórmula

$$\mu_\omega = \left|\frac{\omega^2\mu_J^2}{2} + 1 - \omega + \omega\mu_J\sqrt{\frac{\omega^2\mu_J^2}{4} + 1 - \omega}\right| \tag{11.5.26}$$

Si $\omega > \omega_{ópt}$, el interior del valor absoluto es una raíz compleja de (11.5.25a). El valor absoluto es necesario para calcular el radio espectral. Si $\omega \leqslant \omega_{ópt}$, μ_ω es igual a la raíz más grande de (11.5.25a), la cual es positiva. Por lo tanto, el valor absoluto del lado derecho de (11.5.26) no es necesario.

Para analizar la distribución de ξ_m dada por (11.5.25) y el efecto de ω sobre μ_ω, primero recordemos que todos los η_m son reales y que $-\mu_J \leqslant \eta_m \leqslant \mu_J$.

Definimos ω como la raíz más grande de

$$\frac{1}{4}\overline{\omega}^2\mu_J^2 + 1 - \overline{\omega} = 0$$

Entonces, para cualquier η_m que satisfaga $|\eta_m| < \mu_J$

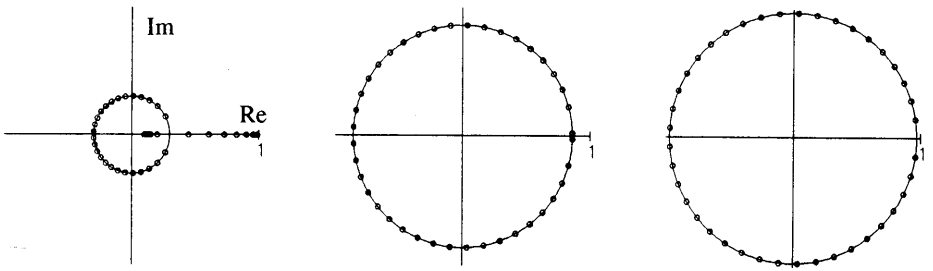
$$\frac{1}{4}\overline{\omega}^2\eta_m^2 + 1 - \overline{\omega} < 0$$

Si $\omega = \overline{\omega}$, el término con la raíz cuadrada de (11.5.25) es imaginario, excepto cuando $\eta_m = \pm\mu_J$ y se puede mostrar fácilmente que $|\xi_m| = \omega - 1$ para cada m. Así, si $\omega = \omega$, entonces $\mu_\omega = \overline{\omega} - 1 \equiv \mu_{\overline{\omega}}$.

Si $\omega > \overline{\omega}$, en la ecuación (11.5.25), el término con la raíz cuadrada se vuelve imaginario y $|\xi_m| = \omega - 1$ para toda m, por lo que $\mu_\omega = \omega - 1$ se vuelve mayor que $\overline{\omega} - 1$.

Si $\omega < \omega$, el término con la raíz se vuelve imaginario únicamente para aquellos η_m que cumplan $\omega^2\eta^2/4 + 1 - \omega < 0$, pero es real para los demás η_m, incluyendo a $\eta_m = \pm\mu_J$. El valor real más grande de (11.5.25), igual a μ_ω, aparece cuando $\eta_m = \mu_J$. Este valor μ_ω es más grande que μ_ω.

Así, el mínimo valor posible de μ_ω es igual a $\mu_{\overline{\omega}}$, por lo que $\overline{\omega}$ se llama el óptimo ω y se denota por $\omega_{\text{ópt}}$. Estos tres casos de w se muestran de manera gráfica en la figura 11.9.



$\omega < \omega_{\text{ópt}}$: $\mu_\omega > \mu_{\omega\,\text{ópt}}$ $\omega = \omega_{\text{ópt}}$: $\mu_\omega = \mu_{\omega\,\text{ópt}}$ $\omega > \omega_{\text{ópt}}$: $\mu_\omega > \mu_{\omega\,\text{ópt}}$

Figura 11.9 Distribución de los valores propios del SOR para los tres valores de ω

Como ya se explicó, $\omega_{ópt}$ satisface $\frac{1}{4}(\omega_{\text{ópt}})^2\mu_J^2 + 1 - \omega_{\text{ópt}} = 0$

o bien, resolviendo la ecuación anterior de segundo grado,

$$\omega_{\text{ópt}} = \frac{2}{1 + \sqrt{1 - \mu_J^2}} \tag{11.5.27}$$

Con este óptimo $\omega_{\text{ópt}}$, μ_ω se transforma en [véase la ecuación (11.5.26)]

$$\mu_{\omega,\,\text{ópt}} = \omega_{\text{ópt}} - 1 = \frac{2}{1+\sqrt{1-\mu_J^2}} - 1$$
$$= \frac{1-\sqrt{1-\mu_J^2}}{1+\sqrt{1-\mu_J^2}} \tag{11.5.28}$$

Las ecuaciones (11.5.26) a (11.5.28) no sólo son válidas para el modelo unidimensional, sino para los problemas de dos y tres dimensiones. También se aplican a la relación entre el método iterativo de Jacobi por líneas y el SOR por líneas. Los resultados del ejemplo 11.3 muestran una tendencia importante de los valores propios y del efecto de ω sobre la tasa de convergencia del SOR.

Ejemplo 11.3

Si $I = 20$ en las ecuaciones (11.5.2) y (11.5.3), evalúe todos los valores propios del método iterativo de Jacobi. A continuación, evalúe $\omega_{ópt}$ para el SOR aplicado al mismo problema, utilizando la ecuación (11.5.28). Calcule todos los valores propios de SOR para los siguientes valores de ω: $\omega = 1.2$, 1.5, $\omega_{ópt}$ y 1.8. Grafíquelos en el plano complejo.

(Solución)

Los valores propios de Jacobi están dados por (11.5.7b). En la segunda columna de la tabla E11.3 se listan los valores propios de Jacobi η_m calculados para $\alpha = \pi/21 = 0.149559$, lo que muestra que $\mu_J = 0.98883$. Todos los valores propios del método iterativo de Jacobi aparecen por parejas, cada una de las cuales está formada por un valor propio positivo y uno negativo de la misma magnitud.

El $\omega_{ópt}$ dado por (11.5.27) es

$$\omega_{\text{ópt}} = \frac{2}{1+\sqrt{1-0.98883^2}} = 1.74057$$

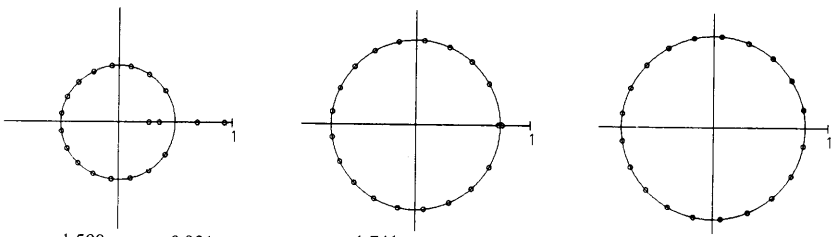
Tabla E11.3 Valores propios del SOR

m	η_m	$\omega = 1.2$	$\omega = 1.5$	$\omega = \omega_{\text{ópt}} = 1.741$	$\omega = 1.8$
1	± 0.988	$\underline{0.966}$, 0.041	$\underline{0.931}$, 0.268	$\underline{0.741}$, 0.741	$0.784 \pm 0.159j$
2	± 0.956	0.869, 0.046	0.695, 0.360	$0.642 \pm 0.368j$	$0.679 \pm 0.423j$
3	± 0.901	0.713, 0.056	$0.413 \pm 0.281j$	$0.489 \pm 0.556j$	$0.515 \pm 0.612j$
4	± 0.826	0.504, 0.079	$0.268 \pm 0.422j$	$0.293 \pm 0.680j$	$0.305 \pm 0.739j$
5	± 0.733	$0.187 \pm 0.071j$	$0.105 \pm 0.489j$	$0.073 \pm 0.736j$	$0.071 \pm 0.797j$
6	± 0.623	$0.080 \pm 0.183j$	$-0.063 \pm 0.496j$	$-0.152 \pm 0.725j$	$-0.170 \pm 0.782j$
7	± 0.500	$-0.020 \pm 0.199j$	$-0.219 \pm 0.450j$	$-0.362 \pm 0.646j$	$-0.395 \pm 0.696j$
8	± 0.365	$-0.104 \pm 0.171j$	$-0.350 \pm 0.357j$	$-0.538 \pm 0.508j$	$-0.584 \pm 0.547j$
9	± 0.223	$-0.164 \pm 0.114j$	$-0.444 \pm 0.229j$	$-0.665 \pm 0.325j$	$-0.720 \pm 0.349j$
10	± 0.075	$-0.196 \pm 0.040j$	$-0.494 \pm 0.079j$	$-0.732 \pm 0.112j$	$-0.791 \pm 0.120j$
	$\mu_J = 0.988$	$\mu_\omega = 0.966$	$\mu_\omega = 0.931$	$\mu_\omega = 0.741$	$\mu_\omega = 0.8$

En la misma tabla se muestran los valores propios del SOR para cada $\omega = 1.2$, 1.5, $\omega_{ópt} = 1.74057$ y $\omega = 1.8$. Aunque cada valor propio de Jacobi da como resultado dos valores propios de SOR, al examinar la tabla se observa que una pareja de valores propios (negativo y positivo) del método iterativo de Jacobi produce los mismos pares de valores propios del SOR. Así, el número total de valores propios del SOR es idéntico al de los valores propios de Jacobi.

Se ve que los valores propios del SOR para $\omega = 1.5$ incluyen valores reales y complejos. Para $\omega = \omega_{ópt}$ todos los valores propios son complejos excepto los correspondientes a μ_J que son raíces reales dobles. Si $\omega > \omega_{ópt}$, todos los valores propios son complejos.

En la figura E11.3 se muestra una gráfica de los valores propios para tres valores de ω en el plano complejo, en analogía con la figura 11.9. El radio espectral (RE) de cada caso aparece debajo de cada gráfica (véanse también los valores propios subrayados en la tabla 11.3, los cuales corresponden al radio espectral). Conviene observar que los valores complejos de cada caso están en un círculo de radio $\omega - 1$. La mayor distancia entre un valor propio y el origen es el radio espectral. Para $\omega = 1.5$, el máximo valor propio real constituye el radio espectral. Por otro lado, si $\omega > \omega_{ópt}$ los valores propios están en el círculo de radio $\omega - 1$, por lo que el radio espectral es igual a $\omega - 1$ y es independiente de los valores propios de Jacobi. Si $\omega = \omega_{ópt}$, una pareja de valores propios dobles se encuentra sobre el eje real. En este caso, el radio espectral es igual a $\omega_{ópt} - 1$, que es menor que el valor correspondiente a los otros dos casos de ω.



$\omega = 1.500$, $\mu_\omega = 0.931$ $\omega = 1.741$, $\mu_\omega = 0.741$ $\omega = 1.800$, $\mu_\omega = 0.800$

Figura E11.3 Distribución de los valores propios del SOR para los tres valores de omega (espectro)

Convergencia del método Iterativo Extrapolado de Jacobi. El EJ para un modelo unidimensional es

$$\phi_i^{(t)} = \frac{\theta}{2}\left[h^2 S + \phi_{i-1}^{(t-1)} + \phi_{i+1}^{(t-1)}\right] + (1-\theta)\phi_i^{(t-2)} \tag{11.5.29}$$

En términos del error definido por la ecuación (11.5.4), (11.5.29) se transforma en

$$e_i^{(t)} = \frac{\theta}{2}\left[e_{i-1}^{(t-1)} + e_{i+1}^{(t-1)}\right] + (1-\theta)e_i^{(t-2)} \tag{11.5.30}$$

Podemos desarrollar el error en términos de las funciones propias del método iterativo de Jacobi de la manera siguiente:

$$e_i^{(t)} = \sum_{m=1}^{I} A_m \zeta_m^t \psi_{m,i} \tag{11.5.31}$$

donde $\psi_{m,i}$ es la función propia del método iterativo de Jacobi, dada por (11.5.7) y ζ_m es el valor propio en este método para la m-ésima función propia. (El EJ comparte las mismas funciones propias, pero sus valores propios son distintos.)

Sustituimos la ecuación (11.5.31) en (11.5.30), con lo que tenemos

$$\zeta_m^t = \theta \eta_m \zeta_m^{t-1} + (1-\theta)\zeta_m^{t-2} \tag{11.5.32}$$

donde η_m es el m-ésimo valor propio de Jacobi; utilizamos (11.5.6) para obtener el primer término del lado derecho. Si dividimos entre ζ_m^{t-2} obtenemos

$$\zeta_m^2 = \theta \eta_m \zeta_m + (1-\theta) \tag{11.5.33}$$

Esta es la ecuación característica que debe satisfacer ζ_m y tiene exactamente la misma forma que (11.5.23), excepto que $\xi_m^{1/2}$ se cambia por ζ_m. Al resolver la ecuación cuadrática (11.5.33), ζ_m es

$$\zeta_m = \frac{\theta \eta_m}{2} \pm \sqrt{\frac{\theta^2 \eta_m^2}{4} + 1 - \theta} \tag{11.5.34}$$

Queremos entonces determinar θ de manera que minimice el radio espectral, definido por

$$\mu_\theta = \max_m \zeta_m \tag{11.5.35}$$

Se puede demostrar, mediante un análisis similar al realizado para la ecuación (11.5.26), que θ es óptimo cuando el término de la raíz cuadrada de (11.5.34) se anula en $\eta_m = \mu_J$. Así, $\theta_{ópt}$ es

$$\theta_{\text{ópt}} = \frac{2}{1 + \sqrt{1 - \mu_J^2}} \tag{11.5.36}$$

Nótese que $\theta_{ópt}$ es idéntico a $\omega_{ópt}$ para SOR. Con este valor de $\theta_{ópt}$, el radio espectral de EJ es

$$\mu_{\theta_{\text{ópt}}} = \frac{\mu_J}{1 + \sqrt{1 - \mu_J^2}} \tag{11.5.37}$$

Elevamos al cuadrado la ecuación (11.5.37) para obtener

$$(\mu_{\theta_{\text{ópt}}})^2 = \frac{(\mu_J)^2}{(1+\sqrt{1-\mu_J^2})^2}$$
$$= \frac{1-\sqrt{1-\mu_J^2}}{1+\sqrt{1-\mu_J^2}} \tag{11.5.38}$$

que es igual a $\mu_{\omega_{ópt}}$ dado por (11.5.28). Así,

$$(\mu_{\theta_{\text{ópt}}})^2 = \mu_{\omega_{\text{ópt}}} \tag{11.5.39}$$

Este resultado indica que la tasa de convergencia de EJ con $\theta_{ópt}$ es precisamente la mitad de la de SOR con $\omega_{ópt}$. Sin embargo, el esfuerzo computacional por cada ciclo de iteración del primero es la mitad del correspondiente al segundo. Por lo tanto, el tiempo total de cómputo para EJ es el mismo que el de SOR.

Resumen de Esta Sección

a) La tasa de convergencia del método iterativo de Jacobi se analiza mediante las funciones propias de dicho método.
b) El parámetro óptimo del SOR se expresa en términos del radio espectral del método iterativo de Jacobi.
c) El parámetro óptimo de extrapolación para EJ, $\theta_{ópt}$, es igual al parámetro óptimo del SOR, $\omega_{ópt}$.
d) Con este parámetro óptimo, la tasa de convergencia de EJ es la mitad de la del SOR en términos del número de iteraciones. Sin embargo, la cantidad de trabajo computacional por iteración del primero es la mitad de la del segundo. Por lo tanto, la eficiencia computacional total de EJ es idéntica a la del SOR.
e) Sin embargo, como se explicó anteriormente, el EJ es varias veces más rápido en una supercomputadora con procesador vectorial.
f) A pesar del modelo sencillo empleado en esta subsección, las ecuaciones para los parámetros óptimos y el radio espectral en los tres métodos no se alteran para los problemas más generales en dos y tres dimensiones.

11.6 COMO OPTIMIZAR LOS PARAMETROS DE ITERACION

A pesar del modelo tan sencillo que utilizamos en la sección anterior, las ecuaciones ahí obtenidas se aplican al SOR puntual y por líneas en dos y tres dimensiones. En esta sección se explica cómo se determina μ_J para un problema dado y cómo optimizar las tasas de convergencia de SOR y EJ.

En cualquier esquema iterativo podemos expresar el error de la solución iterativa mediante la superposición de modos espaciales con distintas tasas de decaimiento:

$$e_{i,j}^{(t)} = \sum_{m=0}^{M} a_m(\gamma_m)^t u_{i,j}^{(m)} \tag{11.6.1}$$

donde t es el contador de las iteraciones, $u_{i,j}^{(m)}$ es el m-ésimo modo espacial (o función propia), γ_m es su factor de amplitud (valor propio) y $M + 1$ es el número total de vectores propios. Los coeficientes a_m dependen de la estimación inicial. De hecho, la ecuación (11.6.1) corresponde al caso unidimensional de las ecuaciones (11.4.1) y (11.4.19). Aquí suponemos que

$$|\gamma_0| = \max_m |\gamma_m| = \mu \tag{11.6.2}$$

γ_0 recibe el nombre de *valor propio dominante*, mientras que $u_{i,j}^{(0)}$ es la *función propia dominante*.

Todos los términos tienden a cero cuando t crece, en el caso en que cada γ_m cumpla $|\gamma_m| < 1$. De los términos en el lado derecho de (11.6.1), mientras más pequeño sea el valor de $|\gamma_m|$, más rápido será el decaimiento a cero. El término con mayor valor $|\gamma_m|$ es el que tiende a cero más lentamente. Cuando utilizamos SOR, con frecuencia ocurre que el error de la solución iterativa decae muy rápido en las primeras iteraciones, pero se va frenando de manera gradual, hasta llegar a cierta constante.* Este fenómeno aparece cuando ω es menor que el óptimo y es precisamente lo que indica la ecuación (11.6.1). La rápida razón de decaimiento del error en la etapa inicial se debe a la contribución de los $|\gamma_m|$ pequeños. Al desaparecer éstos, la tasa de decaimiento del error queda determinada finalmente por el valor propio dominante γ_0. Si ω es el óptimo, todos los $|\gamma_m|$ del SOR son idénticos.

Sin embargo, observemos un fenómeno peculiar que puede ocurrir si se utiliza $\omega_{\text{ópt}}$: en este caso, el error puede crecer hasta que se llega a cierto número de iteraciones. Después de este punto, el error decrecerá a la velocidad óptima. La responsable de este crecimiento temporal de los errores es la deficiencia del vector propio, que aparece cuando $\omega = \omega_{\text{ópt}}$ [Wachpress].

La tasa de convergencia de un esquema iterativo está dada por

$$R = -\log_{10}(\mu) \tag{11.6.3}$$

El error decrece después de N pasos de iteración según el factor

$$\beta = (\mu)^N \tag{11.6.4}$$

Por ejemplo, si β decrece hasta 10^{-5}, el número necesario de ciclos de iteración es $N = 5/R$. La tabla 11.3 muestra el número de ciclos necesarios para $\beta = 10^{-5}$.

El radio espectral de Jacobi μ_J aumenta si el número de puntos en la retícula crece; aquél también se ve afectado por las condiciones en la frontera. Para una geometría dada, con un número fijo de puntos en la retícula, μ_J es mínimo cuando

* Es importante que se utilice una escala semilogarítmica para graficar los errores iterativos contra el número de iteraciones.

Tabla 11.3 Efecto de μ en el número de pasos de iteración necesarios para reducir el error en un factor de 10^{-5}

μ	R	N
0.5	0.3	17
0.7	0.15	32
0.9	0.045	109
0.95	0.022	224
0.99	0.0043	1145
0.999	0.00043	11507

todas las condiciones en la frontera son del tipo del valor fijo. Cuando aumenta la porción de las fronteras con condiciones del tipo derivada, aumenta también μ_J. El efecto de las condiciones de tipo mixto es intermedio entre los efectos de los otros tipos de condiciones.

Los efectos del número de puntos en la retícula y los tipos de condiciones en la frontera sobre el radio espectral del SOR son similares a los efectos sobre el método iterativo de Jacobi. Sin embargo, el radio espectral del SOR también depende de manera significativa del parámetro de iteración ω. Se relaciona con μ_J y ω mediante la ecuación

$$\mu_\omega = \left|\frac{1}{2}\omega^2\mu_J^2 - \omega + 1 + \omega\mu_J\sqrt{\frac{1}{4}\omega^2\mu_J^2 - \omega + 1}\right| \tag{11.6.5}$$

Para graficar μ_ω contra ω en la ecuación (11.6.5), son útiles las siguientes relaciones. Cuando $\omega < \omega_{ópt}$, el término con la raíz cuadrada en la ecuación (11.6.5) es real y lo que se encuentra dentro del valor absoluto es real y positivo, por lo que

$$\mu_\omega = \frac{1}{2}\omega^2\mu_J^2 + 1 - \omega + \omega\mu_J\sqrt{\frac{1}{4}\omega^2\mu_J^2 - \omega + 1} \tag{11.6.6}$$

Si $\omega > \omega_{ópt}$, el término con la raíz cuadrada en (11.6.6) se vuelve imaginario y el radio espectral es

$$\mu_\omega = \omega - 1 \tag{11.6.7}$$

La figura 11.10 muestra las gráficas de (11.6.5) contra ω. Se observa que μ_ω es muy sensible a ω en torno de $\omega = \omega_{ópt}$: decrece en forma rápida cuando ω tiende a $\omega_{ópt}$ por abajo. Esta es la razón de la importancia de determinar $\omega_{ópt}$ al maximizar la tasa de convergencia del SOR.

Optimización de ω. Podemos determinar con relativa facilidad el valor óptimo de ω en un problema dado, realizando una ejecución piloto del SOR con una estimación me-

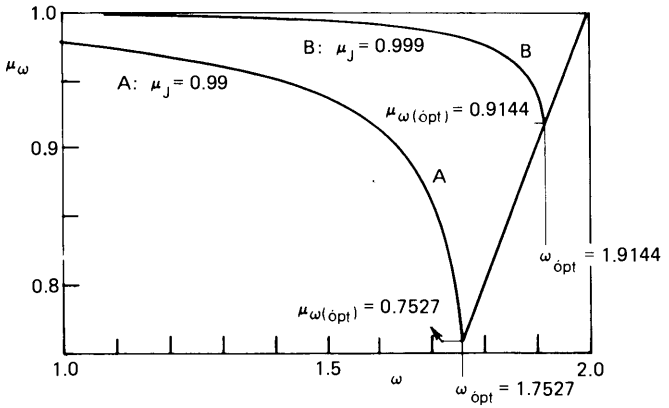


Figura 11.10 Efecto de ω en el radio espectral del SOR

nor de ω. Si $\omega < \omega_{ópt}$, el valor propio dominante del SOR se vuelve real y positivo (igual a μ_ω), como en el ejemplo 11.3. Los demás valores propios tienen una magnitud menor que la del valor propio dominante. Por lo tanto, al crecer el número de ciclos de iteración, (11.6.1) tiende a

$$e_{i,j}^{(t)} = \sum_{m=0}^{M} a_m \gamma_m^t u_{i,j}^{(m)} \longrightarrow a_0 \mu_\omega^t u_{i,j}^{(0)} \tag{11.6.8}$$

donde se supone que $u_{i,j}^{(0)}$ y γ_0 son la función propia dominante y su correspondiente valor propio.

El número definido por

$$N^{(t)} = \sum_{i,j} [\phi_{i,j}^{(t)} - \phi_{i,j}^{(t-1)}]^2 \tag{11.6.9}$$

se puede calcular fácilmente en cada recorrido de la iteración. Sustituimos

$$e_{i,j}^{(t)} = (\phi_{i,j})_{\text{exacto}} - \phi_{i,j}^{(t)} \tag{11.6.10}$$

en la ecuación (11.6.9) para obtener

$$\begin{aligned} N^{(t)} &= \sum_{i,j} [(e_{i,j}^{(t)} - e_{i,j}^{(t-1)}]^2 \\ &\simeq \sum_{i,j} [a_0 \mu_\omega^{t-1} (\mu_\omega - 1) u_{i,j}^{(0)}]^2 \end{aligned} \tag{11.6.11}$$

en donde utilizamos la ecuación (11.6.8). La razón de $N^{(t)}$ en dos ciclos consecutivos es

$$\frac{N^{(t)}}{N^{(t-1)}} \simeq \mu_\omega^2 \tag{11.6.12}$$

Así, en cada iteración se tiene la estimación de μ_ω dada por

$$\mu_\omega^{(t)} = \sqrt{\frac{N^{(t)}}{N^{(t-1)}}} \tag{11.6.13}$$

Una vez obtenido μ_ω se puede calcular una estimación de μ_J mediante la ecuación (11.5.25a) de la manera siguiente:

$$\mu_J^{(t)} = \frac{\mu_\omega^{(t)} + \omega - 1}{\omega\sqrt{\mu_\omega^{(t)}}} \tag{11.6.14}$$

Además, utilizamos la ecuación (11.5.27) para estimar $\mu_{ópt}$:

$$\omega_{\text{ópt}}^{(t)} = \frac{2}{1 + \sqrt{1 - (\mu_J^{(t)})^2}} \tag{11.6.15}$$

Al principio, la precisión de estas estimaciones es pobre, pero mejora al incrementarse el número de ciclos de iteración. El algoritmo de las ecuaciones (11.6.9) a la (11.6.15) se puede incorporar en cualquier programa del SOR puntual o por rectas. Se puede estimar $\omega_{ópt}$ con una precisión razonable durante los primeros pasos de iteración (por ejemplo, a lo más 20). Entonces, la iteración continúa con $\omega = \omega_{ópt}$ hasta lograr la convergencia. Si se resuelve la misma ecuación varias veces con distintos términos fuente, el valor de $\omega_{ópt}$ determinado en el primer problema permanece idéntico para los demás ($\omega_{ópt}$ no se ve afectado por el término fuente). Es frecuente que haya una solución repetitiva del mismo problema elíptico en los cálculos para reactores nucleares, así como en dinámica de fluidos.

En el PROGRAMA 11-3 se ilustra una implantación del algoritmo de optimización.

ESTIMACIÓN DE $\theta_{ópt}$ PARA EL MÉTODO ITERATIVO EXTRAPOLADO DE JACOBI (EJ). Puesto que $\theta_{ópt} = \omega_{ópt}$, podemos utilizar el algoritmo para determinar este último. Sin embargo, el algoritmo descrito arriba no se puede vectorizar. Así, es mejor utilizar un algoritmo basado en el propio EJ.

La suma

$$N^{(t)} = \sum_{i,j} [(\phi_{i,j}^{(t)} - \phi_{i,j}^{(t-2)}]^2 \tag{11.6.16}$$

se calcula para cada ciclo impar t, donde la suma se toma sólo sobre la mitad de los puntos de la retícula, para los cuales se hacen realmente los cálculos. La razón

$N^{(t)}/N^{(t-2)}$ converge a

$$\frac{N^{(t)}}{N^{(t-2)}} \longrightarrow \mu_\theta^4$$

Así, en cada iteración se tiene una estimación de μ_θ dada por

$$\mu_\theta^{(t)} = \left(\frac{N^{(t)}}{N^{(t-2)}}\right)^{1/4} \tag{11.6.17}$$

Utilizamos μ_θ para calcular una estimación de μ_J

$$\mu_J^{(t)} = \frac{(\mu_\theta^{(t)})^2 + \theta - 1}{\theta \mu_\theta^{(t)}} \tag{11.6.18}$$

Así, una estimación de $\theta_{ópt}$ es

$$\theta_{\text{ópt}}^{(t)} = \frac{2}{1 + \sqrt{1 - (\mu_J^{(t)})^2}} \tag{11.6.19}$$

que es exactamente igual a la ecuación (11.6.15).

En el PROGRAMA 11-4, se demuestra una implantación del algoritmo recién descrito.

Resumen de Esta Sección

a) Se estima un parámetro óptimo del SOR —así como el correspondiente del EJ— ejecutando ambos métodos con una estimación por debajo del valor del parámetro.
b) Se puede realizar la optimización de ω o de θ al principio de la solución iterativa, de forma que el resto de la iteración se lleve a cabo con el parámetro optimizado.

11.7 METODO IMPLICITO DE LA DIRECCION ALTERNANTE (IDA)

El IDA es un método de solución iterativa para un conjunto grande de ecuaciones en diferencias para una ecuación diferencial parcial elíptica. Aunque este método requiere un esfuerzo mayor para desarrollar un programa, es popular debido a que la tasa de convergencia resulta en general más rápida que la del SOR, sobre todo para los problemas grandes. En los últimos años se ha renovado el interés en IDA porque es adecuado para las supercomputadoras con procesador vectorial. Sus ventajas son:

a) Su eficiencia computacional es mayor que la del SOR para una amplia gama de problemas.
b) Las condiciones en la frontera no son tan restrictivas como las del SOR o del TRF.
c) Es más adecuado para los procesadores vectoriales de una supercomputadora.

Por otro lado, sus desventajas son:

a) La programación es más complicada que la del SOR.
b) La optimización de los parámetros de iteración es mucho más difícil que la del SOR.
c) La geometría se restringe al caso de una retícula rectangular.

El IDA que explicaremos aquí se basa en el método de factorización aproximada y está íntimamente ligado con el IDA del capítulo 12.

Para hacer más sencilla la explicación, consideremos la ecuación de Poisson

$$(L_x + L_y)\phi(x, y) = S \tag{11.7.1}$$

donde L_x y L_y son operadores diferenciales,

$$L_x = -\frac{\partial^2}{\partial x^2}, \quad L_y = -\frac{\partial^2}{\partial y^2} \tag{11.7.2}$$

La esencia de IDA es la separación de variables en el operador iterativo, lo cual se puede explicar antes de cambiar a las aproximaciones por diferencias. Sin embargo, debemos observar que L_x y L_y se remplazarán por los operadores de diferencias centrales con tres puntos.

El método iterativo para la ecuación (11.7.1) se puede escribir en la forma

$$M\phi^{(t+1)} = M\phi^{(t)} + \theta[S - (L_x + L_y)\phi^{(t)}(x, y)] \tag{11.7.3}$$

donde Mes un procesador que aproxima a $L_x + L_y$ pero que hace más fácil la solución, $\phi^{(t)}$ es la t-ésima iteración y θ es un parámetro de extrapolación, que será igual a 2 en lo sucesivo.

Si definimos

$$\delta\phi(x, y) = \phi^{(t+1)} - \phi^{(t)} \tag{11.7.4}$$

podemos reescribir la ecuación (11.7.3) como

$$M\delta\phi(x, y) = 2R^{(t)} \tag{11.7.5}$$

donde R es el t-ésimo residual, definido por

$$R^{(t)} = [S - (L_x + L_y)\phi^{(t)}(x, y)]$$

La tasa de convergencia y la eficiencia de un método iterativo dependen de la elección del operador iterativo M. Deseamos que M sea fácil de resolver y que sea una buena aproximación de $L_x + L_y$, como mencionamos anteriormente. El M del IDA se escribe como

$$M = \frac{1}{\omega}(\omega + L_x)(\omega + L_y) \tag{11.7.6}$$

donde ω es un parámetro de aceleración que varía en cada iteración para optimizar la tasa de convergencia. Con esta forma, los operadores de x y y se separan. Reescribimos la ecuación (11.7.6) como

$$M = \omega + L_x + L_y + \frac{1}{\omega} L_x L_y \tag{11.7.7}$$

en donde podemos ver que M está formado por los operadores diferenciales verdaderos $(L_x + L_y)$ más otros dos términos. Por lo tanto, podemos pensar que M es una aproximación del operador diferencial real.

Sustituimos la ecuación (11.7.7) en (11.7.5) y obtenemos

$$(\omega + L_x)(\omega + L_y)\delta\phi(x, y) = 2\omega R^{(t)} \tag{11.7.8}$$

Puesto que los operadores diferenciales en x y y están en una forma factorizada, podemos resolver (11.7.8) en dos etapas. La primera es resolver

$$(\omega + L_x)\psi(x, y) = 2\omega R^{(t)} \tag{11.7.9}$$

y la segunda es resolver

$$(\omega + L_y)\delta\phi(x, y) = \psi(x, y) \tag{11.7.10}$$

Para cualquier y, (11.7.9) es un problema unidimensional con valores en la frontera, por lo que podemos obtener sus aproximaciones por diferencias mediante la solución tridiagonal a lo largo de cada línea de la retícula en la dirección de x. La ecuación (11.7.10) es un problema unidimensional con valores en la frontera pero en la dirección de y.

La elección de los parámetros de aceleración es importante para una rápida convergencia. Teóricamente, es posible determinar los parámetros óptimos de aceleración sólo para problemas ideales sencillos (es decir, dominios rectangulares con coeficientes constantes, espaciamiento uniforme en la retícula y condiciones en la frontera del tipo del valor fijo). Para cualquier problema más complejo, se debe estimar un valor óptimo mediante el conocimiento de los casos ideales. Esto es una desventaja del IDA en comparación con el SOR. No obstante, el IDA sigue siendo útil debido a que converge más rápido que el SOR, aun con estimaciones imperfectas de los parámetros óptimos de aceleración. Para los problemas prácticos, seguiremos el siguiente método.

Para una tasa máxima de convergencia, se eligen un conjunto de K valores de ω, que se utilizan en forma secuencial en un ciclo de K pasos de iteración. El valor de K es, por lo general, de 4 a 8. Este conjunto se utiliza en forma cíclica hasta que la iteración converge. Dichos valores son de la forma

$$\omega_k = \alpha_0 \left(\frac{\alpha_1}{\alpha_0}\right)^{(2k-1)/2K}, \quad k = 1, 2, 3, \ldots, K \tag{11.7.3}$$

donde

$$\alpha_0 = \frac{p\pi^2}{[\text{máximo de } V \text{ y } H]^2} \tag{11.7.4}$$

$$\alpha_1 = \frac{4p}{\left[\text{mínimo de } \dfrac{\Delta x}{H} \text{ y } \dfrac{\Delta y}{V}\right]^2}$$

En estas expresiones, H y V son las longitudes horizontales y verticales del dominio y se supone que las condiciones en la frontera son del tipo de valor fijo. Si la condición de la frontera izquierda o derecha es del tipo derivada, la H de (11.7.4) debe cambiarse por el doble de la longitud que se tenga ese momento. Si ambas son del tipo derivada, sugerimos que se cambie H a 4 veces el tamaño que tiene en ese momento. El valor de V cambia de forma análoga.

Resumen de Esta Sección

a) En el IDA, la solución de un conjunto de ecuaciones en diferencias se reduce a dos conjuntos de problemas unidimensionales con valores en la frontera.
b) Las ecuaciones en diferencias a lo largo de una línea de la retícula se resuelven mediante el esquema de solución tridiagonal.
c) El IDA se puede vectorizar de dos maneras: la primera, con un esquema vectorizado de solución tridiagonal [Kershaw]; la segunda, llevando a cabo las soluciones tridiagonales en forma paralela para todas las líneas (sin vectorizar cada esquema tridiagonal). El último es más eficiente que la primera, aunque necesita más espacio de memoria principal.

11.8 METODOS DE SOLUCION DIRECTA

11.8.1 Eliminación de Gauss con base en una estructura de banda

La matriz de coeficientes de la aproximación por diferencias de una EDP elíptica tiene una estructura pentadiagonal, como la de la matriz M de la figura 11.11. Podemos utilizar tanto la eliminación de Gauss como la descomposición LU para resolver las ecuaciones en diferencias del tipo elíptico. Sin embargo, la descomposición LU tiende a ser más eficiente que la eliminación de Gauss cuando hay que volver a determinar la solución con distintos términos de homogéneos.

Al aplicarle la descomposición LU a dicha matriz, es importante observar que ésta es una matriz con una banda diagonal de semiancho ω, donde ω es igual al número de puntos en la retícula en la dirección de x, más uno (si se supone que se utiliza el mismo sistema de numeración del ejemplo 11.1). Al hacer la descomposición, las matrices L y U también tienen estructuras diagonales por bandas, como lo muestra la figura 11.11. La matriz L será una matriz triangular inferior, cuyos ele-

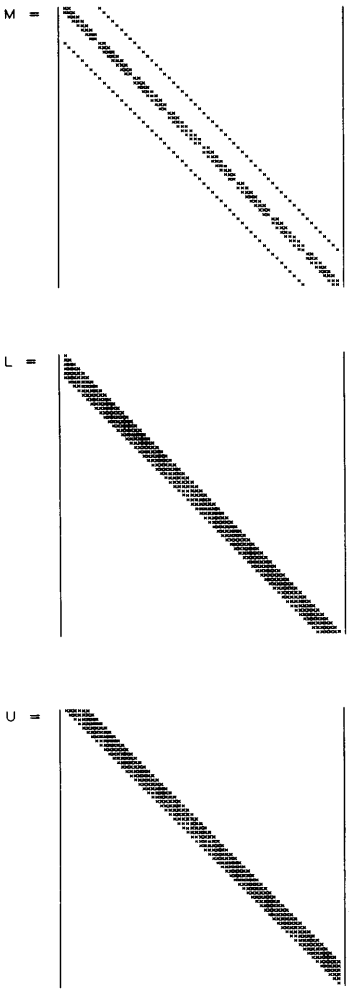


Figura 11.11

mentos distintos de cero se encuentran únicamente sobre la banda. La matriz U será una matriz triangular superior con elementos no nulos sólo sobre la banda. La descomposición LU y la eliminación de Gauss para el caso de las ecuaciones en diferencias de tipo elíptico son en esencia iguales al caso de las matrices ordinarias, excepto que en la memoria se guardan sólo los elementos de la banda.

Para las ecuaciones elípticas en diferencias, no es necesario el pivoteo, puesto que éstas tienen dominancia diagonal.

Una matriz A de banda diagonal se puede almacenar en un espacio de memoria de $(2\omega - 1)N$, que es mucho menor que N^2.

Si la matriz de coeficientes es simétrica, entonces se pueden elegir L y U de forma que $L = U^t$, o en forma equivalente, $A = LL^t$. El método de Cholesky [Strang]

se basa en esta propiedad y sólo utiliza la mitad del espacio central requerido para guardar tanto L como U. Véanse [Swarztrauber y Boisvert y Sweet] para los detalles.

11.8.2 Transformada rápida de Fourier

Esta se puede aplicar a las ecuaciones diferenciales parciales elípticas si se cumplen las siguientes condiciones:

a) Los coeficientes de (11.3.1) no cambian con j (o en la dirección vertical). Esto ocurre si tanto los coeficientes de la ecuación diferencial parcial elíptica como el espaciamiento son constantes en la dirección vertical.
b) Las condiciones en la frontera superior e inferior son cíclicas.
c) El número de puntos en la retícula en las direcciones verticales —sin incluir a las fronteras superior e inferior— está dado por $J = 2^l$, donde l es un entero.

El tiempo de cómputo para la TRF es menor que el de cualquier otro método iterativo, que el de los métodos de solución directa con base en la eliminación de Gauss y que el de la descomposición LU para un conjunto grande de ecuaciones en diferencias.

Como se dijo antes, existe cierta libertad para el espaciamiento de la retícula y para los coeficientes variables en la dirección horizontal. Las condiciones en la frontera tanto izquierda como derecha pueden ser de cualquier forma. También se pueden tomar coordenadas cilíndricas en la dirección horizontal (o dirección de i). Para más detalles acerca de la TRF, véase [Nussbaumer].

11.8.3 Solución directa rápida (SDR) mediante la reducción cíclica

Este método es una variante de la transformada rápida de Fourier y se aplica a las ecuaciones diferenciales parciales elípticas bajo las mismas condiciones que la TRF, excepto que:

a) Las condiciones en la frontera superior e inferior son del tipo del valor fijo.
b) El número de puntos de la retícula en las direcciones verticales —sin incluir las fronteras superior e inferior— es igual a $J = 2^l - 1$, donde l es un entero. Los números admisibles para J aparecen en la siguiente tabla:

l	J
3	7
4	15
5	31
6	63
7	127
8	255
...	...

El tiempo de cómputo de la SDR es comparable con el de la TRF. Tiene la misma libertad para el espaciamiento de la retícula en la dirección de i y para las condiciones en la frontera izquierda y derecha. Se dan más detalles de la SDR, en [Nakamura].

Resumen de Esta Sección. Si se satisfacen ciertas condiciones, los métodos de eliminación de Gauss, de transformada rápida de Fourier y de solución directa rápida pueden ser medios muy rápidos, robustos y eficientes para resolver las ecuaciones en diferencias para una EDP elíptica.

PROGRAMAS

PROGRAMA 11-1 SOR

A) Explicaciones

Este programa resuelve la ecuación de Poisson

$$-\nabla^2\phi = e^{-0.05\sqrt{(x^2+y^2)}}$$
$$0 < x < 5, \quad 0 < y < 5$$

Utiliza el SOR. Las condiciones en la frontera se definen mediante la siguiente forma general:

$$\partial\phi/\partial n = -\beta_{1,k}\phi + \beta_{2,k}$$

donde $\beta_{1,k}$ y $\beta_{2,k}$ son constantes, y k vale 1, 2, 3 y 4 para las fronteras izquierda, superior, derecha e inferior, respectivamente. Suponemos que los espaciamientos de la retícula son variables.

Escribimos las ecuaciones en diferencias en la forma de la ecuación (11.4.1), o su equivalente (11.2.28), con la siguiente definición del término no homogéneo:

$$S_{i,j} = \frac{1}{4}(h_B + h_T)(h_L + h_R)\exp\left[-0.05\sqrt{(x_i^2 + y_j^2)}\right]$$

Inicializamos los valores de B1 y B2 en las instrucciones DATA como

$$\frac{\partial\phi}{\partial n} = -\phi + 1 \quad \text{para las fronteras izquierda e inferior}$$

$$\frac{\partial\phi}{\partial n} = 0 \quad \text{para las fronteras superior y derecha}$$

B) Variables

NI y NJ: número de puntos de la retícula en las direcciones x y y, respectivamente

HX(I) y HX(J): espaciamientos de la retícula

S(I, J): $S_{i,j}$

F(I, J): $\phi_{i,j}$

WB: parámetro de iteración del SOR

EP: criterio de convergencia

EM: error máximo en cada recorrido de la iteración

X, Y: coordenadas

AR(I, J), AT(I, J): a^R y a^T para el punto (i, j) respectivamente

AC(I, J): a^c para el punto (i, j)

ITMAX: límite para el máximo de pasos de iteración

B1(K), B2(K), $k = 1, 2, 3, 4$: parámetros de condiciones en la frontera $\beta_{1,k}$ y $\beta_{2,k}$ para las fronteras izquierda, superior, derecha e inferior, respectivamente

W: parámetro del SOR

C) Listado

```
C       CSL/F11-1.FOR        SOBRERRELAJACION SUCESIVA (SOR)
        DIMENSION AR(20,20),AT(20,20),AC(20,20),F(20,20),S(20,20)
        DIMENSION HX(0:20),HY(0:20),B1(4),B2(4),SF(20)
        CHARACTER*8, CAR,CAT,CAC,CFINAL
        DATA CAR,CAT,CAC,CFINAL/'AR','AT','AC','  '/
        DATA (HX(I),I=0,6)/0,1,1,1,1,1,0/    ! Intervalos de la retícula
        DATA (HY(I),I=0,6)/0,1,1,1,1,1,0/    ! Idem
        DATA (B1(K),K=1,4)/1,0,0,1/ ! Parámetros de condiciones en la frontera: Izq., Sup.,
        DATA (B2(K),K=1,4)/1,0,0,1/ ! Idem                                      Der. e Inf.
        PRINT *,'CSL/F11-1        SOBRERRELAJACION SUCESIVA (SOR) '
        NI=6                                  ! Máximo de i
        NJ=6                        ! Máximo de j
        W=1.5                            ! Omega, parámetro del SOR
        WB=1-W
        ITMAX=200                        ! Máximo número de iteraciones
        EP=.0001                         ! Criterio de convergencia
C             Preparación de los coeficientes
        DO J=1, NJ
           HT=HY(J)
           HB=HY(J-1)
           Y=Y+HB
           X=0
           DO I=1, NI
              EC=0
              Q=0
              HL=HX(I-1)
              HR=HX(I)
              X=X+HX(I-1)
```

```
            S(I,J)=EXP(-.05*SQRT(X*X+Y*Y))*(HL+HR)*(HT+HB)/4  ! Término fuente
            AR(I,J) =0
            IF (HR.GT.0)  AR(I,J)  =(HT+HB)/HR /2
            AT(I,J) =0
            IF (HT.GT.0)  AT(I,J)  =(HL+HR)/HT /2
            IF (I.EQ.1.OR.I.EQ.NI.OR.J.eq.1.OR.J.EQ.NJ) GOTO 310
            GOTO 370
310         IF (I.EQ.1) THEN                  ! Condición en la frontera izquierda
                 EC=EC+(HB+HT)*B1(1)/2
                 Q=Q+(HB+HT)*B2(1)/2
                 END IF
            IF (I.EQ.NI) THEN                 !  Condición en la frontera derecha
                 EC=EC+(HB+HT)*B1(3)/2
                 Q=Q+(HB+HT)*B2(3)/2
                 END  IF
            IF (J.EQ.1) THEN                  ! Condición en la frontera inferior
                 EC=EC+(HL+HR)*B1(4)/2
                 Q=Q+(HL+HR)*B2(4)/2
                 END IF
            IF (J.EQ.NJ) THEN                 ! Condición en la frontera superior
                 EC=EC+(HL+HR)*B1(2)/2
                 Q=Q+(HL+HR)*B2(2)/2
                 END IF
            S(I,J)=S(I,J)+Q
            F(I,J)=0
370         AC(I,J)=AR(I-1,J)+AR(I,J)+AT(I,J-1)+AT(I,J)+EC
          END  DO
        END DO
                CALL PRNT (CAR,AR,NI,NJ,20)
                CALL PRNT (CAT,AT,NI,NJ,20)
                CALL PRNT (CAC,AC,NI,NJ,20)
C-----------------------------------------
C  Comienza SOR
        DO K=1,ITMAX          ! Comienza el ciclo de iteración
          EM=0
          DO J=1, NJ                  !     Comienza el recorrido punto por punto
            DO I=1, NI
              QQ=S(I,J)+ AR(I-1,J)*F(I-1,J)+AR(I,J)*F(I+1,J)
              QQ=QQ+AT(I,J)*F(I,J+1)  +AT(I,J-1)*F(I,J-1)
              BB=F(I,J)
              F(I,J)=W*QQ/AC(I,J)+WB*F(I,J)
              ER=ABS(BB-F(I,J))
              IF (ER.GT.EM) EM=ER
            END DO
          END DO          ! Fin del recorrido
       PRINT 90,K,EM
90     FORMAT ( ' ITER. NO.=',I4,'      ER=',1PE10.3)
       IF (EM.LT.EP) GOTO 730
        END DO
        PRINT *,'SE HA EXCEDIDO EL LIMITE DE ITERACIONES '
C-------------------------------------
730     PRINT *
        PRINT *,'  SOLUCION FINAL (EN ORDEN CRECIENTE DE I PARA CADA J)'
        CALL PRNT(CFINAL,F,NI,NJ,20)
        END
C*****************************************
        SUBROUTINE PRNT(CAPT,F,NI,NJ,IDIM)
        DIMENSION F(IDIM,1)
        CHARACTER *8, CAPT
        PRINT 5,CAPT
```

```
5       FORMAT(2X,A8)
        DO J=NJ, 1 , -1
            PRINT 15, J
            PRINT 10, (F(I,J),I=1,NI)
        END DO
10      FORMAT(5X, 1P7E10.3)
15      FORMAT(' J=',I2)
        RETURN
        END
```

PROGRAMA 11-2 Método iterativo extrapolado de Jacobi (EJ)

A) Explicaciones

El programa es casi igual al PROGRAMA 11-1, excepto que se utiliza EJ en vez del SOR.

B) Variables

Véase el PROGRAMA 11-1.

C) Listado

```
C   La primera parte de este programa es idéntica a la del Programa 11-1
C   excepto por los comentarios y enunciados de impresión.
C-----------------------------------------------
C   Comienza EJ
C
        DO K=1,ITMAX            !  Comienza el ciclo de iteración
          EM=0
          DO J=1, NJ                    !     Comienza el recorrido punto por punto
            I1=1
            IF(K/2*2.EQ.K) I1=2
            DO I=I1, NI, 2
              QQ=S(I,J)+ AR(I-1,J)*F(I-1,J)+AR(I,J)*F(I+1,J)
              QQ=QQ+AT(I,J)*F(I,J+1)  +AT(I,J-1)*F(I,J-1)
              BB=F(I,J)
              F(I,J)=W*QQ/AC(I,J)+WB*F(I,J)
              ER=ABS(BB-F(I,J))
              IF (ER.GT.EM) EM=ER
            END DO
          END DO          !  Fin del recorrido
          IF(K/2*2.EQ.K) THEN
         PRINT 90,K,EM
90       FORMAT ( ' ITER. NO.=',I4,'       ER=',1PE10.3)
         IF (EM.LT.EP) GOTO 730
          END IF
        END DO
        PRINT *,' SE HA EXCEDIDO EL LIMITE DE ITERACIONES '
C-----------------------------------------
730     PRINT *
        PRINT *,'  SOLUCION FINAL (EN ORDEN CRECIENTE DE I PARA CADA J) '
        CALL PRNT(CFINAL,F,NI,NJ,20)
        END
C*****************************************
C     Favor de copiar la subrutina PRINT del Programa 11-1
```

PROGRAMA 11-3 Optimización del parámetro del SOR

A) Explicaciones

Muestra el algoritmo para determinar el parámetro óptimo del SOR. Un problema sencillo es la ecuación de Poisson

$$-\nabla^2\phi(x, y) = 1$$

la cual se resuelve con 21 × 21 puntos de la retícula, que presentan espaciamiento uniforme. Suponemos que todas las condiciones en la frontera son $\phi = 0$.

Se pueden cambiar los valores de NI y NJ para estudiar el efecto de un número creciente de puntos en la retícula.

B) Variables

NI, NJ: número de puntos de la retícula en las direcciones x y y, respectivamente
W: estimación inicial (por debajo del valor) del parámetro del SOR
FM: entero que se utiliza con fines de formateo
F(i, j): $\phi_{i,j}$
FS: lado izquierdo de la ecuación (11.6.9)
MS: estimación del radio espectral del SOR
MJ: estimación del radio espectral de Jacobi
WO: ω óptimo estimado

C) Listado

```
C       CSL/F11-3.FOR   DEMOSTRACION DEL OPTIMO DE OMEGA PARA SOR
        DIMENSION F(0:40,0:40)
        REAL*8   MJ,MS
        PRINT*,'CSL/F11-3   DEMOSTRACION DEL OPTIMO DE OMEGA PARA SOR'
        PRINT *
        NI=20
        NJ=20
        S=1
        W=1.3                  ! Se elige 1.3 como una estimación baja de omega
        WB=1-W
        K=0
        BS=1
        PRINT *, ' LA RETICULA PARA ESTA CORRIDA ES ', NI+1, 'x' , NJ+1
        PRINT *, ' THETA UTILIZADA = ', W
        PRINT *
        DO  I=0 , NI+1
          DO  J=0 , NJ+1
            F(I,J)=0           ! Las iteraciones se inicializan en cero
          END DO
        END DO
        PRINT*,'            RADIOS ESPECTRALES '
```

```
      PRINT*,' ITR.NO.  JACOBI        SOR OMEGA OPTIMO
      K=0                    ! Inicialización del contador de iteraciones
40    K=K+1
      FS=0
      DO  J= 1, NJ
        DO  I= I1, NI
          BB=F(I,J-1)
          IF (J.EQ.NJ) BB=BB*2              ! Condición en la frontera para j = NJ
          BL=F(I-1,J)
          IF (I.EQ.NI) BL=BL*2              ! Condición en la frontera para i = NI
          FB=F(I,J)
          F(I,J)=W*(S+(BL+F(I+1,J)+BB+F(I,J+1))/4)
     &          + WB*F(I,J)                 ! EJ: ecuación (11.4.6)
          FS=FS+(F(I,J)-FB)**2
        END DO
      END DO
      SB=MS
      MS=SQRT(FS/BS)
      BS=FS
      IF (MS.GT.1.0) GOTO 40
      MJ=(MS+W-1)/(W*SQRT(MS))                ! Ecuación (11.6.14)
      IF (MJ.LT.1) WO=2/(1+SQRT(1-MJ*MJ))     ! Ecuación (11.6.15)
      IF (K.EQ.1) GOTO 40
      PRINT 10, K,MJ,MS,WO
      IF (ABS(SB-MS) .LT. 0.0001)  STOP       ! Criterio de convergencia
10    FORMAT( 2x, I5, 3f10.6)
      GOTO 40
      END
```

D) Ejemplo de salida

```
CSL/F11-3   DEMOSTRACION DEL OPTIMO DE OMEGA PARA SOR

 LA RETICULA PARA ESTA CORRIDA ES 21 x 21
 THETA UTILIZADA =      1.300000

            RADIOS ESPECTRALES
 ITR.NO.   JACOBI        SOR      OMEGA OPTIMO
     2   0.988625   0.957808   1.738527
     3   0.992152   0.970879   1.777720
     4   0.993509   0.975911   1.795733
     5   0.994274   0.978747   1.806910
     6   0.994778   0.980616   1.814776
     7   0.995142   0.981967   1.820746
     8   0.995419   0.982995   1.825472
     9   0.995638   0.983807   1.829324
    10   0.995815   0.984465   1.832528
    11   0.995962   0.985008   1.835232
    12   0.996084   0.985463   1.837544
    13   0.996188   0.985848   1.839535
    14   0.996277   0.986178   1.841264
    15   0.996354   0.986463   1.842778
    16   0.996421   0.986711   1.844111
    17   0.996479   0.986928   1.845284
    18   0.996530   0.987119   1.846330
    19   0.996576   0.987287   1.847259
    20   0.996616   0.987437   1.848094
    21   0.996652   0.987570   1.848840
    22   0.996684   0.987690   1.849515
    23   0.996714   0.987799   1.850132
    24   0.996740   0.987895   1.850679
```

PROGRAMA 11-4 Demostración del theta óptimo

A) Explicaciones

El programa es muy similar al anterior, excepto por la optimización del parámetro de iteración para el método iterativo extrapolado de Jacobi.

B) Variables

Idénticas a las del PROGRAMA 11-3, más

θ: parámetro de iteración

C) Listado

```
C       CSL/F11-4     DEMOSTRACION DEL THETA OPTIMO
C
        DIMENSION F(0:40,0:40)
        REAL*8  MJ,MS
        PRINT*,'CSL/F11-4   DEMOSTRACION DEL THETA OPTIMO
        PRINT *
        NI=20
        NJ=20
        S=1
        W=1.3                 ! Se elige 1.3 como una estimación baja de omega
        WB=1-W
        K=0
        BS=1
        PRINT *, ' LA RETICULA PARA ESTA CORRIDA ES ', NI+1, ' x' , NJ+1
        PRINT *, ' THETA UTILIZADA  = ', W
        PRINT *
        DO  I=0 , NI+1
        DO  J=0 , NJ+1
           F(I,J)=0           ! Las iteraciones se inicializan en cero
        END DO
        END DO
        PRINT*,'            RADIOS ESPECTRALES '
        PRINT*,' ITR.NO.   JACOBI          EJ        THETA OPTIMO '
        K=0                   ! Inicialización del contador de iteraciones
40      K=K+1
        FS=0
        DO  J= 1, NJ
        I1=1
          IF ((K+J)/2*2.EQ.K+J) I1=2                ! I1 = 2 si K + J es par
          DO  I= I1, NI, 2
            BB=F(I,J-1)
            IF (J.EQ.NJ) BB=BB*2                    ! Condición en la frontera para J = NJ
            BL=F(I-1,J)
            IF (I.EQ.NI) BL=BL*2                    ! Condición en la frontera para i = NI
            FB=F(I,J)
            F(I,J)=W*(S+(BL+F(I+1,J)+BB+F(I,J+1))/4)
     &            + WB*F(I,J)                       ! EJ: ecuación (11.4.6)
            FS=FS+(F(I,J)-FB)**2
          END DO
        END DO
```

```
      IF (K/2*2 .NE. K) GOTO 40                  ! Se omite lo siguiente si k es impar
      SB=MS
      MS=SQRT(FS/BS)
      BS=FS
      IF (MS.GT.1.0) GOTO 40
      MJ=(MS+W-1)/(W*SQRT(MS))                   !  Ecuación (11.6.14)
      IF (MJ.LT.1) WO=2/(1+SQRT(1-MJ*MJ))        !  Ecuación (11.6.15)
      IF (K.EQ.1) GOTO 40
      PRINT 10, K,MJ,MS,WO
      IF (ABS(SB-MS) .LT. 0.0001)  STOP          ! Criterio de convergencia
10    FORMAT( 2x, I5, 3f10.6)
      GOTO 40
      END
```

D) Ejemplo de salida

```
CSL/F11-4   DEMOSTRACION DEL THETA OPTIMO

LA RETICULA PARA ESTA CORRIDA ES  21 x 21
THETA UTILIZADA  =      1.300000

              RADIOS ESPECTRALES
ITR.NO.   JACOBI        EJ      THETA OPTIMO
    6   0.995352   0.982748   1.824319
    8   0.993105   0.974410   1.790138
   10   0.993751   0.976806   1.799172
   12   0.994359   0.979062   1.808206
   14   0.994799   0.980696   1.815121
   16   0.995129   0.981920   1.820535
   18   0.995386   0.982873   1.824903
   20   0.995593   0.983642   1.828530
   22   0.995764   0.984275   1.831595
   24   0.995908   0.984808   1.834231
   26   0.996030   0.985263   1.836523
   28   0.996136   0.985655   1.838532
   30   0.996228   0.985997   1.840310
   32   0.996309   0.986296   1.841885
   34   0.996380   0.986559   1.843291
   36   0.996442   0.986792   1.844547
   38   0.996498   0.986998   1.845669
   40   0.996547   0.987181   1.846671
   42   0.996591   0.987343   1.847572
   44   0.996630   0.987487   1.848373
   46   0.996665   0.987616   1.849099
   48   0.996696   0.987732   1.849751
   50   0.996723   0.987834   1.850330
   52   0.996748   0.987926   1.850851
```

PROBLEMAS

11.1) La ecuación de Laplace

$$\nabla^2\phi(x, y) = 0$$

está dada para la geometría que se muestra en la figura P11.1. Utilice las condiciones impuestas en la frontera, determine las ecuaciones en diferencias y resuélvalas mediante SOR con $\omega = 1.3$.

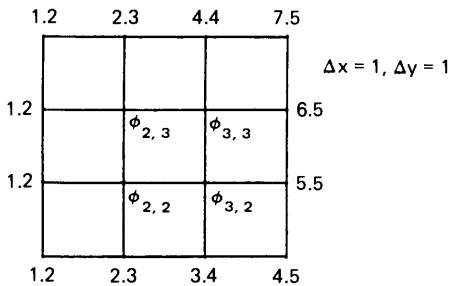


Figura P11.1

11.2) Considere la ecuación de Laplace para una geometría plana:

$$\nabla^2\phi(x, y) = 0$$

a) Escriba las ecuaciones en diferencias para la geometría que aparece en la figura P11.2. La condición en la frontera derecha es $\partial\phi/\partial x = 0$. Las condiciones en las demás fronteras son del tipo fijo y sus valores se muestran en la figura P11.2.

b) Resuelva las ecuaciones en diferencias mediante SOR, con $\omega = 1.3$.

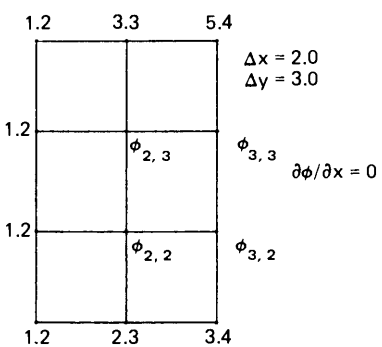


Figura P11.2

11.3) Reescriba la ecuación (L) del ejemplo 11.2 en forma matricial y muestre que tiene las propiedades que se explican en la nota del ejemplo 11.1. (*Sugerencia*: multiplique la primera ecuación por 1/4 y la segunda y tercera por 1/2.)

11.4) En el dominio bidimensional que se muestra en la figura P11.4, la ecuación de conducción del calor es

$$-k\nabla^2\phi = 0$$

Las temperaturas de las fronteras superior, derecha e inferior se muestran en la misma figura. La frontera izquierda está sujeta a la transferencia de calor por convección y la condición en la frontera está dada por

$$k\frac{\partial T}{\partial x} = h_c(T(x, y) - T_\infty)$$

donde k es la conductividad térmica del medio (25 W/mK), h_c es el coeficiente de transferencia de calor (250 W/m^{2}K) y T_∞ es la temperatura del fluido hacia la frontera izquierda (10°C).

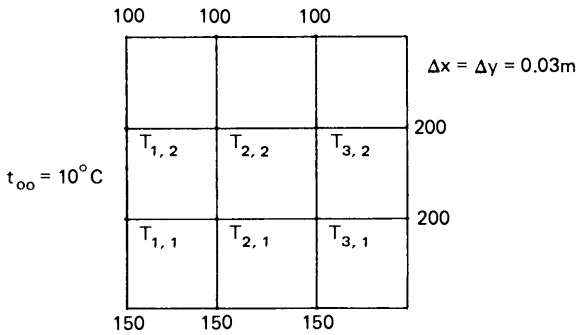


Figura P11.4

a) Obtenga las ecuaciones en diferencias para las seis temperaturas no conocidas.

b) Resuelva las ecuaciones en diferencias mediante SOR.

11.5) La siguiente es una ecuación diferencial parcial elíptica;

$$-\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\phi + 0.1\phi = 0$$

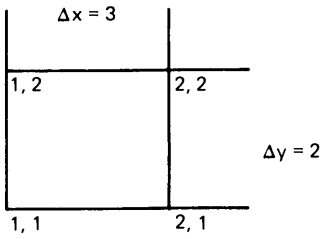


Figura P11.5

con las condiciones en la frontera

$$\frac{\partial \phi}{\partial n} = -\phi \qquad \text{(izquierda)}$$

$$\frac{\partial \phi}{\partial n} = -2\phi + 0.5 \qquad \text{(inferior)}$$

Obtenga la ecuación en diferencias para el punto de la esquina que se muestra en la figura P11.5.

11.6) Determine la ecuación en diferencias para la ecuación de Laplace en la geometría que aparece en la figura P11.6. Resuelva las ecuaciones en diferencias mediante SOR, con $\omega = 1.3$.

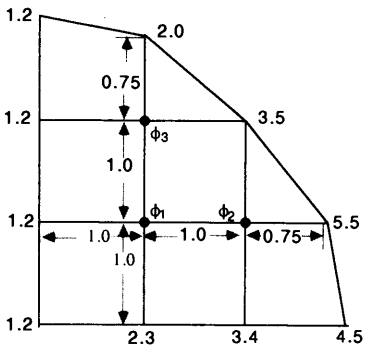


Figura P11.6

11.7) Mediante el PROGRAMA 11-1, determine la solución de la ecuación de Laplace $\nabla^2\phi(x, y) = 0$ para la geometría de la figura P11.7. Utilice 12 intervalos en la retícula en la dirección de x y 18 intervalos de la retícula en la dirección de y.

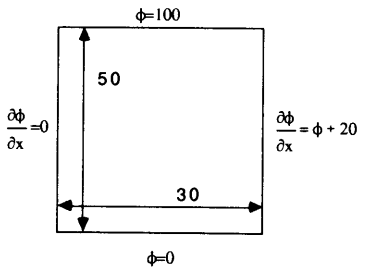


Figura P11.7

11.8) Una hoja rectangular de película de carbón tiene el cátodo y el ánodo como se muestra en la figura P11.8. El voltaje del cátodo está fijo en 5 v y el del ánodo es 0 v. Sin embargo, debido a que la película no es uniforme, la distribución del voltaje entre el cátodo y el ánodo no es lineal. La distribución del potencial eléctrico (voltaje) es la solución de

$$\nabla k(x, y)\nabla E(x, y) = 0$$

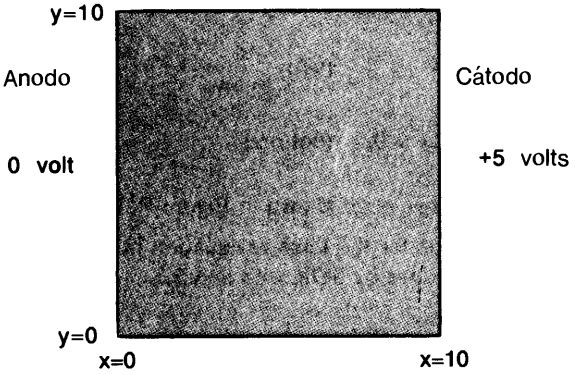


Figura P11.8

donde $k(x, y)$ es la conductividad de la película, en ohms $^{-1}$; E es el potencial eléctrico y x y y están dados en cm. Las condiciones en la frontera son

$$\frac{\partial E}{\partial y} = 0 \quad \text{en los lados superior e inferior}$$

$$E = 0 \text{ v} \quad \text{a lo largo del lado izquierdo}$$

$$E = 5 \text{ v} \quad \text{a lo largo del lado derecho}$$

La conductividad eléctrica k está dada por

$$k(x, y) = 1 + \frac{1}{20}[(x - 3)^2 + (y - 3)^2] \text{ ohm}^{-1}$$

Obtenga las ecuaciones en diferencias con $\Delta x = \Delta y = 1$ cm y resuélvalas mediante SOR con $\omega = 1.7$. ¿Cuál es el valor de $E(5, 5)$?

11.9) Considere la ecuación de Laplace en las coordenadas r-z:

$$\nabla^2 \phi(r, z) = 0$$

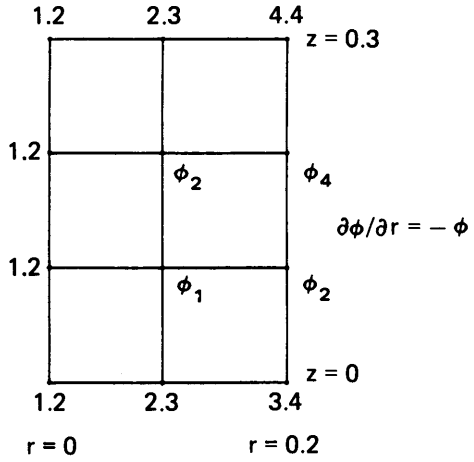


Figura P11.9

a) Escriba las ecuaciones en diferencias para esta ecuación y la geometría dada en la figura P11.9. La condición en la frontera derecha está dada por

$$\frac{\partial \phi}{\partial r} = -\phi$$

En la figura se muestran los valores de ϕ en las demás fronteras.

b) Resuelva las ecuaciones en diferencias mediante el método de Gauss-Seidel (o, en forma equivalente, mediante SOR con $w = 1$).

11.10) Reescriba las ecuaciones en diferencias del ejemplo 11.1, con la hipótesis de que las coordenadas son r-z. Suponga que el lado izquierdo está en el centro del cilindro. Muestre también que las propiedades de las ecuaciones en diferencias descritas en las notas del ejemplo 11.1 también se aplican a estas coordenadas.

11.11) Sustituya el método iterativo del PROGRAMA 11-1 por el método iterativo extrapolado de Jacobi y determine el número de pasos de iteración necesarios con $\theta = 1.5$ para que se satisfaga el mismo criterio de convergencia.

11.12) Escriba un programa que resuelva

$$\phi_i = \frac{1}{2}(\phi_{i-1} + \phi_{i+1}) + 1, \quad i = 1, 2, \ldots, N$$

$$\phi_0 = \phi_{N+1} = 0$$

mediante SOR. Haga la estimación inicial de $\phi = 0$ para todos los puntos de la retícula. Detenga la iteración cuando $|\phi_i^{(t)} - \phi_i^{(t-1)}| < 0.001$ para todos los puntos de la retícula. Cuando termine el programa, ejecútelo para $N = 30$ con distintos valores de ω en $1 < \omega < 2$ y determine el total de pasos de iteración para que se satisfaga este criterio. Grafique el número de pasos de iteración contra ω.

11.13) Una ecuación en diferencias unidimensional está dada por

$$-\phi_{i-1} + 2\phi_i - \phi_{i+1} = 1$$

con las condiciones en la frontera $\phi_0 = \phi_{100} = 0$. Si se utiliza SOR para resolver esta ecuación con $\omega = 1.3$ (que es menor que $\omega_{ópt}$), ¿cuál será la distribución espacial del error después de un número suficientemente grande de pasos de iteración? ¿Con qué tasa decae este error en un ciclo de iteración?

11.14) Considere las ecuaciones en diferencias dadas por

$$-\phi_{i-1} + 2\phi_i - \phi_{i+1} = 1$$

$$i = 1, 2, \ldots, 20$$

con condiciones en la frontera $\phi_0 = \phi_{21} = 0$.

a) Muestre que el $\omega_{ópt}$ para el SOR aplicado a esta ecuación es 1.7406.

b) Grafique todos los valores propios del SOR en el plano complejo para los siguientes casos: $\omega = 1.7$, $\omega = \omega_{ópt}$ y $\omega = 1.8$.

c) Calcule el radio espectral para cada ω de b).

11.15) Repita el problema anterior con el método iterativo extrapolado de Jacobi.

11.16) En un reactor, la distribución de potencia se controla mediante unas varillas en forma de cruz, las cuales absorben los neutrones. La distribución del flujo de neutrones en una vecindad de la varilla de control se puede calcular como la solución de

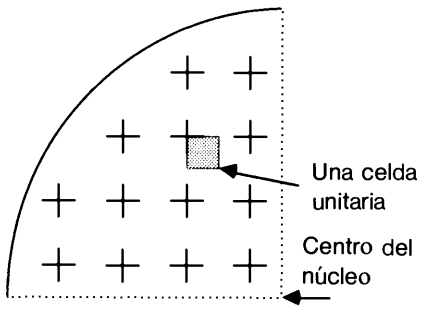
$$-D\nabla^2\psi(x, y) + \Sigma_a\psi(x, y) = S$$

en la geometría que se muestra en la figura P11.16. En esta ecuación, D, Σ_a y S son el flujo de neutrones, la absorción de la sección transversal y la fuente de neutrones, respectivamente. Si $D = 0.2\ \text{cm}^2$, $\Sigma_a = 0.1\ \text{cm}^{-1}$ y $S = 1$ neutrón seg $^{-1}$ cm $^{-3}$, resuelva la ecuación en diferencias para la ecuación anterior con las condiciones en la frontera.

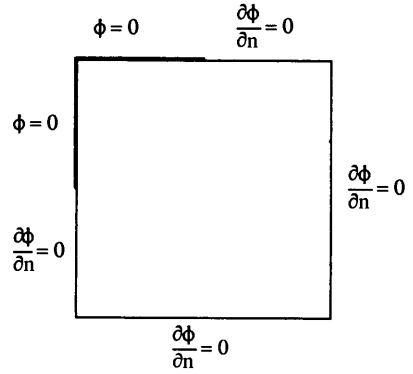
$$\frac{\partial \psi}{\partial n} = 0 \quad \text{a lo largo de todas las fronteras excepto sobre la varilla de control}$$

$$\psi = 0 \quad \text{a lo largo de la superficie de la varilla de control}$$

Suponga que el espesor de la varilla de control es igual a cero.



Arreglo de las varillas de control en un reactor nuclear (se muestra un cuadrante del núcleo)



Una celda unitaria en torno a una varilla de control

Figura P11.16

11.17) Repita el problema anterior utilizando la siguiente condición en la frontera a lo largo de la varilla de control

$$\frac{\partial \psi}{\partial n} = -\psi$$

11.18) La ecuación (11.5.16) se transforma en un método de Gauss-Geidel si $\omega = 1$. Muestre que la mitad de los valores propios de este método se anulan, mientras que los demás son reales y positivos.

11.19) Incorpore al PROGRAMA 11-1 el algoritmo automático para determinar el óptimo de ω.

11.20) Incorpore al PROGRAMA 11-2 el algoritmo automático para determinar el óptimo de ω y ejecute el programa para

$$\left(\frac{\delta^2}{\delta x^2} + \frac{\delta^2}{\delta y^2}\right)\phi_{i,j} = 0$$

con las condiciones en la frontera

$$\phi_{0,j} = \phi_{i,0} = 1$$
$$\phi_{30,j} = \phi_{i,30} = 0$$

a) ¿Cuál es la solución para $\phi_{15,15}$?

b) ¿Cuál es el valor de $\theta_{ópt}$?

11.21) El problema de valores propios asociado con el método iterativo extrapolado de Jacobi se escribe como

$$4\eta u_{i,j} = u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} \tag{A}$$

donde η es un valor propio. Si las condiciones en la frontera son

$$u_{0,j} = u_{I+1,j} = u_{i,0} = u_{i,J+1} = 0$$

entonces las funciones propias son

$$u_{i,j}^{(m,n)} = \text{sen}\,(miA)\ \text{sen}\,(njB)$$

donde m y n son enteros tales que $0 < m < I + 1$, y $0 < n < J + 1$; $A = \pi/(I + 1)$ y $B = \pi/(J + 1)$.

a) Muestre que los valores propios son

$$\eta^{(m,n)} = \frac{1}{2}(\cos(mA) + \cos(nB))$$

b) Muestre que el valor propio máximo aparece cuando $m = n = 1$ y que el mínimo aparece cuando $m = I$ y $n = J$.

c) Muestre que el valor absoluto del mínimo η es igual al máximo η.

11.22) Muestre que podemos aproximar el valor $\eta^{(1,1)}$ del problema anterior por

$$\eta^{(1,1)} = 1 - \frac{1}{4}(A^2 + B^2)$$

11.23) Consideremos la ecuación en diferencias

$$\frac{-\phi_{i-1,j} + 2\phi_{i,j} - \phi_{i+1,j}}{h_x^2} + \frac{-\phi_{i,j-1} + 2\phi_{i,j} - \phi_{i,j+1}}{h_y^2} = S$$

con condiciones en la frontera

$$\phi_{0,j} = \phi_{I+1,j} = 0$$
$$\phi_{i,0} = \phi_{i,J+1} = 0$$

a) Escriba el problema de valores propios asociado con el método iterativo de Jacobi.

b) Escriba las funciones propias y sus correspondientes valores propios.

c) Muestre que el radio espectral del método iterativo de Jacobi es

$$\mu = \frac{1}{2}[\cos(\alpha) + \cos(\beta)]$$

donde $\alpha = \pi/(I + 1)$ y $\beta = \pi/(J + 1)$

11.24) El problema de valores propios asociado al SOR es

$$\xi[a^C v_{i,j} + \omega(a^L v_{i-1,j} + a^B v_{i,j-1})] = -[a^R v_{i+1,j} + a^T v_{i,j+1}] + (1-\omega)a^C v_{i,j} \qquad \text{(A)}$$

La función propia del SOR se relaciona con la función propia asociada de Jacobi de la forma siguiente:

$$v_{i,j} = \xi^{(i+j)/2} u_{i,j} \qquad \text{(B)}$$

donde $u_{i,j}$ es la función propia de Jacobi. Al sustituir la ecuación (B) en la (A) obtenemos

$$\begin{aligned}\xi a^C u_{i,j} = &-\omega\xi^{1/2}[a^L u_{i-1,j} + a^R u_{i+1,j} \\ &+ a^B u_{i,j-1} + a^T u_{i,j+1}] + (1-\omega)a^C u_{i,j}\end{aligned}$$

Muestre que la relación entre el valor propio de Jacobi y el del SOR es

$$\xi - \omega\xi^{1/2}\eta + (\omega - 1) = 0$$

donde η es el valor propio de Jacobi.

11.25) Demuestre que el radio espectral del SOR está dado por la ecuación (11.6.6) si $\omega < \omega_{ópt}$ y por la ecuación (11.6.7) si $\omega > \omega_{ópt}$.

11.26) Después de un cierto número de ciclos de iteración del SOR con $\omega = 1.3$, se determinaron las siguientes estadísticas de los valores iterados:

$$N^{(t)} = 50.32$$
$$N^{(t+1)} = 49.31$$
$$N^{(t+2)} = 48.33$$

donde

$$N^{(t)} = \sum_{i,j} [\phi_{i,j}^{(t)} - \phi_{i,j}^{(t-1)}]^2$$

Estime:

a) El radio espectral del SOR con $\omega = 1.3$, μ_ω.

b) El radio espectral de Jacobi, μ_J.

c) El óptimo de ω, $\omega_{ópt}$.

BIBLIOGRAFIA

Becker, E. B., G. F. Carey y J. T. Oden, *Finite Elements; an Introduction*, Prentice-Hall, 1981.

Boisvert, R. F. y R. A. Sweet, "*Mathematical Software for Elliptic Boundary Value Problems*", en *Sources and Development of Mathematical Software* (Colwel, editor), Prentice-Hall, 1984.

Hageman, L. A. y D. M. Young, *Applied Iterative Methods*, Academic Press, 1981.

Kershaw, D., "Solution of single tridiagonal linear systems and vectorization of the ICCG algorithm on the Cray-1", en *Parallel Computations* (G. Rodrigue, editor), Academic Press, 1981.

Myint-U, U. y L. Debnath, *Partial Differential Equations for Scientists and Engineers*, North-Holland, 1987.

Nakamura, S., *Computational Methods in Engineering and Science with Application to Fluid Dynamics and Nuclear Systems*, Krieger, 1986.

Nogotov, E. F., *Applications of Numerical Heat Transfer*, Hemisphere, 1978.

Norrie, D. H. y G. Devies, *An Introduction to Finite Element Analysis*, Academic Press, 1978.

Nussbaumer, H. J., *Fast Fourier Transform and Convolution Algorithms*, 2a. edición, Springer-Verlag, 1982.

Segerlind, L. J., *Applied Finite Element Analysis*, Wiley, 1976.

Smith, G. D., *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 2a. edición, Oxford University Press, 1978.

Strang, G. W., *Linear Algebra and Its Application*, 2a. edición, Academic Press, 1980.

Swarztrauber, P. N., "The methods of cyclic reduction, Fourier analysis and FACR algorithm for discrete solution of Poisson's equation on a rectangle", *SIAM Rev.*, vol. 19, 490-501, 1977.

Sweet, R.A., "A cyclic reduction algorithm for solving block tridiagonal systems of arbitrary dimension", SIAM *J. Numer. Anal.*, vol. 14, 706-720, 1977.

Thompson, J. F., editores, *Numerical Grid Generation*, North-Holland, 1982.

Thompson, J. F., Z. U. A. Wasri y C. W. Mastin, *Numerical Grid Generation: Foundation and Applications*, North-Holland, 1985.

Varga, R.S., *Matrix iterative analysis*, Prentice-Hall, 1962.

Wachpress, E. L., *Iterative Solution of Elliptic Systems*, Prentice-Hall, 1966.

12
Ecuaciones diferenciales parciales parabólicas

12.1 INTRODUCCION

Las ecuaciones que rigen la difusión de partículas en movimiento o la conducción de calor, son ecuaciones diferenciales parciales (EDP) de tipo parabólico. Es por esto que los métodos de solución numérica de las EDP parabólicas son importantes en campos como difusión molecular, la transferencia de calor, el análisis de reactores nucleares y el flujo de fluidos. Puesto que las EDP parabólicas representan procesos de difusión que dependen del tiempo, usualmente utilizaremos las letras t y x como variables independientes, donde t es el tiempo y x es la coordenada del espacio unidimensional. Para las EDP parabólicas en espacios bidimensionales utilizaremos x y y para las coordenadas espaciales y t como el tiempo.

Los siguientes son ejemplos de EDP parabólicas:

a) Conducción transitoria de calor, con la dimensión espacial igual a uno [Incorpera/DeWitt]:

$$\rho c \frac{\partial T}{\partial t} = k \frac{\partial^2 T(x, t)}{\partial x^2} + Q(x) \tag{12.1.1}$$

b) Ecuación de difusión transitoria de neutrones, con la dimensión espacial igual a uno [Hetric]:

$$\frac{1}{v} \frac{\partial}{\partial t} \phi(x, t) = D \frac{\partial^2 \phi}{\partial x^2} - \Sigma_a \phi + \nu \Sigma_f \phi + S \tag{12.1.2}$$

donde ϕ es el flujo de neutrones.

c) Transporte convectivo de una sustancia química con difusión [Brodkey/ Hershey]:

$$\frac{\partial}{\partial t}\phi = -\frac{\partial}{\partial x}u(x)\phi + D\frac{\partial^2}{\partial x^2}\phi \tag{12.1.3}$$

donde ϕ es la densidad de la sustancia, $u(x)$ es la velocidad del flujo y D es la constante de difusión.

as EDP parabólicas para dos y tres dimensiones se pueden escribir mediante la ampliación de la variable espacial a dos y tres dimensiones del espacio. Por ejemplo, la ecuación de conducción transitoria de calor, en dimensiones espaciales iguales a dos, es

$$\rho c\frac{\partial \phi}{\partial t} = k\left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}\right) + Q(x, y) \tag{12.1.4}$$

En el resto de este capítulo, describiremos los métodos de diferencias finitas para las ecuaciones diferenciales parciales de tipo parabólico y con dimensiones del espacio iguales a dos y tres. En la tabla 12.1 se resumen las ventajas y desventajas de los métodos.

Tabla 12.1 Métodos de diferencias finitas para EDP parabólicas

Método	Ventajas	Desventajas
Dimensión espacial uno		
Euler hacia adelante (explícito)	Sencillez.	Δt debe ser menor que cierto límite de estabilidad.
Euler hacia atrás (implícito)	Incondicionalmente estable.	Necesita el esquema de solución tridiagonal.
Crank-Nicolson	Incondicionalmente estable; más preciso que el de Euler hacia atrás.	Igual que el anterior.
Dimensión espacial dos		
Euler hacia adelante (explícito)	Sencillez.	Para la estabilidad, Δt debe ser menor que cierto criterio.
Euler hacia atrás (implícito)	Incondicionalmente estable.	Necesita la solución simultánea en cada intervalo de tiempo.
Crank-Nicolson	Mejor precisión que el método anterior. Incondicionalmente estable.	La misma desventaja que el método implícito.
IDA	Incondicionalmente estable. Utiliza la solución tridiagonal.	Requiere de más esfuerzo de programación que los dos métodos anteriores.

12.2 ECUACIONES EN DIFERENCIAS

Como se explicó en la sección anterior, una EDP parabólica con dimensión espacial igual a uno es la contraparte transitoria de un problema con valores en la frontera de una EDO de segundo orden (véase el capítulo 10). También es un problema con condiciones iniciales. De hecho, si una EDP parabólica primero se hace discontinua con

respecto al espacio, se convierte en un problema con condiciones iniciales de ecuaciones diferenciales ordinarias simultáneas (véase el ejemplo 9.16). Por lo tanto, entre los métodos numéricos para una EDP parabólica están: 1) un problema con condiciones en la frontera, y 2) un problema con condiciones iniciales.

Por estas razones, se pueden desarrollar los métodos numéricos para una EDP parabólica, combinando un método numérico para los problemas con condiciones iniciales de EDO con otro método numérico para los problemas con valores en la frontera. Aunque en principio se puede utilizar la mayoría de los métodos analizados en el capítulo 9 para los problemas con condiciones iniciales, el uso de métodos de Runge-Kutta, o predictor-corrector de orden mayor, puede traer como resultado métodos muy complicados o, al menos, ineficientes. Esta limitación nos lleva a considerar el grupo más simple de métodos numéricos para los problemas con condiciones iniciales (es decir, los métodos de Euler).

12.2.1 Aproximaciones por diferencias en el dominio del tiempo

Como ejemplo de EDP parabólica, consideremos la ecuación:

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi(x, t)}{\partial x^2} + S(x, t), \quad 0 \leqslant x \leqslant H \tag{12.2.1}$$

donde α es una constante. Las condiciones iniciales y en la frontera son

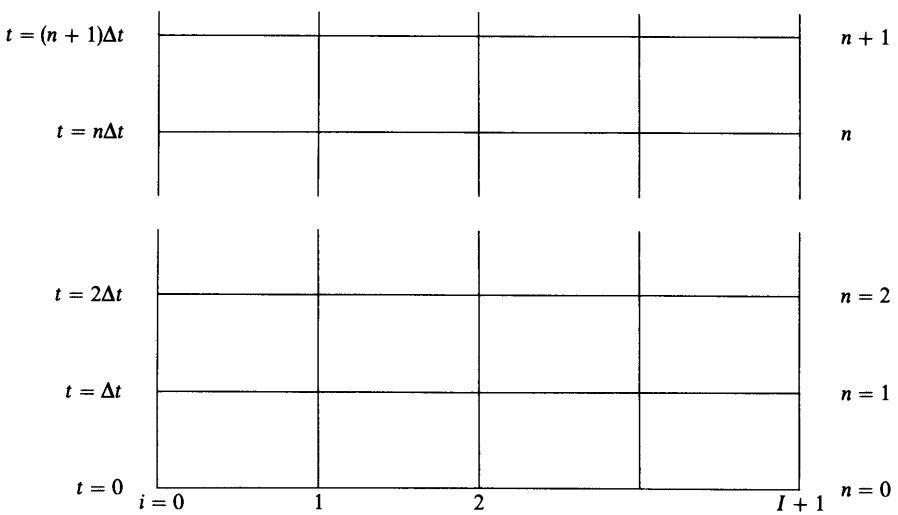
$$\phi(0, t) = \phi_L, \qquad \phi(H, t) = \phi_R \quad \text{(condiciones en la frontera)}$$

$$\phi(x, 0) = \phi_{\text{ini}}(x) \qquad \text{(condición inicial)}$$

Podemos dar también las condiciones en la frontera de forma general y de manera muy similar a la de las ecuaciones (10.2.8) y (10.2.9).

Para obtener aproximaciones por diferencias finitas, consideremos la retícula que se muestra en la figura 12.1. Los puntos de la retícula en el espacio se numeran

Figura 12.1 Sistema reticular para una ecuación diferencial parcial parabólica unidimensional



aquí por i, mientras que los puntos de la retícula sobre la coordenada del tiempo se numeran por n. La solución en x_i y t_n se denotará por

$$\phi_i^{(n)} \equiv \phi(x_i, t_n), \quad i = 0, 1, 2, \ldots, I + 1 \tag{12.2.2}$$

En primer lugar, aplicamos la aproximación por diferencias a la derivada con respecto del tiempo de (12.2.1). Si reescribimos la ecuación (12.2.1) para simplificar los cálculos como

$$\frac{\partial}{\partial t}\phi = DER(x, t) \tag{12.2.3}$$

donde $DER(x, t)$ representa el lado derecho de la ecuación (12.2.1), podemos escribir los métodos de Euler hacia adelante y hacia atrás en el dominio del tiempo de la manera siguiente:

$$\frac{\phi(x, t_{n+1}) - \phi(x, t_n)}{\Delta t} = DER(x, t_n) \text{ (Euler hacia adelante: explícito)} \tag{12.2.4}$$

$$\frac{\phi(x, t_{n+1}) - \phi(x, t_n)}{\Delta t} = DER(x, t_{n+1}) \text{ (Euler hacia atrás: implícito)} \tag{12.2.5}$$

respectivamente. Además, al usar el método modificado de Euler obtenemos

$$\frac{\phi(x, t_{n+1}) - \phi(x, t_n)}{\Delta t}$$

$$= \frac{1}{2}[DER(x, t_n) + DER(x, t_{n+1})] \text{ (Euler modificado: Crank-Nicolson)} \tag{12.2.6}$$

La ecuación (12.2.4) recibe el nombre de método explícito, mientras que las ecuaciones (12.2.5) y (12.2.6) son los métodos implícito y de Crank-Nicolson, respectivamente [Richtmyer/Morton].

12.2.2 Método de Euler hacia adelante

Podemos hacer discontinuo el término de la segunda derivada en la ecuación (12.2.1) mediante la aproximación por diferencias centrales. Entonces, el método de Euler hacia adelante dado por (12.2.4) es

$$\frac{\phi_i^{(n+1)} - \phi_i^{(n)}}{\Delta t} = \alpha \frac{\phi_{i-1}^{(n)} - 2\phi_i^{(n)} + \phi_{i+1}^{(n)}}{(\Delta x)^2} + S_i \tag{12.2.7}$$

con

$$\phi_0 = \phi_L, \quad \phi_{I+1} = \phi_R$$

Reescribimos (12.2.7) para obtener

$$\phi_i^{(n+1)} = \phi_i^{(n)} + \gamma(\phi_{i-1}^{(n)} - 2\phi_i^{(n)} + \phi_{i+1}^{(n)}) + \Delta t S_i \tag{12.2.8}$$

donde

$$\gamma = \frac{\alpha \Delta t}{\Delta x^2} \tag{12.2.9}$$

Este esquema es explícito ya que, si los valores $\phi_i^{(n)}$ se conocen para t_n en todos los puntos de la retícula, los valores $\phi_i^{(n+1)}$ para el nuevo tiempo t_{n+1} se calculan sin resolver ecuaciones simultáneas.

El tamaño del intervalo Δt del método explícito debe cumplir

$$\Delta t \leqslant \frac{0.5\Delta x^2}{\alpha}$$

o, en forma equivalente,

$$\gamma \leqslant 0.5 \tag{12.2.10}$$

(En la sección 12.3 analizaremos cómo se obtiene este criterio.) Por otra parte, puede ocurrir una inestabilidad en la solución. No importa qué tan lento sea el cambio físico del sistema, Δt debe ser menor o igual que $0.5\ \Delta x^2/\alpha$. La ecuación (12.2.10) también indica que Δt debe hacerse cada vez más pequeño si reducimos Δx.

El PROGRAMA 12-1 determina la solución de un problema de conducción de calor mediante el método explícito. Se pueden encontrar más programas del método explícito en [Ferziger, Incorpera y DeWitt], así como en [Rieder y Busby].

Ejemplo 12.1

Utilice el método explícito para resolver la ecuación de conducción de calor

$$\frac{\partial}{\partial t} T(x, t) = \alpha \frac{\partial^2}{\partial x^2} T(x, t), \quad 0 < x < 10, \quad t > 0 \tag{A}$$

donde las condiciones iniciales y en la frontera están dadas por

$$T(0, t) = 0, \quad T(10, t) = 100 \quad \text{(condiciones en la frontera)}$$
$$T(x, 0) = 0 \quad \text{(condiciones iniciales)}$$

y los símbolos son

x coordenada espacial, cm
α difusividad térmica, cm^2/seg
T temperatura, °C
t tiempo, seg

Suponga $\alpha = 10$ y utilice 10 intervalos en la retícula para la coordenada x. Utilice dos valores de Δt: $\Delta t = 0.02$ y $\Delta t = 0.055$.

(Solución)

Puesto que la longitud del dominio es 10 cm y el número de intervalos en la retícula es 10, el espaciamiento es $\Delta x = 1$ cm. El número total de puntos en la retícula es 11, de los cuales dos son los puntos de la frontera con temperaturas fijas, de manera que sólo se requieren hacer los cálculos de temperatura para nueve puntos. La aproximación por diferencias finitas del método explícito para la ecuación (A) es

$$T_i^{(n+1)} = T_i^{(n)} + \gamma(T_{i-1}^{(n)} - 2T_i^{(n)} + T_{i+1}^{(n)}) \tag{B}$$

donde γ se define mediante la ecuación (12.2.9). El criterio de estabilidad que debe satisfacer Δt para que $\gamma \leqslant 0.5$ es $\Delta t \leqslant 0.5(\Delta x)^2/\alpha = 0.05$ seg.

Los resultados computacionales con $\Delta t = 0.02$ ($\gamma = 0.2$) aparecen en la tabla 12.2.

Tabla 12.2 Cálculo de la distribución de temperatura

n	$t(s)$	$i=0$	1	2	3	4	5	6	7	8	9	10
1	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	20.0	100.0
2	0.04	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.0	32.0	100.0
3	0.06	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	8.8	40.0	100.0
4	0.08	0.0	0.0	0.0	0.0	0.0	0.0	0.2	2.2	13.4	45.8	100.0
5	0.10	0.0	0.0	0.0	0.0	0.0	0.0	0.5	4.1	17.7	50.1	100.0
6	0.12	0.0	0.0	0.0	0.0	0.0	0.1	1.1	6.1	21.4	53.6	100.0
7	0.14	0.0	0.0	0.0	0.0	0.0	0.3	1.9	8.2	24.8	56.5	100.0
8	0.16	0.0	0.0	0.0	0.0	0.1	0.6	2.9	10.2	27.8	58.8	100.0
9	0.18	0.0	0.0	0.0	0.0	0.2	0.9	3.9	12.3	30.5	60.9	100.0
10	0.20	0.0	0.0	0.0	0.0	0.3	1.4	5.0	14.2	32.9	62.6	100.0
20	0.40	0.0	0.1	0.5	1.4	3.5	8.0	16.2	29.5	48.5	72.7	100.0
30	0.60	0.0	0.8	2.1	4.4	8.5	15.2	25.2	39.0	56.7	77.5	100.0
40	0.80	0.0	1.9	4.4	8.1	13.5	21.4	32.0	45.6	61.9	80.4	100.0
50	1.00	0.0	3.1	6.7	11.5	18.0	26.5	37.3	50.4	65.6	82.4	100.0
60	1.20	0.0	4.2	8.9	14.7	21.8	30.7	41.5	54.2	68.4	83.9	100.0
70	1.40	0.0	5.2	10.8	17.3	25.0	34.2	44.9	57.1	70.6	85.0	100.0
80	1.60	0.0	6.0	12.4	19.5	27.7	37.0	47.6	59.4	72.3	86.0	100.0
90	1.80	0.0	6.7	13.8	21.4	29.9	39.3	49.9	61.4	73.7	86.7	100.0
100	2.00	0.0	7.3	14.9	22.9	31.7	41.3	51.7	62.9	74.9	87.3	100.0

En la tabla 12.3 se muestran también los resultados, con $\Delta t = 0.055$ ($\gamma = 0.055$), que es un poco mayor que el límite de estabilidad. Las oscilaciones de magnitud creciente tanto en el espacio como en el tiempo son síntomas claros de inestabilidad.

Tabla 12.3 Cálculo de la distribución de temperatura

n	$t(s)$	$i=0$	1	2	3	4	5	6	7	8	9	10
1	0.05	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	55.0	100.0
2	0.11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	30.3	49.5	100.0
3	0.17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	16.6	24.2	66.7	100.0
4	0.22	0.0	0.0	0.0	0.0	0.0	0.0	9.2	11.6	43.4	61.6	100.0
5	0.28	0.0	0.0	0.0	0.0	0.0	5.0	5.5	27.7	36.0	72.7	100.0
6	0.33	0.0	0.0	0.0	0.0	2.8	2.5	17.5	20.0	51.7	67.5	100.0
7	0.38	0.0	0.0	0.0	1.5	1.1	10.9	10.7	36.0	43.0	76.7	100.0
8	0.44	0.0	0.0	0.8	0.5	6.7	5.4	24.7	25.9	57.7	71.0	100.0
9	0.50	0.0	0.5	0.2	4.1	2.5	16.8	14.7	42.7	47.5	79.6	100.0
10	0.55	0.0	0.0	2.5	1.1	11.2	7.8	31.2	30.0	62.5	73.2	100.0
15	0.82	0.0	4.2	0.1	15.9	4.4	34.7	20.0	59.5	53.0	86.4	100.0
20	1.10	0.0	−2.2	19.7	−3.7	41.8	4.9	64.6	30.7	84.3	74.3	100.0
25	1.38	0.0	18.5	−15.0	53.7	−19.1	82.1	−2.2	98.3	40.0	101.4	100.0
30	1.65	0.0	−21.6	66.4	−54.2	117.4	−56.9	140.2	−18.8	131.0	55.6	100.0
35	1.92	0.0	63.8	−93.5	172.3	−145.8	228.5	−128.5	216.6	−37.8	146.5	100.0
40	2.20	0.0	−105.6	232.0	−273.8	385.2	−328.1	407.7	−237.9	296.2	−28.1	100.0
45	2.47	0.0	234.0	−412.6	617.8	−662.0	778.9	−644.4	661.7	−356.6	316.5	100.0

12.2.3 Método implícito

Este método se basa en el de Euler hacia atrás con respecto al dominio del tiempo, como ya se mostró en la ecuación (12.2.5). Al hacer discontinua esta ecuación en la coordenada espacial se obtiene

$$\frac{\phi_i^{(n+1)} - \phi_i^{(n)}}{\Delta t} = \alpha \frac{\phi_{i-1}^{(n+1)} - 2\phi_i^{(n+1)} + \phi_{i+1}^{(n+1)}}{\Delta x^2} + S_i \tag{12.2.11}$$

Utilizamos la definición de γ dada por la ecuación (12.2.9) para reescribir (12.2.11) de la manera siguiente:

$$-\gamma\phi_{i-1}^{(n+1)} + (1 + 2\gamma)\phi_i^{(n+1)} - \gamma\phi_{i+1}^{(n+1)} = \phi_i^{(n)} + \Delta t S_i \tag{12.2.12}$$

La ecuación (12.2.12) no se puede resolver de manera individual para cada punto i de la retícula; todas las ecuaciones deben resolverse en forma simultánea. El conjunto de ecuaciones para $i = 1, 2, \ldots, N$ forman un sistema tridiagonal. El PROGRAMA 12-1 incluye el método implícito. Véase también [Ferziger] para otro programa.

El método implícito siempre es estable, independientemente del tamaño del intervalo de tiempo.

Ejemplo 12.2

Repita el problema del ejemplo 12.1, utilizando el método implícito, con los mismos puntos en la retícula, pero con $\gamma = 2$.

(Solución)

Las ecuaciones en diferencias son

$$-\gamma T_{i-1}^{(n+1)} + (2\gamma + 1)T_i^{(n+1)} - \gamma T_{i+1}^{(n+1)} = T_i^{(n)} \quad i = 1, 2, \ldots, I - 1$$

donde $I = 10$, $T_0 = 0$ y $T_I = 100$. La ecuación anterior se puede escribir en forma matricial como

$$\begin{bmatrix} 2\gamma + 1 & -\gamma & & & \\ -\gamma & 2\gamma + 1 & -\gamma & & \\ & -\gamma & 2\gamma + 1 & -\gamma & \\ & & & -\gamma & 2\gamma + 1 \end{bmatrix} \begin{bmatrix} T_1^{(n+1)} \\ T_2^{(n+1)} \\ T_3^{(n+1)} \\ T_{I-1}^{(n+1)} \end{bmatrix} = \begin{bmatrix} T_1^{(n)} \\ T_2^{(n)} \\ T_3^{(n)} \\ T_{I-1}^{(n)} + 100\gamma \end{bmatrix}$$

la cual se puede resolver con el algoritmo tridiagonal descrito en la sección 8.3. En la tabla 12.4 se muestran los resultados de los cálculos obtenidos mediante el PROGRAMA 12-1.

Tabla 12.4 Cálculo de la distribución de temperatura

n	$t(s)$	$i = 0$	1	2	3	4	5	6	7	8	9	10
0	.00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0
1	.2	0.0	0.2	0.4	0.8	1.6	3.1	6.3	12.5	25.0	50.0	100.0
2	.4	0.0	0.6	1.3	2.5	4.7	8.3	14.6	25.0	41.7	66.7	100.0
3	.6	0.0	1.2	2.8	5.0	8.6	14.1	22.4	34.7	51.8	74.1	100.0
4	.8	0.0	2.1	4.5	7.9	12.7	19.5	29.1	41.9	58.4	78.2	100.0
5	1	0.0	3.0	6.4	10.8	16.6	24.3	34.5	47.4	63.0	80.8	100.0
6	1.2	0.0	3.9	8.2	13.5	20.1	28.5	38.9	51.5	66.3	82.7	100.0
7	1.4	0.0	4.7	9.9	15.9	23.2	32.0	42.5	54.8	68.8	84.1	100.0
8	1.6	0.0	5.5	11.4	18.1	25.8	34.9	45.5	57.5	70.8	85.1	100.0
9	1.8	0.0	6.2	12.8	20.0	28.1	37.4	47.9	59.6	72.4	86.0	100.0
10	2.0	0.0	6.8	13.9	21.6	30.0	39.4	49.9	61.4	73.7	86.7	100.0

12.2.4 Método de Crank-Nicolson

El método numérico se basa en el método modificado de Euler, dado por la ecuación (12.2.6) se conoce como *método de Crank-Nicolson* y se escribe de la manera siguiente:

$$\frac{\phi_i^{(n+1)} - \phi_i^{(n)}}{\Delta t} = \frac{\alpha}{2(\Delta x)^2}\left[(\phi_{i-1}^{(n+1)} - 2\phi_i^{(n+1)} + \phi_{i+1}^{(n+1)}) + (\phi_{i-1}^{(n)} - 2\phi_i^{(n)} + \phi_{i+1}^{(n)})\right] + S_i \tag{12.2.13}$$

La ecuación (12.2.13) se puede escribir de la siguiente forma equivalente:

$$-\frac{\gamma}{2}\phi_{i-1}^{(n+1)} + (1+\gamma)\phi_i^{(n+1)} - \frac{\gamma}{2}\phi_{i+1}^{(n+1)} = \phi_i^{(n)} + \frac{\gamma}{2}(\phi_{i-1}^{(n)} - 2\phi_i^{(n)} + \phi_{i+1}^{(n)}) + \Delta t S_i \tag{12.2.14}$$

Este método requiere la solución tridiagonal en cada intervalo de tiempo.

El método de Crank-Nicolson es incondicionalmente estable (sin que sea relevante el valor de Δt). Aunque el tiempo de cómputo no es muy distinto del método implícito de Euler hacia atrás, su precisión es mayor, debido a que se basa en el método modificado de Euler, que tiene una precisión de segundo orden con respecto al tiempo, como se analizó en la sección 9.2. Se puede ejecutar el PROGRAMA 12-1 con el esquema de Crank-Nicolson.

Resumen de Esta Sección

a) El método de Euler hacia adelante es un método explícito. Es fácil de programar, pero la solución se vuelve inestable si Δt es mayor que el criterio de estabilidad.

Se recomienda este método si se desea una programación sencilla, o bien cuando se calcule una transición muy rápida y corta.

b) Los métodos de Euler hacia atrás y de Crank-Nicolson necesitan la solución tridiagonal en cada intervalo de tiempo, pero son incondicionalmente estables. Se recomiendan estos métodos si se calculan transiciones largas y lentas.

c) La precisión del método de Crank-Nicolson es mayor por una unidad que el de los otros dos métodos.

12.3 ANALISIS DE ESTABILIDAD

Como ya se analizó en la sección anterior, nuestra principal preocupación con los métodos numéricos de las EDP parabólicas es la estabilidad. En caso de ser inestables, los resultados numéricos se comportan en forma errática y divergen con un comportamiento oscilatorio, tanto en el tiempo como en el espacio. Por lo tanto, en esta sección estudiaremos los métodos básicos para el análisis de la estabilidad de un método y determinar el criterio de estabilidad.

Existen al menos cuatro métodos matemáticos para el análisis de la estabilidad de las ecuaciones en diferencias [Godunov/Ryabenkii; Smith]:

a) Método de la función propia
b) Método del desarrollo de Fourier
c) Método matricial
d) Método de la ecuación modificada

Estos métodos se pueden aplicar no sólo a las EDP de tipo parabólico sino también a las de tipo hiperbólico. En todos estos métodos se toman en cuenta las ecuaciones en diferencias para la parte homogénea. Esto se debe a que sin el término no homogéneo (o fuente), la EDP original no tiene una solución definida para tiempos arbitrariamente grandes. Por lo tanto, si la solución numérica de la parte homogénea de la EDP crece sin límite, esto se debe a la inestabilidad de la aproximación numérica utilizada.

En el método de la función propia, la solución numérica se desarrolla en funciones propias de la matriz que representa las ecuaciones en diferencias para cada intervalo de tiempo. Así, se determina el cambio en la amplitud de cada función propia al avanzar en los intervalos de tiempo. En este enfoque, se incorporan los efectos de las condiciones en la frontera. Este método es útil si se conocen las funciones propias para el conjunto finito dado de ecuaciones en diferencias.

En el método del desarrollo de Fourier, se estudia la estabilidad del método en un dominio infinito, desarrollando la solución en una serie de Fourier. Este enfoque es el más usado.

Si no es posible determinar o escribir con facilidad las funciones propias, aunque se tomen en cuenta incluso el efecto de las condiciones en la frontera u otros detalles de las ecuaciones en diferencias, el único camino a seguir es analizar los valores propios de la matriz que representa al conjunto de ecuaciones en diferencias. El método de la ecuación modificada se describe en la sección 13.4.

En el resto de esta sección analizaremos los dos primeros métodos.

12.3.1 Análisis de estabilidad con las funciones propias

Cuando analizamos una aproximación por diferencias de una EDP lineal parabólica sin término advectivo, podemos expresar su solución en términos de las funciones propias, las cuales son funciones senoidales para el caso de las geometrías planas unidimensionales. Cada función seno tiene su propio coeficiente que depende del tiempo. Mostraremos que si el método es inestable, los coeficientes dependientes del tiempo muestran comportamientos anormales en el tiempo, para los modos de frecuencias altas.

Con el fin de facilitar la exposición, consideremos la ecuación (12.2.1)* donde $S = 0$, o $\partial\phi/\partial t = \beta(\partial^2\phi/\partial x^2)$, con las siguientes condiciones en la frontera:

$$\phi(0, t) = \phi(H, t) = 0 \tag{12.3.1}$$

y las condiciones iniciales:

$$\phi(x, 0) = \phi_0(x) \tag{12.3.2}$$

Se puede mostrar fácilmente que una solución analítica de (12.2.1) que satisfaga las condiciones en la frontera se puede escribir de la manera siguiente:

$$\phi(x, t) = f_k(t)\,\mathrm{sen}\,(\eta_k x) \tag{12.3.3}$$

donde k es un entero positivo y

$$\eta_k = \frac{k\pi}{H} \tag{12.3.4}$$

$$f_k(t) = \exp\left[-\beta\eta_k^2 t\right] \tag{12.3.5}$$

Por lo tanto, la solución general se escribe como una suma de todas las soluciones posibles, es decir,

$$\phi(x, t) = \sum_{k=0}^{\infty} a_k \exp\left[-\beta\eta_k^2 t\right]\mathrm{sen}\,(\eta_k x) \tag{12.3.6}$$

donde a_k es un coeficiente determinado mediante la condición inicial. Una característica importante en esta solución analítica es que todos los términos tienden a cero al aumentar el tiempo, puesto que el signo de la función exponencial (12.3.5), es negativo.

Ahora analizamos la estabilidad del esquema de diferencias explícitas

$$\begin{aligned} \phi_i^{(n+1)} &= \phi_i^{(n)} + \gamma(\phi_{i-1}^{(n)} - 2\phi_i^{(n)} + \phi_{i+1}^{(n)}), \quad 0 < i < I + 1 \\ \gamma &= \frac{\beta\Delta t}{\Delta x^2} \end{aligned} \tag{12.3.7}$$

* En esta sección, denotaremos la difusividad térmica α de la ecuación (12.2.1) como β.

con las condiciones en la frontera

$$\phi_0^{(n)} = \phi_{I+1}^{(n)} = 0$$

y una condición inicial para $\phi_i^{(0)}$.

La solución de la ecuación (12.3.7) se puede determinar de manera analítica:

$$\phi_i^{(n)} = (\lambda_k)^n \operatorname{sen}(i\alpha_k) \tag{12.3.8}$$

donde α_k está dada por

$$\alpha_k = \frac{k\pi}{I+1}, \quad k = 1, 2, \ldots, I$$

y λ_k es una constante que recibe el nombre de *factor de amplitud*. Para determinar el valor de λ_k, sustituimos la ecuación (12.3.8) en (12.3.7). Los lados izquierdo y derecho de esta ecuación se transforman en

$$IZQ = (\lambda_k)^{n+1} \operatorname{sen}(i\alpha_k) \tag{12.3.9}$$

$$\begin{aligned} DER &= (\lambda_k)^n\{\operatorname{sen}(i\alpha_k) + \gamma[\operatorname{sen}(i\alpha_k - \alpha_k) - 2\operatorname{sen}(i\alpha_k) + \operatorname{sen}(i\alpha_k + \alpha_k)]\} \\ &= (\lambda_k)^n\{\operatorname{sen}(i\alpha_k) + 2\gamma[\cos(\alpha_k) - 1]\operatorname{sen}(i\alpha_k)\} \\ &= (\lambda_k)^n\{1 + 2\gamma[\cos(\alpha_k) - 1]\}\operatorname{sen}(i\alpha_k) \end{aligned} \tag{12.3.10}$$

Así, al igualar (12.3.9) con (12.3.10) obtenemos

$$\lambda_k = 1 + 2\gamma[\cos(\alpha_k) - 1] \tag{12.3.11}$$

Así, λ_k y $\phi_i = \operatorname{sen}(i\alpha_k)$ son el valor propio y la función propia, respectivamente, de

$$\lambda\phi_i = \phi_i + \gamma(\phi_{i-1} - 2\phi_i + \phi_{i+1})$$

Puesto que las funciones de (12.3.8), con $k = 1, 2, \ldots, I$ satisfacen la ecuación (12.3.7), la solución general es una combinación lineal de todas las soluciones posibles:

$$\phi_i^{(n)} = \sum_{k=1}^{I} a_k(\lambda_k)^n \operatorname{sen}(i\alpha_k) \tag{12.3.12}$$

en donde el coeficiente a_k queda determinado por la condición inicial.

La solución analítica del método explícito es muy similar a la de la ecuación (12.3.6). Sin embargo, una diferencia fundamental es que la ecuación (12.3.12) tiene un término $(\lambda_k)^n$ en vez de $\exp(-\beta\eta_k^2 t)$. Mostraremos a continuación que si k es pequeña, λ_k es una aproximación de $\exp(-\eta_k^2 t)$. Si $\alpha_k \ll 1$, λ_k se convierte en

$$\begin{aligned}\lambda_k &= 1 + 2\gamma[\cos(\alpha_k) - 1] \simeq 1 - \gamma\alpha_k^2 \\ &= 1 - \beta\frac{\Delta t}{\Delta x^2}\left(\frac{k\pi}{I+1}\right)^2 = 1 - \beta\Delta t\left(\frac{k\pi}{H}\right)^2 \\ &= 1 - \beta\Delta t\eta_k^2 \simeq \exp(-\beta\eta_k^2\Delta t)\end{aligned} \tag{12.3.13}$$

en donde utilizamos los desarrollos de Taylor de $\cos(\alpha_k)$ y $\exp(-\beta\eta_k^2\Delta t)$. Sin embargo, al crecer k, λ_k no se aproxima a $\exp(-\beta\eta_k^2\Delta t)$, sino que $(\lambda_k)^n$ se puede comportar en forma errática. Para garantizar la estabilidad, λ_k debe satisfacer

$$-1 \leqslant \lambda_k \leqslant 1 \tag{12.3.14}$$

para toda k. Sustituimos la ecuación (12.3.11) para escribir (12.3.14) de la manera siguiente:

$$-1 \leqslant 1 + 2\gamma[\cos(\alpha_k) - 1] \leqslant 1 \tag{12.3.15}$$

La segunda desigualdad siempre se cumple, ya que $\cos(\alpha_k) \leqslant 1$. Para ver lo que implica la segunda desigualdad, vemos primero que el mínimo de $\cos(\alpha_k)$ es igual a $\cos(\alpha_I) = \cos(\pi I/(I+1))$ que tiende a -1 cuando I tiende a infinito. Por lo tanto, la condición necesaria para que se satisfaga la primera desigualdad se escribe

$$\gamma \leqslant \frac{1}{2}$$

o, en forma equivalente,

$$\Delta t \leqslant \frac{(\Delta x)^2}{2\beta} \tag{12.3.16}$$

La estabilidad de los métodos implícito y de Crank-Nicolson se puede analizar de manera similar. El factor de amplitud para el método implícito es

$$\lambda_k = \frac{1}{1 + 2\gamma[1 - \cos(\alpha_k)]} \tag{12.3.17}$$

Puesto que $1 - \cos(\alpha_k) \geq 0$, el valor de λ_k dado por (12.3.17) siempre es positivo, por lo que siempre se cumple la primera desigualdad de la condición de estabilidad, $-1 \leqslant \lambda_k \leqslant 1$, La segunda desigualdad también se cumple siempre, puesto que el denominador de (12.3.17) es mayor que 1. Por lo tanto, el método implícito es estable independientemente del valor de γ o de Δt.

$$\lambda_k = \frac{1 - \gamma[1 - \cos(\alpha_k)]}{1 + \gamma[1 - \cos(\alpha_k)]} \tag{12.3.18}$$

Se muestra que la ecuación (12.3.18) satisface $|\lambda_k| \leqslant 1$ para toda α_k. Así, también el método de Crank-Nicolson es incondicionalmente estable.

12.3.2 Análisis de estabilidad de Fourier (Von Neumann)

Una desventaja del método examinado en la sección anterior es que funciona si se conocen las funciones propias, pero éste no es el caso para muchos problemas. El análisis de Fourier que presentamos aquí es más universal y se puede aplicar a cualquier tipo de ecuaciones en diferencias para problemas de espacio-tiempo [Mitchell/Griffiths; Richtmyer/Morton].

El análisis de estabilidad de Fourier examina la estabilidad de un método dado para resolver una EDP con las siguientes condiciones:

a) La EDP es lineal.
b) El dominio de interés es infinito.
c) El espaciamiento de la retícula es constante.
d) Los coeficientes de la EDP son constantes.

No se toman en cuenta los efectos de las condiciones reales en la frontera. A veces, un esquema numérico se vuelve estable bajo ciertas condiciones en la frontera, incluso cuando el análisis de estabilidad de Fourier lo declare como inestable. A pesar de esto, el análisis de estabilidad de Fourier se considera como un criterio importante que garantiza la estabilidad de un esquema.

El término fuente de la EDP no se toma en cuenta, debido a la siguiente razón. Si no existe ese término, la solución no debería crecer en el tiempo. Así, si la solución numérica crece, esto se debe a la inestabilidad del esquema.

El análisis de estabilidad de Fourier se puede aplicar a cualquier aproximación por diferencias de una EDP de tipo parabólico o hiperbólico, bajo las condiciones mencionadas antes. Consideremos el método explícito:

$$\phi_i^{(n+1)} = \phi_i^{(n)} + \gamma(\phi_{i-1}^{(n)} - 2\phi_i^{(n)} + \phi_{i+1}^{(n)}) \tag{12.3.19}$$

en el dominio infinito.

Supongamos que la condición inicial para este problema está dada como una función de Fourier:

$$\phi_i^0 = \exp(ij\pi/k)$$

o, en forma equivalente,

$$\phi_i^0 = \exp(ij\theta) \tag{12.3.20}$$

donde $j = \sqrt{-1}$, k es un entero distinto de cero y $\theta = \pi/k$, $-\pi \leqslant \theta \leqslant \pi$ k es la mitad de la longitud de onda en términos del número de intervalos de la retícula. En-

tonces, la solución del método numérico tiene la forma:

$$\phi_i^{(n)} = (G_\theta)^n \exp(ij\theta) \tag{12.3.21}$$

donde G_θ es el factor de amplitud (generalmente complejo), el cual se determina sustituyendo (12.3.21) en (12.3.19), de la manera siguiente:

$$\begin{aligned} G_\theta &= 1 + \gamma[\exp(-j\theta) - 2 + \exp(+j\theta)] \\ &= 1 + 2\gamma[\cos(\theta) - 1] \end{aligned} \tag{12.3.22}$$

Si $|G_\theta| \leqslant 1$ para $-\pi \leqslant \theta \leqslant \pi$, el método es estable, ya que $\phi_i^{(n)}$ no crece con el tiempo. Puesto que $-1 \leq \cos(\theta) \leq 1$, la condición $0 \leqslant |G_\theta| \leqslant 1$ requiere que

$$\gamma \leqslant 0.5 \tag{12.3.23}$$

Así, el método es estable si se satisface la ecuación (12.3.23). Este criterio es exactamente igual a la ecuación (12.3.16), obtenida mediante el uso de funciones propias.

Podría surgir la pregunta de porqué se hace el análisis de la subsección anterior que toma en cuenta el efecto de las condiciones en la frontera y da los mismos resultados que el análisis de estabilidad de Fourier para el dominio infinito. La respuesta es que la estabilidad de los esquemas de diferencias para las EDP parabólicas queda determinada por el modo de Fourier de la longitud de onda más pequeña ($k = \pm 1$, $\theta = \pm\pi$), que no depende de las condiciones en la frontera pero sí (únicamente) del espaciamiento en la retícula.

Un análisis similar muestra que los factores de amplitud para el método (implícito) de Euler hacia atrás es

$$G_\theta = \frac{1}{1 + 2\gamma(1 - \cos(\theta))} \tag{12.3.24}$$

El denominador es mayor o igual que 1, por lo que $G_\theta \leq 1$. Así, el método es incondicionalmente estable.

El factor de amplitud para el método de Crank-Nicolson es

$$G_\theta = \frac{1 - \gamma[1 - \cos(\theta)]}{1 + \gamma[1 - \cos(\theta)]} \tag{12.3.25}$$

que es idéntico a la ecuación (12.3.18).

Resumen de Esta Sección

a) En el análisis de estabilidad por medio de funciones propias, la solución numérica se desarrolla mediante la función propia del operador de diferencias. Si las funciones propias del método numérico están en forma analítica, el análisis toma en cuenta el efecto de las condiciones en la frontera.

b) El análisis con las funciones propias muestra que: 1) el método de Euler hacia adelante es estable sólo si se cumple $\gamma \leqslant 0.5$; ii) los métodos de Euler hacia atrás y modificado son incondicionalmente estables.

c) Aunque en el análisis de estabilidad por medio de funciones propias se toma en cuenta el efecto de las condiciones en la frontera, los resultados muestran que la estabilidad no tiene relación con dichas condiciones. El mismo criterio de estabilidad se obtiene mediante el análisis de estabilidad de Fourier (Von Neumann), el cual considera un dominio infinito y pasa por alto el efecto de las condiciones en la frontera.

d) En el análisis de estabilidad de Fourier, se considera un dominio infinito y la solución se desarrolla en series de Fourier. Este enfoque se basa en el hecho de que, ya sea el espacio discreto o continuo, cualquier función se puede desarrollar mediante la integral de Fourier. Sin embargo, debido al espacio discreto en la retícula, la integral se reduce a la suma de los componentes de Fourier de las frecuencias discretas. Un esquema numérico en estudio se considera estable si las magnitudes de los factores de amplitud de todas las longitudes de onda son menores o iguales que 1.

12.4 METODOS NUMERICOS PARA PROBLEMAS PARABOLICOS BIDIMENSIONALES

Los tres métodos analizados en la sección 12.2 para el caso de las EDP parabólicas con una dimensión espacial igual a uno se pueden extender al caso de las EDP parabólicas con dimensión espacial igual a dos, pero cada uno tiene las siguientes desventajas: el método de Euler hacia adelante es fácil de implantar en un programa, pero los intervalos de tiempo están limitados debido al criterio de estabilidad. El método implícito de Euler hacia atrás y el de Crank-Nicolson son incondicionalmente estables, pero ambos requieren de la solución en forma simultánea de las ecuaciones en diferencias para cada uno de los puntos de la retícula de todo el dominio para cualquier intervalo de tiempo. Las soluciones simultáneas se llevan a cabo ya sea por la solución directa o por un esquema iterativo, los cuales se tardan mucho tiempo debido a que se necesita la solución en cada intervalo de tiempo.

El método basado en una factorización aproximada —que se explica en el resto de esta sección— es incondicionalmente estable y la solución en cada intervalo sólo necesita la solución tridiagonal en cada línea de la retícula. Este esquema también se conoce como el método implícito de la dirección anternante (IDA). La factorización aproximada se aplica de manera amplia a otros tipos de EDP bajo el nombre de método de separación [Mitchell/Griffiths; Steger/Warming].

Consideremos

$$\frac{\partial \phi}{\partial t} = \alpha \nabla^2 \phi + S(x, y) \tag{12.4.1}$$

Hacemos discontinua la variable temporal con el método modificado de Euler para obtener

$$\frac{\phi(x, y, t_{n+1}) - \phi(x, y, t_n)}{\Delta t} = \frac{1}{2}[\alpha \nabla^2 \phi(x, y, t_{n+1}) + \alpha \nabla^2 \phi(x, y, t_n)] + S(x, y) \tag{12.4.2}$$

Si definimos

$$\delta\phi(x, y) = \phi(x, y, t_{n+1}) - \phi(x, y, t_n)$$

entonces podemos reescribir (12.4.2) como

$$\left[1 - \frac{\Delta t\alpha}{2}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right]\delta\phi = \Delta t\alpha\left[\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right]\phi(x, y, t_n) + \Delta t S \quad (12.4.3)$$

La ecuación (12.4.3) es una EDP elíptica que se puede aproximar por una forma factorizada:

$$\left[1 - \frac{\Delta t\alpha}{2}\frac{\partial^2}{\partial x^2}\right]\left[1 - \frac{\Delta t\alpha}{2}\frac{\partial^2}{\partial y^2}\right]\delta\phi = DER \quad (12.4.4)$$

donde *DER* es el lado derecho de (12.4.3). La ecuación (12.4.4) se puede resolver en dos etapas, como sigue:

$$\begin{gathered}\left[1 - \frac{\Delta t\alpha}{2}\frac{\partial^2}{\partial x^2}\right]\psi(x, y) = DER \\ \left[1 - \frac{\Delta t\alpha}{2}\frac{\partial^2}{\partial y^2}\right]\delta\phi = \psi(x, y)\end{gathered} \quad (12.4.5)$$

Ambas ecuaciones se hacen discontinuas ahora en el dominio *x-y*:

$$\begin{gathered}-\frac{1}{2}\gamma_x\psi_{i-1,j} + (1 + \gamma_x)\psi_{i,j} - \frac{1}{2}\gamma_x\psi_{i+1,j} = DER_{i,j} \\ -\frac{1}{2}\gamma_y\delta\phi_{i,j-1} + (1 + \gamma_y)\delta\phi_{i,j} - \frac{1}{2}\gamma_y\delta\phi_{i,j+1} = \psi_{i,j}\end{gathered} \quad (12.4.6)$$

donde $\gamma_x = \Delta t\alpha/\Delta x^2$ y $\gamma_y = \Delta t\alpha/\Delta y^2$. La primera ecuación comprende a un conjunto de ecuaciones tridiagonales para cada *j* (línea horizontal en la retícula), mientras que la segunda ecuación involucra a un conjunto de ecuaciones tridiagonales para cada *i* (línea vertical en la retícula). Así, determinamos la solución de ϕ en el nuevo punto en el tiempo, usando solamente el esquema tridiagonal sin una solución iterativa.

Resumen de Esta Sección

a) El método numérico presentado en esta sección se conoce como método implícito de la dirección alternante (IDA) o de la factorización aproximada. Sólo necesita la solución tridiagonal para cada línea horizontal y vertical de la retícula.
b) El método IDA es incondicionalmente estable.

PROGRAMAS

PROGRAMA 12-1 Ecuación de conducción del calor

A) Explicaciones

El PROGRAMA 12-1 resuelve la ecuación unidimensional de conducción del calor dada por (12.2.1) por medio del método explícito, implícito o de Crank-Nicolson, según lo desee el usuario. Las condiciones en la frontera e iniciales, así como las constantes, son las del ejemplo 12.2. El programa utiliza el método explícito si $\theta = 0$, el implícito si el parámetro θ se hace igual a 1 y el esquema de Crank-Nicolson si $\theta = 0.5$.

B) Variables

N: contador de los intervalos de tiempo
AL: α
DT: Δt
DX: Δx
GA: γ
TH: θ ($\theta = 0$ para el método explícito, $\theta = 1$ para el método implícito, $\theta = 0.5$ para el método de Crank-Nicolson)
TB(I): $T_i^{(n)}$
S(I): $s(x_i)$

C) Listado

```
C-----CSL/F12-1.FOR     ESQUEMA IMPLICITO/EXPLICITO PARA UNA EDP PARABOLICA
      DIMENSION A(100),B(100),C(100),D(100),T(0:100),TB(0:100),S(100)
      PRINT *,'CSL/F12-1  ESQUEMA IMPLICITO/EXPLICITO PARA UNA EDP PARABOLICA '
      PRINT *,'DAR EL VALOR DE THETA '
      READ *, TH
      IF (TH.EQ.0) PRINT *,'SE HA ELEGIDO EL METODO EXPLICITO.'
      IF (TH.GT.0.5) PRINT *,'SE HA ELEGIDO EL METODO IMPLICITO.'
      IF (TH.GT.0.4999.AND.TH.LT.0.5001)
     &          PRINT *,'SE HA ELEGIDO EL METODO DE CRANK-NICOLSON.'
      PRINT *,'DAR EL VALOR DE DT (TAMAÑO DEL INTERVALO DE TIEMPO)
      READ *,DT
      PRINT *,'DAR EL NUMERO MAXIMO DE INTERVALOS DE TIEMPO '
      READ *,NMAX
      AL=10
      DX=1
      MI= 9                                  !   número total de puntos en la retícula
      GA=AL*DT/DX/DX                         !   especificación de gamma
      ZE=TH*GA
      ET=(1-TH)*GA
      PRINT *, 'Gamma=',GA
      DO I=0, MI
        TB(I)=0                              !   inicialización de T(i)
      END DO
```

```
      TB(0)=0                                  !    condición en la frontera izquierda
      TB(MI+1)=100                             !    condición en la frontera derecha
      DO I=1, MI
         S(I)= 0                               !  inicialización de la fuente
      END DO
      NT=0                                     !  inicialización del tamaño del intervalo
  130 NT=NT+1
      TIME=NT*DT
         PRINT *, ' Intervalo de tiempo=' ,NT,' tiempo=', TIME
      DO I=1, MI                          ! Término fuente de la ecuación tridiagonal
         D(I)=TB(I)+ ET*(TB(I-1)-2*TB(I)+TB(I+1))
      END DO
      D(MI)=D(MI)+ZE*TB(MI+1)
C-------------------------------------
      IF (TH.EQ.0) GOTO 60                 ! Método explícito
C-------------------------------------
      DO I=1, MI
         A(I) = -ZE
         B(I) = 1 + 2*ZE
         C(I)=-ZE
      END DO
      CALL TRID(A,B,C,D,MI)
C-------------------------------------
60    DO I=1, MI
         TB(I)=D(I)
      END DO
      PRINT 70, N,TIME,(TB(I),I=0,MI+1)
      IF(NT.LT.NMAX) GOTO 130
70    FORMAT( I3, F5.2, 11F6.1)
      END
C*******************************
C   Favor de copiar la subrutina TRDG del Programa 10-1.
```

D) Ejemplo de salida

Los resultados determinados en los ejemplos 12.1 y 12.2 se obtuvieron mediante este programa.

PROBLEMAS

12.1) Para analizar el efecto del cambio de Δt sobre los resultados del esquema explícito del ejemplo 12.1, calcule la solución del problema de dicho ejemplo con $\Delta t = 0.001$, 0.01, 0.1 y 0.5 y compare los resultados para $x = 0.5$ en $t = 1, 2, 5, 10$.

12.2) Resuelva el problema del ejemplo 9.16 mediante el método explícito.

12.3) Repita el cálculo del ejemplo 12.1 mediante el método de Crank-Nicolson y el PROGRAMA 12-1.

12.4) Verifique las ecuaciones (12.3.22), (12.3.24) y (12.3.25) mediante el análisis de estabilidad de Fourier.

12.5) El método de dos etapas llamado método explícito de la dirección alternante (EDA) para el caso de la ecuación de conducción del calor está dado por

$$T_i^{(n+1)} - T_j^{(n)} = \gamma(T_{i+1}^{(n)} - T_i^{(n)} + T_i^{(n+1)} - T_{i-1}^{(n+1)})$$
$$T_i^{(n+2)} - T_j^{(n+1)} = \gamma(T_{i+1}^{(n+2)} - T_i^{(n+2)} + T_i^{(n+1)} - T_{i-1}^{(n+1)})$$

donde $\gamma = \alpha\Delta t/\Delta x^2$

Mediante el análisis de estabilidad de Fourier, muestre que el método es incondicionalmente estable.

12.6 a) Mediante el análisis de estabilidad de Fourier, muestre que el siguiente método es incondicionalmente estable:

$$3u_i^{(n+1)} - 4u_i^{(n)} + u_i^{(n-1)} - 2\gamma(u_{i-1}^{(n+1)} - 2u_i^{(n+1)} + u_{i+1}^{(n+1)}) = 0$$

donde $\gamma = \beta\Delta t/(\Delta x)^2$

b) Escriba la EDP a la cuai se aproxima la anterior aproximación por diferencias.

BIBLIOGRAFIA

Brodkey, R. S. y H. C. Hershey, *Transport Phenomena*, McGraw-Hill, 1988.

Ferziger, J. H., *Numerical Methods for Engineering Applications*, Wiley-Interscience, 1988.

Godunov, S. K. y V. S. Ryabenkii, *Difference Schemes*, North-Holland, 1987.

Hetric, L. H., *Dynamics of Nuclear Reactors*, Chicago University Press, 1971.

Incorpera, F. P. y D. P. DeWitt, *Introduction to Heat Transfer*, Wiley, 1985.

Mitchell, A. R. y D. F. Griffiths, *The Finite Difference Method in Partial Differential Equations*, Wiley-Interscience, 1980.

Oran, E. S. y J.P. Boris, *Numerical Simulation of Reactive Flow*, Elsevier, 1987.

Richtmyer, R. D. y K. W. Morton, *Difference Methods for Initial-Value Problems*, Wiley-Interscience, 1957.

Rieder, W. G. y H. R. Busby, *Introductory Engineering Modeling*, Wiley, 1986.

Smith, G. D., *Numerical Solution of Partial Differential Equations*, Oxford University Press, 1978.

Steger, J. y R. F. Warming, "Flux Vector Splitting of the Inviscid Gas Dynamic Equations with Application to Finite Difference Methods", *J. Comp. Phys.*, vol. 40, 1981.

13
Ecuaciones diferenciales parciales hiperbólicas

13.1 INTRODUCCION

Las ecuaciones que rigen el comportamiento del transporte convectivo de la materia y sus cantidades físicas —así como el de las ondas elásticas, acústicas y electromagnéticas— son EDP hiperbólicas. Sin embargo, el progreso tan notable de los esquemas numéricos para las EDP hiperbólicas en los años recientes está ligado íntimamente con el avance en el aspecto computacional de la dinámica de fluidos. Las ecuaciones básicas del flujo de fluidos sin viscosidad son EDP hiperbólicas. Incluso las ecuaciones para los flujos viscosos se pueden analizar como si fueran hiperbólicas si el efecto de la viscosidad es débil. El éxito de una simulación computacional del flujo de un fluido depende de la precisión y eficiencia al resolver las EDP hiperbólicas. A esto se debe que el desarrollo de esquemas numéricos para las EDP hiperbólicas sea un tema de investigación apremiante en la parte computacional de la dinámica de fluidos.

Podemos escribir una EDP hiperbólica tanto en la forma de primer orden como en la de segundo. Es fácil pasar de la primera a la segunda y mostrar que cumple con el criterio de la ecuación (11.1.1a). La mayoría de las EDP hiperbólicas para el transporte de materia y sus propiedades están en la forma de primer orden; en tanto que las referentes a las ondas elásticas, acústicas y electromagnéticas están en la forma de segundo orden. Sin embargo, muchos de los esquemas numéricos para las EDP hiperbólicas se basan en la forma de primer orden. Por ello, en este capítulo estudiaremos las siguientes tres formas de primer orden de las EDP hiperbólicas:

a) EDP hiperbólica lineal en forma conservativa:

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(x, t) = s(x, t) \tag{13.1.1}$$

donde f es una función lineal de u; por ejemplo, $f = a(x)\, u(x, t)$; $a(x)$ es una función dada y $s(x, t)$ es un término fuente.

b) EDP hiperbólica lineal en forma no conservativa:

$$\frac{\partial}{\partial t} u(x, t) + a(x, t) \frac{\partial}{\partial x} u(x, t) = s(x, t) \tag{13.1.2}$$

c) EDP hiperbólica no lineal en forma conservativa:

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(u(x, t)) = s(x, t) \tag{13.1.3}$$

donde f es una función no lineal de u. Las interpretaciones físicas de las ecuaciones (13.1.1) a (13.1.3) aparecen en el apéndice D. En la sección 13.7 se analizan los esquemas numéricos para las EDP no lineales hiperbólicas.

Las aproximaciones por diferencias de las ecuaciones hiperbólicas funcionan mejor cuando la solución es suave. Sin embargo, la solución de las EDP hiperbólicas puede incluir saltos discretos. (Al contrario de la solución de las EDP elípticas o parabólicas, que siempre son continuas en el espacio y el tiempo). Por ejemplo, en un flujo tubular de cierta sustancia química, la concentración del compuesto puede tener un salto súbito. Como otro ejemplo, la distribución espacial del momentum es discontinua cuando ocurre un choque en el flujo de un fluido compresible.

Para mostrar el comportamiento de la solución discontinua de una EDP hiperbólica, consideremos

$$u_t + a u_x = 0, \quad x \geqslant 0, \quad t \geqslant 0 \tag{13.1.4}$$

que es idéntica a la ecuación (13.1.2), excepto que a constante y $s = 0$. La condición inicial es

$$\begin{aligned} u(x, 0) &= 1 \quad \text{si } x \leqslant 1 \\ &= 0 \quad \text{si } x > 1 \end{aligned} \tag{13.1.5}$$

y la condición en la frontera es

$$u(0, t) = 0 \quad \text{si } t > 0 \tag{13.1.6}$$

La solución analítica de este problema es

$$u(x, t) = u(x - at, 0) \tag{13.1.7}$$

(Véase el ejemplo 13.1 para una demostración de cómo se obtiene la solución.) Esta solución representa la onda de una forma rectangular que viaja a la velocidad a, como se muestra en la figura 13.1.

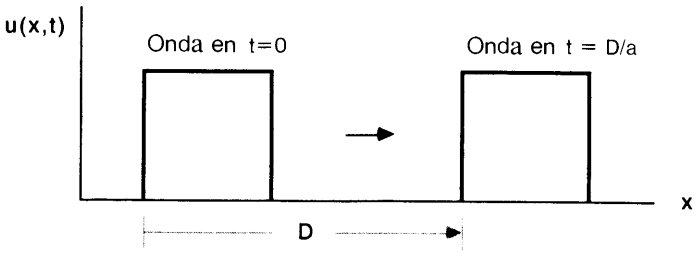


Figura 13.1 Una onda de forma cuadrada que viaja con una velocidad a

La onda en movimiento de forma rectangular presenta ciertas dificultades básicas asociadas con la aproximación por diferencias finitas. En primer lugar, la discontinuidad de la solución no se puede representar en forma exacta sobre la retícula. En segundo lugar, no es fácil una aproximación precisa de las derivadas con respecto del tiempo y el espacio en torno de un salto discreto. Muchos investigadores han estudiado los esquemas numéricos para aumentar su precisión durante la simulación de saltos discretos.

En el resto de este capítulo, nos centraremos en los esquemas numéricos para las EDP hiperbólicas unidimensionales sencillas. Para los lectores que hayan tenido poco contacto con las EDP hiperbólicas, presentamos el concepto de características. Posteriormente se explican los esquemas numéricos con precisión de primer y segundo orden, junto con los aspectos teóricos y computacionales de la estabilidad, difusión y errores de perturbación. Continuamos con el análisis de los errores de los métodos de diferencias. Finalmente, se estudian los esquemas de solución para las EDP hiperbólicas no lineales, así como los métodos de flujo corregido.

13.2 METODO DE CARACTERISTICAS

Esta sección se refiere a las características de una EDP hiperbólica, las cuales son importantes para la comprensión de la solución analítica y los esquemas numéricos [Courant/Hilbert; Garabedian; Mitchell/Griffiths; Smith].

Supongamos que deseamos determinar una solución analítica de

$$u_t + a(x)u_x = s(x, t) \tag{13.2.1}$$

a lo largo de una curva arbitraria, en la cual se encuentran los puntos P y Q a una distancia infinitesimal, como se muestra en la figura 13.2. El cambio de u desde P hasta Q se denota por du y se puede escribir como

$$du = u_t dt + u_x dx$$

Dividimos la ecuación anterior entre dt para obtener

$$\frac{du}{dt} = u_t + u_x \frac{dx}{dt} \tag{13.2.2}$$

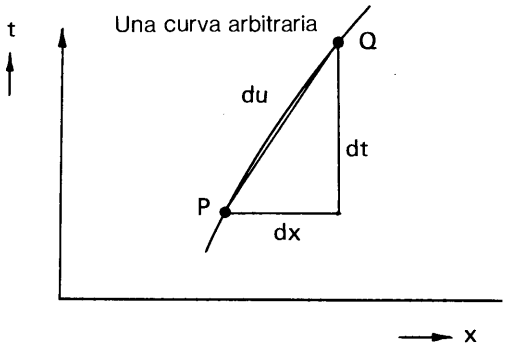


Figura 13.2 Solución a lo largo de una curva

donde dx/dt es el gradiente de la curva PQ en el plano x-t. Si se elige la curva de modo que satisfaga

$$\frac{dx}{dt} = a(x) \tag{13.2.3}$$

entonces el lado derecho de la ecuación (13.2.2) es igual al lado izquierdo de (13.2.1). Así, obtenemos

$$\frac{du}{dt} = s \tag{13.2.4}$$

Por lo tanto, la EDP hiperbólica queda representada por una pareja de EDO, las ecuaciones (13.2.3) y (13.2.4). La primera representa una curva (o línea) en el plano x-t, llamada la *curva* (o *línea*) *característica* mientras que la segunda es una EDO a lo largo de la curva.

Si determinamos la curva $x = x(t)$ mediante la integración de la ecuación (13.2.3), entonces la solución de (13.2.4) se obtiene integrando esta ecuación a lo largo de la curva.

Ejemplo 13.1

Utilice el método de características para resolver (13.1.4) y mostrar que la solución analítica está dada por (13.1.7). Suponga que a = constante, $s = 0$ y que las condiciones iniciales y en la frontera están dadas por las ecuaciones (13.1.5) y (13.1.6), respectivamente.

(Solución)

La ecuación para las líneas características está dada por

$$\frac{dx}{dt} = a \quad \text{(constante)} \tag{A}$$

Al integrar la ecuación (A) se obtiene una línea característica:

$$x = at + b \tag{B}$$

donde b es una constante. Si consideramos la línea característica que pasa por el punto $x = x_0$ en el instante $t = 0$, entonces la ecuación (B) queda

$$x = at + x_0 \tag{C}$$

A lo largo de esta línea, obtenemos la solución de (13.1.1) integrando (13.2.4); es decir, $du/dt = 0$ ya que $s = 0$.

La integración de $du = 0$ da como resultado

$$u(x, t) = k \text{ a lo largo de } x = at + x_0 \tag{D}$$

donde k es una constante que se determina mediante la condición inicial. En $t = 0$ se deben satisfacer las condiciones iniciales. Así, la ecuación (D) queda

$$u(x, t) = \begin{cases} u(x_0, 0) \text{ a lo largo de } & x = at + x_0,\ x_0 \geqslant 0 \\ 0 & x_0 < 0 \end{cases}$$

o, en forma equivalente,

$$u(x, t) = \begin{cases} u(x - at, 0) & \text{si } x \geqslant at \\ 0 & \text{si } x < at \end{cases} \tag{E}$$

en la que hemos eliminado x_0 mediante el uso de $x_0 = x - at$. Así, hemos demostrado la ecuación (13.1.7).

Ejemplo 13.2

Una EDP hiperbólica está dada por

$$u_t + a(x, t)u_x = s(x, t) \tag{A}$$

donde

$$a(x, t) = 3x + 0.1 \tag{B}$$

$$s(x, t) = 1 - x^2 + 0.1t \tag{C}$$

Si la condición inicial está dada por $u(x, 0) = 1$ para $t = 0$, calcule la solución a lo largo de la curva característica que pasa por $x = 0.2$ para $t = 0$.

(Solución)

Mediante la ecuación (B), la ecuación (13.2.3) de la línea característica queda

$$\frac{dx}{dt} = 3x + 0.1, \quad x(0) = 0.2 \tag{D}$$

donde la segunda ecuación es la condición inicial. Este es un problema de EDO con condiciones iniciales. La solución analítica que satisface la condición inicial es

$$x(t) = \frac{1}{3}(0.7e^{3t} - 0.1) \tag{E}$$

Mediante la ecuación (C), la ecuación característica es

$$\frac{du}{dt} = s(x, t) = 1 - x^2 + 0.1t$$

$$= 1 - \frac{1}{9}(0.49e^{6t} - 0.14e^{3t} + 0.01) + 0.1t \qquad \text{(F)}$$

donde se utiliza la ecuación (E), junto con la condición inicial $u(0) = 1$.

Puesto que la ecuación (F) sólo depende de t, ésta se puede integrar de la manera siguiente:

$$u(x, t) = \int_0^t s(x, t')\,dt'$$

$$= t - \frac{1}{9}\left[\frac{0.49}{6}e^{6t} - \frac{0.14}{3}e^{3t} + 0.01t\right] + \frac{0.1}{2}t^2$$

Es posible implantar el método de características con la aproximación por diferencias finitas en una retícula x-t, como se muestra en la figura 13.3.

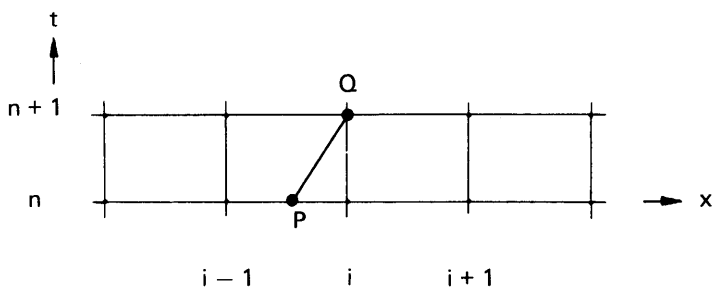


Figura 13.3 Retícula para el método de características

Se traza una línea característica que pase por el punto Q, localizado en $(i, n + 1)$. La intersección de la línea característica y $t = t_n$ se denota por P. Podemos aproximar las dos EDO; a saber, $dx = adt$ y $du = sdt$, a lo largo de un tramo finito de la línea característica como

$$\delta x = a\delta t \quad \text{y} \quad \delta u = s\delta t$$

Al aplicar estas relaciones a la línea PQ (véase la figura 13.3) se obtiene

$$\delta x = x_Q - x_P = a\Delta t, \qquad \delta u = u_Q - u_P = s\Delta t \qquad (13.2.5)$$

Si se conocen los valores de $u_i^{(n)}$ para todos los puntos de la retícula, se puede calcular u_P mediante una interpolación lineal que se escribe como

$$u_P = \frac{\Delta x - a\Delta t}{\Delta x}u_i^{(n)} + \frac{a\Delta t}{\Delta x}u_{i-1}^{(n)}$$

$$= (1 - \gamma)u_i^{(n)} + \gamma u_{i-1}^{(n)} \qquad (13.2.6)$$

donde

$$\gamma = \frac{a\Delta t}{\Delta x}$$

es el número de Courant. Así, u_Q, que es igual a $u_i^{(n)}$, se calcula mediante la sustitución de (13.2.6) en la segunda ecuación de (13.2.5):

$$u_i^{(n+1)} = (1-\gamma)u_i^{(n)} + \gamma u_{i-1}^{(n)} + s\Delta t \tag{13.2.7}$$

Se obtienen los valores de $u_i^{(n+1)}$ para todos los puntos al repetir los mismos cálculos para cada punto. El esquema recibe el nombre de *método explícito de características*.

Si $\gamma = 1$, la ecuación (13.2.7) se reduce a

$$u_i^{(n+1)} = u_{i-1}^{(n)} + s\Delta t \tag{13.2.8}$$

la cual es exacta cuando a y s son ambas constantes.

El método de características en una retícula implica dos tipos de errores: el primero es el efecto de difusión numérica, que introduce errores severos en la solución. El segundo es la inestabilidad. La difusión numérica se debe al uso de la interpolación para calcular u_P cuando $\gamma \neq 1$.

El esquema es estable si

$$\gamma \leqslant 1 \text{ (criterio de estabilidad)}$$

pero es inestable si $\gamma > 1$.

El método de características en una retícula es idéntico al método FTBS que se obtiene en la sección 13.3. En las secciones 13.3 y 13.4 se analizan más detalles de la inestabilidad y la difusión numérica.

Resumen de Esta Sección

a) Se puede reducir una EDP hiperbólica de primer orden a una EDO a lo largo de una curva (o línea) característica. Por lo tanto, se puede resolver al integrar la EDO a lo largo de la línea característica. El esquema de solución basado en este principio recibe el nombre de *método de características*.
b) El método de características en una retícula se desarrolla mediante el uso de un método de interpolación para calcular u en el paso anterior.

13.3 ESQUEMAS DE DIFERENCIAS (EXACTAS) DE PRIMER ORDEN

La mayoría de los esquemas numéricos para las EDP hiperbólicas se basan en las aproximaciones por diferencias finitas. En esta sección, obtendremos aquellos

esquemas básicos por diferencias que tienen una precisión de primer orden. Analizamos la estabilidad de cada esquema mediante el análisis de estabilidad de Fourier.

En toda esta sección estudiaremos la ecuación

$$u_t + au_x = 0 \tag{13.3.1}$$

con

$$u(x, 0) = u_0(x) \quad \text{(condición inicial)} \tag{13.3.2}$$

$$u(0, t) = u_L(t) \quad \text{(condición en la frontera)} \tag{13.3.3}$$

donde a es una constante y $a > 0$. Se puede añadir un término no homogéneo (fuente) del lado derecho sin que esto implique cambios adicionales al siguiente análisis.

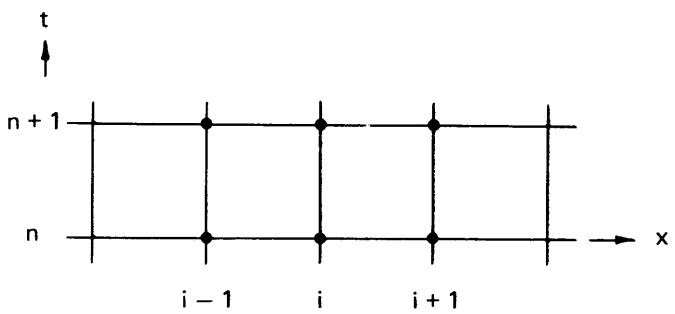


Figura 13.4 Una retícula en el plano x-t

Si consideramos la retícula ilustrada en la figura 13.4, podemos obtener varios esquemas numéricos distintos, según el tipo de aproximación por diferencias elegida para cada u_t y u_x. Puesto que ambos términos son derivadas parciales de primer orden, entre los esquemas candidatos se encuentran las aproximaciones por diferencias hacia atrás, hacia adelante y centrales, tanto en x como en t. Cuando $a > 0$ (el flujo está en la dirección positiva), la aproximación por diferencias hacia atrás

$$u_x = \frac{u_i - u_{i-1}}{\Delta x} \tag{13.3.4}$$

recibe el nombre de *aproximación por diferencias progresivas de primer orden*, donde Δx es el intervalo de la retícula con respecto del espacio, puesto que la aproximación por diferencias se basa en la información del dominio en forma progresiva [Anderson/Tannehill/Pletcher; Patanka]. Si, por otro lado, $a < 0$, la aproximación por diferencias progresivas de primer orden es igual a la aproximación por diferencias hacia adelante

$$u_x = \frac{u_{i+1} - u_i}{\Delta x} \tag{13.3.5}$$

Si consideramos el intervalo de tiempo entre t_n y t_{n+1}, la más simple de las aproximaciones por diferencias de u_t es

$$u_t = \frac{u_i^{(n+1)} - u_i^{(n)}}{\Delta t} \tag{13.3.6}$$

donde $\Delta t = t_{n+1} - t_n$. Todo el esquema se transforma en los esquemas de Euler hacia atrás, hacia adelante o modificado (Crank-Nicolson), dependiendo de si se evalúan las derivadas espaciales en t_n o en t_{n+1}.

Euler hacia adelante en el tiempo y diferencias hacia atrás en el espacio (FTBS, por sus siglas en inglés)

La frase "Euler hacia adelante" indica que u_x se evalúa en t_n, de manera que el esquema es explícito. Cuando $a > 0$, la diferencia hacia atrás es un "esquema progresivo", como ya se explicó anteriormente. Si evaluamos u_x en la ecuación (13.3.1) mediante la aproximación por diferencias hacia atrás en el instante n obtenemos

$$\frac{u_i^{(n+1)} - u_i^{(n)}}{\Delta t} + a\,\frac{u_i^{(n)} - u_{i-1}^{(n)}}{\Delta x} = 0 \tag{13.3.7}$$

Despejamos $u_i^{(n+1)}$ y reescribimos para obtener

$$\begin{aligned} u_i^{(n+1)} &= u_i^{(n)} - \gamma(u_i^{(n)} - u_{i-1}^{(n)}) \\ &= (1-\gamma)u_i^{(n)} + \gamma u_{i-1}^{(n)} \end{aligned} \tag{13.3.8}$$

donde $\gamma = a\Delta t/\Delta x$ es el número de Courant. Cuando $n = 0$, $u_0^{(1)}$ está dado por la condición en la frontera, mientras que todos los valores de $u_i^{(0)}$ están dados por la condición inicial, por lo que podemos evaluar (13.3.8) para todos los puntos de la retícula. Esto también es válido para cualquier intervalo de tiempo, puesto que $u_0^{(n+1)}$ siempre está dado por la condición en la frontera y todos los valores de $u_i^{(n)}$ se conocen a partir del paso anterior. Este esquema resulta ser idéntico al método de características en una retícula.

El esquema FTBS que hemos presentado hasta este momento se basa en la hipótesis de que $a > 0$. Por lo tanto, para $a \leqslant 0$, es necesario cambiar el esquema de diferencias por el esquema de diferencias hacia adelante de forma que se conserve un "esquema progresivo". Si el signo de $a(x, t)$ cambia a mitad del dominio, el esquema se va cambiando de uno al otro. Afortunadamente, los dos casos se pueden escribir en una única ecuación como

$$\frac{u_i^{(n+1)} - u_i^{(n)}}{\Delta t} + a\,\frac{u_{i+1}^{(n)} - u_{i-1}^{(n)}}{2\Delta x} - |a|\Delta x\,\frac{u_{i+1}^{(n)} - 2u_i^{(n)} + u_{i-1}^{(n)}}{2\Delta x^2} = 0 \tag{13.3.9}$$

El segundo término es la aproximación por diferencias centrales para u_x. El tercero es una aproximación por diferencias centrales de $-|a|\Delta x u_{xx}/2$. Podemos interpretar

el esquema FTBS como el uso de la aproximación por diferencias centrales de u_x, sumándole en forma artificial la aproximación por diferencias centrales de $-|a|\Delta x u_{xx}/2$, que recibe el nombre de *término de viscosidad numérica*.

Cuando las condiciones iniciales y en la frontera son todas no negativas, la solución con este esquema nunca es negativa. Esta propiedad es importante por la razón obvia de que, si mediante la ecuación se representa el transporte de material real, la solución nunca debe ser negativa.

A continuación investigamos la estabilidad de este esquema por medio del análisis de estabilidad de Fourier, el cual presentamos en la sección 12.3. En el análisis de estabilidad de Fourier, al término no homogéneo se le asigna el valor de cero. En un dominio infinito desarrollamos la solución en una serie de Fourier. Si consideramos sólo un componente de Fourier a la vez, podemos escribir la solución de la ecuación como

$$u_i^{(n)} = G^n \exp(ij\pi/k)$$

o, en forma equivalente,

$$u_i^{(n)} = G^n \exp(ij\theta) \tag{13.3.10}$$

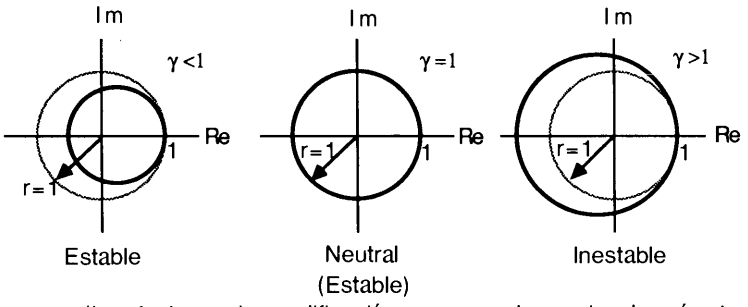
con

$$\theta = \pi/k, \quad k = \pm 1, \pm 2, \pm 3, \ldots, \pm\infty$$

$G = G(\theta)$ es el factor de amplificación (generalmente una función compleja de θ), $j = \sqrt{-1}$, k es la longitud de onda en términos del número de intervalos de la retícula. Puesto que la magnitud mínima de k es 1, la máxima magnitud de θ es π, por lo que $-\pi \leqslant \theta \leqslant \pi$. Sustituimos la ecuación (13.3.10) en (13.3.8) y dividimos entre $\exp(ij\theta)$ para obtener

$$G = 1 - \gamma(1 - e^{-j\theta}) \tag{13.3.11}$$

Figura 13.5 Efecto de γ sobre el factor de amplificación de FTBS



(Los factores de amplificación se encuentran sobre los círculos gruesos)

Podemos examinar la dependencia de $|G(\theta)|$ con respecto a γ graficando ésta en el plano complejo, como se muestra en la figura 13.5. Se observa que si $\gamma \leqslant 1$, la curva que representa a G se encuentra dentro o en el círculo unitario, lo que indica que el factor de amplificación nunca excede a la unidad para todos los valores de θ. Así, el esquema es estable si $\gamma \leq 1$. Sin embargo, si $\gamma > 1$, está por fuera del círculo unitario, por lo que el esquema es inestable.

Euler hacia adelante en el tiempo y diferencia central en el espacio (FTCS, por sus siglas en inglés)

En este caso, se utiliza la aproximación por diferencias centrales en el espacio para aproximar a u_x:

$$u_x = \frac{u_{i+1}^{(n)} - u_{i-1}^{(n)}}{2\Delta x} \tag{13.3.12}$$

Así, la ecuación en diferencias es

$$u_i^{(n+1)} = u_i^{(n)} - \frac{\gamma}{2}(u_{i+1}^{(n)} - u_{i-1}^{(n)}) \tag{13.3.13}$$

El factor de amplificación G de esta ecuación es

$$\begin{aligned} G &= 1 - \frac{\gamma}{2}(e^{j\theta} - e^{-j\theta}) \\ &= 1 - \gamma j \operatorname{sen}(\theta) \end{aligned} \tag{13.3.14}$$

donde $j = \sqrt{-1}$. Su magnitud es

$$|G| = \sqrt{G\overline{G}} = \sqrt{1 + \gamma^2 \operatorname{sen}^2 \theta} \geqslant 1 \quad \text{para toda } \theta \tag{13.3.15}$$

Así, este esquema siempre es inestable.

La ecuación (13.3.13) es idéntica a la (13.3.9), excepto que el tercer término de esta última no aparece en la primera. Esto indica que el tercer término de (13.3.9) juega un papel importante en la estabilización del esquema progresivo de primer orden.

Euler hacia atrás en el tiempo y diferencias centrales en el espacio (BTCS, por sus siglas en inglés)

Con la aproximación de Euler hacia atrás en el tiempo, se utiliza la aproximación por diferencias centrales en el espacio para el instante $n + 1$:

$$\frac{\partial u}{\partial x} = \frac{u_{i+1}^{(n+1)} - u_{i-1}^{(n+1)}}{2\Delta x} \tag{13.3.16}$$

Escribimos la aproximación por diferencias de la ecuación (13.3.1) como

$$-\frac{\gamma}{2}u_{i-1}^{(n+1)} + u_i^{(n+1)} + \frac{\gamma}{2}u_{i+1}^{(n+1)} = u_i^n \tag{13.3.17}$$

El lado izquierdo de esta ecuación tiene tres incógnitas. Mediante la condición en la frontera izquierda, la ecuación (13.3.17) para $i = 1$ es

$$u_1^{(n+1)} + \frac{\gamma}{2}u_2^{(n+1)} = u_1^{(n)} + \frac{\gamma}{2}u_0 \tag{13.3.18}$$

Cuando $a > 0$, el término u_{I+1} (para la ecuación con $i = I$) no está dado a priori. Por lo tanto, es necesaria una condición artificial de frontera para u_{I+1}. Aunque existen varios esquemas alternativos para las condiciones artificiales, un método que se utiliza con frecuencia es el de extrapolar u_{I+1} desde adentro [Yee/Beam/Warming] en la forma siguiente:

$$u_{I+1} = 2u_I - u_{I-1} \tag{13.3.19}$$

Utilizamos esta ecuación para escribir (13.3.17) para $i = I$ como

$$-\gamma u_{I-1}^{(n+1)} + (\gamma + 1)u_I^{(n+1)} = u_I^{(n)} \tag{13.3.20}$$

El conjunto de ecuaciones para $i = 1, 2, \ldots, I$ forma un conjunto tridiagonal de ecuaciones simultáneas.

El factor de amplificación es

$$G = \frac{1}{1 + \frac{\gamma}{2}(e^{j\theta} - e^{-j\theta})}$$

$$= \frac{1}{1 + j\gamma \operatorname{sen}(\theta)} \tag{13.3.21}$$

El valor absoluto de G es

$$|G| = \frac{1}{\sqrt{1 + \gamma^2 \operatorname{sen}^2 \theta}} \leqslant 1 \tag{13.3.22}$$

Por lo tanto, el esquema es incondicionalmente estable.

Resumen de Esta Sección

a) El esquema FTBS, que también recibe el nombre de *esquema explícito progresivo de primer orden*, es estable si γ —el número de Courant— es menor o igual que 1.

b) El esquema FTCS siempre es inestable.

c) El esquema BTCS es incondicionalmente estable.

13.4 ANALISIS DEL ERROR POR TRUNCAMIENTO

Los esquemas numéricos para las EDP hiperbólicas tienen errores que se originan de los errores de truncamiento de las aproximaciones por diferencias. Los errores de truncamiento dan cierta naturaleza artificial a la solución de un esquema numérico. Para analizar el efecto de los errores de truncamiento, se utilizan las ecuaciones modificadas, que son ecuaciones diferenciales similares a las EDP hiperbólicas originales. Las ecuaciones modificadas incluyen todos los efectos de los errores de truncamiento. Se obtienen al sustituir los desarrollos de Taylor por las aproximaciones por diferencias. Tanto las soluciones de las ecuaciones originales y modificadas se pueden obtener en forma analítica en una retícula con espaciamiento uniforme en un dominio infinito. Así, se pueden comparar los efectos de los errores de truncamiento mediante la comparación de las dos soluciones analíticas.

Primero examinamos el esquema FTBS obtenido en la sección anterior:

$$u_i^{(n+1)} - u_i^{(n)} + \gamma[u_i^{(n)} - u_{i-1}^{(n)}] = 0 \tag{13.4.1}$$

Los desarrollos de Taylor de $u^{(n+1)} = u(x_i, t_{n+1})$ y $u_{i-1}^{(n)} = u(x_{i-1}, t_n)$ en torno de $x = x_i$ y $t = t_n$ son, respectivamente,

$$u_i^{(n+1)} = u + \Delta t u_t + \frac{\Delta t^2}{2} u_{tt} + \cdots \tag{13.4.2}$$

$$u_{i-1}^{(n)} = u - \Delta x u_x + \frac{\Delta x^2}{2} u_{xx} - \cdots \tag{13.4.3}$$

donde u sin índice superior denota a $u = u(x_i, t_n)$ y u_t y u_x son derivadas parciales de u en (x_i, t_n). Sustituimos los desarrollos de Taylor en (13.4.1) y obtenemos

$$u_t + \frac{\Delta t}{2} u_{tt} + \frac{\Delta t^2}{6} u_{ttt} + \cdots + \left[a u_x - \frac{a\Delta x}{2} u_{xx} + \frac{a\Delta x^2}{6} u_{xxx} - \cdots \right] = 0 \tag{13.4.4}$$

No es posible analizar la ecuación (13.4.4) en su forma original debido a que incluye derivadas de orden superior tanto en t como en x. Así, la transformamos en una EDP de primer orden con respecto a t eliminando todas las derivadas de orden mayor o igual que dos con respecto a t.

En general, una ecuación dada por

$$u_t + A_2 u_{tt} + A_3 u_{ttt} + \cdots + B_1 u_x + B_2 u_{xx} + B_3 u_{xxx} + B_4 u_{xxxx} + \cdots = 0 \tag{13.4.5}$$

con $B_1 = a$ se puede transformar en

$$u_t + c_1 u_x + c_2 u_{xx} + c_3 u_{xxx} + c_4 u_{xxxx} + \cdots = 0 \tag{13.4.6}$$

donde

$$
\begin{aligned}
c_1 &= B_1 = a \\
c_2 &= B_2 + a^2 A_2 \\
c_3 &= B_3 + 2aA_2B_2 + a^3(2A_2^2 - A_3) \\
c_4 &= B_4 + A_2B_2^2 + 2aA_2B_3 + 6a^2A_2^2B_2 - 3a^2A_3B_2 \\
&\quad + a^4(5A_2^3 - 5A_2A_3 + A_4)
\end{aligned}
\tag{13.4.7}
$$

En el apéndice F se explica la transformación anterior. Las ecuaciones en diferencias, como la (13.4.1) se puede cambiar a la forma de (13.4.6), que recibe el nombre de *ecuación modificada*.

Si aplicamos la relación entre las ecuaciones (13.4.5) y (13.4.6) a (13.4.4), la ecuación modificada es

$$u_t + au_x - \frac{a\Delta x}{2}(1-\gamma)u_{xx} + \frac{a\Delta x^2}{6}(2\gamma^2 - 3\gamma + 1)u_{xxx} + \cdots = 0 \tag{13.4.8}$$

donde

$$c_2 = -\frac{a\Delta x}{2}(1-\gamma)$$

$$c_3 = \frac{a\Delta x^2}{6}(2\gamma^2 - 3\gamma + 1)$$

La ecuación modificada (13.4.8) es una ecuación diferencial que representa a la ecuación en diferencias del esquema FTBS en este análisis. Si comparamos la ecuación (13.4.8) con (13.3.1), vemos que todos los términos distintos del primero son causa de los errores de truncamiento.

Buscamos entonces, en un dominio infinito, la solución analítica de (13.4.8) de la forma

$$u(x, t) = f_\theta(t) \exp(jx\theta) \tag{13.4.9}$$

donde $j = \sqrt{-1}$ y θ tiene el mismo significado definido después de la ecuación (13.3.10) y es una constante relacionada con la frecuencia de un componente de Fourier en el espacio. Sustituimos la ecuación (13.4.9) en (13.4.6) para obtener

$$\frac{d}{dt}f_\theta(t) + [ja\theta - c_2\theta^2 - jc_3\theta^3 + c_4\theta^4 + \cdots]f_\theta(t) = 0 \tag{13.4.10}$$

donde $c_1 = a$. Una solución analítica de (13.4.10) es

$$f_\theta(t) = f_\theta(0) \exp(-ja\theta t + c_2\theta^2 t + jc_3\theta^3 t - c_4\theta^4 t + \cdots) \tag{13.4.11}$$

donde $f_\theta(0)$ queda determinado por una condición inicial. La solución general de (13.4.8) es la suma de (13.4.11) para todos los valores posibles de θ (o, más precisamente, es una integral con respecto de θ desde $-\pi$ hasta π), pero sólo estamos interesados en (13.4.11) para un valor de θ a la vez.

Podemos expresar la ecuación (13.4.11) como

$$f_\theta(t) = f_\theta(0) \exp(-ja\theta t) \exp(c_2\theta^2 t) \exp(jc_3\theta^3 t) \exp(-c_4\theta^4 t) \cdots \tag{13.4.12}$$

[Anderson et al.]. Por otro lado, la solución exacta de (13.3.1) es

$$f_\theta(t) = f_\theta(0) \exp(-ja\theta t) \tag{13.4.13}$$

Así, todos los términos exponenciales excepto el primero de la ecuación (13.4.12) son causa de los errores de truncamiento.

Analizaremos ahora los términos exponenciales de (13.4.12). Si $c_2 > 0$, el segundo término exponencial, $\exp(c_2\theta^2 t)$, crece con el tiempo; es decir, el esquema se vuelve inestable. El valor absoluto del tercer término exponencial, $\exp(jc_3\theta^3 t)$, no cambia con el tiempo puesto que es imaginario. Sin embargo, si el tercer término exponencial se combina con el primero, el producto es

$$\exp(-aj\theta) \exp(jc_3\theta^3 t) = \exp[-j(a - c_3\theta^2)\theta t] \tag{13.4.14}$$

lo que significa que la velocidad de la onda en la solución numérica es $a - c_3\theta^2$ en vez de a. En la solución exacta, la velocidad de la onda a es independiente de θ. La dependencia de la velocidad de la onda con respecto de θ es provocada por la tercera derivada en el error de truncamiento. Mientras mayor sea el valor de θ, más se retarda o adelanta la rapidez de la onda en la solución numérica. El efecto de una velocidad variable de onda se llama *error de perturbación* y provoca la oscilación de la solución numérica, particularmente en donde la solución tiene un cambio espacial pronunciado, como en un choque.

El término de cuarto orden $\exp(-c_4\theta^4 t)$ es creciente o decreciente con respecto al tiempo, según $c_4 <$, o $c_4 > 0$, respectivamente. El efecto combinado de los términos c_2 y c_4 se expresa como

$$\exp[(c_2 - c_4\theta^2)\theta^2 t] \tag{13.4.15}$$

Si $c_2 = 0$, entonces c_4 determina la estabilidad: el esquema es estable si $c_4 > 0$, e inestable si $c_4 < 0$. Si $c_2 < 0$ y $c_4 \geqslant 0$, el esquema es estable. Si $c_2 < 0$ pero $c_4 < 0$, el esquema es estable si $|c_2| > |c_4|\pi^2$, en donde utilizamos el hecho de que el máximo de θ es igual a π, puesto que θ está acotada por $-\pi \leq \theta \leq \pi$. Siempre se satisface la condición de estabilidad si Δt decrece. Si $c_2 > 0$, el esquema es inestable incluso si $c_4 > 0$, puesto que $c_2 - c_4\theta^2$ es positivo para valores pequeños de θ^2.

Concluimos que el esquema FTBS es inestable para $\gamma > 1$ puesto que el término c_2 definido después de (13.4.8) es positivo para $\gamma > 1$. Sin embargo, cuando $\gamma = 1$, c_3 se anula, por lo que no hay error de perturbación. Para $\gamma < 1$, el esquema es es-

table, pero con una desventaja significativa. Es decir, el segundo término exponencial de (13.4.12) amortigua la solución: ésta tiende a cero cuando aumenta el tiempo. El mismo efecto aparece cuando aumenta la distancia que recorre una onda. Puesto que la solución exacta dada por (13.4.3) no tiene tal término, el efecto de amortiguamiento de c_2 negativa es el efecto del error de truncamiento de la ecuación en diferencias, que recibe el nombre de *amortiguamiento numérico* o *efecto numérico de viscosidad de segundo orden*. También una c_4 tiene un efecto de amortiguamiento llamado *efecto numérico de viscosidad de cuarto orden*. Ambos efectos de viscosidad amortiguan más las ondas de las frecuencias espaciales altas, que lo que amortiguan las ondas de frecuencias bajas. Al aumentar la frecuencia espacial de la onda, el efecto de amortiguamiento de cuarto orden crece más rápidamente que el efecto de segundo orden, puesto que el primero es proporcional a θ^4 en el término exponencial, mientras que el segundo es proporcional a θ^2.

Un análisis similar de BTCS da como resultado la siguiente ecuación modificada:

$$u_t + au_x - \frac{1}{2}a^2\Delta t u_{xx} + \left[\frac{1}{6}a(\Delta x)^2 + \frac{1}{3}a^3\Delta t^3\right]u_{xxx} + \cdots \tag{13.4.16}$$

Esta ecuación indica que el esquema es incondicionalmente estable, pues $c_2 = -\frac{1}{2}a^2\Delta t < 0$, pero tiene errores de perturbación pues $c_3 > 0$ siempre.

Resumen de Esta Sección

a) Se analizan los efectos de los errores de truncamiento, transformando una ecuación en diferencias a una ecuación modificada.
b) El esquema es estable si $c_2 \leqslant 0$. Si $c_2 = 0$, entonces es necesario que $c_4 > 0$ para que haya estabilidad. El valor de c_3 no tiene efecto sobre la estabilidad, pero provoca errores de perturbación.
c) Un valor positivo de c_4 tiene efectos fuertes de amortiguamiento en las ondas de frecuencias espaciales altas.

13.5 ESQUEMAS DE ORDEN SUPERIOR

Esquema de Lax-Wendroff

En este caso consideramos

$$u_t + au_x = 0 \text{ (Hacemos } s = 0 \text{ para simplificar la exposición.)} \tag{13.5.1}$$

El desarrollo de Taylor de $u_i^{(n+1)}$ en torno a x_i y t_n es

$$\begin{aligned} u_i^{(n+1)} &= u_i^{(n)} + \Delta t(u_t)_i^n + \frac{1}{2}\Delta t^2(u_{tt})_i^n + \cdots \\ &= u_i^{(n)} - a\Delta t(u_x)_i^n + \frac{1}{2}a^2\Delta t^2(u_{xx})_i^n + \cdots \end{aligned} \tag{13.5.2}$$

de donde hemos eliminado u_t mediante la ecuación (13.5.1) y u_x mediante

$$u_{tt} = -au_{xt} = a^2u_{xx}$$

Si truncamos después del término de segundo orden de (13.5.2) y aplicamos las aproximaciones por diferencias centrales para u_x y u_{xx} obtenemos

$$u_i^{(n+1)} = u_i^{(n)} - \frac{\gamma}{2}(u_{i+1}^{(n)} - u_{i-1}^{(n)}) + \frac{\gamma^2}{2}(u_{i-1}^{(n)} - 2u_i^{(n)} + u_{i+1}^{(n)}) \tag{13.5.3}$$

donde

$$\gamma = a\Delta t/\Delta x$$

La ecuación (13.5.3) es un esquema explícito que recibe el nombre de *esquema de Lax-Wendroff*.

El error de truncamiento del esquema de Lax-Wendroff proviene de dos causas: 1) el truncamiento del desarrollo de Taylor después de la segunda derivada, y 2) las aproximaciones por diferencias centrales para u_x y u_{xx}. El orden del error de truncamiento del desarrollo de Taylor de $u_i^{(n+1)}$ es Δt^3, el orden del error de la aproximación por diferencias centrales de u_x es $\Delta t\Delta x^2$, y el de u_{xx} es $\Delta t^2\Delta x^2$.

El factor de amplificación del esquema de Lax-Wendroff es

$$G = 1 - \gamma^2[1 - \cos(\theta)] - j\gamma \operatorname{sen}(\theta) \tag{13.5.4}$$

El esquema es estable si $0 \leqslant |\gamma| \leqslant 1$. Cuando $\gamma = 1$, el esquema se reduce a $u_i^{(n+1)} = u_{i-1}^{(n)}$ y es exacto.

La ecuación modificada es

$$u_t + au_x - \frac{1}{6}a\Delta x^2(1-\gamma^2)u_{xxx} + \frac{1}{8}a\Delta x^3\gamma(1-\gamma^2)u_{xxxx} + \cdots = 0 \tag{13.5.5}$$

Esta ecuación indica que el error de truncamiento del esquema de Lax-Wendroff se anula si $\gamma = 1$. En el caso $\gamma < 1$, el error principal de truncamiento es la tercera derivada con coeficiente positivo. Así, el esquema tiene una precisión de segundo orden. La magnitud de cada término del error aumenta cuando Δt decrece, pero Δx está fijo (γ tiende a 0). El esquema es estable para $\gamma < 1$, puesto que c_4, el coeficiente del término de la cuarta derivada, cumple que $c_4 > 0$, aunque el término de la derivada de segundo orden se anule.

Esquema de MacCormack

Este esquema es

$$\begin{aligned} \bar{u}_i^{(n+1)} &= u_i^{(n)} - \gamma(u_{i+1}^{(n)} - u_i^{(n)}) \\ u_i^{(n+1)} &= \frac{1}{2}[u_i^{(n)} + \bar{u}_i^{(n+1)} - \gamma(\bar{u}_i^{(n+1)} - \bar{u}_{i-1}^{(n+1)})] \end{aligned} \tag{13.5.6}$$

La primera ecuación es un predictor y la segunda, un corrector. Para el tipo de problemas lineales considerados aquí, se puede eliminar el predictor, sustituyendo la primera ecuación en la segunda, de forma que el esquema de MacCormack quede idéntico al esquema de Lax-Wendroff. La ecuación modificada y el criterio de estabilidad son iguales a los del esquema de Lax-Wendroff.

Esquema progresivo de tercer orden

Podemos aproximar la derivada con respecto al espacio mediante la aproximación por diferencias exactas de tercer orden dadas por

$$a(u_x)_i = \begin{cases} a\dfrac{2u_{i+1} + 3u_i - 6u_{i-1} + u_{i-2}}{6\Delta x} & \text{si } a > 0 \\ a\dfrac{-u_{i+2} + 6u_{i+1} - 3u_i - 2u_{i-1}}{6\Delta x} & \text{si } a < 0 \end{cases}$$

que se pueden escribir juntas en la forma

$$a(u_x)_i = a\frac{-u_{i+2} + 8u_{i+1} - 8u_{i-1} + u_{i-2}}{12\Delta x} + |a|\frac{u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}}{12\Delta x} \tag{13.5.7}$$

[Kawamura; Kawamura/Kuwahara; Leonard]. Se puede demostrar que si se desarrolla cada término en su serie de Taylor, el lado derecho de (13.5.7) se puede escribir como

$$a(u_x)_i + \frac{1}{12}|a|\Delta x^3(u_{xxxx})_i + \cdots \tag{13.5.8}$$

En esta ecuación, el primer término es igual al lado izquierdo de (13.5.7) y el segundo es el error de truncamiento, el cual es proporcional a Δx^3 y también a la cuarta derivada de u.

Por medio de (13.5.7), podemos escribir una aproximación por semidiferencias de (13.5.1) en la forma

$$u_t + a\frac{-u_{i+2} + 8u_{i+1} - 8u_{i-1} + u_{i-2}}{12\Delta x} + |a|\frac{u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}}{12\Delta x} = 0 \tag{13.5.9}$$

La ecuación (13.5.9) no tiene error de perturbación debido a un término de tercera derivada.

Podemos hacer discontinua completamente a la ecuación (13.5.9), sustituyendo los esquemas de Euler hacia adelante o hacia atrás, o el esquema de Crank-Nicolson, para u_t.

Con el esquema de Euler hacia adelante con respecto al tiempo, se obtiene un esquema explícito que se escribe en la forma

$$u_i^{(n+1)} = u_i^{(n)} + \Delta t\left[-a\frac{-u_{i+2}^{(n)} + 8u_{i+1}^{(n)} - 8u_{i-1}^{(n)} + u_{i-2}^{(n)}}{12\Delta x} - |a|\frac{u_{i+2}^{(n)} - 4u_{i+1}^{(n)} + 6u_i^{(n)} - 4u_{i-1}^{(n)} + u_{i-2}^{(n)}}{12\Delta x}\right] \tag{13.5.10}$$

Para estudiar los efectos de los errores de truncamiento en el esquema explícito, calculamos la ecuación modificada. Utilizamos los desarrollos de Taylor en (13.5.10) para obtener

$$u_t + \frac{\Delta t}{2}u_{tt} + \frac{\Delta t^2}{6}u_{ttt} + \cdots + au_x + \frac{1}{12}|a|u_{xxxx} + \cdots = 0 \tag{13.5.11}$$

en donde también utilizamos la ecuación (13.5.8). Eliminamos las derivadas de orden mayor o igual que dos con respecto a t —según el algoritmo descrito en la sección anterior— y obtenemos

$$u_t + au_x + a^2\Delta t u_{xx} + \frac{5a^3\Delta t^2}{6}u_{xxx} + \left(\frac{|a|\Delta x^3}{12} + \frac{a^4\Delta t^3}{4}\right)u_{xxxx} + \cdots = 0 \tag{13.5.12}$$

El término principal del error de truncamiento es (una derivada) de segundo orden y tiene signo positivo, por lo que el esquema global con base en el esquema de Euler hacia adelante se reduce a un esquema con una precisión de primer orden. El término de segundo orden con signo positivo tiene un efecto antidifusivo que provoca la inestabilidad del esquema, a menos que dicho efecto se haga más pequeño que el efecto del error de truncamiento de cuarto orden. Sin embargo, al disminuir Δt, los errores de segundo y tercer orden tienden a cero, por lo que se pueden hacer tan pequeños como se quiera, con el costo de utilizar un Δt muy pequeño.

Si se utiliza la diferencia de Euler hacia atrás con respecto del tiempo, el esquema se convierte en implícito:

$$u_i^{(n+1)} + a\frac{-u_{i+2}^{(n+1)} + 8u_{i+1}^{(n+1)} - 8u_{i-1}^{(n+1)} + u_{i-2}^{(n+1)}}{12\Delta x} + |a|\frac{u_{i+2}^{(n+1)} - 4u_{i+1}^{(n+1)} + 6u_i^{(n+1)} - 4u_{i-1}^{(n+1)} + u_{i-2}^{(n+1)}}{12\Delta x} = u_i^{(n)} \tag{13.5.13}$$

El conjunto de ecuaciones simultáneas debe resolverse mediante un esquema pentadiagonal en cada intervalo de tiempo. El esquema implícito es incondicionalmente estable. El análisis de la ecuación modificada revela que el término principal del error es de segundo orden, al igual que en la versión explícita de Euler hacia adelante (pero de signo opuesto). Los términos del error de segundo y tercer orden se pueden

hacer decrecer utilizando un Δt, pequeño, pero en ese caso desaparece el beneficio del uso de un esquema implícito.

Un esquema explícito con precisión de segundo orden con respecto al tiempo se basa en el predictor de Adams-Bashfort y se escribe como [Kawamura]

$$\frac{u_i^{(n+1)} - u_i^{(n)}}{\Delta t} + \frac{1}{2}[3F^{(n)} - F^{(n-1)}] = 0 \tag{13.5.14}$$

donde F_n es la aproximación por diferencias de tercer orden para au_x. Debido a la precisión de segundo orden con respecto al tiempo, la ecuación modificada de (13.5.14) no incluye el término u_{xx} (el cual es la causa del efecto antidifusivo con respecto al tiempo del esquema de Euler hacia adelante). Por lo tanto, la ecuación (13.5.14) es más estable que (13.5.13).

Resumen de Esta Sección

a) Los esquemas de Lax-Wendroff y MacCormack tienen una precisión de segundo orden. Dicha precisión es la mejor cuando $\gamma = 1$. Por lo tanto, la precisión disminuye incluso al disminuir Δt, a menos que también decrezca Δx.
b) El esquema progresivo de tercer orden tiene una precisión de tercer orden con respecto al espacio, pero las diferencias de Euler hacia adelante o hacia atrás en el tiempo introducen errores de segundo orden. Sin embargo, se puede hacer más pequeña la magnitud de los errores de segundo orden utilizando un Δt independiente de Δx.

13.6 ESQUEMAS DE DIFERENCIAS EN LA FORMA CONSERVATIVA

En las secciones anteriores analizamos las ecuaciones en diferencias en las formas no conservativas. Con la forma no conservativa, se puede ganar o perder el total de la propiedad en todo el dominio en cada intervalo de tiempo debido a los errores numéricos, y tales efectos se pueden acumular con el paso del tiempo. Si se escribe una ecuación en diferencias en la forma conservativa, la suma de las ecuaciones con respecto al espacio satisface la conservación de la propiedad en todo el dominio.

Para analizar el concepto de la forma conservativa de una EDP hiperbólica, consideremos el flujo de un fluido incompresible en un tubo recto con sección transversal constante. La ecuación de continuidad es

$$\frac{\partial}{\partial t}\rho(x, t) + \frac{\partial}{\partial x} f(x, t) = 0 \tag{13.6.1}$$

donde $p(x, t)$ es la densidad del fluido, $f(x, t) = u(x, t)\, p(x, t)$ es la razón de flujo de masa por unidad de área transversal y $u(x, t)$ es la velocidad del fluido. La

ecuación (13.6.1) representa la conservación de la masa. Si integramos la ecuación con respecto del espacio, desde $x = a$ hasta $x = b$, obtenemos

$$\frac{d}{dt}\left[\int_a^b \rho(x, t)\,dx\right] = f(a, t) - f(b, t) \tag{13.6.2}$$

Con la hipótesis de área transversal unitaria, el lado izquierdo es la razón de cambio en la masa total en $a < x < b$, el primer término de la derecha es la razón de flujo de masa que entra por $x = a$, y el segundo término es el análogo que sale por $x = b$. Así, la ecuación (13.6.2) representa la conservación de la masa en $a < x < b$.

Podemos escribir de otra forma la ecuación (13.6.1), después de sustituir $f = u(x, t)p(x, t)$ y derivar en el segundo término:

$$\frac{\partial}{\partial t}\rho(x, t) + \rho(x, t)\frac{\partial}{\partial x}u(x, t) + u(x, t)\frac{\partial}{\partial x}\rho(x, t) = 0 \tag{13.6.3}$$

La ecuación (13.6.3) es matemáticamente equivalente a (13.6.1), pero de esta forma no se puede explicar de manera inmediata la conservación de la masa, por lo que se pierde el significado físico de la conservación. Así, diremos que (13.6.3) es una forma no conservativa y (13.6.1) es una forma conservativa.

Las diferencias entre estas dos formas son importantes para las ecuaciones en diferencias de las EDP. La forma conservativa de una ecuación en diferencias siempre se puede escribir como

$$\frac{\rho_i^{(n+1)} - \rho_i^{(n)}}{\Delta t} + \frac{g_{i+\frac{1}{2}} - g_{i-\frac{1}{2}}}{\Delta x} = 0 \tag{13.6.4}$$

donde $g_{i+(1/2)}$ es una aproximación numérica de f en $x_{i+(1/2)}$ y generalmente es una función de f_{i+1} y f_i. Existe libertad en la elección de la forma particular de $g_{i+(1/2)}$. Por ejemplo, si

$$g_{i+\frac{1}{2}} = \frac{f_{i+1} + f_i}{2} \tag{13.6.5}$$

entonces el segundo término de (13.6.4) se reduce a

$$\frac{\rho_i^{(n+1)} - \rho_i^{(n)}}{\Delta t} + \frac{f_{i+1} - f_{i-1}}{2\Delta x} = 0 \tag{13.6.6}$$

que es una aproximación por diferencias centrales; sin embargo, es inestable.

Como otro ejemplo, $g_{i+(1/2)}$ se puede escribir como

$$g_{i+\frac{1}{2}} = \frac{1}{2}(f_{i+1} + f_i) - |a_{i+\frac{1}{2}}|(u_{i+1} - u_i) \tag{13.6.7}$$

donde

$$a_{i+\frac{1}{2}} = \frac{f_{i+1} - f_i}{u_{i+1} - u_i}$$

Con esta elección, la ecuación (13.6.4) queda como

$$\frac{\rho_i^{(n+1)} - \rho_i^{(n)}}{\Delta t} + \frac{f_{i+1} - f_{i-1}}{2\Delta x} + \frac{-|a_{i+\frac{1}{2}}|u_{i+1} + (|a_{i+\frac{1}{2}}| + |a_{i-\frac{1}{2}}|)u_i - |a_{i-\frac{1}{2}}|u_{i-1}}{2\Delta x} = 0 \qquad (13.6.8)$$

Podemos interpretar al último término como un término de viscosidad numérica.

En la ecuación (13.6.4), g recibe el nombre de flujo numérico ya que —como lo muestra la ecuación (13.6.7) —consta de un flujo de masa y un término artificial que da como resultado un efecto de viscosidad.

La razón por la que (13.6.6) está en forma conservativa es obvia: sumamos (13.6.4) para $i = j, j + 1, \ldots, k$ y reagrupamos para obtener

$$\Delta x \sum_{i=j}^{k} \rho_i^{(n+1)} - \Delta x \sum_{i=j}^{k} \rho_i^{(n)} = -\Delta t(g_{k+\frac{1}{2}} - g_{j-\frac{1}{2}}) \qquad (13.6.9)$$

El lado izquierdo es el cambio de la masa total en $[x_{j-(1/2)}, x_{k+(1/2)}]$ entre el tiempo t_n y t_{n+1}. El primer término del lado derecho es el flujo total de la cantidad numérica en $x_{k+(1/2)}$ en Δt; el segundo término es su análogo en $x_{j-(1/2)}$. La ecuación (13.6.9) indica que la masa total en la porción del tubo considerada, está determinada por el flujo numérico en los dos extremos. Si las condiciones en la frontera para los flujos numéricos en los extremos son iguales precisamente al flujo de masa, la ecuación (13.6.9) mantiene el balance de masa en el tubo. Es importante observar que si el esquema numérico está en forma conservativa, la elección particular de una aproximación numérica no afecta la masa total.

Ejemplo 13.3

Escriba la aproximación por diferencias en forma conservativa, que sea explícita de primer orden con respecto al tiempo y con una aproximación por diferencias progresivas de tercer orden con respecto al espacio, para la ecuación

$$u_t(x, t) + [a(x)u(x, t)]_x = 0$$

(Solución)

La aproximación explícita por diferencias con base en la diferenciación progresiva de tercer orden se puede escribir de la forma siguiente:

$$\frac{u_i^{(n+1)} - u_i^{(n)}}{\Delta t} + \frac{G_{i+\frac{1}{2}} - G_{i-\frac{1}{2}}}{\Delta x} = 0 \qquad (A)$$

donde G es un flujo dado por

$$G_{i+\frac{1}{2}} = \frac{-(au)_{i+2}^{(n)} + 7(au)_{i+1}^{(n)} + 7(au)_i^{(n)} - (au)_{i-1}^{(n)}}{12\Delta x} + \frac{(|a|u)_{i+2}^{(n)} - 3(|a|u)_{i+1}^{(n)} + 3(|a|u)_i^{(n)} - (|a|u)_{i-1}^{(n)}}{12\Delta x} \qquad \text{(B)}$$

La ecuación (A) se reduce a la ecuación (13.5.9) si a = constante.

Las ecuaciones en diferencias pueden pasar a la forma no conservativa por varias razones. Una de las causas principales de que surja la forma no conservativa es la de obtener las ecuaciones en diferencias a partir de una forma no conservativa de la EDP. Las ecuaciones en diferencias obtenidas de esta manera no se pueden escribir como (13.6.4). Un ejemplo de forma no conservativa es la aproximación por diferencias progresivas (FTBS cuando $u > 0$). Si u cambia de signo positivo a negativo entre dos puntos consecutivos de la retícula (digamos, i e $i + 1$) las ecuaciones en diferencias son

$$\frac{\rho_i^{(n+1)} - \rho_i^{(n)}}{\Delta t} + \frac{u_i\rho_i^{(n)} - u_{i-1}\rho_{i-1}^{(n)}}{\Delta x} = 0, \quad u_i > 0 \qquad (13.6.10)$$

y

$$\frac{\rho_{i+1}^{(n+1)} - \rho_{i+1}^{(n)}}{\Delta t} + \frac{u_{i+2}\rho_{i+2}^{(n)} - u_{i+1}\rho_{i+1}^{(n)}}{\Delta x} = 0, \quad u_{i+1} < 0 \qquad (13.6.11)$$

Al sumar las ecuaciones (13.6.10) y (13.6.11), no se cancela el flujo en la interfase entre los puntos i e $i + 1$ de la retícula, por lo que no se cumple la conservación.

También puede surgir una forma no conservativa cuando el análisis geométrico no es el apropiado. Consideremos la ecuación

$$\frac{\partial}{\partial t} A(x)\rho(x, t) + \frac{\partial}{\partial x} A(x)u(x, t)\rho(x, t) = 0 \qquad (13.6.12)$$

que es una ley de conservación de un flujo unidimensional con áreas transversales variables y donde $A(x)$ es el área de la sección transversal en x. Si diferenciamos el segundo término obtenemos

$$A(x)\frac{\partial}{\partial t}\rho(x, t) + A(x)\frac{\partial}{\partial x}u(x, t)\rho(x, t) + A_x(x)u(x, t)\rho(x, t) = 0 \qquad (13.6.13)$$

Podemos entonces escribir, por ejemplo, una aproximación por diferencias de (13.6.13) de la manera siguiente

$$A(x_i)\frac{\rho_i^{(n+1)} - \rho_i^{(n)}}{\Delta t} + A(x_i)\frac{u_{i+(1/2)}^{(n)}\rho_{i+(1/2)}^{(n)} - u_{i-(1/2)}^{(n)}\rho_{i-(1/2)}^{(n)}}{\Delta x} + (A_x)_i u_i^{(n)}\rho_i^{(n)} = 0 \qquad (13.6.14)$$

Cuando se sumen las ecuaciones para los puntos consecutivos de la retícula, no se cancelan los términos del flujo, por lo que (13.6.14) no cumple con la conservación.

En la solución real de una EDP hiperbólica, se utilizan tanto la forma conservativa como la no conservativa. Es frecuente que se utilice una forma no conservati-

va para que el algoritmo de solución sea más sencillo. Sin embargo, se preferirá —siempre que sea posible— la forma conservativa.

Resumen de Esta Sección

a) Se utilizan tanto la forma conservativa como la no conservativa de las ecuaciones en diferencias, pero la primera es la más recomendable.
b) Las ecuaciones en diferencias conservativas se obtienen a partir de una EDP hiperbólica en forma conservativa.
c) Las ecuaciones en diferencias con forma conservativa siempre se pueden escribir como la ecuación (13.6.4).

13.7 COMPARACION DE LOS ESQUEMAS MEDIANTE ONDAS DE PRUEBAS

Un eficaz método para analizar el desempeño de un esquema es el de resolver problemas de prueba. Aquí resolveremos la ecuación

$$u_t + u_x = 0$$

con las condiciones iniciales de una onda cuadrada. La solución exacta de la ecuación para cualquier tiempo tiene la misma forma cuadrada de la distribución inicial, pero la localización de la onda avanza de manera continua con velocidad unitaria.

Durante las pruebas, el espacio en la retícula es $\Delta x = 0.1$ y el tamaño del intervalo de tiempo $\Delta t = p\Delta x$ es $\Delta t = p = 0.01$. La figura 13.6 muestra los resultados de los cálculos para la onda cuadrada de los siguientes esquemas numéricos:

FTBS
Lax-Wendroff
Explícito progresivo de tercer orden

La solución numérica mediante el esquema FTBS nunca toma valores negativos, ni tampoco presenta un comportamiento oscilatorio. Sin embargo, la onda tiende a expanderse y aplastarse al viajar. La altura de la onda es cada vez menor y su ancho cada vez mayor.

En el esquema de Lax-Wendroff hay menos efectos de aplastamiento de las ondas con respecto del esquema anterior, pero tiene un comportamiento oscilatorio significativo, que corresponde al error de perturbación asociado al término de tercer orden del error de truncamiento. Esta tendencia aumenta al tender γ a 0.

Con el esquema progresivo de tercer orden, se mantienen mejor la altura y el ancho de la onda cuadrada. Sin embargo, las alturas de la onda se vuelven negativas en el inicio y final de cada onda. Esta tendencia de oscilación en torno a un cambio drástico de la solución se debe al error de perturbación. El esquema de flujo corregido de la sección 13.9 se aproxima mejor a la onda viajera que los ejemplos dados en esta sección.

13.8 ESQUEMAS NUMERICOS PARA EDP HIPERBOLICAS NO LINEALES

Las EDP hiperbólicas no lineales se pueden escribir en la forma conservativa

$$u_t + F_x = 0 \tag{13.8.1}$$

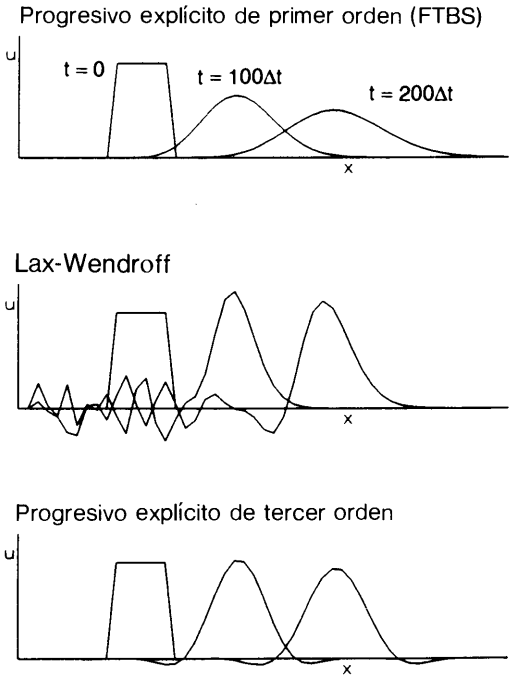


Figura 13.6 Simulación numérica de una onda cuadrada que se mueve con velocidad constante a

donde $F = F(u)$ es una función lineal de u. Podemos también escribir (13.8.1) en forma no conservativa:

$$u_t + \frac{\partial F}{\partial u} u_x = 0 \tag{13.8.2}$$

Esquema de Courant-Isaacson-Rees

El esquema FTBS no funciona en su forma original para el caso de una EDP no lineal hiperbólica. Sin embargo, se han desarrollado esquemas similares a los FTBS para EDP hiperbólicos lineales, los cuales satisfacen los siguientes criterios:

a) F_x es aproximado mediante diferencias centrales.
b) Se agrega un término de difusión numérica.
c) Las ecuaciones en diferencias están en la forma conservativa.

Los esquemas de Courant-Isaacson-Rees y de Lax-Friedrich se ubican dentro de esta categoría. El primero está dado por

$$u_i^{(n+1)} = u_i^{(n)} - k(f_{i+(1/2)}^{(n)} - f_{i-(1/2)}^{(n)}) \tag{13.8.3}$$

donde $k = \Delta t/\Delta x$ y, omitiendo los índices superiores (x) de F y u.

$$f_{i+\frac{1}{2}} = \frac{1}{2}[F_i + F_{i+1} - |a_{i+\frac{1}{2}}|(u_{i+1} - u_i)]$$

$$a_{i+\frac{1}{2}} = \begin{cases} \dfrac{F_{i+1} - F_i}{u_{i+1} - u_i}, & \text{si } u_{i+1} - u_i \neq 0 \\ 0, & \text{si } u_{i+1} - u_i = 0 \end{cases} \tag{13.8.4}$$

Si eliminamos $f_{i+\frac{1}{2}}$, la ecuación (13.8.3) se transforma en

$$\begin{aligned} u_i^{n+1} = u_i^n &- \frac{\Delta t}{2\Delta x}(F_{i+1} - F_{i-1}) \\ &- \frac{\Delta t}{2\Delta x}[-|F_{i+1} - F_i| \operatorname{sign}(u_{i+1} - u_i) \\ &+ |F_i - F_{i-1}| \operatorname{sign}(u_i - u_{i-1})] \end{aligned} \tag{13.8.5}$$

El último término de esta ecuación es el correspondiente a la difusión numérica.

Esquema de Lax-Wendroff

La deducción de este esquema para el caso de las EDP hiperbólicas no lineales es esencialmente la misma que para la versión lineal: se parte del desarrollo de Taylor de $u_i^{(n+1)}$:

$$u_i^{(n+1)} = u_i^{(n)} + \Delta t(u_t)_i^{(n)} + \frac{\Delta t^2}{2}(u_{tt})_i^{(n)} + \cdots \tag{13.8.6}$$

La u_t de la última ecuación se elimina mediante (13.8.1), en tanto que u_{tt} se elimina de la manera siguiente:

Al derivar F con respecto de t obtenemos

$$F_t = F_u u_t = A u_t \tag{13.8.7}$$

donde $A = F_u$. Si eliminamos u_t de la ecuación (13.8.7) mediante (13.8.1), obtenemos

$$F_t = -AF_x \tag{13.8.8}$$

Al derivar (13.8.1) con respecto a t se obtiene

$$u_{tt} = -F_{xt} = -\frac{\partial}{\partial x}F_t = \frac{\partial}{\partial x}AF_x \tag{13.8.9}$$

Al eliminar u_t y u_{tt} de (13.8.6) se obtiene el esquema de Lax-Wendroff:

$$u_i^{(n+1)} = u_i^{(n)} - \frac{\Delta t}{2\Delta x}(F_{i+1}^{(n)} - F_{i-1}^{(n)})$$
$$+ \frac{1}{2}\left(\frac{\Delta t}{\Delta x}\right)^2 [A_{i+(1/2)}^{(n)}(F_{i+1}^{(n)} - F_i^{(n)}) - A_{i-(1/2)}^{(n)}(F_i^{(n)} - F_{i-1}^{(n)})] \quad (13.8.10)$$

el cual también se puede escribir en la forma

$$u_i^{(n+1)} = u_i^{(n)} - k(f_{i+(1/2)}^{(n)} - f_{i-(1/2)}^{(n)}) \quad (13.8.11)$$

donde f recibe el nombre de *flujo numérico* y se define como

$$f_{i+\frac{1}{2}} = \frac{1}{2}[F_{i+1}^{(n)} + F_i^{(n)} - kA_{i+(1/2)}^{(n)}(F_{i+1}^{(n)} - F_i^{(n)})]$$
$$f_{i-\frac{1}{2}} = \frac{1}{2}[F_i^{(n)} + F_{i-1}^{(n)} - kA_{i-(1/2)}^{(n)}(F_i^n - F_{i-1}^n)] \quad (13.8.12)$$

donde $k = \Delta t/\Delta x$.

Esquema de MacCormack

El esquema de MacCormack [von Lavante/Thompkins] para una EDP no lineal está dado por

$$\bar{u}_i^{(n+1)} = u_i^{(n)} - \frac{\Delta t}{\Delta x}(F_{i+1}^{(n)} - F_i^{(n)})$$
$$u_i^{(n+1)} = \frac{1}{2}\left[u_i^{(n)} + \bar{u}_i^{(n+1)} - \frac{\Delta t}{\Delta x}(\bar{F}_i - \bar{F}_{i-1})\right] \quad (13.8.13)$$

donde la primera ecuación es el predictor, la segunda es el corrector y

$$\bar{F}_i = F(\bar{u}_i^{(n+1)})$$

Como se vio en un análisis anterior, este esquema tiene el mismo orden de precisión que el esquema de Lax-Wendroff, pero es más fácil de usar puesto que se evalúan en forma directa los valores de F.

Esquema implícito de Beam-Warming

Este esquema empleado en la ecuación (13.8.1) parte del esquema modificado de Euler y la aproximación por diferencias centrales con respecto al espacio:

$$u_i^{(n+1)} - u_i^{(n)} = \frac{\Delta t}{4\Delta x}[F_{i+1}^{(n+1)} - F_{i-1}^{(n+1)} + F_{i+1}^{(n)} - F_{i-1}^{(n)}] \quad (13.8.14)$$

En la ecuación (13.8.14), $F_{i\pm 1}^{(n+1)}$ son funciones no lineales de las incógnitas $u_{i\pm 1}^{(n+1)}$. Así, las desarrollamos en series de Taylor

$$F_i^{(n+1)} = F_i^{(n)} + \Delta t A_i^{(n)} \delta u_i + \cdots$$

donde

$$\delta u_i = u_i^{(n+1)} - u_i^{(n)} \tag{13.8.15}$$

$$A_i^{(n)} = \left(\frac{\partial F}{\partial u}\right)_i^{(n)} \tag{13.8.16}$$

Sustituimos (13.8.15) y (13.8.16) en (13.8.14) para obtener

$$-\Delta t A_{i-1} \delta u_{i-1} + \delta u_i + \Delta t A_{i+1} \delta u_{i+1} = \frac{\Delta t}{2\Delta x}\left[F_{i+1}^{(n)} - F_{i-1}^{(n)}\right] \tag{13.8.17}$$

donde omitimos los índices superiores de las A_i para simplificar, pero están evaluados en $u_i^{(n)}$. El conjunto de las ecuaciones (13.8.7) se resuelve para todos los puntos en forma simultánea mediante el esquema tridiagonal. El requerimiento sobre la condición numérica de la frontera derecha, se satisface de igual forma que como se describe en el esquema implícito de la sección 13.3.

La estabilidad lineal de este esquema es neutra. Sin embargo, la no linealidad de la ecuación provoca con frecuencia la inestabilidad, de forma que una práctica común es la de añadir un término de viscosidad numérica de cuarto orden:

$$-\Delta t A_{i-1} \delta u_{i-1} + \delta u_i + \Delta t A_{i+1} \delta u_{i+1} = \frac{\Delta t}{2\Delta x}\left[F_{i+1}^{(n)} - F_{i-1}^{(n)}\right]$$
$$- \frac{\varepsilon}{\Delta x^4}\left(u_{i-2}^{(n)} - 4u_{i-1}^{(n)} + 6u_i^{(n)} - 4u_{i+1}^{(n)} + u_{i+2}^{(n)}\right)$$

en donde el último término es el de viscosidad numérica de cuarto orden y ε es el coeficiente artificial de viscosidad.

13.9 ESQUEMAS DE FLUJO CORREGIDO

Las pruebas de la sección 13.7 muestran que la difusión numérica y el error de perturbación son problemas importantes inherentes a los esquemas numéricos para las EDP hiperbólicas. La positividad de la solución se ve violada por los efectos de perturbación de los términos truncados. En general, si se utiliza un esquema de orden alto, se reduce la difusión numérica pero surge el error de perturbación. Por otro lado, la supresión del error de perturbación implica un aumento del efecto de la viscosidad numérica (o difusión numérica). Así, es imposible suprimir la difusión numérica y eliminar los errores de perturbación al mismo tiempo en un único esquema de diferencias [Oran y Boris].

Para mejorar la solución, se han propuesto muchos esquemas con distintos nombres, entre los cuales se encuentran el *esquema de flujo corregido* y el esquema de disminución de la variación total (DVT) [Yee; Yee/Warming/Harten; Nittmann; Book/Boris/Zalesak]. El principio fundamental de estos esquemas es el uso de un esquema de orden bajo en el que haya riesgo de oscilación, pero utilizando también un esquema de orden alto, siempre que el efecto del error de perturbación no sea tan grande como para provocar la oscilación. En cada intervalo de tiempo, se calculan tanto una solución de orden bajo (con una viscosidad numérica de segundo orden) como una de orden alto (sin el efecto de la viscosidad numérica de segundo orden); se obtiene finalmente la solución para ese intervalo de tiempo mezclando los resultados de ambos cálculos.

Explicaremos el esquema de flujo corregido para una EDP hiperbólica dada en forma conservativa por

$$u_t + F_x = 0 \tag{13.9.1}$$

El esquema consta de dos partes: la primera es un esquema numérico de primer orden dado por

$$\bar{u}_i^{(n+1)} = u_i^{(n)} - \frac{\Delta t}{\Delta x}(g_{i+(1/2)}^{(n)} - g_{i-(1/2)}^{(n)}) \tag{13.9.2}$$

donde g es un flujo numérico. Si utilizamos el esquema de Courant-Isaacson-Rees en esta parte, escribimos a $g_{i+(1/2)}^{(n)}$ como
como

$$g_{i+(1/2)}^{(n)} = \frac{1}{2}[F_i^{(n)} + F_{i+1}^{(n)} - |a_{i+\frac{1}{2}}|(u_{i+1}^{(n)} - u_i^{(n)})]$$

$$a_{i+\frac{1}{2}} = \begin{cases} \dfrac{F_{i+1} - F_i}{u_{i+1}^{(n)} - u_i^{(n)}}, & \text{si } u_{i+1}^{(n)} - u_i^{(n)} \neq 0 \\ 0, & \text{si } u_{i+1}^{(n)} - u_i^{(n)} = 0 \end{cases} \tag{13.9.3}$$

La segunda parte recibe el nombre de *proceso de antidifusión* y se escribe como

$$u_i^{(n+1)} = \bar{u}_i^{(n+1)} - \frac{\Delta t}{\Delta x}(\delta f_{i+\frac{1}{2}} - \delta f_{i-\frac{1}{2}}) \tag{13.9.4}$$

donde δf corrige al flujo y su propósito es el de cancelar el efecto de la viscosidad (o difusión) numérica de la primera parte.

Es difícil determinar en forma analítica la magnitud del efecto de difusión que debe cancelarse ya que, en los problemas reales, el espaciamiento de la retícula y los coeficientes (tales como la velocidad del fluido y la sección transversal de un tubo) cambian con respecto del espacio. A esto se debe que utilicemos un esquema de orden alto, que no tenga efecto de difusión de segundo orden.

Un esquema de orden alto sería

$$U_i^{(n+1)} = u_i^{(n)} - \frac{\Delta t}{\Delta x}(G_{i+(1/2)}^{(n)} - G_{i-(1/2)}^{(n)}) \tag{13.9.5}$$

La diferencia entre el valor de $u_i^{(n+1)}$ obtenido mediante el esquema de primer orden y su valor obtenido mediante el esquema de orden mayor es, en primera instancia, el efecto de difusión numérica de segundo orden. Por lo tanto, para quitar el efecto de difusión numérica, podemos hacer $\delta f_{i+\frac{1}{2}}$ igual a

$$\delta f_{i+\frac{1}{2}} = G_{i+(1/2)}^{(n)} - g_{i+(1/2)}^{(n)} \tag{13.9.6}$$

Si elegimos este valor de $\delta f_{i+(1/2)}$ para todos los puntos, entonces la segunda parte simplemente es el esquema de orden alto. Por lo tanto, son necesarios ciertos ajustes de δf de punto a punto. El ajuste es tal que $\delta f_{i+(1/2)}$ no sufre ningún cambio mientras que no haya riesgo de una oscilación ficticia. Sin embargo, si existe riesgo de oscilación, entonces debemos reducir o incluso anular el valor de δf en ese intervalo. El algoritmo con base en este concepto está dado por

$$\delta f_{i+\frac{1}{2}} = S \text{ máx}\{0, \text{mín } [S(\bar{u}_{i+2}^{(n+1)} - \bar{u}_{i+1}^{(n+1)}), |\delta \hat{f}_{i+\frac{1}{2}}|, S(\bar{u}_i^{(n+1)} - \bar{u}_{i-1}^{(n+1)})]\} \tag{13.9.7}$$

con

$$\delta \hat{f}_{i+\frac{1}{2}} = G_{i+(1/2)}^{(n)} - g_{i+(1/2)}^{(n)}$$

$$S = \text{sign } (\bar{u}_{i+1}^{(n+1)} - \bar{u}_i^{(n+1)}) \text{ con } |S| = 1.$$

Para explicar el significado de (13.9.7), supongamos que $\bar{u}_{i+1}^{(n+1)} - \bar{u}_i^{(n+1)} > 0$ o, en forma equivalente, la $\bar{u}_i^{(n+1)}$ es creciente del punto i al punto $i + 1$, por lo que $S = 1$. Entonces la ecuación (13.9.7) toma el valor de

$$\delta f_{i+\frac{1}{2}} = \text{mín } [S(\bar{u}_{i+2}^{(n+1)} - \bar{u}_{i+1}^{(n+1)}), |\delta f_{i+\frac{1}{2}}|, S(\bar{u}_i^{(n+1)} - \bar{u}_{i-1}^{(n+1)})] \tag{13.9.8}$$

o bien

$$\delta f_{i+\frac{1}{2}} = 0 \tag{13.9.9}$$

según lo que sea más grande. Ahora bien, si

$$\bar{u}_{i+2}^{(n+1)} - \bar{u}_{i+1}^{(n+1)} < 0 \tag{13.9.10}$$

o bien

$$\bar{u}_i^{(n+1)} - \bar{u}_{i-1}^{(n+1)} < 0 \tag{13.9.11}$$

lo cual significa que existe una oscilación y (13.9.8) toma el valor más pequeño de éstos, pero entonces la ecuación hace $\delta f_{i+(1/2)} = 0$. En otras palabras, si se cumplen (13.9.10) o (13.9.11), u_i oscila, de manera que se hace $\delta f_{i+(1/2)}$ igual a cero y permanece el flujo calculado mediante el esquema de orden menor.

Ejemplo 13.4

a) Desarrolle un esquema de flujo corregido para

$$u_t(x, t) + au_x(x, t) = 0 \tag{A}$$

donde a es una constante. Utilice el esquema progresivo de primer orden como el esquema de orden bajo y el esquema progresivo de tercer orden (ejemplo 13.3) como el esquema de orden alto.

b) Calcule en forma numérica la solución de (A) con $a = 1$, $\Delta x = 1$, $\Delta t = \gamma = 0.01$, con las condiciones iniciales y en la frontera:

$$u(0, t) = 0$$
$$u(x, 0) = 1 \text{ para } 10\Delta x_i \leqslant x \leqslant 15\Delta x_i$$

pero

$$u(x, 0) = 0 \text{ para } x < 10\Delta x_i \quad \text{y} \quad 15\Delta x_i < x$$

Grafique u_i para $t = 0$, $t = 100\,\Delta t$ y $t = 200\Delta T$. Compare los resultados con los de la figura 13.5, que también son soluciones del mismo problema.

(Solución)

a) El esquema de orden bajo se escribe como

$$\frac{\overline{u}_i^{(n+1)} - u_i^{(n)}}{\Delta t} + \frac{g_{i+\frac{1}{2}} - g_{i-\frac{1}{2}}}{\Delta x} = 0 \tag{B}$$

con

$$g_{i+\frac{1}{2}} = au_i^{(n)}$$

El flujo con base en el esquema progresivo de tercer orden está dado por

$$G_{i+\frac{1}{2}} = a\,\frac{-u_{i+2}^{(n)} + 7u_{i+1}^{(n)} + 7u_i^{(n)} - u_{i-1}^{(n)}}{12\Delta x} + |a|\,\frac{u_{i+2}^{(n)} - 3u_{i+1}^{(n)} + 3u_i^{(n)} - u_{i-1}^{(n)}}{12\Delta x} \tag{C}$$

Si los términos g de (B) se sustituyeran por (C), la ecuación (B) sería el esquema progresivo de tercer orden (véase el ejemplo 13.3).

El esquema de flujo corregido es

$$u_i^{(n+1)} = \overline{u}_i^{(n+1)} - \frac{\Delta t}{\Delta x}\left(\delta f_{i+\frac{1}{2}} - \delta f_{i-\frac{1}{2}}\right)$$

donde

$$\delta f_{i+\frac{1}{2}} = S \text{ máx } \{0, \text{ mín } [S(\overline{u}_{i+2} - \overline{u}_{i+1}), |\delta\hat{f}_{i+\frac{1}{2}}|, S(\overline{u}_i - \overline{u}_{i-1})]\}$$

y

$$\delta\hat{f}_{i+\frac{1}{2}} = G_{i+\frac{1}{2}} - g_{i+\frac{1}{2}}$$
$$S = \begin{cases} 1 & \text{si } \overline{u}_{i+1}^{(n+1)} > \overline{u}_i^{(n+1)} \\ -1 & \text{si } \overline{u}_{i+1}^{(n+1)} < \overline{u}_i^{(n+1)} \end{cases}$$

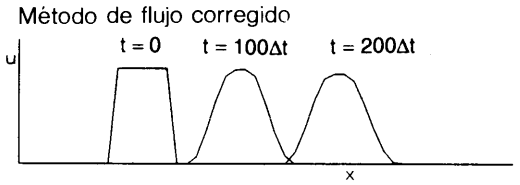


Figura 13.7 Simulación numérica de una onda cuadrada que se mueve a velocidad constante, mediante el método de flujo corregido.

b) Los resultados de los cálculos se muestran en la figura 13.7, la cual se puede comparar con los resultados de los métodos progresivos de primer orden, de Lax-Wendroff y progresivo de tercer orden de la figura 13.6, que también son soluciones del mismo problema. La figura 13.7 muestra que el esquema de flujo corregido simula la onda viajera mejor que el método progresivo de tercer orden. En la figura 13.6, las ondas determinadas mediante el método progresivo de tercer orden oscilan y tienen valores negativos antes y después de la onda, pero esta oscilación no aparece en el método de flujo corregido.

PROBLEMAS

13.1) Obtenga la ecuación modificada para el esquema FTCS de la ecuación (13.3.13).

13.2) Elimine el predictor para mostrar que el esquema de MacCormack dado por (13.8.13) es idéntico al esquema de Lax-Wendroff.

13.3) El esquema de Lax es

$$u_i^{(n+1)} = \frac{1}{2}(u_{i+1}^{(n)} + u_{i-1}^{(n)}) - \frac{\gamma}{2}[u_{i+1}^{(n)} - u_{i-1}^{(n)}]$$

donde $\gamma \equiv a\Delta t/\Delta x$.

a) Muestre que su factor de amplificación es

$$G = \cos(\theta) - j\gamma \operatorname{sen}(\theta), \quad j = \sqrt{-1}$$

b) Grafique G en el plano complejo.

c) Muestre que el esquema es estable si

$$0 < \gamma < 1$$

13.4) Muestre que la ecuación modificada del esquema de Lax del problema (13.3) es

$$u_t + au_x - \frac{a\Delta x}{2}(1/\gamma - \gamma)u_{xx} - \frac{a\Delta x^2}{3}(1 - \gamma^2)u_{xxx} + \cdots = 0$$

13.5) Muestre que la ecuación modificada para el esquema BTCS (Euler hacia atrás con

respecto al tiempo y diferencia central con respecto al espacio) y su factor de amplificación son, respectivamente,

$$u_t + au_x - \frac{1}{2}a^2\Delta t u_{xx} + \left[\frac{1}{6}\Delta x^2 + \frac{1}{3}a^3\Delta t^2\right]u_{xxx} = 0$$

$$G = \frac{1 - j\gamma \operatorname{sen}(\theta)}{1 + \gamma^2 \operatorname{sen}^2(\theta)}$$

13.6) El esquema siguiente recibe el nombre de *esquema de Lax-Wendroff de dos fases*. Muestre que es idéntico al esquema de Lax-Wendroff original para el caso de una EDP lineal hiperbólica como la de la ecuación (13.3.1):

Primera fase

$$\frac{1}{2}\Delta t[2u_{i+(1/2)}^{n+(1/2)} - (u_{i+1}^{(n)} + u_i^{(n)})] + \frac{a}{\Delta x}(u_{i+1}^{(n)} - u_i^{(n)}) = 0$$

Segunda fase

$$\frac{1}{\Delta t}(u_i^{(n+1)} - u_i^{(n)}) + \frac{a}{\Delta x}[u_{i+(1/2)}^{n+(1/2)} - u_{i-(1/2)}^{n+(1/2)}] = 0$$

13.7) Demuestre que si los valores iniciales y en la frontera de u son no negativos, entonces la solución del esquema FTBS es no negativa para $\gamma < 1$.

13.8) Se puede escribir la ecuación en diferencias de

$$u_t + au_x = bu_{xx}, \quad a > 0 \quad \text{y} \quad b > 0$$

mediante el esquema de Euler hacia adelante con respecto al tiempo y la aproximación por diferencias centrales para u_x y u_{xx}. Determine bajo qué condiciones el esquema de diferencias es estable.

13.9) Utilice el análisis de Fourier para determinar la estabilidad de la aproximación por diferencias de

$$u_t + au_x = 0, \quad a > 0$$

dada por

$$\frac{1}{\Delta t}[u_i^{(n+1)} - u_i^{(n)}] + a\frac{3u_i^{(n)} - 4u_{i-1}^{(n)} + u_{i-2}^{(n)}}{2\Delta x} = 0$$

13.10) Obtenga la ecuación modificada para el esquema de diferencias del problema (13.9).

13.11) Muestre que el esquema siguiente es una aproximación por diferencias de tipo conservativo para la ecuación (13.6.1) y que se basa en el esquema progresivo de tercer orden:

$$u_i^{(n+1)} = u_i^{(n)} - k(f_{i+(1/2)}^{(n)} - f_{i-(1/2)}^{(n)})$$

donde $k = \Delta t/\Delta x$

$$\begin{aligned} f_{i+\frac{1}{2}} &= \frac{1}{12}[-F_{i+2} + 7F_{i+1} + 7F_i - F_{i-1} \\ &\quad + |a_{i+\frac{1}{2}}|(u_{i+2} - 3u_{i+1} + 3u_i - u_{i-1})] \\ a_{i+\frac{1}{2}} &= \begin{cases} \dfrac{F_{i+1} - F_i}{u_{i+1} - u_i}, & \text{si } u_{i+1} - u_i \neq 0 \\ 0, & \text{si } u_{i+1} - u_i = 0 \end{cases} \end{aligned} \tag{13.8.4}$$

BIBLIOGRAFIA

Anderson, D. A., J. C. Tannehill y R. H. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, Hemisphere, 1984.

Book, D. L., J. P. Boris y S. T. Zalesak, "Flux-Corrected Transport", *Finite-Difference Techniques for Vectorized Fluid Dynamics Calculations*, D. L. Book, editor, Springer-Verlag, 1981.

Courant, R. y D. Hilbert, *Methods of Mathematical Physics*, Wiley-Interscience, 1962.

Ferziger, J. H., *Numerical Methods for Engineering Application*, Wiley-Interscience, 1981.

Garabedian, P. R., *Partial Differential Equations*, Wiley, 1965.

Jameson, A., "Numerical Solution of the Euler Equation for Compressible Inviscid Fluids", *Numerical Methods for Euler Equations of Fluid Dynamics*, SIAM, 1985.

Kawamura, T., "Computation of Turbulent Pipe and Duct Flow Using Third Order Upwind Scheme", AIAA-86-1042, AIAA/ASME 4th Fluid Mechanics, Plasma Dynamics and Lasers Conference, mayo 12-14, 1986/Atlanta, Ga.

Kawamura, T. y K. Kuwahara, "Computation of High Reynolds Number Flow around a Circular Cylinder with Surface Roughness", AIAA 22nd Aerospace Science Meeting, enero 9-12, Reno, Nevada, 1984: AIAA-84-0340.

Leonard, B. P., "Third-order Upwind es a Rational Basis for Computational Fluid Dynamics", en *Computational Techniques and Applications: CTA-83*, Noye & Fletcher, editores, Elsevier, 1984.

Mitchell, A. R. y D. F. Griffiths, *The Finite Difference Methods in Partial Differential Equations*, Wiley-Interscience, 1980.

Nittmann, J., "Doner Cell, FCT-Shasta and Flux Splitting Method: Three Finite Difference Equations Applied to Astrophysical Schock-Cloud Interactions", *Numerical Methods for Fluid Dynamics*, Morton y Baines, editores, Academic Press, 1982.

Oran, E. S. y J. P. Boris, *Numerical Simulation of Reactive Flow*, Elsevier, 1987.

Patankar, S. V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere, 1980.

Roe, P. L., *Upnawind Schemes Using Various Formulations of the Euler Equations, Numerical Methods for Euler Equations of Fluid Dynamics*, SIAM, 1985.

Smith, D. G., *Numerical Solution of Partial Differential Equations*, Oxford University Press, 1978.

von Lavante, E. y W. T. Thoompkins, *An Implicit, Bi-Diagonal Numerical Method for Solving the Navier-Stokes Equations*, AIAA-82-0063.

Yee, H. C., On Symmetric and Upwind TVD Schemes, Proceedings of the Sixth GAMM-Conference on Numerical Methods in Fluid Mechanics, Notes on Numerical Fluid Mechanics, vol. 13, Friedr, Vieweg & Sohn, 1986.

Yee, H. C., R. M. Beam y R. F. Warming, Stable Boundary Approximations for a Class of Implicit Schemes for the One-Dimensional Inviscid Equations of Gas Dynamics, AIAA Computational Dynamics Conference, Palo Alto, CA., junio 22-23, 1981.

Yee, H. C. y J. L. Shinn, Semi-Implicit and Fully Implicit Shock-Capturing Methods for Hyperbolic Conservation Laws with Stiff Source Terms, NASA TM-89415, 1986.

Yee, H. C., R. F. Warming y A. Harten, Application of TVD for the Euler Equations of Gas Dynamics, Lecture in Applied Mathematics, vol. 22, 1985.

APENDICE A

Error de las interpolaciones polinomiales

A.1 ERROR DE UNA INTERPOLACION LINEAL

El error de una interpolación lineal dada por la ecuación (2.2.1) se define como

$$e(x) = f(x) - g(x) \tag{A.1}$$

donde $f(x)$ es la función exacta. Puesto que el error se anula en $x = a$ y $x = b$, $e(x)$ se puede expresar en la forma

$$e(x) = (x - a)(x - b)s(x) \tag{A.2}$$

donde $s(x)$ es una función.

Fijemos un valor η que satisfaga $a < \eta < b$ y definamos una nueva función como

$$p(x) = f(x) - g(x) - (x - a)(x - b)s(\eta) \tag{A.3}$$

o en forma equivalente

$$p(x) = (x - a)(x - b)[s(x) - s(\eta)] \tag{A.4}$$

es claro que la función $p(x)$ se anula en tres puntos, $x = a$, b, y η como se muestra en la figura A-1. Ahí podemos observar que $p'(x)$ tiene dos raíces, una a la izquierda de η y otra a la derecha de η. Además, $p''(x)$ tiene una raíz (que se denota como ξ) entre las dos raíces de $p'(x)$.

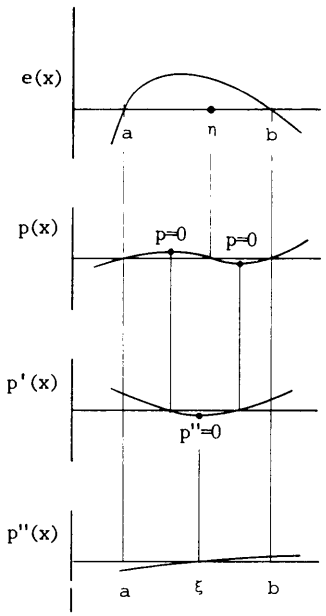


Figura A-1 Raíces de $p(x)$, $p'(x)$ y $p''(x)$

Al derivar la ecuación (A.3) dos veces, se obtiene

$$p''(x) = f''(x) - 0 - 2s(\eta) \tag{A.5}$$

donde $g''(x)$ se anula, puesto que $g(x)$ es una función lineal por definición. En $x = \xi$, que es la raíz de $p''(x) = 0$, la ecuación (A.5) se transforma en

$$0 = f''(\xi) - 2s(\eta)$$

o en forma equivalente

$$s(\eta) = \frac{1}{2}f''(\xi) \tag{A.7}$$

Esta ecuación indica que $s(\eta)$ (para un valor dado de η que satisfaga $a < \eta < b$) está dada por $\frac{1}{2}f''(\xi)$, donde ξ cumple $a < \xi < b$. Así, al cambiar el símbolo η a x, podemos escribir

$$s(x) = \frac{1}{2}f''(\xi), \quad a < \xi < b, \quad a < x < b$$

donde ξ depende de x pero siempre está en $[a, b]$. Así, el error expresado en la ecuación (A.2) se transforma en

$$e(x) = \frac{1}{2}(x - a)(x - b)f''(\xi), \quad a < \xi < b, \quad a < x < b \tag{A.8}$$

Si se supone que el cambio de $f''(x)$ en el intervalo $[a, b]$ es pequeño, $f''(\xi)$ se puede aproximar por $f''(a)$, $f''(b)$ o $f''(x_m)$, donde $x_m = (a + b)/2$. Así, el error se escribe ahora aproximadamente como

$$e(x) \simeq \frac{1}{2}(x - a)(x - b)f''(x_m) \tag{A.9}$$

Esta ecuación indica que el máximo de $|e(x)|$ ocurre aproximadamente en el punto medio de $[a, b]$, y

$$\max_{a<x<b} |e(x)| \simeq \frac{h^2}{8} |f''(x_m)| \tag{A.10}$$

A.2 ERROR DE UNA EXTRAPOLACION LINEAL

La extrapolación lineal es el uso de la interpolación lineal fuera de los dos puntos que se dan como datos. El error de una extrapolación lineal se puede expresar análogamente como una extensión del análisis anterior.

El error de una extrapolación lineal también está dado por la ecuación (A.2). Aquí se muestra que $s(x)$ en la ecuación (A.2) para la extrapolación está dado por

$$s(x) = f''(\xi) \tag{A.11a}$$

donde

$$x \leqslant \xi \leqslant b \quad \text{si } x < a < b \tag{A.11b}$$
$$a \leqslant \xi \leqslant x \quad \text{si } a < b < x \tag{A.11c}$$

Consideremos una extrapolación que se extiende a la izquierda del rango de interpolación. En la ecuación (A.3) —o su equivalente la ecuación (A.4)— fijamos η en un valor que satisface $\eta < a$. Conviene observar que $p(x)$ así definido es una nueva función de x. Se ve que $p(x)$ tiene tres raíces, $x = \eta, a, b$ y que la raíz de $p''(\xi) = 0$ está en $\eta \leqslant \xi \leqslant b$:

$$0 = f''(\xi) = 2s(\eta), \quad \eta < a, \quad \eta \leqslant \xi \leqslant b$$

Mediante un cambio de notación, la ecuación anterior se describe como

$$s(x) = \frac{1}{2}f''(\xi), \quad x < a, \quad x \leqslant \xi \leqslant b \tag{A.12a}$$

De modo semejante, si la extrapolación se extiende hacia la derecha del rango de interpolación, $b < x$, entonces

$$s(x) = \frac{1}{2}f''(\xi), \quad b < x, \quad a \leqslant \xi \leqslant x \tag{A.12b}$$

A.3 ERROR DE UNA INTERPOLACION POLINOMIAL

La expresión del error de una interpolación lineal analizada en la sección anterior se puede extender a las interpolaciones con polinomios de orden superior, incluyendo las interpolaciones de Lagrange y Newton.

Definimos el error de una interpolación polinomial como

$$e(x) = f(x) - g(x) \tag{A.13}$$

donde $f(x)$ es la función exacta de la cual se obtiene como muestra f_i; además, $g(x)$ es un polinomio de orden N. Se supone que $g(x)$ y $f(x)$ no se intersecan en ningún otro punto distinto de $x = x_i$, donde $x_0 \leqslant x \leqslant x_N$. Puesto que $g(x)$ es exacta en los puntos x_i de la retícula, $e(x)$ se puede escribir en la forma

$$e(x) = (x - x_0)(x - x_1) \cdots (x - x_N)s(x) \tag{A.14}$$

Ahora mostraremos que, para un valor x en $[x_0, x_N]$, $s(x)$ está dado por

$$s(x) = \frac{1}{N!} f^{(N+1)}(\xi), \quad x_0 < \xi < x_N \tag{A.15}$$

donde ξ depende de x pero siempre está en $[x_0, x_N]$.

Elegimos un valor fijo η que satisfaga $x_0 < \eta < x_N$ y definimos una nueva función como

$$p(x) = f(x) - g(x) - (x - x_0)(x - x_1) \cdots (x - x_N)s(\eta) \tag{A.16}$$

o en forma equivalente

$$p(x) = (x - x_0)(x - x_1) \cdots (x - x_N)[s(x) - s(\eta)] \tag{A.17}$$

Es claro que la función $p(x)$ se anula en los $N + 1$ puntos x_i de la retícula, $i = 0, 1, 2, \ldots, N$ y también en $x = \eta$. No tiene otras raíces en $[x_0, x_N]$. Por medio de un argumento similar al de la sección anterior, podemos decir que todas las raíces de $p'(x)$ están entre las dos raíces extremas de $p(x)$ y que todas las raíces de $p''(x)$ están entre las dos raíces extremas de $p'(x)$, y así sucesivamente. Así, la raíz de $p^{(N+1)}$ también debe estar entre x_0 y x_N. La derivada $(N + 1)$-ésima de la ecuación (A.17) es

$$p^{(N+1)}(x) = f^{(N+1)}(x) - 0 - s(\eta)N! \tag{A.18}$$

que se transforma en

$$p^{(N+1)}(\xi) = f^{(N+1)}(\xi) - s(\eta)N! = 0 \tag{A.19}$$

donde ξ denota la raíz de $p^{(N+1)} = 0$ que satisface $x_0 < \xi < x_N$. Así,

$$s(\eta) = \frac{1}{N!} f^{(N+1)}(\xi) \tag{A.20}$$

donde tanto η como ξ cumplen que $a < \eta < b$ y $a < \xi < b$, respectivamente.

Mediante un cambio de notación de η y ξ a x y ξ, respectivamente, obtenemos

$$s(x) = \frac{1}{N!} f^{(N+1)}(\xi) \tag{A.21}$$

Así, el error de una interpolación de orden N se expresa como

$$e(x) = \frac{1}{N!}(x - x_0)(x - x_N) \ldots (x - x_N) f^{(N+1)}(\xi), \quad x_0 < \xi < x_N \tag{A.22}$$

Si $g(x)$ se usa como una extrapolación para $x < a$ o $x > b$, el error está dado por la ecuación (A.22), excepto que ξ es un valor que satisface $x < \xi < b$, o $a < \xi < x$ respectivamente.

APENDICE B

Polinomios de Legendre

Los polinomios de Legendre, en los que se basan las cuadraturas de Gauss-Legendre, se escriben como

$$P_0(x) = 1$$

$$P_1(x) = x$$

$$P_2(x) = \frac{1}{2}[3x^2 - 1]$$

$$P_3(x) = \frac{1}{2}[5x^3 - 3x]$$

$$P_4(x) = \frac{1}{8}[35x^4 - 30x^2 + 3]$$

$$\vdots$$

Cualquier polinomio de Legendre de grado superior puede obtenerse utilizando la fórmula de recursión

$$nP_n(x) - (2n - 1)xP_{n-1}(x) + (n - 1)P_{n-2}(x) = 0$$

De manera alternativa, un polinomio de Legendre de orden n también se puede expresar como

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n(x^2 - 1)^n}{dx^n}$$

Una propiedad importante de los polinomios de Legendre es la ortogonalidad:

$$\int_{-1}^{1} P_m(x)P_n(x)\,dx = 0, \qquad \text{para } n \neq m$$

$$= \frac{2}{2n+1}, \quad \text{para } m = n$$

La ecuación anterior indica que la integral en $[-1, 1]$ de dos polinomios de Legendre distintos es igual a cero.

Todo polinomio de orden menor o igual que $N - 1$ es ortogonal al polinomio de Legendre de orden N. Esto se puede demostrar fácilmente, ya que un polinomio de orden menor o igual que $N - 1$ se puede expresar como una combinación lineal de polinomios de Legendre de orden a lo más $N - 1$.

APENDICE C

Cálculo de diferencias de orden superior con el operador de traslación

El operador de traslación se define como

$$Ef_i = f_{i+1} \tag{C.1}$$

Aquí, el operador E desplaza el índice de f_i por uno en la dirección positiva. Su inversa, E^{-1}, desplaza el índice en la dirección negativa, es decir

$$E^{-1}f_i = f_{i-1} \tag{C.2}$$

Una aplicación múltiple de E da por resultado

$$E^n f_i = f_{i+n} \tag{C.3}$$

donde n puede ser cero o cualquier entero positivo o negativo.

Con E, los operadores de diferencia hacia atrás y hacia adelante se pueden escribir respectivamente como

$$\Delta = E - 1 \tag{C.4}$$

$$\nabla = 1 - E^{-1} \tag{C.5}$$

Por lo tanto, la diferencia hacia adelante de orden n se puede obtener como

$$\begin{aligned}\Delta^n f_i &= (E-1)^n f_i \\ &= \left[E^n - nE^{n-1} + \binom{n}{2}E^{n-2} - \binom{n}{3}E^{n-3} + \cdots + (-1)^n\binom{n}{n}E^0\right]f_i \\ &= f_{i+n} - nf_{i+n-1} + \frac{1}{2}n(n-1)f_{i+n-2} - \frac{1}{6}n(n-1)(n-2)f_{i+n-3} \\ &\quad + \cdots + (-1)^{n-1}nf_{i+1} + (-1)^n f_i \qquad \text{(C.6)}\end{aligned}$$

donde $\binom{n}{k}$ es un coeficiente binomial igual a $n!/(n-k)!k!$

Análogamente, se obtiene la diferencia hacia atrás de orden n como

$$\begin{aligned}\nabla^n f_i &= (1-E^{-1})^n f_i \\ &= \left[1 - nE^{-1} + \binom{n}{2}E^{-2} - \binom{n}{3}E^{-3} + \cdots + (-1)^n\binom{n}{n}E^{-n}\right]f_i \\ &= f_i - nf_{i-1} + \frac{1}{2}n(n-1)f_{i-2} - \cdots + (-1)^{n-1}nf_{i-n+1} + (-1)^n f_{i-n} \qquad \text{(C.7)}\end{aligned}$$

APENDICE D

Obtención de EDP hiperbólicas de dimensión uno para problemas de flujo

Supongamos que un fluido incompresible fluye en un tubo perfectamente aislado cuyas secciones tienen área variable $A(x)$ y que la temperatura del fluido cambia en la dirección del flujo. Ignoramos la conducción de calor en la dirección del eje del tubo y también suponemos que la temperatura es constante en el plano perpendicular a la dirección del eje. Entonces, el calor en un elemento de volumen localizado en x, tal como se muestra en la figura A2, es como sigue:

$$c_p \rho A(x)\, dx\, dT = [c_p \rho A(x) v(x) T(x, t) - c_p \rho A(x + dx) v(x + dx) T(x + dx, t)]\, dt \quad \text{(D.1)}$$

donde c_p es el calor específico y ρ es la densidad del fluido. El lado izquierdo es la tasa de incremento de la energía interna durante dt, el primer término del lado derecho es el flujo de calor que entra por la frontera izquierda del elemento de volumen durante dt y el segundo término es el flujo de calor que sale por la frontera derecha durante dt. Si suponemos que tanto c_p y ρ son constantes, al dividir la ecuación (D.1) entre dt se obtiene

$$A(x)\, dx \frac{\partial T}{\partial t} = A(x) v(x) T(x, t) - A(x + dx) v(x + dx) T(x + dx, t) \quad \text{(D.2)}$$

Al dividir entre dx y expresar al lado derecho en forma de derivada obtenemos

$$A(x) \frac{\partial T}{\partial t} = -\frac{\partial}{\partial x} [A(x) v(x) T(x)] \quad \text{(D.3a)}$$

o, en forma equivalente, T_t

$$T_t + v(x)[T(x)]_x = 0 \tag{D.3b}$$

donde $A(x)$ $v(x)$ es constante. La ecuación (D.3a) está en la forma conservativa, en tanto que la ecuación (D.3b) no lo está. Si existe una fuente de calor, el término correspondiente a ésta se suma en el lado derecho como

$$T_t + v(x)[T(x)]_x = q(x)/c_p\rho \tag{D.4}$$

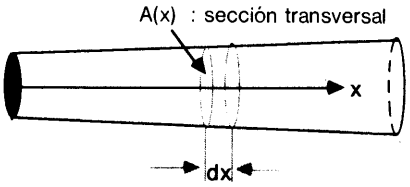


Figura A-2 Un tubo de sección no uniforme

En caso de que el flujo en el tubo sea de una composición química tal que esté distribuida de manera no uniforme, la ecuación para la distribución del compuesto es

$$c_t(x, t) + v(x)[c(x, t)]_x = s(x, t) \tag{D.5}$$

donde $c(x, t)$ es la concentración del compuesto y s es la fuente de volumen del mismo. La ecuación (D.5) está en forma no conservativa.

La conservación de masa de un fluido compresible que fluye en un tubo es

$$A(x)\rho_t(x, t) + [A(x)v(x)\rho(x, t)]_x = 0 \tag{D.6}$$

Si la sección transversal $a(x)$ es constante, se obtienen las siguientes ecuaciones para la temperatura, el compuesto y la densidad del fluido compresible, respectivamente:

$$T_t + [v(x)T(x)]_x = q(x, t)/c_p\rho \tag{D.7}$$

$$c_t(x, t) + [v(x)c(x, t)]_x = s(x, t) \tag{D.8}$$

$$\rho_t(x, t) + [v(x)\rho(x, t)]_x = 0 \tag{D.9}$$

APENDICE E

Disminución de la variación total DVT

La variación total (VT) de una función de onda $u(x)$ se define como

$$VT = \int \left| \frac{\partial u}{\partial x} \right| dx \tag{E.1}$$

[Jameson]. La VT de la solución numérica se define por consiguiente como

$$VT = \sum_{i=-\infty}^{i=+\infty} |u_{i+1} - u_i| \tag{E.2}$$

El análisis de la entropía del fluido muestra que la VT nunca puede crecer. Por lo tanto, deseamos que la VT de la solución del modelo numérico no aumente.

Si la ecuación en diferencias se puede escribir en la forma

$$\frac{d}{dt} u_i(t) = c^+_{i+(1/2)}(u_{i+1} - u_i) - c^-_{i-(1/2)}(u_i - u_{i-1}) \tag{E.3}$$

donde c^- y c^+ son ambos no negativos, entonces el método es un método DVT.

La ecuación (E.2) se puede escribir como

$$VT = \sum_{i=-\infty}^{i=+\infty} s_{i+\frac{1}{2}}(u_{i+1} - u_i) \tag{E.4}$$

donde la suma es sobre i desde $-\infty$ hasta $+\infty$ y

$$s_{i+\frac{1}{2}} = \begin{cases} 1 & \text{si } u_{i+1} - u_i \geqslant 0 \\ -1 & \text{si } u_{i+1} - u_i < 0 \end{cases} \tag{E.5}$$

La derivada de DVT con respecto al tiempo es

$$\begin{aligned} \frac{d}{dt}\,\mathrm{DVT} &= \sum_{i=-\infty}^{i=+\infty} s_{i+(1/2)} \frac{d}{dt}(u_{i+1} - u_i) \\ &= \sum_{i=-\infty}^{i=+\infty} s_{i+(1/2)}[c^{+}_{i+(3/2)}(u_{i+2} - u_{i+1}) - c^{-}_{i+(1/2)}(u_{i+1} - u_i) \\ &\quad - c^{+}_{i+(1/2)}(u_{i+1} - u_i) + c^{-}_{i-(1/2)}(u_i - u_{i-1})] \\ &= \sum_{i=-\infty}^{i=+\infty} v_{i+(1/2)}(u_{i+1} - u_i) \end{aligned} \tag{E.6}$$

donde la ecuación (E.3) se usa y

$$v_{i+(1/2)} = c^{+}_{i+(1/2)}(s_{i+(1/2)} - s_{i-(1/2)}) + c^{-}_{i+(1/2)}(s_{i+(1/2)} - s_{i+(3/2)}) \tag{E.7}$$

El segundo término del lado derecho de la ecuación (E.6) tiene el mismo signo que $u_{i+1} - u_i$ o bien se anula; lo mismo ocurre para el segundo término. Por lo tanto, el término después del signo de suma es no negativo. Así,

$$\frac{d}{dt} VT \leqslant 0 \tag{E.8}$$

Esto termina la demostración de la ecuación (E.3).

BIBLIOGRAFIA

Jameson, A., "Numerical Solution of the Euler Equation for Compressible Inviscid Fluids", *Numerical Methods for Euler Equations of Fluid Dynamics* (F. Angrand, A. Dervieux, J. A. Desideri y R. Glowinski, editores). SIAM, 1985.

APENDICE F

Obtención de las ecuaciones modificadas

Supongamos que una ecuación está dada por

$$u_t + A_2 u_{tt} + A_3 u_{ttt} + \cdots + B_1 u_x + B_2 u_{xx} + B_3 u_{xxx} + B_4 u_{xxxx} + \cdots = 0 \qquad \text{(F.1)}$$

Queremos transformar la ecuación (F.1) en la forma

$$u_t + c_1 u_x + c_2 u_{xx} + c_3 u_{xxx} + c_4 u_{xxxx} + \cdots \qquad \text{(F.2)}$$

eliminando u_{tt}, u_{ttt} y todos los términos con derivadas de orden superior con respecto a t.

Escribamos la ecuación (F.1) como

$$F(x, t) = 0 \qquad \text{(F.1a)}$$

donde F es exactamente igual al lado izquierdo de la ecuación (F.1). Al derivar parcialmente la ecuación (F.1a), podemos escribir

$$F_t = 0 \qquad \text{(F.3a)}$$

$$F_x = 0 \qquad \text{(F.3b)}$$

$$F_{tt} = 0 \qquad \text{(F.3c)}$$

$$F_{tx} = 0 \qquad \text{(F.3d)}$$

$$F_{xx} = 0 \qquad \text{(F.3e)}$$

$$\vdots$$

Es claro que los términos con derivadas de menor orden en la ecuación (F.3a) son u_{tt} y u_{xt}; los términos menores de la ecuación (F.3b) son u_{tx} y u_{xx}; mientras que los términos menores de la ecuación (F.3c) son u_{ttt}, u_{xtt} y así sucesivamente.

Escribimos una combinación lineal de estas ecuaciones como

$$F + P_1F_t + P_2F_x + P_3F_{tt} + P_4F_{tx} + P_5F_{xx} + P_6F_{ttt} + P_7F_{ttx} + P_8F_{txx} + P_9F_{xxx} + \cdots = 0 \tag{F.4}$$

donde $P_1, P_2, \ldots$ son constantes indeterminadas. Entonces debería poderse determinar los coeficientes P_n de forma que todas las derivadas con respecto al tiempo, como u_{tt}, u_{tx}, u_{ttt}, u_{ttx}, u_{txx} etc. se eliminen excepto u_t.

Para implantar este algoritmo, expresamos todas las ecuaciones (F.1a), (F.3a), (F.3b), ... en forma de tabla. En la tabla F.1, la columna más a la izquierda muestra las derivadas de u en orden creciente en t y x. La segunda columna es para los coeficientes de las derivadas en la ecuación (F.1a), $F = 0$. Las columnas restantes son para las ecuaciones (F.3a), (F.3b), (F.3c), etc.

Al igualar a cero los coeficientes de las derivadas no deseadas; a saber, u_{tt}, u_{tx}, u_{ttt}, u_{ttx}, u_{txx}, u_{tttt}, u_{tttx}, u_{ttxx}, $u_{txxx}, \ldots$, en la ecuación (F.4), obtenemos las ecuaciones siguientes:

$$\begin{aligned}
A_2 + P_1 &= 0\\
P_1B_1 + P_2 &= 0\\
A_3 + P_1A_2 + P_3 &= 0\\
P_2A_2 + P_3B_1 + P_4 &= 0\\
P_1B_2 + P_4B_1 + P_5 &= 0\\
A_4 + P_1A_3 + P_3A_2 + P_6 &= 0\\
P_2A_3 + P_4A_2 + P_6B_1 + P_7 &= 0\\
P_3B_2 + P_5A_2 + P_7B_1 + P_8 &= 0\\
P_1B_3 + P_4B_2 + P_8B_1 + P_9 &= 0
\end{aligned} \tag{F.5}$$

Se pueden determinar los coeficientes P_n resolviendo las ecuaciones (F.5) en forma secuencial desde la parte de arriba. Si hacemos $B_1 = c_1 = a$ de acuerdo con la ecuación (13.4.5), las soluciones son como se indica:

$$\begin{aligned}
P_1 &= -A_2\\
P_2 &= aA_2\\
P_3 &= -A_3 + A_2^2\\
P_4 &= a(-2A_2^2 + A_3)\\
P_5 &= A_2B_2 + a^2(2A_2^2 - A_3)\\
P_6 &= -A_4 + 2A_2A_3 - A_2^3\\
P_7 &= a(-4A_2A_3 + 3A_2^3 + A_4)\\
P_8 &= (A_3 - 2A_2^2)B_2 + a^2(-5A_2^3 + 5A_2A_3 - A_4)\\
P_9 &= A_2B_3 + a(4A_2^2 - 2A_3)B_2 + a^3(5A_2^3 - 5A_2A_3 + A_4)
\end{aligned} \tag{F.6}$$

De la tabla F.1, los coeficientes de u_x, u_{xx}, u_{xxx} y u_{xxxx} son:

$$\begin{aligned} c_1 &= B_1 \\ c_2 &= B_2 + P_2 B_1 \\ c_3 &= B_3 + P_2 B_2 + P_5 B_1 \\ c_4 &= B_4 + P_2 B_3 + P_5 B_2 + P_9 B_1 \end{aligned} \tag{F.7}$$

Por medio de la ecuación (F.6), la ecuación (F.7) se transforma en

$$\begin{aligned} c_1 &= B_1 = a \\ c_2 &= B_2 + a^2 A_2 \\ c_3 &= B_3 + 2aA_2B_2 + a^3(2A_2^2 - A_3) \\ c_4 &= B_4 + A_2B_2^2 + 2aA_2B_3 + 6a^2A_2^2B_2 - 3a^2A_3B_2 + a^4(5A_2^3 - 5A_2A_3 + A_4) \end{aligned} \tag{F.8}$$

Tabla F.1 Derivadas de la ecuación modificada

	1 F	P_1 F_t	P_2 F_x	P_3 F_{tt}	P_4 F_{tx}	P_5 F_{xx}	P_6 F_{ttt}	P_7 F_{ttx}	P_8 F_{txx}	P_9 F_{xxx}
u_t	1									
u_x	$B1$									
u_{tt}	$A2$	1								
u_{tx}		$B1$	1							
u_{xx}	$B2$		$B1$							
u_{ttt}	$A3$	$A2$		1						
u_{ttx}			$A2$	$B1$	1					
u_{txx}		$B2$			$B1$	1				
u_{xxx}	$B3$		$B2$			$B1$				
u_{tttt}	$A4$	$A3$		$A2$						
u_{tttx}			$A3$		$A2$		1			
u_{ttxx}				$B2$		$A2$		$B1$	1	
u_{txxx}		$B3$			$B2$				$B1$	1
u_{xxxx}	$B4$		$B3$			$B2$				$B1$
	F	F_t	F_x	F_{tt}	F_{tx}	F_{xx}	F_{ttt}	F_{ttx}	F_{txx}	F_{xxx}

APENDICE G

Interpolación con splines cúbicos

Es frecuente que un número grande de datos tengan que ajustarse a una única curva suave, pero la interpolación de Lagrange o de Newton con polinomios de orden alto no son adecuadas para este propósito, ya que los errores de un único polinomio tienden a crecer en forma drástica al hacer mayor el orden. La interpolación con splines cúbicos está diseñada para adecuarse a este fin.

En la interpolación con splines cúbicos, se usa un polinomio cúbico en cada intervalo entre dos puntos consecutivos. Un polinomio cúbico tiene cuatro coeficientes, por lo que requiere cuatro condiciones. Dos de ellas provienen de la restricción de que el polinomio debe pasar por los puntos en los dos extremos del intervalo. Las otras dos son las condiciones de que la primera y segunda derivadas del polinomio sean continuas en cada uno de los puntos dados.

Los splines cúbicos son polinomios cúbicos por pedazos que, en cierta medida, son análogos a los polinomios cúbicos de Hermite. Aunque la interpolación cúbica de Hermite es más precisa que la interpolación con splines cúbicos en lo que atañe a los valores de la función, esta última es más suave que la primera, debido a que se pide que la función interpolante sea continua en el valor de la función, al igual que la primera y segunda derivadas.

Consideremos un intervalo, $x_i \leqslant x \leqslant x_{i+1}$ de longitud $h_i = x_{i+1} - x_i$ en el rango de interpolación que se muestra en la figura G.1. Por medio de la coordenada local $s = x - x_i$, se puede escribir un polinomio cúbico para un intervalo como

$$g(s) = a + bs + cs^2 + es^3 \tag{G.1}$$

$$x_i \leqslant x \leqslant x_{i+1} \text{ o, en forma equivalente, } 0 \leqslant s \leqslant h_i$$

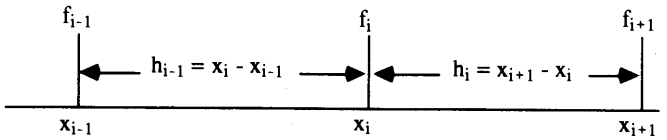


Figura G.1 Notaciones en las interpolaciones con splines cúbicos

Primero pedimos que $g(s)$ sea igual al valor conocido de la función $f(s)$ en $s = 0$ y $s = h_i$, es decir

$$f_i = a \tag{G.2}$$

$$f_{i+1} = a + bh_i + ch_i^2 + eh_i^3 \tag{G.3}$$

donde f_i y f_{i+1} son valores conocidos en $s = 0$ y $s = h_i$, respectivamente. Además, se pide que g' y g'' sean continuas en i e $i + 1$ con el polinomio cúbico de los intervalos adyacentes. Denotamos el valor de g' y g'' en el punto i de la retícula como g'_i y g''_i.

La segunda derivada de la ecuación (G.1), es

$$g''(s) = 2c + 6es \tag{G.4}$$

se iguala con g''_i y g''_{i+1} (que siguen siendo desconocidas) en i e $i + 1$, respectivamente:

$$g''_i = 2c \tag{G.5}$$

$$g''_{i+1} = 2c + 6eh_i \tag{G.6}$$

Al resolver las dos ecuaciones anteriores, c y e se expresan en términos de g''_i y g''_{i+1} como

$$c = \frac{g''_i}{2} \tag{G.7}$$

$$e = \frac{g''_{i+1} - g''_i}{6h_i} \tag{G.8}$$

El coeficiente a ya está dado por la ecuación (G.2). El coeficiente b se determina eliminando a, c y e en la ecuación (G.3) mediante las ecuaciones (G.2), (G.7) y (G.8) y después se resuelve para obtener:

$$b = \frac{f_{i+1} - f_i}{h_i} - \frac{g''_{i+1} + 2g''_i}{6} h_i \tag{G.9}$$

Así, el polinomio cúbico de la ecuación (G.1) se puede escribir como

$$g(s) = f_i + \left[\frac{f_{i+1} - f_i}{h_i} - \frac{g''_{i+1} + 2g''_i}{6} h_i\right] s + \frac{g''_i}{2} s^2 + \frac{g''_{i+1} - g''_i}{6h_i} s^3 \tag{G.10}$$

La primera derivada de la ecuación (G.10) en $s = 0$ y $s = h$ es respectivamente

$$g'_i = -\frac{h_i}{6}[g''_{i+1} + 2g''_i] + \frac{1}{h_i}[f_{i+1} - f_i] \tag{G.11}$$

$$g'_{i+1} = \frac{h_i}{6}[2g''_{i+1} + g''_i] + \frac{1}{h_i}[f_{i+1} - f_i] \tag{G.12}$$

donde $h = x_{i+1} - x_i$. Para otro intervalo de $x_{i-1} < x < x_i$, (G.12) se convierte en

$$g'_i = \frac{h_{i-1}}{6}[2g''_i + g''_{i-1}] + \frac{1}{h_{i-1}}[f_i - f_{i-1}] \tag{G.13}$$

donde $h_{i-1} = x_i - x_{i-1}$. Ya que el término g'_i de la ecuación (G.14) debe ser igual al mismo término pero de la ecuación (G.11) por la continuidad de la primera derivada, al eliminar este término de ambas ecuaciones se obtiene

$$h_{i-1}g''_{i-1} + (2h_{i-1} + 2h_i)g''_i + h_i g''_{i+1} = 6\left[\frac{1}{h_{i-1}}f_{i-1} - \left(\frac{1}{h_{i-1}} + \frac{1}{h_i}\right)f_i + \frac{1}{h_i}f_{i+1}\right] \tag{G.15}$$

Conviene observar que la ecuación (G.15) se aplica a cada punto de la retícula excepto en los extremos. Si prescribimos el valor de g'' en los extremos o lo estimamos mediante la extrapolación de dos puntos interiores de la retícula, entonces coinciden el número de constantes indeterminadas g''_i y el número de ecuaciones. Si suponemos que los puntos dados se denotan por $i = 0, 1, \ldots, N$, el conjunto de ecuaciones es

$$\begin{aligned}
(2h_0 + 2h_1)g''_1 + h_1 g''_2 &= 6\left[\frac{1}{h_0}f_0 - \left(\frac{1}{h_0} + \frac{1}{h_1}\right)f_1 + \frac{1}{h_1}f_2\right] - h_0 g''_0 \\
&\vdots \\
h_{i-1}g''_{i-1} + (2h_{i-1} + 2h_i)g''_i + h_i g''_{i+1} &= 6\left[\frac{1}{h_{i-1}}f_{i-1} - \left(\frac{1}{h_{i-1}} + \frac{1}{h_i}\right)f_i + \frac{1}{h_i}f_{i+1}\right] \\
&\vdots \\
h_{N-1}g''_{N-2} + (2h_{N-2} + 2h_{N-1})g''_{N-1} &= 6\left[\frac{1}{h_{N-2}}f_{N-2} - \left(\frac{1}{h_{N-2}} + \frac{1}{h_{N-1}}\right)f_{N-1} + \frac{1}{h_{N-1}}f_N\right] - h_{N-1}g''_N
\end{aligned} \tag{G.16}$$

Por lo tanto, podemos resolver el conjunto de ecuaciones para determinar las g''_i. El conjunto de ecuaciones es un conjunto tridiagonal, para el cual se describió el algoritmo de solución en la sección 10.3.

Hay tres formas de determinar las condiciones en la frontera:

a) Especificar g'' en la frontera (como ya se ha explicado)
b) Extrapolación desde adentro
c) Condición de frontera cíclica

Si se conoce la segunda derivada de la función en los extremos, se puede usar a). Sin embargo, en la mayoría de los casos esto no ocurre. Una forma de especificar las condiciones en la frontera es la de suponer que $g'' = 0$ en los extremos. Otra forma es b), es decir, extrapolar desde adentro. Al considerar el punto con $i = 0$, la extrapolación para g''_0 se puede escribir como

$$g''_0 = -\frac{h_0}{h_1} g''_2 + \left(1 + \frac{h_0}{h_1}\right) g''_1$$

Al eliminar g''_0 en la ecuación (G.15) con $i = 1$ se obtiene

$$\left(3h_0 + 2h_1 + \frac{h_0^2}{h_1}\right) g''_1 + \left(h_1 - \frac{h_0^2}{h_1}\right) g''_2 = 6\left[\frac{1}{h_0} f_0 - \left(\frac{1}{h_0} + \frac{1}{h_1}\right) f_1 + \frac{1}{h_1} f_2\right] \quad \text{(G.17)}$$

La condición de frontera cíclica se aplica cuando el primer y últimos datos son idénticos y además se pide que las segundas derivadas en estos puntos también sean idénticos. Esto ocurre, por ejemplo, si todo el conjunto de datos representa puntos en un ciclo cerrado de un contorno.

Supongamos, para simplificar la explicación, que existen cuatro intervalos con igual espaciamiento en la retícula. Ilustraremos a continuación las aplicaciones de dos tipos de condiciones en la frontera:

Prescripción de g en los extremos. El sistema de ecuaciones es

$$\begin{aligned} 4g''_1 + g''_2 \qquad &= \frac{6}{h^2}[f_0 - 2f_1 + f_2] - g''_0 \\ g''_1 + 4g''_2 + g''_3 &= \frac{6}{h^2}[f_1 - 2f_2 + f_3] \\ g''_2 + 4g''_3 &= \frac{6}{h^2}[f_2 - 2f_3 + f_4] - g''_4 \end{aligned} \quad \text{(G.18)}$$

donde g''_0 y g''_4 son valores prescritos.

Extrapolación de g en los extremos a partir de puntos internos de la rejilla. g''_0 se puede extrapolar mediante

$$g''_0 = 2g''_1 - g''_2 \tag{G.19}$$

Análogamente, la g'' de la frontera derecha se puede extrapolar mediante

$$g''_N = 2g''_{N-1} - g''_{N-2} \tag{G.20}$$

Por medio de estas extrapolaciones, el conjunto de ecuaciones es

$$\begin{aligned} 6g''_1 &= \frac{6}{h^2}[f_0 - 2f_1 + f_2] \\ g''_1 + 4g''_2 + g''_3 &= \frac{6}{h^2}[f_1 - 2f_2 + f_3] \\ 6g''_3 &= \frac{6}{h^2}[f_2 - 2f_3 + f_4] \end{aligned} \tag{G.21}$$

La solución de las ecuaciones anteriores es casi trivial: g''_1 y g''_3 se obtienen inmediatamente de la primera y tercera ecuación, respectivamente, para después determinar g''_2 mediante la segunda ecuación.

Si el número de puntos en la retícula es grande, el sistema de ecuaciones necesita la eliminación de Gauss (véase el capítulo 6) o el modelo de solución tridiagonal (sección 10.3).

Una de las desventajas de la interpolación con splines cúbicos es que los polinomios de interpolación pueden presentar un comportamiento oscilatorio de los errores. Se ha propuesto el *método del spline de tensión* [Barsky] como una técnica para suprimir la oscilación de la interpolación por medio de splines cúbicos.

Ejemplo G.1

Se tiene la siguiente tabla de datos:

x	$f(x)$
0	0.000000
0.1	0.099833
$\pi/4$	0.707106
$\pi/2$	1.000000
$3\pi/4$	0.707106
π	0.000000
$5\pi/4$	−0.707106
$3\pi/2$	−1.000000
$7\pi/4$	−0.707106
$2\pi - 0.1$	−0.099833
2π	0.000000

Mediante una interpolación con splines cúbicos, estime los valores de $f(x)$ para $x = 0.1, 0.2, \ldots, 1.0$. Los datos anteriores se muestrearon a partir de una función de prueba $f(x) = \text{sen}\,(x)$. Evalúe el error de la estimación de $f(x)$ mediante la interpolación con splines, tomando en cuenta este hecho.

(Solución)

En la tabla dada, no se dan valores frontera para la segunda derivada. Por lo tanto, usamos la extrapolación. En la tabla G.1 se muestran los resultados de la interpolación y la evaluación del error.

Tabla G.1

x	$g(x)$	$f(x)$	error
0.000000	0.000000	0.000000	0.000000
0.698131	0.643082	0.642787	−0.000295
1.396262	0.984166	0.984808	0.000642
2.094393	0.865193	0.866026	0.000833
2.792525	0.341525	0.342022	0.000497
3.490656	−0.341526	−0.342017	−0.000492
4.188787	−0.865193	−0.866024	−0.000831
4.886918	−0.984167	−0.984808	−0.000641
5.585049	−0.643088	−0.642791	0.000297
6.283180	0.000000	−0.000005	−0.000005

$g(x)$: interpolación con splines
$f(x)$: función exacta
error $= f - g$

PROGRAMA

PROGRAMA G-1 SPLINE.FOR Interpolación con splines

A) Explicaciones

El programa SPLINE.FOR lleva a cabo la interpolación con splines cúbicos para una tabla de una función dada. Antes de ejecutar el programa, el usuario debe definir la tabla de valores en las instrucciones DATA del programa. A continuación, el programa obtiene un valor aproximado de la función para el valor de x especificado por el usuario, mediante interpolación con splines. Se pide que el usuario especifique el tipo de condiciones en la frontera escogiendo entre: 1) la especificación de g'' en las fronteras, 2) la extrapolación y 3) la cíclica. En el caso de las condiciones de frontera cíclica, el primer y último datos de la definición deben tener el mismo valor.

B) Variables

NI: Número de puntos (en la tabla de datos), menos uno
X(I): valor x del punto i (tabla de datos)

F(I): valor de la función en el punto i (tabla de datos)

h(I): intervalo de la retícula entre los puntos i e $i + 1$

A(I), B(I), C(I) y S(I): coeficientes de las ecuaciones tridiagonales

DD(I): segundas derivadas del spline cúbico

KBC: tipo de condiciones en la frontera

XA(j): valor de x especificado por el usuario para el que se tiene que hallar el valor de la función mediante la interpolación con splines.

JA: número de valores x en XA

C) Listado

```
C----CSL/SPLINE.FOR        Código de interpolación con splines
      DIMENSION X(0:100),F(0:100),A(0:100),B(0:100),C(0:100)
     % ,S(0:100),H(0:100),DD(0:100), XA(99),FA(99)
C          X(i):    valores x de los datos
C          F(i):    Valores funcionales de la tabla de datos
C          NI   :   número de puntos menos uno
C          JAN  :   número de valores x a los que se aplica la interpolación
C          XA(i):   valores x para los que tienen que determinarse los valores funcionales
      DATA NI/10/
      DATA (X(I), I=0,10)
     %     / 0.0, 0.1, 0.78539, 1.57079, 2.35619, 3.14159, 3.92699,
     %       4.71239, 5.4978, 6.18318,  6.28318/
      DATA (F(I), I=0,10)
     %     /0.0, 0.099833,  0.707106, 1.0, 0.707106, 0.0,
     %     -0.707106, -1.0, -0.707106, -0.099833, 0.0/
      DATA JAN/10/
      DATA (XA(I),I=1,10)
     % /0.0, 0.698131, 1.396262, 2.094393, 2.792525, 3.490656,
     %  4.188787, 4.886918, 5.585049, 6.28318/

      PRINT *,' CSL/SPLINE       Código de interpolación con splines '
      PRINT *
      CALL SPL1(NI,X,F,JAN,XA,FA,DD)
      PRINT *, '     I          F(I)            F"(I) ( solución ) '
      DO I=0,NI
        PRINT 34,I,F(I),DD(I)
34      FORMAT(1x,I5,2x, 2F14.7)
      END DO
      PRINT *,'       X      F( interpolado ) '
      DO 44 I=1,JAN
      PRINT 50, XA(I),FA(I)
44    CONTINUE
55    FORMAT(5F12.6)
50    FORMAT( 4F12.6)
      END
C*****************************************************************************
      SUBROUTINE SPL1(NI,X,F,JAN,XA,FA,DD)
      DIMENSION X(0:100),F(0:100),A(0:100),B(0:100),C(0:100)
     % ,S(0:100),H(0:100),DD(0:100),XA(30),FA(30)
C-------------------------- Indique las condiciones en la frontera
      PRINT*,' OPRIMA O PARA ESPECIFICAR LAS SEGUNDAS DERIVADAS EN LOS EXTREMOS ',
     %  ' 0 '
      PRINT *,'       1 PARA EXTRAPOLAR, 0 '
```

```
      PRINT *,'       2 PARA CONDICIONES DE FRONTERA CICLICA '
      READ *, KBC
      IF (KBC.EQ.0) THEN
        PRINT *, ' PROPORCIONE EL VALOR DE LA SEGUNDA DERIVADA PARA EL EXTREMO
        READ *, DD(0)                                                  IZQUIERDO'
        PRINT *, ' PROPORCIONE EL VALOR DE LA SEGUNDA DERIVADA PARA EL EXTREMO
        READ *, DD(NI)                                                  DERECHO'
      END IF
C-----------------------------------------> DETERMINACION DEL SPLINE
      IM=NI-1
      DO  202 I=0,NI-1
         H(I) = X(I+1)-X(I)
202   CONTINUE
      DO 305 I=1,NI-1
         A(I)=H(I-1)
         C(I)=H(I)
         B(I)=2*(A(I)+C(I))
         S(I)= 6*( (F(I-1)-F(I))/H(I-1) + (F(I+1)-F(I))/H(I) )
305   CONTINUE
      IF (KBC.LT.2) THEN
         IF (KBC.EQ.0) THEN
           S(1)=S(1)-A(1)*DD(0)
           S(NI-1)=S(NI-1)-C(NI-1)*DD(NI)
           END IF
         B(1)=B(1)+2*A(1)
         C(1)=C(1)-A(1)
         B(IM)=B(IM)+2*C(IM)
         A(IM)=A(IM)-C(IM)
         CALL TRID(A,B,C,S,DD,NI-1)
         IF (KBC.EQ.1) THEN
            DD(0)=2*DD(1)-DD(2)
            DD(NI)=2*DD(NI-1)-DD(NI-2)
            END IF
       ELSE
         A(NI)=H(NI-1)
         C(NI)=H(0)
         B(NI)=2*(A(NI)+C(NI))
         S(NI)= 6*( (F(NI-1)-F(NI))/H(NI-1) + (F(1)-F(0))/H(0) )
         CALL TRIDCY(A,B,C,S,DD,NI)
         DD(0)=DD(NI)
      END  IF
C ---------------------------------> INTERPOLACION
444   J=0
      DO 400 K=1,JAN
      IF (XA(K) .LT.X(0) .OR. XA(K).GT.X(NI)) GO TO 550
360   IF (XA(K).GE.X(J) .AND. XA(K).LE.X(J+1)) GOTO 380
      IF (J.GT.NI) GOTO 560
      J=J+1
      GOTO 360
380   Z=XA(K)-X(J)
      FA(K) = F(J) +(-(2*DD(J)+DD(J+1))/6*H(J) + (F(J+1)-F(J))/H(J))*Z
     & + (DD(J+1)-DD(J))/6/H(J)*Z**3 + DD(J)*Z**2/2
400   CONTINUE
      RETURN
550   PRINT *,' XA(K) = ',XA(K), ': FUERA DE RANGO', K
      RETURN
560   PRINT *,' J = ',J, ': FUERA DE RANGO'
      END
C*****************************
      SUBROUTINE TRID(A,B,C,S,DD,IM)          ! Solución tridiagonal
      DIMENSION A(0:1),B(0:1),C(0:1),S(0:1),DD(0:1)
```

```
      DO 410 I=2,IM
         R=A(I)/B(I-1)
         B(I)=B(I)-R*C(I-1)
         S(I)=S(I)-R*S(I-1)
410   CONTINUE
      DD(IM)=S(IM)/B(IM)
      DO 540 I=IM-1,1,-1
          DD(I)=(S(I)-C(I)*DD(I+1))/B(I)
540   CONTINUE
      RETURN
      END
C**********************
      SUBROUTINE TRIDCY(A,B,C,S,DD,N)   ! TRIDIAGONAL CON CONDICION DE
                                          FRONTERA CICLICA
      DIMENSION A(0:1), B(0:1),C(0:1),S(0:1),DD(0:1),H(0:100),V(0:100)
      V(1)=A(1)
      H(1)=C(N)
      H(N-1)=A(N)
      H(N)=B(N)
      V(N-1)=C(N-1)
      IM=N-1
      DO I=2,IM
        R=A(I)/B(I-1)
        B(I)=B(I)-R*C(I-1)
        S(I)=S(I)-R*S(I-1)
        V(I)=V(I)-R*V(I-1)
        P=H(I-1)/B(I-1)
        H(I)=H(I)-P*C(I-1)
        H(N)=H(N)-P*V(I-1)
        S(N)=S(N)-P*S(I-1)
      END DO
      T=H(N-1)/B(N-1)
      H(N)=H(N)-T*V(N-1)
      DD(N)=(S(N)-T*S(N-1))/H(N)
      DD(N-1)=(S(N-1)-V(N-1)*DD(N))/B(N-1)
      DO I=N-2,1,-1
         DD(I)=(S(I)-V(I)*DD(N)-C(I)*DD(I+1))/B(I)
      END DO
      RETURN
      END
```

BIBLIOGRAFIA

Barsky, B. A., *Computer Graphics and Geometric Modeling Using Beta-Splines*, Springer-Verlag, 1988.

Rogers, D. F., y J. A. Adams, *Mathematical Elements for Computer Graphics*, McGraw-Hill, 1976.

Gerald, C. F., y P. O. Wheatley, *Applied Numerical Analysis*, 4a. edición, Addison-Wesley, 1989.

donde el lado derecho de cada ecuación es la función analítica que el cliente dio al arquitecto. Conviene señalar que no existe una solución única a este problema, al igual que no existe una única forma de interpolar los datos proporcionados. Sin embargo, hay varias formas posibles de hallar tal función, entre las que se encuentran: 1) resolver una ecuación de Laplace

$$\nabla^2 F(x, y) = 0$$

con las condiciones de frontera, y 2) la interpolación transfinita.

La interpolación transfinita para este problema se puede escribir como

$$F(x, y) = \sum_{m=0}^{1} \phi_m(x) F(x_m, y) + \sum_{n=0}^{1} \psi_n(y) F(x, y_n) - \sum_{m=0}^{1} \sum_{n=0}^{1} \phi_m(x) \psi_n(y) F(x_m, y_n) \tag{H.2}$$

donde

$$\begin{aligned} \phi_0(x) &= \frac{x_1 - x}{x_1 - x_0} \\ \phi_1(x) &= \frac{x_0 - x}{x_0 - x_1} \\ \psi_0(y) &= \frac{y_1 - y}{y_1 - y_0} \\ \psi_1(y) &= \frac{y_0 - y}{y_0 - y_1} \end{aligned} \tag{H.3}$$

La interpolación transfinita así obtenida es una función suave y satisface las condiciones en la frontera. En el análisis más general que se presenta a continuación, se usa la interpolación de Lagrange en vez de la interpolación lineal.

La interpolación transfinita anterior se puede generalizar un poco más para incluir las funciones especificadas a lo largo de líneas múltiples. Consideremos un dominio rectangular dividido por líneas verticales y horizontales, como se muestra en la figura H.1. La línea vertical más a la izquierda es la frontera izquierda y la línea vertical más a la derecha es la frontera derecha. Las líneas verticales se identifican mediante el índice m, donde el índice de la línea vertical más a la izquierda es $m = 0$; mientras que la última tiene índice $m = M$. Análogamente, los índices de las líneas horizontales se indican mediante la letra n, donde $n = 0$ es la frontera inferior y $n = N$ es la frontera superior. Supongamos que se conocen los valores de $F(x, y)$ a lo largo de todas las líneas horizontales y verticales. La función dada a lo largo de la m-ésima línea vertical se denotará por $f_{v,m}(y)$, y la función a lo largo de la n-ésima línea horizontal es $f_{h,n}(x)$. Así, el problema es hallar la función $F(x, y)$ que sea igual a las funciones dadas a lo largo de las líneas verticales y horizontales. Este

APENDICE H

Interpolación transfinita en dos dimensiones

La interpolación transfinita es un método de interpolación para un espacio de dimensión dos en el cual se conocen los valores de la función a lo largo de las fronteras externas, al igual que a lo largo de las líneas verticales y horizontales dentro de las fronteras. Las interpolaciones dobles analizadas en la sección 4.9 se aplican cuando sólo se conocen los valores de la función en las intersecciones de las líneas verticales y horizontales. Por el contrario, la interpolación transfinita se ajusta a funciones continuas especificadas a lo largo de las líneas horizontales y verticales.

Para ilustrar una aplicación de la interpolación transfinita, imaginemos un arquitecto que diseñará un techo curvo en un edificio rectangular, cuya parte superior satisface

$$x_0 \leqslant x \leqslant x_1, \quad y_0 \leqslant y \leqslant y_1$$

el cliente ha especificado la forma de la bóveda a lo largo en las cuatro orillas, las cuales son cuatro funciones analíticas que expresan la altura del techo a lo largo de las orillas. Estas cuatro funciones son continuas en las esquinas por lo que no hay ningún cambio súbito de altura en ninguna de las esquinas del techo. Ahora, el desea crear una superficie curva suave que se ajuste a las alturas en las orillas proporcionadas por el cliente.

Esta pregunta se puede reformular como sigue: determinar una función suave $F(x, y)$ que satisfaga las condiciones en la frontera dadas por

$$\begin{aligned} F(x_0, y) &= f_W(y) \\ F(x_1, y) &= f_E(y) \\ F(x, y_0) &= f_S(x) \\ F(x, y_1) &= f_N(x) \end{aligned} \tag{H.1}$$

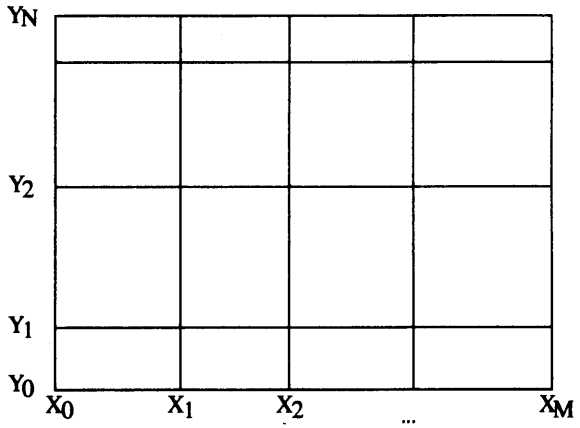


Figura H.1 Dominio rectangular con líneas horizontales y verticales

problema se puede reformular como sigue: hallar una función suave $F(x, y)$ que satisfaga las condiciones dadas por

$$\begin{aligned} F(x, y) &= F(x_m, y) = f_{v,m}(y), \quad \text{a lo largo de } x = x_m \text{ (m-ésima línea vertical)} \\ F(x, y) &= F(x, y_n) = f_{h,n}(x), \quad \text{a lo largo de } y = y_n \text{ (n-ésima línea horizontal)} \end{aligned} \tag{H.4}$$

La interpolación transfinita que satisface las ecuaciones dadas es

$$\begin{aligned} F(x, y) = &\sum_{m=0}^{M} \phi_m(x)F(x_m, y) + \sum_{n=0}^{N} \psi_n(y)F(x, y_n) \\ &- \sum_{m=0}^{M} \sum_{n=0}^{N} \phi_m(x)\psi_n(y)F(x_m, y_n) \end{aligned} \tag{H.5}$$

donde

$$\phi_m(y) = \prod_{k=0, k\neq m}^{M} \frac{x - x_k}{x_m - x_k}$$

$$\psi_n(y) = \prod_{k=0, k\neq n}^{N} \frac{y - y_k}{y_n - y_k}$$

Se puede ver que el primer término de la ecuación (H.5) es la interpolación de Lagrange en la coordenada x de las funciones $F(x_m, y)$, mientras que el segundo término es la interpolación de Lagrange en la coordenada y de las funciones dadas a lo largo de las líneas horizontales. El tercer término es una doble interpolación de Lagrange de los datos dados en las intersecciones de las líneas verticales y horizontales. La interpolación transfinita satisface todas las condiciones en la frontera, lo mismo en las fronteras exteriores como en las interiores.

Aunque hemos supuesto que las funciones a lo largo de las líneas verticales y horizontales son funciones analíticas, esto se puede aplicar a una función definida en forma discreta, como se ilustra a continuación.

Ejemplo H.1

En la siguiente tabla de una función, los valores están dados a lo largo de ciertas columnas y lineas

Tabla H.1 Una tabla de datos dados para $F(I, J)$

j\i	1	2	3	4	5	6	7	8	9	10	11
1	0.2955	0.3894	0.4794	0.5646	0.6442	0.7174	0.7833	0.8415	0.8912	0.9320	0.9636
2	0.4794					0.8415					0.9975
3	0.6442					0.9320					0.9917
4	0.7833	0.8415	0.8912	0.9320	0.9636	0.9854	0.9975	0.9996	0.9917	0.9738	0.9463
5	0.8912					0.9996					0.8632
6	0.9636					0.9738					0.7457
7	0.9975	0.9996	0.9917	0.9738	0.9463	0.9093	0.8632	0.8085	0.7457	0.6755	0.5985

Llene los espacios en blanco con la interpolación transfinita.

(Solución)

La tabla completada con la interpolación transfinita aparece en la tabla H.2. El error de interpolación se evalúa y aparece en la tabla H.3.

Tabla H.2 Resultados de la interpolación transfinita

1	0.2955	0.3894	0.4794	0.5646	0.6442	0.7174	0.7833	0.8415	0.8912	0.9320	0.9636
2	0.4794	0.5647	0.6443	0.7174	0.7834	0.8415	0.8912	0.9320	0.9635	0.9854	0.9975
3	0.6442	0.7174	0.7834	0.8415	0.8912	0.9320	0.9635	0.9854	0.9974	0.9995	0.9917
4	0.7833	0.8415	0.8912	0.9320	0.9636	0.9854	0.9975	0.9996	0.9917	0.9738	0.9463
5	0.8912	0.9320	0.9635	0.9854	0.9975	0.9996	0.9917	0.9739	0.9464	0.9094	0.8632
6	0.9636	0.9854	0.9974	0.9995	0.9916	0.9738	0.9464	0.9094	0.8633	0.8086	0.7457
7	0.9975	0.9996	0.9917	0.9738	0.9463	0.9093	0.8632	0.8085	0.7457	0.6755	0.5985

Tabla H.3 Error de la interpolación transfinita

1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	0.0000	-0.0001	-0.0001	-0.0001	0.0000	0.0000	0.0001	0.0001	0.0001	0.0001	0.0000
3	0.0000	-0.0001	-0.0001	-0.0001	0.0000	0.0000	0.0000	0.0001	0.0001	0.0001	0.0000
4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0001	0.0001	0.0001	0.0000	0.0000	0.0000	-0.0001	-0.0001	-0.0001	0.0000
6	0.0000	0.0001	0.0001	0.0001	0.0001	0.0000	-0.0001	-0.0001	-0.0001	-0.0001	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

PROGRAMA

PROGRAMA H-1 TRANS.FOR Programas de interpolación transfinita

A) Explicaciones

Este programa estima los valores de una tabla para $F(I, J)$ cuando se conocen los valores de ciertas columnas i y los renglones j. Los valores de x correspondientes a cada incremento de i deben tener la misma separación, al igual que los valores de y correspondientes a cada incremento de j.

B) Variables

NI: número máximo de columnas de la tabla (igual a M + 1)

NJ: número máximo de renglones de la tabla (igual a N + 1)

MMX: número de columnas donde se conocen los valores de la función

NNY: número de renglones donde se conocen los valores de la función

IX(m): valor *i* (número de columna) de la *m*-ésima columna donde se conocen los valores de la función

JE(n): valor *j* (número de renglón) del *n*-ésimo renglón donde se conocen los valores de la función

C) Listado

```
C       CSL/TRANSF.FOR    INTERPOLACION TRANSFINITA
        DIMENSION F(20,20), A(20,20)
        DIMENSION IX(6), JE(6), PH(20,100),PS(20,100)
cc        DATA (IX(N), N=1,3)/ 1,6,11/      ! Para correr el problema de prueba,
cc        DATA (JE(M), M=1,3) / 1,4,7/      ! quite cc
cc        DATA MMX, NNY/ 3,3/
cc        DATA NI,NJ/11,7/
        PRINT  *,' CSL/H.FOR   INTERPOLACION TRANSFINITA'
cc        DO J=1,NJ                          ! Inicialización con datos de prueba
cc        DO I=1,NI                          ! Para correr con los datos de prueba, quite cc
cc          Z= 0.1*I+0.2*J                   ! de estas 8 líneas
cc          F(I,J)= SIN(Z)
cc        END DO
cc          write (4,41) j, (f(i,j),i=1,ni)
cc        END DO
41        format( i3, 11f8.4)
cc        IF (1.EQ.1) GOTO 3

        READ (7,*) NI,NJ
        READ (7,*) MMX,NNY
        READ (7,*) (IX(I),I=1,MMX)
        READ (7,*) (JE(J),J=1,NNY)
        DO J=1,NI
         READ (7,*) (F(I,J),I=1,NI)
        END DO
3       CONTINUE
        DO I=1,NI                    ! Preparación de la función de forma phi
         FI=I
         DO M=1,MMX
          IM=IX(M)
          FIM=IM
          Z=1.0
           DO MD=1,MMX
            IMD=IX(MD)
               IF(IM.NE.IMD)  THEN
                 FIMD=IMD
                 Z=Z*(FI-FIMD)/(FIM-FIMD)
               END IF
           END DO
           PH(M,I)=Z
         END DO
        END DO
```

```
C-----------------------------------------------
      DO J=1,NJ                          ! Preparación de la función de forma psi
       FJ=J
       DO N=1,NNY
        JN=JE(N)
        FJN=JN
        Z=1.0
         DO ND=1,NNY
          JND=JE(ND)
              IF(JN.NE.JND)   THEN
                FJND=JND
                Z=Z*(FJ-FJND)/(FJN-FJND)
              END IF
         END DO
         PS(N,J)=Z
       END DO
      END DO
      DO I=1,NI          ! Inicialización de A(i,j)
        DO J=1,NJ
          A(I,J)=0.0
        END DO
      END DO
C------------------ ! Comienza la interpolación transfinita
      DO J=1,NJ
       DO I=1,NI
        G=0.0
        DO M=1,MMX
            IM=IX(M)
            G = G+ PH(M,I)*F(IM,J)
        END DO
        DO N=1,NNY
            JN=JE(N)
            G = G +PS(N,J)*F(I,JN)
        END DO
          DO M=1,MMX
              IM=IX(M)
              DO N=1,NNY
                  JN=JE(N)
                   G = G - PH(M,I)*PS(N,J)*F(IM,JN )
              END DO
          END DO
          A(I,J)=G
       END DO
      END DO
C                              ! Fin de la interpolación transfinita
       DO J=1,NJ
         PRINT *,(A(I,J),I=1,NI)
         PRINT *
       END DO
10    FORMAT( 5F10.5)
      END
```

BIBLIOGRAFIA

Gordan, W. J. y C. A. Hall, "Construction of Curvilinear Coordinate Systems and Application to Mesh Generation", *International Journal for Numerical Methods in Engineering*, Vol. 7, 461-477, 1973.

Smith, R. E., "Algebraic Grid Generation", *Numerical Grid Generation* (J. Thompson, editor), North-Holland, 1982.

Thompson, J. F., Z. U. A. Wasri y C. W. Masin, "Boundary-Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations—A Review", *J. Comp. Physics*. vol. 47, 1-108, 1982.

Erikson, L., "Practical Three-dimensional Mesh Generation Using Transfinite Interpolation", *SIAM J. Sci. Stat. Comput.*, Vol. 6, 1985.

Respuestas a los problemas capítulos 1-9

Capítulo 1

1.2) a) -1.

b) -32768

1.a) b) $\dfrac{1+e^{-2x}}{2}$

Capítulo 2

2.3)

a) $$g(x) = \frac{(x-0.5)(x-0.75)}{(-0.25)(-0.5)}0.8109 + \frac{(x-0.25)(x-0.75)}{(0.25)(-0.25)}0.6931 + \frac{(x-0.25)(x-0.5)}{(0.50)(0.25)}0.5596$$

b) El error de la fórmula de interpolación de Lagrange dada en el inciso a) es

$$\text{Error} \simeq (x-0.25)(x-0.5)(x-0.75)f'''(0.5)/6 = 0.00023 \text{ para } x = 0.6$$

2.5 b) La ecuación (2.3.9) se transforma en

$$e(x) = \frac{(x-0.0)(x-0.4)(x-0.8)(x-1.2)}{(4)(3)(2)(1)}\exp(0.6)$$

donde $f''(x_M) = \exp(0.6)$. Las estimaciones de los errores para $x = 0.2$, 0.6 y 1.0 son

$$e(0.2) \simeq -0.0018$$
$$e(0.6) \simeq 0.0011$$
$$e(1.0) \simeq -0.0018$$

c) Los errores reales evaluados mediante $e(x) = \exp(x) - g(x)$ son

$$e(0.2) = -0.0017$$
$$e(0.6) = 0.011$$
$$e(1.0) = -0.0020$$

2.16) El polinomio de interpolación hacia adelante de Newton que pasa por $i = 2$, 3 y 4 es

$$g(x) = 0.8109 - 0.1178s - 0.0157s(s-1)/2$$

donde $s = (x - 0.25)/0.25$.

b) El error del anterior polinomio hacia adelante de Newton es el término que se añade si se usa un dato más, por lo que

$$\text{Error} = -0.0049s(s-1)(s-2)/6$$

El valor de s para $x = 0.6$ es $s = (0.6 - 0.25)/0.25 = 1.4$, por lo que

$$\text{Error} = -0.0049(1.4)(1.4-1)(1.4-2)/6 = 0.00027$$

2.27 a) Los cuatro puntos de Chebyshev y sus respectivos valores de la función son:

i	x_i	f_i
1	1.0381	0.0374
2	1.3087	0.2690
3	1.6913	0.5255
4	1.9619	0.6739

b)/c) Las estimaciones de los errores mediante la ecuación (2.3.4) son

$$e(x) \simeq \frac{(x-1.0381)(x-1.3087)(x-1.6913)(x-1.9619)}{4!} f''''(x_m)$$

donde $f''''(x_M) = -6/(.15)^4$. Los errores estimados se calculan y muestran junto con los errores reales evaluados para c):

x	*Error estimado*	*Error real*
1	−3.8E − 4	−5.7E − 4
1.1	3.2E − 4	4.4E − 4
1.2	3.2E − 4	4.2E − 4
1.3	2.9E − 5	3.5E − 5
1.4	−2.6E − 4	−3.0E − 4
1.5	−3.8E − 4	−4.1E − 4
1.6	−2.6E − 4	−2.7E − 4
1.7	2.9E − 4	2.8E − 4
1.8	3.2E − 4	3.0E − 4
1.9	3.2E − 4	2.8E − 4
2.0	−3.9E − 4	−3.2E − 4

Capítulo 3

3.1) $x = 1.7626$ (Valor exacto $= 1.762441$)

3.2) Respuesta final $x = 0.3139$ (valor exacto $= 0.3189289$)

3.5)

a) [0.6, 0.7], [4.7, 4.8]: las raíces son 0.60527, 4.70794

b) [1.6, 1.7]: la raíz es 1.61804

c) [4, 4.1]: la raíz es 4.0

3.6) $x = 0.7697$

3.7)

a) [3.14, 4.71] $x = 4.4283$
[6.28, 7.85] $x = 7.7056$

b) [0, 1] $x = 0.5419$
[1, 2] $x = 1.0765$

c) [1, 2] $x = 1.3248$

d) [0, 0.1] $x = 0$
[0.5, 0.6] $x = 0.5906$
[0.9, 1.0] $x = 0.9511$

3.8)

a) [−5, −4] $x = -4.7368$
[−2, −1] $x = -1.3335$
[0, 1] $x = -0.8530$
[50, 51] $x = 50.1831$

b) [0.3, 0.4] $x = 0.3786$
[3.3, 3.4] $x = 3.3155$

c) [−3.2, −3.1] $x = -3.1038$

3.9) $v = 37.73$ m/seg

3.13)

a) $x = 0.6772$, $x = 1.9068$

b) $x = 0.0$, $x = 0.7469$

c) $x = -0.3714$, $x = 0.6053$, $x = 4.7079$

d) $x = 0.4534$

e) $x = 2$

3.14)

K	*Raíces*	
1	−1.1183	−35.77
2	−2.5917	−25.42
3	−4.894	−17.81

3.15) $x = 3.47614$

3.17) 1.8751, 4.6940, 7.8547

3.18) Primera raíz 3.927
Segunda raíz 7.068
Tercera raíz 10.210

3.19)

b) $f(x) = \text{sen}(x) - 0.3 \exp(x)$
Raíz encontrada: $x = 1.0764$
Raíz encontrada: $x = 0.5419$

c) Estimación inicial: $x_0 = 5.0$
Raíz encontrada: $x = 0.8513$

d) $f(x) = 16x_5 - 20x^3 + 5x$

Estimación inicial x_0	*Raíces halladas*
−5.0	−0.9511
1	0.9511
0	0
0.5	0.5877

3.20)

a) $f(x) = 0.5 \exp(x/3) - \text{sen}(x), \quad x > 0$

Estimación inicial	*x al que converge*
1	0.6772
3	1.9068

b) $f(x) = \log(1 + x) - x^2$

Estimación inicial	*x al que converge*
1	0.7468
0.1	0

c) $f(x) = \exp(x) - 5x^2$

Estimación inicial	*x al que converge*
5	4.7079
3	0.6052
0	−0.3715

d) $f(x) = x^3 + 2x - 1$

Estimación inicial	*x al que converge*
5	0.4533

e) $f(x) = \sqrt{x + 2} - x$

Estimación inicial	*x al que converge*
4	2

3.22) $x = 0.1929$

3.23)

Estimación inicial	*x al que converge*
0.5	0.4717

3.27)

a) $f(x) = 0.5 \exp(x/3) - \operatorname{sen}(x) = 0$
Solución a la que converge: $x = 0.6773$

b) $f(x) = \log(1 + x) - x^2 = 0$
Estimación inicial = 0.1
Solución a la que converge: $x = 0.7469$

3.28 a) Por medio de una estimación de $f = 0.01$ en el lado derecho, consecutivamente

Número de iteraciones	*f*
1	0.05557
2	0.05409
3	0.05411
4	0.05411

b) Con una estimación de $f = 0.01$, la ecuación converge con dos pasos de iteración a $f = 0.01967$.

3.29)

a) $x^2 - 1$ con polinomio reducido $x^2 - 4$

b) $2.386 + 1.9676x + x^2$
El polinomio reducido es $-2.934 + 2x$

c) $0.3733 - 0.325\,x + x^2$
El polinomio reducido es $-8.034 - 4.325\,x - x^2$, el cual también es un factor cuadrático.

d) $2.403 - 2.343\,x + x^2$
El polinomio reducido es $(0.6657 - x)$

e) $0.563 - 2.068\,x + x^2$
Polinomio reducido $= x^2 - 13.931 + 42626$

f) $x^2 - 2x + 1$
Polinomio reducido $= x^4\,4x^3 + 5x^2 - 4\,x + 1$
Al aplicar nuevamente el programa de Bairstow se obtiene

$$x^2 - x + 1$$

con un polinomio reducido $x^2 - 3x + 1$
Así, los tres factores cuadráticos hallados son:

$$(x^2 - 2x + 1),\ (x^2 - x + 1) \text{ y } (x^2 - 3x + 1)$$

3.31)

a) Factores cuadráticos encontrados:

$$x^2 - x + 1,\ x^2 + x + 2$$

Raíces: $(1 \pm \sqrt{3}i)/2,\ (1 \pm \sqrt{7}i)/2$

b) Factor cuadrático encontrado: $x^2 + 2x + 2$
Polinomio reducido: $x + 1$
Raíces: $-1 \pm i,\ -1$

c) Factor cuadrático encontrado: $x^2 + 0.5x - 0.5$
Polinomio reducido: $x - 2.2$
Raíces: $0.5,\ -1,\ 2.2$

d) Factor cuadrático: $x^2 - 0.4x - 1.65$
Polinomio reducido: $x^2 + 1.5x - 7$
Raíces: $1.5,\ -1.1,\ -3.5,\ 2$

e) Factor cuadrático: $x^2 - 2x + 1$
Polinomio reducido: $x^2 - 4x + 4$
Raíces: 1, 1, 2, 2

3.33) $k = 0 \quad s = 0, -3, 1 \pm i$
$k = 1 \quad s = -0.472, -3.065, -0.731 \pm 0.92i$
$k = 10: \ s = -1.679, -3.570, 0.126 \pm 3.64i$

3.34)

K	*Raíces*		
0	1,	−2,	−5
1	0.945,	−1.78,	−5.05
10	0,	−0.55,	−5.45
10.392	−0.260,	−0.27,	−5.46
11	−0.256	±0.68*i*,	−5.48
20	−0.109	±2.62*i*,	−5.78
25	−0.039	±3.18*i*,	−5.92
35	0.085	±4.02*i*,	−6.17
50	0.244	±4.94*i*,	−6.48
100	0.642	±6.91*i*,	−7.28

Capítulo 4

4.1) $3x^3 + 5x - 1 \quad [0, 1]$

$N = 2$	$I = 2.62500$
$N = 4$	$I = 2.53125$
$N = 8$	$I = 2.50781$
$N = 16$	$I = 2.50195$
$N = 32$	$I = 2.50048$

$x^3 - 2x^2 + x + 2 \quad [0, 3]$

$N = 2$	$I = 15.56250$
$N = 4$	$I = 13.45312$
$N = 8$	$I = 12.92578$
$N = 16$	$I = 12.79394$
$N = 32$	$I = 12.76098$

$x^4 + x^3 - x^2 + x + 3 \quad [0, 1]$

$N = 2$	$I = 3.71875$
$N = 4$	$I = 3.64257$
$N = 8$	$I = 3.62316$
$N = 16$	$I = 3.61829$
$N = 32$	$I = 3.61707$

$\tan(x) \quad \left[0, \frac{\pi}{4}\right]$

$N = 2$	$I = .35901$
$N = 4$	$I = .34975$
$N = 8$	$I = .34737$
$N = 16$	$I = .34677$
$N = 32$	$I = .34662$

$e^x \quad [0, 1]$

$N = 2$	$I = 1.75393$
$N = 4$	$I = 1.72722$
$N = 8$	$I = 1.72051$
$N = 16$	$I = 1.71884$
$N = 32$	$I = 1.71842$

$1/(2 + x) \quad [0, 1]$

$N = 2$	$I = .40833$
$N = 4$	$I = .40618$
$N = 8$	$I = .40564$
$N = 16$	$I = .40551$
$N = 32$	$I = .40547$

4.2)

N	I	*Error*
2	0.94805	0.05195
4	0.98711	0.01289
8	0.99687	0.00313
25	0.99966	0.00034
50	0.99991	0.00009
100	0.99997	0.00003

4.3) $h = 0.4 \quad 1.62312$
$h = 0.2 \quad 1.91924$
$h = 0.1 \quad 1.99968$

4.4) $I = 0.1 + \frac{1}{3}(I_{0.1} - I_{0.2})$

$$= 1.99968 + \frac{1}{3}(1.99968 - 1.91924) = 2.02649$$

4.5)

a) Con $h = 0.25$:

$$I_{0.25} = \frac{0.25}{2}[0.9162 + 2(0.8109 + 0.6931 + 0.5596) + 0.4055] = 0.68111$$

Con $h = 0.5$:

$$I_{0.5} = \frac{0.5}{2} 0.9162 + 2(0.6931) + 0.4055] = 0.67697$$

b) $I = I_{0.25} + \frac{1}{3}[I_{0.25} - I_{0.5}] = 0.68249$

4.6)

N	I	*Error, %*
2	1.00228	−0.22
4	1.00013	−0.013
8	1.00001	−0.001
16	1.00000	−0.00005

4.7) $3x^3 + 5x - 1 \quad [0, 1]$

$N = 4$	$I = 2.5000$
$N = 8$	$I = 2.5000$
$N = 16$	$I = 2.5000$

$x^3 - 2x^2 + x + 2 \quad [0, 3]$

$N = 4$	$I = 12.7500$
$N = 8$	$I = 12.7500$
$N = 16$	$I = 12.7500$

$x^4 + x^3 - x^2 + x + 3 \quad [0, 1]$

$N = 4$	$I = 3.61718$
$N = 8$	$I = 3.61669$
$N = 16$	$I = 3.61666$

$\tan(x) \quad \left[0, \frac{\pi}{4}\right]$

$N = 4$	$I = .34667$
$N = 8$	$I = .34657$
$N = 16$	$I = .34657$

$e^x \quad [0, 1]$

$N = 4$	$I = 1.71831$
$N = 8$	$I = 1.71828$
$N = 16$	$I = 1.71828$

$\frac{1}{2 + x} \quad [0, 1]$

$N = 4$	$I = .40547$
$N = 8$	$I = .40546$
$N = 16$	$I = .40546$

4.10)

a) 1.8137

b) 0.6142

c) 1.1107

4.11)

a) 2.0972

b) 1.2912

c) 8.5527

4.12)

N	I
2	11.7809
4	11.7318
8	11.7288

4.13) Longitud = 56.52 m

4.14) Distancia = 291.59 m (Valor exacto = 291.86)

4.16) $N = 2 \quad I = 18.8059013$
$N = 4 \quad I = 21.5262439$
$N = 6 \quad I = 21.5406829$

4.18) $N = 4 \quad I = 0.82224$
$N = 6 \quad I = 0.82246 \quad (I_{\text{exacto}} = 0.82246)$

4.19) 1.34329

4.20) **(a)** 0.9063459, **(b)** 3.104379

4.22) **(a)** 1.8138, **(b)** 0.6142, **(c)** 2.0972, **(d)** 1.2914

4.23) $I = \frac{0.5}{2}[0.93644 + 2(0.85364) + 0.56184]$

$= 0.80139$

4.24) $I = \frac{0.5}{3}[0 + 4(0.4309) + 1.2188] = 0.4904$

4.25)

2 × 2	2.666666
4 × 4	2.976067
8 × 8	3.083595
16 × 16	3.121189
32 × 32	3.134398
64 × 64	3.139052

4.26) $I = 0.36686$

4.27) $I \simeq \frac{0.5}{3}[1.9682 + 4(1.8154) + 1.5784] = 1.8014$

4.28) $I = \left[\left(5 - \frac{1}{2}\right)\right][(0.55555)(1.29395)$
$+ (0.88888)(0.04884)$
$+ (0.55555)(0.63524)]$
$= 2.23036$

Capítulo 5

5.13)
$$\begin{aligned} 27a_3 + 8a_2 + a_1 + 0 &= 0 \\ 9a_3 + 4a_2 + a_1 + 0 &= 1 \\ 3a_3 + 2a_2 + a_1 + 0 &= 0 \\ a_3 + a_2 + a_1 + a_0 &= 0 \end{aligned}$$

5.14)

a) $[f(0.2) + 3f(0) - 4f(-0.1)]/(6*0.1)$

b) $0(h^3) = -\frac{1}{3}h^2 f'''$

c) $(4.441 + 3 * 4.020 - 4 * 4.157)/0.6 = -0.2117$

5.15) $f_i'(x) = \dfrac{-f_{i+3} + 9f_{i+1} - 8f_i}{6h} + 0(h^2)$

$0(h^2) = \frac{1}{2}h^2 f_i'''$

5.18) $m = -4$

5.28)
$$\begin{aligned} g'(x_i) &= \frac{1}{h}\left[\Delta f_i + \frac{1}{2}(-1)\Delta^2 f_i + \frac{1}{6}(2)\Delta^3 f_i\right] \\ &= \frac{1}{h}\left[f_{i+1} - f_i - \frac{1}{2}(f_{i+2} - 2f_{i+1} + f_i)\right. \\ &\quad \left. + \frac{1}{3}(f_{i+3} - 3f_{i+2} + 3f_{i+1} - f_i)\right] \\ &= \frac{2f_{i+3} - 9f_{i+2} + 18f_{i+1} - 11f_i}{6h} \end{aligned}$$

$$\text{error} = -\frac{1}{4}h^3 f''''$$

$$\begin{aligned} g''(x_i) &= \frac{1}{h^2}(\Delta^2 f_i - \Delta^3 f_i) \\ &= \frac{1}{h^2}[(f_{i+2} - 2f_{i+1} + f_i) \\ &\quad - (f_{i+3} - 3f_{i+2} + 3f_{i+1} - f_i)] \\ &= \frac{1}{h^2}[-f_{i+3} + 4f_{i+2} - 5f_{i+1} + 2f_i] \end{aligned}$$

$$\text{error} = \frac{11}{12}h^2 f''''$$

5.34)

a)
$$\begin{aligned} \frac{\partial}{\partial y} f(1, 0) &= \frac{-f(1, 1) + 4f(1, 0.5) - 3f(1, 0)}{2h} \\ &= \frac{-0.7002 + 4(0.4767) - 3(0.2412)}{2(0.5)} \\ &= 0.483 \end{aligned}$$

b)
$$\begin{aligned} \frac{\partial^2}{\partial x^2} f &= \frac{f(0.5, 1) - 2f(1, 1) + f(1.5, 1)}{(0.5)^2} \\ &= \frac{0.4547 - 2(0.7002) + 0.9653}{(0.5)^2} = 0.0784 \end{aligned}$$

c)
$$\begin{aligned} \frac{\partial^2 f}{\partial x\, \partial y} = \frac{\partial}{\partial y} f_x &\simeq \frac{-f_x(0, 1) + 4f_x(0, 0.5) - 3f_x(0, 0)}{2h} \\ &= \frac{-0.4481 + 4(0.3065) - 3(0.1555)}{2(0.5)} = 0.3114 \end{aligned}$$

Capítulo 6

6.1)

a) $x_1 = 1, x_2 = 3, x_3 = 2$

b) $x_1 = 1, x_2 = 66, x_3 = 23$

6.2)

a) $x = 2.3829, y = 1.4893, z = 2.0212$

b) $x = 11, y = 11, z = 10$

6.6) $\begin{bmatrix} 1 & 0 & 0 & 0.25 & 2.00 & 0.25 \\ 0 & 1 & 0 & -0.3125 & -3.25 & 1.6875 \\ 0 & 0 & 1 & 1.0625 & -0.75 & 0.0625 \end{bmatrix}$

a) $x_1 = 0.25, x_2 = -0.3125, x_3 = 1.0625$

b) $x_1 = 2.00, x_2 = -3.25, x_3 = -0.75$

c) $x_1 = 0.25, x_2 = -1.6875, x_3 = 0.0625$

6.10)

$$F = AB = \begin{bmatrix} 13 & 9 & 10 \\ 9 & 8 & 9 \\ 21 & 8 & 10 \end{bmatrix}$$

$$H = BC = \begin{bmatrix} 37 \\ 27 \\ 9 \\ 22 \end{bmatrix}$$

6.11) $E = \begin{bmatrix} 14 & 6 & 15 \\ -3 & 5 & 3 \\ 4 & 7 & 20 \end{bmatrix}$

6.12) La inversa es $A^{-1} = \begin{bmatrix} 0.16129, & -0.032258 \\ -0.12903, & 0.22581 \end{bmatrix}$

6.13)

a) $A^{-1} = \begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 3 & 2 & 1 \\ 2 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$, $\det(A) = 1$

b) $A^{-1} = \begin{bmatrix} -0.04 & 0.04 & 0.12 \\ 0.56 & -1.56 & 0.32 \\ -0.24 & 1.24 & -0.28 \end{bmatrix}$, $\det(B) =$

6.14) $\begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} & -\frac{3}{2} \\ \frac{1}{2} & -\frac{3}{2} & \frac{5}{2} \end{bmatrix}$

6.16) En forma compacta:

a) LU: $\begin{matrix} 2 & -0.5 & 0 \\ -1 & 1.5 & -0.6666 \\ 0 & -1 & 1.3333 \end{matrix}$

b) LU: $\begin{matrix} -3 & -1.3333 & 0.3333 \\ 1 & 0.3333 & 5 \\ 2 & 1.6666 & -8 \end{matrix}$

con $P = (2, 3, 1)$

6.18) $\det(A) = -10$
$\det(B) = 7$
$\det(C) = 51$
$\det(D) = -199$

6.19) $\text{Det}(A) = \det(L)\det(U)$
$= (8)(8.75)(2.2)(4.8052) = 740.00$

6.20) $\det(A^{-1}) = 1/\det(A)$, y $\det(A)$
$= \det(B)\det(C)\det(D)$

$\det(B) = 72, \quad \det(C) = -4, \quad \det(D) = 7$

Por lo que $\det(A) = (72)(-4)(7) = -2016$

y $\det(A^{-1}) = 1/\det(A) = -4.96 \times 10^{-4}$

6.21) $\det(A^t) = \det\begin{bmatrix} 2 & 4 \\ 1 & 2 \end{bmatrix} = 4 - 4 = 0$

$\det(B^t) = \det\begin{bmatrix} 3 & 1 \\ 2 & -1 \end{bmatrix} = -3 - 2 = -5$

6.23) $\det(A) = (2 - s)(-1 - s)(1 - s) + 40$
$+ 12(1 + s) - 4(1 - s)$
$= -2s^3 + 2s^2 17s + 46$

Capítulo 7

7.1) $f(\lambda) = 8 - 5\lambda + \lambda^2$

7.2) $-1 - x + 5x^2 - x^3$

7.4) La serie de potencias que se obtiene es

$$g(x) = -20 + 33x + 8x^2 - 0.99998x^3$$

Por medio del método de Bairstow aplicado al polinomio anterior se obtiene

$$x = 0.5401, -3.407, 10.86$$

7.5) $g(x) = 182.49 + 323.406x$
$+ 28.230x^2 - 22.5x^3 + x^4$

Los valores propios son 2.78, 20.86, $-1.024 \pm 1.602\,i$

7.8) Resultados finales (valores propios de la matriz completa):

3.07980E−01 6.43103E−01 5.04892E+00

7.9) Resultados finales (valores propios de la matriz completa):

2.83250E−01 4.26022E−01
9.99996E−01 8.29086E+00

7.10) Resultados finales (valores propios de la matriz completa):

−3.86748E+00 9.28788E−01
5.40035E+00 1.91383E+01

7.11) Valores propios (resultados del esquema QR):

No.	*Parte real*	*Parte imaginaria*
1	20.867570	+0.000000*i*
2	−1.024574	+1.601827*i*
3	−1.024574	−1.601827*i*
4	2.781592	+0.000000*i*

7.12) Valores propios (resultados del esquema QR):

No.	*Parte real*	*Parte imaginaria*
1	19.138330	+0.000000*i*
2	−3.867481	+0.000000*i*
3	0.928787	+0.000000*i*
4	5.400349	+0.000000*i*

7.13)

a) −3 −4 −1.1
Las raíces son $-1.319 \pm 1.1429i$, -0.3609

b) Las raíces son 2, −1.1196, $-0.9902 \pm 0.05185i$.

Capítulo 8

8.1) Línea de regresión: $g(x) = 0.2200 + 1.90000x$

8.2) Línea de regresión: $g(x) = -10.01212x + 11.0267$

8.4) $g(x) = -3.333861E-02 + 2.632148x - .2488103x^2$

8.5) $g(x) = -1.667578E-02 + 2.576617x - .223822x^2 - 2.776449E-03x^3$

8.6) $g(x) = .1019003 + 240.21x$
$g(x) = -9.956598E-03 + 352.0669x - 13982.13x^2$
$g(x) = -1.672983E-04 + 316.9891x - 1745.694x^2 - 1019702x^3$

8.7) $g(x) = .8796641 + 37.30039x - 39.26325x^2$

8.8) $g(x) = -.1177177 + 60.07453x - 101.6016x^2 + 41.55901x^3$

8.9) $g(x) = -1.857601 + 3.814397x + 3.241863 \operatorname{sen}(\pi x) + 1.09415 \operatorname{sen}(2\pi x)$

Capítulo 9

9.4)

x	*Euler* $h = 0.001$	*Euler modificado* $h = 0.001$
1	0.42467 (0.08)	0.42461 (0.03)
2	−0.30287 (0.45)	−0.30405 (0.03)
5	−21.7175 (0.18)	−21.7634 (0.02)
9	−723.529 (0.28)	−725.657 (0.01)

x	*Euler* $h = 0.01$	*Euler modificado* $h = 0.01$
1	0.42802 (0.87)	0.42741 (0.36)
2	−0.29047 (4.52)	−0.30233 (3.06)
5	−21.3809 (1.72)	−21.8364 (0.18)
9	−705.877 (2.71)	−728.118 (0.18)

9.5) $h = 0.01$

t	y
0.999999	4.8016E−01
1.999998	4.6073E−01
2.999998	4.4169E−01
3.999997	4.2306E−01
5.000019	4.0483E−01
10.000130	3.1970E−01
20.000360	1.7954E−01
30.000590	7.9538E−02
39.999300	1.9685E−02
48.997790	1.5954E−04

9.6)

x	y *(Euler hacia adelante)*	y *(Analítico)*
1.0000	0.0434	0.0432
2.0000	0.0491	0.0491
3.0000	0.0499	0.0499
4.0000	0.0500	0.0500
5.0000	0.0500	0.0500

9.10)

Tiempo (hr)	*N (yodo)*	*N (xenón)*
0.0000	1.0000E+05	0.0000
5.0000	5.9333E+04	3.3339E+04
10.0000	3.5204E+04	4.2660E+04
15.0000	2.0888E+04	4.1012E+04
20.0000	1.2393E+04	3.5108E+04
24.9999	7.3533E+03	2.8225E+04
29.9998	4.3629E+03	2.1821E+04
34.9998	2.5886E+03	1.6430E+04
39.9997	1.5359E+03	1.2138E+04
44.9996	9.1131E+02	8.8418E+03
49.9995	5.4071E+02	6.3716E+03

9.12) $n = 1$: $y_1 = 0.9$, $z_1 = 0.8074$
$n = 2$: $y_2 = 1.6153$

9.14) $y_1 = y(0) + \frac{1}{2}(0.5 + 0.375) = 0.4375$

$$z_1 = z(0) + \frac{1}{2}(-0.25 - 0.2656) = 0.7422$$

$$y_2 = y_1 + \frac{1}{2}(0.3711 + 0.2476) = 0.7468 = y(1)$$

$$z_2 = z_1 + \frac{1}{2}(-0.2471 - 0.2634) = 0.4869 = y'(1)$$

9.16)

Tiempo (min)	y_1	y_2
1.0000	9.2289	1.5296
2.0000	8.5173	2.3305
3.0000	7.8606	2.7028
4.0000	7.2545	2.8260
5.0000	6.6951	2.8072
6.0000	6.1789	2.7104
7.0000	5.7024	2.5733
8.0000	5.2628	2.4181
9.0000	4.8570	2.2576
10.0000	4.4825	2.0991
20.0000	2.0092	0.9537
30.0001	0.9006	0.4276
40.0000	0.4037	0.1917
49.9998	0.1810	0.0859

9.19) $y(1) = y_1 = y(0) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

$$= 1 + \frac{1}{6}[-1 + 2(-0.6666 - 0.7059) - 0.2707] = 0.3307$$

9.20)

a) El error local del modelo de Runge-Kutta de segundo orden es proporcional a h^3, por lo que podemos escribir

$$E_h = Ah^3$$

$$[y(0.2)]_{h=0.1} + 2A(0.1)^3 = [y(0.2)]_{h=0.2} + A(0.2)^3$$

Al introducir la solución calculada a partir de la tabla dada

$$0.894672 + 0.002A = 0.8947514 + 0.008A$$

o después de reescribir,

$$-0.000079 = 0.006\text{ A o bien } A = -0.01316$$

Por lo tanto, el error de $[y(0.2)]_{0.1}$ es igual a

$$2A(0.1)^3 = 2(-0.013)(0.001) = -0.000026$$

b) El error de y(1) con $h = 0.1$ es

$$[y(1)]_{h=0.1} + 10A(0.1)^3 = [y(1)]_{h=0.2} + 5A(0.2)^3$$

$$0.3226759 + 10A(0.001) = 0.3240404 + 5A(0.2)^3$$

$$-0.0013645 = 0.04A - 0.01A = 0.03A$$

$$A = -0.04548$$

Por lo que $10A(0.01)^3 = 0.01(-0.04548) = -0.0004548$

La estimación del valor exacto es

$$0.3226759 + 10A(0.001) = 0.3226759 - 0.0004548 = 0.3222211$$

El verdadero valor es 0.32219.

9.21)

t	$h = 0.5$	$h = 1.0$
0	1	1
1	0.32233	0.32388
2	−0.59577	−0.59636

Indice

A

aeronave modelo NACA 0012, 104
agrupamiento, 14
ajuste de curvas, 274
ajuste de polinomios, 282
analítico, 2
aplicación, 45 (véase *también* transformación de coordenadas)
autoiniciable, 315
automóvil, 15

B

base, 5
Beam-Warming, esquema de, 515
binario, 5
bit, 6-7
bóveda, 549
byte, 7-8

C

cable flexible, 399
característica(o)s
  curva (lineal), 492
  ecuación, 62, 104-05
  valor, 240
ceros, 62
cicloide, 152
coeficiente:
  binomial, 33, 34, 532
  indeterminado, 164
  principal, 41
Colebrook, correlación de, 106-107
compleción, 376
compleja(s)
  raíces conjugadas, 83, 243
  raíz, 74, 98
complemento a dos, 8-9
condición:
  adiabática en la frontera, 355
  en la frontera, 351, 356, 409, 414, 472
  en la frontera de Dirichlet, 409
  en la frontera de tipo mixto, 356
  inicial, 289, 472
conducción de calor, 326, 470
conductividad térmica, 351, 399, 401
conjunto de EDO, 338
conmutar, 198
conservación (conservativo), 353, 363, 508
control
  varilla de, 465, 466
  volumen de, 327

convección, 351
convergencia, 375
razón de, 435, 443
coordenadas:
cilíndricas, 352
r-z, 423
corrector de Adams-Moulton, 314, 316
Courant-Issacson-Rees, esquema de, 513-517
Crank-Nicolson, 471, 506
cuadrático(a)
factor, 83
interpolación, 115
cuerpo de revolución, 111
Chebyshev
método semi-iterativo, 432
polinomio de, 128
punto de, 23
raíz de, 43

D

decimal, 5
defectos, 62, 65
deflección de una varilla, 400
depósito:
cónico, 343
de agua, 345
desborde (*overflow*), 133
descomposición LU, 184, 207, 381, 450
determinante, 212, 227, 239, 258
diagonal
coeficiente, 192
dominante, 425, 426
diferencia
central, 158, 354, 410
operador, 166
de cocientes, 83
dividida, 42, 55
progresiva, 511
de primer orden, 496, 520
de tercer orden, 506, 510, 512, 519, 522-23
diferencias
aproximación por, 29, 74, 409
ecuación en, 380
operador de, 166
tabla de, 3
dirección alternante, método implícito de la (ADI), 471, 447, 484, 487
disminución de la variación total (DVT), 517, 535
división entre cero, 1
doble
integración, 109, 135
interpolación, 50, 549
precisión, 14, 194, 223, 279
transformación exponencial, 110, 130, 132, 146
dominante
función propia, 443
valor propio, 443

E

ecuación
homogénea, 238
rígida, 290, 329
transcendente, 62
ecuaciones diferenciales ordinarias (EDO), 289, 351
de orden superior, 291, 294, 302
EDP
elíptica, 407
hiperbólica, 407, 489, 533
parabólica, 407, 470, 489, 533
efecto de antidifusión, 507
EJ (*véase* método de Jacobi extrapolado), 428
enfoque gráfico, 67, 106-07
épsilon de la máquina, 6-7, 11-12, 223
error, 28, 36, 113, 116
aleatorio, 274
de perturbación, 503, 516
de redondeo, 1, 11-13, 34, 43, 78
de truncamiento, 1, 14, 309, 501
extrapolación lineal, 526, 524
interpolación polinomial, 527
local, 318
regla del trapecio, 112
esquema
de flujo corregido, 513, 517
de Lax-Wendroff de dos pasos, 521

de orden bajo, 517
iterativo, 75, 250, 253, 426
estabilidad
criterio para EDP, 495
del método de Runge-Kutta, 309
EDP, 478
predictor-corrector, 318
estado crítico, 82
estrictamente diagonal dominante, 426
estructura de banda, 450
explícito
método, 473
método característico, 495
exponencial, 290
método, 331
transformación, 132, 290
extendida
regla de 1/3 de Simpson, 400
regla de trapecio, 146, 400
extrapolación, 45, 51-52, 77, 543
parámetro de, 448
extremo fijo, 69

F

factor de amplitud, 480, 483, 490
factorización aproximada, 448, 484
fluido incompresible, 534
forma no conservativa, 509
Fourier
expansión (serie) de, 22, 478
Frobenius, matriz de, 256
frontera
cíclica, 543
curva, 417
función propia, 434
método de la, 478

G

Gauss-Jordan eliminación de, 184, 188, 203, 219-220
Gauss-Seidel, método de, 428, 465, 466
gaussiana
cuadratura, 110, 123, 137
eliminación, 207, 279, 381, 427, 453
geometría cilíndrica, 401
Gram-Schmit, algoritmo de, 254
Green, teorema de, 420

H

hacia adelante
aproximación por diferencias, 74, 513
eliminación, 185
en el tiempo y hacia atrás en el espacio (FTCS), 497, 512
Euler, 290-292, 471, 499
hacia atrás
diferencia, 74
operador, 166, 513
tabla, 38
eliminación, 188
en el tiempo y central en el espacio (BTCS), 499
método de Euler, 298, 471, 499
sustitución, 185
Hermite
interpolación de, 23, 47
polinomio de, 128
Hessenberg, 261, 324
hexadecimal, 5
Hilbert, matriz de, 217, 237
Housedolder, 242, 261

I

IBM-370, 432
IBM PC, 133, 432
inestabilidad, 293, 309, 318, 320, 474
integro-diferencial, 291, 322
interpolación
de Lagrange, 23, 127-28, 551
transfinita, 549
intervalo entre las marcas, 92
inversa
matriz, 202
método de la potencia, 242, 252, 372, 386
iteración
de punto fijo, 79
QR, 83, 86, 242, 250, 253

L

Laplace, ecuación de, 408, 550
Lax-Friedrich, esquema de, 513

Lax-Wendroff, esquema de, 505, 512-514
Legendre
polinomio de, 529
punto de, 123
lineal
álgebra, 184
combinación, 280
ecuación, 25
interpolación, 22
método de relajación, 428
regresión, 274
linealmente
dependientes, 195
independientes, 218
longitud de arco, 103-04

M

MacCormack, esquema de, 505, 515
Maclaurin, 3
mantisa, 8-9
Markov, coeficiente de, 31, 244, 258
matrices L y U, 395
matricial
método, 478
notación, 184
matriz, 196, 242
aumentada, 203
identidad, 201
inversión de una, 225
no simétrica, 242
nxm, 219
simétrica, 242
triangular inferior, 22, 207
triangular superior, 22, 207
Matriz o vector nulo, 201
matriz-s, 426
membrana, 402
método
características, 491
de Baistow, 63, 82, 243
de bisección, 63, 246
de Cholesky, 452
de Euler, 292
de integración, 419
de interpolación, 243
de Jacobi extrapolado (EJ), 428, 431, 456
de la falsa posición, 68
de la potencia inversa con desplazamiento o de Wielandt, 242, 250-253, 373, 382, 389
de la secante, 63, 77
de líneas, 329
de los mínimos cuadrados, 274
de potencias, 250
de SOR rojo-negro, 432
del disparo, 327, 353, 354
implícito, 290, 330, 473
iterativo de Jacobi, 428
valor propio del, 439
predictor-corrector de Adams, 314
predictor-corrector de Euler, 300
Milne, método predictor-corrector de, 349
modificado (a)
ecuación, 478, 501, 507, 537
método de Euler, 290, 295
método de la falsa posición, 68
modo de alta frecuencia, 479
monóxido de carbono, 105-06
Moody, tabla de, 106-07
multiplicidad, 273

N

natural
convección, 322
frecuencia, 348
Neumann, condición en la frontera de, 409
neutrones
difusión de, 470
flujo de, 352, 369
Newton-Cotes, integración de, 51-52, 109, 110, 120, 143
Newton
interpolación de, 23, 25, 32, 120, 271
hacia adelante, 25, 32, 168
hacia atrás, 25, 38, 315-316
método de, 63, 873, 367, 403
normalización, 387
numérico(a)
amortiguamiento, 504
de cuarto orden, 504
de segundo orden, 504

diferenciación, 22
difusión, 516
flujo, 510, 515
integración, 22
viscosidad, 498, 510, 516

O

octal, 5
operador de traslación, 531
optimización
del parámetro de iteración, 442
del parámetro de SOR, 438, 457
del parámetro EJ, 459
ortogonal, 127-128, 376, 530
polinomio, 110
oscilación armónica (vibración), 241, 369

P

par conjugado, 243
permutación, 213, 214
pieza metálica, 347-48
pivote, 186, 220
pivoteo, 191, 222-223
Poisson, 408
polinomio
de Laguerre, 128
osculatriz, 47
reducido, 86
positiva
definida, 242, 371
matriz, 363
precisión
doble, 14, 194, 223, 279
simple, 11-12, 194
predictor-corrector, 51-52, 290, 312, 340,
predictor de Adams-Bashfort, 316, 508
problema
con valores en la frontera, 351
irresoluble, 195
mal condicionado, 185, 216, 279, 374
no lineal con valores en la frontera, 353
propiedad bicíclica, 429
proyectil, 78, 103-04, 345
prueba y error, 389, 395
punto
flotante normalizado, 9
hipotético, 381

Q

químico(a)
equilibrio, 105-06
sustancia, 534

R

radio
dominante, 377, 405
efectivo, 378
espectral, 437, 453, 467
raíces
búsqueda de, 90
cálculo de, 62
raíz, 5
rápida
solución directa (FDS), 452
transformada de Fourier (FFT), 427, 452
rayos gamma, 402
reactor nuclear, 82, 399, 404
recursiva
fórmula, 529
relación, 44
regla
del spaghetti, 213
del trapecio, 23, 110, 130, 131, 136, 139
relajación, 424
residuo, 83
rigidez en la unión, 104-05, 379
Romberg, integración de, 51-52, 114-115, 151, 311
Runge-Kutta, método de, 290, 299
de cuarto orden, 306, 324, 336
de segundo orden, 296
de tercer orden, 304

S

Simpson
regla de 1/3, 110, 115, 306
regla de 3/8, 110, 119
regla extendida de, 116, 119
reglas de, 110, 137, 139
singularidad, 2, 65, 74, 127-28, 130
sistema
inconsistente, 196
masa-resorte, 348
solución no única, 218
spline, 22
cúbico, 23, 540
método de tensión del, 544
Stieltjes, matriz de, 426
subrelajación, 384
sucesiva(s)
aproximaciones, 74
sobre-relajación (SOR), 428-430
sustitución, 63, 79, 367, 403

T

tanh, 62, 105
Taylor, expansión (serie) de, 1, 84, 118, 157, 164, 309
teorema de la suma de dos ángulos, 405, 434
término advectivo, 408
tolerancia, 88, 296
transferencia de calor por radiación, 322, 325
transformación
de coordenadas, 45, 125, 131
de similaridad, 246
tridiagonal, 355, 358, 265, 382, 402, 476
valor propio de una matriz, 242, 248
tubo de enfriamiento, 399
tubo en forma de u, 34

V

valor propio, 240, 434, 439
de una matriz tridiagonal, 242, 248
de problema de, 369
van der Waals, 104-05
variable (básica/libre), 218
varilla, 104-106, 379, 395, 400, 404,
Vax, 133, 432
vector, 196, 198
matriz transpuesta, 202
notación, 184
propio, 240
unitario, 202
velocidad terminal, 78, 103-04
vibración, 104-05, 379, 395
Von Neumann, 482

X

xenón-135, 344

Y

yodo-135, 344

JUN
IMPRESORA ROMA, S. A. DE C. V.
TOMÁS VÁZQUEZ No. 152
COL. SAN PEDRO IXTACALCO
C. P. 08220 MÉXICO, D. F.
2005

SHOICHIRO NAKAMURA

METODOS NUMERICOS APLICADOS CON SOFTWARE

Este extraordinario volumen ofrece un panorama eficaz de los métodos numéricos contemporáneos, que son fundamentales para usar y escribir software. Cada uno de los capítulos de esta obra comienza con un enfoque simple, que guía al lector paso a paso mediante ejemplos. Cada método numérico se presenta con un ejemplo conciso y totalmente resuelto, que ilustra cómo aplicarlo. Además, el autor muestra métodos alternativos y los describe en forma paralela, analiza las relaciones entre ellos y compara los aspectos negativos y positivos de cada uno. Al progresar en el texto, se presentan métodos más avanzados, a la vez que se desarrollan expresiones matemáticas más generales.

Los programas de computadora muestran cómo implantar los métodos numéricos y el lector puede experimentarlos y probarlos en la computadora. Los programas se pueden adoptar como parte del propio programa del lector.

Características:

- Permite que los lectores practiquen los métodos en la microcomputadora.
- Capacita a quienes tienen poco conocimiento matemático para entender rápidamente el tema.
- Ofrece programas cortos y versátiles que también se pueden usar para aplicaciones científicas, con o sin modificaciones.
- Contiene software que es fácil de entender y manejar.

ISBN 968-880-263-8
90000
9 789688 802632

PEARSON Educación ®

Visítenos en:
www.pearsoneducacion.net