

ANÁLISIS NUMÉRICO I

RESUMEN DE LAS CLASES TEÓRICAS

ING. RODOLFO A. SCHWARZ

Año 2021

Índice general

Prólogo	XI
1. Errores en los métodos numéricos	1
1.1. Una definición de Análisis Numérico	1
1.2. El concepto y las fuentes de error	2
1.2.1. Introducción	2
1.2.2. Concepto de error	3
1.2.3. Fuentes de error	3
1.3. Error absoluto y error relativo	4
1.4. Propiedades de los algoritmos	5
1.4.1. Condición de un problema	6
1.4.2. Estabilidad de un algoritmo	6
1.5. Errores	9
1.5.1. Error inherente	9
1.5.2. Error de redondeo	10
1.5.3. Error de truncamiento/discretización	12
1.5.4. Errores por « <i>overflow</i> » y « <i>underflow</i> »	15
1.6. Propagación de errores	16
1.6.1. Propagación del error inherente	16
1.6.2. Propagación del error de redondeo	17
1.6.3. Propagación de los errores inherentes y de redondeo	17
1.7. Gráfica de proceso	18
1.8. Perturbaciones experimentales	21
1.8.1. Estimación del número de condición	21
1.8.2. Estimación del término de estabilidad	23
1.9. Inestabilidad en los algoritmos	24
1.9.1. Cancelación	24
1.9.2. Acumulación del error de redondeo	25
1.9.3. Aumento de la precisión	25
1.10. Diseño de algoritmos estables	26
2. Ecuaciones no Lineales	31
2.1. Introducción	31
2.2. Método de la bisección	32
2.3. Método de la falsa posición o «regula falsi»	34
2.4. Método de las aproximaciones sucesivas o punto fijo	35
2.5. Método de Newton-Raphson	39
2.6. Análisis del error	41
2.7. Métodos de convergencia acelerada	43
2.8. Método de Steffensen	44
2.9. Método de Halley	46

2.10. Notas finales	49
3. Sistemas de Ecuaciones Lineales y No Lineales	53
3.1. Introducción	53
3.2. Definiciones	53
3.3. Matrices triangulares	55
3.4. Eliminación de Gauss y sustitución inversa	56
3.5. Factorización LU	59
3.6. Método de Cholesky	61
3.6.1. Matrices simétricas y definidas positivas	61
3.6.2. Algoritmo de Cholesky	62
3.7. Condición de una matriz	63
3.8. Refinamiento Iterativo de la Solución	65
3.9. Errores de los métodos directos	67
3.10. Métodos iterativos	68
3.10.1. Métodos estacionarios	69
3.10.2. Convergencia de los métodos estacionarios	74
3.10.3. Métodos no estacionarios	75
3.10.4. Convergencia de los métodos no estacionarios	82
3.10.5. Aspectos computacionales	84
3.11. Errores de los métodos iterativos	85
3.12. Sistemas de Ecuaciones No Lineales	88
3.13. Notas finales	92
4. Interpolación de curvas	95
4.1. Introducción	95
4.2. Interpolación o Método de Lagrange	96
4.3. Interpolación o Método de Newton	100
4.4. Interpolación baricéntrica de Lagrange	103
4.5. Fenómeno de Runge	104
4.6. Interpolación por Trazadores cúbicos o «splines»	106
4.7. Interpolación o Método de Hermite	111
4.8. Interpolación por el método de Akima	116
4.9. Notas finales	117
5. Mejor aproximación y ajuste de funciones	121
5.1. Mejor aproximación	121
5.1.1. Introducción	121
5.1.2. Error y normas vectoriales	122
5.1.3. Método de los cuadrados mínimos	123
5.2. Ajuste de funciones	127
5.2.1. Introducción	127
5.2.2. Aproximación por mínimos cuadrados	128
5.2.3. Polinomios de Legendre	130
5.3. Notas finales	131
6. Diferenciación e integración numérica	133
6.1. Diferenciación numérica	133
6.1.1. Diferencias progresivas, regresivas y centradas	133
6.1.2. Aproximación por polinomios de Taylor	137
6.1.3. Extrapolación de Richardson	139
6.1.4. Notas finales	143

6.2.	Integración numérica	143
6.2.1.	Fórmulas de Newton-Cotes	144
6.2.2.	Fórmulas cerradas de Newton-Cotes	144
6.2.3.	Fórmulas abiertas de Newton-Cotes	154
6.2.4.	Cuadratura de Gauss	155
6.2.5.	Integrales múltiples	159
6.3.	Notas finales	161
7.	Ecuaciones diferenciales ordinarias	165
7.1.	Ecuaciones diferenciales ordinarias con valores iniciales	165
7.1.1.	Introducción	165
7.1.2.	Condición de Lipschitz	167
7.1.3.	Problema bien planteado	167
7.1.4.	Métodos de Euler explícito e implícito	168
7.1.5.	Método Predictor-Corrector de Euler	169
7.1.6.	Error cometido al resolver una ecuación diferencial	169
7.1.7.	Métodos de Taylor de orden superior	170
7.1.8.	Métodos de Runge-Kutta	171
7.1.9.	Métodos de paso múltiple	175
7.1.10.	Métodos predictores-correctores	180
7.2.	Análisis de estabilidad	181
7.3.	Consistencia y convergencia	183
7.4.	Ec. diferenciales ordinarias de orden superior	184
7.4.1.	Aplicación de los métodos para ecuaciones diferenciales de primer orden	184
7.4.2.	A partir de la serie de Taylor	186
7.4.3.	Aproximación de la derivada segunda	187
7.5.	Sistemas de ecuaciones diferenciales	188
7.6.	Ecuaciones diferenciales con cond. de contorno	188
7.6.1.	Introducción	188
7.6.2.	Método del tiro o disparo lineal	189
7.6.3.	Diferencias finitas	192
7.7.	Notas finales	194
A.	El análisis numérico y la ingeniería	201
B.	Un poco de historia	205
B.1.	Los egipcios	206
B.2.	Los babilonios	208
B.3.	Los indios	210
B.4.	Los chinos	211
B.5.	Los árabes	213
B.6.	Los mayas	215
B.7.	Los incas	216

Índice de tablas

1.1. Cálculo de los y_i	8
1.2. Valores de $f'(x_0)$ en función de h	14
1.3. Valores de $f(n)$ y diferencia con e	25
2.1. Método de Steffensen - Algoritmo 1	45
2.2. Método de Steffensen - Algoritmo 2	46
4.1. Datos para una interpolación	96
4.2. Interpolación o Método de Newton	102
4.3. Conjunto de datos a interpolar	104
4.4. Datos incluyendo la primera derivada	111
4.5. Interpolación de Hermite aplicando el Método de Newton	114
4.6. Interpolación Hermite segmentada aplicando el Método de Newton	115
6.1. Extrapolación de Richardson	142
6.2. Método de Romberg	154
6.3. Raíces y coeficientes de la cuadratura de Gauss-Legendre	158
7.1. Resultados obtenidos aplicando el Método de Euler Explícito	191

Índice de figuras

1.1. Curvas de las distintas funciones.	8
1.2. Evolución del error del algoritmo.	14
1.3. Gráfica de proceso de la suma.	19
1.4. Gráfica de proceso del producto.	19
1.5. Gráfica de proceso del algoritmo.	20
2.1. Aproximaciones de la raíz por el método de la bisección.	33
2.2. Aproximaciones de la raíz por el método de la «Regula Falsi».	35
2.3. Método de las aproximaciones sucesivas con la función $G_1(x)$	36
2.4. Método de las aproximaciones sucesivas con la función $G_2(x)$	37
2.5. Método de Newton-Raphson.	40
3.1. Forma cuadrática en dos dimensiones.	77
3.2. Forma cuadrática y plano tangente.	78
4.1. Representación gráfica de la función $\text{sen}(x)$	95
4.2. Interpolación lineal de un conjunto de datos.	97
4.3. Conjunto de puntos distribuidos uniformemente.	105
4.4. Curva obtenida por interpolación por Lagrange.	105
4.5. Interpolación gráfica «intuitiva».	105
4.6. Interpolación por Trazadores cúbicos o «spline».	110
4.7. Interpretación geométrica de la aproximación de la pendiente.	116
4.8. Cuadro de diálogo del LibreOffice Calc para «Tipo de línea: Suavizado».	118
4.9. Cuadro de diálogo del MS Excel 2016 para «Línea suavizada».	118
5.1. Error cuadrático	128
6.1. Pendiente según cada aproximación.	135
6.2. Aproximación por polinomios de Taylor.	139
6.3. Aproximaciones con Extrapolación de Richardson.	142
6.4. Área bajo la curva.	145
6.5. Aproximación por rectángulos.	145
6.6. Aproximación por trapecios.	145
6.7. Aproximación por arcos de parábola cuadrática.	147
6.8. Aproximación compuesta por rectángulos.	150
6.9. Aproximación compuesta por trapecios.	150
6.10. Aproximación compuesta por Simpson.	151
6.11. Fórmula del punto medio.	155
6.12. Cuadratura usando curvas de aproximación.	156
7.1. Viga doblemente empotrada	189
7.2. Sistema masa-resorte.	198

7.3. Esferas concéntricas.	199
7.4. Viga simplemente apoyada.	199
B.1. Números egipcios en escritura jeroglífica.	206
B.2. Números egipcios en escritura hierática.	207
B.3. El <i>Papiro de Rhind</i>	207
B.4. Números babilónicas.	208
B.5. Ejemplo de tablas babilónicas.	209
B.6. Evolución de los números de la India.	211
B.7. Representación numérica utilizada en las pizarras o tablas de cálculo.	212
B.8. Números chinos.	213
B.9. Números árabigos.	214
B.10. Números mayas.	215
B.11. Representación del número 586 en un «quipu».	217
B.12. Ejemplo de un «quipu» con cuerdas subsidiarias.	217

Licencia

Esta obra está licenciada bajo una Licencia Attribution-NonCommercial-ShareAlike 2.5 Argentina de Creative Commons.

Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-nc-sa/2.5/ar/> o envíenos una carta a Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Prólogo

Esto no pretende ser un libro sobre Análisis Numérico ni algo que se le parezca. Simplemente es un resumen, incompleto, de las clases dadas en el ámbito de la Facultad de Ingeniería, durante los años 2005 y 2006, orientadas originalmente a la parte práctica y luego reconvertidas como clases teóricas durante los años 2007, 2008 y 2009.

El objetivo es dar una guía de los temas y enfocar a los alumnos en los temas más importantes del Análisis Numérico que suelen aplicarse en el ámbito de la ingeniería. No intenta ser un manual ni un libro de texto, sino servir como ayuda-memoria a la hora de repasar lo visto en clase. Textos y libros de gran calidad existen por doquier, algunos de los cuales se refieren en la bibliografía, a los que no busca reemplazar. Es más, muchas de las demostraciones se deben buscar en esos textos.

Al mismo tiempo, algunos temas que no suelen incluirse en los libros tradicionales se han desarrollado con mayor o menor fortuna. Algunos ejemplos de ello son el *Método de Interpolación Baricéntrica de Lagrange*, una forma alternativa de construir los polinomios interpolantes de Lagrange, el método de interpolación de Akima, método desarrollado a fines de los años 60, y una aproximación al *Método de los Gradientes Conjugados* para resolver sistemas de ecuaciones lineales en forma iterativa. El primero de ellos no figura en ningún libro conocido y es una interesante alternativa para desarrollar en una computadora, tal como refieren quienes publicaron el método, los matemáticos Jean-Paul Berrut (Département de Mathématiques, Université de Fribourg) y Lloyd N. Trefethen (Computing Laboratory, Oxford University). Algo similar ocurre con la interpolación de Akima.

El tercero, en cambio, aparece en varios libros dedicados al *método de los elementos finitos*, pero no siempre en los textos de Análisis Numérico. Por ejemplo, recién en la séptima edición del libro *Análisis Numérico* de Burden & Faires se lo incluye como parte importante del libro. En otros textos ni siquiera se lo menciona, a pesar de ser uno de los métodos iterativos más importantes para resolver sistemas de ecuaciones lineales raras, sobre todo en los últimos años, gracias al desarrollo de las computadoras personales. Para las clases, tanto teóricas como prácticas, la base que se usó para la explicación es la publicada por el profesor Dr. Jonathan Richard Shewchuk (School of Computer Science, Carnegie Mellon University), que es de libre disponibilidad en la web, y muy buena desde el punto de vista de la interpretación y comprensión del método. El texto actual usa parte de esa publicación y agrega también algunas ideas propias.

Respecto a versiones anteriores, además de correcciones tipográficas y de redacción, se han agregado algunos temas que se desarrollaron en los últimos cuatrimestres, como son el *método de Halley* para ecuaciones no lineales, el análisis de la estabilidad, la consistencia y la convergencia de los métodos para resolver ecuaciones diferenciales ordinarias de primer orden con valores iniciales, y los métodos numéricos para resolver ecuaciones diferenciales de segundo orden, también con valores iniciales.

Finalmente, este resumen no sería posible sin la ayuda de todos aquellos que han intervenido e intervienen en el curso 008 de Análisis Numérico I de la Facultad de Ingeniería de la Universidad de Buenos Aires. Quiero agradecer a Germán Sosa, Rafael Barrera Oro, Florencia Lanteri, Micaela Suriano, María Ciminieri e Iván Flores Lagos, que actualmente se desempeñan como colaboradores. También a Mariano Ameijeiras, que fue fundamental para armar la clases cuando empezamos, y a Adolfo Ibáñez, Guillermo Scolastico, Darío Kubar, Mariano Castro Con-

de, Carolina Tortoriello, María Actis y Exequiel Escobar, que han colaborado con el curso en años anteriores y aportaron sus visiones de cómo encarar las clases y mejorar las explicaciones o ejemplos usados durante el desarrollo del mismo. Y fundamentalmente al Ing. Carlos Amura, que me incorporó a su equipo allá por el 2003. Ellos han hecho y hacen posible que este resumen sea efectivo.

Una vez más, gracias a los alumnos, quienes han aportado (y siguen aportando) mucho, revisando y encontrando errores, y puntos que no resultaron muy claros o fáciles de seguir. Quiero agradecer a la alumna María Inés Parnisari que me ayudó a encontrar correctores ortográficos para L^AT_EX, después de encontrar gran cantidad de errores tipográficos, además de sugerir mejoras y modificaciones más que oportunas, que espero se incorporen en próximas versiones.

Rodolfo A. Schwarz, Buenos Aires, febrero de 2013.

Capítulo 1

Errores en los métodos numéricos

1.1. Una definición de Análisis Numérico

Es usual que el análisis numérico esté asociado estrictamente a la siguiente definición general: *Es el estudio de los errores de redondeo*. De acuerdo con Lloyd Trefethen (véase [18]), profesor en la universidad de Oxford, esta definición es errónea. Entiende que si esta percepción es correcta, resulta poco sorprendente, entonces, que el análisis numérico sea visto como una asignatura aburrida y tediosa. Es cierto que los errores de redondeo son inevitables, y que su análisis es complejo y tedioso, pero *no son fundamentales*. Al analizar varios libros dedicados al tema, encuentra que los capítulos iniciales siempre están referidos al error de redondeo o sus temas asociados: precisión, exactitud, aritmética finita, etc. Veamos algunos ejemplos de la bibliografía disponible en español:

- *Burden & Faires, Métodos Numéricos (2005)*: 1. Preliminares matemáticos y análisis del error.
- *González, Análisis Numérico, primer curso (2002)*: 1. Errores en el cálculo numérico.
- *Curtis & Wheatley, Análisis numérico con aplicaciones (2002)*: 0. Cálculo numérico y computadoras (0.5 Aritmética por computadoras y errores).
- *Nakamura, Métodos numéricos aplicados con software (1992)*: 1. Causas principales de errores en los métodos numéricos.
- *Ramírez González y otros, Cálculo Numérico con Mathematica (2005)*: 1. Introducción al Cálculo Numérico. Errores.
- *Maron & López, Análisis Numérico, un enfoque práctico (1998)*: 1. Algoritmos, errores y dispositivos digitales.
- *Quintana y otros, Métodos numéricos con aplicaciones en Excel (2005)*: Capítulo 1. Definición de error.

Esto ayuda a que los alumnos tengan una percepción equivocada del objeto principal de la materia. Para evitar esto, Trefethen propone una definición alternativa:

Análisis numérico es el estudio de los algoritmos para resolver problemas de la matemática continua.

Para él la palabra clave es *algoritmo*. De hecho, en Wikipedia podemos encontrar esta definición:

El análisis numérico es la rama de la matemática que se encarga de diseñar algoritmos para, a través de números y reglas matemáticas simples, simular procesos matemáticos más complejos aplicados a procesos del mundo real;

cuya referencia es justamente, ¡Lloyd (Nick) Trefethen! Y, según él, el principal objetivo del análisis numérico es diseñar algoritmos para aproximar valores desconocidos (no lo conocido de antemano), y hacerlo en forma rápida, muy rápida.

Por esa razón, este capítulo tiene por objeto desmitificar la influencia de los errores al aplicar métodos numéricos, y en particular, la influencia del error de redondeo como fuente básica de los problemas en la utilización de algoritmos para resolver problemas matemáticos, aún cuando la existencia de los mismos debe llevar a tenerlos en cuenta en determinados casos en los que no se los puede soslayar. Para ello, empezaremos viendo los errores que intervienen en cualquier procedimiento o cálculo numérico. (Para un análisis más detallado acerca del estudio de los errores y la estabilidad de los algoritmos, véase [10].)

1.2. El concepto y las fuentes de error

1.2.1. Introducción

Tal como dijimos en la definición de análisis numérico, su objetivo principal no es analizar en detalle los errores que intervienen en el cómputo de cantidades. Pero sí es uno de los puntos en los cuales cualquier matemático (o de otra rama de la ciencia o tecnología asociada a la matemática) que se dedique a desarrollar algoritmos deberá ser un especialista en el tema. ¿Por qué? Simplemente, porque sus algoritmos serán utilizados para resolver problemas que seguramente no tengan una solución analítica o que la obtención de esa solución está fuera de los alcances del usuario de ese algoritmo. Por ejemplo, es usual que los ingenieros utilicen programas que resuelven estructuras por el *método de los elementos finitos* para dimensionar determinadas piezas o establecer las formas definitivas de las mismas, con el fin de optimizar el uso de los materiales o para darle ciertas características especiales a la estructura. Si bien es posible que varios de esos problemas puedan ser resueltos con modelos analíticos, lo más probable es que esos modelos sólo tengan una definición general (aún cuando sea compleja) en forma de ecuaciones diferenciales o de sistemas de ecuaciones diferenciales, tanto ordinarias como en derivadas parciales. Y si bien existen métodos de resolución analíticos (simbólicos) para las ecuaciones diferenciales, las condiciones de borde de un problema particular puede hacer inútil la búsqueda de soluciones analíticas o simbólicas. Por lo tanto, el único camino viable para obtener una respuesta al problema planteado es la aplicación de un método numérico.

Si no contamos con una solución analítica, ¿cómo sabremos si los resultados obtenidos sirven? Esta es una de las razones por las cuales los analistas numéricos deben ocuparse de analizar qué tipos de errores afectan a los algoritmos que desarrollan y hasta qué punto son responsables de los posibles errores en los resultados que se obtendrán por aplicaciones de los mismos. Pero debe tenerse en cuenta que, por otro lado, estos algoritmos deben ser rápidos (de convergencia rápida) y que serán aplicados en computadoras, algo que no suele remarcarse con debida propiedad, que, por supuesto, están sometidas a limitaciones propias.

El problema de los errores en los cálculos no es propiedad del siglo XX (o XXI) y de los que sigan. Desde los inicios de la matemática y de las ciencias asociadas, es un problema que interesó e interesa a todos los involucrados. Como ejemplo, tomemos un típico método de interpolación que se enseña en cualquier curso, el de los *polinomios de Lagrange*. La fórmula para

obtener los polinomios es:

$$P_n(x) = \sum_{j=0}^n f_j l_j(x); \quad l_j(x) = \frac{\prod_{\substack{k=0 \\ k \neq j}}^n (x - x_k)}{\prod_{\substack{k=0 \\ k \neq j}}^n (x_j - x_k)}$$

El propio Lagrange advertía en su época, que el método no era totalmente confiable, pues muchas veces los resultados obtenidos no eran correctos. De todos modos, el método suele estudiarse como una herramienta teórica a pesar de que tiene las siguientes desventajas:

1. Cada evaluación de $p(x)$ requiere $O(n^2)$ sumas/restas y multiplicaciones/divisiones.
2. Añadir un nuevo par de datos $x_{n+1}; f_{n+1}$ requiere recalcular todo de cero.
3. *El cálculo es numéricamente inestable.*

En tanto que las dos primeras se refieren a la eficiencia del algoritmo para obtener el polinomio, la última está estrictamente relacionada con los errores que pueden aparecer por las operaciones de cálculo involucradas en el procedimiento. Esto último se hizo muy evidente al utilizar las computadoras como elemento de cálculo. En consecuencia, para analizar cuán inestable (y/o mal condicionado) es el algoritmo, debemos analizar cómo se propagan los errores. Veremos a continuación el concepto y la definición de lo que denominamos *error*.

1.2.2. Concepto de error

La palabra *error* suele llevar a interpretaciones confusas según quien la exprese. En el lenguaje coloquial de uso diario, el concepto de error está relacionado con falla o mal hecho. Una expresión como «el error fue ...» suele asociarse con la causa que produjo un resultado no aceptable o equivocado, y que, por lo tanto, debe ser evitado o enmendado para que al hacer de nuevo el cálculo (o cualquier otra cosa), el resultado obtenido sea aceptable o correcto.

En cambio, en el ámbito del análisis numérico (y en general, en las ciencias e ingeniería), el término error está relacionado específicamente con la incertidumbre de los datos de ingreso como de los resultados obtenidos, *sin que esto signifique necesariamente que los resultados sean equivocados*. Dicho de otra manera, no pone en duda la confiabilidad del método en sí, sino que analiza el grado de incertidumbre de los valores numéricos. En la ingeniería esto es de particular relevancia, puesto que los datos que utilizamos provienen de mediciones en campo, estimaciones probabilísticas, hipótesis y modelos matemáticos simplificados, o de la experiencia profesional. Rara vez se cuenta con datos con validez «exacta». Sin embargo, si una leve modificación de estos datos produce resultados considerablemente diferentes que no reflejan la realidad, estamos ante la presencia de un problema que sí puede objetar el procedimiento utilizado. Es decir, el procedimiento es inestable o mal condicionado, conceptos diferentes.

Para analizar cuán confiable es un procedimiento o algoritmo, se vuelve necesario el estudio de los errores que afectan los cálculos y las operaciones que intervienen en dicho algoritmo, y cómo se propagan hasta afectar los resultados que éste entrega.

1.2.3. Fuentes de error

Las fuentes de error que analizaremos son las siguientes:

- **Error inherente:** Es el error de los datos de entrada que puede estar dado por la precisión en la medición de los datos, por la representación numérica, por provenir de cálculos previos, etc.

- **Error de redondeo/corte:** Es el error debido estrictamente a la representación numérica utilizada y está asociado a la precisión usada en los cálculos, generalmente una calculadora o una computadora.
- **Error de truncamiento/discretización:** Es el error que aparece al transformar un procedimiento infinito en uno finito, por ejemplo, transformar una serie de infinitos términos en una función finita, o de usar una aproximación discreta para representar un fenómeno o modelo continuo.
- **Error del modelo matemático:** Es el debido a las simplificaciones e hipótesis introducidas para definir el modelo matemático que representa el problema físico.
- **Error humano y/o de la máquina:** Es el error que se produce por la intervención humana, ya sea por una mala transcripción o interpretación incorrecta de los datos originales, por programas de computación mal hechos y/o fallas en el diseño, implementación o configuración de programas o computadoras.

La última fuente de error suele ser asociada al concepto coloquial de «error». Desde la óptica del análisis numérico, los dos últimos errores están fuera de su alcance, si bien no deben ser despreciados a la hora de evaluar los resultados obtenidos, en particular, el debido al modelo matemático.

1.3. Error absoluto y error relativo

Empezaremos por analizar las fórmulas más sencillas de error. Supongamos que obtenemos de alguna forma (por ejemplo, una medición) cierto valor \bar{m} . Sabemos que el valor «exacto» de dicho valor es m . Como conocemos ese valor m podemos definir dos tipos de errores:

1. **Error absoluto:** $e_a = m - \bar{m}$;
2. **Error relativo:** $e_r = \frac{m - \bar{m}}{m} = \frac{e_a}{m}$ (siempre que $m \neq 0$).

Generalmente, el error relativo es una medida mucho más representativa del error, especialmente cuando $|m| \gg 1$. Cuando $|m| \approx 1$, entonces ambos errores coinciden. En la práctica suele ser poco probable conocer el valor m , por lo que no podemos calcular e_a ni e_r . Entonces, ¿cómo sabemos qué error estamos teniendo? Si no conocemos la solución del problema pareciera que no hay forma de saberlo.

Partamos de no conocer m y de que el valor \bar{m} fue obtenido por medición usando un instrumento cuya precisión¹ es e_m (por error de medición). Si tomamos el concepto de error absoluto podemos obtener una idea del valor de m . En efecto, tenemos que:

$$e_m = e_a = m - \bar{m} \Rightarrow m = \bar{m} + e_a;$$

que podemos generalizar a:

$$m = \bar{m} \pm e_a; \tag{1.1}$$

si tenemos en cuenta que el valor de e_a puede ser positivo o negativo. Así, una forma más general de escribir el error absoluto y el relativo es:

1. $|E| = |e_a| = |m - \bar{m}|$;
2. $|e_r| = \left| \frac{m - \bar{m}}{m} \right| = \frac{|m - \bar{m}|}{|m|} = \frac{|E|}{|m|}$.

¹En este caso, precisión se refiere a la unidad más chica que el instrumento puede medir.

Como hemos supuesto que $e_a = e_m$, sabemos cual es nuestro error absoluto, pero seguimos sin saber cuál es nuestro error relativo. Tenemos dos posibilidades para obtener m :

$$m = \bar{m} + e_a \quad \text{o} \quad m = \bar{m} - e_a,$$

y entonces el error relativo sería:

$$e_r = \frac{e_a}{\bar{m} + e_a} \quad \text{o} \quad e_r = \frac{-e_a}{\bar{m} - e_a} \Rightarrow |e_r| = \frac{|e_a|}{|\bar{m} + e_a|} \quad \text{o} \quad |e_r| = \frac{|-e_a|}{|\bar{m} - e_a|}.$$

Resulta, entonces, más conveniente definirlo como:

$$e_r = \frac{|e_a|}{|\bar{m}|}, \tag{1.2}$$

cuando se conoce \bar{m} y e_a .

1.4. Propiedades de los algoritmos

Hemos dicho que el análisis numérico se ocupa de estudiar algoritmos para resolver problemas de la matemática continua. Dado que estos algoritmos son una aproximación al problema matemático, resulta evidente que los resultados obtenidos estarán afectados por alguno de los errores mencionados. Y como en muchas ocasiones los datos de entrada de ese algoritmo también tienen errores, la pregunta que surge inmediatamente es: ¿cómo sabemos si los resultados que arroja el algoritmo son confiables? La pregunta no tiene una única respuesta, depende del tipo de error que analicemos o que tenga mayor influencia y de las características del problema matemático. Podemos tener varias aproximaciones acerca de un algoritmo, a saber:

1. Una primera aproximación a una respuesta sería analizar cuan sensible son los resultados que arroja un algoritmo cuando los datos de entrada se modifican levemente, o dicho de otra forma, cuando sufren una perturbación. Un análisis de este tipo tiene dos formas de ser encarado: por un lado, estudiando la *propagación de errores* (en inglés, *forward error*), es decir, perturbar los datos de entrada y ver qué consecuencia tiene en el resultado; por otro lado, podemos estudiar eso de manera inversa, partiendo de una perturbación en los resultados, y analizar qué grado de perturbación (error) pueden tener los datos de entrada. Esta última se conoce como *análisis retrospectivo* (en inglés, *backward error*). En ambos casos estamos estudiando la influencia del *error inherente* en el resultado.
2. Una segunda aproximación puede ser analizar el algoritmo con diferentes representaciones numéricas en los datos de entrada y estudiar qué ocurre con los resultados. En este caso estudiamos la incidencia del *error de redondeo*.
3. Finalmente, y tal vez el más sencillo de todos, otra aproximación puede ser analizar qué ocurre cuando se trunca un procedimiento o discretiza el dominio de nuestro problema matemático. Este tipo de análisis puede que requiera solamente de un trabajo algebraico más que numérico, y, a veces, suele combinarse con el error de redondeo.

La enumeración anterior en tres aproximaciones es a los efectos de identificar las causas y la forma de encarar el problema. Sin embargo, la realidad suele ser mucho más compleja, y los errores que surgen de aplicar uno o varios algoritmos, resultan ser una combinación de todos y dependen, muchas veces, de las características de los datos del problema.

1.4.1. Condición de un problema

El primer caso, el análisis de la propagación de los errores inherentes, permite establecer si el problema está *bien o mal condicionado*. Si al analizar un pequeño cambio (o perturbación) en los datos el resultado se modifica levemente (o tiene un pequeño cambio), entonces estamos ante un problema *bien condicionado*. Si, por el contrario, el resultado se modifica notablemente o se vuelve oscilante, entonces el problema está *mal condicionado*. Si éste fuera el caso, no hay forma de corregirlo cambiando el algoritmo (como se verá después) pues el problema está en el modelo matemático.

Definición 1.1. Un problema matemático (numérico) se dice que está *bien condicionado* si pequeñas variaciones en los datos de entrada se traducen en pequeñas variaciones de los resultados.

Observación 1.1.1. Un problema mal condicionado puede ser resuelto con exactitud, si realmente es posible, solamente si se es muy cuidadoso en los cálculos.

Observación 1.1.2. Si f representa al algoritmo «real» y f^* al algoritmo «computacional», y x a la variable «real» y x^* a la variable «computacional», entonces el error en los resultados se puede definir como:

$$|f(x) - f^*(x^*)| \leq \underbrace{|f(x) - f(x^*)|}_{\text{condición}} + \underbrace{|f(x^*) - f^*(x)|}_{\text{estabilidad}} + \underbrace{|f^*(x) - f^*(x^*)|}_{\text{truncamiento}}.$$

Veremos más adelante que las «pequeñas variaciones» en los datos de entrada están asociadas al problema en cuestión. No es posible «a priori» definir cuantitativamente cuándo una variación es «pequeña» y cuándo no lo es. El análisis de los errores inherentes es importante para establecer la «sensibilidad» del modelo numérico a los cambios en los datos, puesto que rara vez los datos de entrada están exentos de error.

1.4.2. Estabilidad de un algoritmo

El segundo caso es el que suele ser un «dolor de cabeza» para los analistas numéricos. Si analizamos un algoritmo ingresando los datos con diferentes representaciones numéricas, esto es, con diferente precisión, y los resultados no cambian demasiado (salvo por pequeñas diferencias en los decimales), entonces estamos en presencia de un algoritmo *estable*. Caso contrario, el algoritmo es *inestable*.

El último caso está asociado a procedimientos o algoritmos basados en series o iteraciones «infinitas», y suele combinarse con alguno de los otros errores, como veremos más adelante.

En consecuencia, lo que debemos buscar de un algoritmo es que sea *estable*. ¿Qué significa esto en la práctica? Supongamos (una vez más, supongamos) que E_n mide un cierto error cometido en el paso n de un algoritmo. Podemos expresar este error en función del error inicial, que puede tener una de estas dos expresiones:

1. Error con crecimiento lineal: $E_n \approx c \cdot n \cdot E_0$
2. Error con crecimiento exponencial: $E_n \approx c^n \cdot E_0$

Es evidente que el primer error es «controlable», en tanto que el segundo, no. Puesto que es imposible que no haya errores al trabajar con un algoritmo, lo que se debe buscar es que el error siga una ley lineal (como en el primer caso) y no una ley exponencial (o potencial). A partir de esta comprobación se desprende la siguiente definición:

Definición 1.2. Un algoritmo se considera *estable* cuando la propagación de los errores de redondeo es lineal o casi-lineal.

En cambio, un algoritmo que propaga los errores en forma exponencial (o potencial) es *inestable*.

Una de las razones principales de analizar la propagación de los errores de redondeo es conseguir que un algoritmo sea estable. Sin embargo, debemos tener bien presente que un algoritmo estable en ciertas condiciones puede volverse inestable en otras, por lo que muchas veces no existe el algoritmo «universalmente estable». Dado que la estabilidad (o la inestabilidad) es una propiedad exclusiva del algoritmo, si un problema se vuelve inestable podemos, muchas veces, corregirlo cambiando el algoritmo inestable por otro estable. (Sin embargo, nunca hay que olvidar que un problema puede volverse mal condicionado para determinadas condiciones de base, lo que hace más complejo el análisis.)

Veamos un ejemplo que muestra la inestabilidad de un algoritmo. Tomemos la siguiente integral definida:

$$y_n = \int_0^1 \frac{x^n}{x+10} dx;$$

con $n = 1; 2; \dots; 20$.

Es fácil ver que las primeras integrales analíticas son relativamente sencillas de obtener (por ejemplo, para $n = 1$ o $n = 2$). En efecto, si queremos hallar y_1 podemos hacer:

$$y_1 = \int_0^1 \frac{x}{x+10} dx = x \Big|_0^1 - 10 \ln(x+10) \Big|_0^1 = 1 - 10 \ln\left(\frac{11}{10}\right);$$

$$y_1 = 1 - 10 \ln(1,1) = 0,0468982019570.$$

Pero si queremos obtener y_{15} la situación ya no es tan sencilla. Deberíamos calcular la siguiente integral:

$$y_{15} = \int_0^1 \frac{x^{15}}{x+10} dx.$$

Para facilitar el cálculo de cada una de las y_n integrales, desarrollemos un algoritmo que nos permita obtener los valores de las mismas sin tener que integrar o que al menos utilice aquellas integrales «fáciles». Para un $n > 1$ cualquiera podemos decir que:

$$y_n + 10 y_{n-1} = \int_0^1 \frac{x^n + 10 x^{n-1}}{x+10} dx = \int_0^1 \frac{x+10}{x+10} x^{n-1} dx = \int_0^1 x^{n-1} dx \Rightarrow$$

$$y_n + 10 y_{n-1} = \frac{1}{n} \Rightarrow y_n = \frac{1}{n} - 10 y_{n-1}$$

Si queremos calcular y_1 necesitamos obtener y_0 , que también resulta muy sencillo de obtener, pues:

$$y_0 = \int_0^1 \frac{1}{x+10} dx = \ln(x+10) \Big|_0^1 = \ln(11) - \ln(10)$$

$$y_0 = \ln(1,1) = 0,0953101798043.$$

Para analizar si el algoritmo arroja resultados confiables, empezaremos por calcular algunos valores. Hemos calculado el valor de y_1 en forma analítica, por lo tanto, tenemos un valor de comprobación. Por otro lado, por las características del problema, sabemos que $0 \leq y_n \leq 1$. Si definimos las funciones $f(x, i) = \frac{x^i}{x+10}$ y las graficamos, podemos ver que el área bajo esas funciones es menor a $\frac{0,1}{2} = 0,05$. En la figura 1.1 se pueden ver representadas algunas de estas curvas.

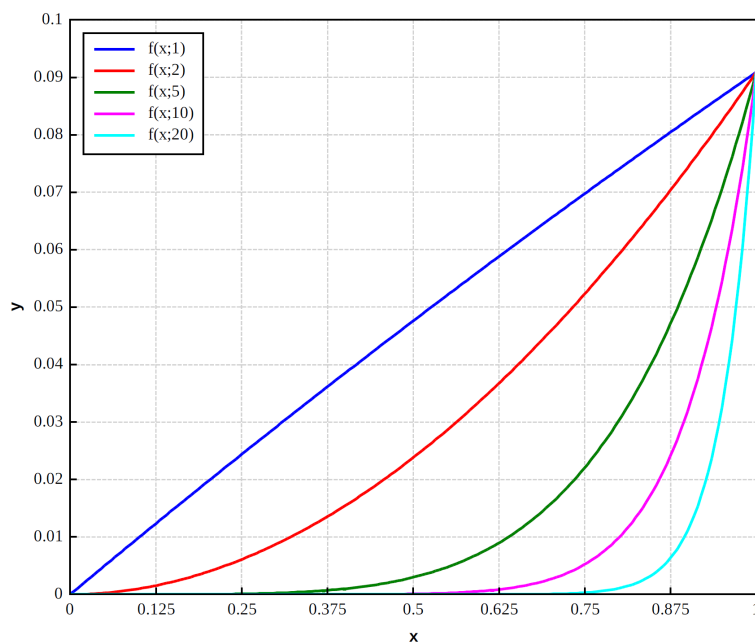


Figura 1.1: Curvas de las distintas funciones.

Para comprobar la eficacia del algoritmo hemos utilizado dos programas, el SMath Studio y el LibreOffice Calc, y hemos programado el algoritmo en Python. Además hemos obtenido las y_i en forma analítica. En la tabla 1.1 podemos ver los resultados obtenidos.

Tabla 1.1: Cálculo de los y_i

i	«Analítico»	SMath Studio	LibreOffice Calc	Python
1	0,0468982019567514000	0,046898201956751	0,04689820195675	0,04689820195675
2	0,0310179804324860060	0,031017980432490	0,03101798043249	0,03101798043249
3	0,0231535290084732900	0,023153529008433	0,02315352900840	0,02315352600840
4	0,0184647099152671080	0,018464709915667	0,01846470991602	0,01846470991602
5	0,0153529008473289370	0,015352900843333	0,01535290083985	0,01535290083985
6	0,0131376581933772860	0,013137658233333	0,01313765826821	0,01313765826821
7	0,0114805609233700040	0,011480560523810	0,01148056017502	0,01148056017502
8	0,0101943907662999780	0,010194394761905	0,01019439824976	0,01019439824976
9	0,0091672034481113700	0,009167163492063	0,00916712861348	0,00916712861347
10	0,0083279655188863120	0,008328365079365	0,00832871386525	0,00832871386525
11	0,0076294357202277880	0,007625440115440	0,00762195225655	0,00762195225655
12	0,0070389761310554600	0,007078932178932	0,00711381076780	0,00711381076780
13	0,0065333156125223285	0,006133755133755	0,00578496924512	0,00578496924512
14	0,0060954153033481685	0,010091020091020	0,01357887897740	0,01357887897740
15	0,0057125136331849920	-0,034243534243434	-0,06912212310731	-0,06912212310730
16	0,0053748636681500880	0,404935342435342	0,75372123107305	0,75372123107705
17	0,0050748927302638440	-3,99052989494166	-7,4783887813187	-7,4783887813187
18	0,0048066282529171250	39,9608545049722	74,839443368743	74,839443368743
19	0,0045652964181971910	-399,555913470774	-748,34180210848	-748,34180210848
20	0,0043470358180281100	3995,60913470774	7483,4680210848	7483,4680210848

Inicialmente, los primeros valores obtenidos con el algoritmo, tanto con el SMath Studio como con LibreOffice Calc y Python, resultan una buena aproximación de los valores de y_n . Los problemas aparecen a partir del y_{13} ². Detengámonos a analizar dichos resultados.

Hemos visto que los valores de y_n , o sea, las áreas bajo las curvas, están limitados superiormente por 0,05. Además podemos ver que $y_n > y_{n+1}$, es decir, que las áreas bajo la curva van disminuyendo a medida que crece n . Si miramos los resultados obtenidos con el SMath Studio podemos ver que el y_{14} es mayor que el y_{13} , algo que es evidentemente incorrecto. Algo similar ocurre con los resultados obtenidos con LibreOffice Calc y Python, también el y_{14} es mayor que el y_{13} . Con todos programas y con Python, el y_{15} es, ¡negativo! En este caso, el área bajo la curva no puede ser negativa.

A partir de estos valores, los resultados se vuelven oscilantes (cambian de signo), y para $i > 16$ los y_i son mayores a uno, nuevamente algo incorrecto. En consecuencia, resulta evidente que el algoritmo tiene algún problema para calcular los valores de y_i cuando $i \geq 13$, por lo que no nos sirve para obtener el y_{20} . Aún cuando no tuviéramos el resultado exacto, mirando la curva nos damos cuenta que hay una diferencia muy grande entre el valor «real» y el obtenido con el algoritmo. Más aún, el error que estamos teniendo no sigue una ley lineal (se va multiplicando por 10), lo que dice claramente que el algoritmo analizado es *inestable*.

Este ejemplo nos muestra cómo un algoritmo mal diseñado nos puede entregar resultados que inicialmente son bastante aproximados pero que en pasos posteriores son incorrectos, y por lo tanto, inútiles.

Definición 1.3. Un algoritmo debe ser diseñado procurando que sea bien condicionado y estable.

Observación 1.3.1. Un algoritmo inestable a la larga da resultados incorrectos, por más que esté bien condicionado.

Es por eso que debemos desarrollar algún tipo de análisis que nos permita detectar si un algoritmo está bien condicionado o no y si es estable o no. Para ello, empezaremos por analizar algunos tipos de errores.

1.5. Errores

1.5.1. Error inherente

Éste suele ser el error más fácil de entender. Es el que está relacionado directamente con los datos de entrada o de base. Dado que estos datos suelen provenir de mediciones, cálculos anteriores, proyecciones estadísticas, etc., el valor numérico de los datos no es «exacto» sino que está asociado a un intervalo de validez. Cuando se mide una longitud con una cinta métrica con divisiones hasta el centímetro, el error por la apreciación del instrumento es un centímetro o medio centímetro (5 mm). Es decir, si mide 145,01 m, en realidad, se está diciendo que el valor es $145,01 \pm 0,01$ o $145,010 \pm 0,005$. Lo mismo ocurre si los datos se obtienen por un cálculo anterior o una estimación estadística. En esos casos, el error se obtiene por otros métodos.

Veamos un ejemplo. Supongamos que tenemos las siguientes cantidades, $a = 3,0 \pm 0,1$ y $b = 5,0 \pm 0,1$ y queremos hallar $z = a + b$. Lo que deberemos hacer es:

$$z = (3,0 \pm 0,1) + (5,0 \pm 0,1)$$

Al efectuar esta operación obtendremos cinco resultados posibles: 7,8; 7,9; 8,0; 8,1 y 8,2. Es decir, z está en el intervalo $[7,8; 8,2]$, o, lo que es lo mismo, $z = 8,0 \pm 0,2$. Así cualquier resultado obtenido dentro del intervalo dado se puede considerar «correcto».

²Un análisis más pormenorizado debería ocuparse de evaluar el hecho de que con el mismo algoritmo, los resultados del SMath Studio hasta y_{13} son mejores que los resultados de los otros dos, que son iguales.

Esto muestra la sencillez del análisis cuando las operaciones son pocas (en este caso, una). Sin embargo, si el algoritmo es más complejo, hacer las n combinaciones posibles de operaciones con los datos de ingreso puede ser imposible y nada práctico. De ahí que el análisis de la propagación de los errores inherentes es la forma más conveniente para establecer la incidencia de los mismos en los resultados finales. Más adelante veremos la diversas formas de analizar esta propagación.

1.5.2. Error de redondeo

Antes de analizar el error de redondeo, veremos la manera de representar un número según la forma adoptada. A partir de esta representación se entenderá cual es la incidencia del error en los cálculos efectuados con ayuda de una computadora.

Representación numérica

Para empezar, supongamos el siguiente número: $\frac{4}{3}$. En el sistema decimal suele representarse como 1,3333... . Una forma alternativa es:

$$\frac{4}{3} \cong \left(\frac{1}{10} + \frac{3}{10^2} + \frac{3}{10^3} + \frac{3}{10^4} + \frac{3}{10^5} + \dots \right) \times 10^1 = 1,3333\dots ;$$

o sea, un número que sólo puede representarse con una serie de infinitos términos, algo imposible desde el punto de vista práctico. Su única forma de expresión «exacta» es simbólica. Una calculadora, por ejemplo, sólo puede representarlo en forma numérica (en base diez, como la escrita arriba) y, por ende, la única representación posible es finita.³ En consecuencia, debe truncarse esta serie en « n » términos. Por ejemplo, una representación posible es:

$$\frac{4}{3} \cong \left(\frac{1}{10} + \frac{3}{10^2} + \frac{3}{10^3} + \frac{3}{10^4} \right) \times 10^1 = 0,1333 \times 10^1 = 1,333.$$

Podemos ver que esta representación está formada por un coeficiente (0,1333), una base (10) y un exponente (1). Esta forma de representación se conoce como *representación de coma (punto) flotante*. Una generalización de esta representación se puede escribir como:

$$fl(x) = \pm 0.d_1d_2d_3\dots d_{t-1}d_t \times 10^e = \pm \left(\frac{d_1}{10} + \frac{d_2}{10^2} + \frac{d_3}{10^3} + \dots + \frac{d_t}{10^{t-1}} + \frac{d_t}{10^t} \right) \times 10^e.$$

La forma normalizada es que d_1 sea distinto de cero ($1 \leq d_1 \leq 9$) y que los restantes d_i estén comprendidos en el siguiente intervalo: $0 \leq d_i \leq 9$, para $i = 2; 3; 4; \dots; t$. También se limita el exponente e , con dos valores, $I < 0$ y $S > 0$, por lo que se cumple que $I \leq e \leq S$. Así, podemos hallar el máximo número a representar, que es $0,99\dots 99 \times 10^S \approx 10^S$, y el más chico, $0,10\dots 00 \times 10^I = 10^{I-1}$.

Una vez definida la forma de representar los números, pasemos a definir nuestra *precisión*, que significa cuantos términos d_i usaremos, esto es, el t que vimos, y el exponente e de la base.

Para complicar más las cosas, las calculadoras y fundamentalmente, las computadoras, usan una representación numérica con base 2.⁴ Esto trae ventajas y desventajas. Por ejemplo, puesto que se usa base 2, los d_i sólo pueden valer 0 o 1, con excepción del d_1 , que vale siempre 1. Esto facilita la representación de los números y las operaciones. Pero la desventaja es que sólo los números que pueden representarse como sumas de $\frac{1}{2^i}$ resultan exactos. Veamos cómo funciona esto.

³Distinto sería el caso si se usara base 3. Entonces $\frac{4}{3}$ sería igual a 1,1; una representación «exacta».

⁴Existen, sin embargo, procesadores que no usan una representación binaria.

Esto nos permite obtener una cota del error absoluto para ambos casos:

$$e_A = \begin{cases} 10^{-t} \times 10^e & \text{para corte} \\ \frac{1}{2} 10^{-t} \times 10^e & \text{para redondeo.} \end{cases}$$

Y como definimos el error absoluto, también podemos definir un límite para el error relativo, que será:

1. **Corte:** $e_r \leq \frac{10^{-t} \times 10^e}{0,1 \times 10^e} = 10^{1-t}$.
2. **Redondeo:** $e_r \leq \frac{1}{2} \frac{10^{-t} \times 10^e}{0,1 \times 10^e} = \frac{1}{2} 10^{1-t}$.

Al valor 10^{1-t} lo identificaremos con la letra μ , y resulta ser importante porque nos da una idea del error relativo que cometemos al utilizar una representación de coma flotante. Suele denominarse como *unidad de máquina* o *unidad de redondeo*. El negativo del exponente de μ suele llamarse también *cantidad de dígitos significativos*.

Dígitos de guarda

Supongamos el siguiente caso. Tomemos el número 0,1425 que debe ser redondeado a tres dígitos significativos. Aplicando el criterio anterior rápidamente obtenemos que el resultado es 0,143 pero, ¿es correcto este redondeo? ¿Por qué no redondear a 0,142; si está a medio camino de ambos? Supongamos que hacemos la operación $2 \times 0,1425$, cuyo resultado es 0,2850, ¿qué pasa con la misma operación si el número está redondeado? Evidentemente da diferente puesto que la operación es $2 \times 0,143$ cuyo resultado es 0,286. La diferencia entre ambos es 0,001 que es justamente la unidad de redondeo. Esto se vuelve aún más importante cuando se tiene la resta de números similares ($a - b$ con $a \approx b$). De ahí que la mayoría de las computadoras actuales (y los programas) trabajen con lo que se conoce como *dígitos de guarda*, es decir, más precisión que la mostrada en forma «normal» en pantalla. Pero este ejemplo sirve además para desarrollar otra forma de redondeo.

Redondeo exacto

Tal como dijimos, el número 0,1425 está mitad de camino de ser redondeado a 0,143 como a 0,142. Este problema ha llevado a desarrollar el concepto de *redondeo exacto*, que consiste en redondear todos los números que terminan en 5 de manera de que el último dígito significativo sea par. En consecuencia, aplicando este criterio, 0,1425 se redondea a 0,142 y no a 0,143. El criterio va de la mano del *dígito de guarda* y debería ser el redondeo «normal». (Para más detalles respecto a dígitos de guarda y el redondeo exacto, véase [7].)

1.5.3. Error de truncamiento/discretización

Este error surge de aproximar procesos continuos mediante procedimientos discretos o de procesos «infinitos» mediante procedimientos «finitos». Como ejemplo del primer caso suele tomarse la diferenciación numérica como forma de aproximar el cálculo de una derivada en un punto (o su equivalente, la integración numérica), en tanto que para el otro, el ejemplo más usual es la utilización de métodos iterativos para resolver sistemas de ecuaciones lineales.

En general, este error está asociado al uso de la serie de Taylor para aproximar funciones, de modo que estimar una cota del error no conlleva una dificultad mayor. Sin embargo, en él suelen interactuar el error inherente y/o el de redondeo, con lo que muchas veces su influencia no es bien advertida o es muy reducida. Para ello veamos un ejemplo típico.

Supongamos que queremos calcular una aproximación de $f'(x_0)$ para una función continua, pues no es posible obtener la derivada en forma analítica o resulta muy difícil. Por lo tanto, usaremos un entorno del punto x_0 para calcular $f'(x_0)$ utilizando solamente $f(x)$. Para ello nos valdremos de la serie de Taylor. En efecto, para cualquier punto distante h de x_0 tendremos:

$$f(x_0 + h) = f(x_0) + f'(x_0)h + f''(x_0)\frac{h^2}{2} + f'''(x_0)\frac{h^3}{6} + f^{(4)}(x_0)\frac{h^4}{24} + \dots$$

Entonces podemos despejar $f'(x_0)$, que resulta ser:

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} - \left[f''(x_0)\frac{h}{2} + f'''(x_0)\frac{h^2}{6} + f^{(4)}(x_0)\frac{h^3}{24} + \dots \right].$$

Si nuestro algoritmo para aproximar $f'(x_0)$ es:

$$\frac{f(x_0 + h) - f(x_0)}{h},$$

el error que cometemos en la aproximación está dado por:

$$\left| f'(x_0) - \frac{f(x_0 + h) - f(x_0)}{h} \right| = \left| f''(x_0)\frac{h}{2} + f'''(x_0)\frac{h^2}{6} + f^{(4)}(x_0)\frac{h^3}{24} + \dots \right|.$$

El término de la derecha es el denominado *error de truncamiento*, pues es lo que se *truncó* a la serie de Taylor para aproximar el valor buscado. Este error suele asociarse también con la convergencia (o la velocidad de convergencia), que suele representarse como $O(n)$ (generalmente, como $O(h^n)$), siendo n el parámetro que determina la velocidad o la convergencia. En nuestro caso, y dado que h generalmente es menor que 1, podemos decir que la aproximación es del tipo:

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} + O(h),$$

que indica que el error que se comete es proporcional a h . (Está claro que además están los términos con h^2 , h^3 , etc., pero como $h < 1$ entonces $h^2 \ll h$, $h^3 \ll h^2$, etc., la influencia de éstos es mucho menor y despreciable.)

Nuevamente, supongamos por un momento que se cumple que todas las derivadas de orden superior a dos son nulas, es decir, $f^{(i)}(x_0) = 0$ para $i \geq 3$. Entonces tendremos que:

$$\left| f'(x_0) - \frac{f(x_0 + h) - f(x_0)}{h} \right| = \frac{h}{2} |f''(\xi)| \quad \text{con } \xi \in [x; x + h],$$

con lo cual, si conociéramos $f''(\xi)$, podríamos acotar el error que estamos cometiendo por despreciar el término $\frac{h}{2}f''(x_0)$.

Como ejemplo, apliquemos este algoritmo para obtener la derivada en $x_0 = 0,45$ ($f'(0,45)$) de la función $f(x) = \text{sen}(2\pi x)$. Como verificación tomemos el valor analítico de la derivada en cuestión: $f'(0,45) = 2\pi \cos(2\pi \cdot 0,45) = -5,97566$. Para calcular la aproximación tomemos $h = 0,1$. Así, tendremos:

$$f'(0,45) = \frac{f(0,55) - f(0,45)}{0,1} = \frac{\text{sen}(2\pi \cdot 0,55) - \text{sen}(2\pi \cdot 0,45)}{0,1} = -6,18034.$$

En la tabla 1.2 podemos ver los resultados obtenidos para distintos h .

Tabla 1.2: Valores de $f'(x_0)$ en función de h

h	$f'(x_0)$	Error
10^{-1}	-6,18033988749895	$2,04676 \times 10^{-1}$
10^{-2}	-6,03271072100927	$5,70464 \times 10^{-2}$
10^{-3}	-5,98172474217345	$6,06041 \times 10^{-3}$
10^{-4}	-5,97627391137889	$6,09582 \times 10^{-4}$
10^{-5}	-5,97572532307633	$6,09936 \times 10^{-5}$
10^{-6}	-5,97567042914804	$6,09966 \times 10^{-6}$
10^{-7}	-5,97566494175972	$6,12277 \times 10^{-7}$
10^{-8}	-5,97566438553798	$5,60549 \times 10^{-8}$
10^{-9}	-5,97566451876474	$1,89282 \times 10^{-7}$
10^{-10}	-5,97566607307698	$1,74359 \times 10^{-6}$
10^{-11}	-5,97566995885756	$5,62937 \times 10^{-6}$
10^{-12}	-5,97544236313752	$2,21966 \times 10^{-4}$
10^{-13}	-5,97633054155722	$6,66212 \times 10^{-4}$
10^{-14}	-5,99520433297584	$1,95400 \times 10^{-2}$
10^{-15}	-5,88418203051333	$9,14823 \times 10^{-2}$
10^{-16}	-8,32667268468867	2,35101

Si observamos con atención, veremos que el algoritmo utilizado aproxima muy bien el valor buscado hasta $h = 10^{-8}$. Si estimamos la cota de error con $f''(x_0) \frac{10^{-8}}{2}$ obtenemos un valor muy parecido al error indicado en la tabla 1.2⁵:

$$f''(0,45) \frac{10^{-8}}{2} = 6,09975 \times 10^{-8} \quad (5,60549 \times 10^{-8}).$$

Sin embargo, a partir de $h < 10^{-8}$ el error vuelve a crecer. En la figura 1.2 se puede ver como evoluciona el error:

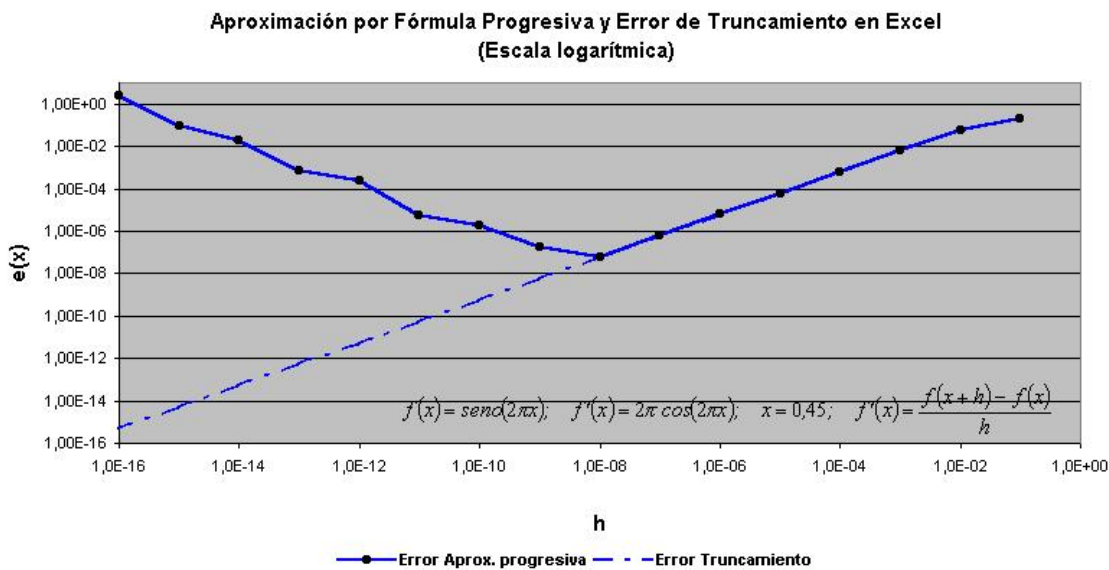


Figura 1.2: Evolución del error del algoritmo.

⁵En forma rigurosa deberíamos hallar ξ , pero dado que el intervalo es tan pequeño, puede tomarse x_0 .

Si analizamos en detalle, vemos que la tendencia del error de truncamiento es lineal (en escala logarítmica) pero para $h < 10^{-8}$ el error aumenta y no sigue una ley determinada. Este «empeoramiento» de la aproximación se debe a la incidencia del error de redondeo, es decir, la unidad de máquina pasa a ser más importante que el error de truncamiento. Es por eso que no siempre el utilizar una «mejor precisión» ayuda a mejorar los resultados finales. En este tipo de problemas, es conveniente que el error que domine los cálculos sea el de truncamiento/discretización.

Veremos más adelante que esta incidencia del paso en el cálculo de una aproximación numérica de la derivada primera, nos alerta de la inestabilidad de la diferenciación numérica, es decir, es muy sensible a la propagación del error de redondeo.

1.5.4. Errores por «*overflow*» y «*underflow*»

Asociados a la representación numérica existen otros dos tipos de errores. Son los denominados errores por «*overflow*» y por «*underflow*». Estos errores surgen por las limitaciones de nuestro sistema para representar números muy grandes («*overflow*») o muy chicos («*underflow*»). Es usual que los manuales del usuario de una calculadora indiquen el número más grande (y el más chico) que puede ser representado. Por ejemplo, las calculadoras Casio de la década de los 80 no podían representar $n!$ si $n > 69$ pues el número más grande que podían representar era $9,99999999 \times 10^{99}$ ($69! = 1,71122452428141 \times 10^{98}$ y $70! = 1,19785716699699 \times 10^{100}$). Algo similar ocurre con los números muy chicos.

Un error muy común es «olvidarse» que en los cálculos intermedios pueden aparecer números muy grandes o muy chicos, fuera del rango de nuestra representación numérica, que vuelven a un algoritmo inútil. Por ejemplo, supongamos que nuestro sistema de representación numérica en una calculadora represente solamente los números entre $-10,000$ y $-0,0001$; y entre $0,0001$ y $10,000$. Si queremos obtener el resultado de $\sqrt{101^2 - 50}$, como $101^2 = 10\,201 > 10,000$ y no lo puede representar, indicará un error por «*overflow*», es decir, número más grande que el máximo a representar, y cortará la ejecución del algoritmo.

Un ejemplo de error por «*overflow*» es lo ocurrido con el cohete europeo *Ariane 5*. El 4 de junio de 1996, un cohete Ariane 5, lanzado por la Agencia Espacial Europea, explotó justo 40 segundos después de su despegue de Korou, Guyana Francesa. El cohete iniciaba su primer viaje (reemplazaba al Ariane 4), después de una década de desarrollo a un costo de unos siete mil millones de dólares. El cohete destruido y su carga se valoraron en unos quinientos millones de dólares. Una junta de investigación investigó las causas de la explosión y en dos semanas presentó un informe. El resultado de la investigación determinó que la causa de la falla fue un «error» de programación (o de «*software*») en el sistema de referencia inercial. Específicamente, un número expresado utilizando un formato de coma flotante de 64 bit relacionado con la velocidad horizontal de cohete respecto de la plataforma de lanzamiento, se convirtió a un entero de 16 bit con signo. En el segundo 39 luego del despegue, el número en cuestión resultó más grande que 32.767, el entero más grande que podía representarse con el entero de 16 bit con signo, con lo que la conversión falló y el cohete explotó.

El error por «*underflow*» es parecido. En este caso, el problema es no poder representar un número muy pequeño, por lo que lo define como cero (0). Si modificamos levemente el ejemplo anterior, y queremos obtener el resultado de $\sqrt{0,01 - 0,006^2}$, como $0,006^2 = 0,000036 < 0,0001$ y no le es posible representarlo, hará $0,006^2 = 0,0000$ y la operación quedará como $\sqrt{0,01 - 0,0} = \sqrt{0,01} = 0,1$.

La diferencia entre ambos es que el error por «*overflow*» no pasa desapercibido, mientras que el «*underflow*» sí, y en consecuencia, puede ser más peligroso.

1.6. Propagación de errores

Hemos visto varios ejemplos que nos mostraron en forma evidente la incidencia que pueden llegar a tener los errores en los resultados que entrega un algoritmo, particularmente, el error de redondeo. Veremos a continuación la propagación de dos de los errores más problemáticos, el inherente y el de redondeo.

1.6.1. Propagación del error inherente

Supongamos que tenemos un problema numérico tal que podemos expresarlo como $x \rightarrow y(x)$, siendo x un vector de \mathfrak{R}^n , que corresponde a los datos de entrada, e y un vector de \mathfrak{R}^m , que corresponde a los resultados. Podemos escribir entonces que:

$$x \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \rightarrow y(x) = \begin{bmatrix} y_1(x) \\ y_2(x) \\ \vdots \\ y_m(x) \end{bmatrix},$$

donde $y_i(x) : \mathfrak{R}^n \rightarrow \mathfrak{R}$; $y(x) : \mathfrak{R}^n \rightarrow \mathfrak{R}^m$.

Por otra parte, supongamos que en lugar de x conocemos \tilde{x} , es decir, una aproximación de x ; podemos definir que $e_{x_i} = x_i - \tilde{x}_i$, que también conocemos. Y nuestra última suposición es que las $y_i(x)$ pertenecen a $C^\infty(x)$, lo que nos permite desarrollar $y(x)$ en una serie de Taylor alrededor de \tilde{x} :

$$y(x) = y(\tilde{x}) + \frac{\partial [y_1(\tilde{x}); y_2(\tilde{x}); \dots; y_m(\tilde{x})]}{\partial [x_1; x_2; \dots; x_n]}(x - \tilde{x}) + T(x - \tilde{x}).$$

Podemos suponer ahora que $e_{x_i} = x_i - \tilde{x}_i$ para $i \in [1, n]$ es muy pequeño, y que por eso $T(x - \tilde{x})$ es despreciable, con lo que nos queda:

$$y_i(x) - y_i(\tilde{x}) = \sum_{j=1}^n \left[\frac{\partial y_i(\tilde{x})}{\partial x_j} (x_j - \tilde{x}_j) \right] \quad \text{para } i = 1; 2; \dots; m,$$

que por analogía a e_{x_i} podemos expresar como:

$$e_{y_i} = \sum_{j=1}^n \frac{\partial y_i(\tilde{x})}{\partial x_j} e_{x_j}; \quad \text{para } i = 1; 2; \dots; m, \quad (1.3)$$

que nos da el error de y_i en función de del error de x_j . Esta expresión es muy útil porque nos permite obtener o determinar el error de un resultado si conocemos el error de los datos de entrada, es decir, *cómo se propagan* los errores inherentes. Veamos algunos ejemplos:

1. **Suma:** Hagamos $y(x_1; x_2) = x_1 + x_2$, entonces tendremos:

$$e_y = e_{x_1+x_2} = \frac{\partial y(\tilde{x}_1; \tilde{x}_2)}{\partial x_1} e_{x_1} + \frac{\partial y(\tilde{x}_1; \tilde{x}_2)}{\partial x_2} e_{x_2}, \quad (1.4)$$

o sea,

$$e_y = 1 \cdot e_{x_1} + 1 \cdot e_{x_2} \Rightarrow e_y = e_{x_1} + e_{x_2}. \quad (1.5)$$

El error relativo será:

$$e_{r_y} = \frac{e_y}{y} = \frac{e_{x_1} + e_{x_2}}{x_1 + x_2} = \frac{e_{x_1}}{x_1 + x_2} + \frac{e_{x_2}}{x_1 + x_2}. \quad (1.6)$$

Sabemos que $e_{x_1} = x_1 \cdot e_{r_{x_1}}$ y $e_{x_2} = x_2 \cdot e_{r_{x_2}}$, por lo que podemos escribir:

$$e_{r_y} = \frac{x_1 \cdot e_{r_{x_1}}}{x_1 + x_2} + \frac{x_2 \cdot e_{r_{x_2}}}{x_1 + x_2} = \frac{x_1}{x_1 + x_2} e_{r_{x_1}} + \frac{x_2}{x_1 + x_2} e_{r_{x_2}}. \quad (1.7)$$

2. **Producto:** En este caso tenemos $y(x_1; x_2) = x_1 \cdot x_2$, entonces:

$$e_y = x_2 \cdot e_{x_1} + x_1 \cdot e_{x_2}. \quad (1.8)$$

El error relativo para el producto será:

$$e_{r_y} = \frac{e_y}{y} = \frac{x_2 \cdot e_{x_1}}{x_1 \cdot x_2} + \frac{x_1 \cdot e_{x_2}}{x_1 \cdot x_2} = e_{r_{x_1}} + e_{r_{x_2}}. \quad (1.9)$$

Hasta aquí no pareciera haber problemas. Sin embargo, raramente se conoce el error con su signo, de ahí que lo que se busca es una *cota* del error, no el error en sí mismo. En ese caso, las expresiones del error relativo se modifican levemente:

1. **Suma:** $e_{r_y} = \frac{|x_1|}{|x_1 + x_2|} |e_{r_{x_1}}| + \frac{|x_2|}{|x_1 + x_2|} |e_{r_{x_2}}|.$

2. **Producto:** $e_{r_y} = |e_{r_{x_1}}| + |e_{r_{x_2}}|.$

A partir de este razonamiento es que la suma es una operación mal condicionada cuando se da que $|x_1| \approx |x_2|$ y $x_2 < 0$ es decir, la suma algebraica. Suponiendo que $e_{r_{x_i}} \leq r$ se tiene:

$$e_{r_y} = \frac{|x_1| + |x_2|}{|x_1 - x_2|} r.$$

lo que hace que e_{r_y} crezca en forma incontrolada, pues el coeficiente siempre es mayor a uno, y puede ser mucho mayor que 1 si $x_1 - x_2$ es muy chico.

Analizaremos ahora la propagación del error de redondeo.

1.6.2. Propagación del error de redondeo

Supongamos ahora que en nuestro problema no tenemos errores inherentes. Por lo tanto, para $x \rightarrow y(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ sólo tendremos errores de redondeo debido al algoritmo utilizado. Sea $P(x)$ nuestro algoritmo para obtener $y(x)$. Si no hubieran errores por redondeo, entonces $y(x) = P(x)$, pero lo que en realidad obtendremos es $\tilde{y}(x) = P(x)$, es decir que podemos escribir que:

$$y(x) = \tilde{y}(x) + E(x) \rightarrow y_i(x) = \tilde{y}_i(x) \times \left[1 + \sum_{k=1}^p F_{i,k}(x) \epsilon_k \right],$$

con $|\epsilon_k| \leq \eta$, y donde los $F_{i,k}$ son los *factores de amplificación*.

1.6.3. Propagación de los errores inherentes y de redondeo

Ya hemos visto la expresión para calcular la propagación de los errores inherentes, que es:

$$e_{y_i} = \sum_{j=1}^n \frac{\partial y_i(x)}{\partial x_j} e_{x_j} \cong \sum_{j=1}^n \frac{\partial y_i(\tilde{x})}{\partial x_j} e_{\tilde{x}_j}. \quad (1.10)$$

Como además tendremos $P(x)$ en vez de $y(x)$, entonces:

$$e_{y_i} \cong e_{P_i} = \sum_{j=1}^n \frac{\partial P_i(\tilde{x})}{\partial x_j} e_{\tilde{x}_j}, \quad (1.11)$$

y el error relativo será:

$$e_{r_{P_i}} = \frac{\sum_{j=1}^n \frac{\partial P_i(\tilde{x})}{\partial x_j} \tilde{x}_j}{P_i(\tilde{x})} e_{r_{\tilde{x}_j}}, \quad (1.12)$$

en consecuencia, el coeficiente que afecta a $e_{r_{\tilde{x}_j}}$ será el *número de condición del problema*, que se define como :

$$C_{p_i} = \frac{\sum_{j=1}^n \frac{\partial P_i(\tilde{x})}{\partial x_j} \tilde{x}_j}{P_i(\tilde{x})}. \quad (1.13)$$

Del mismo modo, tendremos el *término de estabilidad*, que se define como:

$$y_i(x) - P_i(\tilde{x}) = P_i(\tilde{x}) \times \sum_{k=1}^p F_{i,k}(\tilde{X}) \epsilon_k \Rightarrow T_e = \sum_{k=1}^p F_{i,k}(\tilde{X}) \epsilon_k \cong \sum_{k=1}^p F_{i,k}(\tilde{X}) \mu. \quad (1.14)$$

Si suponemos que $e_{r_{\tilde{x}_j}} \leq r$, entonces, tendremos:

$$e_{r_{y_i}} \cong C_{p_i} \cdot r + T_{e_i} \cdot \mu, \quad (1.15)$$

que será el *error relativo total*.

Finalmente, si suponemos ahora que $r \cong \mu$, entonces tenemos:

$$e_{r_{y_i}} \cong (C_{p_i} + T_{e_i}) \cdot \mu = C_{p_i} \frac{C_{p_i} + T_{e_i}}{C_{p_i}} \cdot \mu,$$

y podemos decir que un algoritmo es estable si:

$$\frac{C_{p_i} + T_{e_i}}{C_{p_i}} > / > 1 \rightarrow 1 + \frac{T_{e_i}}{C_{p_i}} > / > 1, \quad (1.16)$$

es decir, *un algoritmo es estable si los errores de redondeo no tienen gran incidencia en el error del resultado o al menos son del mismo orden que los errores inherentes* ($1 + \frac{T_{e_i}}{C_{p_i}} \cong 2$). Sin embargo, esta afirmación debe tomarse con cuidado. Dado que lo que se analiza es la relación $\frac{T_e}{C_p}$, debe tenerse en cuenta que si $C_p \gg 1$ y $\frac{T_e}{C_p} \approx 1$ entonces $T_e \gg 1$, por lo que es posible que el algoritmo sea inestable.

1.7. Gráfica de proceso

Una forma de obtener los coeficientes C_p y T_e es mediante la «gráfica de proceso». Ésta consiste en un diagrama de flujo que representa gráficamente todo el proceso de una operación dada, permitiendo el análisis de los errores relativos y de redondeo que intervienen en él. No se incluyen en esta gráfica los errores debidos a truncamiento/discretización, que deben ser analizados en forma separada.

En las figuras 1.3 y 1.4 se pueden ver las gráficas de proceso de la suma y el producto.

Analicemos brevemente los errores inherentes y de redondeo en ambos casos. Si nos fijamos en la gráfica de la suma, y tomamos una cota superior para los errores relativos inherentes de x e y , por ejemplo, $|e_{r_x}|; |e_{r_y}| < |r|$, entonces el coeficiente C_p se puede escribir como:

$$C_p = \frac{|x| + |y|}{|x + y|}$$

que es el mismo resultado obtenido antes para la suma. Algo similar se obtiene para el producto.

La ventaja de este método es que facilita el análisis del error de redondeo al introducirlo en cada operación, permitiendo el cálculo del término de estabilidad (T_e). Según estas gráficas, en ambos casos el T_e es igual a 1.

Veamos un ejemplo. Analicemos la propagación de errores del algoritmo inestable ya visto

$$y_n = \frac{1}{n} - 10 y_{n-1},$$

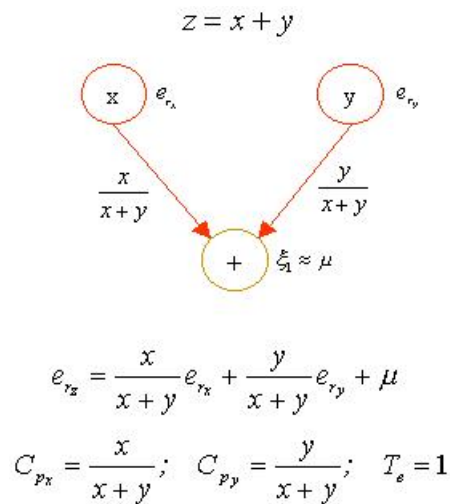


Figura 1.3: Gráfica de proceso de la suma.

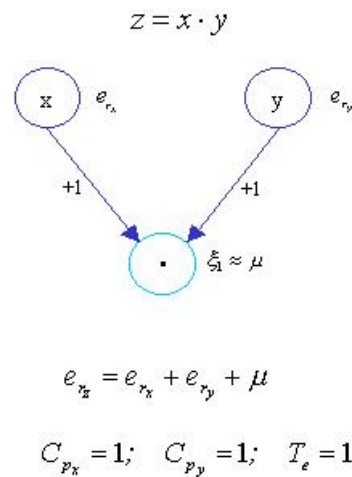


Figura 1.4: Gráfica de proceso del producto.

pero limitándonos a la tercera iteración, con $y_0 = \ln(1,1)$.

Podemos ver lo laborioso que resulta el armado de la gráfica, aún cuando lo hemos limitado hasta obtener el valor y_3 .

Supongamos que y_0 no tiene error ($E_{y_0} = 0$) por lo tanto $e_{r_{y_0}} = 0$. También podemos considerar que todas las constantes no tienen errores inherentes, pues no son valores obtenidos por cálculo. En consecuencia, al no existir error inherente, lo único que se propaga es el error de redondeo de cada una de las operaciones. Así, el desarrollo completo de la propagación de los errores resulta ser:

$$e_{r_{y_3}} = \frac{1}{y_3} [100 \cdot \mu_1 + 100 \cdot \mu_3 + 100 \cdot \mu_4 - 5 \cdot \mu_5 - 5 \cdot \mu_6 + 100 \cdot \mu_6 +$$

$$- 5 \cdot \mu_7 + 100 \cdot \mu_7 + \frac{1}{3} \cdot \mu_8 - 5 \cdot \mu_9 + 100 \cdot \mu_9 +$$

$$+ 1000 \cdot y_0 (\mu_2 - \mu_3 - \mu_4 - \mu_6 - \mu_7 - \mu_8 - \mu_9)].$$

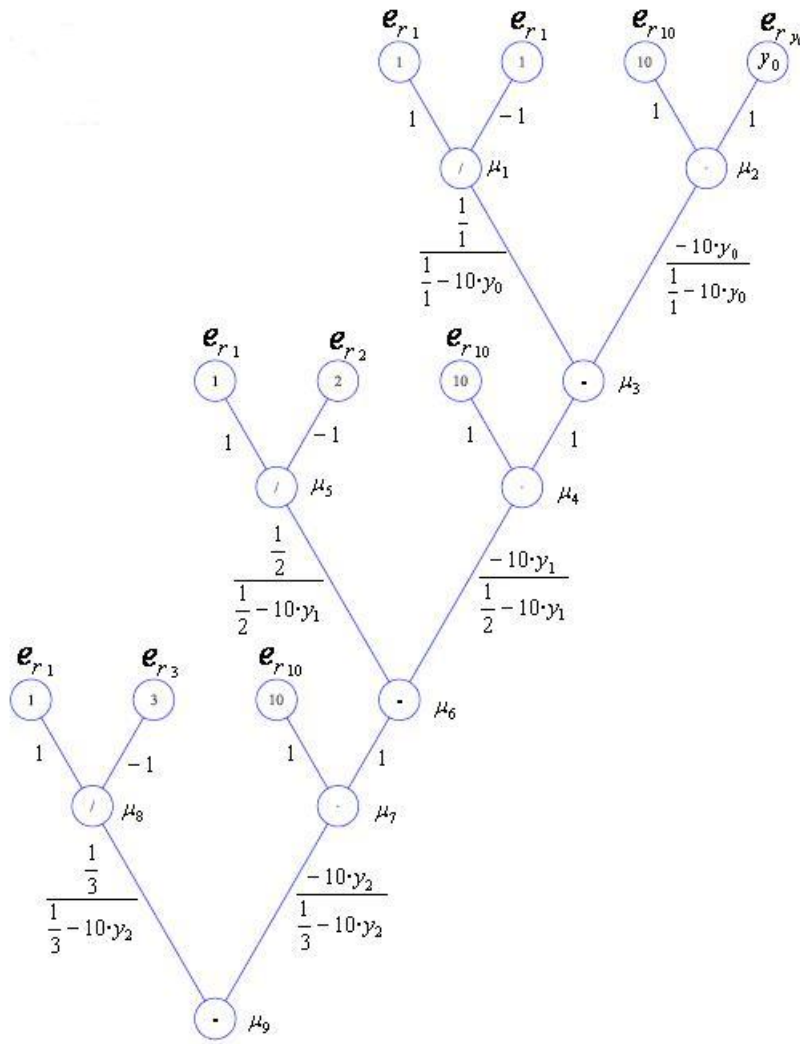


Figura 1.5: Gráfica de proceso del algoritmo.

Si además imponemos que $\mu_i < \mu$, y con esto definimos nuestra cota de error relativo, tendremos que

$$e_{ry_3} = \frac{1}{y_3} (620 + 6000 \cdot y_0) \cdot \mu = \frac{1}{y_3} [620 + 6000 \cdot \ln(1,1)] \cdot \mu.$$

Como

$$y_3 = \frac{1}{3} - 10 \cdot \underbrace{\left[\frac{1}{2} - 10 \cdot \underbrace{(1 - 10 \cdot y_0)}_{y_1} \right]}_{y_2},$$

el valor de y_3 podemos escribirlo como:

$$y_3 = 95,3333 \dots - 1000 \cdot \ln(1,1).$$

De esta forma, tenemos que nuestro coeficiente T_e resulta ser

$$T_e = \frac{620 + 6000 \cdot \ln(1,1)}{95,3333 \dots - 1000 \cdot \ln(1,1)},$$

y si reemplazamos los valores numéricos, obtenemos el siguiente coeficiente de estabilidad para el caso de y_3 :

$$T_e \approx 25319.$$

Resulta evidente que el algoritmo es inestable para cualquier valor de y_0 . Si analizamos el coeficiente de condición (C_p), obtendremos lo siguiente:

$$C_p = \frac{1000 \cdot y_0}{95,3333 \dots - 1000 \cdot \ln(1,1)}.$$

Cuando reemplazamos los valores obtenemos que

$$C_p \approx 4116;$$

y si analizamos la relación entre C_p y T_e nos queda que

$$1 + \frac{T_e}{C_p} = 1 + \frac{25319}{4116} \approx 7,15 > 2;$$

lo que nos muestra que el algoritmo es inestable y, ciertamente, mal condicionado.

Vimos que la gráfica de proceso es bastante útil para obtener ambos coeficientes, pero también que puede convertirse en algo muy difícil de desarrollar cuando el algoritmo cuenta con miles (o millones) de pasos, como puede ser la resolución de un sistema de ecuaciones lineales mediante un método directo. Analizar millones de operaciones mediante la gráfica de proceso puede ser una tarea imposible. Por lo tanto, debemos buscar otra manera de estimar ambos coeficientes.

1.8. Perturbaciones experimentales

Supongamos que queremos estudiar la condición o la estabilidad de un algoritmo con miles de pasos. Ya dijimos que hacer la gráfica de proceso puede ser una tarea imposible. Entonces, ¿cómo hacemos para saber si dicho algoritmo está bien condicionado o es estable? Veamos. Para empezar, estudiemos cómo obtener una aproximación de la condición del problema. Puesto que la condición viene dada por la propagación (o no) de los errores relativos inherentes, busquemos la manera de obtener en forma numérica una estimación del coeficiente de condición, o sea, del C_p . En el mismo sentido, el término de estabilidad, T_e está relacionado con la propagación de los errores de redondeo. Busquemos también algún procedimiento que nos permita obtener una estimación de dicho coeficiente.

1.8.1. Estimación del número de condición

Partamos de la expresión final del error relativo de un resultado:

$$e_r = C_p r + T_e \mu$$

y supongamos por un momento que no tenemos errores de redondeo, es decir, despreciamos $T_e \mu$. En consecuencia, lo que tendremos es:

$$e_r = C_p r \Rightarrow C_p = \frac{e_r}{r} \quad (1.17)$$

Y con esto podemos estimar valor del C_p . ¿Cómo lo hacemos? Perturbando los valores de los datos de entrada. La idea es la siguiente: tomamos los datos de entrada (x , y , etc.), y aplicamos el algoritmo a analizar, obteniendo el resultado correspondiente. Luego «perturbamos» los datos de entrada, es decir, les incorporamos un error. Con estos datos de entrada, volvemos a calcular un resultado, que seguramente diferirá del anterior, pues los datos no son iguales. Este último

paso podemos hacerlo varias veces introduciendo distintas perturbaciones (errores) a los datos de entrada.

Una vez obtenidos los distintos valores de los resultados, tomamos el resultado sin perturbar como resultado «exacto», con el cual vamos a calcular los errores relativos de los otros resultados «perturbados». Con cada uno de éstos obtendremos diferentes e_{r_i} . Como además tendremos diferentes r_i , lo que obtendremos finalmente son diferentes C_{p_i} . Como hemos supuesto que los errores de redondeo son despreciables, todos los C_{p_i} deberían ser similares, con lo cual tendremos una estimación de la condición del problema, es decir, estimamos un C_p . Con esta estimación podremos establecer si el problema está bien o mal condicionado.

Veamos un ejemplo. Tomemos la siguiente función para calcular $\sin(x)$:

$$f(x) = x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040} + \frac{x^9}{362880},$$

función obtenida a partir del truncamiento de la serie de MacLaurin. Con ella calculemos $\sin(\frac{\pi}{4})$ y luego perturbemos el dato de entrada.

El primer resultado lo obtenemos con $x = \frac{\pi}{4}$:

$$f\left(\frac{\pi}{4}\right) = \frac{\pi}{4} - \frac{\left(\frac{\pi}{4}\right)^3}{6} + \frac{\left(\frac{\pi}{4}\right)^5}{120} - \frac{\left(\frac{\pi}{4}\right)^7}{5040} + \frac{\left(\frac{\pi}{4}\right)^9}{362880} = 0,70711$$

Perturbemos ahora x haciendo $x_1 = x \cdot (1 + 0,001)$ ($r_1 = 0,001$), y calculemos $f(x_1)$:

$$f\left(\frac{\pi}{4} \cdot (1 + 0,001)\right) = 0,70655$$

Introduzcamos una nueva perturbación, esta vez haciendo $x_2 = x \cdot (1 - 0,001)$ ($r_2 = -0,001$), y calculemos $f(x_2)$:

$$f\left(\frac{\pi}{4} \cdot (1 - 0,001)\right) = 0,70766$$

Ahora calculemos los dos C_p . Para el primer caso tenemos:

$$C_p = \frac{0,70711 - 0,70655}{0,70711} \cdot \frac{1}{0,001} = 0,78571$$

Para el segundo caso tenemos:

$$C_p = \frac{0,70711 - 0,70766}{0,70711} \cdot \frac{1}{-0,001} = 0,78509$$

Si calculamos el C_p en forma analítica obtenemos:

$$C_p = \frac{\frac{\partial f(x)}{\partial x} \cdot x}{f(x)} = \frac{\frac{d f(x)}{d x} \cdot x}{f(x)} = \left[1 - \frac{\left(\frac{\pi}{4}\right)^2}{2} + \frac{\left(\frac{\pi}{4}\right)^4}{24} - \frac{\left(\frac{\pi}{4}\right)^6}{720} + \frac{\left(\frac{\pi}{4}\right)^8}{40320} \right] \frac{0,78540}{0,70711}$$

$$C_p \approx 0,78540 \frac{\cos\left(\frac{\pi}{4}\right)}{\sin\left(\frac{\pi}{4}\right)} \Rightarrow C_p \approx 0,78540$$

Esto demuestra que la estimación del C_p es muy buena y que el problema está bien condicionado, pues $C_p < 1$.⁶

⁶De hecho, las calculadoras poseen algoritmos de este tipo para obtener los valores de las funciones trigonométricas y trascendentes.

1.8.2. Estimación del término de estabilidad

Para obtener una estimación del término de estabilidad, seguiremos un esquema similar al visto para el número de condición. Partamos nuevamente de la expresión final para el error relativo:

$$e_r = C_p r + T_e \mu$$

Ahora consideremos como hipótesis que los errores inherentes son despreciables, por lo que podemos decir que el error relativo es:

$$e_r = T_e \mu. \quad (1.18)$$

El error relativo está definido como:

$$e_r = \frac{y - \bar{y}}{y},$$

por lo tanto podemos escribir:

$$\frac{y - \bar{y}}{y} = T_e \mu.$$

Al calcular el valor de y con dos «precisiones» diferentes t y s , ($\mu_s = 10^{1-s}$ y $\mu_t = 10^{1-t}$), y asumiendo que $t > s$, obtenemos los siguientes errores relativos:

$$e_{r_t} = \frac{y - \bar{y}_t}{y} = T_e \mu_t; \quad e_{r_s} = \frac{y - \bar{y}_s}{y} = T_e \mu_s.$$

Si restamos e_{r_t} a e_{r_s} tenemos:

$$e_{r_s} - e_{r_t} = \frac{\bar{y}_t - \bar{y}_s}{y} = T_e (\mu_s - \mu_t),$$

de donde despejamos T_e :

$$T_e = \frac{\bar{y}_t - \bar{y}_s}{y (\mu_s - \mu_t)}.$$

Como el valor de y no lo conocemos, tomamos \bar{y}_t en su lugar. En consecuencia, la expresión queda:

$$T_e = \frac{\bar{y}_t - \bar{y}_s}{\bar{y}_t (\mu_s - \mu_t)}. \quad (1.19)$$

Esta expresión nos permite obtener una estimación del T_e calculando dos aproximaciones de y , \bar{y}_t y \bar{y}_s , con diferente precisión, utilizando el mismo algoritmo.

Como ejemplo, utilicemos el mismo algoritmo del caso anterior. Calculemos el valor de $\sin(\frac{\pi}{4})$ con tres precisiones distintas: $s = 4$; $t = 8$ y $u = 15$. Para cada caso tendremos: $\bar{y}_s = 0,706$; $\bar{y}_t = 0,7071068$ y $\bar{y}_u = 0,70710678293687$. Con estos valores calculamos los T_e , tomando como valor de referencia \bar{y}_u . Así, obtenemos los siguientes valores:

$$T_{e_s} = \frac{\bar{y}_u - \bar{y}_s}{\bar{y}_u (\mu_s - \mu_u)} = \frac{0,70710678293687 - 0,706}{0,70710678293687 (10^{-3} - 10^{-14})} = 1,565;$$

y

$$T_{e_t} = \frac{\bar{y}_u - \bar{y}_t}{\bar{y}_u (\mu_t - \mu_u)} = \frac{0,70710678293687 - 0,7071068}{0,70710678293687 (10^{-7} - 10^{-14})} = 0,241.$$

Si analizamos un poco los valores obtenidos, vemos que en el primer caso el error de redondeo se amplifica, puesto que el T_e es mayor que 1. En cambio, en el segundo, la situación es muy buena porque los errores se mantienen acotados, no se amplifican ($T_e < 1$). Podríamos decir que calcular el valor de y con más precisión mejora el resultado final, pero hemos visto que no siempre esto es cierto.

1.9. Inestabilidad en los algoritmos

Como hemos dicho, uno de los objetivos del análisis numérico es obtener algoritmos que estén bien condicionados y sean estables. Hasta ahora nos hemos referido a los principales errores que afectan a los algoritmos y hemos analizado los distintos errores y su propagación, según sea el caso. Además, hemos visto que la condición de un problema es independiente del algoritmo, en tanto que la estabilidad es una «propiedad» el mismo. Es por eso que el análisis numérico se concentra más en estudiar cómo hacer que un algoritmo sea estable más que en analizar su condicionamiento, aunque en algunos casos este último análisis sea muy importante, como por ejemplo, para resolver sistemas de ecuaciones lineales.

La mayoría de los libros y cursos de análisis numérico hacen hincapié en varios conceptos para obtener un algoritmo estable. Alguno de éstos son:

1. La resta de dos números muy similares (cancelación) siempre debe ser evitada.
2. El problema del error de redondeo es su acumulación.
3. Aumentar la precisión en los cálculo mejora la exactitud de los resultados.

Según N. Higham (véase [10], capítulo 1), estos conceptos son en realidad malos entendidos, y desarrolla algunos ejemplos que muestran que no siempre es así. Veamos alguno de ellos.

1.9.1. Cancelación

En su libro, Higham presenta el siguiente caso. Supongamos que debemos hacer la siguiente operación:

$$f(x) = \frac{1 - \cos(x)}{x^2},$$

con $x = 1,2 \times 10^{-5}$ y con $\cos(x) = c$ redondeado a 10 dígitos significativos, con un valor de

$$c = 0,9999999999;$$

de manera que

$$1 - c = 0,0000000001.$$

Al calcular $f(x) = \frac{1-c}{x^2}$ se obtiene $f(x) = \frac{10^{-10}}{1,44 \times 10^{-10}} = 0,6944\dots$, resultado evidentemente incorrecto pues es claro que $0 \leq f(x) \leq 1/2$ para todo $x \neq 0$.

Al analizar la cota del error relativo para la resta $\hat{x} = \hat{a} - \hat{b}$, donde $\hat{a} = a(1 + \Delta a)$ y $\hat{b} = b(1 + \Delta b)$ obtiene:

$$\left| \frac{x - \hat{x}}{x} \right| = \left| \frac{-a\Delta a + b\Delta b}{a - b} \right| \leq \max(|\Delta a|, |\Delta b|) \frac{|a| + |b|}{|a - b|}.$$

La cota del error relativo de \hat{x} es muy grande cuando $|a - b| \ll |a| + |b|$. Por lo tanto, afirma que una resta con esta condición *da preeminencia a los errores iniciales*.

También afirma que la cancelación no siempre es mala, por varias razones. La primera es que los números a restar pueden ser libres de error. La segunda, que la cancelación puede ser una señal de un problema intrínsecamente mal condicionado y, por lo tanto, inevitable. Tercero, los efectos de la cancelación dependen del contexto en que se efectúa. Si $x \gg y \approx z > 0$, la resta en la operación $x + (y - z)$ es inocua.

1.9.2. Acumulación del error de redondeo

Desde que se creó la primera computadora, la acumulación del error de redondeo ha sido uno de los «dolores de cabeza» de los especialistas, como se puede ver en esta frase: «*La extraordinaria rapidez de las actuales máquinas significa que en un problema típico se realizan millones de operaciones con coma (punto) flotante. Esto quiere decir que la acumulación de errores de redondeo puede ser desastrosa*». Para Higham esta afirmación, si bien cierta, no es del todo correcta o está mal enfocada. En muchas ocasiones la inestabilidad está dada por la incidencia de unos pocos errores de redondeo y no por la acumulación de millones de ellos. Un ejemplo en ese sentido está dado por el algoritmo del ejemplo inicial, en el cual el error está dado por el redondeo de y_{n-1} , que se propaga a medida que el valor es cada vez más chico. Otro ejemplo es el cálculo de e usando su definición:

$$f(n) = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n,$$

tomando n finito pero lo suficientemente grande. En la tabla 1.3 podemos ver los resultados para distintos n obtenidas en LibreOffice Calc.

Tabla 1.3: Valores de $f(n)$ y diferencia con e .

n	$f(n)$	$ e - f(n) $
10^1	2,593742460100000	$1,24539 \times 10^{-1}$
10^2	2,704813829421530	$1,34680 \times 10^{-2}$
10^3	2,716923932235590	$1,35790 \times 10^{-3}$
10^4	2,718145926824930	$1,35902 \times 10^{-4}$
10^5	2,718268237192300	$1,35913 \times 10^{-5}$
10^6	2,718280469095750	$1,35936 \times 10^{-6}$
10^7	2,718281694132080	$1,34327 \times 10^{-7}$
10^8	2,718281798347360	$3,01117 \times 10^{-8}$
10^9	2,718282052011560	$2,23553 \times 10^{-7}$
10^{10}	2,718282053234790	$2,24776 \times 10^{-7}$
10^{11}	2,718282053357110	$2,24898 \times 10^{-7}$
10^{12}	2,718523496037240	$2,41668 \times 10^{-4}$
10^{13}	2,716110034086900	$2,17179 \times 10^{-3}$
10^{14}	2,716110034087020	$2,17179 \times 10^{-3}$
10^{15}	3,035035206549260	$3,16753 \times 10^{-1}$

Como podemos observar, a medida que n aumenta, mejora la aproximación de e . Sin embargo, eso ocurre sólo para $n < 10^8$. Cuando $n \geq 10^9$ la aproximación se vuelve cada vez peor, como es el caso de $n = 10^{15}$. Al igual que en el ejemplo ya citado, el problema es la imposibilidad de representar correctamente $\frac{1}{n}$ cuando n es muy grande y, en consecuencia, un solo error de redondeo incide negativamente en el resultado obtenido.

1.9.3. Aumento de la precisión

El caso anterior muestra también que el aumento de la precisión no siempre significa una mejora en los resultados obtenidos. Es usual que cuando la única fuente de error es el redondeo, la forma tradicional de corregir esto es aumentar la precisión y ver qué ocurre con los resultados, comparando cuántos dígitos coinciden en los resultados original y con mayor precisión.

Pero en el caso de trabajar con un problema mal condicionado, el aumento de la precisión no resulta en una mejora en los resultados. En ese caso, es muy posible que los resultados

obtenidos no tengan ningún dígito en común. Un ejemplo típico es el siguiente. Supongamos que resolvemos el siguiente sistema de ecuaciones lineales:

$$\begin{bmatrix} 10^{-4} & 2 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \end{bmatrix}.$$

Si utilizamos dos precisiones diferentes para resolver el sistema, una con cuatro decimales y otra con tres, obtenemos los siguientes vectores $[x]$:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_1 = \begin{bmatrix} 0,01 \\ 2 \end{bmatrix} \text{ con tres decimales,} \quad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_2 = \begin{bmatrix} 1,0001 \\ 2 \end{bmatrix} \text{ con cuatro decimales.}$$

Vemos que el aumento de la precisión nos da un resultado completamente distinto para la primera componente y por consiguiente, no son comparables. Este es un típico caso de una matriz considerada como «mal condicionada» y que debemos transformarla para obtener resultados mejores. Así, si intercambiamos filas tenemos:

$$\begin{bmatrix} 1 & 1 \\ 10^{-4} & 2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \end{bmatrix},$$

la solución que obtenemos es:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1,0 \\ 2,0 \end{bmatrix}$$

cualquiera sea la precisión utilizada y que corresponde a la solución correcta.

Es evidente que el aumento en la precisión de los coeficientes no mejora los resultados. Este es un caso especial de matrices cuya solución merece un estudio más detallado que se verá en el capítulo 3 *Sistemas de Ecuaciones Lineales y No Lineales*.

1.10. Diseño de algoritmos estables

El análisis de los errores y, fundamentalmente, de la propagación de estos errores, nos ayuda a obtener algunos lineamientos para diseñar algoritmos estables, si bien no hay «recetas» simples para ello. La mejor recomendación es estar alerta en obtener un algoritmo estable cuando se lo diseña y no concentrarse solamente en otras cuestiones, como el costo computacional o la posibilidad de su «paralelización».

En su libro, Higham da una serie de lineamientos, entre los cuales se destacan los siguientes:

1. Evitar la resta de cantidades con errores.
2. Minimizar el tamaño de las cantidades intermedias relativas al resultado final. La razón es que si las cantidades intermedias son demasiado grandes, el resultado final puede ser consecuencia de una resta dañina. O visto de otra manera, cantidades grandes «tapan» los datos iniciales y en consecuencia, se pierde información.
3. Es más ventajoso escribir una expresión que actualice la información como

$$\text{valor}_{\text{nuevo}} = \text{valor}_{\text{viejo}} + \text{pequeña corrección}$$

si la pequeña corrección se puede calcular con muchos dígitos significativos.⁷ Muchos de los métodos numéricos se expresan de esta forma, como por ejemplo, el método de Newton-Raphson, el Método de los Gradientes Conjugados para resolver sistemas de ecuaciones

⁷Sin embargo, Higham mismo reconoce que no es necesario operar con muchos dígitos significativos para obtener buenos resultados utilizando este procedimiento. Véase [11]

lineales, etc. Un ejemplo clásico es el método del refinamiento iterativo de la solución para un sistema de ecuaciones lineales de la forma $Ax = B$, en el que se calcula el residuo $r_1 = B - A\tilde{x}_1$, y con él un valor δ_1 resolviendo $A\delta_1 = r_1$, para luego mejorar el resultado obtenido con la iteración $\tilde{x}_2 = \tilde{x}_1 + \delta_1$.

4. Usar transformaciones bien condicionadas.

Una recomendación importante es que se revisen los resultados intermedios, es decir, los que se generan durante el procedimiento de cálculo. Esta práctica era muy común en los inicios de la computación electrónica. En su libro, Higham señala lo siguiente:

Wilkinson, el padre del análisis de la propagación de errores, ganó una gran experiencia respecto a la estabilidad numérica gracias a ese tipo de revisión. Es irónico que con las grandes facilidades que se tienen hoy para rastrear los pasos de un algoritmo (ventanas múltiples, herramientas gráficas, impresoras rápidas), a veces se obtengan menos resultados que en esa época en las cuales sólo se contaba con papel y lámparas (válvulas).

Ejercicios

Errores - Error inherente

- Identifique y describa las principales fuentes de error.
- Desarrolle la propagación de los errores inherentes de las siguientes expresiones:

$$\begin{array}{llll} a) & y = a + b & b) & y = a - b & c) & y = a \cdot b & d) & y = \frac{a}{b} \\ e) & y = a^2 & f) & y = \sqrt{a} & g) & y = \frac{1}{a} \end{array}$$

- Obtenga los valores indicados utilizando las expresiones dadas y compare el resultado obtenido según se indica:

- a) Calcule $f(2)$ y compare el resultado con $\sqrt{2}$, aplicando la siguiente función:

$$f(1+x) = 1 + \frac{x}{2!} - \frac{x^2}{4 \cdot 2!} + \frac{3 \cdot x^3}{16 \cdot 4!} - \frac{7 \cdot 5 \cdot 3 \cdot x^5}{32 \cdot 5!} - \frac{9 \cdot 7 \cdot 5 \cdot 3 \cdot x^6}{64 \cdot 6!}.$$

- b) Calcule $f\left(\frac{\pi}{10}\right)$ y compare con $\sin\left(\frac{\pi}{10}\right)$, aplicando la siguiente función:

$$f(x) = x - \frac{x^3}{3!}$$

- c) Con la misma función del punto anterior obtenga $f\left(\frac{\pi}{6}\right)$ y $f\left(\frac{\pi}{3}\right)$ y compare con $\sin\left(\frac{\pi}{6}\right)$ y $\sin\left(\frac{\pi}{3}\right)$. Calcule los errores relativos.

- Dada la siguiente función:

$$f(x) = x - \frac{x^2}{2} + \frac{x^3}{3} + \dots + (-1)^{n+1} \frac{x^n}{n},$$

que aproxima la función $\ln(1+x)$, ¿cuántos términos se deben considerar si se quiere aproximar $\ln(2)$ con $e_R < 10^{-4}$?

- Hallar una cota del error de:

$$f(x, y, z) = \frac{x \cdot y^2}{\sqrt{z}}$$

si $x = 2,0 \pm 0,1$; $y = 3,1 \pm 0,2$ y $z = 1,0 \pm 0,1$.

Representación numérica

1. Represente los resultados de las siguientes expresiones con cuatro decimales:

$$a) \sqrt{2} \quad b) 2,1445 \cdot \pi \quad c) e^2 - \pi \quad d) \sqrt[3]{3}.$$

En lo casos *b)* y *c)*, primero represente los valores de π y e con cuatro decimales.

2. dada la ecuación de segundo grado, $a x^2 + b x + c = 0$, con $a = 1$; $b = 62,10$ y $c = 1$. Si representa todos los números y los resultados intermedios y finales con notación de coma flotante con cuatro decimales y redondeo, aplique los algoritmos indicados y compare los resultados obtenidos:

- a) Algoritmo 1:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \wedge \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

- b) Algoritmo 2:

$$x_1 = \frac{-2c}{-b + \sqrt{b^2 - 4ac}} \quad \wedge \quad x_2 = \frac{-2c}{-b - \sqrt{b^2 - 4ac}}.$$

3. La función

$$P(x) = x - \frac{x^3}{3} + \frac{x^5}{5},$$

corresponde a los tres primeros términos no nulos de la serie de MacLaurin para la función $\arctg x$. Calcule las siguientes expresiones usando $P(x)$ y solamente cinco decimales, y obtenga los errores absoluto y relativo, respecto de los valores obtenidos mediante las funciones incluidas en un programa como el MathCad, SMath Studio, Octave, MatLab, NumPy o similar.

$$a) 4 \left[\arctg \left(\frac{1}{2} \right) + \arctg \left(\frac{1}{3} \right) \right] \quad b) 16 \cdot \arctg \left(\frac{1}{2} \right) - 4 \cdot \arctg \left(\frac{1}{239} \right).$$

4. La Fórmula de Hudson es una de las expresiones utilizadas para el diseño y dimensionamiento de escolleras de talud tendido:

$$\frac{H_S}{\Delta \cdot D_n} = (K_D \cdot \cot \alpha)^{\frac{1}{3}},$$

donde H_S es la altura de ola significativa, $\Delta = \frac{\rho_M}{\rho_a} - 1$, ρ_M el peso específico del material de la coraza, ρ_a el peso específico del agua, D_n el diámetro o dimensión equivalente del bloque a utilizar en la coraza, K_D un coeficiente de forma y colocación y $\cot \alpha$ ($\cot \alpha = \frac{1}{\tan \alpha}$) la pendiente del talud. Si los valores datos son $\rho_M = 23,5 \text{ kN/m}^3$, $\rho_a = 10,25 \text{ kN/m}^3$, $H_S = 6,0 \text{ m}$, $K_D = 9$ y $\cot \alpha = 2$, obtenga un valor de D_n si el error en la medición de la altura de ola significativa (H_S) es de $\pm 0,5 \text{ m}$ y todas las operaciones deben hacerse considerando solamente dos decimales.

Gráfica de proceso

1. Para la función del ejercicio 5, obtenga el C_p y el T_e mediante la aplicación de la gráfica de proceso.
2. Dada la fórmula del área de un anillo circular, $\pi(R^2 - r^2)$, desarrolle la gráfica de proceso de los siguientes algoritmos:

a) Algoritmo 1: $\pi \cdot (R \cdot R - r \cdot r)$;

b) Algoritmo 2: $\pi \cdot (R - r) \cdot (R + r)$.

Asuma $\pi = 3,14159$ e indique cual de los dos algoritmo es el más conveniente.

Perturbaciones experimentales

1. Mediante la aplicación de las perturbaciones experimentales, obtenga el C_p de:

a) $f(x) = \frac{1}{2} \left(x + \frac{5}{x} \right)$, con $x = 2,236$ y $\Delta x = 0,001$;

b) $A = \pi(R^2 - r^2)$, con $R = 20$, $r = 10$, $\Delta R = 0,2$ y $\Delta r = 0,1$;

c) $D_n = \frac{H_S}{\Delta \cdot (K_D \cdot \cot \alpha)^{\frac{1}{3}}}$.

Use los datos del ejercicio 4 de Representación numérica y considere una perturbación del 1% para H_S y ρ_M .

2. La integral

$$y_n = \int_0^1 x^n \cdot e^{-x} dx,$$

con $n = 1; 2; 3; \dots$, se puede aproximar con el siguiente algoritmo:

$$y_n = n \cdot y_{n-1} - \frac{1}{e}.$$

Analice la condición y la estabilidad del algoritmo para el caso de $n = 3$, si el valor inicial es $y_0 = 1 - \frac{1}{e}$.

Capítulo 2

Ecuaciones no Lineales

2.1. Introducción

Al aprender matemática estamos acostumbrados a que el planteo para resolver un problema matemático está directamente relacionado con un procedimiento de tipo algebraico que nos facilita encontrar la solución en forma directa. Basta con reemplazar los datos en ese procedimiento para encontrar el resultado buscado. Un ejemplo es la solución algebraica de las raíces de la ecuación de segundo grado $ax^2 + bx + c = 0$ dada por:

$$x_{1;2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Pero no siempre es así.

Tomemos el siguiente caso: supongamos que queremos desarrollar una mejora en la costa y para ello necesitamos un recinto cerrado, el cual vamos a rellenar con arena. Para conseguir ese recinto necesitamos una pared de contención que construiremos con tablestacas. Para diseñar las tablestacas debemos resolver una ecuación del tipo $a_0 + a_1 x + a_2 x^2 + a_3 x^3 = 0$, donde x es la longitud de hinca, también conocida como «ficha». Esta ecuación tiene tres soluciones posibles (tres raíces). Si bien existe una solución algebraica para obtener las raíces de una ecuación de tercer grado, en general, es mucho más práctico resolverla mediante algún método iterativo, y obtener aquella solución (raíz) que sea compatible con el problema.

Existen también aquellas ecuaciones que no tienen solución algebraica y que, por lo tanto, sólo podrán resolverse mediante aproximaciones. Tenemos como ejemplo, el cálculo de la longitud de onda de una ola marítima en aguas intermedias (entre aguas poco profundas, cerca de la costa, y aguas profundas). La expresión para esto es:

$$L = L_0 \tanh\left(\frac{2\pi}{L}d\right),$$

donde L_0 es la longitud de onda en aguas profundas ($d \geq \frac{L}{2}$) y d es la profundidad del mar.

Esta expresión es válida para $\frac{L}{20} \leq d \leq \frac{L}{2}$. Como podemos ver, esta ecuación no tiene solución algebraica, y, en consecuencia, el único modo de obtenerla es mediante un método iterativo. (En realidad, cuando $d > \frac{L}{2}$, $\tanh\left(\frac{2\pi}{L}d\right) \cong 1$, y $L = L_0$, y cuando $d < \frac{L}{20}$, $\tanh\left(\frac{2\pi}{L}d\right) \cong \frac{2\pi}{L}d$ y por lo tanto $L \cong \sqrt{2\pi L_0 d}$.)

Dado que este tipo de problemas son regularmente comunes en la ingeniería, en este capítulo nos ocuparemos de estudiar los distintos métodos para resolver ecuaciones no lineales, de manera de obtener resultados muy precisos.

Como repaso, recordemos los teorema del valor medio y del valor intermedio.

Teorema 2.1. (*Teorema del valor medio.*) Si $f \in C[a, b]$ y f es diferenciable en (a, b) , entonces existirá un número c en (a, b) tal que

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

Teorema 2.2. (*Teorema del valor intermedio.*) Si $f \in C[a, b]$ y M es un número cualquiera entre $f(a)$ y $f(b)$, existirá un número c en (a, b) para el cual $f(c) = M$.

Del primero podemos considerar que en un intervalo (a, b) podemos obtener la pendiente «media» a partir de esos dos valores (a, b) . El segundo nos ayuda a encontrar la solución pues si $f(a) < 0$ y $f(b) > 0$ entonces existirá un valor x_p tal que $f(x_p) = 0$. Lo mismo ocurre si $f(a) > 0$ y $f(b) < 0$.

2.2. Método de la bisección

Supongamos una función cualquiera $f(x)$ y hallemos el valor de \bar{x} , tal que $f(\bar{x}) = 0$. Asumamos que \bar{x} está incluido en el intervalo (a, b) , con $b > a$. Para que esto sea cierto, generalmente se verifica que $f(a) \cdot f(b) < 0$. (Sin embargo, hay casos en que $f(\bar{x}) = 0$, $\bar{x} \in (a, b)$ y no se cumple que $f(a) \cdot f(b) < 0$.)

Puesto que $\bar{x} \in (a, b)$, calculemos nuestra primera aproximación de \bar{x} tomando el valor medio del intervalo, es decir,

$$x_1 = a + \frac{b - a}{2} = \frac{b + a}{2}. \quad (2.1)$$

Para saber si es o no solución debemos verificar que $f(x_1) = 0$. Si no lo es, debemos volver a obtener una aproximación mediante un esquema similar. Para ello, verifiquemos que $f(a) \cdot f(x_1) < 0$. Si es cierto, entonces nuestro nuevo intervalo será (a, x_1) , si no lo es, nuestro intervalo será (x_1, b) . Supongamos, por simplicidad, que $f(a) \cdot f(x_1) < 0$ y que nuestro nuevo intervalo es (a, x_1) . Con el mismo método usado antes, nuestra nueva aproximación es

$$x_2 = \frac{x_1 + a}{2} = \frac{\frac{b+a}{2} + a}{2} = \frac{b + a}{4} + \frac{a}{2}, \quad (2.2)$$

que también puede escribirse como

$$x_2 = x_1 - \frac{b - a}{4}. \quad (2.3)$$

Nuevamente verificamos si $f(x_2) = 0$. Si seguimos iterando hasta obtener x_n , tenemos que

$$x_n = x_{n-1} - \frac{b - a}{2^n}, \quad (2.4)$$

por lo que también podemos decir que

$$|\bar{x} - x_n| \leq \frac{b - a}{2^n}. \quad (2.5)$$

Las figuras 2.1.(a), 2.1.(b) y 2.1.(c) muestran el proceso de aproximación del método, reduciendo el intervalo.

Si queremos hallar el valor exacto de \bar{x} deberíamos iterar hasta que $|\bar{x} - x_n| = 0$. Pero si establecemos una tolerancia de modo que $|\bar{x} - x_n| < \varepsilon$, entonces tendremos que la cantidad aproximada de iteraciones para obtener este resultado es

$$|\bar{x} - x_n| \leq \frac{b - a}{2^n} < \varepsilon \quad (2.6)$$

$$\frac{b - a}{2^n} < \varepsilon \quad (2.7)$$

$$n > \frac{\ln\left(\frac{b-a}{\varepsilon}\right)}{\ln(2)}. \quad (2.8)$$

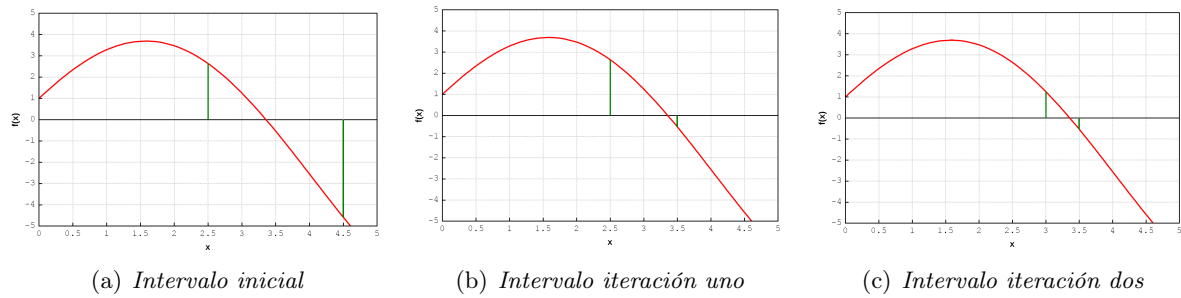


Figura 2.1: Aproximaciones de la raíz por el método de la bisección.

Por ejemplo, si nuestra tolerancia la expresamos como $\varepsilon = 10^{-t}$, para que el método converja se necesitarán $n > \frac{\ln(b-a) - \ln(10^{-t})}{\ln(2)}$ iteraciones. Si lo desarrollamos un poco más, tenemos:

$$n > \frac{\ln(b-a)}{\ln(2)} + t \cdot \frac{\ln(10)}{\ln(2)} \tag{2.9}$$

$$n > \frac{1}{\ln(2)} [\ln(b-a) + t \cdot \ln(10)], \tag{2.10}$$

y podemos escribir que

$$n > \frac{1}{0,69} [\ln(b-a) + t \cdot 2,30] \tag{2.11}$$

$$n > 1,44 \cdot \ln(b-a) + 3,32 \cdot t, \tag{2.12}$$

con lo cual la cantidad de iteraciones depende mucho más de la tolerancia que del intervalo.

Si en vez del error absoluto usamos el error relativo como criterio de interrupción, la cantidad de iteraciones queda:

$$\frac{|\bar{x} - x_n|}{|\bar{x}|} \leq \frac{b-a}{2^n \cdot |\bar{x}|} < \varepsilon \tag{2.13}$$

$$\frac{b-a}{2^n \cdot |\bar{x}|} < \varepsilon \tag{2.14}$$

$$n > \frac{\ln\left(\frac{b-a}{\varepsilon \cdot |\bar{x}|}\right)}{\ln(2)}. \tag{2.15}$$

Si consideramos que $\bar{x} \approx \frac{b-a}{2}$, entonces nos queda:

$$n > \frac{\ln\left(\frac{b-a}{\varepsilon \cdot \frac{b-a}{2}}\right)}{\ln(2)}, \tag{2.16}$$

$$n > \frac{\ln\left(\frac{2}{\varepsilon}\right)}{\ln(2)}, \tag{2.17}$$

si

$$\varepsilon = 10^{-t}, \tag{2.18}$$

entonces, al reemplazar en la ecuación anterior queda

$$n > \frac{\ln\left(\frac{2}{10^{-t}}\right)}{\ln(2)} = \frac{\ln(2) + t \cdot \ln(10)}{\ln(2)}, \quad (2.19)$$

$$n > 1 + t \cdot \frac{\ln(10)}{\ln(2)}. \quad (2.20)$$

es decir, incide poco el intervalo, siempre que el mismo sea pequeño o sus extremos no estén muy alejados del valor exacto.

Este método de aproximación de las soluciones se conoce como *Método de la Bisección*. Es muy sencillo y tiene la ventaja de que siempre converge, pues nada exige a la función a la cual se le quiere calcular la raíz, salvo que se cumpla que $f(x_{k-1}) \cdot f(x_k) < 0$, mientras se está iterando. Hemos estimado cuantas iteraciones son necesarias para encontrar una solución aceptable a partir del error absoluto, pero en realidad los criterios para detener el procedimiento pueden ser los siguientes:

$$|x_n - x_{n-1}| \leq \varepsilon, \quad (2.21)$$

$$\frac{|x_n - x_{n-1}|}{|x_n|} \leq \varepsilon, \quad (2.22)$$

$$|f(x_n)| \leq \varepsilon. \quad (2.23)$$

donde ε es la tolerancia que ya mencionamos.

Cualquiera de los tres criterios es bueno para detener el proceso, pero el segundo es el más efectivo, pues se basa en el error relativo, y nos da una idea aproximada de la cantidad de cifras o dígitos significativos que tiene el resultado obtenido, pero a costa de hacer todavía mucho más lenta la convergencia, por lo ya visto al estimar n . En cambio, el último es el menos confiable, pues por definición $f(x_n)$ tiende a cero, con lo cual siempre es pequeño.

Si bien no tiene problemas con la convergencia, hemos visto que el método resulta muy lento para alcanzar un resultado aceptable. Además, según sea el criterio de interrupción aplicado, en muchas ocasiones puede desprestigiar un resultado intermedio más preciso. Es por eso que no suele utilizarse como único método para alcanzar la solución.

2.3. Método de la falsa posición o «regula falsi»

Hay otro método que también se basa en ir «achicando» el intervalo en el que se encuentra la solución. Se trata del *Método de la Falsa Posición* o «Regula Falsi». Consiste en trazar la cuerda que une los puntos $f(a)$ y $f(b)$ de la función dada e ir reduciendo el intervalo hasta obtener el valor de \bar{x} tal que $f(\bar{x}) = 0$. Al igual que para el método de la bisección, debemos empezar por calcular x_1 . Existen dos métodos equivalentes para obtenerlo:

$$x_1 = a - \frac{f(a)(b-a)}{f(b)-f(a)}, \text{ o} \quad (2.24)$$

$$x_1 = b - \frac{f(b)(b-a)}{f(b)-f(a)}. \quad (2.25)$$

La forma de aplicar el método es la siguiente. Utilicemos la primera expresión para obtener x_1 ; entonces verifiquemos que $f(x_1) \cdot f(a) < 0$. Si esto es cierto, para obtener nuestra segunda aproximación tendremos la siguiente expresión:

$$x_2 = a - \frac{f(a)(x_1-a)}{f(x_1)-f(a)}, \quad (2.26)$$

caso contrario, nos queda esta otra expresión:

$$x_2 = x_1 - \frac{f(x_1)(b - x_1)}{f(b) - f(x_1)}. \quad (2.27)$$

Algo similar ocurre si partimos de la segunda. Si $f(x_1) \cdot f(b) < 0$, nos queda

$$x_2 = b - \frac{f(b)(b - x_1)}{f(b) - f(x_1)}, \quad (2.28)$$

y si no

$$x_2 = x_1 - \frac{f(x_1)(x_1 - a)}{f(x_1) - f(a)}. \quad (2.29)$$

Este método no es una gran mejora al método de la bisección, aunque al trazar las cuerdas, hace uso de la función y generalmente suele converger un poco más rápido (figuras 2.2.(a), 2.2.(b) y 2.2.(c)). Un punto importante a tener en cuenta es que al igual que en el método de la bisección, los sucesivos x_k se encuentran siempre en el intervalo de análisis (el intervalo (x_{k-2}, x_{k-1})) y, por lo tanto, en el intervalo (a, b) inicial. Y análogamente al método de la bisección, es posible que desprecie soluciones más precisas obtenidas en pasos intermedios.

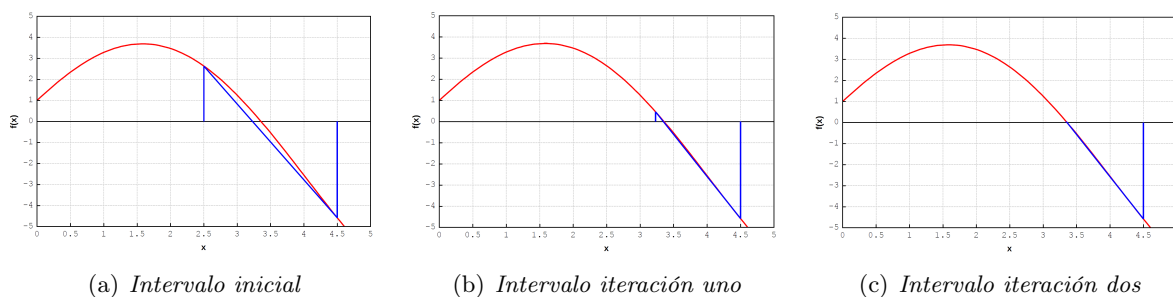


Figura 2.2: Aproximaciones de la raíz por el método de la «Regula Falsi».

2.4. Método de las aproximaciones sucesivas o punto fijo

Puesto que los métodos anteriores no tienen una convergencia rápida, no son muy prácticos para resolver problemas de gran complejidad. Veremos a continuación un método mucho más poderoso y efectivo.

Supongamos que nuestro problema a resolver, $f(x) = 0$, lo escribimos de una manera levemente diferente:

$$f(x) = x - g(x) = 0. \quad (2.30)$$

Es evidente que podemos despejar x de esta ecuación sin problemas, por lo que finalmente nos queda:

$$x = g(x), \quad (2.31)$$

es decir, nuestro problema se resume a encontrar una función $g(x)$. Como estamos resolviéndolo en forma iterativa, la expresión que nos queda es:

$$x_{k+1} = g(x_k), \quad (2.32)$$

para $k = 0; 1; \dots; n$. El esquema entonces es sencillo: partiendo de una solución inicial, por ejemplo x_0 , luego de efectuar n iteraciones tendremos nuestra solución aproximada x_n , que estará mucho más cerca del resultado «exacto» que nuestro valor inicial.

Veamos entonces un ejemplo de cómo aplicar el método. Supongamos que nuestra función es

$$-e^{-0,1 \cdot x} + \ln(x) = 0.$$

No hay un procedimiento algebraico para obtener la raíz de esta función. Para ello, proponemos la siguiente función $G_1(x)$ en el intervalo $(1,5; 2,5)$:

$$G_1(x) = x - e^{-0,1 \cdot x} + \ln(x),$$

y resolvamos en forma iterativa tomando $x_0 = 2$. Calculemos $G_1(x_0)$ y obtengamos x_1 , luego x_2 y así sucesivamente:

$$\begin{aligned} x_1 &= G_1(x_0) = 2 - e^{-0,1 \cdot 2} + \ln(2) = 1,874 \\ x_2 &= G_1(x_1) = 1,874 - e^{-0,1 \cdot 1,874} + \ln(1,874) = 1,674 \\ x_3 &= G_1(x_2) = 1,674 - e^{-0,1 \cdot 1,674} + \ln(1,674) = 1,343. \end{aligned}$$

La figura 2.3 muestra la progresión de las iteraciones.

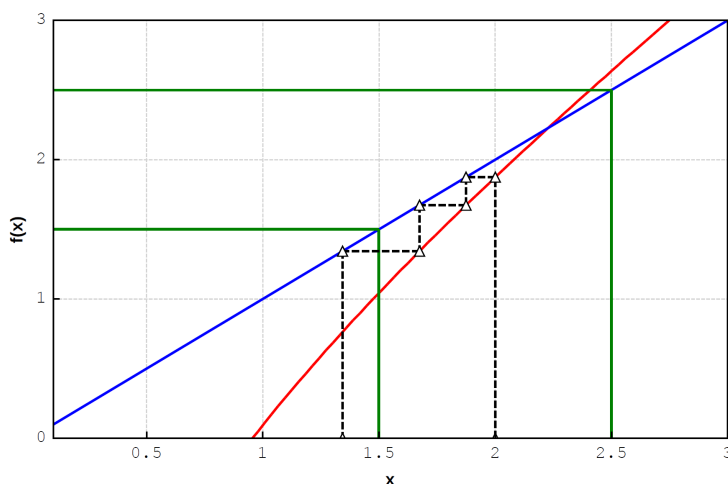


Figura 2.3: Método de las aproximaciones sucesivas con la función $G_1(x)$.

Es fácil notar que tenemos un problema. Como dijimos, la raíz buscada se encuentra en el intervalo $(1,5; 2,5)$, pero el último resultado nos dio fuera de dicho intervalo. Evidentemente, esta función $G_1(x)$ no nos sirve.

Cambiamos la función y volvamos a intentarlo. Probemos con la siguiente función:

$$G_2(x) = e^{e^{-0,1 \cdot x}}.$$

Esta vez iniciemos el proceso con $x_0 = 1,5$. Si repetimos lo hecho con $G_1(x)$, obtenemos:

$$\begin{aligned} x_1 &= G_2(x_0) = e^{e^{-0,1 \cdot 1,5}} = 2,365; \\ x_2 &= G_2(x_1) = e^{e^{-0,1 \cdot 2,365}} = 2,202; \\ x_3 &= G_2(x_2) = e^{e^{-0,1 \cdot 2,202}} = 2,231. \end{aligned}$$

A la tercer iteración obtenemos el siguiente resultado: $x_3 = 2,231$, valor que está dentro del intervalo. Verifiquemos si este valor es «correcto» calculando $f(x_3)$:

$$f(2,231) = -e^{-0,1 \cdot 2,231} + \ln(2,231) = 0,0023.$$

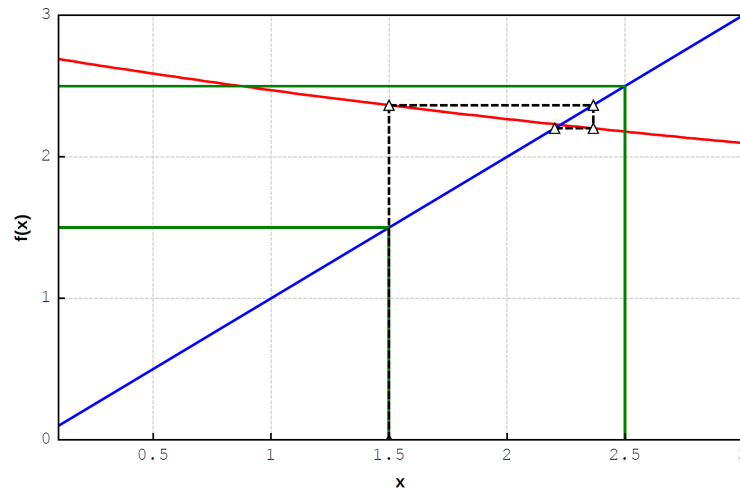


Figura 2.4: Método de las aproximaciones sucesivas con la función $G_2(x)$.

La figura 2.4 muestra la progresión de las iteraciones para esta última función.

Este valor puede considerarse cercano cero y por lo tanto, hemos podido encontrar la raíz buscada. (La raíz de esta función es $x = 2,226$.)

¿Pero por qué fallamos al usar la primera función? Para entender esto veamos los siguientes teoremas.

Teorema 2.3. Si $g(x) \in C(a; b)$ y $g(x) \in [a, b]$ para todo $x \in [a, b]$, entonces $g(x)$ tiene un punto fijo en $[a, b]$.

Teorema 2.4. Si $g'(x)$ existe en $[a, b]$, y existe una constante $m < 1$, tal que

$$|g'(x)| \leq m, \text{ para toda } x \in [a, b],$$

entonces, el punto fijo en $[a, b]$ es único.

La demostración de estos teoremas puede verse en [3].

Estos teoremas son *suficientes* pero no necesarios, es decir, pueden no cumplirse y existir dicho punto, tal como vimos en el ejemplo anterior. La función $G_1(x)$ no cumple con los teoremas antes expuestos, sin embargo el punto fijo existe¹.

Si miramos la función $G_1(x)$ rápidamente notamos que el valor de la misma en 1,5 no pertenece al intervalo dado, pues $G_1(1,5) = 1,045$, por lo tanto, no se puede asegurar que exista un punto fijo. Y si hallamos la primera derivada en ese punto tenemos que $|G_1'(1,5)| = |-1,753| = 1,753 > 1$, con lo cual si existiera el punto fijo, no podríamos asegurar que dicho punto fijo sea único. No ocurre lo mismo con la función $G_2(x)$ puesto que $G_2(1,5) = 2,365$ que está incluido en el intervalo $[0,5; 1,5]$ y $|G_2'(1,5)| = |-0,074| = 0,074 < 1$, con lo cual el punto fijo es único. En realidad, deberíamos haber verificado ambas funciones $G(x)$ ($G_1(x)$ y $G_2(x)$) para varios puntos del intervalo, pero al comprobar que el punto de partida no cumple con las condiciones de ambos teoremas (función $G_1(x)$), nos indica que esta función no es convergente.

Verificado que la función $g(x)$ es convergente, nos falta definir el o los criterios de interrupción. Como en los casos anteriores, éstos son similares a los ya vistos, es decir,

$$|x_n - x_{n-1}| \leq \varepsilon, \quad (2.33)$$

$$\frac{|x_n - x_{n-1}|}{|x_n|} \leq \varepsilon, \quad (2.34)$$

$$|f(x_n)| \leq \varepsilon. \quad (2.35)$$

¹Para $x = 2,226$ se tiene $G_1(x) = 2,226$, por lo tanto, el punto fijo existe.

Con el mismo ejemplo tenemos una pregunta: ¿cómo podemos obtener una solución por aproximaciones sucesivas (o punto fijo) que tenga una convergencia rápida? Para ello tenemos el siguiente teorema.

Teorema 2.5. Sea $g(x) \in C[a, b]$ tal que $g(x) \in [a, b]$ para todo x en $[a, b]$, que existe $g'(x)$ en $[a, b]$ y que una constante $k < 1$ cuando

$$|g'(x)| \leq k, \text{ para toda } x \in (a, b).$$

Entonces, para cualquier número $x_0 \in [a, b]$, la sucesión definida por

$$x_n = g(x_{n-1}), \quad n \geq 1,$$

converge en el único punto fijo \bar{x} en $[a, b]$.

Demostración El teorema 2.5 implica que existe un punto fijo en $[a, b]$. Como $g(x)$ pertenece a $[a, b]$ para todo x que pertenece a $[a, b]$, la sucesión $\{x_n\}_{n=0}^{\infty}$ se define para todo $n \geq 0$ y $x_n \in [a, b]$ para todo n . Dado que $|g'(x)| \leq k$, si aplicamos el teorema del valor medio, tenemos

$$|x_n - \bar{x}| = |g(x_{n-1}) - g(\bar{x})| = |g'(\xi)| |x_{n-1} - \bar{x}| \leq k |x_{n-1} - \bar{x}|, \quad (2.36)$$

donde $\xi \in (a, b)$. En forma inductiva obtenemos

$$|x_n - \bar{x}| \leq k |x_{n-1} - \bar{x}| \leq k^2 |x_{n-2} - \bar{x}| \leq \dots \leq k^n |x_0 - \bar{x}|. \quad (2.37)$$

Como $k < 1$, entonces

$$\lim_{n \rightarrow \infty} |x_n - \bar{x}| \leq \lim_{n \rightarrow \infty} k^n |x_0 - \bar{x}| = 0, \quad (2.38)$$

y la sucesión $\{x_n\}_{n=0}^{\infty}$ converge a \bar{x} .

Corolario 2.5.1. Si $g(x)$ satisface las hipótesis de teorema 2.5, las cotas de error que supone utilizar x_n para aproximar \bar{x} están dadas por

$$|x_n - \bar{x}| \leq k^n \max\{x_0 - a, b - x_0\},$$

y por

$$|x_n - \bar{x}| \leq \frac{k^n}{1 - k} |x_1 - x_0|, \text{ para toda } n \geq 1.$$

Demostración La primera cota viene de:

$$|x_n - \bar{x}| \leq k^n |x_0 - \bar{x}| \leq k^n \max\{x_0 - a, b - x_0\}, \quad (2.39)$$

porque $x \in (a, b)$.

Con $n \geq 1$, la demostración del teorema 2.5 implica que

$$|x_{n+1} - \bar{x}| = |g(x_n) - g(x_{n-1})| \leq |x_n - x_{n-1}| \leq \dots \leq k^n |x_1 - x_0|. \quad (2.40)$$

En consecuencia, cuando $m > n \geq 1$,

$$|x_m - x_n| = |x_m - x_{m-1} + x_{m-1} - x_{m-2} + \dots + x_{n+1} - x_n| \quad (2.41)$$

$$\leq |x_m - x_{m-1}| + |x_{m-1} - x_{m-2}| + \dots + |x_{n+1} - x_n| \quad (2.42)$$

$$\leq k^{m-1} |x_1 - x_0| + k^{m-2} |x_1 - x_0| + \dots + k^n |x_1 - x_0| \quad (2.43)$$

$$= k^n (1 + k + k^2 + \dots + k^{m-n-1}) |x_1 - x_0|. \quad (2.44)$$

Por el mismo teorema, tenemos que $\lim_{n \rightarrow \infty} x_n = \bar{x}$, por lo tanto

$$|\bar{x} - x_n| = \lim_{n \rightarrow \infty} \leq k^n |x_1 - x_0| \sum_{i=0}^{\infty} k^i. \quad (2.45)$$

Pero $\sum_{i=0}^{\infty} k^i$ es una serie geométrica con razón k . Como $0 < k < 1$, esta sucesión converge a $\frac{1}{1-k}$, por lo que nos queda que

$$|\bar{x} - x_n| \leq \frac{k^n}{1-k} |x_1 - x_0|. \quad (2.46)$$

Podemos ver que como $|g'(x)| \leq k$, la convergencia depende de la primera derivada de $g(x)$. Cuanto más chico sea k , más rápida será convergencia.

2.5. Método de Newton-Raphson

Este método es uno de los más poderosos que se conocen para resolver ecuaciones de la forma $f(x) = 0$. Curiosamente, el método fue desarrollado tanto por Isaac Newton como por Joseph Raphson, pero fue éste quien lo publicó primero, casi 50 años antes y en una forma levemente diferente a la de Newton, quien sólo la desarrolló para resolver raíces de polinomios. Su formulación actual fue desarrollada en realidad por otro matemático, Thomas Simpson².

Una primera aproximación para obtenerlo es partir del método de la falsa posición, y en vez de trazar una cuerda entre los dos extremos del intervalo, trazar una tangente, que pase por un punto. Supongamos que para el mismo intervalo $[a, b]$ trazamos la tangente que pasa por $f(b)$. La ecuación de la recta tangente será

$$t(x) = f'(b)(x - b) + f(b). \quad (2.47)$$

Cuando se cumpla que $f(x) = 0$ se deberá cumplir que $t(x) = 0$. Por lo tanto podríamos hallar un valor x_1 tal que $t(x_1) = 0$ para ir aproximando nuestra raíz. Así obtenemos

$$t(x_1) = 0 = f'(b)(x_1 - b) + f(b) \quad (2.48)$$

$$x_1 = b - \frac{f(b)}{f'(b)}. \quad (2.49)$$

Si $f(x_1) \neq 0$, podemos repetir el procedimiento otra vez para obtener un x_2 . En definitiva, podemos crear una aproximación iterativa de la siguiente forma:

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}. \quad (2.50)$$

La figura 2.5 muestra la aproximación de la raíz por este método.

Existe una forma de deducirlo a través de la serie de Taylor. Supongamos que $f(x) \in C^2[a, b]$, y sea \hat{x} una aproximación de \bar{x} tal que $f(\hat{x}) = 0$. También que $f'(\hat{x}) \neq 0$ y $|\hat{x} - \bar{x}|$ sea pequeño. Desarrollemos el primer polinomio de Taylor para $f(\hat{x})$ expandida alrededor de x ,

$$f(x) = f(\hat{x}) + f'(\hat{x})(x - \hat{x}) + f''(\xi(x)) \frac{(x - \hat{x})^2}{2}, \quad (2.51)$$

donde $\xi(x)$ está entre x y \hat{x} . Puesto que $f(\bar{x}) = 0$, entonces para $x = \bar{x}$ tenemos

$$0 = f(\hat{x}) + f'(\hat{x})(\bar{x} - \hat{x}) + f''(\xi(\bar{x})) \frac{(\bar{x} - \hat{x})^2}{2}. \quad (2.52)$$

²Matemático inglés, nacido en 1710 y fallecido en 1761. Fue quien dio a conocer el famoso *Método de Simpson* de integración numérica. Lo curioso es que Simpson atribuyó este método a Newton. Más datos acerca de Thomas Simpson se pueden obtener en [14].

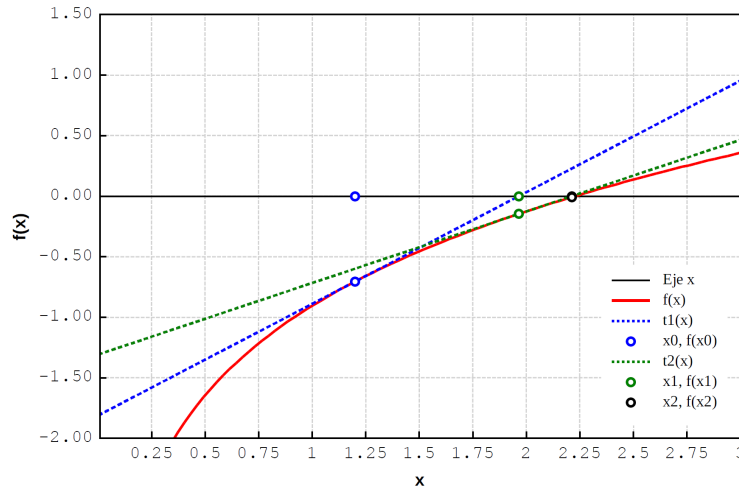


Figura 2.5: Método de Newton-Raphson.

Al suponer que $|\hat{x} - \bar{x}|$ es pequeño, podemos despreciar $(\bar{x} - \hat{x})^2$, con lo que nos queda

$$0 = f(\hat{x}) + f'(\hat{x})(\bar{x} - \hat{x}), \quad (2.53)$$

y despejando \bar{x} de la ecuación nos queda

$$\bar{x} = \hat{x} - \frac{f(\hat{x})}{f'(\hat{x})}. \quad (2.54)$$

Y si en lugar de aproximar con \hat{x} lo hacemos con x_0 , entonces generamos una sucesión $\{x_i\}$ definida por

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}, \quad (2.55)$$

que es la misma expresión que ya vimos.

De este desarrollo podemos ver que el error cometido es proporcional a $(\bar{x} - x_i)^2$ o a $f''(x_i)$ (puesto que cuando $x_i \approx \bar{x}$ podemos suponer que $\xi(x_i) \approx x_i$). De ahí que podemos aplicar los mismos criterios de interrupción que en los otros métodos.

También podemos observar que si no elegimos un x_0 lo suficientemente cerca, el método puede no converger. Para esto tenemos el siguiente teorema.

Teorema 2.6. Sea $f \in C^2[a, b]$; si $\bar{x} \in [a, b]$ es tal que $f(\bar{x}) = 0$ y $f'(\bar{x}) \neq 0$, entonces existe un $\delta > 0$ tal que el método de Newton-Raphson genera una sucesión $\{x_i\}_{i=1}^{\infty}$ que converge a \bar{x} para cualquier aproximación inicial $x_0 \in [\bar{x} - \delta, \bar{x} + \delta]$.

Demostración La demostración se basa en analizar el método de Newton-Raphson como si fuera el método de las aproximaciones sucesivas, tomando que $x_i = g(x_{i-1})$, $n \geq 1$, y que

$$g(x) = x - \frac{f(x)}{f'(x)}.$$

Entonces, sea k un número cualquiera en $(0, 1)$. En primer lugar debemos encontrar un intervalo $[\bar{x} - \delta, \bar{x} + \delta]$ que g «mapee» en sí mismo y en el que $|g'(x)| \leq k$ para toda $x \in [\bar{x} - \delta, \bar{x} + \delta]$.

Como $f'(\bar{x}) \neq 0$ y $f'(\bar{x})$ es continua, existe $\delta_1 > 0$ tal que $f'(x) \neq 0$ para $x \in [\bar{x} - \delta_1, \bar{x} + \delta_1] \subset [a, b]$. Por lo tanto, g está definida y es continua en $[\bar{x} - \delta_1, \bar{x} + \delta_1]$. Por otro lado tenemos que

$$g'(x) = 1 - \frac{f'(x)f'(x) - f(x)f''(x)}{[f'(x)]^2} = \frac{f(x)f''(x)}{[f'(x)]^2},$$

para $x \in [\bar{x} - \delta_1, \bar{x} + \delta_1]$ y como $f \in C^2[a, b]$, tendremos que $g \in C^1[a, b]$.

Como hemos supuesto que $f(\bar{x}) = 0$, entonces

$$g'(\bar{x}) = \frac{f(\bar{x})f''(\bar{x})}{[f'(\bar{x})]^2} = 0.$$

Además g' es continua y k es tal que $0 < k < 1$, entonces existe un δ , tal que $0 < \delta < \delta_1$, y

$$|g'(x)| \leq k \text{ para toda } x \in [\bar{x} - \delta, \bar{x} + \delta].$$

Nos falta todavía demostrar que $g : [\bar{x} - \delta, \bar{x} + \delta] \rightarrow [\bar{x} - \delta, \bar{x} + \delta]$. Si $x \in [\bar{x} - \delta, \bar{x} + \delta]$. El teorema del valor medio implica que existe un número ξ entre x y \bar{x} para el que se cumple

$$|g(x) - g(\bar{x})| = |g'(\xi)| |x - \bar{x}|.$$

Por lo tanto, se cumple que

$$|g(x) - \bar{x}| = |g(x) - g(\bar{x})| = |g'(\xi)| |x - \bar{x}| \leq k |x - \bar{x}| < |x - \bar{x}|.$$

Como $x \in [\bar{x} - \delta, \bar{x} + \delta]$, podemos deducir que $|x - \bar{x}| < \delta$ y que $|g(x) - \bar{x}| < \delta$. Este último resultado nos muestra que $g : [\bar{x} - \delta, \bar{x} + \delta] \rightarrow [\bar{x} - \delta, \bar{x} + \delta]$.

En consecuencia, la función $g(x) = x - f(x)/f'(x)$ satisface todas las hipótesis del teorema 2.5, de modo que la sucesión $\{x_i\}_{i=1}^{\infty}$ definida por

$$x_i = g(x_{i-1}) = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}, \text{ para } i \geq 1,$$

converge a \bar{x} para cualquier $x_0 \in [\bar{x} - \delta, \bar{x} + \delta]$.

Como vimos, este método es una variante del método de las aproximaciones sucesivas. Si la función $f(x)$ no tiene derivada en el entorno $[a, b]$ no es posible aplicarlo, pero si resulta difícil calcularla o evaluarla, existe un método alternativo denominado *método de la secante*, el cual reemplaza $f'(x_{i-1})$ por su aproximación discreta, es decir,

$$f'(x_{i-1}) = \frac{f(x_{i-1}) - f(x_{i-2})}{x_{i-1} - x_{i-2}}. \quad (2.56)$$

Si reemplazamos esto último en la fórmula de Newton-Raphson tenemos

$$x_i = x_{i-1} - \frac{f(x_{i-1})(x_{i-1} - x_{i-2})}{f(x_{i-1}) - f(x_{i-2})}, \quad (2.57)$$

que también podemos escribir como

$$x_i = \frac{f(x_{i-1})x_{i-2} - f(x_{i-2})x_{i-1}}{f(x_{i-1}) - f(x_{i-2})}. \quad (2.58)$$

2.6. Análisis del error

En este punto analizaremos la convergencia de los métodos iterativos vistos. Nos basaremos en la siguiente definición.

Definición 2.1. Una sucesión $\{x_i\}_{i=0}^{\infty}$ convergirá a \bar{x} de orden α con una constante asintótica λ si se cumple que

$$\lim_{i \rightarrow \infty} \frac{|x_{i+1} - \bar{x}|}{|x_i - \bar{x}|^\alpha} = \lambda,$$

con $x_i \neq \bar{x}$ para todo i , y α y λ son dos constantes positivas.

En consecuencia, tenemos que la convergencia puede ser *lineal* ($\alpha = 1$), *cuadrática* ($\alpha = 2$), *cúbica* ($\alpha = 3$), etc. Dado que obtener un procedimiento con convergencia mayor a la cuadrática no es sencillo, nos ocuparemos de analizar solamente los dos primeros casos.

Enunciaremos dos teoremas que se refieren a la convergencia lineal y a la cuadrática, que están basados en el método de las aproximaciones sucesivas.

Teorema 2.7. (*Convergencia lineal.*) Sea $g \in C[a, b]$ tal que $g \in [a, b]$ para toda $x \in [a, b]$. Si g' es continua en (a, b) y existe una constante $k < 1$ tal que

$$|g'(x)| \leq k, \text{ para todo } x \in (a, b),$$

y si $g'(\bar{x}) \neq 0$, entonces para cualquier $x_0 \in [a, b]$ la sucesión

$$x_i = g(x_{i-1}), \text{ para } i \geq 1,$$

converge sólo linealmente al punto fijo $\bar{x} \in [a, b]$.

Teorema 2.8. (*Convergencia cuadrática.*) Sea \bar{x} la solución de la ecuación $x = g(x)$. Si $g'(\bar{x}) = 0$ y g'' es continua y está estrictamente acotada por una constante M en un intervalo abierto I que contiene a \bar{x} , entonces existirá un $\delta > 0$ tal que, para $x_0 \in [\bar{x} - \delta, \bar{x} + \delta]$, la sucesión definida por $x_i = g(x_{i-1})$ cuando $i \geq 1$, converge al menos cuadráticamente \bar{x} . Además para valores suficientemente grandes de i , se tiene

$$|x_{i+1} - \bar{x}| < \frac{M}{2} |x_i - \bar{x}|^2.$$

Las demostraciones de ambos teoremas pueden verse en [3].

El primer teorema (teorema 2.7) nos dice que para que la convergencia sea cuadrática o superior, se debe cumplir que $g'(\bar{x}) = 0$, en tanto que el segundo, nos da las condiciones que aseguran que la convergencia sea al menos cuadrática. Este teorema nos indica que el método de las aproximaciones sucesivas nos puede llevar a desarrollar métodos con orden de convergencia cuadrática o superior. En efecto, si partimos de

$$x_i = g(x_{i-1}), \tag{2.59}$$

podemos suponer que $g(x)$ se puede escribir como

$$g(x) = x - \phi(x) f(x). \tag{2.60}$$

De acuerdo con el segundo teorema (teorema 2.8), para obtener una convergencia al menos cuadrática debemos plantear que $g'(\bar{x}) = 0$. Dado que:

$$g'(x) = 1 - \phi'(x)f(x) - \phi(x)f'(x), \tag{2.61}$$

podemos escribir que

$$g'(\bar{x}) = 1 - \phi(\bar{x})f'(\bar{x}), \tag{2.62}$$

pues $f(\bar{x}) = 0$, entonces $g'(\bar{x}) = 0$ si y sólo si $\phi(\bar{x}) = 1/f'(\bar{x})$. Si reemplazamos esto en la función original nos queda

$$x_i = g(x_{i-1}) = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}, \tag{2.63}$$

que no es otra cosa que el método de Newton-Raphson.

Del segundo teorema, obtenemos además que $M = |g''(x)|$. De ahí que si $|g''(x)| = 0$, la convergencia podría ser cúbica, es decir, si $f''(x)$ se anula en el intervalo, la convergencia será superior a la cuadrática ³.

³Si se desarrolla $g''(x)$, la derivada que más incide en la convergencia es $f''(x)$.

2.7. Métodos de convergencia acelerada

Si bien hemos visto que el método de Newton-Raphson es de convergencia cuadrática, no siempre es posible utilizarlo. La principal razón es que debemos conocer la derivada de la función. Aunque vimos un método alternativo, el método de la secante, éste no resulta ser un método de convergencia cuadrática. Veremos ahora un procedimiento para obtener convergencia cuadrática a partir de un método linealmente convergente.

Supongamos que tenemos la sucesión $\{x_i\}_{i=0}^{\infty}$ que converge linealmente y que los signos de $x_i - \bar{x}$, $x_{i+1} - \bar{x}$ y $x_{i+2} - \bar{x}$ son iguales y que i es suficientemente grande. Para construir una nueva sucesión $\{\tilde{x}_i\}_{i=0}^{\infty}$ que converja más rápido que la anterior vamos a plantear que

$$\frac{x_{i+1} - \bar{x}}{x_i - \bar{x}} \approx \frac{x_{i+2} - \bar{x}}{x_{i+1} - \bar{x}}, \quad (2.64)$$

con lo cual nos queda

$$(x_{i+1} - \bar{x})^2 \approx (x_{i+2} - \bar{x})(x_i - \bar{x}). \quad (2.65)$$

Si la desarrollamos nos queda

$$x_{i+1}^2 - 2x_{i+1}\bar{x} + \bar{x}^2 \approx x_{i+2}x_i - (x_{i+2} + x_i)\bar{x} + \bar{x}^2, \quad (2.66)$$

y

$$(x_{i+2} + x_i - 2x_{i+1})\bar{x} \approx x_{i+2}x_i - x_{i+1}^2. \quad (2.67)$$

Si despejamos \bar{x} nos queda

$$\bar{x} \approx \frac{x_{i+2}x_i - x_{i+1}^2}{x_{i+2} - 2x_{i+1} + x_i}. \quad (2.68)$$

Si ahora sumamos y restamos x_i^2 y $2x_ix_{i+1}$ en el numerador, tenemos

$$\bar{x} \approx \frac{x_i^2 + x_{i+2}x_i - 2x_ix_{i+1} - x_i^2 + 2x_ix_{i+1} - x_{i+1}^2}{x_{i+2} - 2x_{i+1} + x_i} \quad (2.69)$$

$$\approx \frac{x_i(x_{i+2} - 2x_{i+1} + x_i) - (x_i^2 - 2x_ix_{i+1} + x_{i+1}^2)}{x_{i+2} - 2x_{i+1} + x_i} \quad (2.70)$$

$$\approx x_i - \frac{(x_{i+1} - x_i)^2}{x_{i+2} - 2x_{i+1} + x_i}. \quad (2.71)$$

Si definimos la nueva sucesión $\{\tilde{x}_n\}_{i=0}^{\infty}$ como

$$\tilde{x}_i = x_i - \frac{(x_{i+1} - x_i)^2}{x_{i+2} - 2x_{i+1} + x_i}, \quad (2.72)$$

obtenemos una técnica denominada *Método Δ^2 de Aitken*, que supone que la sucesión $\{\tilde{x}_i\}_{i=0}^{\infty}$ converge más rápidamente a \bar{x} que la sucesión $\{x_i\}_{i=0}^{\infty}$.

La notación Δ asociada a esta técnica está dada por:

Definición 2.2. Dada la sucesión $\{x_i\}_{i=0}^{\infty}$, la *diferencia progresiva* Δx_i está definida por

$$\Delta x_i = x_{i+1} - x_i, \text{ para } i \geq 0.$$

Las potencias más altas $\Delta^k x_i$ se definen por medio de

$$\Delta^k x_i = \Delta(\Delta^{k-1} x_i), \text{ para } k \geq 2.$$

A partir de estas definiciones tenemos que $\Delta^2 x_i$ se expresa como

$$\Delta^2 x_i = \Delta(\Delta^1 x_i) = \Delta(x_{i+1} - x_i) \quad (2.73)$$

$$= \Delta x_{i+1} - \Delta x_i = (x_{i+2} - x_{i+1}) - (x_{i+1} - x_i) \quad (2.74)$$

$$= x_{i+2} - 2x_{i+1} + x_i, \quad (2.75)$$

por lo que el método Δ^2 de Aitken puede escribirse como

$$\tilde{x}_i = x_i - \frac{(\Delta x_i)^2}{\Delta^2 x_i}. \quad (2.76)$$

Para analizar la convergencia de este método tenemos el siguiente teorema.

Teorema 2.9. Sea la sucesión $\{x_i\}_{i=0}^{\infty}$ que converge linealmente a \bar{x} , y que para valores suficientemente grandes de i , se cumpla que $(x_i - \bar{x})(x_{i+1} - \bar{x}) > 0$. Entonces la sucesión $\{\tilde{x}_i\}_{i=0}^{\infty}$ converge a \bar{x} con mayor rapidez que $\{x_i\}_{i=0}^{\infty}$ en el sentido de que

$$\lim_{i \rightarrow \infty} \frac{\tilde{x}_i - \bar{x}}{x_i - \bar{x}} = 0.$$

Si aplicamos el método Δ^2 de Aitken a una sucesión cuya convergencia sea lineal, podemos acelerar la convergencia a cuadrática. Podemos entonces desarrollar otros métodos a partir de esta técnica.

2.8. Método de Steffensen

Si aplicamos esta técnica a una sucesión obtenida por el *Método de las Aproximaciones Sucesivas* tendremos como resultado *Método de Steffensen*. Este método, en realidad, tiene una leve modificación al *Método Δ^2 de Aitken*.

Al aplicar este último método a una sucesión linealmente convergente, la nueva sucesión convergente cuadráticamente se construye mediante los siguientes términos:

$$x_0; x_1 = g(x_0); x_2 = g(x_1); \tilde{x}_0 = \{\Delta^2\}(x_0); x_3 = g(x_2); \tilde{x}_1 = \{\Delta^2\}(x_1); \dots$$

En cambio, el método de Steffensen calcula las tres primeras aproximaciones de la forma indicada pero introduce una leve modificación al calcular x_3 . En lugar de obtener a éste a partir de x_2 aplicando el método de las aproximaciones sucesivas, lo hace a partir de \tilde{x}_0 . La secuencia queda así de la siguiente forma:

$$x_0^{(0)}; x_1^{(0)} = g(x_0^{(0)}); x_2^{(0)} = g(x_1^{(0)}); x_0^{(1)} = \{\Delta^2\}(x_0^{(0)}); \quad (2.77)$$

$$x_1^{(1)} = g(x_0^{(1)}); x_2^{(1)} = g(x_1^{(1)}); x_0^{(2)} = \{\Delta^2\}(x_0^{(1)}); \dots \quad (2.78)$$

o sea, no arma una sucesión completa con el método de las aproximaciones sucesivas sino que con las tres primeras aproximaciones calcula una nueva aproximación haciendo uso del *Método Δ^2 de Aitken*, para continuar otra vez con el método de las aproximaciones sucesivas, obtener dos nuevas aproximaciones y nuevamente aplicar el *Método Δ^2 de Aitken*.

De esta manera, el método se asegura una convergencia cuadrática y mejora notablemente la precisión en los resultados obtenidos por el método de las aproximaciones sucesivas. En el siguiente ejemplo podemos ver la diferencia en la convergencia.

Supongamos que para aplicar el método de las aproximaciones sucesivas tenemos la expresión

$$x_{k+1} = \frac{2 - e^{x_k} + x_k^2}{3}, x_0 = 0,50.$$

Tabla 2.1: *Método de Steffensen - Algoritmo 1*

i	x_i	k	i	$x_i^{(k)}$
0	0,50000	0	0	0,50000
1	0,20043		1	0,20043
2	0,27275		2	0,27275
3	0,25361	1	0	0,25868
4	0,25855		1	0,25723
5	0,25727		2	0,25761
6	0,25760	2	0	0,25753
7	0,25751			
8	0,25753			

Para ver la eficacia del método y poder comparar, obtendremos la raíz por el *Método de las Aproximaciones Sucesivas* primero, y por el *Método de Steffensen*, después.

En la tabla 2.1 podemos ver los resultados obtenidos al aplicar ambos métodos. En la segunda columna están los obtenidos con *Aproximaciones Sucesivas* y en la última, los obtenidos con *Steffensen*. Observemos que el *Método de Steffensen* alcanzó más rápidamente el resultado «correcto» que el *Método de las Aproximaciones Sucesivas*. Mientras este último necesitó ocho iteraciones, el de Steffensen requirió solamente seis.

Aunque el método es muy conveniente para aproximar la raíz, su algoritmo es algo engorroso para implementarlo en una calculadora o computadora. Veamos la forma de mejorarlo. Como hemos partido del método de las aproximaciones sucesivas, se cumple que $x_{i+1} = g(x_i)$. Podemos redefinir la notación de Aitken de esta manera:

$$\Delta x_i = x_{i+1} - x_n = g(x_i) - x_i \quad (2.79)$$

$$\Delta^2 x_i = x_{i+2} - 2x_{i+1} + x_i = g(x_{i+1}) - 2g(x_i) + x_i. \quad (2.80)$$

De esta forma eliminamos explícitamente las iteraciones $i + 1$ e $i + 2$ de aproximaciones sucesivas, pero la incorporamos en forma implícita. Pero además como $f(x_i) = g(x_i) - x_i$, pues así surgió nuestra $g(x)$, podemos volver a modificar la notación anterior y dejarla así:

$$\Delta x_i = g(x_i) - x_i = f(x_i) \quad (2.81)$$

$$\Delta^2 x_i = \{g[g(x_i)] - g(x_i)\} - [g(x_i) - x_i] = f[x_i + f(x_i)] - f(x_i), \quad (2.82)$$

pues $g[g(x_i)] - g(x_i) = f[g(x_i)] = f[x_i + f(x_i)]$. Ahora, reemplacemos en el *Método de Steffensen* y eliminemos el supraíndice y redefinamos el subíndice para pasos sucesivos:

$$x_{i+1} = x_i - \frac{f(x_i)^2}{f[x_i + f(x_i)] - f(x_i)}. \quad (2.83)$$

Esta formulación nueva lo asemeja a los métodos de *Newton-Raphson* y de la *Secante*, y reduce la cantidad de iteraciones, como podemos ver en la siguiente tabla:

Este otro algoritmo del *Método de Steffensen* es mucho más conveniente, pues es fácil de codificar. Sólo requiere conocer $f(x)$ al igual que en el método de la secante, pero con la ventaja de que su orden de convergencia es similar al de *Newton-Raphson*. Comparado con aquél, tiene la ventaja de su orden de convergencia cuadrático ($O(h^2)$) y la desventaja de calcular dos veces $f(x)$ ⁴.

⁴La mayoría de las calculadoras de bolsillo incorporan el *Método de la Secante* para obtener la raíz de una ecuación no lineal; podría ser un trabajo interesante incluir este algoritmo de *Steffensen* en calculadoras más modernas o en programas como Matlab, Octave, MathCAD, e incluso SMath Studio.

Tabla 2.2: Método de Steffensen - Algoritmo 2

i	x_i	x_{i+1}
0	0,50000	0,27455
1	0,27455	0,25761
2	0,25761	0,25753
3	0,25753	0,25753

2.9. Método de Halley

Hemos visto que podemos desarrollar métodos con un orden de convergencia mayor al lineal a partir del método de las aproximaciones sucesivas, cuya convergencia es justamente lineal. Las dos mejoras obtenidas fueron el *Método de Newton-Raphson*, y su alternativa, el *Método de la Secante*, y los métodos Δ^2 de *Aitken* y de *Steffensen*. En los casos más favorables, estos métodos alcanzan un orden de convergencia cuadrático, si bien el Δ^2 de *Aitken* puede alcanzar órdenes superiores. El método de la secante no alcanza una convergencia cuadrática pero es supralineal.

También mencionamos que obtener un método con un orden de convergencia cúbico no era una tarea sencilla pero tampoco imposible, pues al plantear alguna función $g(x)$ tal que $g''(x) = 0$ en todo el intervalo, es posible que alcanzar dicha convergencia cúbica. En los últimos años, ha resurgido un método cuya convergencia es, justamente, cúbica y que fue obtenido a fines del siglo XVII o a principios del siglo XVIII, es decir, contemporáneo al método de Newton-Raphson. Este método se conoce como *Método de Halley*, aunque el mismo Halley reconoció que su método se basó en otro creado por el matemático francés Thomas Fautet de Lagny en 1691 para aproximar raíces cúbicas mediante fórmulas racionales e irracionales (ver [5]).

Existen varias formas de obtenerlo. Nos concentraremos solamente en dos.

Como mejora del Método de Newton-Raphson

De la misma forma que mejoramos el *Método de las Aproximaciones Sucesivas* podemos obtener una mejora del *Método de Newton-Raphson*. Entonces, definamos una nueva función $\phi(x)$ de la siguiente manera:

$$\phi(x) = f(x) \cdot \gamma(x). \quad (2.84)$$

Si a nuestra raíz de la función $f(x) = 0$ la volvemos llamar \bar{x} , impondremos que esta nueva función cumpla con lo siguiente:

$$\phi'(\bar{x}) \neq 0, \quad \phi''(\bar{x}) = 0 \quad \text{y} \quad \phi'''(\bar{x}) \neq 0. \quad (2.85)$$

Al imponer que $\phi''(\bar{x}) = 0$ nos queda:

$$f''(\bar{x}) \cdot \gamma(\bar{x}) + 2 \cdot f'(\bar{x}) \cdot \gamma'(\bar{x}) + f(\bar{x}) \cdot \gamma''(\bar{x}) = 0, \quad (2.86)$$

y como $f(\bar{x}) = 0$, podemos escribir lo anterior así:

$$f''(\bar{x}) \cdot \gamma(\bar{x}) + 2 \cdot f'(\bar{x}) \cdot \gamma'(\bar{x}) = 0, \quad (2.87)$$

que puede ser convertida en una ecuación diferencial de la forma

$$f''(x) \cdot \gamma(x) + 2 \cdot f'(x) \cdot \gamma'(x) = 0, \quad (2.88)$$

cuya solución general es:

$$\gamma(x) = \frac{1}{\sqrt{f'(x)}}. \quad (2.89)$$

Por lo tanto, nuestra función $\phi(x)$ queda así:

$$\phi(x) = \frac{f(x)}{\sqrt{f'(x)}}. \quad (2.90)$$

Con esta nueva función $\phi(x)$ desarrollemos el método de Newton-Raphson:

$$x_{i+1} = x_i - \frac{\phi(x_i)}{\phi'(x_i)} = x_i - \frac{f(x_i)}{\sqrt{f'(x_i)}} \frac{2\sqrt{f'(x_i)^3}}{2f'(x_i)^2 - f(x_i) \cdot f''(x_i)}. \quad (2.91)$$

Si operamos algebraicamente, obtenemos una fórmula más compacta:

$$x_{i+1} = x_i - \frac{2}{2 - \frac{f(x_i) \cdot f''(x_i)}{f'(x_i)^2}} \frac{f(x_i)}{f'(x_i)}. \quad (2.92)$$

Por desarrollo de la serie de Taylor

En forma análoga a como obtuvimos el *Método de Newton-Raphson*, podemos obtener el *Método de Halley* a partir de la serie de Taylor. Efectivamente, podemos obtener $f(\bar{x})$ a partir de cualquier x si hacemos que

$$f(\bar{x}) = f(x) + f'(x) (\bar{x} - x) + f''(x) \frac{(\bar{x} - x)^2}{2!} + f'''(x) \frac{(\bar{x} - x)^3}{3!} + \dots, \quad (2.93)$$

y como $f(\bar{x}) = 0$, nos queda

$$0 = f(x) + f'(x) (\bar{x} - x) + f''(x) \frac{(\bar{x} - x)^2}{2!} + f'''(x) \frac{(\bar{x} - x)^3}{3!} + \dots \quad (2.94)$$

Si ahora definimos $\varepsilon = \bar{x} - x$ y reemplazamos en la anterior, truncando en $f'''(x)$, nos queda

$$0 = f(x) + f'(x) \varepsilon + f''(x) \frac{\varepsilon^2}{2!} + f'''(x) \frac{\varepsilon^3}{3!} + \dots \quad (2.95)$$

Si dejamos de lado el término de truncamiento, nos queda que

$$0 = f(x) + f'(x) \varepsilon + f''(x) \frac{\varepsilon^2}{2!}, \quad (2.96)$$

que nos permite aproximar ε de esta forma:

$$\varepsilon = - \frac{f(x)}{f'(x) + \frac{f''(x)}{2} \varepsilon}. \quad (2.97)$$

Como nos queda una ecuación implícita, podemos aproximar ε con Newton-Raphson, es decir, hacer que

$$\varepsilon = - \frac{f(x)}{f'(x)}, \quad (2.98)$$

y luego reemplazar en la expresión anterior, de modo de obtener lo siguiente:

$$\varepsilon = - \frac{f(x)}{f'(x) - \frac{f''(x) f(x)}{2 f'(x)}}. \quad (2.99)$$

Si reordenamos la expresión última nos queda:

$$\varepsilon = -\frac{2}{2 - \frac{f(x) f''(x)}{f'(x)^2}} \frac{f(x)}{f'(x)}, \quad (2.100)$$

y como hemos definido que $\varepsilon = \bar{x} - x$, entonces $\bar{x} = x + \varepsilon$, por lo que si reemplazamos ε por la expresión hallada, nos queda

$$\bar{x} = x - \frac{2}{2 - \frac{f(x) \cdot f''(x)}{f'(x)^2}} \frac{f(x)}{f'(x)}, \quad (2.101)$$

que para un proceso iterativo se convierte en

$$x_{i+1} = x_i - \frac{2}{2 - \frac{f(x_i) \cdot f''(x_i)}{f'(x_i)^2}} \frac{f(x_i)}{f'(x_i)}, \quad (2.102)$$

nuevamente, el método de Halley. Si definimos

$$L_f(x_i) = \frac{f(x_i) \cdot f''(x_i)}{f'(x_i)^2}, \quad (2.103)$$

otra forma de escribir el método es:

$$x_{i+1} = x_i - \frac{2}{2 - L_f(x_i)} \frac{f(x_i)}{f'(x_i)}. \quad (2.104)$$

Observemos que nuevamente debemos imponer que $f'(x) \neq 0$ en el intervalo, pues de lo contrario $L_f(x_i)$ y $\frac{f(x_i)}{f'(x_i)}$ quedarán indeterminadas.

Orden de convergencia del método

Para analizar el orden de convergencia del método, nuevamente nos ayudaremos con la serie de Taylor. Si desarrollamos $g(x)$ a partir de $g(\bar{x})$ tenemos que

$$g(x) = g(\bar{x}) + g'(\bar{x})(x - \bar{x}) + g''(\bar{x}) \frac{(x - \bar{x})^2}{2!} + g'''(\bar{x}) \frac{(x - \bar{x})^3}{3!} + \dots$$

Al mismo tiempo, como \bar{x} es nuestra raíz, $g(\bar{x}) = \bar{x}$, o sea que el desarrollo anterior se puede escribir así:

$$g(x) = \bar{x} + g'(\bar{x})(x - \bar{x}) + g''(\bar{x}) \frac{(x - \bar{x})^2}{2!} + g'''(\bar{x}) \frac{(x - \bar{x})^3}{3!} + \dots$$

Por otro lado, sabemos que $g(x) = x - \frac{\phi(x)}{\phi'(x)}$, así que podemos obtener $g'(\bar{x})$, $g''(\bar{x})$, $g'''(\bar{x})$, etc. Si nos limitamos a estas tres derivadas, obtenemos lo siguiente:

$$g'(x) = \frac{\phi(x) \phi''(x)}{[\phi'(x)]^2} \quad (2.105)$$

$$g''(x) = \frac{\phi''(x)}{\phi'(x)} + \frac{\phi(x) \phi'''(x)}{[\phi'(x)]^2} - 2 \frac{\phi(x) \phi''(x) \phi'''(x)}{[\phi'(x)]^3} \quad (2.106)$$

$$g'''(x) = 2 \frac{\phi'''(x)}{\phi'(x)} + \frac{\phi(x) \phi^{(iv)}(x) - 3 [\phi''(x)]^2}{[\phi'(x)]^2} + 6 \frac{\phi(x)}{[\phi'(x)]^4} \{[\phi''(x)]^3 - \phi'(x) \phi''(x) \phi'''(x)\}. \quad (2.107)$$

Si reemplazamos $x = \bar{x}$, como $\phi(\bar{x}) = 0$, $\phi'(\bar{x}) \neq 0$, $\phi''(\bar{x}) = 0$ y asumiendo que $\phi^{iv}(\bar{x})$ está acotada, nos queda

$$g'(\bar{x}) = \frac{\phi(\bar{x}) \phi''(\bar{x})}{[\phi'(\bar{x})]^2} = 0 \quad (2.108)$$

$$g''(\bar{x}) = \frac{\phi''(\bar{x})}{\phi'(\bar{x})} + \frac{\phi(\bar{x}) \phi'''(\bar{x})}{[\phi'(\bar{x})]^2} - 2 \frac{\phi(\bar{x}) \phi''(\bar{x}) \phi'''(\bar{x})}{[\phi'(\bar{x})]^3} = 0 \quad (2.109)$$

$$\begin{aligned} g'''(\bar{x}) &= 2 \frac{\phi'''(\bar{x})}{\phi'(\bar{x})} + \frac{\phi(\bar{x}) \phi^{iv}(\bar{x}) - 3 [\phi''(\bar{x})]^2}{[\phi'(\bar{x})]^2} + 6 \frac{\phi(\bar{x})}{[\phi'(\bar{x})]^4} \{[\phi''(\bar{x})]^3 - \phi'(\bar{x}) \phi''(\bar{x}) \phi'''(\bar{x})\} \\ &= 2 \frac{\phi'''(\bar{x})}{\phi'(\bar{x})}. \end{aligned} \quad (2.110)$$

Ahora podemos reescribir la serie de Taylor:

$$g(x) = \bar{x} + g'''(\bar{x}) \frac{(x - \bar{x})^3}{6} + \dots = \bar{x} + 2 \frac{\phi'''(\bar{x})}{\phi'(\bar{x})} \frac{(x - \bar{x})^3}{6} + \dots \quad (2.111)$$

Si truncamos en el término con $g'''(\bar{x})$ nos queda

$$g(x) = \bar{x} + \frac{\phi'''(\xi)}{\phi'(\xi)} \frac{(x - \bar{x})^3}{3}, \quad (2.112)$$

Ahora consideremos que $x = x_k$, entonces

$$g(x_k) = \bar{x} + \frac{\phi'''(\xi)}{\phi'(\xi)} \frac{(x_k - \bar{x})^3}{3}. \quad (2.113)$$

Como $g(x_n) = x_{k+1}$ la expresión queda así:

$$x_{k+1} - \bar{x} = \frac{\phi'''(\xi)}{3 \cdot \phi'(\xi)} (x_k - \bar{x})^3 = \lambda (x_k - \bar{x})^3, \quad (2.114)$$

o, de esta otra forma

$$\frac{x_{k+1} - \bar{x}}{(x_k - \bar{x})^3} = \lambda \Rightarrow \frac{e_{k+1}}{e_k^3} = \lambda. \quad (2.115)$$

Hemos podido relacionar el error en la iteración $k + 1$ con el error en la iteración k .

Como vimos en la definición 2.1, el orden de convergencia lo obtenemos cuando

$$\lim_{k \rightarrow \infty} \frac{|x_{k+1} - \bar{x}|}{|x_k - \bar{x}|^\alpha} = \lambda.$$

Si aplicamos esta definición al error del método de Halley, $\alpha = 3$, el método es de convergencia cúbica.

2.10. Notas finales

Hasta aquí hemos visto seis métodos iterativos para obtener las raíces de una ecuación del tipo $f(x) = 0$. Los dos primeros, el de la *Biseción* y el de la *Falsa Posición* («*Regula Falsi*») son métodos que aseguran la convergencia pero que son muy lentos. Suelen usarse como una primera aproximación cuando no se tiene información más detallada del punto \bar{x} , de ahí que son conocidos como *métodos de arranque*. Sirven para acotar el intervalo en el cual se encuentra la raíz buscada. Los otros cuatro, los métodos de las *Aproximaciones Sucesivas*, *Newton-Raphson*, de la *Secante*, de *Steffensen* y de *Halley* son mucho más potentes.

Los métodos de *Newton-Raphson* y de *Steffensen* son de convergencia cuadrática, en tanto que el *Método de Halley* es de convergencia cúbica. Si bien este último asegura una rapidez importante, exige conocer la segunda derivada de la función $f(x)$, además de la primera, lo que convierte al método en algo no muy práctico para su implementación.

De todos métodos vistos, los más usuales para programar son el de las *Aproximaciones Sucesivas* y el de la *Secante*, puesto que son sencillos y no requieren conocer la derivada primera ni la derivada segunda. Es común, además, que cuando no disponemos de un intervalo lo suficientemente acotado para trabajar con los métodos de refinamiento, comenzar con el método de la bisección, y así, disminuir el «costo computacional».

Los métodos analizados son muy eficientes para resolver ecuaciones no lineales con raíces simples. Cuando la ecuación $f(x) = 0$ tiene raíces múltiples, ninguno de estos métodos puede distinguir rápidamente esta situación. Ejemplo de ello son algunos polinomios. Una forma de determinar si una función tiene raíces múltiples está en el análisis de las derivadas. Por ejemplo, una raíz x_p es de multiplicidad m si se cumple que:

$$f(x_p) = f'(x_p) = f''(x_p) \dots = f^{(m-1)}(x_p) = 0 \quad \text{y} \quad f^{(m)}(x_p) \neq 0.$$

Para este tipo de funciones existen otros métodos que pueden se pueden ver en [3].

Ejercicios

Métodos de arranque

1. Obtenga la la raíz de las siguientes ecuaciones no lineales con una tolerancia $\varepsilon = 10^{-4}$, mediante el *Método de la Bisección*:

a) $f(x) = x - \cos(x)$ en el intervalo $[0; 1]$

b) $x \cos(x) = \ln(x)$ en el intervalo $[0; 1,6]$

2. Sea

$$f(x) = 3(x + 1) \left(x - \frac{1}{2} \right) (x - 1),$$

aplique el *Método de la Bisección* y el *Método de la Falsa Posición* o *Regula Falsi* para obtener las raíces en los intervalos $[-2; 1,5]$ y $[-1,25; 2,5]$ con una tolerancia $\varepsilon = 10^{-6}$.

3. Aplique los *Métodos de la Bisección* y de la *Falsa Posición* para obtener las raíces del siguiente polinomio:

$$x^4 - 2x^3 - 4x^2 + 4x - 4,$$

e indique cuál de los métodos converge más rápido a la solución, si toma una tolerancia $\varepsilon = 10^{-6}$, para los siguientes intervalos: $[-2; -1]$, $[0; 2]$, $[2; 3,5]$ y $[-1; 0]$.

4. Encuentre una aproximación de $\sqrt{3}$ con una tolerancia $\varepsilon = 10^{-4}$ aplicando los *Métodos de la Bisección* y de la *Falsa Posición*. (Sugerencia: considere $f(x) = x^2 - 3$ y el intervalo $[0; 2]$.)

Métodos de refinamiento

Método de las Aproximaciones Sucesivas o del Punto Fijo

1. Efectúe cuatro iteraciones para obtener la raíz de la función $f(x) = x^4 + 2x^2 - x - 3$ en los intervalos $[-1; 0]$ y $[0; 2]$ con las funciones $g_i(x)$ que se proponen, aplicando el *Método*

de las Aproximaciones Sucesivas o de Punto Fijo:

$$\begin{array}{ll} a) & g_1(x) = (3 + x - 2x^2)^{\frac{1}{4}} \\ b) & g_2(x) = \sqrt{\frac{3 + x - x^4}{2}} \\ c) & g_3(x) = \sqrt{\frac{x + 3}{x^2 + 2}} \\ d) & g_4(x) = \frac{3x^4 + 2x^2 + 3}{4x^3 + 4x - 1} \end{array}$$

(Sugerencia: verifique si las funciones $g_i(x)$ cumplen con las condiciones suficientes para su convergencia.)

2. Para cada una de las siguientes ecuaciones, determine la función $g(x)$ y un intervalo $[a, b]$ que asegure la convergencia:

$$a) \quad 3x^2 - e^x = 0 \qquad b) \quad x - \cos x = 0 \qquad c) \quad x \operatorname{sen} x - \ln x = 0.$$

3. Obtenga el valor de L , longitud de onda de una ola marítima, en la siguiente ecuación:

$$L = \frac{g T^2}{2\pi} \tanh\left(\frac{2\pi d}{L}\right),$$

donde: g es la aceleración de la gravedad ($9,81 \text{ m/s}^2$), T es el período (8 s) y d es la profundidad del mar (7 m). Sugerencia: tome $L_0 = \frac{g T^2}{2\pi}$.

Método de Newton-Raphson

1. Obtenga la raíz de las siguientes ecuaciones mediante la aplicación del *Método de Newton-Raphson*:

$$\begin{array}{l} a) \quad f(x) = x^2 - 6 \text{ y } x_0 = 1; \\ b) \quad f(x) = x^3 + \cos x \text{ y } x_0 = -1; \\ c) \quad f(x) = 3x^2 - e^x \text{ y } x_0 = 2. \end{array}$$

2. Obtenga las raíces de las ecuaciones siguientes con un precisión de 10^{-4} :

$$\begin{array}{l} a) \quad x^3 - 2x^2 - 5 = 0 \text{ en } [0; 4]; \\ b) \quad x^3 - 3x^2 \cdot 2^{-x} + 3x \cdot 4^{-x} - 8^{-x} = 0 \text{ en } [0, 1]; \\ c) \quad e^{6x} + 3 \cdot (\ln 2)^2 \cdot 3^{2x} - e^{4x} \cdot \ln 8 - (\ln 2)^3 = 0 \text{ en } [-1; 0]. \end{array}$$

3. Obtenga la raíz cúbica de un número c . Sugerencia: considere $f(x) = x^3 - c = 0$.

Método de la Secante

1. Obtenga la raíz de las siguientes ecuaciones, mediante la aplicación del *Método de la Secante*:

$$\begin{array}{l} a) \quad f(x) = x^2 - 6 \text{ y } x_0 = 1; \\ b) \quad f(x) = x^3 + \cos x \text{ y } x_0 = -1; \\ c) \quad f(x) = 3x^2 - e^x \text{ y } x_0 = 2. \end{array}$$

Compare con los resultados obtenidos con el *Método de Newton-Raphson*.

2. Obtenga las raíces de las ecuaciones siguientes con un precisión de 10^{-4} :

$$\begin{array}{l} a) \quad x^3 - 2x^2 - 5 = 0 \text{ en } [0; 4]; \\ b) \quad x^3 - 3x^2 \cdot 2^{-x} + 3x \cdot 4^{-x} - 8^{-x} = 0 \text{ en } [0, 1]; \\ c) \quad e^{6x} + 3 \cdot (\ln 2)^2 \cdot 3^{2x} - e^{4x} \cdot \ln 8 - (\ln 2)^3 = 0 \text{ en } [-1; 0]. \end{array}$$

Compare con los resultados obtenidos con el *Método de Newton-Raphson*.

Método de Steffensen

1. Obtenga la raíz de las siguientes ecuaciones:

a) $f(x) = x^2 - 6$ y $x_0 = 1$;

b) $f(x) = x^3 + \cos x$ y $x_0 = -1$;

c) $f(x) = 3x^2 - e^x$ y $x_0 = 2$.

Compare con los resultados obtenidos con el *Método de Newton-Raphson* y el *Método de la Secante*.

2. Obtenga las raíces de las ecuaciones siguientes con un precisión de 10^{-4} :

a) $x^3 - 2x^2 - 5 = 0$ en $[0; 4]$;

b) $x^3 - 3x^2 \cdot 2^{-x} + 3x \cdot 4^{-x} - 8^{-x} = 0$ en $[0, 1]$;

c) $e^{6x} + 3 \cdot (\ln 2)^2 \cdot 3^{2x} - e^{4x} \cdot \ln 8 - (\ln 2)^3 = 0$ en $[-1; 0]$.

Compare con los resultados obtenidos con el *Método de Newton-Raphson* y el *Método de la Secante*.

3. Obtenga L de la ecuación del punto 3 de *Métodos de las Aproximaciones Sucesivas* y compare los resultados obtenidos.

Método de Halley

1. Obtenga la raíz de las siguientes ecuaciones mediante el *Método de Halley*:

a) $f(x) = x^2 - 6$ y $x_0 = 1$;

b) $f(x) = x^3 + \cos x$ y $x_0 = -1$;

c) $f(x) = 3x^2 - e^x$ y $x_0 = 2$.

Compare con los resultados obtenidos con el *Método de Newton-Raphson*, el *Método de la Secante* y el *Método de Steffensen*.

2. Obtenga las raíces de las ecuaciones siguientes con un precisión de 10^{-4} :

a) $x^3 - 2x^2 - 5 = 0$ en $[0; 4]$;

b) $x^3 - 3x^2 \cdot 2^{-x} + 3x \cdot 4^{-x} - 8^{-x} = 0$ en $[0; 1]$;

c) $e^{6x} + 3 \cdot (\ln 2)^2 \cdot 3^{2x} - e^{4x} \cdot \ln 8 - (\ln 2)^3 = 0$ en $[-1; 0]$.

Compare con los resultados obtenidos con el *Método de Newton-Raphson*, el *Método de la Secante* y el *Método de Steffensen*.

Capítulo 3

Sistemas de Ecuaciones Lineales y No Lineales

3.1. Introducción

Una de las características fundamentales del uso de las computadoras es la dificultad para trabajar con métodos simbólicos. Si bien hoy existen varios programas que trabajan con matemática simbólica (Mathematica, Maple, MathCAD, Maxima) y otros tienen módulos simbólicos (como Sympy de Python), no es lo más usual y muchas veces la capacidad de esos programas se ve excedida por las demandas ingenieriles en cantidad de cálculo. Más de una vez la necesidad de obtener un resultado en el menor tiempo posible hace imperioso contar con algún método que estime el valor en forma numérica.

Buena parte de los problemas ingenieriles de la actualidad hacen un uso intensivo de sistemas de ecuaciones lineales, usualmente definidos como $\mathbf{Ax} = \mathbf{B}$. En particular, el uso extendido de programas que aplican el método de los elementos finitos o de las diferencias finitas es un ejemplo de ello. En esos programas, como los de análisis estructural, el núcleo principal del programa es la resolución de sistemas de ecuaciones lineales de grandes dimensiones ($1,000 \times 1,000$, $10,000 \times 10,000$, etc.). En este tipo de problemas no resulta muy eficiente invertir la matriz de coeficientes para hallar la solución del sistema. También la aplicación de métodos de regresión múltiple requieren la solución de sistemas de ecuaciones lineales, algo usual en estadística. Podemos decir, entonces, que en ingeniería el uso de sistemas de ecuaciones lineales es una práctica habitual.

Por lo tanto, uno de los temas más importantes del análisis numérico es el estudio de la resolución de estos sistemas de ecuaciones. Si bien conocemos métodos muy precisos (exactos) para resolver sistemas de pequeñas dimensiones, el problema es analizar cómo resolver sistemas de grandes a muy grandes dimensiones.

Del álgebra lineal sabemos que podemos obtener la solución de $\mathbf{Ax} = \mathbf{B}$ si hacemos $\mathbf{x} = \mathbf{A}^{-1}\mathbf{B}$, pero obtener la inversa de \mathbf{A} no es una tarea sencilla, más si la matriz no sigue un patrón determinado o si está *mal condicionada*, concepto que estudiaremos más adelante.

Como introducción y repaso, veremos primero algunas definiciones relacionadas con el álgebra vectorial y matricial para luego estudiar varios métodos que resuelven un sistema de ecuaciones sin invertir la matriz de coeficientes de manera muy eficiente y para distintas condiciones.

3.2. Definiciones

Empezaremos dar algunas definiciones relacionadas con los vectores y las matrices.

Definición 3.1. Una matriz que tiene la misma cantidad de filas que de columnas (\mathbf{A} es de $n \times n$ dimensiones) se denomina *matriz cuadrada*.

Para que una matriz pueda tener inversa debe ser necesariamente cuadrada.

Definición 3.2. Una matriz cuyo determinante es no nulo ($\det(\mathbf{A}) \neq 0$) se denomina *matriz no singular*.

Definición 3.3. Una matriz \mathbf{A} cuadrada tiene inversa, es decir, existe \mathbf{A}^{-1} , si \mathbf{A} es una matriz *no singular*.

A partir de esta última definición podemos decir que un sistema de ecuaciones lineales tiene solución única si la matriz \mathbf{A} del sistema $\mathbf{Ax} = \mathbf{B}$ es *cuadrada y no singular*.

Definición 3.4. Se denomina *rango de un matriz* al número de filas que son linealmente independiente.

Por lo tanto, el rango de una matriz cualquiera siempre es menor o igual al número de filas ($\text{rango}(\mathbf{A}) \leq \text{número de filas}$). De esto último se puede inferir que una matriz \mathbf{A} de $n \times n$ dimensiones es no singular si su rango es n ($\text{rango}(\mathbf{A}) = n$). Si el vector \mathbf{B} se puede escribir como combinación lineal de las columnas de la matriz \mathbf{A} y la matriz \mathbf{A} es singular, entonces existen infinitas soluciones para el sistema.

Definición 3.5. Una norma vectorial en \mathfrak{R}^n es una función, $\|\cdot\|$, de \mathfrak{R}^n en \mathfrak{R} , con las siguientes propiedades:

1. $\|\mathbf{x}\| > 0$ para todo $x \in \mathfrak{R}^n$;
2. $\|\mathbf{x}\| = 0$ si y sólo si $x \equiv 0$ ($x = [0; 0; \dots; 0]^T$);
3. $\|\alpha \cdot \mathbf{x}\| = |\alpha| \cdot \|\mathbf{x}\|$ para todo $\alpha \in \mathfrak{R}$ y $x \in \mathfrak{R}^n$, y ;
4. $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ para todo $x; y \in \mathfrak{R}^n$.

Definición 3.6. Las normas l_2 y l_∞ de un vector están definidas por:

1. $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$ (también llamada norma euclídea);
2. $\|\mathbf{x}\|_\infty = \max_{1 \leq i \leq n} |x_i|$.

Definición 3.7. Una norma matricial sobre un conjunto de todas las matrices $n \times n$ es una función de valor real, $\|\cdot\|$, definida en este conjunto y que satisface para todas las matrices A y B de $n \times n$ y todos los números reales α :

1. $\|\mathbf{A}\| > 0$;
2. $\|\mathbf{A}\| = 0$ si y sólo si $\mathbf{A} \equiv 0$, es decir, \mathbf{A} es la matriz nula;
3. $\|\alpha \cdot \mathbf{A}\| = |\alpha| \cdot \|\mathbf{A}\|$;
4. $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$;
5. $\|\mathbf{A} \cdot \mathbf{B}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$.

3.3. Matrices triangulares

Una matriz triangular es aquella que sólo tiene coeficientes no nulos en la diagonal principal y por encima o por debajo de ella. Hay dos tipos: la matriz triangular superior, generalmente denominada \mathbf{U} , cuando los coeficientes nulos están por debajo de la diagonal principal, y la matriz triangular inferior, denominada \mathbf{L} , cuando los ceros están por encima de la diagonal principal. Estas matrices son muy convenientes cuando se deben resolver sistemas de ecuaciones lineales puesto que permiten una rápida obtención de los resultados sin la necesidad de invertir la matriz de coeficientes \mathbf{A} . Estos dos tipos de matrices dan lugar a dos métodos muy utilizados: la sustitución inversa, para matrices \mathbf{U} , y la sustitución directa, para matrices \mathbf{L} .

Por ejemplo, para el primer caso, una matriz \mathbf{U} de dimensiones 4×4 tiene la siguiente forma:

$$\mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix}$$

Para resolver un sistema $\mathbf{U}\mathbf{x} = \mathbf{B}$ basta con empezar por la última fila para obtener x_4 y luego ir reemplazando este valor en las ecuaciones anteriores, es decir, hacer:

$$\begin{aligned} x_4 &= \frac{b_4}{u_{44}} \\ x_3 &= \frac{b_3 - u_{34} \cdot x_4}{u_{33}} \\ &\vdots \\ x_i &= \frac{b_i - \sum_{j=i+1}^n u_{ij} \cdot x_j}{u_{ii}} \end{aligned}$$

Esta forma de resolver el sistema de ecuaciones lineales se denomina *sustitución inversa*.

Cuando la matriz es triangular inferior el procedimiento para resolver $\mathbf{L}\mathbf{x} = \mathbf{B}$ es:

$$\begin{aligned} x_1 &= \frac{b_1}{l_{11}} \\ x_2 &= \frac{b_2 - l_{21} \cdot x_1}{l_{22}} \\ &\vdots \\ x_i &= \frac{b_i - \sum_{j=1}^{i-1} l_{ij} \cdot x_j}{l_{ii}} \end{aligned}$$

En este caso, el método se denomina *sustitución directa*.

Cualquiera de estos métodos es sencillo de aplicar y evita tener que invertir la matriz de coeficiente de un sistema de ecuaciones lineales, lo que facilita la resolución del mismo. En consecuencia, los métodos directos se basan en transformar la matriz de coeficientes original no triangular, en una nueva matriz de coeficientes triangular.

3.4. Eliminación de Gauss y sustitución inversa

El *Método de Eliminación de Gauss*¹ es un método directo muy efectivo que transforma una matriz cualquiera en una matriz triangular superior y luego aplica el método de sustitución inversa para obtener la solución del sistema dado. Para ello se basa en la propiedad que tienen las matrices de que la misma no cambia si se reemplaza alguna de sus filas por una combinación lineal de ella con alguna de las restantes filas. El procedimiento en líneas generales es:

- Se fija la primera fila de la matriz \mathbf{A} .
- Se transforman las filas siguientes de manera de que el coeficiente a_{i1} se anule, es decir, se utiliza el coeficiente a_{11} de la diagonal principal como *pivote*.
- Se fija la siguiente fila, se fija el pivote en la diagonal principal y se repite el paso anterior.
- Se continúa hasta que la matriz queda transformada en una matriz triangular superior.
- Se aplica la sustitución inversa para hallar los x_i .

Por ejemplo, supongamos que tenemos la siguiente matriz \mathbf{A} de dimensiones $n = 4$, con su vector independiente \mathbf{B} , y generamos la matriz ampliada:

$$\mathbf{A} = \left[\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ a_{21} & a_{22} & a_{23} & a_{24} & b_2 \\ a_{31} & a_{32} & a_{33} & a_{34} & b_3 \\ a_{41} & a_{42} & a_{43} & a_{44} & b_4 \end{array} \right].$$

Para obtener el vector \mathbf{x} debemos proceder así:

1. Fijar la primera fila de la matriz;
2. Calcular el coeficiente m_{21} :

$$m_{21} = \frac{a_{21}}{a_{11}}$$

3. Luego calcular los coeficientes a_{2i}^* y b_2^* :

$$a_{22}^* = a_{22} - m_{21} \times a_{12}$$

$$a_{23}^* = a_{23} - m_{21} \times a_{13}$$

$$a_{24}^* = a_{24} - m_{21} \times a_{14}$$

$$b_2^* = b_2 - m_{21} \times b_1$$

4. Proceder de la misma forma con el resto de las filas, calculando los coeficientes m_{31} y m_{41} y los coeficientes a_{3i}^* , a_{4i}^* , b_3^* y b_4^* , hasta obtener la primera transformación de la matriz ampliada:

$$\mathbf{A} = \left[\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & a_{22}^* & a_{23}^* & a_{24}^* & b_2^* \\ 0 & a_{32}^* & a_{33}^* & a_{34}^* & b_3^* \\ 0 & a_{42}^* & a_{43}^* & a_{44}^* & b_4^* \end{array} \right];$$

¹Es interesante la historia de este método. Una buena reseña se puede leer en [9].

5. Fijar la siguiente fila (la segunda) de la matriz transformada y repetir los pasos 2 a 4, es decir, calcular los coeficientes m_{kj} , (m_{32} y m_{42}), y efectuar una nueva transformación. Operando sucesivamente de esta forma (al calcular el coeficiente m_{43}), obtendremos finalmente la matriz ampliada triangulada:

$$\mathbf{A} = \left[\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & a_{23}^* & a_{23}^\# & a_{24}^* & b_2^* \\ 0 & 0 & a_{33}^\# & a_{34}^\# & b_3^\# \\ 0 & 0 & 0 & a_{44}^+ & b_4^+ \end{array} \right].$$

6. Con la matriz ampliada triangulada, obtenemos el vector x por sustitución inversa, haciendo:

$$\begin{aligned} x_4 &= \frac{b_4^+}{a_{44}^+} \\ x_3 &= \frac{b_3^\# - a_{34}^\# x_4}{a_{33}^\#} \\ x_2 &= \frac{b_2^* - a_{23}^* x_3 - a_{24}^* x_4}{a_{22}^*} \\ x_1 &= \frac{b_1 - a_{12} x_2 - a_{13} x_3 - a_{14} x_4}{a_{11}} \end{aligned}$$

Entonces, la expresión general para la transformación de las filas de una matriz ampliada es la siguiente:

$$a_{ij}^* = a_{ij} - m_{il} \times a_{lj}, \quad (3.1)$$

para los coeficientes de la matriz \mathbf{A} , y:

$$b_i^* = b_i - m_{il} \times b_l \quad (3.2)$$

para los coeficientes del vector de términos independientes (\mathbf{B}), con $m_{il} = \frac{a_{il}}{a_{ll}}$.

Este procedimiento es muy útil puesto que se conoce exactamente la cantidad de pasos que deben efectuarse, es decir, el método tiene un cantidad *finita* de pasos, inclusive si el sistema a resolver cuenta con varios vectores \mathbf{B} . En ese caso, basta con transformarlos conjuntamente con la matriz \mathbf{A} .

Con este procedimiento, es posible conocer el «costo computacional» del método, es decir, establecer cuanto tiempo lleva todo el proceso. Una forma de estimar este costo de transformación de la matriz en triangular superior es mediante la siguiente expresión que cuenta las operaciones realizadas (sumas, restas, multiplicaciones y divisiones). Para la transformación de la matriz \mathbf{A} ampliada con el vector \mathbf{B} en una matriz triangular superior tenemos la siguiente cantidad de operaciones:

$$\sum_{k=1}^{n-1} [(n-k) + 2(n-k)(n-k+1)] = \frac{2}{3}n^3 + \frac{n^2}{2} - \frac{7}{6}n. \quad (3.3)$$

A su vez, para la sustitución inversa tenemos esta cantidad de operaciones:

$$1 + \sum_{k=1}^{n-1} [2(n-k) + 1] = n^2. \quad (3.4)$$

En consecuencia, si se suman ambos valores, tenemos que el costo de efectuar la eliminación de Gauss es:

$$\frac{2}{3}n^3 + \frac{3}{2}n^2 - \frac{7}{6}n; \quad (3.5)$$

es decir, proporcional a n^3 .

Conviene tener presente que esta estimación es aproximada, pues no se han tenido en cuenta otros «costos» difíciles de evaluar como son el manejo de las prioridades de memoria, la forma de guardar los datos, etc. Sin embargo, esta estimación sirve para establecer que a medida que la dimensión de la matriz aumenta, el costo es proporcional al cubo de la misma, es decir, el aumento del tiempo empleado en resolver el sistema completo (el «costo computacional») es potencial y no lineal. Es por ello que resolver un sistema de 1000×1000 insume un costo proporcional a 1 000 000 000 operaciones.

Un problema que puede surgir en este método es si alguno de los elementos de la diagonal principal al ser transformados se anulan. Si esto ocurriera, de acuerdo con el algoritmo anterior, el procedimiento se detendría y en consecuencia no podría obtenerse solución alguna. En estos casos se aplican versiones más desarrolladas, denominadas *Eliminación de Gauss con Pivoteo Parcial* (EGPP) o *Eliminación de Gauss con Pivoteo Total* (EGPT).

En el primer caso, lo que se hace es primero intercambiar las filas, reordenándolas de manera tal que el coeficiente nulo quede fuera de la diagonal principal, y luego se continúa con el algoritmo tradicional. Veamos un ejemplo. Supongamos el siguiente sistema:

$$\begin{aligned}x_1 + x_2 - x_3 &= 1 \\x_1 + x_2 + 4x_3 &= 2 \\2x_1 - x_2 + 2x_3 &= 3.\end{aligned}$$

Armemos el sistema ampliado para aplicar el método de Eliminación de Gauss. Entonces nos queda:

$$\left[\begin{array}{ccc|c} 1 & 1 & -1 & 1 \\ 1 & 1 & 4 & 2 \\ 2 & -1 & 2 & 3 \end{array} \right] \rightarrow \left[\begin{array}{ccc|c} 1 & 1 & -1 & 1 \\ 0 & 0 & 5 & 1 \\ 0 & -3 & 4 & 1 \end{array} \right].$$

Como vemos, la transformación de la matriz nos deja nulo el coeficiente a_{22}^* de la segunda fila, lo que nos impide seguir operando. Para poder seguir debemos intercambiar las filas dos y tres, en consecuencia tendremos:

$$\left[\begin{array}{ccc|c} 1 & 1 & -1 & 1 \\ 0 & -3 & 4 & 1 \\ 0 & 0 & 5 & 1 \end{array} \right] \Rightarrow \begin{cases} x_1 \\ x_2 \\ x_3 \end{cases} = \begin{bmatrix} 1,2667 \\ -0,0667 \\ 0,2000 \end{bmatrix}.$$

El intercambio entre las filas 2 y 3 evitó que el procedimiento se detuviera. Pero también es posible que valores muy chicos en los coeficientes de la diagonal principal generen un problema en la mecánica del sistema. Por ejemplo, consideremos el siguiente sistema:

$$\begin{aligned}0,03x_1 + 58,9x_2 &= 59,2 \\5,31x_1 - 6,10x_2 &= 47,0;\end{aligned}$$

que debe ser resuelto con una precisión de solamente tres dígitos y aplicando corte en vez de redondeo. Si aplicamos Eliminación de Gauss tendremos:

$$\left[\begin{array}{cc|c} 0,03 & 58,9 & 59,2 \\ 0 & -10400 & -10300 \end{array} \right];$$

pues al hacer los cálculos obtenemos que:

$$m_{21} = \frac{5,31}{0,03} = 177 \Rightarrow a_{22}^* = -6,10 - 177 \times 58,9 \approx -6,10 - 10400 \approx -10400$$

$$b_2^* = 47,0 - 177 \times 59,2 \approx 47,0 - 10400 \approx -10300.$$

Así, la solución del sistema es:

$$\begin{cases} x_1 \\ x_2 \end{cases} = \begin{bmatrix} 30,0 \\ 0,990 \end{bmatrix},$$

Pero si resolvemos el sistema anterior con precisión «infinita», el resultado que obtenemos es:

$$\begin{cases} x_1 \\ x_2 \end{cases} = \begin{bmatrix} 10 \\ 1 \end{bmatrix},$$

lo que nos indica que el resultado anterior es incorrecto. Esta diferencia está dada por el coeficiente 0,03 en la diagonal principal. Si reordenamos el sistema original tenemos:

$$\begin{aligned} 5,31x_1 - 6,10x_2 &= 47,0 \\ 0,03x_1 + 58,9x_2 &= 59,2; \end{aligned}$$

y si utilizamos la misma precisión, resulta:

$$\left[\begin{array}{cc|c} 5,31 & -6,10 & 47,0 \\ 0 & 58,9 & 58,9 \end{array} \right];$$

puesto que al hacer los cálculos obtenemos:

$$m_{21} = \frac{0,03}{5,31} = 0,005649 \approx 0,005 \Rightarrow a_{22}^* = 58,9 - 0,005 \times (-6,10) \approx 58,9 + 0,030 \approx 58,9$$

$$b_2^* = 59,2 - 0,005 \times 47,0 = 59,2 - 0,235 \approx 58,9.$$

La solución del sistema es:

$$\begin{cases} x_1 \\ x_2 \end{cases} = \begin{bmatrix} 10 \\ 1 \end{bmatrix},$$

resultado que coincide con el obtenido con precisión «infinita».

Es por eso que el método de *Eliminación de Gauss con Pivoteo Parcial* (EGPP) se usa también cuando alguno de los coeficientes de la diagonal principal es muy chico con respecto a los demás coeficientes de la matriz.

En el caso del pivoteo total se efectúa no sólo un reordenamiento de las filas sino también de las columnas, lo que complica aún más el procedimiento.

Ambos casos insumen un mayor costo computacional que resulta muy difícil estimar puesto que no se trata de contar operaciones aritméticas como en la estimación anterior, si bien se considera que una comparación es equivalente a una suma/resta.

3.5. Factorización LU

El método de *Eliminación de Gauss* es muy potente. Sin embargo, no siempre es conveniente su utilización. Supongamos por un momento que para resolver un determinado problema debemos resolver el sistema de ecuaciones en forma anidada. Es decir, cada nueva solución depende del resultado obtenido en un paso anterior, es decir, cada vector \mathbf{B} depende de la solución anterior ($\mathbf{B}^{(i)} = f(\mathbf{x}^{(i-1)})$).

Si queremos resolver estos sistemas nos encontraremos con la desventaja de que en cada paso tendremos que recalcular la matriz triangular superior, lo que significa un costo computacional muy grande, tal como vimos en el punto anterior. Por lo tanto, deberíamos buscar un método que nos evite repetir dichos cálculos.

Un método muy eficiente para estos casos es la *Descomposición o Factorización LU*. Ésta consiste en descomponer la matriz \mathbf{A} original en el producto de dos matrices: una triangular inferior (\mathbf{L}) y una triangular superior (\mathbf{U}), para armar el siguiente sistema:

$$\mathbf{Ax} = \mathbf{LUx} = \mathbf{A} \quad \text{con} \quad \mathbf{A} = \mathbf{LU}.$$

De esta forma obtenemos dos sistemas de ecuaciones:

$$\mathbf{L}\mathbf{y} = \mathbf{B}, \quad (3.6)$$

$$\mathbf{U}\mathbf{x} = \mathbf{y} \quad (3.7)$$

En el primer caso, para obtener la solución intermedia \mathbf{y} , aplicamos la sustitución directa, y en el segundo, la sustitución inversa. Vemos que en este método el vector \mathbf{B} no es transformado en ninguno de los sistemas resueltos, que es lo que estábamos buscando. ¿Pero cómo se obtienen las dos matrices triangulares?

En el caso de la matriz triangular superior, la forma más sencilla de obtenerla es aplicar el mismo algoritmo que el utilizado para *Eliminación de Gauss*, lo que significa que el costo computacional es similar (pero no igual, puesto que no debe transformarse al vector \mathbf{B}). Nos falta la matriz \mathbf{L} . Pero esta matriz es muy sencilla de obtener. Planteemos el esquema para obtener los coeficientes de la matriz \mathbf{L} partiendo que los elementos de la diagonal principal son iguales a 1 ($l_{ii} = 1$):

$$\begin{aligned} u_{11} &= a_{11} \\ l_{21}u_{11} &= a_{21} \Rightarrow l_{21} = \frac{a_{21}}{u_{11}} = \frac{a_{21}}{a_{11}} = m_{21} \\ l_{31}u_{11} &= a_{31} \Rightarrow l_{31} = \frac{a_{31}}{u_{11}} = \frac{a_{31}}{a_{11}} = m_{31} \\ &\dots\dots\dots \\ l_{31}u_{12} + l_{32}u_{22} &= a_{32} \Rightarrow l_{32}u_{22} = a_{32} - l_{31}u_{12} = \underbrace{a_{32} - m_{31}a_{12}}_{a_{32}^*} = a_{32}^* \\ l_{32} &= \frac{a_{32}^*}{u_{22}} = \frac{a_{32}^*}{a_{22}^*} = m_{32} \end{aligned}$$

Como vemos, la matriz \mathbf{L} está compuesta por los coeficientes de la diagonal principal iguales a 1 ($l_{ii} = 1$), en tanto que los coeficientes por debajo de la diagonal principal iguales a los coeficientes m_{ij} del método de *Eliminación de Gauss* ($l_{ij} = m_{ij}$). Es decir, las matrices tienen la siguiente forma:

$$L = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{21} & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ m_{n-1,1} & \dots & m_{n-1,n-2} & 1 & 0 \\ m_{n1} & \dots & m_{n,n-2} & m_{n,n-1} & 1 \end{bmatrix}$$

y

$$U = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^* & a_{23}^* & \dots & a_{2n}^* \\ 0 & 0 & a_{33}^* & \dots & a_{3n}^* \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & a_{nn}^* \end{bmatrix}$$

donde los a_{ij}^* son los coeficientes transformados del método de *Eliminación de Gauss*.

Obtenidas \mathbf{L} y \mathbf{U} , la solución del sistema la obtenemos aplicando, primero, la sustitución directa para hallar el vector \mathbf{y} y luego, sustitución inversa para hallar \mathbf{x} . Para el primer caso,

aplicamos el siguiente algoritmo:

$$\begin{aligned} y_1 &= b_1 \\ y_2 &= b_2 - l_{21}y_1 \\ &\vdots \\ y_i &= b_i - \sum_{j=1}^{i-1} l_{ij}y_j \end{aligned}$$

puesto que los coeficientes l_{ii} son iguales a uno ($l_{ii} = 1$).

Como dijimos, obtenido \mathbf{y} , se aplica la sustitución inversa para obtener el vector \mathbf{x} solución del sistema. El algoritmo es:

$$\begin{aligned} x_n &= \frac{y_n}{u_{nn}} \\ x_{n-1} &= \frac{y_{n-1} - u_{n-1,n}y_n}{u_{n-1,n-1}} [1mm]: \\ x_i &= \frac{y_i - \sum_{j=i+1}^n u_{ij}x_j}{u_{ii}} \end{aligned}$$

Como vemos, en ningún caso hemos modificado o transformado al vector \mathbf{B} , por lo que una vez que obtenemos las matrices \mathbf{U} y \mathbf{L} , podemos resolver los distintos sistemas aplicando sustitución directa primero e inversa después. Este método se conoce como *Método de Doolittle*.

Ahora nos quedaría analizar el costo computacional del método. Sin embargo, dado que hemos utilizado el método de *Eliminación de Gauss* para obtener las matrices \mathbf{U} y \mathbf{L} , el costo para este método es muy similar al de dicho método. En consecuencia, la ventaja está principalmente en no tener que repetir la triangulación de la matriz \mathbf{A} para cada sistema con un \mathbf{B} distinto.

Al obtener la matriz \mathbf{U} mediante *Eliminación de Gauss* podemos tener el mismo problema ya visto: que un coeficiente de la diagonal principal se haga nulo en los pasos intermedios. En ese sentido, valen las mismas aclaraciones respecto al *Pivoteo Parcial* y al *Pivoteo Total*. Es por eso que suele decirse que existe un par de matrices \mathbf{L} y \mathbf{U} que cumplen con:

$$\mathbf{PA} = \mathbf{LU}, \quad (3.8)$$

donde \mathbf{P} es una *matriz de permutación*.

3.6. Método de Cholesky

3.6.1. Matrices simétricas y definidas positivas

Antes de analizar un caso particular de factorización de matrices conviene recordar la definición de un algunos tipos de matrices. En primer lugar, se dice que una matriz es simétrica cuando dicha matriz es igual a su transpuesta, es decir:

$$\mathbf{A} = \mathbf{A}^T.$$

Otro tipo de matriz es la conocida como definida positiva². En este caso se debe cumplir que:

$$\mathbf{x}^T \mathbf{A} \mathbf{x} > 0 \text{ para todo } \mathbf{x} \neq 0.$$

²Algunos autores exigen que \mathbf{A} sea simétrica y definida positiva. Sin embargo, en principio, se puede decir que no es necesario que una matriz sea simétrica para que sea definida positiva.

Es de notar que lo que se impone para que una matriz sea definida positiva es que el escalar resultante de la operación $\mathbf{x}^T \mathbf{A} \mathbf{x}$ sea no nulo y mayor que cero. En general demostrar esto resulta muy engorroso, por lo que suelen utilizarse algunos procedimientos alternativos. Para ello veamos los siguiente conceptos.

Definición 3.8. Una *primera submatriz principal* de una matriz \mathbf{A} es la que tiene la forma:

$$\mathbf{A}_k = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1k} \\ a_{21} & a_{22} & \dots & a_{2k} \\ \dots & \dots & \dots & \dots \\ a_{k1} & a_{k2} & \dots & a_{kk} \end{bmatrix}$$

para alguna $1 \leq k \leq n$.

Teorema 3.1. Una matriz simétrica \mathbf{A} es *definida positiva* si y sólo si sus primeras submatrices principales tienen determinante positivo.

Teorema 3.2. La matriz simétrica \mathbf{A} es *definida positiva* si y sólo si la eliminación de Gauss sin pivoteo puede efectuarse en el sistema $\mathbf{A} \mathbf{x} = \mathbf{B}$ con todos los pivotes positivos.

Corolario 3.2.1. La matriz simétrica \mathbf{A} es *definida positiva* si y sólo si \mathbf{A} puede factorizarse en la forma $\mathbf{L} \mathbf{D} \mathbf{L}^T$, donde \mathbf{L} es una matriz triangular inferior con coeficientes iguales a uno en la diagonal principal ($l_{ii} = 1$) y \mathbf{D} es una matriz diagonal con coeficientes positivos ($d_{ii} > 0$).

Corolario 3.2.2. La matriz \mathbf{A} es *simétrica y definida positiva* si y sólo si \mathbf{A} puede factorizarse en la forma $\mathbf{L} \mathbf{L}^T$ donde \mathbf{L} es una matriz triangular inferior con elementos no nulos en su diagonal.

3.6.2. Algoritmo de Cholesky

Con el último corolario se puede efectuar una factorización de la matriz \mathbf{A} conocida como método o algoritmo de *Cholesky*. En efecto, si la matriz \mathbf{A} es simétrica definida positiva, es posible obtener una matriz \mathbf{S} que cumpla:

$$\mathbf{S} \mathbf{S}^T = \mathbf{A}.$$

Veamos como podemos obtener esta matriz a partir de la *Factorización LU*. De acuerdo con el corolario 3.2.1, la matriz simétrica \mathbf{A} puede ser factorizada como $\mathbf{L} \mathbf{D} \mathbf{L}^T$. Si además es definida positiva, entonces los coeficientes de \mathbf{D} son positivos. En consecuencia, podemos obtener sin problemas $\sqrt{\mathbf{D}}$, con lo cual tenemos $\mathbf{A} = \mathbf{L} \sqrt{\mathbf{D}} \sqrt{\mathbf{D}} \mathbf{L}^T$. Así nuestra matriz \mathbf{A} puede ser expresada como:

$$\mathbf{A} = \underbrace{\mathbf{L} \sqrt{\mathbf{D}}}_{\mathbf{S}} \cdot \underbrace{\sqrt{\mathbf{D}} \mathbf{L}^T}_{\mathbf{S}^T} = \mathbf{S} \mathbf{S}^T.$$

Finalmente, las expresiones para obtener esta matriz \mathbf{S} son:

$$s_{ii} = \left[a_{ii} - \sum_{k=1}^{i-1} s_{ik}^2 \right]^{1/2} \quad y \quad (3.9)$$

$$s_{ji} = \frac{1}{s_{ii}} \left[a_{ji} - \sum_{k=1}^{i-1} s_{jk} s_{ik} \right], \quad (3.10)$$

con $j > i$.

Este método es mucho más eficiente puesto que sólo debemos calcular y guardar una sola matriz, a diferencia de la *Factorización LU* en la que debemos calcular y guardar dos matrices, si bien algunos algoritmos permiten guardar ambas matrices en una sola. Además, el *Método*

Cholesky no aumenta considerablemente el «costo computacional» que analizamos en los puntos anteriores, por más que deban extraerse n raíces cuadradas.

Este método es muy aplicado en programas estructurales que aplican el método de los elementos finitos, dado que la matriz de coeficientes es una matriz simétrica y definida positiva. De todos modos, tiene las mismas desventajas vistas para los otros métodos cuando la dimensión de la matriz es cada vez más grande.

3.7. Condición de una matriz

Uno de los puntos a tener en cuenta es qué error cometemos al resolver un sistema de ecuaciones lineales mediante un método directo. Una forma de conocer el error de nuestro vector solución \mathbf{x} sería analizar el algoritmo utilizado con ayuda de la gráfica de proceso. Este procedimiento resulta un tanto engorroso y largo, además de poco práctico. Una segunda manera es analizar lo siguiente: puesto que nuestro sistema se puede expresar como $\mathbf{Ax} = \mathbf{B}$, una forma alternativa es $\mathbf{x} = \mathbf{A}^{-1}\mathbf{B}$. Si definimos $\mathbf{N} = \mathbf{A}^{-1}$, nos queda $\mathbf{x} = \mathbf{N} \cdot \mathbf{B}$. Si desarrollamos esta expresión para cada componente de \mathbf{x} nos queda:

$$x_i = \sum_{j=1}^n n_{ij}b_j. \quad (3.11)$$

Armemos un algoritmo que tenga la siguiente forma:

$$s_j = n_{ij}b_j; \quad (3.12)$$

$$x_i = \sum_{j=1}^n s_j. \quad (3.13)$$

Analicemos los errores en cada paso. Para el primero tenemos:

$$e_{s_j} = b_j e_{n_{ij}} + n_{ij} e_{b_j} = b_j e_{ij} + n_{ij} e_j \quad (3.14)$$

$$er_{s_j} = \frac{b_j e_{n_{ij}} + n_{ij} e_{b_j}}{n_{ij} b_j} + \mu_j = er_{n_{ij}} + er_{b_j} + \mu_j = er_{ij} + er_j + \mu_j. \quad (3.15)$$

Con el error relativo podemos recalcular el error total de s_j :

$$e_{s_j} = b_j e_{ij} + n_{ij} e_j + n_{ij} b_j \mu_j. \quad (3.16)$$

Para el segundo paso tendremos:

$$e_{x_i} = \sum_{j=1}^n e_{s_j} = \sum_{j=1}^n (b_j e_{ij} + n_{ij} e_j) + \sum_{j=1}^n n_{ij} b_j \mu_j, \quad (3.17)$$

$$er_{x_i} = \sum_{j=1}^n \frac{e_{s_j}}{x_i} + \sum_{k=2}^n \mu_k = \sum_{j=1}^n \frac{(b_j e_{ij} + n_{ij} e_j)}{x_i} + \sum_{j=1}^n \frac{n_{ij} b_j \mu_j}{x_i} + \sum_{k=2}^n \mu_k. \quad (3.18)$$

Esta última expresión la podemos escribir también como:

$$er_{x_i} = \sum_{j=1}^n \frac{n_{ij} b_j (er_{ij} + er_j + \mu_j)}{x_i} + \sum_{k=2}^n \mu_k. \quad (3.19)$$

Si reordenamos los términos tendremos:

$$er_{x_i} = \sum_{j=1}^n \frac{n_{ij} b_j}{x_i} (er_{ij} + er_j) + \sum_{j=1}^n \frac{n_{ij} b_j}{x_i} \mu_j + \sum_{k=2}^n \mu_k. \quad (3.20)$$

De esta forma se puede decir que el error relativo de x_i es:

$$er_{x_i} \approx \sum_{j=1}^n Cp_j (er_{ij} + er_j) + \sum_{j=1}^n Te_j \mu_j, \quad (3.21)$$

con

$$Cp_j = \frac{2n_{ij}b_j}{x_i} \text{ y} \quad (3.22)$$

$$Te_j \approx \frac{n_{ij}b_j}{x_i} + 1, \quad (3.23)$$

si tomamos que $er_{ij}; er_j < r$ y que $\mu_j < \varepsilon$.

Hemos encontrado para cada x_i la expresión del error relativo, o mejor dicho, una idea aproximada del error. Pero, en la práctica, ¿sirve esto? Todos los cálculos son engorrosos y además hemos partido de un algoritmo no del todo práctico, pues hemos dicho que invertir la matriz no es conveniente³. Entonces, ¿qué hacemos?

Supongamos que hemos resuelto nuestro sistema $\mathbf{A} \mathbf{x} = \mathbf{B}$ con un algoritmo cualquiera y que en consecuencia hemos obtenido una solución $\hat{\mathbf{x}}$. Lo que nos interesa conocer es una cota del error absoluto, $\|\mathbf{x} - \hat{\mathbf{x}}\|$, o del error relativo, $\frac{\|\mathbf{x} - \hat{\mathbf{x}}\|}{\|\mathbf{x}\|}$, en alguna norma, por ejemplo, la norma infinito.

Como, en principio, no conocemos el resultado exacto de \mathbf{x} , lo que podemos hacer es calcular lo siguiente:

$$\mathbf{R} = \mathbf{B} - \mathbf{A}\hat{\mathbf{x}},$$

donde \mathbf{R} lo denominamos *residuo*. Si nuestra solución $\hat{\mathbf{x}}$ fuera la solución exacta, entonces nuestro vector \mathbf{R} debería ser nulo. Sin embargo, en la práctica, siempre obtendremos un vector \mathbf{R} no nulo, debido a la propagación de los errores de redondeo o de los errores inherentes y de redondeo. ¿Qué conclusiones podemos sacar conociendo \mathbf{R} ? Veamos el siguiente ejemplo.

Supongamos la matriz \mathbf{A} y el vector \mathbf{B} dados a continuación:

$$\mathbf{A} = \begin{bmatrix} 1,2969 & 0,8648 \\ 0,2161 & 0,1441 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 0,8642 \\ 0,1440 \end{bmatrix}.$$

Supongamos también que usando un determinado algoritmo hemos obtenido las siguientes soluciones:

$$\hat{\mathbf{x}}_1 = \begin{bmatrix} 0,9911 \\ -0,4870 \end{bmatrix}; \hat{\mathbf{x}}_2 = \begin{bmatrix} -0,0126 \\ 1,0182 \end{bmatrix}.$$

Entonces, tendremos:

$$\mathbf{R} = \mathbf{B} - \mathbf{A}\hat{\mathbf{x}}_i \approx \begin{bmatrix} -10^{-7} \\ 10^{-7} \end{bmatrix}.$$

Por lo tanto, tendremos que $\|\mathbf{R}\|_{\infty} = 10^{-7}$. Podemos decir que el residuo es muy chico. Sin embargo, la solución correcta es:

$$\mathbf{x} = \begin{bmatrix} 2 \\ -2 \end{bmatrix}.$$

Es decir, ¡el error cometido es del mismo orden de la solución, o sea, 10^7 veces el residuo!

Es importante tener en cuenta que cualquiera sea el algoritmo utilizado, no podemos esperar sino un residuo pequeño o muy pequeño, lo que significa que este residuo \mathbf{R} por sí solo no nos sirve de mucho para estimar el error que hemos cometido al obtener $\hat{\mathbf{x}}$.

³Esta deducción es interesante, pues nos muestra que tanto el C_p como el T_e dependen de la matriz \mathbf{A} , dado que la matriz \mathbf{N} no es otra cosa que \mathbf{A}^{-1} . De ahí que la solución de cualquier sistema de ecuaciones lineales mediante la inversión de la matriz es potencialmente inestable.

¿Cómo se relaciona, entonces, este residuo con el error en $\hat{\mathbf{x}}$? Veamos. Escribamos el residuo como:

$$\mathbf{R} = \mathbf{B} - \mathbf{A}\hat{\mathbf{x}} = \mathbf{A}\mathbf{x} - \mathbf{A}\hat{\mathbf{x}} = \mathbf{A}(\mathbf{x} - \hat{\mathbf{x}}).$$

es decir:

$$\mathbf{x} - \hat{\mathbf{x}} = \mathbf{A}^{-1}\mathbf{R}.$$

Elijamos cualquier norma vectorial, por ejemplo, la infinita. Entonces tendremos:

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_{\infty} = \|\mathbf{A}^{-1}\mathbf{R}\|_{\infty} \leq \|\mathbf{A}^{-1}\|_{\infty} \|\mathbf{R}\|_{\infty}.$$

Esto nos da una cota del error absoluto en términos de \mathbf{A}^{-1} . Usualmente el error relativo es más significativo que el absoluto. Como $\|\mathbf{B}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$ implica que $\frac{1}{\|\mathbf{x}\|} \leq \frac{\|\mathbf{A}\|}{\|\mathbf{B}\|}$, tendremos que:

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}\|}{\|\mathbf{x}\|} \leq \|\mathbf{A}^{-1}\| \|\mathbf{R}\| \frac{\|\mathbf{A}\|}{\|\mathbf{B}\|} = \|\mathbf{A}\| \|\mathbf{A}^{-1}\| \frac{\|\mathbf{R}\|}{\|\mathbf{B}\|}. \quad (3.24)$$

Esta expresión nos permite establecer que el residuo por sí mismo no nos alcanza para estimar el error de nuestro vector solución $\hat{\mathbf{x}}$, sino que también debemos conocer algunas características de la matriz \mathbf{A} . En particular, vemos que el error relativo de $\hat{\mathbf{x}}$ depende de $\|\mathbf{A}^{-1}\| \|\mathbf{A}\|$. A este número lo denominaremos *condición de \mathbf{A}* y lo expresaremos como $\text{cond}_{\infty}(\mathbf{A})$ o $\kappa(\mathbf{A})^4$. Tanto $\|\mathbf{A}^{-1}\|$ como $\|\mathbf{A}\|$ son números reales (son normas de las matrices) por lo tanto para que el error relativo de $\hat{\mathbf{x}}$ no sea muy grande, el producto de $\|\mathbf{A}^{-1}\| \|\mathbf{A}\|$ debiera ser cercano a uno, es decir:

$$\|\mathbf{A}\| \|\mathbf{A}^{-1}\| = \kappa(\mathbf{A}) > / > 1.$$

Si la matriz es no singular debe cumplirse que:

$$1 = \|\mathbf{I}\| = \|\mathbf{A}^{-1}\mathbf{A}\| \leq \kappa(\mathbf{A}),$$

que puede considerarse el límite inferior, en tanto que si la matriz \mathbf{A} es singular (no existe \mathbf{A}^{-1}), $\kappa(\mathbf{A}) \rightarrow \infty$, que puede ser considerado el límite superior. Así, puede decirse que el número de condición da una idea de cuan cerca está la matriz de ser *singular*, o lo que es lo mismo, de que el sistema no tenga solución o que sean infinitas.

Una conclusión interesante es que si la matriz \mathbf{A} del sistema está mal condicionada, pequeños desvíos en el residuo \mathbf{R} pueden llevar a grandes desvíos en $\hat{\mathbf{x}}$, es decir, si definimos que $\|\Delta\mathbf{x}\| = \|\mathbf{x} - \hat{\mathbf{x}}\|$, entonces puede darse que $\|\Delta\mathbf{x}\| \gg 1$, algo que no es aceptable.

3.8. Refinamiento Iterativo de la Solución

Hemos visto en los puntos anteriores que los métodos directos pueden resolver muy bien un sistema de ecuaciones lineales, con excepción de un sistema con la matriz de coeficientes \mathbf{A} mal condicionada. Aún así, existe la posibilidad de obtener una solución aceptable, dentro de cierto rango. Al analizar el error cometido, introdujimos el concepto del vector «residuo», que denominamos \mathbf{R} , y que obtuvimos de la siguiente manera:

$$\mathbf{R} = \mathbf{B} - \mathbf{A}\hat{\mathbf{x}}. \quad (3.25)$$

Como vimos, con ese vector residuo podemos calcular el error de nuestra aproximación $\hat{\mathbf{x}}$ respecto de nuestra solución «exacta» \mathbf{x} , pues tenemos que:

$$\mathbf{B} - \mathbf{A}\hat{\mathbf{x}} = \mathbf{A}\mathbf{x} - \mathbf{A}\hat{\mathbf{x}} = \mathbf{A} \underbrace{(\mathbf{x} - \hat{\mathbf{x}})}_{\delta} = \mathbf{A} \delta = \mathbf{R}, \quad (3.26)$$

⁴En este caso hemos utilizado la norma infinito. Podría haberse usado la norma euclídea y obtener el $\text{cond}_2(\mathbf{A})$.

y, en consecuencia, resolviendo este nuevo sistema de ecuaciones podemos obtener nuestro valor δ . Dado que hemos definido que $\delta = \mathbf{x} - \hat{\mathbf{x}}$, entonces podemos decir que:

$$\mathbf{x} = \hat{\mathbf{x}} + \delta, \quad (3.27)$$

y con ello hemos obtenido nuestra solución «exacta». Sin embargo, esto no suele ocurrir al primer intento, de manera que lo que obtendremos en realidad es una nueva aproximación de nuestra solución, que llamaremos $\tilde{\mathbf{x}}$. Para sistematizar esto, digamos que

$$\hat{\mathbf{x}} = \mathbf{x}_1; \mathbf{R}_1 = \mathbf{B} - \mathbf{A}\mathbf{x}_1; \mathbf{A}\delta_1 = \mathbf{R}_1,$$

por lo que tendremos:

$$\tilde{\mathbf{x}} = \mathbf{x}_2 = \mathbf{x}_1 + \delta_1.$$

El paso siguiente es obtener \mathbf{R}_2 y δ_2 , en forma análoga a δ_1 . En consecuencia, tendremos que

$$\mathbf{x}_3 = \mathbf{x}_2 + \delta_2 = \mathbf{x}_1 + \delta_1 + \delta_2 = \hat{\mathbf{x}} + \delta_1 + \delta_2.$$

Si generalizamos, tenemos que la solución «exacta» se puede obtener con la expresión

$$\mathbf{x} = \hat{\mathbf{x}} + \sum_{i=1}^{n \rightarrow \infty} \delta_i, \quad (3.28)$$

es decir, que a la solución aproximada le sumamos todos los «errores» para obtener la solución «exacta». Por supuesto, es imposible efectuar infinitas iteraciones, por lo que es imprescindible establecer algún criterio de corte. Un criterio puede ser cortar las iteraciones cuando $\|\mathbf{R}_k\| \leq Tol$, pero vimos que esto no asegura que el error sea pequeño. Otro criterio, tal vez más acertado, es interrumpir las iteraciones o cálculos cuando $\|\delta_k\| \leq Tol$, que tiene en cuenta el error de $\hat{\mathbf{x}}$. Usualmente, el criterio de interrupción se refiere a la norma infinita: $\|\delta_k\|_\infty \leq Tol$. (Algunos autores, en función de la potencia de cálculo actual, admiten la posibilidad de trabajar con todos los componentes de $\hat{\mathbf{x}}$.)

Este procedimiento que obtiene la solución de nuestro sistema sumando los errores, se conoce como *Método del Refinamiento Iterativo de la Solución* y ha cobrado gran desarrollo en los últimos años, pues pueden obtenerse buenos resultados con matrices mal condicionadas. Suele decirse que para obtener una buena solución, los sistemas $\mathbf{A}\delta_i = \mathbf{R}_i$ deben resolverse con mayor precisión que el sistema original. Si hemos resuelto el sistema $\mathbf{A}\mathbf{x} = \mathbf{B}$ en simple precisión, entonces debe usarse doble precisión para resolver cada uno de estos sistemas. Esto no es del todo cierto, ya que pueden obtenerse buenos resultados usando la misma precisión, tal como ha demostrado N. Higham (véase [11]). Pero existe otra cuestión. ¿Cuándo conviene aplicar este método?

Supongamos (una vez más) que obtenemos la aproximación $\hat{\mathbf{x}}$. Con esta solución, podemos obtener el vector residuo mediante

$$\mathbf{R}_1 = \mathbf{B} - \mathbf{A}\hat{\mathbf{x}}.$$

Si realizamos los cálculos utilizando una precisión de t dígitos, podemos demostrar que

$$\|\mathbf{R}_1\| \approx 10^{-t} \|\mathbf{A}\| \|\hat{\mathbf{x}}\|. \quad (3.29)$$

Para saber si el método es convergente, podemos obtener una aproximación o estimación del *número de condición* de \mathbf{A} . Para ello vamos a obtener el vector δ_1 según vimos arriba, es decir, haciendo

$$\mathbf{A}\delta_1 = \mathbf{R}_1.$$

Entonces, podemos escribir lo siguiente:

$$\begin{aligned} \|\delta_1\| &\approx \|\mathbf{x} - \hat{\mathbf{x}}\| = \|\mathbf{A}^{-1}\mathbf{R}_1\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{R}_1\| \approx \|\mathbf{A}^{-1}\| (10^{-t} \|\mathbf{A}\| \|\hat{\mathbf{x}}\|) \\ \|\delta_1\| &\approx 10^{-t} \|\hat{\mathbf{x}}\| \kappa(\mathbf{A}), \end{aligned} \quad (3.30)$$

con lo cual podemos estimar $\kappa(\mathbf{A})$ mediante

$$\kappa(\mathbf{A}) \approx \frac{\|\delta_1\|}{\|\hat{\mathbf{x}}\|} 10^t. \quad (3.31)$$

Como hemos dicho, este método permite obtener buenos resultados inclusive con matrices mal condicionadas. Sin embargo, si $\kappa(\mathbf{A}) \gg 10^t$, el sistema está tan mal condicionado que debe modificarse la precisión original usada en la obtención de $\hat{\mathbf{x}}$ para obtener un resultado aproximado aceptable.

3.9. Errores de los métodos directos

Hemos visto que el hecho de obtener un vector residuo pequeño no es garantía para inferir que el resultado obtenido tiene un error también pequeño. Analicemos el sistema en una forma más detallada. Supongamos ahora que tanto la matriz \mathbf{A} como el vector \mathbf{B} tienen pequeñas perturbaciones que llamaremos $\delta\mathbf{A}$ y $\delta\mathbf{B}$ respectivamente, y que nuestra solución sea $\hat{\mathbf{x}}$. Entonces tendremos:

$$(\mathbf{A} + \delta\mathbf{A})\hat{\mathbf{x}} = \mathbf{B} + \delta\mathbf{B}. \quad (3.32)$$

Podemos escribir que

$$\mathbf{A}\hat{\mathbf{x}} + \delta\mathbf{A}\hat{\mathbf{x}} = \mathbf{B} + \delta\mathbf{B}. \quad (3.33)$$

Sabemos que $\mathbf{x} = \hat{\mathbf{x}} + \delta\mathbf{x}$, por lo tanto podemos escribir:

$$\mathbf{A}(\mathbf{x} - \delta\mathbf{x}) + \delta\mathbf{A}(\mathbf{x} - \delta\mathbf{x}) = \mathbf{B} + \delta\mathbf{B}, \quad (3.34)$$

$$\mathbf{A}\mathbf{x} - \mathbf{A}\delta\mathbf{x} + \delta\mathbf{A}\mathbf{x} - \delta\mathbf{A}\delta\mathbf{x} = \mathbf{B} + \delta\mathbf{B}. \quad (3.35)$$

Si despreciamos $\delta\mathbf{A}\delta\mathbf{x}$, tendremos

$$\mathbf{A}\mathbf{x} + \mathbf{A}\delta\mathbf{x} - \delta\mathbf{A}\mathbf{x} = \mathbf{B} + \delta\mathbf{B}, \quad (3.36)$$

$$\mathbf{A}\delta\mathbf{x} - \delta\mathbf{A}\mathbf{x} = \delta\mathbf{B}, \quad (3.37)$$

$$\mathbf{A}\delta\mathbf{x} = \delta\mathbf{B} + \delta\mathbf{A}\mathbf{x}, \quad (3.38)$$

$$\delta\mathbf{x} = \mathbf{A}^{-1}\delta\mathbf{B} + \mathbf{A}^{-1}\delta\mathbf{A}\mathbf{x}. \quad (3.39)$$

Si tomamos normas a ambos lados tendremos:

$$\|\delta\mathbf{x}\| \leq \|\mathbf{A}^{-1}\| \|\delta\mathbf{B}\| + \|\mathbf{A}^{-1}\| \|\delta\mathbf{A}\| \|\mathbf{x}\|, \quad (3.40)$$

y como además tenemos que $\|\mathbf{B}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$, entonces podemos dividir todo de manera de obtener:

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{A}\| \|\mathbf{x}\|} \leq \frac{\|\mathbf{A}^{-1}\| \|\delta\mathbf{B}\|}{\|\mathbf{B}\|} + \frac{\|\mathbf{A}^{-1}\| \|\delta\mathbf{A}\| \|\mathbf{x}\|}{\|\mathbf{A}\| \|\mathbf{x}\|}, \quad (3.41)$$

$$\leq \frac{\|\mathbf{A}^{-1}\| \|\delta\mathbf{B}\|}{\|\mathbf{B}\|} + \frac{\|\mathbf{A}^{-1}\| \|\delta\mathbf{A}\|}{\|\mathbf{A}\|}. \quad (3.42)$$

Si multiplicamos por $\|\mathbf{A}\|$ tendremos que:

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \frac{\|\mathbf{A}^{-1}\| \|\mathbf{A}\|}{\|\mathbf{B}\|} \|\delta\mathbf{B}\| + \|\mathbf{A}^{-1}\| \|\delta\mathbf{A}\| \frac{\|\mathbf{A}\|}{\|\mathbf{A}\|} \quad (3.43)$$

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \underbrace{\|\mathbf{A}\| \|\mathbf{A}^{-1}\|}_{\text{cond}(\mathbf{A})} \left(\frac{\|\delta\mathbf{B}\|}{\|\mathbf{B}\|} + \frac{\|\delta\mathbf{A}\|}{\|\mathbf{A}\|} \right) \quad (3.44)$$

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \text{cond}(\mathbf{A}) \left(\frac{\|\delta\mathbf{B}\|}{\|\mathbf{B}\|} + \frac{\|\delta\mathbf{A}\|}{\|\mathbf{A}\|} \right). \quad (3.45)$$

Podemos ver que para que los errores de \mathbf{x} sean pequeños no basta con que $\delta\mathbf{B}$ y $\delta\mathbf{A}$ sean pequeños (es decir, que los errores inherentes sean pequeños), sino que es necesario que el número de condición de \mathbf{A} ($\text{cond}(\mathbf{A})$) sea cercano a 1.

Analicemos ahora los errores de redondeo. Vamos a buscar una cota de estos errores. Supongamos que aplicamos el método de factorización \mathbf{LU} para resolver el sistema. Si suponemos que solamente se producen errores de redondeo, entonces tendremos en realidad que

$$\mathbf{LU} = \mathbf{A} + \delta\mathbf{A},$$

donde $\delta\mathbf{A}$ son las perturbaciones producidas por los errores de redondeo al obtener \mathbf{L} y \mathbf{U} . Entonces, nuestro sistema queda como:

$$(\mathbf{A} + \delta\mathbf{A})(\mathbf{x} - \delta\mathbf{x}) = \mathbf{B} \quad (3.46)$$

$$\mathbf{Ax} - \mathbf{A}\delta\mathbf{x} + \delta\mathbf{Ax} - \delta\mathbf{A}\delta\mathbf{x} = \mathbf{B} \quad (3.47)$$

$$\mathbf{A}\delta\mathbf{x} = \delta\mathbf{A}(\mathbf{x} - \delta\mathbf{x}). \quad (3.48)$$

por lo tanto,

$$\delta\mathbf{x} = \mathbf{A}^{-1}\delta\mathbf{A}(\mathbf{x} - \delta\mathbf{x}). \quad (3.49)$$

Si tomamos la norma infinito tenemos

$$\|\delta\mathbf{x}\|_{\infty} \leq \|\mathbf{A}^{-1}\|_{\infty} \|\delta\mathbf{A}\|_{\infty} \|\mathbf{x} - \delta\mathbf{x}\|_{\infty} \quad (3.50)$$

$$\leq \|\mathbf{A}^{-1}\|_{\infty} \|\mathbf{A}\|_{\infty} \|\vec{x} - \delta\mathbf{x}\|_{\infty} \frac{\|\delta\mathbf{A}\|_{\infty}}{\|\mathbf{A}\|_{\infty}} \quad (3.51)$$

$$\|\delta\mathbf{x}\|_{\infty} \leq \kappa(\mathbf{A}) \|\mathbf{x} - \delta\mathbf{x}\|_{\infty} \frac{\|\delta\mathbf{A}\|_{\infty}}{\|\mathbf{A}\|_{\infty}}. \quad (3.52)$$

Se puede demostrar que $\|\delta\mathbf{A}\|_{\infty} \leq 1,01(n^3 + 3n^2)\rho \|\mathbf{A}\|_{\infty} \mu$, donde $\rho = \max \frac{|a_{ij}^k|}{\|\mathbf{A}\|_{\infty}}$, n es la dimensión de la matriz \mathbf{A} y μ es la unidad de máquina; entonces tenemos que

$$\frac{\|\delta\mathbf{x}\|_{\infty}}{\|\mathbf{x} - \delta\mathbf{x}\|_{\infty}} \approx \frac{\|\delta\mathbf{x}\|_{\infty}}{\|\mathbf{x}\|_{\infty}} \leq \kappa(\mathbf{A}) 1,01(n^3 + 3n^2)\rho \mu, \quad (3.53)$$

y entonces definimos el error total para los métodos directos como

$$\frac{\|\delta\mathbf{x}\|_{\infty}}{\|\mathbf{x}\|_{\infty}} \leq \kappa(\mathbf{A}) \left[\frac{\|\delta\mathbf{B}\|_{\infty}}{\|\mathbf{B}\|_{\infty}} + \frac{\|\delta\mathbf{A}\|_{\infty}}{\|\mathbf{A}\|_{\infty}} + 1,01(n^3 + 3n^2)\rho \mu \right]. \quad (3.54)$$

Podemos ver que si la matriz es de grandes dimensiones, comienzan a tener gran incidencia los errores de redondeo, con lo cual el sistema puede volverse inestable si $\kappa(\mathbf{A}) \gg 1$, además de mal condicionado.

3.10. Métodos iterativos

Hasta ahora hemos estudiado los llamados *métodos directos* para resolver sistemas de ecuaciones lineales. Son llamados de esta forma porque el algoritmo tiene una cantidad conocida («finita») de pasos y los resultados que obtenemos al aplicarlos deberían ser exactos, salvo por el error de redondeo, aunque vimos que esto no siempre es así. Estos métodos se suelen usar con matrices densas o casi llenas, como por ejemplo las surgidas del análisis matricial de estructuras planas, las cuales tienen muchos coeficientes distintos de cero ($a_{ij} \neq 0$).

Pero existen muchos otros problemas en los cuales el sistema de ecuaciones tiene una matriz \mathbf{A} que no es densa, sino por el contrario, es *rala*, es decir, tiene muchos coeficientes nulos,

como es el caso del análisis estructural en tres dimensiones. Entonces trabajar con los métodos directos se vuelve muy poco práctico, pues debemos hacer muchas operaciones con coeficientes nulos y, lo que es peor, muchas veces transformar un coeficiente nulo en otro no nulo, incorporando un error que antes no existía. Es por eso que se han desarrollado métodos que tienen en cuenta este tipo de matrices. Son los métodos denominados *iterativos*.

En estos métodos, la solución la obtenemos a partir de una solución inicial, la cual se va *corrigiendo* en sucesivas iteraciones hasta obtener la solución «correcta», de ahí el nombre de iterativos. En principio, podemos suponer que la cantidad de iteraciones es «infinita», es decir, que la solución exacta la obtenemos luego de infinitas iteraciones. Como efectuar esto es imposible, lo que se hace es iterar hasta que la solución esté dentro de las tolerancias impuestas.

Para analizar estos métodos partamos de definirlos en forma matricial. Sabemos que nuestro sistema se expresa como

$$\mathbf{A} \mathbf{x} = \mathbf{B},$$

o, lo que es lo mismo, como

$$\mathbf{B} - \mathbf{A} \mathbf{x} = 0. \quad (3.55)$$

En consecuencia, podemos sumar en ambos miembros $\mathbf{M} \mathbf{x}$ sin cambiar la igualdad. Nos queda que

$$\mathbf{M} \mathbf{x} = \mathbf{M} \mathbf{x} - \mathbf{A} \mathbf{x} + \mathbf{B} \Rightarrow \mathbf{M} \mathbf{x} = (\mathbf{M} - \mathbf{A}) \mathbf{x} + \mathbf{B}. \quad (3.56)$$

Si despejamos \mathbf{x} de la expresión anterior, nos queda:

$$\mathbf{x} = \mathbf{M}^{-1} (\mathbf{M} - \mathbf{A}) \mathbf{x} + \mathbf{M}^{-1} \mathbf{B}, \quad (3.57)$$

que puede escribirse como

$$\mathbf{x} = (\mathbf{M}^{-1} \mathbf{M} - \mathbf{M}^{-1} \mathbf{A}) \mathbf{x} + \mathbf{M}^{-1} \mathbf{B} = (\mathbf{I} - \mathbf{M}^{-1} \mathbf{A}) \mathbf{x} + \mathbf{M}^{-1} \mathbf{B}, \quad (3.58)$$

a partir del cual se puede obtener el método iterativo para resolver un sistema de ecuaciones, que toma la siguiente forma:

$$\mathbf{x}^{(n+1)} = (\mathbf{I} - \mathbf{M}^{-1} \mathbf{A}) \mathbf{x}^{(n)} + \mathbf{M}^{-1} \mathbf{B}, \quad (3.59)$$

donde n es la iteración.

La expresión anterior puede escribirse en forma general como

$$\mathbf{x}^{(n+1)} = \mathbf{T} \mathbf{x}^{(n)} + \mathbf{C}, \quad (3.60)$$

donde

$$\mathbf{T} = \mathbf{I} - \mathbf{M}^{-1} \mathbf{A} \quad \text{y} \quad \mathbf{C} = \mathbf{M}^{-1} \mathbf{B}.$$

Con esta última expresión podemos definir dos tipos de métodos iterativos: los estacionarios, aquellos en los que \mathbf{T} y \mathbf{C} no sufren modificaciones durante las iteraciones, y los no estacionarios, aquellos en los que los valores de \mathbf{T} y \mathbf{C} dependen de la iteración.

3.10.1. Métodos estacionarios

Como hemos visto, los métodos iterativos estacionarios son aquellos en los que \mathbf{T} y \mathbf{C} son *invariantes*, es decir, permanecen constantes en las sucesivas iteraciones necesarias para hallar la solución.

Supongamos por un momento que conocemos nuestra solución «exacta» $\bar{\mathbf{x}}$. Entonces podemos decir que:

$$\begin{aligned}\bar{\mathbf{x}} &= \mathbf{x}^{(n+1)} + \mathbf{e}^{(n+1)} \Rightarrow \\ \mathbf{x}^{(n+1)} + \mathbf{e}^{(n+1)} &= \mathbf{T} \left(\mathbf{x}^{(n)} + \mathbf{e}^{(n)} \right) + \mathbf{C} \\ \mathbf{x}^{(n+1)} + \mathbf{e}^{(n+1)} &= \underbrace{\mathbf{T} \mathbf{x}^{(n)} + \mathbf{C}}_{\mathbf{x}^{(n+1)}} + \mathbf{T} \mathbf{e}^{(n)} \\ \mathbf{x}^{(n+1)} + \mathbf{e}^{(n+1)} &= \mathbf{x}^{(n+1)} + \mathbf{T} \mathbf{e}^{(n)} \Rightarrow \\ \mathbf{e}^{(n+1)} &= \mathbf{T} \mathbf{e}^{(n)}.\end{aligned}\tag{3.61}$$

De la última expresión podemos deducir que:

$$\mathbf{e}^{(n+1)} = \mathbf{T} \mathbf{e}^{(n)} = \mathbf{T} \mathbf{T} \mathbf{e}^{(n-1)} = \mathbf{T}^2 \mathbf{e}^{(n-1)} = \dots = \mathbf{T}^{n+1} \mathbf{e}^{(0)},\tag{3.62}$$

expresión que nos indica que para que un método iterativo estacionario sea convergente se debe cumplir que $\|\mathbf{T}\| < 1$, y que $\|\mathbf{T}\| \ll 1$ para que la convergencia sea rápida.

Método de Jacobi

El método estacionario más sencillo es el *Método de Jacobi*. Podemos descomponer \mathbf{A} como $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$. Este método es el que define $\mathbf{M} = \mathbf{D}$, y por lo tanto, nuestra ecuación quedará como:

$$\begin{aligned}\mathbf{x}^{(n+1)} &= (\mathbf{I} - \mathbf{D}^{-1} \mathbf{A}) \mathbf{x}^{(n)} + \mathbf{D}^{-1} \mathbf{B} \\ &= [\mathbf{I} - \mathbf{D}^{-1}(\mathbf{L} + \mathbf{D} + \mathbf{U})] \mathbf{x}^{(n)} + \mathbf{D}^{-1} \mathbf{B} \\ &= [\mathbf{I} - \underbrace{\mathbf{D}^{-1} \mathbf{D}}_{\mathbf{I}} - \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})] \mathbf{x}^{(n)} + \mathbf{D}^{-1} \mathbf{B} \\ \mathbf{x}^{(n+1)} &= \mathbf{D}^{-1} [\mathbf{B} - (\mathbf{L} + \mathbf{U}) \mathbf{x}^{(n)}],\end{aligned}\tag{3.63}$$

donde \mathbf{L} , \mathbf{D} y \mathbf{U} son:

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ a_{21} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ a_{m1} & \dots & a_{mm-1} & 0 \end{bmatrix}, \mathbf{D} = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{mm} \end{bmatrix} \text{ y } \mathbf{U} = \begin{bmatrix} 0 & a_{12} & \dots & a_{1m} \\ \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & a_{m-1m} \\ 0 & \dots & 0 & 0 \end{bmatrix}.$$

En su forma tradicional este método se expresa como:

$$x_i^{(n+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(n)} - \sum_{j=i+1}^n a_{ij} x_j^{(n)}}{a_{ii}}.\tag{3.64}$$

En sí, el método consiste en suponer una solución inicial, generalmente el vector nulo ($\mathbf{x} = [0]$), e iterar hasta obtener la solución, usando siempre el vector obtenido en el paso anterior. Para analizar la convergencia, debemos recordar algunas definiciones.

Definición 3.9. Una matriz \mathbf{A} se denomina *diagonal dominante* si se cumple que

$$|a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|.$$

Definición 3.10. Una matriz \mathbf{A} se denomina *estrictamente diagonal dominante* si se cumple que

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|.$$

Definición 3.11. Una matriz \mathbf{A} se denomina *diagonal dominante en forma irreductible* si se cumple que

$$|a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|,$$

para $i = 1; 2; \dots; n$ y en al menos una fila que

$$|a_{kk}| > \sum_{\substack{j=1 \\ j \neq k}}^n |a_{kj}|.$$

El *Método de Jacobi* converge rápidamente si la matriz \mathbf{A} es *estrictamente diagonal dominante*, como se verá más adelante. En cambio, la convergencia es lenta si la matriz \mathbf{A} es cualquiera de las otras dos. Finalmente, si la matriz \mathbf{A} no cumple con ninguna de las definiciones anteriores, el método de Jacobi no converge.

Método de Gauss-Seidel

El *Método de Jacobi* es de convergencia muy lenta. Para mejorar esta velocidad de convergencia, imaginemos que usamos parte de los resultados ya obtenidos para obtener los siguientes, es decir, obtener el x_i aprovechando los x_j para $j < i$. Este método se conoce como *Método de Gauss-Seidel* y resulta de definir $\mathbf{M} = \mathbf{D} + \mathbf{L}$. Desarrollemos la expresión final sabiendo que $\mathbf{M} \mathbf{x}^{(n+1)} = \mathbf{M} \mathbf{x}^{(n)} - \mathbf{A} \mathbf{x}^{(n)} + \mathbf{B}$:

$$\begin{aligned} (\mathbf{D} + \mathbf{L}) \mathbf{x}^{(n+1)} &= [(\mathbf{D} + \mathbf{L}) - \mathbf{A}] \mathbf{x}^{(n)} + \mathbf{B} \\ &= [(\mathbf{D} + \mathbf{L}) - (\mathbf{L} + \mathbf{D} + \mathbf{U})] \mathbf{x}^{(n)} + \mathbf{B} \\ &= [\mathbf{D} + \mathbf{L} - \mathbf{L} - \mathbf{D} - \mathbf{U}] \mathbf{x}^{(n)} + \mathbf{B} \\ &= \mathbf{B} - \mathbf{U} \mathbf{x}^{(n)} \Rightarrow \end{aligned} \tag{3.65}$$

$$\begin{aligned} \mathbf{D} \mathbf{x}^{(n+1)} &= \mathbf{B} - \mathbf{L} \mathbf{x}^{(n+1)} - \mathbf{U} \mathbf{x}^{(n)} \Rightarrow \\ \mathbf{x}^{(n+1)} &= \mathbf{D}^{-1} [\mathbf{B} - \mathbf{L} \mathbf{x}^{(n+1)} - \mathbf{U} \mathbf{x}^{(n)}]. \end{aligned}$$

En su forma tradicional el método se escribe de la siguiente manera:

$$x_i^{(n+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(n+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(n)}}{a_{ii}}. \tag{3.66}$$

En principio, los métodos de *Jacobi* y *Gauss-Seidel* convergen cuando la matriz \mathbf{A} es *estrictamente diagonal dominante*. Más adelante se analizará en detalle la convergencia de los métodos iterativos estacionarios. Pero es cierto que generalmente el método de *Gauss-Seidel* es mejor (converge más rápido) que el método de *Jacobi*. Sin embargo, hay casos en los cuales el método de *Jacobi* converge y el de *Gauss-Seidel* no, y casos en los cuales el método de *Jacobi* no converge y el de *Gauss-Seidel* sí.

Método de las sobrerrelajaciones sucesivas (SOR)

Si bien *Gauss-Seidel* es más rápido que *Jacobi*, la velocidad de convergencia no es muy buena. Busquemos algún método que nos mejore esta velocidad. Partamos nuevamente de la expresión general $\mathbf{M} \mathbf{x}^{(n+1)} = \mathbf{M} \mathbf{x}^{(n)} - \mathbf{A} \mathbf{x}^{(n)} + \mathbf{B}$. Si reordenamos un poco la expresión tenemos:

$$\mathbf{M} \mathbf{x}^{(n+1)} = \mathbf{M} \mathbf{x}^{(n)} + \underbrace{\mathbf{B} - \mathbf{A} \mathbf{x}^{(n)}}_{\mathbf{R}^{(n)}} = \mathbf{M} \mathbf{x}^{(n)} + \mathbf{R}^{(n)}, \quad (3.67)$$

que podemos escribir también como

$$\mathbf{x}^{(n+1)} = \underbrace{\mathbf{M}^{-1} \mathbf{M}}_{\mathbf{I}} \mathbf{x}^{(n)} + \mathbf{M}^{-1} \mathbf{R}^{(n)} = \mathbf{x}^{(n)} + \mathbf{M}^{-1} \mathbf{R}^{(n)}. \quad (3.68)$$

La idea es buscar una matriz \mathbf{M} que nos mejore la velocidad de convergencia. Supongamos, entonces, que tomamos $\mathbf{M} = \mathbf{L} + \frac{1}{\omega} \mathbf{D}$. Si partimos de la expresión conocida tenemos que:

$$\begin{aligned} \left(\frac{1}{\omega} \mathbf{D} + \mathbf{L}\right) \mathbf{x}^{(n+1)} &= \left[\left(\frac{1}{\omega} \mathbf{D} + \mathbf{L}\right) - \mathbf{A}\right] \mathbf{x}^{(n)} + \mathbf{B} \\ &= \left[\left(\frac{1}{\omega} \mathbf{D} + \mathbf{L}\right) - (\mathbf{L} + \mathbf{D} + \mathbf{U})\right] \mathbf{x}^{(n)} + \mathbf{B} \\ &= \left[\frac{1}{\omega} \mathbf{D} + \mathbf{L} - \mathbf{L} - \mathbf{D} - \mathbf{U}\right] \mathbf{x}^{(n)} + \mathbf{B} \\ &= \mathbf{B} - \left(1 - \frac{1}{\omega}\right) \mathbf{D} \mathbf{x}^{(n)} - \mathbf{U} \mathbf{x}^{(n)} \Rightarrow \\ \frac{1}{\omega} \mathbf{D} \mathbf{x}^{(n+1)} &= \mathbf{B} - \mathbf{L} \mathbf{x}^{(n+1)} - \left(1 - \frac{1}{\omega}\right) \mathbf{D} \mathbf{x}^{(n)} - \mathbf{U} \mathbf{x}^{(n)} \Rightarrow \\ \mathbf{x}^{(n+1)} &= -\omega \left(1 - \frac{1}{\omega}\right) \underbrace{\mathbf{D}^{-1} \mathbf{D}}_{\mathbf{I}} \mathbf{x}^{(n)} + \omega \mathbf{D}^{-1} \left[\mathbf{B} - \mathbf{L} \mathbf{x}^{(n+1)} - \mathbf{U} \mathbf{x}^{(n)}\right] \\ &= (1 - \omega) \mathbf{x}^{(n)} + \omega \mathbf{D}^{-1} \left[\mathbf{B} - \mathbf{L} \mathbf{x}^{(n+1)} - \mathbf{U} \mathbf{x}^{(n)}\right] \\ \mathbf{x}^{(n+1)} &= (1 - \omega) \mathbf{x}^{(n)} + \omega \mathbf{x}_{GS}^{(n+1)}. \end{aligned} \quad (3.69)$$

Este método se conoce como *Método de las Sobrerrelajaciones Sucesivas* (o SOR por sus siglas en inglés), y pondera el $\mathbf{x}^{(n)}$ con el $\mathbf{x}^{(n+1)}$ obtenido con el *Método de Gauss-Seidel*, tomando como factor de ponderación el coeficiente ω . En su forma tradicional se suele escribir como:

$$x_i^{(n+1)} = (1 - \omega) x_i^{(n)} + \omega \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(n+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(n)}}{a_{ii}}. \quad (3.70)$$

En este método la velocidad de convergencia está dada por el ω . Se puede asegurar que existe un valor que hace máxima la velocidad de convergencia para un sistema dado, que puede ser estimado conociendo el radio espectral de la matriz del *Método de Jacobi*. Si observamos con detenimiento veremos que el *Método de Gauss-Seidel* es un caso especial del SOR, pues surge de tomar $\omega = 1$. En efecto, si $\omega = 1$ tenemos:

$$\begin{aligned}
 x_i^{(n+1)} &= (1 - \omega)x_i^{(n)} + \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(n+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(n)}}{a_{ii}} \\
 &= \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(n+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(n)}}{a_{ii}},
 \end{aligned} \tag{3.71}$$

que es el *Método de Gauss-Seidel*.

En realidad, al imponer que $0 < \omega < 2$ existen dos métodos: cuando $0 < \omega < 1$, estamos en presencia de un método de *sub-relajación*, también conocido como *Método de Jacobi Modificado*, en tanto que cuando $1 < \omega < 2$, se trata de un método de *Sobrerrelajación* propiamente dicho. En general, estos métodos convergen mucho más rápido que los otros dos, y puede decirse que cuando *Gauss-Seidel* no converge, utilizando un $\omega < 1$ se logra una mejor convergencia que con el *Método de Jacobi*.

Criterios de interrupción

Hasta acá hemos visto los distintos métodos iterativos estacionarios más tradicionales que se aplican para resolver sistemas de ecuaciones lineales. Pero no hemos analizado los criterios para interrumpir dichas iteraciones. Dado que los métodos convergen a una solución cuando $n \rightarrow \infty$, es decir, que se debe dar que $\mathbf{x} - \mathbf{x}^{(n)} = 0$ cuando $n \rightarrow \infty$, entonces podemos tomar como criterios para interrumpir las iteraciones, que $\mathbf{x} - \mathbf{x}^{(n)} < Tol$, siendo *Tol* un valor definido arbitrariamente, generalmente relacionado con la precisión utilizada (μ). Existen varios criterios que pueden aplicarse. Estos son:

1. Que la norma infinita del vector $\mathbf{r}^{(n)}$ sea menor a la tolerancia, esto es:

$$\|\mathbf{r}^{(n)}\|_{\infty} < Tol. \tag{3.72}$$

2. Que la norma infinita del error absoluto entre dos soluciones sucesivas de \mathbf{x} sea menor a la tolerancia, es decir, que:

$$\|\mathbf{x}^{(n)} - \mathbf{x}^{(n-1)}\|_{\infty} < Tol. \tag{3.73}$$

3. Que la norma infinita del error relativo entre dos soluciones sucesivas sea menor a la tolerancia, o sea:

$$\frac{\|\mathbf{x}^{(n)} - \mathbf{x}^{(n-1)}\|_{\infty}}{\|\mathbf{x}^{(n)}\|_{\infty}} < Tol. \tag{3.74}$$

El mejor de los criterios es último, pues hemos visto que es el error relativo el que mejor representa la incidencia del error en los resultados.

Sin embargo, debemos recordar del análisis de la cota de error para los métodos directos que

$$\frac{\|\mathbf{x} - \tilde{\mathbf{x}}\|}{\|\mathbf{x}\|} \leq \|\mathbf{A}^{-1}\| \|\mathbf{A}\| \frac{\|\mathbf{R}\|}{\|\mathbf{B}\|}$$

expresión que puede ampliarse al caso de un método iterativo como

$$\begin{aligned}
 \frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|}{\|\mathbf{x}^{(k+1)}\|} &\leq \|\mathbf{A}^{-1}\| \|\mathbf{A}\| \frac{\|\mathbf{R}\|}{\|\mathbf{B}\|} \\
 \frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|}{\|\mathbf{x}^{(k+1)}\|} &\leq \kappa(\mathbf{A}) \frac{\|\mathbf{R}\|}{\|\mathbf{B}\|} \approx \kappa(\mathbf{A}) Tol.
 \end{aligned}$$

con lo cual debemos cuidarnos al momento de elegir la tolerancia cuando aplicamos un método iterativo. Queda evidente que cuando la matriz tiende a ser mal condicionada, la tolerancia debe ser más chica.

3.10.2. Convergencia de los métodos estacionarios

Hemos dicho que los métodos de *Jacobi* y *Gauss-Seidel* convergen para matrices \mathbf{A} estrictamente diagonal dominantes. Los siguientes teoremas aseguran la convergencia de ambos métodos.

Teorema 3.3. Si \mathbf{A} es una matriz de $n \times n$, entonces se cumple que:

1. $\|\mathbf{A}\|_2 = [\rho(\mathbf{A}^T \mathbf{A})]^{1/2}$.
2. $\rho(\mathbf{A}) \leq \|\mathbf{A}\|$, para toda norma natural.

Teorema 3.4. Si la matriz \mathbf{A} es estrictamente diagonal dominante, entonces con cualquier elección de $\mathbf{x}^{(0)}$, tanto el *Método de Jacobi* como el de *Gauss-Seidel* dan las sucesiones $\{\mathbf{x}^{(k)}\}_{k=0}^{\infty}$ que convergen a una única solución del sistema $\mathbf{A} \mathbf{x} = \mathbf{B}$.

Teorema 3.5. Si $a_{ij} \leq 0$ para cada $i \neq j$, y si $a_{ii} > 0$ para cada $i = 1; 2; \dots; n$, entonces será válida una y sólo una de las siguientes afirmaciones:

1. $0 \leq \rho(\mathbf{T}_G) < \rho(\mathbf{T}_J) < 1$;
2. $1 < \rho(\mathbf{T}_J) < \rho(\mathbf{T}_G)$;
3. $\rho(\mathbf{T}_J) = \rho(\mathbf{T}_G) = 0$;
4. $\rho(\mathbf{T}_J) = \rho(\mathbf{T}_G) = 1$;

donde \mathbf{T}_J es la matriz de Jacobi, y \mathbf{T}_G es la matriz de Gauss-Seidel.

Para analizar la convergencia del método de las sobre-relajaciones sucesivas se deben tener en cuenta estos otros teoremas.

Teorema 3.6. Para cualquier $\mathbf{x}^{(0)} \in \mathfrak{R}^n$, la sucesión $\{\mathbf{x}^{(k)}\}_{k=0}^{\infty}$ definida por

$$\mathbf{x}^{(k+1)} = \mathbf{T} \mathbf{x}^{(k)} + \mathbf{C}, \text{ para cada } k \geq 1,$$

converge en la solución única de $\mathbf{x} = \mathbf{T} \mathbf{x} + \mathbf{C}$ si y sólo si $\rho(\mathbf{T}) < 1$.

Este teorema nos dice que cualquier método iterativo converge cuando el radio espectral de la matriz \mathbf{T} es menor a 1, tal como vimos al comenzar. Recordemos que la definición del radio espectral de una matriz \mathbf{A} cualquiera es

$$\rho(\mathbf{A}) = \max |\lambda|, \tag{3.75}$$

donde λ es un autovalor de \mathbf{A} . En efecto, habíamos dicho que para que cualquier método iterativo sea convergente, se debía cumplir que $\|\mathbf{T}\| < 1$. Como $\rho(\mathbf{T}) \leq \|\mathbf{T}\| < 1$, si los módulos de los autovalores de \mathbf{T} son menores que 1, entonces los método convergen a la solución buscada.

Teorema 3.7. Si \mathbf{A} es una matriz definida positiva y si $0 < \omega < 2$, entonces el *Método SOR* converge para cualquier elección del vector aproximado $\mathbf{x}^{(0)}$.

Este teorema lo podemos aplicar también al *Método de Gauss-Seidel*. Efectivamente, puesto que cuando $\omega = 1$, el *Método SOR* resulta ser el de *Gauss-Seidel*, y como el teorema 3.7 asegura la convergencia del *Método SOR* para cualquier vector inicial cuando $0 < \omega < 2$, entonces asegura también la convergencia del *Método de Gauss-Seidel* cuando la matriz \mathbf{A} es definida positiva. Este teorema puede ampliarse a matrices simétricas definidas positivas.

Teorema 3.8. Si \mathbf{A} es una matriz definida positiva y tridiagonal, entonces $\rho(\mathbf{T}_G) = [\rho(\mathbf{T}_J)]^2 < 1$, y la elección óptima de ω para el *Método SOR* es

$$\omega = \frac{2}{1 + \sqrt{1 - [\rho(\mathbf{T}_J)]^2}}. \quad (3.76)$$

Este último vincula los autovalores de la matriz \mathbf{T}_J , es decir la matriz \mathbf{T} del *Método de Jacobi*, con el valor de ω . Aunque se refiere a una matriz tridiagonal, es posible ver que cuanto menor sea el valor de $\rho(\mathbf{T}_J)$ más se acerca ω a 1. (Si $\rho(\mathbf{T}_J)^2$ es mayor que uno, entonces no hay un ω real que haga convergente al método.)

3.10.3. Métodos no estacionarios

Vimos en el punto anterior los métodos estacionarios, aquellos en los cuales las matrices \mathbf{T} y \mathbf{C} se mantienen invariantes en las sucesivas iteraciones. Existen otros métodos en los cuales estas dos matrices sí se van modificando en las sucesivas iteraciones. Son los llamados *métodos no estacionarios*.

Supongamos que en nuestra expresión general definimos que $\mathbf{M} = \frac{1}{\alpha}\mathbf{I}$. Si reemplazamos obtenemos:

$$\begin{aligned} \mathbf{x}^{(i+1)} &= (\mathbf{I} - \alpha\mathbf{I} \cdot \mathbf{A}) \mathbf{x}^{(i)} + \alpha\mathbf{I} \cdot \mathbf{B}, \\ &= \mathbf{x}^{(i)} + \alpha (\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i)}), \\ &= \mathbf{x}^{(i)} + \alpha\mathbf{R}^{(i)}. \end{aligned} \quad (3.77)$$

Tenemos ahora un método iterativo que depende de un parámetro α para ir corrigiendo el vector solución. Nos falta definir ese parámetro. Pero también depende de otro vector, el ya visto *residuo*. Por lo tanto tenemos dos elementos que podemos manejar para obtener una mejor aproximación. Veremos a continuación algunos de los métodos no estacionarios más sencillos que han servido de base para el desarrollo de los más modernos y complejos.

Método de los residuos mínimos

Una primera aproximación para esta expresión es buscar que el vector $\mathbf{R}^{(i+1)}$ sea mínimo en cada iteración. De esta manera siempre tenderemos a la solución del sistema, pues el ideal es que sea nulo. Una forma de obtener el mínimo es minimizar la norma euclídea, es decir, el módulo de $\mathbf{R}^{(i+1)}$. Partamos precisamente de la definición del módulo:

$$\|\mathbf{R}^{(i+1)}\|_2 = \|\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i+1)}\|_2. \quad (3.78)$$

Si lo elevamos al cuadrado tenemos

$$\begin{aligned} \|\mathbf{R}^{(i+1)}\|_2^2 &= \|\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i+1)}\|_2^2 \\ \|\mathbf{R}^{(i+1)}\|_2^2 &= \|\mathbf{B} - \mathbf{A} (\mathbf{x}^{(i)} + \alpha\mathbf{R}^{(i)})\|_2^2 \\ \mathbf{R}^{(i+1)T} \cdot \mathbf{R}^{(i+1)} &= [\mathbf{B} - \mathbf{A} (\mathbf{x}^{(i)} + \alpha\mathbf{R}^{(i)})]^T \cdot [\mathbf{B} - \mathbf{A} (\mathbf{x}^{(i)} + \alpha\mathbf{R}^{(i)})]. \end{aligned} \quad (3.79)$$

Como queremos minimizar el módulo de $\mathbf{R}^{(i+1)}$, lo mismo es minimizar el cuadrado del módulo. Para ello vamos a derivar la última expresión respecto de α , que es nuestro parámetro,

y lo igualaremos a cero. Así tenemos que:

$$\begin{aligned}
 -2 \left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \left[\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i)} - \alpha \mathbf{A} \cdot \mathbf{R}^{(i)} \right] &= 0 \\
 \left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \left[\mathbf{R}^{(i)} - \alpha \mathbf{A} \cdot \mathbf{R}^{(i)} \right] &= 0 \\
 \left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{R}^{(i)} &= \alpha \left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{A} \cdot \mathbf{R}^{(i)} \Rightarrow \\
 \alpha_i &= \frac{\left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{R}^{(i)}}{\left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}}.
 \end{aligned} \tag{3.80}$$

Este coeficiente α_i nos asegura que el residuo sea mínimo. Así nuestro esquema iterativo queda de la siguiente forma:

$$\begin{aligned}
 \mathbf{R}^{(i)} &= \mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i)} \\
 \alpha_i &= \frac{\left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{R}^{(i)}}{\left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \\
 \mathbf{x}^{(i+1)} &= \mathbf{x}^{(i)} + \alpha_i \cdot \mathbf{R}^{(i)}.
 \end{aligned} \tag{3.81}$$

Este método es convergente si la matriz \mathbf{A} es simétrica definida positiva ⁵, pues de lo contrario no obtendremos un mínimo. (En [16] puede verse una demostración de esta afirmación.)

Existe un segundo algoritmo que tiene la siguiente forma:

$$\begin{aligned}
 \mathbf{R}^{(0)} &= \mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(0)} \\
 \alpha_i &= \frac{\left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{R}^{(i)}}{\left(\mathbf{A} \cdot \mathbf{R}^{(i)} \right)^T \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \\
 \mathbf{x}^{(i+1)} &= \mathbf{x}^{(i)} + \alpha_i \mathbf{R}^{(i)} \\
 \mathbf{R}^{(i+1)} &= \mathbf{R}^{(i)} - \alpha_i \mathbf{A} \cdot \mathbf{R}^{(i)}.
 \end{aligned} \tag{3.82}$$

En ambos algoritmos las iteraciones finalizan cuando $\mathbf{R}^{(i+1)} < Tol$, pues $\mathbf{R}^{(n)} = 0$ para $n \rightarrow \infty$.

Método del descenso más rápido

Un segundo método no estacionario es el denominado *Método del descenso más rápido*. Este método mejora la aproximación obtenida en el punto anterior. Para poder deducirlo antes necesitamos recordar qué es una forma cuadrática.

Forma cuadrática: Es una función vectorial que se expresa como:

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{B}^T \mathbf{x} + C, \tag{3.83}$$

similar a una ecuación de segundo grado en el campo escalar, donde \mathbf{A} es una matriz, \mathbf{x} y \mathbf{B} son vectores y C es una constante (escalar). La figura 3.1 ilustra una forma cuadrática en dos dimensiones.

⁵Algunos autores sólo exigen que la matriz \mathbf{A} sea definida positiva.

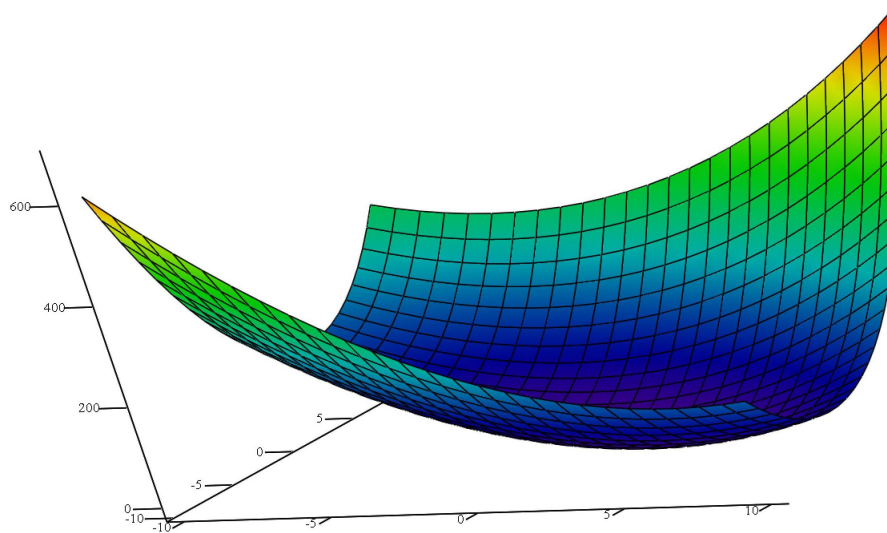


Figura 3.1: Forma cuadrática en dos dimensiones.

Supongamos ahora que queremos hallar el mínimo (o máximo) de esta función. Entonces debemos obtener su derivada e igualarla a cero, es decir, hacer que:

$$\frac{d f(\mathbf{x})}{d\mathbf{x}} = \frac{1}{2} \mathbf{A}^T \mathbf{x} + \frac{1}{2} \mathbf{A} \mathbf{x} - \mathbf{B} = 0. \quad (3.84)$$

Si \mathbf{A} es una matriz simétrica entonces $\mathbf{A} = \mathbf{A}^T$, y podemos escribir:

$$\frac{d f(\mathbf{x})}{d\mathbf{x}} = \mathbf{A} \mathbf{x} - \mathbf{B} = 0, \quad (3.85)$$

que no es otra cosa que nuestro sistema de ecuaciones lineales original. Si además \mathbf{A} es definida positiva, nos aseguramos que la solución que se obtenga haga mínima a la forma cuadrática. En consecuencia, para aplicar este método, la matriz \mathbf{A} también debe ser *simétrica definida positiva*.

Recordemos también qué es el gradiente de una función vectorial. Para una función $f(\mathbf{x})$ el gradiente se expresa como:

$$\frac{d f(\mathbf{x})}{d\mathbf{x}} = f'(\mathbf{x}) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \frac{\partial f(\mathbf{x})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\mathbf{x})}{\partial x_n} \end{bmatrix} \quad (3.86)$$

El gradiente nos da una idea de la «pendiente» o del crecimiento de la forma cuadrática, que también podemos representar como un plano normal a dicho vector gradiente (figura 3.2).

Si queremos hallar el valor mínimo de la función $f(\mathbf{x})$ partiendo de una solución inicial, lo ideal sería utilizar estas direcciones de mayor crecimiento pero en sentido inverso, es decir, usar $-f'(\mathbf{x})$, que puede escribirse como:

$$-f'(\mathbf{x}) = \mathbf{B} - \mathbf{A} \mathbf{x}. \quad (3.87)$$

Pero como estamos iterando, tenemos en realidad que:

$$-f'(\mathbf{x}^{(i)}) = \mathbf{B} - \mathbf{A} \mathbf{x}^{(i)} = \mathbf{R}^{(i)}, \quad (3.88)$$

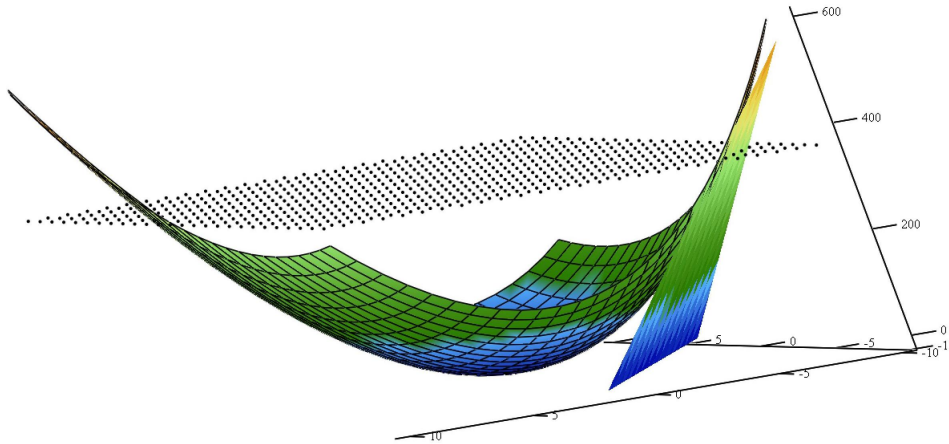


Figura 3.2: Forma cuadrática y plano tangente.

que resulta ser el *residuo*. En consecuencia, el residuo no es otra cosa que la dirección descendente más empinada para llegar al mínimo, o sea, la del *descenso más rápido*. Como partimos de un vector inicial, lo que nos interesa es obtener un coeficiente α que optimice cada paso utilizando la dirección más empinada y así obtener una aproximación $i + 1$ más cercana a la solución «exacta». Para ello partamos de la expresión general

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \alpha \mathbf{R}^{(i)}. \quad (3.89)$$

Para obtener el α , minimizaremos la forma cuadrática. Así tenemos que:

$$\frac{df(\mathbf{x}^{(i+1)})}{d\alpha} = f'(\mathbf{x}^{(i+1)})^T \frac{d\mathbf{x}^{(i+1)}}{d\alpha} = f'(\mathbf{x}^{(i+1)})^T \cdot \mathbf{R}^{(i)} = 0, \quad (3.90)$$

lo que equivale a decir que el residuo y el gradiente son *ortogonales*. Como además sabemos que $\mathbf{R}^{(i+1)} = -f'(\mathbf{x}^{(i+1)})$, entonces tenemos:

$$\begin{aligned} -\mathbf{R}^{(i+1)T} \cdot \mathbf{R}^{(i)} &= 0 \\ (\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i+1)})^T \cdot \mathbf{R}^{(i)} &= 0 \\ [\mathbf{B} - \mathbf{A} (\mathbf{x}^{(i)} + \alpha_i \cdot \mathbf{R}^{(i)})]^T \cdot \mathbf{R}^{(i)} &= 0 \\ (\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i)})^T \cdot \mathbf{R}^{(i)} - \alpha_i (\mathbf{A} \cdot \mathbf{R}^{(i)})^T \cdot \mathbf{R}^{(i)} &= 0 \\ (\mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(i)})^T \cdot \mathbf{R}^{(i)} &= \alpha_i (\mathbf{A} \cdot \mathbf{R}^{(i)})^T \cdot \mathbf{R}^{(i)} \\ \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)} &= \alpha_i \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)} \\ \alpha_i &= \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}}. \end{aligned} \quad (3.91)$$

Así, nuestro nuevo algoritmo es:

$$\begin{aligned}\mathbf{R}^{(0)} &= \mathbf{B} - \mathbf{A} \mathbf{x}^{(0)}, \\ \alpha_i &= \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}}, \\ \mathbf{x}^{(i+1)} &= \mathbf{x}^{(i)} + \alpha_i \mathbf{R}^{(i)}, \\ \mathbf{R}^{(i+1)} &= \mathbf{R}^{(i)} - \alpha_i \mathbf{A} \cdot \mathbf{R}^{(i)}.\end{aligned}\tag{3.92}$$

El criterio para interrumpir las iteraciones es el mismo que el aplicado para el método de los residuos mínimos.

Método de los Gradientes Conjugados

El método anterior es una mejora notable al método de los residuos mínimos. No sólo mejora la velocidad de convergencia sino que reduce la cantidad de operaciones. Sin embargo tiene una desventaja importante: suele usar varias veces la misma dirección de acercamiento. Esto significa que no utiliza bien las direcciones más empinadas. Veamos por qué.

Vimos que el vector residuo es el gradiente de nuestra forma cuadrática. Analicemos la figura 3.2 más en detalle. El gradiente lo hemos representado con un plano que pasa por un punto cuya «inclinación» nos da una idea del crecimiento (decrecimiento) en ese punto. Pero en realidad lo que tenemos son varias direcciones posibles que descienden rápidamente hacia el mínimo. El Método del Descenso Más Rápido sólo exige que los residuos sean ortogonales, pero no se ocupa de las direcciones con las cuales se aproxima al siguiente resultado, con lo cual puede repetir cualquier dirección en el proceso iterativo hasta obtener la solución. Así pierde eficiencia.

La forma más rápida de llegar sería usar direcciones que no se repitan durante el proceso de descenso. ¿Cuál sería el conjunto de direcciones que harían más rápido ese descenso? La respuesta es: tomemos un conjunto de direcciones $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(n-1)}$, tales que sean ortogonales entre sí, o sea que se cumpla que:

$$\begin{aligned}\mathbf{d}^{(0)} \cdot \mathbf{d}^{(1)} &= 0; \\ \mathbf{d}^{(1)} \cdot \mathbf{d}^{(2)} &= 0; \\ &\dots\dots \\ \mathbf{d}^{(i)} \cdot \mathbf{d}^{(j)} &= 0, \text{ para } i \neq j.\end{aligned}$$

Entonces nuestra expresión inicial será

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)},\tag{3.93}$$

en lugar de la utilizada en los métodos vistos previamente.

Como hemos definido que las direcciones que aproximan nuestra solución son ortogonales, entonces también el error $\mathbf{e}^{(i+1)}$ debería ser ortogonal, es decir, se debería cumplir que:

$$\begin{aligned}\mathbf{d}^{(i)T} \cdot \mathbf{e}^{(i+1)} &= 0 \\ \mathbf{d}^{(i)T} \cdot (\mathbf{e}^{(i)} + \alpha_i \mathbf{d}^{(i)}) &= 0 \\ \mathbf{d}^{(i)T} \cdot \mathbf{e}^{(i)} + \alpha_i \mathbf{d}^{(i)T} \cdot \mathbf{d}^{(i)} &= 0 \\ \mathbf{d}^{(i)T} \cdot \mathbf{e}^{(i)} &= -\alpha_i \mathbf{d}^{(i)T} \cdot \mathbf{d}^{(i)} \\ \alpha_i &= -\frac{\mathbf{d}^{(i)T} \cdot \mathbf{e}^{(i)}}{\mathbf{d}^{(i)T} \cdot \mathbf{d}^{(i)}}.\end{aligned}\tag{3.94}$$

Sin embargo, este algoritmo no es muy útil pues debemos conocer el error que estamos cometiendo para obtener el coeficiente α_i . Y si conocemos $\mathbf{e}^{(i+1)}$, conocemos la solución y no tendría sentido obtener el coeficiente α .

En lugar de proponer que el error sea ortogonal a la dirección, vamos a proponer que las direcciones sean *conjugadas*, también llamadas direcciones *ortogonales por A*. ¿Qué significa esto? Supongamos por un momento que trabajamos sobre una superficie esférica similar a un globo, y dibujamos sobre ésta dos líneas que sean ortogonales, como lo son, un meridiano y un paralelo. Si deformamos nuestro globo de manera que deje de ser esférico y se convierta en un elipsoide de revolución, las dos líneas se cortarían, pero *no serán ortogonales*. Si volvemos a transformar ese globo deformado en una esfera otra vez, dichas líneas volverán a ser ortogonales. Las direcciones en el elipsoide se denominan *conjugadas*.

La idea del método es partir de la situación del elipsoide, transformar los vectores de forma de llevarlos a la esfera, obtener allí las direcciones ortogonales y luego trabajar nuevamente en el elipsoide. De esa forma, las direcciones serán ortogonales en la superficie de la esfera, y conjugadas en el elipsoide. (Otro ejemplo en ese mismo sentido sería proyectar la esfera sobre un plano, práctica común de la *cartografía*.)

Para obtener nuestro nuevo algoritmo, vamos a proponer que la dirección $\mathbf{d}^{(i)}$ sea ortogonal a $\mathbf{R}^{(i+1)}$, algo que surge de minimizar el gradiente pues

$$\frac{d f(\mathbf{x}^{(i+1)})}{d\alpha} = f'(\mathbf{x}^{(i+1)})^T \frac{d\mathbf{x}^{(i+1)}}{d\alpha} = \underbrace{f'(\mathbf{x}^{(i+1)})^T}_{-\mathbf{R}^{(i+1)T}} \cdot \mathbf{d}^{(i)} = 0, \quad (3.95)$$

y podemos plantear lo siguiente:

$$\begin{aligned} -\mathbf{d}^{(i)T} \cdot \mathbf{R}^{(i+1)} &= 0 \\ \mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i+1)} &= 0 \\ \mathbf{d}^{(i)T} \cdot \mathbf{A} \left(\mathbf{e}^{(i)} + \alpha_i \mathbf{d}^{(i)} \right) &= 0 \\ \mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i)} + \alpha_i \mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(i)} &= 0 \\ \mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i)} &= -\alpha_i \mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(i)} \\ \alpha_i &= -\frac{\mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i)}}{\mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(i)}} \\ \alpha_i &= \frac{\mathbf{d}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(i)}}. \end{aligned} \quad (3.96)$$

Con este coeficiente α_i nos aseguramos que nuestro método va aproximando la solución mediante direcciones conjugadas. Pero nos faltan hallar estas direcciones. ¿Cómo las obtenemos? La forma más sencilla es aplicar el método de Gram-Schmidt para ortogonalizar vectores. En este caso lo que haremos es obtener vectores conjugados a partir de un vector inicial, por lo que la fórmula de Gram-Schmidt queda de la siguiente forma:

$$\mathbf{d}^{(i)} = \mathbf{u}^{(i)} + \sum_{j=0}^{i-1} \beta_{ij} \mathbf{A} \cdot \mathbf{d}^{(j)}, \quad (3.97)$$

y el coeficiente β_{ij} lo obtenemos mediante:

$$\beta_{ij} = -\frac{\mathbf{u}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}}{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}}, \quad (3.98)$$

siendo $\mathbf{u}^{(i)}$ el vector a partir del cual obtenemos las direcciones conjugadas (ortogonales por \mathbf{A}). (Véase [17].)

Nos falta definir el vector $\mathbf{u}^{(i)}$. Si proponemos al vector $\mathbf{R}^{(i)}$ tendremos que:

$$\mathbf{d}^{(i)T} \cdot \mathbf{R}^{(i)} = \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}, \quad (3.99)$$

y entonces, obtendremos lo siguiente:

$$\beta_{ij} = -\frac{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}}{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}}. \quad (3.100)$$

Ahora vamos a obtener el β_{ij} para poder encontrar nuestras direcciones conjugadas. Así, tenemos que:

$$\begin{aligned} \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(j+1)} &= \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(j)} - \alpha_j \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(j)} \\ \alpha_j \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(j)} &= \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(j)} - \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(j+1)} \\ \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(j)} &= \begin{cases} \frac{1}{\alpha_j} \mathbf{R}^{(j)T} \cdot \mathbf{R}^{(j)} & \text{si } i = j \\ -\frac{1}{\alpha_j} \mathbf{R}^{(j+1)T} \cdot \mathbf{R}^{(j+1)} & \text{si } i = j + 1 \end{cases} \\ \beta_{j+1j} &= \frac{1}{\alpha_j} \frac{\mathbf{R}^{(j+1)T} \cdot \mathbf{R}^{(j+1)}}{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}} \end{aligned} \quad (3.101)$$

Antes hemos obtenido que:

$$\alpha_j = \frac{\mathbf{d}^{(j)T} \cdot \mathbf{R}^{(j)}}{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}} \Rightarrow \frac{1}{\alpha_j} = \frac{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}}{\mathbf{d}^{(j)T} \cdot \mathbf{R}^{(j)}}, \quad (3.102)$$

por lo tanto, finalmente tendremos que:

$$\beta_{j+1j} = \frac{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}}{\mathbf{d}^{(j)T} \cdot \mathbf{R}^{(j)}} \frac{\mathbf{R}^{(j+1)T} \cdot \mathbf{R}^{(j+1)}}{\mathbf{d}^{(j)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(j)}} = \frac{\mathbf{R}^{(j+1)T} \cdot \mathbf{R}^{(j+1)}}{\mathbf{d}^{(j)T} \cdot \mathbf{R}^{(j)}} = \frac{\mathbf{R}^{(j+1)T} \cdot \mathbf{R}^{(j+1)}}{\mathbf{R}^{(j)T} \cdot \mathbf{R}^{(j)}}, \quad (3.103)$$

pues hemos visto que $\mathbf{d}^{(j)T} \cdot \mathbf{R}^{(j)} = \mathbf{R}^{(j)T} \cdot \mathbf{R}^{(j)}$. Simplificando la notación tenemos:

$$\beta_{j+1j} = \frac{\mathbf{R}^{(j+1)T} \cdot \mathbf{R}^{(j+1)}}{\mathbf{R}^{(j)T} \cdot \mathbf{R}^{(j)}}. \quad (3.104)$$

Con este último coeficiente, y unificando todo en el índice i , tenemos el algoritmo para el *Método de los Gradientes Conjugados*:

$$\begin{aligned} \mathbf{d}^{(0)} &= \mathbf{R}^{(0)} = \mathbf{B} - \mathbf{A} \cdot \mathbf{x}^{(0)} \\ \alpha_i &= \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{d}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{d}^{(i)}} \\ \mathbf{x}^{(i+1)} &= \mathbf{x}^{(i)} + \alpha_i \mathbf{d}^{(i)} \\ \mathbf{R}^{(i+1)} &= \mathbf{R}^{(i)} - \alpha_i \mathbf{A} \cdot \mathbf{d}^{(i)} \\ \beta_{i+1i} &= \frac{\mathbf{R}^{(i+1)T} \cdot \mathbf{R}^{(i+1)}}{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}} \\ \mathbf{d}^{(i+1)} &= \mathbf{R}^{(i+1)} + \beta_{i+1i} \mathbf{d}^{(i)}. \end{aligned} \quad (3.105)$$

Este método es muy poderoso y converge muy rápidamente, salvo por la aparición de los errores de redondeo en las operaciones, lo que hace que no siempre el producto interno sea nulo. Más adelante se verán algunas características sobre la convergencia que lo convierten en uno de los métodos iterativos para sistemas de ecuaciones lineales raras con matrices simétricas definidas positivas.

3.10.4. Convergencia de los métodos no estacionarios

Analizaremos brevemente la convergencia de los métodos no estacionarios. En primer lugar nos ocuparemos rápidamente del método de los residuos mínimos, y luego de los otros dos métodos.

Método de los residuos mínimos

Ya habíamos dicho que para garantizar la convergencia de este método, la matriz \mathbf{A} debe ser definida positiva. El siguiente teorema demuestra esta afirmación.

Teorema 3.9. Sea \mathbf{A} una matriz definida positiva y sea

$$\mu = \lambda_{\min} \left(\frac{\mathbf{A} + \mathbf{A}^T}{2} \right); \quad \sigma = \|\mathbf{A}\|_2,$$

entonces el vector $\mathbf{R}^{(i+1)}$ generado por el método de los residuos mínimos satisface la relación

$$\|\mathbf{R}^{(i+1)}\|_2 \leq \left(1 - \frac{\mu^2}{\sigma^2} \right)^{1/2} \|\mathbf{R}^{(i)}\|_2,$$

y el algoritmo correspondiente converge para cualquier valor inicial de $\mathbf{x}^{(0)}$.

La demostración de este teorema puede verse en [15].

Método del descenso más rápido

Para el análisis de la convergencia de este método (y el de los gradientes conjugados) nos basaremos en el estudio de los autovalores y autovectores de la matriz \mathbf{A} .

Supongamos que el vector $\mathbf{e}^{(i)}$ sea un autovector asociado a un autovalor λ_e . Entonces el residuo se puede escribir como:

$$\mathbf{R}^{(i)} = -\mathbf{A} \mathbf{e}^{(i)} = -\lambda_e \mathbf{e}^{(i)}, \quad (3.106)$$

por lo tanto, es también un autovector.

De la misma forma podemos obtener $\mathbf{e}^{(i+1)}$, pues es:

$$\mathbf{e}^{(i+1)} = \mathbf{e}^{(i)} + \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \mathbf{R}^{(i)} = \mathbf{e}^{(i)} + \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\lambda_e \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}} \left(-\lambda_e \mathbf{e}^{(i)} \right) = 0. \quad (3.107)$$

Si uno elige $\alpha_i = \lambda_e$, ¡basta con una iteración para obtener el resultado «exacto»! Pero en realidad, debemos expresar $\mathbf{e}^{(i)}$ como una combinación lineal de autovectores, es decir,

$$\mathbf{e}^{(i)} = \sum_{j=1}^n \xi_j \mathbf{v}^{(j)}, \quad (3.108)$$

donde los $\mathbf{v}^{(j)}$ son vectores ortonormales (elegidos así por conveniencia), y los ξ_j son las longitudes de cada vector. Entonces nos queda

$$\begin{aligned}
 \mathbf{R}^{(i)} &= -\mathbf{A} \cdot \mathbf{e}^{(i)} = -\sum_{j=1}^n \xi_j \lambda_j \mathbf{v}^{(j)} \\
 \|\mathbf{e}^{(i)}\|^2 &= \mathbf{e}^{(i)T} \cdot \mathbf{e}^{(i)} = \sum_j \xi_j^2 \\
 \mathbf{e}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i)} &= \left[\sum_j \xi_j \mathbf{v}^{(j)T} \right] \left[\sum_j \xi_j \lambda_j \mathbf{v}^{(j)} \right] = \sum_j \xi_j^2 \lambda_j \\
 \|\mathbf{R}^{(i)}\|^2 &= \mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)} = \sum_j \xi_j^2 \lambda_j^2 \\
 \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)} &= \sum_j \xi_j^2 \lambda_j^3
 \end{aligned} \tag{3.109}$$

Esta última expresión la obtenemos al tener en cuenta que el $\mathbf{R}^{(i)}$ también se puede expresar como la combinación lineal de autovectores, y que su longitud es $-\xi_j \lambda_j$. Si volvemos a la expresión del vector $\mathbf{e}^{(i+1)}$ tenemos:

$$\mathbf{e}^{(i+1)} = \mathbf{e}^{(i)} + \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \mathbf{R}^{(i)} = \mathbf{e}^{(i)} + \frac{\sum_j \xi_j^2 \lambda_j^2}{\sum_j \xi_j^2 \lambda_j^3} \mathbf{R}^{(i)}, \tag{3.110}$$

que nos muestra que α_i es un promedio ponderado de $\frac{1}{\lambda_j}$.

Para analizar la convergencia en forma más general vamos a definir primero la *norma energética* $\|\mathbf{e}\|_A = (\mathbf{e}^T \cdot \mathbf{A} \cdot \mathbf{e})^{1/2}$. Con esta norma tenemos:

$$\begin{aligned}
 \|\mathbf{e}^{(i+1)}\|_A^2 &= \mathbf{e}^{(i+1)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i+1)} \\
 &= \left(\mathbf{e}^{(i)T} + \alpha_i \mathbf{R}^{(i)T} \right) \mathbf{A} \left(\mathbf{e}^{(i)} + \alpha_i \mathbf{R}^{(i)} \right) \\
 &= \mathbf{e}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i)} + 2\alpha_i \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{e}^{(i)} + \alpha_i^2 \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)} \\
 &= \|\mathbf{e}^{(i)}\|_A^2 + 2 \frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \left[-\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)} \right] + \\
 &\quad + \left[\frac{\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)}}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \right]^2 \mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)} \\
 &= \|\mathbf{e}^{(i)}\|_A^2 - \frac{\left[\mathbf{R}^{(i)T} \cdot \mathbf{R}^{(i)} \right]^2}{\mathbf{R}^{(i)T} \cdot \mathbf{A} \cdot \mathbf{R}^{(i)}} \\
 &= \|\mathbf{e}^{(i)}\|_A^2 \left[1 - \frac{\left(\sum_j \xi_j^2 \lambda_j^2 \right)^2}{\sum_j \xi_j^2 \lambda_j^3 \sum_j \xi_j^2 \lambda_j} \right] \Rightarrow \\
 \|\mathbf{e}^{(i+1)}\|_A^2 &= \|\mathbf{e}^{(i)}\|_A^2 \omega^2 \quad \text{con } \omega^2 = 1 - \frac{\left(\sum_j \xi_j^2 \lambda_j^2 \right)^2}{\sum_j \xi_j^2 \lambda_j^3 \sum_j \xi_j^2 \lambda_j}
 \end{aligned} \tag{3.111}$$

Esto quiere decir que el error de la iteración $i + 1$ es función de los autovalores de \mathbf{A} . Como lo que interesa es un límite superior del error, y no el error en si mismo, si definimos que

$$\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}, \quad (3.112)$$

se puede demostrar que

$$\omega = \frac{\kappa - 1}{\kappa + 1}, \quad (3.113)$$

con lo cual tenemos que

$$\|e^{(i)}\|_A \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^i \|e^{(0)}\|_A. \quad (3.114)$$

(La demostración indicada se puede ver en [17].)

Método de los gradientes conjugados

Para el método de los gradientes conjugados vale el mismo desarrollo hecho para el descenso más rápido, pero con una leve modificación se llega a que

$$\|e^{(i)}\|_A \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^i \|e^{(0)}\|_A. \quad (3.115)$$

Podemos decir que el método converge más rápido que el método del descenso más rápido, pues en el primero la convergencia depende de $\sqrt{\kappa}$, mientras que en el segundo depende de κ . Puesto que κ es equivalente a la condición de \mathbf{A} , finalmente tenemos que una matriz bien condicionada converge rápidamente a la solución, en tanto que no lo hace si está mal condicionada. Por esta razón, este método rara vez se aplica directamente sobre el sistema $\mathbf{A} \mathbf{x} = \mathbf{B}$, sino que se *precondiciona* a la matriz \mathbf{A} con una matriz \mathbf{N} , formando el sistema $\mathbf{N}^T \mathbf{A} \mathbf{x} = \mathbf{N}^T \mathbf{B}$ de manera tal que $\mathbf{N}^T \mathbf{A}$ suele tener un número de condición mucho menor que \mathbf{A} .

Por otra parte, si la matriz está bien condicionada, el método de los gradientes conjugados converge luego de n iteraciones. Es más, si no hubieran problemas derivados de la representación numérica en las computadoras, el método convergería después de k iteraciones, siendo k el número de autovalores *no repetidos* de \mathbf{A} . (Ver en [15] y [17].)

3.10.5. Aspectos computacionales

En general, obtener una solución eficiente de un sistema de ecuaciones lineales por medio de métodos iterativos depende fuertemente de la elección del método. Si bien podemos esperar una menor eficiencia de estos métodos respecto de los métodos directos, los métodos iterativos suelen ser más fáciles de implementar y, como no hay que factorizar la matriz, permiten resolver sistemas mucho más grandes que los directos.

Como resumen de los métodos vistos, tenemos lo siguiente:

1. **Método de Jacobi:** Muy fácil de usar, pero sólo converge si la matriz es estrictamente diagonal dominante. Actualmente sólo se lo considera como una forma de introducción a los métodos iterativos.
2. **Método de Gauss-Seidel:** Converge más rápido que el de Jacobi, pero no puede competir con los métodos no estacionarios. Tiene la ventaja de que también converge si la matriz del sistema es simétrica y definida positiva.
3. **Método de las sobrerrelajaciones sucesivas:** Converge más rápido que Gauss-Seidel si $\omega > 1$, y suele converger con $\omega < 1$ cuando Gauss-Seidel no converge. Como vimos, la velocidad de convergencia depende de ω , valor que no es fácil de obtener en forma analítica. Obtener ese valor puede llevar a perder parte de esa ventaja.

4. **Método de los residuos mínimos:** Converge si la matriz A del sistema es definida positiva y mejora si además es simétrica. Es más fácil de programar pues hay que hacer operaciones matriciales (vectoriales). La convergencia puede ser lenta, similar a Jacobi.
5. **Método del descenso más rápido:** Se aplica a sistemas con matrices simétricas definidas positivas. Converge más rápido que el anterior pero si la matriz no está bien condicionada, no converge. Es más fácil de programar que el anterior porque reduce la cantidad de operaciones matriciales. Es equivalente a Gauss-Seidel.
6. **Método de los gradientes conjugados:** Se aplica a matrices simétricas definidas positivas. Cuando la matriz está bien condicionada y además tiene p autovalores repetidos y bien distribuidos, converge para $k = n - p$ iteraciones (convergencia supralineal). Por este motivo, suele usarse preconditionado para conseguir convergencias supralineales. Es más fácil de implementar que los anteriores métodos no estacionarios, pero suele tener problemas con el error de redondeo.

3.11. Errores de los métodos iterativos

En este punto analizaremos fundamentalmente los errores de los métodos iterativos estacionarios, pues son conceptualmente más fáciles de entender. Empezaremos por el error de truncamiento.

Supongamos que \mathbf{x} sea la solución de nuestro sistema de ecuaciones y $\mathbf{x}^{(k+1)}$ el resultado luego de $k + 1$ iteraciones. Entonces podemos definir el error como

$$\mathbf{x}^{(k+1)} - \mathbf{x} = \mathbf{T} \left(\mathbf{x}^{(k)} - \mathbf{x} \right). \quad (3.116)$$

Si sumamos y restamos $\mathbf{T} \mathbf{x}^{(k+1)}$ tenemos

$$\begin{aligned} \mathbf{x}^{(k+1)} - \mathbf{x} &= \mathbf{T} \left(\mathbf{x}^{(k)} - \mathbf{x} \right) + \mathbf{T} \mathbf{x}^{(k+1)} - \mathbf{T} \mathbf{x}^{(k+1)} \\ &= \mathbf{T} \left(\mathbf{x}^{(k)} - \mathbf{x}^{(k+1)} \right) + \mathbf{T} \left(\mathbf{x}^{(k+1)} - \mathbf{x} \right). \end{aligned} \quad (3.117)$$

Si tomamos las normas tenemos que

$$\begin{aligned} \left\| \mathbf{x}^{(k+1)} - \mathbf{x} \right\| &\leq \|\mathbf{T}\| \left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\| + \|\mathbf{T}\| \left\| \mathbf{x}^{(k+1)} - \mathbf{x} \right\| \\ (1 - \|\mathbf{T}\|) \left\| \mathbf{x}^{(k+1)} - \mathbf{x} \right\| &\leq \|\mathbf{T}\| \left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\| \\ \left\| \mathbf{x}^{(k+1)} - \mathbf{x} \right\| &\leq \frac{\|\mathbf{T}\|}{(1 - \|\mathbf{T}\|)} \left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\|. \end{aligned} \quad (3.118)$$

Por lo tanto, el error de truncamiento está dado por

$$E_T \cong \frac{\|\mathbf{T}\|}{(1 - \|\mathbf{T}\|)} \left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\|. \quad (3.119)$$

Para el caso del error inherente partimos de

$$\mathbf{x}^{(k+1)} = \mathbf{T} \mathbf{x}^{(k)} + \mathbf{C}. \quad (3.120)$$

Si consideramos los errores inherentes del sistema, el resultado que obtendremos será en realidad $\bar{\mathbf{x}}^{(k+1)}$. Supongamos que desechamos todos los errores de los pasos anteriores, es decir, que $\mathbf{x}^{(k)} \equiv \bar{\mathbf{x}}^{(k)}$, entonces tenemos que

$$\bar{\mathbf{x}}^{(k+1)} = \mathbf{x}^{(k)} - \delta \mathbf{x}^{(k)} = (\mathbf{T} - \delta \mathbf{T}) \mathbf{x}^{(k)} + (\mathbf{C} - \delta \mathbf{C}). \quad (3.121)$$

Como $\mathbf{x} = \mathbf{T} \mathbf{x} + \mathbf{C}$, podemos hacer lo siguiente:

$$\begin{aligned}
 \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} &= (\mathbf{T} - \delta\mathbf{T})\mathbf{x}^{(k)} + (\mathbf{C} - \delta\mathbf{C}) - \mathbf{T} \mathbf{x} - \mathbf{C} \\
 &= \mathbf{T} (\mathbf{x}^{(k)} - \mathbf{x}) - \delta\mathbf{T}\mathbf{x}^{(k)} - \delta\mathbf{C} \\
 &= \mathbf{T} (\mathbf{x}^{(k)} - \mathbf{x}) - \delta\mathbf{T}\mathbf{x}^{(k)} - \delta\mathbf{C} + \mathbf{T} \mathbf{x}^{(k+1)} - \mathbf{T} \mathbf{x}^{(k+1)} \\
 &= \mathbf{T} (\mathbf{x}^{(k+1)} - \mathbf{x}) + \mathbf{T} (\mathbf{x}^{(k)} - \mathbf{x}^{(k+1)}) - \delta\mathbf{T}\mathbf{x}^{(k)} - \delta\mathbf{C}.
 \end{aligned} \tag{3.122}$$

Si nuevamente tomamos las normas, obtenemos

$$\begin{aligned}
 \|\mathbf{x}^{(k+1)} - \mathbf{x}\| &\leq \|\mathbf{T}\| \|\mathbf{x}^{(k+1)} - \mathbf{x}\| + \|\mathbf{T}\| \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| + \|\delta\mathbf{T}\| \|\mathbf{x}^{(k)}\| + \\
 &\quad + \|\delta\mathbf{C}\| \\
 (1 - \|\mathbf{T}\|) \|\mathbf{x}^{(k+1)} - \mathbf{x}\| &\leq \|\mathbf{T}\| \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| + \|\delta\mathbf{T}\| \|\mathbf{x}^{(k)}\| + \|\delta\mathbf{C}\| \\
 \|\mathbf{x}^{(k+1)} - \mathbf{x}\| &\leq \frac{\|\mathbf{T}\|}{1 - \|\mathbf{T}\|} \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| + \frac{\|\delta\mathbf{T}\|}{1 - \|\mathbf{T}\|} \|\mathbf{x}^{(k)}\| + \frac{\|\delta\mathbf{C}\|}{1 - \|\mathbf{T}\|}.
 \end{aligned} \tag{3.123}$$

Si analizamos en detalle esta última expresión, vemos que se repite el error de truncamiento (primer término de la derecha). En consecuencia, el error inherente está dado por

$$E_I \cong \frac{\|\delta\mathbf{T}\|}{1 - \|\mathbf{T}\|} \|\mathbf{x}^{(k)}\| + \frac{\|\delta\mathbf{C}\|}{1 - \|\mathbf{T}\|}. \tag{3.124}$$

Finalmente, analicemos el error de redondeo. Una vez más, partamos de la expresión

$$\mathbf{x}^{(k+1)} = \mathbf{T} \mathbf{x}^{(k)} + \mathbf{C},$$

y nuevamente supongamos que lo que obtenemos es en realidad es $\bar{\mathbf{x}}^{(k+1)}$ y que $\mathbf{x}^{(k)} \equiv \bar{\mathbf{x}}^{(k)}$. Entonces nos queda:

$$\bar{\mathbf{x}}^{(k+1)} = \mathbf{T} \bar{\mathbf{x}}^{(k)} + \mathbf{C}.$$

Para cada componente de $\mathbf{x}^{(k+1)}$ tenemos

$$x_i^{(k+1)} - \delta x_i^{(k+1)} = \left[\text{fl} \left(\sum_{j=1}^n t_{ij} x_j^{(k)} \right) + c_i \right] (1 - \delta_i). \tag{3.125}$$

Si hacemos un análisis retrospectivo del error («backward error»), y asumimos que $n\mu \leq 0,01$, nos queda que

$$\delta x_i^{(k+1)} = \left[\sum_j t_{ij} x_j^{(k)} 1,01(n+2-j)\mu\theta_j \right] (1 + \delta_i) + x_i^{(k+1)} \tag{3.126}$$

con $|\theta_j| \leq 1$ y $|\delta_i| \leq \mu$.

Consideremos ahora el hecho de que generalmente las matrices de los sistemas son ralas. Entonces podemos definir que

$$p = \max_{1 \leq i \leq n} \{p_i\}, \text{ con } p_i : \text{ cantidad de elementos no nulos en una fila.} \tag{3.127}$$

$$q = \max_{1 \leq i, j \leq n} \{|t_{ij}|\}, \tag{3.128}$$

entonces, si tomamos normas nos queda

$$\left\| \delta x_i^{(k+1)} \right\| \leq q \left\| \mathbf{x}^{(k)} \right\| 1,01 \left[\sum_{j=1}^p (p+2-j) \right] \mu + \left\| x_i^{(k+1)} \right\| \mu, \quad (3.129)$$

y como $\mathbf{x}^{(k)} \approx \mathbf{x}^{(k+1)}$, podemos escribir que

$$\frac{\left\| \delta \mathbf{x}^{(k+1)} \right\|}{\left\| \mathbf{x}^{(k)} \right\|} \leq \left(q 1,01 \frac{p^2 + 3p}{2} + 1 \right) \mu. \quad (3.130)$$

Ahora estimemos la diferencia $\bar{\mathbf{x}}^{(k+1)} - \mathbf{x}$. Sabemos que

$$\bar{\mathbf{x}}^{(k+1)} = \mathbf{T} \mathbf{x}^{(k)} + \mathbf{C} - \delta \mathbf{x}^{(k+1)},$$

entonces

$$\begin{aligned} \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} &= \mathbf{T} \mathbf{x}^{(k)} + \mathbf{C} - \delta \mathbf{x}^{(k+1)} - \mathbf{T} \mathbf{x} - \mathbf{C} \\ &= \mathbf{T} \left(\mathbf{x}^{(k)} - \mathbf{x} \right) - \delta \mathbf{x}^{(k+1)}. \end{aligned} \quad (3.131)$$

Si nuevamente sumamos y restamos $\mathbf{T} \bar{\mathbf{x}}^{(k+1)}$, obtenemos

$$\bar{\mathbf{x}}^{(k+1)} - \mathbf{x} = \mathbf{T} \left(\mathbf{x}^{(k)} - \bar{\mathbf{x}}^{(k+1)} \right) + \mathbf{T} \left(\bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right) - \delta \mathbf{x}^{(k+1)}. \quad (3.132)$$

Una vez más, tomemos las normas, con lo cual nos queda

$$\begin{aligned} \left\| \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right\| &= \left\| \mathbf{T} \right\| \left\| \mathbf{x}^{(k+1)} - \bar{\mathbf{x}}^{(k)} \right\| + \left\| \mathbf{T} \right\| \left\| \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right\| + \left\| \delta \mathbf{x}^{(k+1)} \right\| \\ (1 - \left\| \mathbf{T} \right\|) \left\| \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right\| &= \left\| \mathbf{T} \right\| \left\| \mathbf{x}^{(k+1)} - \bar{\mathbf{x}}^{(k)} \right\| + \left\| \delta \mathbf{x}^{(k+1)} \right\| + \\ \left\| \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right\| &= \frac{\left\| \mathbf{T} \right\|}{1 - \left\| \mathbf{T} \right\|} \left\| \mathbf{x}^{(k+1)} - \bar{\mathbf{x}}^{(k)} \right\| - \frac{\left\| \delta \mathbf{x}^{(k+1)} \right\|}{1 - \left\| \mathbf{T} \right\|} \end{aligned} \quad (3.133)$$

Puesto que $\left\| \delta x_i^{(k+1)} \right\| \leq \left(q 1,01 \frac{p^2 + 3p}{2} + 1 \right) \left\| \mathbf{x}^{(k)} \right\| \mu$ y como el primer término corresponde al error de truncamiento, nos queda que

$$E_R \leq \frac{\left\| \mathbf{x}^{(k)} \right\|}{1 - \left\| \mathbf{T} \right\|} \left(q 1,01 \frac{p^2 + 3p}{2} + 1 \right) \mu. \quad (3.134)$$

Finalmente, el error total al aplicar un método iterativo estacionario es la suma de todos los errores, es decir,

$$\begin{aligned} \left\| \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right\| &\leq E_T + E_I + E_R \\ &\leq \frac{\left\| \mathbf{T} \right\|}{(1 - \left\| \mathbf{T} \right\|)} \left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\| + \frac{\left\| \delta \mathbf{T} \right\|}{1 - \left\| \mathbf{T} \right\|} \left\| \mathbf{x}^{(k)} \right\| + \frac{\left\| \delta \mathbf{C} \right\|}{1 - \left\| \mathbf{T} \right\|} \\ &\quad + \frac{\left\| \mathbf{x}^{(k)} \right\|}{1 - \left\| \mathbf{T} \right\|} \left(q 1,01 \frac{p^2 + 3p}{2} + 1 \right) \mu \\ \left\| \bar{\mathbf{x}}^{(k+1)} - \mathbf{x} \right\| &\leq \frac{1}{1 - \left\| \mathbf{T} \right\|} \left[\left\| \mathbf{T} \right\| \left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\| + \left\| \delta \mathbf{T} \right\| \left\| \mathbf{x}^{(k)} \right\| + \left\| \delta \mathbf{C} \right\| \right. \\ &\quad \left. + \left\| \mathbf{x}^{(k)} \right\| \left(q 1,01 \frac{p^2 + 3p}{2} + 1 \right) \mu \right]. \end{aligned} \quad (3.135)$$

Como hemos visto en Errores, siempre es conveniente que los errores de truncamiento e inherentes predominen respecto al de redondeo. En consecuencia, siempre debemos tratar que $E_R < E_T < E_I$, es decir, que el error de redondeo sea el de menor incidencia, y si es posible, despreciable ⁶.

3.12. Sistemas de Ecuaciones No Lineales

En el capítulo anterior hemos analizado como resolver ecuaciones no lineales de una sola variable aplicando varios métodos. Y en la primera parte de este capítulo, hemos resuelto sistemas de ecuaciones lineales con varios métodos también, todos ellos basados en la facilidad de trabajar con matrices.

Pero no siempre nuestros problemas no lineales son de una sola variable. Consideremos el siguiente ejemplo:

$$f_1(x_1, x_2) = x_1^2 + x_2^2 - 1 = 0, \quad (3.136)$$

$$f_2(x_1, x_2) = x_1^2 - 2x_1 - x_2 + 1 = 0. \quad (3.137)$$

Este sistema sencillo no tiene solución única, consecuencia de las ecuaciones de segundo grado. Sin embargo, conocemos métodos que podemos aplicar para obtener una solución. Por ejemplo, podemos partir de los métodos iterativos vistos en para resolver sistemas de ecuaciones lineales malos y reordenar el sistema así:

$$x_1 = \pm \sqrt{1 - x_2^2}, \quad (3.138)$$

$$x_2 = x_1^2 - 2x_1 + 1. \quad (3.139)$$

Pero este algoritmo también puede ser considerado como la aplicación del *Método de las Aproximaciones Sucesivas*, que vimos para resolver ecuaciones no lineales. Así, nuestro sistema quedaría como un esquema iterativo que escribiremos de esta manera:

$$x_1^{(k+1)} = \pm \sqrt{1 - (x_2^{(k)})^2}, \quad (3.140)$$

$$x_2^{(k+1)} = (x_1^{(k)})^2 - 2x_1^{(k)} + 1, \quad (3.141)$$

donde

$$g_1(x_1^{(k)}, x_2^{(k)}) = \pm \sqrt{1 - (x_2^{(k)})^2}, \quad (3.142)$$

$$g_2(x_1^{(k)}, x_2^{(k)}) = (x_1^{(k)})^2 - 2x_1^{(k)} + 1. \quad (3.143)$$

Podemos intuir que la resolución de sistemas de ecuaciones no lineales es un aplicación conjunta de ambos: ecuaciones no lineales y sistemas de ecuaciones lineales.

Por otro lado, observemos que el esquema anterior es equivalente al *Método de Jacobi*. Entonces, una forma de mejorar la convergencia para llegar a la solución es aplicar el *Método Gauss-Seidel*:

$$x_1^{(k+1)} = \pm \sqrt{1 - (x_2^{(k)})^2}, \quad (3.144)$$

$$x_2^{(k+1)} = (x_1^{(k+1)})^2 - 2x_1^{(k+1)} + 1, \quad (3.145)$$

⁶González, en su libro, dice que $E_T < E_R$ pero eso se contrapone con lo que afirman otros autores. La razón principal es que el error de redondeo tiene un comportamiento «errático», lo que hace difícil acotarlo. (Ver ejemplo en el capítulo 1 con el error de discretización.)

que aprovecha la solución de $x_1^{(k+1)}$ obtenida en la primera ecuación. Notemos, sin embargo, que para x_1 existen dos soluciones posibles, por lo tanto, también las hay para x_2 . Además, como ahora no podemos construir una matriz \mathbf{A} como en el caso de los sistemas de ecuaciones lineales, tampoco podemos hacer un análisis de la condición de dicha matriz, por lo que se nos dificulta averiguar si el sistema es convergente o no. Solamente podemos estudiar a las funciones $g_1(x_1, x_2)$ y $g_2(x_1, x_2)$ según lo ya visto para ecuaciones no lineales, es decir, adaptar los teoremas vistos para el caso del *Método de las Aproximaciones Sucesivas* para una variable.

Si aprovechamos esto, inmediatamente podemos hacer una analogía y pensar, ¿por qué no aplicar el *Método de Newton-Raphson*? En realidad, habría que pensar en una adaptación. Para ello, apliquemos un desarrollo en serie de Taylor para funciones de dos variables, que queda de esta forma:

$$f_1(\bar{x}_1, \bar{x}_2) = f_1(x_1^{(k)}, x_2^{(k)}) + \frac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_1} (\bar{x}_1 - x_1^{(k)}) + \quad (3.146)$$

$$+ \frac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_2} (\bar{x}_2 - x_2^{(k)}) + \dots, \quad (3.147)$$

$$f_2(\bar{x}_1, \bar{x}_2) = f_2(x_1^{(k)}, x_2^{(k)}) + \frac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_1} (\bar{x}_1 - x_1^{(k)}) + \quad (3.148)$$

$$+ \frac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_2} (\bar{x}_2 - x_2^{(k)}) + \dots, \quad (3.149)$$

donde se cumple que:

$$f_1(\bar{x}_1, \bar{x}_2) = 0, \quad (3.150)$$

$$f_2(\bar{x}_1, \bar{x}_2) = 0. \quad (3.151)$$

Si truncamos el desarrollo en los términos de la primeras derivadas y reordenamos la expresión en función de lo anterior, tenemos que:

$$\frac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_1} (\bar{x}_1 - x_1^{(k)}) + \frac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_2} (\bar{x}_2 - x_2^{(k)}) = -f_1(x_1^{(k)}, x_2^{(k)}), \quad (3.152)$$

$$\frac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_1} (\bar{x}_1 - x_1^{(k)}) + \frac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_2} (\bar{x}_2 - x_2^{(k)}) = -f_2(x_1^{(k)}, x_2^{(k)}), \quad (3.153)$$

que ahora sí podemos expresar en forma matricial:

$$\begin{bmatrix} \frac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_1} & \frac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \\ \frac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_1} & \frac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \end{bmatrix} \begin{bmatrix} \bar{x}_1 - x_1^{(k)} \\ \bar{x}_2 - x_2^{(k)} \end{bmatrix} = - \begin{bmatrix} f_1(x_1^{(k)}, x_2^{(k)}) \\ f_2(x_1^{(k)}, x_2^{(k)}) \end{bmatrix}. \quad (3.154)$$

Si recordamos un poco de análisis matemático, la primera matriz no es otra cosa que el *Jacobiano* de las funciones $f_1(x_1, x_2)$ y $f_2(x_1, x_2)$, lo que nos permite escribir el sistema de una forma más sencilla:

$$\mathbf{J}(\mathbf{x}^{(k)}) \cdot (\bar{\mathbf{x}} - \mathbf{x}^{(k)}) = -\mathbf{F}(\mathbf{x}^{(k)}). \quad (3.155)$$

Con esta forma de expresarla es fácil obtener nuestra solución aproximada, pues

$$\bar{\mathbf{x}} - \mathbf{x}^{(k)} = -\mathbf{J}(\mathbf{x}^{(k)})^{-1} \cdot \mathbf{F}(\mathbf{x}^{(k)}), \quad (3.156)$$

$$\bar{\mathbf{x}} = \mathbf{x}^{(k)} - \mathbf{J}(\mathbf{x}^{(k)})^{-1} \cdot \mathbf{F}(\mathbf{x}^{(k)}), \quad (3.157)$$

que expresado en términos iterativos queda de esta manera:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{J}(\mathbf{x}^{(k)})^{-1} \cdot \mathbf{F}(\mathbf{x}^{(k)}), \quad (3.158)$$

y se conoce, como no podía ser de otra manera, como *Método de Newton*.

Hemos encontrado una forma bastante sencilla de obtener nuestra solución. Pero de todos modos, analicemos un poco el método. En primer lugar, podemos ver que para cada caso tenemos que calcular el *Jacobiano* de las funciones en cada iteración. Eso ya nos dificulta un poco el procedimiento. No es muy distinto al caso del *Método de Newton-Raphson* para una sola variable, pero en el caso matricial es agregar más pasos de cálculo, pues hay que ingresar cada derivada parcial y luego calcular su valor para el par $x_1^{(k)}, x_2^{(k)}$.

Pero en segundo lugar, y tal vez más importante, en cada iteración debemos invertir la *matriz jacobiana*. Vimos cuando analizamos cómo resolver un sistema de ecuaciones lineales que invertir la matriz no era un procedimiento sencillo ni fácil de llevar a cabo y por esa razón ningún método estudiado invertía la matriz de coeficientes. Busquemos una forma de evitar esto.

Volvamos un poco atrás y analicemos esto:

$$\mathbf{J}(\mathbf{x}^{(k)}) \cdot (\bar{\mathbf{x}} - \mathbf{x}^{(k)}) = -\mathbf{F}(\mathbf{x}^{(k)}). \quad (3.159)$$

Si hacemos:

$$\mathbf{J}(\mathbf{x}^{(k)}) \cdot (\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}) = -\mathbf{F}(\mathbf{x}^{(k)}), \quad (3.160)$$

esta expresión no es otra cosa que el *Método de Newton*. Definamos un nuevo vector $\mathbf{s}^{(k)}$ de la siguiente manera:

$$\mathbf{s}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}, \quad (3.161)$$

y reemplacemos en la expresión anterior:

$$\mathbf{J}(\mathbf{x}^{(k)}) \cdot \mathbf{s}^{(k)} = -\mathbf{F}(\mathbf{x}^{(k)}). \quad (3.162)$$

Ahora lo que tenemos es, . . . ¡un Sistema de Ecuaciones Lineales! Por lo tanto, para hallar nuestra solución debemos hacer:

$$\mathbf{J}(\mathbf{x}^{(k)}) \cdot \mathbf{s}^{(k)} = -\mathbf{F}(\mathbf{x}^{(k)}), \quad (3.163)$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{s}^{(k)}. \quad (3.164)$$

Este esquema resulta ser más sencillo que el anterior, pues ahora no es necesario invertir la matriz $\mathbf{J}(\mathbf{x}^{(k)})$ en cada iteración, aunque debemos recalcular para cada paso dicha matriz, así como el vector $\mathbf{F}(\mathbf{x}^{(k)})$. De todos modos, estamos mucho mejor que al principio, pues ahora podemos analizar nuestro sistema como si fuera un sistema de ecuaciones lineales y por lo tanto, aplicar cualquier método de los estudiados. Sólo dependemos de las características propias de la matriz jacobiana para decidir si aplicamos un método directo o un método iterativo.

Pero el hecho de tener que calcular la *matriz jacobiana* para cada iteración no resulta muy alentador. Tenemos una alternativa, que podemos implementar. Cuando estudiamos el *Método de Newton-Raphson* vimos que podíamos aproximar la derivada mediante un método discreto que resultaba en el *Método de la Secante*. Si bien el orden de convergencia es menor, la gran ventaja de este método es no tener que ingresar la derivada de la función. Este mismo procedimiento para el caso de *Sistemas de Ecuaciones No Lineales* se conoce como *Métodos Quasi-Newton*. Por lo tanto, estimemos las derivadas parciales en forma discreta de la siguiente forma:

$$\frac{\partial f_j(x_1, x_2)}{\partial x_1} = \frac{f_j(x_1 + h_1, x_2) - f_j(x_1, x_2)}{h_1}, \quad (3.165)$$

$$\frac{\partial f_j(x_1, x_2)}{\partial x_2} = \frac{f_j(x_1, x_2 + h_2) - f_j(x_1, x_2)}{h_2}. \quad (3.166)$$

Como hemos aproximado la *matriz jacobiana*, y nuestro esquema (algoritmo) es iterativo, empecemos definiendo que nuestra estimación inicial sea la siguiente:

$$\mathbf{x}^{(0)} \rightarrow \mathbf{F}(\mathbf{x}^{(0)}); \mathbf{J}(\mathbf{x}^{(0)}), \quad (3.167)$$

y con la que obtenemos nuestra primera aproximación iterativa $\mathbf{x}^{(1)}$. Ahora contamos con dos soluciones, $\mathbf{x}^{(0)}$ y $\mathbf{x}^{(1)}$. Como la idea es aproximar el jacobiano para $k = 1$, necesario para obtener nuestra siguiente aproximación, $\mathbf{x}^{(2)}$, planteemos que

$$\mathbf{A}^{(1)} \cdot (\mathbf{x}^{(1)} - \mathbf{x}^{(0)}) = \mathbf{F}(\mathbf{x}^{(1)}) - \mathbf{F}(\mathbf{x}^{(0)}). \quad (3.168)$$

Podríamos decir que $\mathbf{A}^{(1)}$ es algo parecido a la matriz jacobiana, aunque no existe nada que pueda definirse como la inversa del vector $\mathbf{x}^{(1)} - \mathbf{x}^{(0)}$. Para relacionar la matriz $\mathbf{A}^{(1)}$ con una matriz jacobiana, vamos a plantear lo siguiente:

$$\mathbf{A}^{(1)} \cdot \mathbf{z} = \mathbf{J}(\mathbf{x}^{(0)}) \cdot \mathbf{z} \quad \text{siempre que} \quad (\mathbf{x}^{(1)} - \mathbf{x}^{(0)})^T \cdot \mathbf{z} = 0. \quad (3.169)$$

Esta condición surge de considerar que todo vector distinto de cero puede ser expresarse como la suma de un múltiplo de $\mathbf{x}^{(1)} - \mathbf{x}^{(0)}$ y un múltiplo de un vector en el complemento ortogonal de $\mathbf{x}^{(1)} - \mathbf{x}^{(0)}$. Para que podamos definir a $\mathbf{A}^{(1)}$ sin ninguna información acerca del comportamiento de F en dicho complemento ortogonal, debemos imponer que $(\mathbf{x}^{(1)} - \mathbf{x}^{(0)})^T \cdot \mathbf{z} = 0$.

De acuerdo con [4], las ecuaciones anteriores determinan unívocamente lo siguiente:

$$\mathbf{A}^{(1)} = \mathbf{J}(\mathbf{x}^{(0)}) + \frac{[\mathbf{F}(\mathbf{x}^{(1)}) - \mathbf{F}(\mathbf{x}^{(0)}) - \mathbf{J}(\mathbf{x}^{(0)}) \cdot (\mathbf{x}^{(1)} - \mathbf{x}^{(0)})] \cdot (\mathbf{x}^{(1)} - \mathbf{x}^{(0)})^T}{\|\mathbf{x}^{(1)} - \mathbf{x}^{(0)}\|_2^2}. \quad (3.170)$$

Como hemos calculado $\mathbf{x}^{(1)}$, $\mathbf{F}(\mathbf{x}^{(1)})$ y aproximado $\mathbf{J}(\mathbf{x}^{(1)})$ con $\mathbf{A}^{(1)}$, podemos plantear la segunda iteración como:

$$\mathbf{A}^{(1)} \cdot \mathbf{s}^{(1)} = -\mathbf{F}(\mathbf{x}^{(1)}), \quad (3.171)$$

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \mathbf{s}^{(1)}. \quad (3.172)$$

Pero también podemos hacer

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} - \mathbf{A}^{(1)^{-1}} \cdot \mathbf{F}(\mathbf{x}^{(1)}), \quad (3.173)$$

lo que parece una total contradicción pues hemos desarrollado todo el método anterior para no tener que invertir una matriz.

Nuevamente, la matemática (más precisamente el álgebra matricial) viene a ayudarnos. En su momento hemos definido que $\mathbf{s}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$, por lo tanto también se cumple que $\mathbf{s}^{(k-1)} = \mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}$. También definimos que $\Delta_1 \mathbf{F}(\mathbf{x}^{(0)}) = \mathbf{F}(\mathbf{x}^{(1)}) - \mathbf{F}(\mathbf{x}^{(0)})$. Por simplicidad, definamos $\mathbf{y}^{(0)} = \Delta_1 \mathbf{F}(\mathbf{x}^{(0)})$. Si generalizamos, podemos definir $\mathbf{y}^{(k-1)} = \Delta_1 \mathbf{F}(\mathbf{x}^{(k-1)})$. Finalmente, podemos decir que $\mathbf{J}(\mathbf{x}^{(0)}) = \mathbf{A}^{(0)}$.

Con todo esto podemos generalizar un método para obtener la sucesivas matrices $\mathbf{A}^{(k)}$. Si escribimos que

$$\mathbf{A}^{(1)} = \mathbf{A}^{(0)} + \frac{[\mathbf{y}^{(0)} - \mathbf{A}^{(0)} \cdot \mathbf{s}^{(0)}] \cdot \mathbf{s}^{(0)^T}}{\|\mathbf{s}^{(0)}\|_2^2}, \quad (3.174)$$

entonces también podemos escribir

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \frac{[\mathbf{y}^{(k-1)} - \mathbf{A}^{(k-1)} \cdot \mathbf{s}^{(k-1)}] \cdot \mathbf{s}^{(k-1)^T}}{\|\mathbf{s}^{(k-1)}\|_2^2}, \quad (3.175)$$

y obtenemos nuestras aproximaciones $\mathbf{A}^{(k)}$ en forma iterativa. Eso nos permite obtener $\mathbf{A}^{(k)^{-1}}$ también en forma iterativa, pues existe el siguiente algoritmo para calcularla ⁷:

$$\mathbf{A}^{(k)^{-1}} = \mathbf{A}^{(k-1)^{-1}} + \frac{[\mathbf{s}^{(k-1)} - \mathbf{A}^{(k-1)^{-1}} \cdot \mathbf{y}^{(k-1)}] \cdot \mathbf{s}^{(k-1)^T} \cdot \mathbf{A}^{(k-1)^{-1}}}{\mathbf{s}^{(k-1)^T} \cdot \mathbf{A}^{(k-1)^{-1}} \cdot \mathbf{y}^{(k-1)}}, \quad (3.176)$$

⁷Se trata de la fórmula de inversión matricial de Sherman y Morrison.

para $\mathbf{A}^{(k)}$ no singular.

A partir de esto, nuestro algoritmo de solución resulta ser

$$\begin{aligned}
 \mathbf{x}^{(0)} &\rightarrow \mathbf{F}(\mathbf{x}^{(0)}) \rightarrow \mathbf{J}(\mathbf{x}^{(0)}) = \mathbf{A}^{(0)} \\
 \mathbf{x}^{(1)} &= \mathbf{x}^{(0)} - \mathbf{A}^{(0)^{-1}} \cdot \mathbf{F}(\mathbf{x}^{(0)}) \rightarrow \mathbf{F}(\mathbf{x}^{(1)}) \\
 \mathbf{s}^{(k-1)} &= \mathbf{x}^{(k)} - \mathbf{x}^{(k-1)} \\
 \mathbf{y}^{(k-1)} &= \mathbf{F}(\mathbf{x}^{(k)}) - \mathbf{F}(\mathbf{x}^{(k-1)}) \\
 \mathbf{A}^{(k)^{-1}} &= \mathbf{A}^{(k-1)^{-1}} + \frac{\left[\mathbf{s}^{(k-1)} - \mathbf{A}^{(k-1)^{-1}} \cdot \mathbf{y}^{(k-1)} \right] \cdot \mathbf{s}^{(k-1)^T} \cdot \mathbf{A}^{(k-1)^{-1}}}{\mathbf{s}^{(k-1)^T} \cdot \mathbf{A}^{(k-1)^{-1}} \cdot \mathbf{y}^{(k-1)}} \\
 \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} - \mathbf{A}^{(k)^{-1}} \cdot \mathbf{F}(\mathbf{x}^{(k)}).
 \end{aligned} \tag{3.177}$$

Este algoritmo requiere invertir la matriz $\mathbf{A}^{(0)}$, para luego utilizar solamente la multiplicación de matrices para obtener el resultado. Este método se conoce como *Método de Broyden de primer orden*, si bien la convergencia puede tender a ser mayor si se utiliza la $\mathbf{J}(\mathbf{x}^{(0)})$ analítica.

La necesidad de invertir la matriz le quita practicidad al método. Es por eso que en algunos libros se indica que, dado que el cálculo de la inversa de la *matriz jacobiana* $\mathbf{J}(\mathbf{x}^{(k)})$ es en realidad una aproximación numérica de la misma, puede proponerse el mismo algoritmo visto pero con un cambio: $\mathbf{A}^{(0)} = \mathbf{I}$. De esta forma, no es necesario aplicar ningún método para invertir $\mathbf{A}^{(0)}$, al costo de aumentar la cantidad de iteraciones para obtener un mejor resultado.

3.13. Notas finales

Los métodos vistos no son los únicos disponibles para resolver sistemas de ecuaciones lineales y no lineales. Dentro de los métodos directos también están el *Método QR* y el de la *Descomposición por el Valor Singular* (un equivalente a los autovalores para matrices no cuadradas), método muy usado con matrices muy mal condicionadas, aunque algunos autores sostienen que debería ser un método básico, igual que *Eliminación de Gauss*.

Algo similar ocurre con los métodos iterativos, particularmente con los no estacionarios. Además de los tres que hemos visto, están el *Método de los Residuos Mínimos Generalizado*, el *Método de los Gradientes Biconjugados*, el *Método de los Gradientes Conjugados Cuadrático*, el *Método por Iteraciones de Chebichev*, más otros derivados fundamentalmente a partir del *Método de los Gradientes Conjugados* y de los *Residuos Mínimos*.

Para resolver los sistemas de ecuaciones no lineales también se puede aplicar una versión del *Método del Descenso Más Empinado* y el *Método de Broyden de segunda especie*.

La existencia de varios métodos refleja que la elección de uno en particular depende fundamentalmente de las propiedades de la matriz de coeficientes del sistema. Es por esto que cada vez es más importante saber qué problema (o fenómeno físico) está siendo representado con el sistema a resolver. Si buscamos información sobre la utilización de cada método, veremos que están muy ligados al tipo de problema que se estudia y resuelve.

En muchos campos de la ingeniería, los sistemas de ecuaciones lineales están directamente relacionados con la resolución de ecuaciones diferenciales en derivadas parciales, por eso es que métodos para resolver este tipo de problemas, como el de las diferencias finitas o de los elementos finitos, han impulsado el desarrollo de métodos más potentes y más precisos, dado que mayormente trabajan con matrices de dimensiones muy grandes que además suelen ser ralas, simétricas y definidas positivas. Ejemplo de esto es el análisis estructural en tres dimensiones, que modela las piezas a dimensionar o verificar, en los cuales los programas generan sistemas de ecuaciones lineales con las características antes mencionadas, por aplicación del *Método de los Elementos Finitos*.

Quien quiera adentrarse en los métodos iterativos no estacionarios, el libro de Y. Saad es una muestra muy interesante de cómo la necesidad de contar con algoritmos cada vez más veloces y con capacidad de resolver grandes sistemas de ecuaciones, disparan el desarrollo y la investigación de la matemática aplicada.

Ejercicios

Sistemas de Ecuaciones Lineales

Método de Eliminación de Gauss

1. Aplique el Método de Eliminación de Gauss para resolver los sistemas de ecuaciones indicados. No reordene las ecuaciones.

$$\begin{array}{rcl} 4x_1 - x_2 + x_3 & = & 8 \\ 2x_1 + 5x_2 + 2x_3 & = & 3 \\ x_1 + 2x_2 + 4x_3 & = & 11 \end{array} \qquad \begin{array}{rcl} 4x_1 + x_2 + 2x_3 & = & 9 \\ 2x_1 + 4x_2 - x_3 & = & -5 \\ x_1 + x_2 - 3x_3 & = & -9 \end{array}$$

2. Como en el ejercicio anterior, aplique el Método de Eliminación de Gauss para resolver los sistemas de ecuaciones lineales indicados. Determine si es necesario intercambiar filas.

$$\begin{array}{rcl} x_1 - x_2 + 3x_3 & = & 2 \\ 3x_1 - 3x_2 + x_3 & = & -1 \\ x_1 + 2x_2 & = & 3 \end{array} \qquad \begin{array}{rcl} x_1 + x_2 + x_4 & = & 2 \\ 2x_1 + x_2 - x_3 + x_4 & = & 1 \\ 4x_1 - x_2 - 2x_3 + 2x_4 & = & 0 \\ 3x_1 - x_2 - x_3 + 2x_4 & = & -3 \end{array}$$

3. Aplique nuevamente el Método de Eliminación de Gauss y utilice solamente tres decimales en todas las operaciones para resolver los sistemas de ecuaciones lineales indicados a continuación:

$$\begin{array}{rcl} 0,03x_1 + 58,9x_2 & = & 59,2 \\ 5,31x_1 - 6,10x_2 & = & 47,0 \end{array} \qquad \begin{array}{rcl} 3,03x_1 - 12,1x_2 + 14x_3 & = & -119 \\ -3,03x_1 + 12,1x_2 - 7x_3 & = & 120 \\ 6,11x_1 - 14,2x_2 + 21x_3 & = & -139 \end{array}$$

Factorización de matrices

1. Factorice las siguientes matrices aplicando la descomposición LU:

$$a) \begin{bmatrix} 2 & -1 & 1 \\ 3 & 3 & 9 \\ 3 & 3 & 5 \end{bmatrix} \qquad b) \begin{bmatrix} 1,012 & -2,132 & 3,104 \\ -2,132 & 4,096 & -7,013 \\ 3,104 & -7,013 & 0,014 \end{bmatrix}$$

2. Aplique el método de descomposición LU para resolver estos sistemas de ecuaciones lineales:

$$a) \begin{bmatrix} 2 & 3 & -1 \\ 4 & 4 & -1 \\ -2 & -3 & 2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} \qquad b) \begin{bmatrix} 2 & 2 & 2 \\ -1 & 0 & 1 \\ 3 & 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 \\ 3 \\ 0 \end{bmatrix}$$

3. Factorice las siguientes matrices por el método de Cholesky

$$a) \begin{bmatrix} 5 & -3,2 & 0 \\ -3,2 & 6 & -2,7 \\ 0 & -2,7 & 4 \end{bmatrix} \qquad b) \begin{bmatrix} 4 & 2,25 & 1,275 & -1 \\ 2,25 & 6 & 3,875 & 1 \\ 1,275 & 3,875 & 8 & -2,5 \\ -1 & 1 & -2,5 & 9 \end{bmatrix}$$

4. Aplique los métodos de Factorización LU y de Cholesky para resolver los siguientes sistemas de ecuaciones lineales. Para el último, verificar que la matriz cumpla con las condiciones que impone el método.

$$a) \begin{bmatrix} 7 \cdot 10^5 & -3,25 \cdot 10^5 & 0 \\ -3,25 \cdot 10^5 & 6 \cdot 10^5 & -2,75 \cdot 10^5 \\ 0 & -2,75 \cdot 10^5 & 5 \cdot 10^5 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0,75 \\ 0,5 \\ 0,25 \end{bmatrix}$$

$$b) \begin{bmatrix} 6,8 \cdot 10^3 & -4,08 \cdot 10^3 & -3,4 \cdot 10^3 & 0 \\ -4,08 \cdot 10^3 & 6,8 \cdot 10^3 & -2,04 \cdot 10^3 & 0 \\ -3,4 \cdot 10^3 & -2,04 \cdot 10^3 & 1,19 \cdot 10^4 & -1,088 \cdot 10^4 \\ 0 & 0 & -1,088 \cdot 10^4 & 1,19 \cdot 10^4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1,4 \end{bmatrix}$$

Métodos de Jacobi, Gauss-Seidel, SOR Y de los Gradientes Conjugados

1. Aplique los métodos de *Jacobi*, *Gauss-Seidel*, *SOR* y de los *Gradientes Conjugados* (verificando las condiciones que debe cumplir la matriz A) para resolver los siguientes sistemas de ecuaciones lineales:

$$a) \begin{bmatrix} 3 & -1 & 3 \\ 3 & 6 & 2 \\ 3 & 3 & 7 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 4 \end{bmatrix} \quad b) \begin{bmatrix} 10 & -1 & 0 \\ -1 & 10 & -2 \\ 0 & -2 & 10 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 9 \\ 7 \\ 6 \end{bmatrix}$$

$$c) \begin{bmatrix} 10 & 5 & 0 & 0 \\ 5 & 10 & -4 & 0 \\ 0 & -4 & 8 & -1 \\ 0 & 0 & -1 & 5 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 6 \\ 25 \\ -11 \\ -11 \end{bmatrix} \quad d) \begin{bmatrix} 4 & 1 & -1 & 1 \\ 1 & 4 & -1 & -1 \\ -1 & -1 & 5 & 1 \\ 1 & -1 & 1 & 3 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -2 \\ -1 \\ 0 \\ 1 \end{bmatrix}$$

2. Aplique los mismos métodos del punto anterior para resolver las ecuaciones del punto 4.

Sistemas de Ecuaciones No Lineales

1. Obtenga las raíces de este sistema de ecuaciones mediante el *Método de Newton*:

$$f_1(x, y) = x^2 + y^2 - 4 = 0$$

$$f_2(x, y) = xy - 1 = 0$$

2. Obtenga las raíces de las siguientes ecuaciones:

$$a) \begin{aligned} f_1(x) &= 3x_1^2 - x_2^2 = 0 \\ f_2(x) &= 3x_1 x_2^2 - x_1^3 - 1 = 0 \end{aligned} \quad b) \begin{aligned} f_1(x) &= 4x_1^2 - 20x_1 + 0,25x_2^2 + 8 = 0 \\ f_2(x) &= 0,5x_1 x_2^2 + 2x_1 - 5x_2 + 8 = 0 \end{aligned}$$

Para el caso $a)$ tome $x^{(0)} = [1; 1]$ y para el caso $b)$ tome $x^{(0)} = [0; 0]$. En ambos casos, tome $\varepsilon = 10^{-6}$.

3. Obtenga las raíces de los siguientes sistemas de ecuaciones no lineales:

$$a) \begin{aligned} f_1(\mathbf{x}) &= x_1 + x_3 = 2 \\ f_2(\mathbf{x}) &= x_1 x_2 + x_3 x_4 = 0 \\ f_3(\mathbf{x}) &= x_1 x_2^2 + x_3 x_4^2 = \frac{2}{3} \\ f_4(\mathbf{x}) &= x_1 x_2^3 + x_3 x_4^3 = 0 \end{aligned} \quad b) \begin{aligned} f_1(\mathbf{x}) &= x_1 + x_3 + x_5 = 2 \\ f_2(\mathbf{x}) &= x_1 x_2 + x_3 x_4 + x_5 x_6 = 0 \\ f_3(\mathbf{x}) &= x_1 x_2^2 + x_3 x_4^2 + x_5 x_6^2 = \frac{2}{3} \\ f_4(\mathbf{x}) &= x_1 x_2^3 + x_3 x_4^3 + x_5 x_6^3 = 0 \\ f_5(\mathbf{x}) &= x_1 x_2^4 + x_3 x_4^4 + x_5 x_6^4 = \frac{2}{5} \\ f_6(\mathbf{x}) &= x_1 x_2^5 + x_3 x_4^5 + x_5 x_6^5 = 0 \end{aligned}$$

Considere que en $a)$, $\mathbf{x} = (x_1, x_2, x_3, x_4)$, y en $b)$, $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6)$. En ambos casos, tome $\varepsilon = 10^{-6}$.

Capítulo 4

Interpolación de curvas

4.1. Introducción

En este capítulo nos concentraremos en el estudio de los métodos de interpolación de curvas. Es usual que los ingenieros trabajen con datos extraídos de mediciones, relevamientos, ensayos de laboratorio, etc., los cuales no siempre entregan el valor necesitado para el problema que se está tratando de resolver. Un ejemplo típico de interpolación sencilla utilizado por cualquier profesional de la ingeniería es la interpolación lineal en una tabla de datos (por ejemplo, de estadísticas) para obtener un valor entre dos puntos dados. Este tipo de interpolación lineal era muy usado cuando no existían las calculadoras científicas de bolsillo (ni hablar de computadoras) y debían usarse las famosas *Tablas de logaritmos* para obtener logaritmos, senos, cosenos y cualquier otra función trigonométrica o trascendente.

Hoy estamos acostumbrados a usar programas que representan gráficamente funciones matemáticas de variadas formas en la pantalla de una computadora. Todos o casi todos esos programas aplican algún método de interpolación. La gran mayoría utiliza una interpolación lineal con una gran cantidad de puntos, de ahí que la unión de esa gran cantidad de puntos con segmentos de recta, crean la ilusión de dibujar una «curva». Un ejemplo en este sentido son los gráficos siguientes:

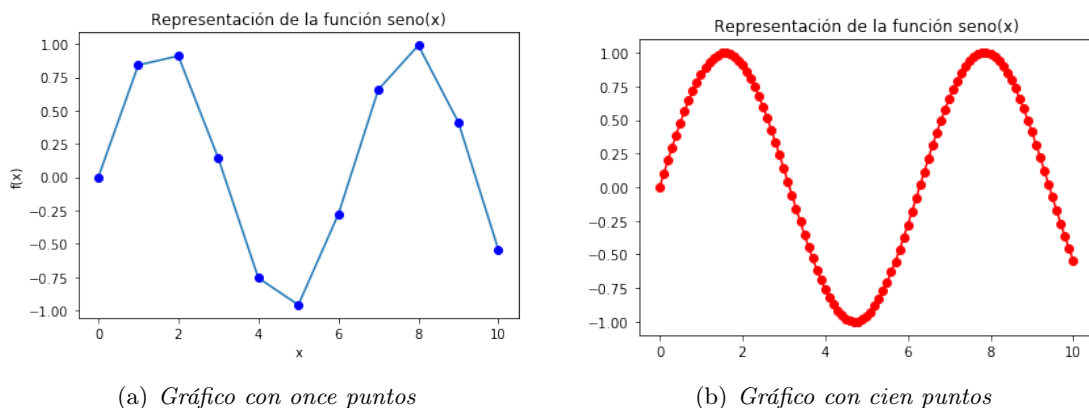


Figura 4.1: Representación gráfica de la función $\text{sen}(x)$.

Estas dos representaciones se han hecho con *Python* y *Matplotlib*. En el primer caso se dejó que el *Matplotlib* representara la función $\text{sen}(x)$ entre 0 y 10. El programa lo hizo calculando los valores para $x = 0; 1; 2; \dots; 10$. En el segundo, se le indicó que lo hiciera con los siguientes valores: $x = 0; 0,1; 0,2; \dots; 10$. El segundo gráfico parece una curva, sin embargo, si se mira con más detalle, entre dos puntos sucesivos hay un segmento de recta que los une. Como a propósito

se representaron cada uno de los puntos usados para graficar la función, en donde los valores se asemejan mucho, se observa la superposición de puntos y no se ven los segmentos de recta que los unen.

Otro ejemplo de interpolación muy interesante es la función «spline» del AutoCAD[®], que permite dibujar curvas que pasen por puntos determinados en el dibujo, y que los usuarios no siempre saben usar en forma eficiente.

Otro ejemplo de interpolación más avanzado es la utilización de polinomios interpolantes en la resolución de estructuras cuando se utilizan programas de análisis estructural que aplican el *Método de los Elementos Finitos*. Allí es de fundamental importancia entender los tipos de polinomios que se pueden usar y los datos necesarios para poder obtener estos polinomios.

También un uso que suele darse a la interpolación es para obtener qué valor de x hace $y(x)$ nulo, cuando disponemos de un conjunto de datos en los cuales se tiene que $y(x_j) > 0$ para $j = 0; 1; \dots; i$ e $y(x_k) < 0$ para $k = i + 1, i + 2, \dots, n$ (o a la inversa). Interpolando entre estos valores podemos hallar \hat{x} tal que $y(\hat{x}) = 0$. Este tipo de interpolación se denomina *interpolación inversa*, pues en lugar de interpolar el conjunto $[x_i, y_i]$, interpolamos el conjunto $[y_i, x_i]$ para obtener una función $x(y)$.

Puesto que hay muchos métodos y formas de interpolar, nos ocuparemos de los métodos clásicos y veremos algunas mejoras que se han desarrollado a estos métodos. En particular, gracias al artículo de L. N. Trefethen y J. P. Berrut (véase [20]), analizaremos una mejora al método de Lagrange básico, denominada *Interpolación Baricéntrica de Lagrange*.

4.2. Interpolación o Método de Lagrange

Supongamos que tenemos una lista con datos ordenados en pares como en la tabla 4.1, y que queremos conocer el valor de $y(x_A)$ para un x_A entre x_1 y x_2 .

Tabla 4.1: Datos para una interpolación

x	y
x_0	y_0
x_1	y_1
x_2	y_2
x_3	y_3

La forma sencilla de obtener este valor es graficar estos puntos y trazar un segmento de recta que una y_1 e y_2 , ubicar x_A en las abscisas y trazar por él una línea recta paralela al eje de ordenadas que corte el segmento ya mencionado. Finalmente, desde este punto, trazamos una línea recta paralela al eje de abscisas hasta cortar el eje de ordenadas, con lo cual hemos obtenido el valor de $y(x_A)$.

Queda muy evidente que este procedimiento es muy engorroso si se quiere hacerlo en forma metódica. Sin embargo, es la forma más sencilla de interpolación polinomial, la interpolación lineal. Efectivamente, si tomamos los dos puntos en cuestión podemos armar una recta mediante el siguiente sistema:

$$y_1 = m x_1 + n \quad (4.1)$$

$$y_2 = m x_2 + n \quad (4.2)$$

Si restamos y_1 a y_2 obtenemos m :

$$y_2 - y_1 = m(x_2 - x_1) \Rightarrow m = \frac{y_2 - y_1}{x_2 - x_1}. \quad (4.3)$$

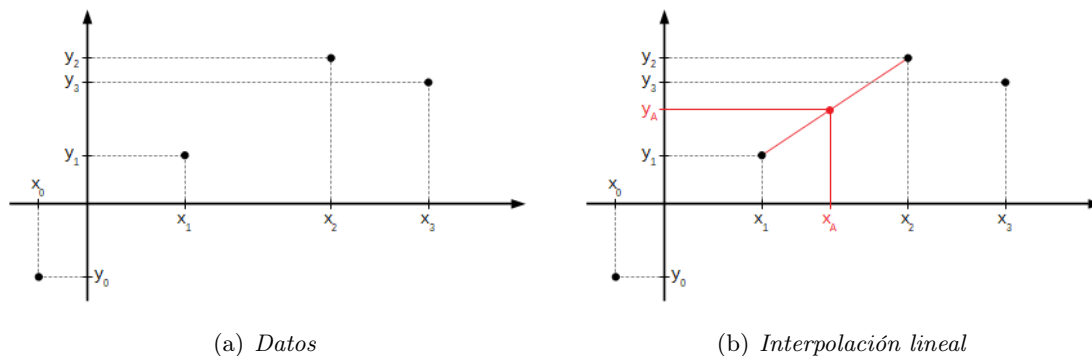


Figura 4.2: Interpolación lineal de un conjunto de datos.

Reemplacemos m en la primera ecuación obtenemos n :

$$y_1 = \frac{y_2 - y_1}{x_2 - x_1} x_1 + n \Rightarrow n = y_1 - \frac{y_2 - y_1}{x_2 - x_1} x_1. \quad (4.4)$$

Finalmente la ecuación de la recta que pasa por y_1 e y_2 es:

$$y(x) = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1) + y_1, \quad (4.5)$$

que también puede escribirse como

$$y(x) = y_1 \frac{x - x_2}{x_1 - x_2} + y_2 \frac{x - x_1}{x_2 - x_1}. \quad (4.6)$$

Para hallar $y(x_A)$ basta con reemplazar x_A en cualquiera de las expresiones anteriores. Los dos procedimientos anteriores son equivalentes al procedimiento gráfico de la figura 4.2. Pero, ¿qué pasa si queremos usar más de dos puntos? Supongamos que necesitamos usar los cuatro puntos de la tabla 4.1 para interpolar un punto cualquiera entre x_0 y x_3 . En ese caso, el polinomio de mayor grado posible es un polinomio cúbico, porque tiene cuatro coeficientes, y se puede expresar así:

$$y(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3. \quad (4.7)$$

Al reemplazar los cuatro puntos en esta ecuación obtenemos el siguiente sistema de ecuaciones lineales:

$$y_0 = a_0 + a_1 x_0 + a_2 x_0^2 + a_3 x_0^3, \quad (4.8)$$

$$y_1 = a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3, \quad (4.9)$$

$$y_2 = a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3, \quad (4.10)$$

$$y_3 = a_0 + a_1 x_3 + a_2 x_3^2 + a_3 x_3^3, \quad (4.11)$$

con las incógnitas a_i , coeficientes del polinomio.

Basta con resolver este sistema de ecuaciones lineales para obtener esos coeficientes. Analicemos el sistema escribiéndolo en forma matricial:

$$\underbrace{\begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ 1 & x_3 & x_3^2 & x_3^3 \end{bmatrix}}_A \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} \quad (4.12)$$

La matriz \mathbf{A} de este sistema de ecuaciones lineales es una matriz muy conocida denominada *matriz de VanderMonde*. Tiene la particularidad de ser mal condicionada (ver capítulo 3). Resolver este sistema es relativamente fácil, pues disponemos de varios métodos para resolverlo. Sin embargo, para evitar resolver un sistema de ecuaciones lineales, existen varios procedimientos para generar una función interpolante en forma analítica.

Uno de esos procedimientos es la *Interpolación o Método de Lagrange*¹. El polinomio interpolador lo obtenemos siguiendo estos pasos:

1. Calculamos los $n + 1$ polinomios $L_{n;i}(x)$ relacionados cada uno con cada dato x_i , donde n es el grado del polinomio e i indica el punto considerado, mediante:

$$L_{n,i}(x) = \frac{\prod_{\substack{j=0 \\ j \neq i}}^n (x - x_j)}{\prod_{\substack{j=0 \\ j \neq i}}^n (x_i - x_j)} = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}. \quad (4.13)$$

con $i = 0; 1; \dots; n$, $j = 0; 1; \dots; n$, y x_i y x_j referen a los datos disponibles. Estos polinomios cumplen con la particularidad de que:

$$L_{n,i}(x) = \begin{cases} 1 & \text{si } x = x_i \\ 0 & \text{si } x = x_j \text{ con } j \neq i. \end{cases} \quad (4.14)$$

2. El polinomio interpolador lo obtenemos mediante la expresión:

$$P_n(x) = \sum_{i=0}^n y_i L_{n,i}(x). \quad (4.15)$$

Por ejemplo, podemos armar una interpolación lineal aplicando este método o procedimiento entre los puntos x_1 y x_2 . Al aplicar el método obtenemos:

$$\begin{aligned} L_{1;0} &= \frac{x - x_2}{x_1 - x_2} \\ L_{1;1} &= \frac{x - x_1}{x_2 - x_1} \\ P_1(x) &= y_1 L_{1;0}(x) + y_2 L_{1;1}(x) \\ P_1(x) &= y_1 \frac{x - x_2}{x_1 - x_2} + y_2 \frac{x - x_1}{x_2 - x_1}, \end{aligned} \quad (4.16)$$

que es la ecuación de la recta que obtuvimos antes.

Para obtener el polinomio de tercer grado tendremos:

$$\begin{aligned} L_{3;0}(x) &= \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)}, \\ L_{3;1}(x) &= \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)}, \\ L_{3;2}(x) &= \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)}, \\ L_{3;3}(x) &= \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)}, \\ P_3(x) &= y_0 L_{3;0}(x) + y_1 L_{3;1}(x) + y_2 L_{3;2}(x) + y_3 L_{3;3}(x). \end{aligned} \quad (4.17)$$

¹Si bien se atribuye a Lagrange, quien primero lo desarrolló fue el matemático inglés Edward Waring (1736-1798).

Como hemos utilizado todos los puntos de los datos, es evidente que no podemos crear un polinomio de mayor grado que el cúbico. Por lo tanto, existe un sólo polinomio posible de construir con todos los datos disponibles. El siguiente teorema define a este único polinomio.

Teorema 4.1. Sean x_0, x_1, \dots, x_n , $n + 1$ números diferentes, y sea f una función tal que sus valores se obtengan a partir de los números dados ($f(x_0), f(x_1), \dots, f(x_n)$), entonces existe un único polinomio $P_n(x)$ de grado n , que cumple con la propiedad

$$f(x_k) = P(x_k) \text{ para cada } k = 0; 1; \dots; n;$$

y este polinomio está dado por la siguiente expresión

$$P_n(x) = f(x_0)L_{n,0}(x) + f(x_1)L_{n,1}(x) + \dots + f(x_n)L_{n,n}(x) = \sum_{i=0}^n f(x_i)L_{n,i}(x),$$

donde

$$L_{n,i}(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j},$$

para $i = 0; 1; \dots; n$.

Sin embargo, podemos crear varios polinomios de grados menores a n . Así, con los datos de la tabla 4.1 estamos en condiciones de construir al menos tres polinomios de grado 1 y dos polinomios de grado 2. (Podemos construir más polinomios para ambos grados, pero no siempre son de utilidad práctica.)

Obtenido el polinomio interpolante nos queda un punto por definir: ¿cuál es el error que estamos cometiendo al interpolar mediante un polinomio respecto de la función original? El siguiente teorema nos permite aproximar ese error.

Teorema 4.2. Sean $x_0, x_1, x_2, \dots, x_n$, números distintos en el intervalo $[a, b]$ y sea $f \in C^{n+1}[a, b]$. Entonces, para cualquier $x \in [a, b]$ existe un número $\xi(x) \in [a, b]$ para que se cumple que

$$f(x) = P_n(x) + \frac{f^{(n+1)}(\xi(x))}{(n+1)!} \prod_{i=0}^n (x - x_i),$$

donde $P_n(x)$ es el máximo polinomio interpolante.

Demostración Si $x = x_i$ para $i = 0; 1; 2; \dots; n$ entonces $f(x_i) = P_n(x_i)$ y para cualquier $\xi(x_i) \in [a, b]$ se cumple lo expresado en el teorema. En cambio, si $x \neq x_i$ para $i = 0; 1; 2; \dots; n$, se puede definir la siguiente función $g(u)$ para $u \in [a; b]$:

$$g(u) = f(u) - P_n(u) - [f(x) - P_n(x)] \prod_{i=0}^n \frac{(u - x_i)}{(x - x_i)}.$$

Como $f \in C^{n+1}[a, b]$, $P_n \in C^\infty[a, b]$, y $x \neq x_i$ para cualquier i , entonces $g \in C^{n+1}[a, b]$. Si $u = x_j$ tendremos que

$$g(x_j) = f(x_j) - P_n(x_j) - [f(x) - P_n(x)] \prod_{i=0}^n \frac{(x_j - x_i)}{(x - x_i)} = 0 - [f(x) - P_n(x)]0 = 0.$$

También tenemos que $g(x) = 0$, pues

$$g(x) = f(x) - P_n(x) - [f(x) - P_n(x)] \prod_{i=0}^n \frac{(x - x_i)}{(x - x_i)} = f(x) - P_n(x) - [f(x) - P_n(x)] = 0,$$

y en consecuencia, $g \in C^{n+1}[a, b]$ y se anula para x, x_0, x_1, \dots, x_n , es decir, para $n + 2$ números distintos. De acuerdo con el Teorema de Rolle, existe entonces un $\xi \in (a, b)$ tal que $g^{(n+1)}(\xi) = 0$. Así tendremos que

$$0 = g^{(n+1)}(\xi) = f^{(n+1)}(\xi) - P_n^{(n+1)}(\xi) - [f(x) - P_n(x)] \frac{d^{n+1}}{du^{n+1}} \left[\prod_{i=0}^n \frac{(u - x_i)}{(x - x_i)} \right]_{u=\xi}.$$

Como $P_n(u)$ es un polinomio de grado n , entonces $P_n^{(n+1)}(u) = 0$. A su vez, $\prod_{i=0}^n \frac{(u - x_i)}{(x - x_i)}$ es un polinomio de grado $n + 1$, entonces su derivada de orden $n + 1$ será

$$\frac{d^{n+1}}{du^{n+1}} \left[\prod_{i=0}^n \frac{(u - x_i)}{(x - x_i)} \right] = \frac{(n+1)!}{\prod_{i=0}^n (x - x_i)}.$$

Si reemplazamos, tendremos que

$$0 = f^{(n+1)}(\xi) - 0 - [f(x) - P_n(x)] \frac{(n+1)!}{\prod_{i=0}^n (x - x_i)}.$$

Al despejar $f(x)$ de la ecuación anterior nos queda

$$f(x) = P_n(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i).$$

Desde el punto de vista teórico, esta expresión del error es muy importante porque muchas de las técnicas de derivación e integración numérica se desarrollan al aplicar la interpolación por el método de Lagrange. Sin embargo, para otros casos, no debemos olvidarnos que no conocemos $f(x)$ (y por lo tanto, tampoco $f^{(n+1)}(x)$), por lo tanto, como expresamos antes, el error calculado es sólo una aproximación o una cota del mismo.

Finalmente, podemos ver que el método tiene algunas desventajas:

1. Cada evaluación del polinomio $P_n(x)$ requiere $O(n^2)$ operaciones aritméticas.
2. Agregar un par de datos $x_{n+1}, f(x_{n+1})$ requiere rehacer todos los polinomios $L_{n,i}(x)$.
3. Es numéricamente inestable.

4.3. Interpolación o Método de Newton

Otra forma de plantear la construcción del polinomio interpolador es la siguiente. Supongamos que queremos usar solamente los primeros tres puntos de nuestra tabla. Entonces planteemos el siguiente sistema de ecuaciones:

$$y_0 = a_0 + a_1 x_0 + a_2 x_0^2 \quad (4.18)$$

$$y_1 = a_0 + a_1 x_1 + a_2 x_1^2 \quad (4.19)$$

$$y_2 = a_0 + a_1 x_2 + a_2 x_2^2. \quad (4.20)$$

Al eliminar a_0 tenemos este nuevo sistema

$$y_1 - y_0 = a_1(x_1 - x_0) + a_2(x_1^2 - x_0^2) \quad (4.21)$$

$$y_2 - y_1 = a_1(x_2 - x_1) + a_2(x_2^2 - x_1^2), \quad (4.22)$$

que puede escribirse como

$$\frac{y_1 - y_0}{x_1 - x_0} = a_1 + a_2(x_1 + x_0) \quad (4.23)$$

$$\frac{y_2 - y_1}{x_2 - x_1} = a_1 + a_2(x_2 + x_1). \quad (4.24)$$

Si ahora eliminamos a_1 obtenemos el coeficiente a_2 que resulta ser

$$\begin{aligned} a_2(x_2 - x_0) &= \frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0} \\ a_2 &= \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}. \end{aligned} \quad (4.25)$$

Ahora reemplacemos a_2 en una de las ecuaciones anteriores para obtener a_1

$$\begin{aligned} \frac{y_1 - y_0}{x_1 - x_0} &= a_1 + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x_1 + x_0) \\ a_1 &= \frac{y_1 - y_0}{x_1 - x_0} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x_1 + x_0). \end{aligned} \quad (4.26)$$

Ahora reemplacemos a_1 y a_2 en la primera ecuación de todas para obtener a_0 :

$$\begin{aligned} y_0 &= a_0 + \left[\frac{y_1 - y_0}{x_1 - x_0} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x_1 + x_0) \right] x_0 + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} x_0^2 \\ a_0 &= y_0 - \left(\frac{y_1 - y_0}{x_1 - x_0} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} x_1 \right) x_0. \end{aligned} \quad (4.27)$$

Armemos finalmente el polinomio interpolante reemplazando a_0 , a_1 y a_2

$$\begin{aligned} P(x) &= y_0 - \left(\frac{y_1 - y_0}{x_1 - x_0} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} x_1 \right) x_0 + \\ &+ \left[\frac{y_1 - y_0}{x_1 - x_0} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x_1 + x_0) \right] x + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} x^2 \\ &= y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} [x^2 - (x_0 + x_1)x + x_0x_1] \\ &= y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x - x_0)(x - x_1). \end{aligned} \quad (4.28)$$

Esta forma de armar el polinomio se denomina *Método de las Diferencias Divididas de Newton*, y podemos sistematizarla para que sea muy sencillo de aplicar. En primer lugar, podemos decir que $f(x_i) = y_i$. Seguidamente vamos a definir que:

$$f(x_0, x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad (4.29)$$

$$f(x_1, x_2) = \frac{f(x_2) - f(x_1)}{x_2 - x_1}, \quad (4.30)$$

y generalizando

$$f(x_i, x_{i+1}) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}. \quad (4.31)$$

Análogamente tenemos que:

$$f(x_0, x_1, x_2) = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2-x_0} = \frac{f(x_1; x_2) - f(x_0; x_1)}{x_2-x_0}, \quad (4.32)$$

y si generalizamos nuevamente tenemos

$$f(x_i, x_{i+1}, x_{i+2}) = \frac{f(x_{i+1}, x_{i+2}) - f(x_i, x_{i+1})}{x_{i+2} - x_i}. \quad (4.33)$$

Finalmente podemos generalizar totalmente las expresiones anteriores a la siguiente expresión:

$$f(x_k, x_{k+1}, \dots, x_{n-1}, x_n) = \frac{f(x_{k+1}, x_{k+2}, \dots, x_n) - f(x_k, x_{k+1}, \dots, x_{n-1})}{x_n - x_k}. \quad (4.34)$$

Si utilizamos esta notación para el polinomio que hallamos más arriba nos queda:

$$P(x) = f(x_0) + f(x_0, x_1) \cdot (x - x_0) + f(x_0, x_1, x_2) \cdot (x - x_0) \cdot (x - x_1). \quad (4.35)$$

Esta forma nos permite agregar un punto más y aumentar el grado del polinomio en forma sencilla. Efectivamente, si queremos agregar x_3 , solamente debemos agregar al polinomio anterior el término $f(x_0, x_1, x_2, x_3)(x - x_0)(x - x_1)(x - x_2)$, con lo cual nos queda

$$P(x) = f(x_0) + f(x_0, x_1)(x - x_0) + f(x_0, x_1, x_2)(x - x_0)(x - x_1) + f(x_0, x_1, x_2, x_3)(x - x_0)(x - x_1)(x - x_2). \quad (4.36)$$

Así, armar los polinomios de esta manera facilita notablemente aumentar la cantidad de puntos para obtener un polinomio interpolante, pues permite usar el polinomio anterior. En la tabla 4.2 podemos ver cómo operar.

Tabla 4.2: *Interpolación o Método de Newton*

\mathbf{x}	$\mathbf{f}(\mathbf{x})$	$\mathbf{f}(\mathbf{x}_i, \mathbf{x}_{i+1})$	$\mathbf{f}(\mathbf{x}_i, \mathbf{x}_{i+1}, \mathbf{x}_{i+2})$	$\mathbf{f}(\mathbf{x}_i, \mathbf{x}_{i+1}, \mathbf{x}_{i+2}, \mathbf{x}_{i+3})$
x_0	$f(x_0)$			
x_1	$f(x_1)$	$f(x_0, x_1)$	$f(x_0, x_1, x_2)$	
x_2	$f(x_2)$	$f(x_1, x_2)$	$f(x_1, x_2, x_3)$	$f(x_0, x_1, x_2, x_3)$
x_3	$f(x_3)$	$f(x_2, x_3)$		

Observemos que podemos armar dos polinomios con todos los puntos aplicando la Interpolación o Método de Newton. Uno es el que obtuvimos antes, por el denominado *Método de la Diferencias Divididas Progresivas*. El otro podemos obtenerlo partiendo de x_3 , que resulta ser

$$P(x) = f(x_3) + f(x_2, x_3)(x - x_3) + f(x_1, x_2; x_3)(x - x_3)(x - x_2) + \quad (4.37)$$

$$+ f(x_0, x_1, x_2, x_3)(x - x_3)(x - x_2)(x - x_1),$$

que se denomina *Método de las Diferencias Divididas Regresivas*.

El *Método de Newton*, en sus dos variantes, es muy usado cuando se trabaja con datos que pueden ser modificados (aumentando la cantidad de puntos disponibles para la interpolación) y, en consecuencia, aplicar el *Método de Lagrange* se vuelve muy engorroso. Otra ventaja es que para evaluar los polinomios $P_n(x)$ requerimos n operaciones aritméticas, algo bastante menor al $O(n^2)$ que requiere el *Método de Lagrange*². Sin embargo, el método exige que los datos deban estar ordenados, según x_i , en forma ascendente (o descendente) para poder implementarlo. Si agregamos algún dato intermedio, la ventaja anterior se pierde porque la tabla 4.2 debe rehacerse, perdiendo practicidad.

Para mejorar esto existe una variante del *Método de Lagrange* que nos permite interpolar de manera sencilla y al que resulta muy fácil agregarle puntos en cualquier orden.

4.4. Interpolación baricéntrica de Lagrange

Supongamos que definimos un polinomio genérico $L(x)$ tal que

$$L(x) = (x - x_0)(x - x_1) \dots (x - x_n). \quad (4.38)$$

Definamos además los pesos baricéntricos como

$$w_i = \prod_{\substack{k=0 \\ k \neq i}}^n \frac{1}{x_i - x_k}, \text{ para todo } i = 0; 1; \dots; n. \quad (4.39)$$

Entonces podemos escribir cualquier polinomio de Lagrange como

$$L_{n,i} = L(x) \frac{w_i}{x - x_i}, \quad (4.40)$$

y, en consecuencia, el polinomio interpolante será

$$P_n(x) = \sum_{i=0}^n f(x_i) \frac{L(x)w_i}{x - x_i} = L(x) \sum_{i=0}^n f(x_i) \frac{w_i}{x - x_i}, \quad (4.41)$$

pues $L(x)$ es constante para todos los términos de la sumatoria.

Esto es una gran ventaja en dos sentidos. Primero, para evaluar $P_n(x)$ se necesitan sólo $O(n)$ operaciones, lo cual hace mucho más rápido el procedimiento. Y segundo, si agregamos el par de datos $x_{n+1}, f(x_{n+1})$, sólo debemos hacer lo siguiente:

- Dividir cada w_i por $x_i - x_{n+1}$.
- Calcular un nuevo w_{i+1} .

En ambos casos el costo computacional es de $n + 1$ operaciones. Es decir, ¡podemos actualizar el polinomio $P_n(x)$ con sólo $O(n)$ operaciones! A esta variante del *Método de Lagrange* suele llamársela *Método Mejorado de Lagrange* y tiene una ventaja adicional respecto al método de Newton que rara vez se menciona: los coeficientes w_i no dependen de los datos $f(x_i)$. Esto permite que podamos interpolar varias funciones con el mismo polinomio. Y mantiene, además, la ventaja de no necesitar ordenar los datos, como sí requiere el método de Newton.

Pero todavía no hemos terminado. Supongamos ahora que interpolamos la constante 1 con el polinomio hallado. En ese caso tenemos

$$1 = \sum_{i=0}^n 1 \cdot L_{n,i}(x) = L(x) \sum_{i=0}^n \frac{w_i}{x - x_i}, \quad (4.42)$$

²De todos modos, se requieren $O(n^2)$ operaciones para obtener los coeficientes $f(x_k, x_{k+1}, \dots, x_n)$.

pues hemos visto que $L_{n,i}(x) = 1$ cuando $x = x_i$.

Si dividimos $P_n(x)$ por la expresión anterior, o sea, que la dividimos por 1, nos queda:

$$P_n(x) = \frac{L(x) \sum_{i=0}^n f(x_i) \frac{w_i}{x - x_i}}{L(x) \sum_{i=0}^n \frac{w_i}{x - x_i}}, \quad (4.43)$$

y simplificando $L(x)$, obtenemos que

$$P_n(x) = \frac{\sum_{i=0}^n f(x_i) \frac{w_i}{x - x_i}}{\sum_{i=0}^n \frac{w_i}{x - x_i}}, \quad (4.44)$$

que se denomina *Interpolación Baricéntrica de Lagrange*. Al igual que en el caso del método mejorado, sólo se necesitan $O(n)$ operaciones para actualizar el polinomio si agregamos un par de datos $x_{n+1}, f(x_{n+1})$ adicionales.

En general, la *Interpolación Baricéntrica de Lagrange* es más estable numéricamente que el *Método de Lagrange* tradicional y que el *Método de Newton*, según el análisis hecho por N. J. Higham en [12].

4.5. Fenómeno de Runge

Supongamos que debemos interpolar los datos que se muestran en la tabla 4.3. Al aplicar

Tabla 4.3: Conjunto de datos a interpolar

i	x_i	y_i
0	0,000	0,500
1	1,000	0,933
2	2,000	0,067
3	3,000	0,500
4	4,000	0,933
5	5,000	0,067
6	6,000	0,500
7	7,000	0,933
8	8,000	0,067
9	9,000	0,500
10	10,000	0,933

el *Método de Lagrange* tradicional para obtener un polinomio interpolante, el resultado es un polinomio de grado 10 ($n = 10$). Las figuras 4.3, 4.4 y 4.5 muestran el proceso y los resultados de interpolar un conjunto de datos distribuidos uniformemente.

Como podemos ver, en la figura 4.3, los puntos del conjunto de datos están distribuidos en forma uniforme, tal como vemos en la tabla 4.3.

Al aplicar una interpolación tradicional, por ejemplo, la *Interpolación Baricéntrica de Lagrange*, obtenemos la curva de la figura 4.4.

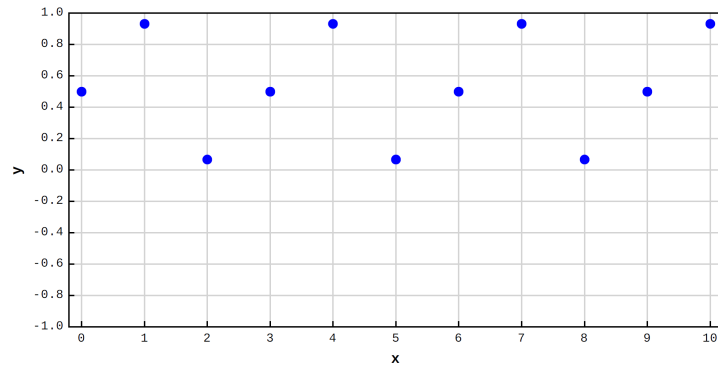


Figura 4.3: *Conjunto de puntos distribuidos uniformemente.*

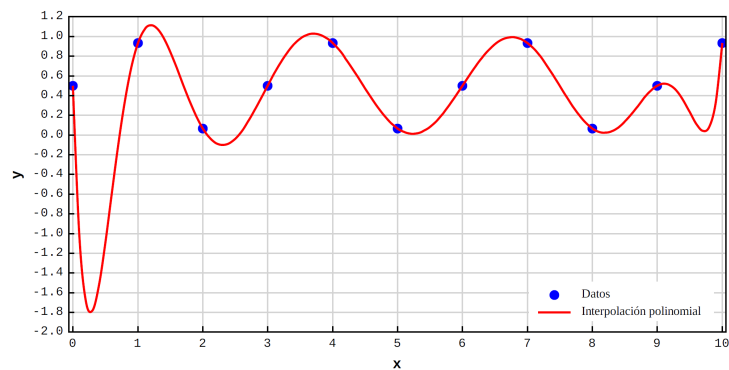


Figura 4.4: *Curva obtenida por interpolación por Lagrange.*

Como primera apreciación, la curva obtenida muestra una forma en los extremos que no parece corresponderse con los datos. Es más evidente en el extremo izquierdo que en el derecho. En cambio, en la zona central, la interpolación polinómica parece corresponderse mejor con esos datos. Podríamos dibujar una aproximación más «intuitiva», como vemos en la figura 4.5 con color azul.

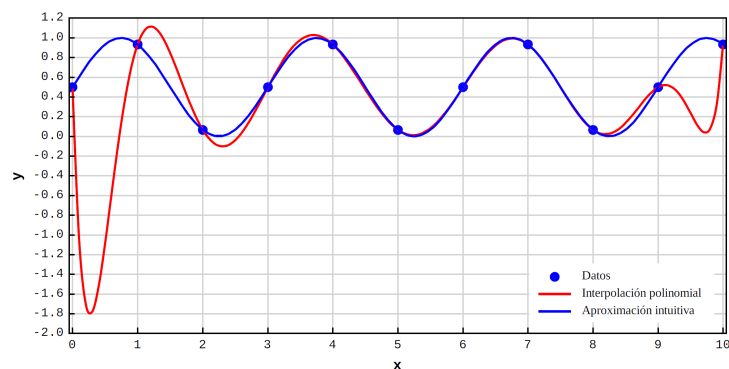


Figura 4.5: *Interpolación gráfica «intuitiva».*

La diferencia en los extremos es notable. Entre x_0 y x_1 , la interpolación polinómica aproxima valores negativos, en tanto que la interpolación gráfica «intuitiva», valores positivos. Podemos decir que la interpolación polinómica no resulta muy confiable para estimar o aproximar valores en los extremos, con lo cual pierde efectividad. Estas oscilaciones que aparecen en los polinomios de mayor grado cuando los datos están *distribuidos uniformemente* se conoce como *Fenómeno de Runge*. Veamos una forma de utilizar polinomios pero que evitarlo.

4.6. Interpolación por Trazadores cúbicos o «splines»

Supongamos que en lugar de proponer interpolar los datos de la tabla 4.1 mediante un solo polinomio que pase por todos los puntos, lo hagamos mediante segmentos de curvas, en este caso con polinomios de tercer grado, denominados *Trazadores cúbicos*. Definamos las curvas de la siguiente forma:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3, \quad (4.45)$$

con $i = 0; 1; \dots; n - 1$. Observemos que tenemos cuatro constantes para cada polinomio pero disponemos solamente dos datos en el tramo considerado. Debemos agregar condiciones para poder armar nuestra curva interpolante. Al no disponemos de más datos, vamos a imponer que las curvas cumplan con estas condiciones:

1. $S_i(x_i) = f(x_i)$ para cada $i = 0; 1; \dots; n$;
2. $S_{i+1}(x_{i+1}) = S_i(x_{i+1})$ para cada $i = 0; 1; \dots; n - 2$;
3. $S'_{i+1}(x_{i+1}) = S'_i(x_{i+1})$ para cada $i = 0; 1; \dots; n - 2$;
4. $S''_{i+1}(x_{i+1}) = S''_i(x_{i+1})$ para cada $i = 0; 1; \dots; n - 2$;
5. Alguna de las siguiente condiciones de borde:

- a) $S''_0(x_0) = S''_{n-1}(x_n) = S''_n(x_n) = 0$ (frontera libre);
- b) $S'_0(x_0) = f'(x_0) = \alpha$ y $S'_{n-1}(x_n) = S'_n(x_n) = f'(x_n) = \beta$ (frontera sujeta).

La primera condición nos asegura que las curvas pasen por los datos, en tanto que las tres condiciones siguientes aseguran la continuidad del conjunto de curvas tanto para las funciones $S_i(x)$ como para sus derivadas primera y segunda.

Para obtener cada polinomio, empecemos por plantear las condiciones definidas arriba. En primer lugar, como $S_i(x_i) = f(x_i)$, tendremos que:

$$S_i(x_i) = a_i = f(x_i). \quad (4.46)$$

Al aplicar la segunda condición tenemos que:

$$a_{i+1} = S_{i+1}(x_{i+1}) = S_i(x_{i+1}) = a_i + b_i(x_{i+1} - x_i) + c_i(x_{i+1} - x_i)^2 + d_i(x_{i+1} - x_i)^3, \quad (4.47)$$

para cada $i = 0; 1; \dots; n - 2$. Para simplificar la notación definamos que $h_i = (x_{i+1} - x_i)$, y que $a_n = f(x_n)$. Entonces nos queda que

$$a_{i+1} = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3, \quad (4.48)$$

es válida para cada $i = 0; 1; \dots; n - 1$.

En forma análoga tenemos que

$$S'_i(x_i) = b_i, \quad (4.49)$$

por lo tanto, también se cumple que

$$b_{i+1} = b_i + 2c_i h_i + 3d_i h_i^2, \quad (4.50)$$

es válida para cada $i = 0; 1; \dots; n - 1$.

Finalmente, tenemos que

$$S_i''(x_i) = 2c_i. \quad (4.51)$$

Como se cumple que $c_n = S_n''(x_n)/2$, nos queda que:

$$c_{i+1} = c_i + 3d_i h_i, \quad (4.52)$$

una vez más, para cada $i = 0; 1; \dots; n - 1$. Si despejamos d_i y reemplazamos en las dos expresiones anteriores, nos queda:

$$a_{i+1} = a_i + b_i h_i + \frac{h_i^2}{3}(2c_i + c_{i+1}), \quad (4.53)$$

$$b_{i+1} = b_i + h_i(c_i + c_{i+1}), \quad (4.54)$$

para cada $i = 0; 1; \dots; n - 1$.

En la primera ecuación podemos despejar b_i , que resulta ser

$$b_i = \frac{a_{i+1} - a_i}{h_i} - \frac{h_i}{3}(2c_i + c_{i+1}). \quad (4.55)$$

Si usamos 4.54 para obtener b_i en vez de b_{i+1} y utilizamos 4.55 para obtener b_{i-1} , nos queda

$$\frac{a_{i+1} - a_i}{h_i} - \frac{h_i}{3}(2c_i + c_{i+1}) = \frac{a_i - a_{i-1}}{h_{i-1}} - \frac{h_{i-1}}{3}(2c_{i-1} + c_i) + h_{i-1}(c_{i-1} + c_i) \quad (4.56)$$

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_i c_{i+1} = \frac{3}{h_i}(a_{i+1} - a_i) - \frac{3}{h_{i-1}}(a_i - a_{i-1}), \quad (4.57)$$

para cada $i = 1; 2; \dots; n - 1$.

Ahora nos falta determinar si con este esquema podemos obtener un resultado único para los valores de c_i . El siguiente teorema nos lo asegura:

Teorema 4.3. Sea f en $a = x_0 < x_1 < \dots < x_n = b$, entonces f tendrá un interpolante único de frontera libre o natural en los nodos $x_0; x_1; \dots; x_n$.

Demostración Si la curva es de frontera libre o natural, entonces se cumple que $S_0''(a) = 0$ y $S_{n-1}''(b) = S_n''(b) = 0$, por lo tanto tendremos que

$$c_n = \frac{S_n''(x_n)}{2} = 0;$$

y que

$$0 = S_0''(x_0) = 2c_0 + 6d_0(x_0 - x_0) \Rightarrow c_0 = 0.$$

En consecuencia, nos queda un sistema de ecuaciones de la forma $Ax = B$ con

$$A = \begin{bmatrix} 1 & 0 & 0 & \dots & \dots & 0 \\ h_0 & 2(h_0 + h_1) & h_1 & \ddots & \ddots & \vdots \\ 0 & h_1 & 2(h_1 + h_2) & h_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} \\ 0 & \dots & \dots & 0 & 0 & 1 \end{bmatrix},$$

$$B = \begin{bmatrix} 3 \left[\frac{a_1 - a_0}{h_0} - f'(a) \right] \\ \frac{3}{h_1} (a_2 - a_1) - \frac{3}{h_0} (a_1 - a_0) \\ \vdots \\ \frac{3}{h_{n-1}} (a_n - a_{n-1}) - \frac{3}{h_{n-2}} (a_{n-1} - a_{n-2}) \\ 3 \left[f'(b) - \frac{a_n - a_{n-1}}{h_{n-1}} \right] \end{bmatrix}, \text{ y } x = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{bmatrix}.$$

Como en el caso anterior, el sistema de ecuaciones lineales tiene solución única, es decir, existe un único vector c_0, c_1, \dots, c_n , y consecuentemente, un sólo conjunto de curvas $S_i(x)$.

En cuanto al error que cometemos al interpolar una curva utilizando *Trazadores cúbicos*, para el caso con frontera libre podemos expresarlo como

$$\max_{a \leq x \leq b} |f(x) - S(x)| \leq \frac{5}{384} M \max_{0 \leq i \leq n-1} |h_i|^4, \quad (4.58)$$

donde $S(x)$ es el conjunto de las $S_i(x)$ curvas, $M = f^{iv}(\xi)$ con $\xi \in [x_0; x_n]$ y $h_i = x_{i+1} - x_i$. Sin embargo, cuando se utiliza este caso, el orden del error en los extremos es proporcional a $|h_i|^2$ y no a $|h_i|^4$, por lo que no siempre es bueno aplicar el caso de frontera libre o natural.

Finalmente, existe un tercer caso cuando no conocemos las derivadas extremas ($f'(a)$ y $f'(b)$), denominado *aproximación sin un nodo*³, en el cual se considera que $d_0 = d_1$ y $d_{n-2} = d_{n-1}$, que es lo mismo que considerar que $S_0(x) = S_1(x)$ y $S_{n-2}(x) = S_{n-1}(x)$, lo cual también introduce un error en los extremos del orden de $|h_i|^2$.

Para este último caso tenemos lo siguiente:

$$c_1 = c_0 + 3d_0h_0 \quad \Rightarrow \quad d_0 = \frac{c_1 - c_0}{3h_0} \quad (4.59)$$

$$c_2 = c_1 + 3d_1h_1 \quad \Rightarrow \quad d_1 = \frac{c_2 - c_1}{3h_1}. \quad (4.60)$$

Como $d_0 = d_1$, entonces

$$\frac{c_1 - c_0}{3h_0} = \frac{c_2 - c_1}{3h_1} \quad (4.61)$$

$$h_1c_1 - h_1c_0 = h_0c_2 - h_0c_1, \quad (4.62)$$

lo que nos deja la siguiente expresión para la primera fila del sistema:

$$h_1c_0 - (h_0 + h_1)c_1 + h_0c_2 = 0. \quad (4.63)$$

Análogamente, para d_{n-2} y d_{n-1} tenemos algo similar:

$$c_{n-1} = c_{n-2} + 3d_{n-2}h_{n-2} \Rightarrow d_{n-2} = \frac{c_{n-1} - c_{n-2}}{3h_{n-2}} \quad (4.64)$$

$$c_n = c_{n-1} + 3d_{n-1}h_{n-1} \Rightarrow d_{n-1} = \frac{c_n - c_{n-1}}{3h_{n-1}}, \quad (4.65)$$

con las cuales obtenemos la última fila del sistema:

$$h_{n-1}c_{n-2} - (h_{n-2} + h_{n-1})c_{n-1} + h_{n-2}c_n = 0. \quad (4.66)$$

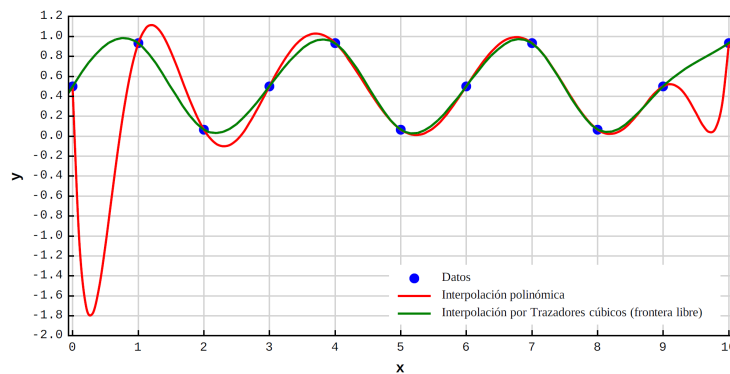
³Algunos textos denominan a esta aproximación como condición *no un nodo*, por la expresión en inglés *not a knot approximation*.

Así, el sistema queda como:

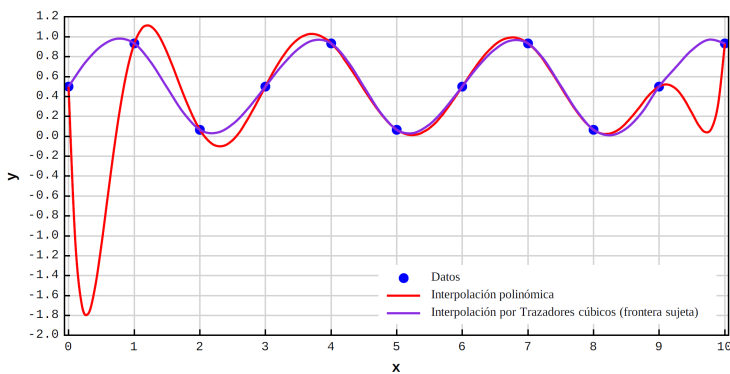
$$A = \begin{bmatrix} h_1 & -(h_0 + h_1) & h_0 & \dots & \dots & 0 \\ h_0 & 2(h_0 + h_1) & h_1 & \ddots & \ddots & \vdots \\ 0 & h_1 & 2(h_1 + h_2) & h_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} \\ 0 & \dots & \dots & h_{n-1} & -(h_{n-2} + h_{n-1}) & h_{n-2} \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ \frac{3}{h_1}(a_2 - a_1) - \frac{3}{h_0}(a_1 - a_0) \\ \vdots \\ \vdots \\ \frac{3}{h_{n-1}}(a_n - a_{n-1}) - \frac{3}{h_{n-2}}(a_{n-1} - a_{n-2}) \\ 0 \end{bmatrix}, \text{ y } x = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ \vdots \\ c_n \end{bmatrix}.$$

Esta variante de la interpolación mediante *Trazadores cúbicos* es poco usada porque tiene muchas más indefiniciones que la de frontera libre o natural.



(a) Frontera libre



(b) Frontera sujeta

Figura 4.6: Interpolación por Trazadores cúbicos o «spline».

Para verificar que la interpolación por *Trazadores cúbicos* arroja mejores resultados que la interpolación polinomial para datos uniformemente distribuidos, veamos las curvas que hemos

obtenido aplicando los dos casos, frontera libre y frontera sujeta, asumiendo que conocemos las derivadas primeras en los puntos extremos para el caso de frontera sujeta, para el conjunto de valores de la tabla 4.3.

En la figura 4.6(a) vemos que la aproximación por *Trazadores cúbicos* con frontera libre es bastante buena con excepción del último segmento, que no se parece al «intuitivo». En cambio, en la figura 4.6(b), la aproximación completa es mucho mejor y es prácticamente igual a la hecha en forma «intuitiva». Pero al compararla con la aproximación polinomial tradicional, la diferencia de las dos es casi despreciable respecto de las oscilaciones de la interpolación polinomial tradicional. Por eso los *Trazadores cúbicos* suelen ser muy eficiente como procedimiento de interpolación.

Nos queda pendiente qué hacer cuando no conocemos las derivadas primeras en los puntos extremos. Veremos en el capítulo 6 como aproximar esas derivadas.

4.7. Interpolación o Método de Hermite

Hay casos en los que disponemos de más datos para interpolar o generar una función que represente esos datos. Por ejemplo, supongamos que para una partícula que se desplaza conocemos los siguientes datos: el instante t_i , la coordenada de la trayectoria y_i y la velocidad v_i , para $i = 0; 1; \dots; n$. En este caso además de los valores de $f(t_i)$ conocemos también los de $f'(t_i)$ pues $v_i = f'(t_i)$. Por lo tanto nuestra tabla original podría ser reescrita como (tabla 4.4):

Tabla 4.4: *Datos incluyendo la primera derivada*

t	y	v
t_0	y_0	v_0
t_1	y_1	v_1
t_2	y_2	v_2
t_3	y_3	v_3

Es evidente que contamos con más información para construir nuestro polinomio interpolante. En efecto, de disponer de sólo cuatro valores asociados a nuestros puntos (en este caso, el instante t_i), pasamos a tener ocho valores. Si queremos utilizar todos los datos disponibles, en lugar de interpolar con una curva de tercer grado, podemos usar ahora un polinomio de grado siete (7), pues tiene ocho coeficientes, a saber:

$$y(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 + a_6 t^6 + a_7 t^7, \quad (4.67)$$

y del cual podemos hallar la primera derivada:

$$v(t) = y'(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4 + 6a_6 t^5 + 7a_7 t^6. \quad (4.68)$$

Al igual que al principio, podemos reemplazar cada uno de los valores en las dos funciones, con lo cual obtendremos un sistema de ocho ecuaciones con ocho incógnitas, sistema que puede resolverse sin problemas. Cuando conocemos el valor de la función en el punto como así también su derivada, la interpolación se denomina *Interpolación o Método de Hermite*. El siguiente teorema lo define.

Teorema 4.5. Sea $f \in C^1[a; b]$ y sean $x_0; x_1; \dots; x_n \in [a; b]$ distintos, el polinomio único de menor grado que concuerda con f y f' en $x_0; x_1; \dots; x_n$ es el polinomio de Hermite de grado a lo sumo $2n + 1$, que está dado por la siguiente expresión:

$$H_{2n+1}(x) = \sum_{i=0}^n f(x_i) H_{n;i}(x) + \sum_{i=0}^n f'(x_i) \hat{H}_{n;i}(x),$$

donde

$$H_{n;i}(x) = [1 - 2(x - x_i)L'_{n;i}(x_i)]L_{n;i}^2(x),$$

y

$$\hat{H}_{n;i}(x) = (x - x_i)L_{n;i}^2(x),$$

donde $L_{n;i}(x)$ es el i -ésimo polinomio de Lagrange de grado n . Si además $f \in C^{2n+2}[a; b]$, entonces se cumple que

$$f(x) = H_{2n+1}(x) + \frac{(x - x_0)^2 \cdots (x - x_n)^2}{(2n + 2)!} f^{(2n+2)}(\xi),$$

con ξ tal que $a < \xi < b$.

Demostración Primero, recordemos que

$$L_{n;i}(x) = \begin{cases} 1 & \text{si } x = x_i, \\ 0 & \text{si } x = x_j \text{ con } j \neq i, \end{cases}$$

por lo tanto, tenemos que:

$$H_{n,i}(x_j) = 0 \wedge \hat{H}_{n,i}(x_j) = 0,$$

para $j \neq i$, en tanto que

$$H_{n,i}(x_i) = [1 - 2(x_i - x_i)L'_{n;i}(x_i)]L_{n;i}^2(x_i) = [1 - 2(0)L'_{n;i}(x_i)] \cdot 1^2 = 1,$$

y

$$\hat{H}_{n,i}(x_i) = (x_i - x_i)L_{n;i}^2(x_i) = (x_i - x_i) \cdot 1^2 = 0.$$

Entonces, nos queda que:

$$H_{2n+1}(x_i) = \sum_{i=0}^n f(x_i)H_{n;i}(x_i) + \sum_{i=0}^n f'(x_i)\hat{H}_{n;i}(x_i) = f(x_i) + \sum_{i=0}^n f'(x_i) \cdot 0 = f(x_i),$$

para $i = 0; 1; 2; \dots; x_n$, es decir $H_{2n+1}(x) = f(x)$ en los puntos dados.

Demostremos ahora que $H'_{2n+1}(x) = f'(x)$. Como $L_{n;i}(x)$ es un factor de $H'_{n;i}(x)$, entonces se cumple que $H'_{n;i}(x_j) = 0$ cuando $j \neq i$. Si $j = i$, tenemos que

$$\begin{aligned} H'_{n;j}(x_j) &= -2 \cdot L_{n;j}^2(x_j) + [1 + 2(x_j - x_j)L'_{n;j}(x_j)]2L_{n;j}(x_j)L'_{n;j}(x_j) \\ &= -2 \cdot L_{n;j}^2(x_j) + 2 \cdot L_{n;j}^2(x_j) = 0, \end{aligned}$$

o sea, $H'_{n;i}(x_j) = 0$ para todas la j e i .

Por otro lado, observemos que

$$\begin{aligned} \hat{H}'_{n;i}(x_j) &= L_{n;i}^2(x_j) + (x_j - x_i)2L_{n;i}(x_j)L'_{n;i}(x_j) \\ &= L_{n;i}(x_j)[L_{n;i}(x_j) + 2(x_j - x_i)L'_{n;i}(x_j)], \end{aligned}$$

y en consecuencia, cuando $j \neq i$ tendremos que:

$$\hat{H}'_{n;i}(x_j) = L_{n;i}^2(x_j) + (x_j - x_i)2L_{n;i}(x_j)L'_{n;i}(x_j) = 0 + 0 = 0,$$

pues $L_{n;i}(x_j) = 0$, y cuando $j = i$

$$\hat{H}'_{n;j}(x_j) = L_{n;j}^2(x_j) + (x_j - x_j)2L_{n;j}(x_j)L'_{n;j}(x_j) = 1^2 + 0 = 1.$$

Si combinamos ambos casos tenemos

$$\begin{aligned} H'_{2n+1}(x_j) &= \sum_{i=0}^n f(x_j) H'_{n;i}(x_j) + \sum_{i=0}^n f'(x_j) \hat{H}'_{n;i}(x_j) \\ &= \sum_{i=0}^n f(x_j) \cdot 0 + f'(x_j) \cdot 1 = 0 + f'(x_j) = f'(x_j), \end{aligned}$$

entonces $H_{2n+1}(x) = f(x)$ y $H'_{2n+1}(x) = f'(x)$ para x_0, x_1, \dots, x_n .

En realidad, la *Interpolación o Método de Hermite* es un caso particular de los denominados *polinomios osculantes*, cuando $m_i = 1$. Veamos la siguiente definición.

Definición 4.1. Dados x_0, x_1, \dots, x_n , todos distintos y los enteros no negativos m_0, m_1, \dots, m_n , se denomina *polinomio osculante* que aproxima una función $f \in C^m[a, b]$ donde se cumple que $m = \max\{m_0, m_1, \dots, m_n\}$ y $x_i \in [a, b]$ para cada $i = 0; 1; \dots; n$, al polinomio de menor grado que concuerda con la función f y con todas sus derivadas de orden menor o igual m_i en x_i para cada $i = 0; 1; \dots; n$. El máximo grado de este polinomio es

$$M = \sum_{i=0}^n m_i + n,$$

pues el número de condiciones que debe cumplir es

$$\sum_{i=0}^n (m_i + 1) = \sum_{i=0}^n m_i + (n + 1),$$

y un polinomio de grado M tiene $M + 1$ coeficientes.

Esto quiere decir que además de las derivadas primeras podemos tener las derivadas segundas, terceras, etc., para armar el polinomio interpolante. Con esos datos (inclusive puede ocurrir que contemos con datos parciales de las derivadas), el procedimiento visto para la interpolación de Hermite se puede ampliar para obtener curvas que tengan segundas o terceras derivadas, si bien esto no resulta tan sencillo de implementar. (Para más detalles, ver [3].)

Volvamos al *Método de Hermite*. Como está basado en los polinomios del *Método de Lagrange*, si se agregan datos, el método tiene las mismas desventajas que el de Lagrange, porque deben repetirse todos los cálculos para obtener el nuevo polinomio interpolante.

Pero tal como vimos para ese método, existe también una forma alternativa de armar el polinomio buscado aplicando el *Método de Newton*, que nos permite desarrollarlo con la siguiente fórmula:

$$P_n(x) = f(x_0) + \sum_k^n f(x_0; x_1; \dots; x_k) \prod_{j=0}^{k-1} (x - x_j). \quad (4.69)$$

Dado que conocemos los valores de la derivada primera, debemos redefinir nuestra sucesión de datos. Por ejemplo, si tomamos los datos de la tabla 4.4, nuestra nueva sucesión de puntos es $t_0; t_0; t_1; t_1; t_2; t_2; t_3; t_3$, es decir, definimos una nueva sucesión $z_0; z_1; \dots; z_{2n+1}$ tal que

$$z_{2i} = z_{2i+1} = t_i, \quad (4.70)$$

con $i = 0; 1; 2; \dots; n$. Como con esta nueva sucesión no podemos definir $f(z_{2i}, z_{2i+1})$ de la forma vista anteriormente, resulta conveniente definirla aprovechando que conocemos $f'(z_{2i}) = f'(x_i)$, con lo que aprovechamos los datos conocidos. En consecuencia, podemos construir la tabla 4.5 con los coeficientes para armar el polinomio según el Método de Newton.

Construida nuestra tabla, el polinomio de Hermite lo obtenemos de la siguiente manera:

$$H_{2n+1}(x) = f(z_0) + \sum_{k=1}^{2n+1} \left[f(z_0, z_1, \dots, z_k) \prod_{j=0}^{k-1} (x - z_j) \right]. \quad (4.71)$$

Tabla 4.5: Interpolación de Hermite aplicando el Método de Newton

\mathbf{z}	$\mathbf{f}(\mathbf{z})$	$\mathbf{f}(\mathbf{z}_i, \mathbf{z}_{i+1})$	$\mathbf{f}(\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{z}_{i+2})$	$\mathbf{f}(\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{z}_{i+2}, \mathbf{z}_{i+3})$
$z_0 = x_0$	$f(z_0) = f(x_0)$			
$z_1 = x_0$	$f(z_1) = f(x_0)$	$f(z_0, z_1) = f'(x_0)$	$f(z_0, z_1, z_2)$	
$z_2 = x_1$	$f(z_2) = f(x_1)$	$f(z_1, z_2)$	$f(z_1, z_2, z_3)$	$f(z_0, z_1, z_2, z_3)$
$z_3 = x_1$	$f(z_3) = f(x_1)$	$f(z_2, z_3) = f'(x_1)$	$f(z_2, z_3, z_4)$	$f(z_1, z_2, z_3, z_4)$
$z_4 = x_2$	$f(z_4) = f(x_2)$	$f(z_3, z_4)$	$f(z_3, z_4, z_5)$	$f(z_2, z_3, z_4, z_5)$
$z_5 = x_2$	$f(z_5) = f(x_2)$	$f(z_4, z_5) = f'(x_2)$	$f(z_4, z_5, z_6)$	$f(z_3, z_4, z_5, z_6)$
$z_6 = x_3$	$f(z_6) = f(x_3)$	$f(z_5, z_6)$	$f(z_5, z_6, z_7)$	$f(z_4, z_5, z_6, z_7)$
$z_7 = x_3$	$f(z_7) = f(x_3)$	$f(z_6, z_7) = f'(x_3)$		

De manera similar a lo visto para el *Método de Lagrange*, el error al aplicar el *Método de Hermite* es función de los puntos usados para dicha interpolación, que se expresa así:

$$E(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!} \prod_{i=0}^n (x - x_i)^2. \quad (4.72)$$

Si aplicamos esto a nuestros datos originales de la tabla 4.4, obtendríamos un polinomio de grado 7. En el caso de muchos puntos, el grado del polinomio interpolante crece muy rápidamente, lo que complica su obtención. Es por eso que el *Método de Hermite* no suele usarse de esta forma, sino como parte de una interpolación por segmentos de curva, similar al caso de los *Trazadores cúbicos*. Así, para cada intervalo entre puntos sucesivos tenemos cuatro datos que podemos utilizar para interpolar valores. Veamos como aplicarlo a nuestra tabla 4.4.

Para armar la curva que interpola entre t_0 y t_1 , contamos con los valores de y_0 , y_1 , v_0 y v_1 , con lo cual podemos armar un polinomio de Hermite de tercer grado que cumpla con las condiciones $H_3(t_0) = f(t_0) = y_0$; $H_3(t_1) = f(t_1) = y_1$, $H'_3(t_0) = f'(t_0) = v_0$ y $H'_3(t_1) = f'(t_1) = v_1$. Lo mismo podemos hacer entre t_1 y t_2 , y para el intervalo t_2 y t_3 . Tendremos, entonces, cuatro polinomios de Hermite para todo el intervalo, a saber, $H_{1;0}(t)$, $H_{1;1}(t)$, $\hat{H}_{1;0}(t)$ y $\hat{H}_{1;1}(t)$. Los polinomios resultantes son:

$$H_{1;0}(t) = \left[1 - 2(t - t_0) \frac{1}{t_0 - t_1} \right] \left(\frac{t - t_1}{t_0 - t_1} \right)^2 \quad (4.73)$$

$$H_{1;1}(t) = \left[1 - 2(t - t_1) \frac{1}{t_1 - t_0} \right] \left(\frac{t - t_0}{t_1 - t_0} \right)^2 \quad (4.74)$$

$$\hat{H}_{1;0}(t) = (t - t_0) \left(\frac{t - t_1}{t_0 - t_1} \right)^2 \quad (4.75)$$

$$\hat{H}_{1;1}(t) = (t - t_1) \left(\frac{t - t_0}{t_1 - t_0} \right)^2 \quad (4.76)$$

Como además se cumple que $H_{3,i}(t_{i+1}) = H_{3,i+1}(t_{i+1})$ y $H'_{3,i}(t_{i+1}) = H'_{3,i+1}(t_{i+1})$, tenemos continuidad para la curva y su primera derivada. Podemos armar una curva con segmentos

de curvas de tercer grado, que puede representar a la función y a la primera derivada, sin tener que preocuparnos por los efectos negativos de las oscilaciones no deseadas en los extremos⁴.

También podemos armar este polinomio aplicando el *Método de Newton* adaptado, a partir de la siguiente tabla:

Tabla 4.6: *Interpolación Hermite segmentada aplicando el Método de Newton*

\mathbf{z}	$\mathbf{f}(\mathbf{z})$	$\mathbf{f}(\mathbf{z}_i, \mathbf{z}_{i+1})$	$\mathbf{f}(\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{z}_{i+2})$	$\mathbf{f}(\mathbf{z}_i, \mathbf{z}_{i+1}, \mathbf{z}_{i+2}, \mathbf{z}_{i+3})$
$z_0 = t_i$	$f(z_0) = f(t_i)$	$f(z_0, z_1) = f'(t_i)$	$f(z_0, z_1, z_2)$	$f(z_0, z_1, z_2, z_3)$
$z_1 = t_i$	$f(z_1) = f(t_i)$			
$z_2 = t_{i+1}$	$f(z_2) = f(t_{i+1})$	$f(z_1, z_2)$	$f(z_1, z_2, z_3)$	
$z_3 = t_{i+1}$	$f(z_3) = f(t_{i+1})$	$f(z_2, z_3) = f'(t_{i+1})$		

A partir de la tabla 4.6, el polinomio de Hermite segmentado lo armamos de la siguiente manera:

$$H_3(t) = f(z_0) + f(z_0, z_1)(t - z_0) + f(z_0, z_1, z_2)(t - z_0)(t - z_1) + f(z_0, z_1, z_2, z_3)(t - z_0)(t - z_1)(t - z_2), \tag{4.77}$$

que podemos escribir así:

$$H_3(t) = f(t_i) + f'(t_i)(t - t_i) + f(t_i, t_i, t_{i+1})(t - t_i)^2 + f(t_i, t_i, t_{i+1}, t_{i+1})(t - t_i)^2(t - t_{i+1}), \tag{4.78}$$

donde:

$$f(t_i, t_i, t_{i+1}) = \frac{f(t_i; t_{i+1}) - f'(t_i)}{t_{i+1} - t_i}, \tag{4.79}$$

$$f(t_i, t_{i+1}, t_{i+1}) = \frac{f'(t_{i+1}) - f(t_i, t_{i+1})}{t_{i+1} - t_i}, \tag{4.80}$$

y

$$f(t_i, t_i, t_{i+1}, t_{i+1}) = \frac{f(t_i, t_{i+1}, t_{i+1}) - f(t_i, t_i, t_{i+1})}{t_{i+1} - t_i}. \tag{4.81}$$

Al igual que para el caso de la interpolación completa, el error cometido en una interpolación segmentada es proporcional a la derivada $2n + 2$ de la función. En este caso, puesto que solo armamos una curva entre i e $i + 1$, el error en cada tramo está dado por la siguiente expresión:

$$E(h) = \frac{f^{(iv)}(\xi)}{384} \max(h_i^4),$$

con $\xi \in [x_0; x_n]$ y $h_i = x_{i+1} - x_i$.

De todos modos, como para poder armar este tipo de curvas debemos conocer los valores de las derivadas en cada punto, algo que no siempre está disponible, usar estos segmentos de curvas con polinomios de Hermite no siempre resultan ser una solución aplicable.

⁴Esta interpolación se usa en el *método de los elementos finitos* para generar las funciones de forma en los elementos de viga.

4.8. Interpolación por el método de Akima

A fines de los años 60 el uso de las computadoras empezó a generalizarse en muchos ámbitos dedicados a la investigación y desarrollo. En ese contexto, aún algunas tareas seguían haciéndose de manera manual. Una era la construcción de curvas que pasaran por un conjunto de datos, es decir, una forma de interpolar datos en forma manual sin ayuda de la computadora. Un investigador, H. Akima, probó interpolar conjuntos de curvas utilizando los métodos tradicionales de la época, entre ellos, la interpolación polinomial tradicional y los trazadores cúbicos. Sin embargo, los resultados no siempre le satisfacían, pues no coincidían con las curvas que los ingenieros o científicos dibujaban a mano. Esas diferencias generalmente consistían en oscilaciones entre puntos sucesivos que intuitivamente se veían claramente fuera de lugar.⁵

Según Akima, cuando alguien dibujaba una curva a mano, el forma surgía de considerar solamente la información local y no el conjunto completo de datos. Así, tanto la interpolación polinómica tradicional como los trazadores cúbicos usaban el conjunto completo de datos para obtener los coeficientes del o los polinomios. Esto hacía que las curvas no representaran correctamente a los datos.

Se concentró en investigar un método que se pareciera a los *Trazadores cúbicos* pero que solamente usara datos locales. Encontró que un tipo de *Trazadores cúbicos* podría ser la interpolación por Hermite segmentada. Pero su idea era un método para interpolar conjuntos de datos que no incluyeran la primera derivada. Para salvar esta falta, Akima propuso aproximar esas derivadas. Una forma usual de hacer esto era mediante una aproximación de la pendiente de cualquier punto mediante dos puntos adicionales, uno a cada lado del punto en análisis. Como los resultados no eran muy satisfactorios, Akima propuso aproximar la primera derivada de cualquier punto (salvo los extremos) con ayuda de cuatro puntos adicionales, dos a cada lado.

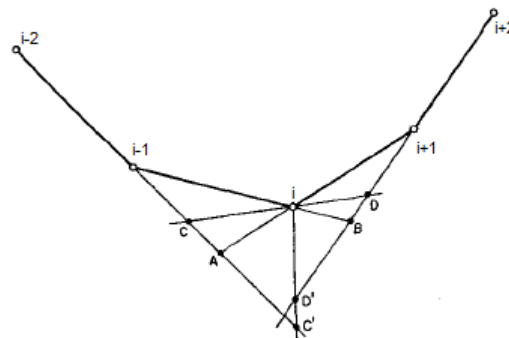


Figura 4.7: Interpretación geométrica de la aproximación de la pendiente.

Esta aproximación la hizo mediante un razonamiento geométrico (como puede verse en la figura 4.7 adaptada de [1], donde el segmento \overline{CD} es la pendiente aproximada del punto i), lo que le permitió generalizar esta aproximación para cualquier caso y para cualquier sistema de coordenadas. Como aproximación de las derivadas primeras de todos los puntos intermedios propuso la siguiente expresión:

$$t_i = \frac{|m_{i+1} - m_i| m_{i-1} + |m_{i-1} - m_{i-2}| m_i}{|m_{i+1} - m_i| + |m_{i-1} - m_{i-2}|},$$

donde m_{i-2} , m_{i-1} , m_i y m_{i+1} son las pendientes de los segmentos que unen los puntos (x_{i-2}, y_{i-2}) y (x_{i-1}, y_{i-1}) , (x_{i-1}, y_{i-1}) y (x_i, y_i) , (x_i, y_i) y (x_{i+1}, y_{i+1}) , y (x_{i+1}, y_{i+1}) y (x_{i+2}, y_{i+2}) , respectivamente. Con esta definición de la pendiente del punto (x_i, y_i) obtenemos los siguientes casos:

1. Cuando $m_{i-2} = m_{i-1}$ y $m_i \neq m_{i+1}$, entonces $t_i = m_{i-1}$;

⁵El programa SMath Studio incluye el método de Akima como uno de los tres métodos de interpolación que dispone. Los otros dos son la interpolación lineal segmentada y los *Trazadores cúbicos* o «splines».

2. Cuando $m_{i-2} \neq m_{i-1}$ y $m_i = m_{i+1}$, entonces $t_i = m_i$;
3. Cuando $m_{i-1} = m_i$, entonces $t_i = m_{i-1} = m_i$.

Existe un caso que no puede ser resuelto en forma directa: aquel en el que $m_{i-2} = m_{i-1} \neq m_i = m_{i+1}$, pues la pendiente queda indeterminada. Esto fue resuelto por Akima proponiendo que

$$t_i = \frac{m_{i-1} + m_i}{2},$$

al que consideró como cuarto caso.

4.9. Notas finales

Hemos visto diferentes métodos para interpolar valores a partir de datos discretos usando funciones polinómicas completas, como son los métodos de *Lagrange*, de *Newton* y de *Hermite*, y también la interpolación mediante segmentos de curvas, como es el caso del método de *Hermite fragmentado*, el de los *Trazadores cúbicos* y el de *Akima*. Dentro de este último conjunto está también el método de interpolación lineal por segmentos, cuyas funciones se obtienen utilizando el *Método de Lagrange* tradicional entre dos puntos. Así, los dos polinomios de Lagrange necesarios son:

$$L_{1;0}(x) = \frac{x - x_1}{x_0 - x_1}$$

$$L_{1;1}(x) = \frac{x - x_0}{x_1 - x_0},$$

donde x_0 es el punto inicial y x_1 el punto final de la interpolación. Con estos dos polinomios, el polinomio completo de Lagrange resulta ser:

$$\begin{aligned} P_1(x) &= f(x_0)L_{1;0}(x) + f(x_1)L_{1;1}(x) \\ &= f(x_0)\frac{x - x_1}{x_0 - x_1} + f(x_1)\frac{x - x_0}{x_1 - x_0} = f(x_1)\frac{x - x_0}{x_1 - x_0} - f(x_0)\frac{x - x_1}{x_1 - x_0} \\ &= \frac{f(x_1) - f(x_0)}{x_1 - x_0}x - \frac{f(x_1)x_0 - f(x_0)x_1}{x_1 - x_0}. \end{aligned}$$

Además de los métodos vistos, existen otros más complejos que mejoran nuestra aproximación de valores intermedios, pero que suelen ser también más difíciles de implementar y con mayor costo computacional. En particular, para ciertas curvas que no pueden ser definidas mediante polinomios contamos con curvas paramétricas denominadas *curvas de Bezier*. (Para más datos, véase [3].)

Otro caso es el uso cada vez más generalizado de la *Transformada Rápida de Fourier* como procedimiento para interpolar datos usando funciones trigonométricas.

Respecto a la *Interpolación Baricéntrica de Lagrange*, Berrut y Trefethen (véase [20]) señalan en su artículo que resulta curioso que el método no figure en ningún libro de texto de análisis numérico como alternativa al método tradicional, teniendo en cuenta la simplicidad del mismo para ser implementado en una computadora.

Por último, resulta interesante observar que las hojas de cálculo y el diseño asistido por computadora (CAD por sus siglas en inglés) hace un uso intensivo de la interpolación fragmentada (o segmentada), con las «spline», y las curvas paramétricas (curvas de Bezier). Las primeras son usadas en programas como el AutoCAD[®], en tanto que las segundas, en programas como el Corel Draw[®], LibreOffice Draw o similares. Por tal motivo, resulta útil conocer los fundamentos matemáticos de cada una de ellas.

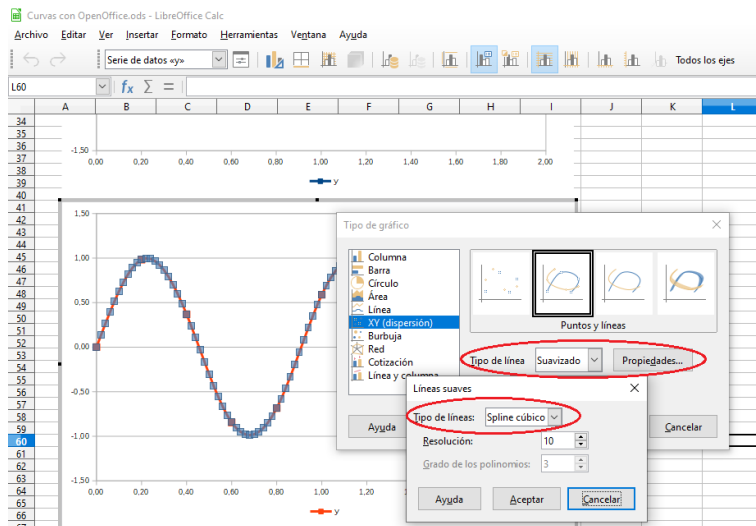


Figura 4.8: Cuadro de diálogo del LibreOffice Calc para «Tipo de línea: Suavizado».

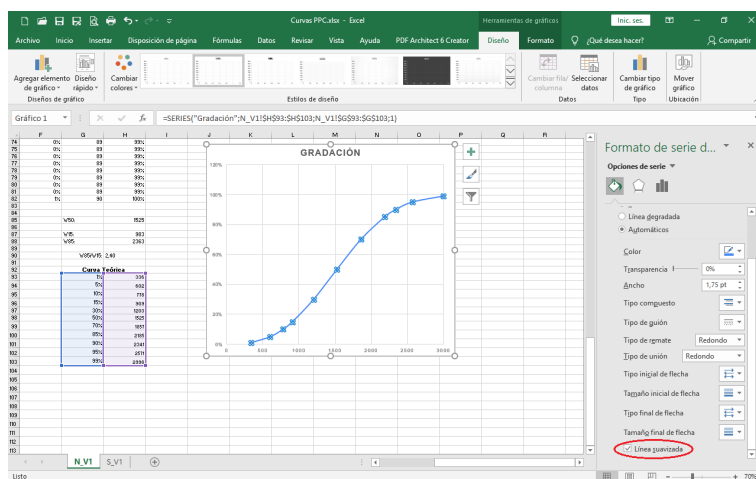


Figura 4.9: Cuadro de diálogo del MS Excel 2016 para «Línea suavizada».

Un caso interesante en el uso de interpolación con segmentos de curvas es la versión 6.2 del programa LibreOffice Calc ⁶, la planilla de cálculo del paquete de distribución gratuita LibreOffice®. Para graficar curvas suavizadas aplica la interpolación mediante trazadores cúbicos o «splines», si se quiere que las curvas pasen por los puntos, o mediante «B-splines» si lo que se quiere es que las curvas tengan una forma determinada. Ambas opciones están disponibles en el cuadro de control de la curva, como se puede ver en la figura 4.8. No ocurre lo mismo en el MS Excel® (figura 4.9), al menos hasta la versión 2016, pues la graficación de la curva suavizada es hecha mediante un algoritmo no especificado, ni es posible ajustar la densidad de puntos adicionales para dicha representación gráfica.

Ejercicios

Métodos de Lagrange tradicional y baricéntrico

1. Mediante la aplicación del *Método de Lagrange tradicional*:

⁶Este paquete de oficina gratuito se puede bajar en: <http://es.libreoffice.org/>.

a) Obtenga $f(8,1)$, $f(8,4)$ y $f(8,5)$ a partir de los siguientes datos:

x	8,0	8,3	8,6	8,7
$f(x)$	16,94410	17,56492	18,50515	18,82091

b) Obtenga $f(-0,6)$, $f(-0,4)$ y $f(-0,333333)$ a partir de los siguientes datos:

x	-0,75	-0,5	-0,25	0
$f(x)$	-0,07181250	-0,02475000	0,33493750	1,10100000

c) Obtenga $f(0,15)$, $f(0,25)$ y $f(0,35)$ a partir de los siguientes datos:

x	0,1	0,2	0,3	0,4
$f(x)$	-0,62049958	-0,28398668	0,00660095	0,24842440

d) Obtenga $f(1,03)$, $f(1,09)$ y $f(1,12)$ a partir de los siguientes datos:

x	1,00	1,05	1,10	1,15
$f(x)$	0,1924	0,2414	0,2933	0,3492

2. Verifique los resultados del ejercicio anterior aplicando el *Método de Lagrange Baricéntrico*.

Método de de las diferencias divididas de Newton

1. Aplique el *Método de las Diferencias Divididas Progresivas de Newton* con los datos del punto 4.9 para aproximar los siguientes valores:

- $f(8,2)$ y verifique $f(8,4)$ y $f(8,5)$.
- $f(-0,2)$ y verifique $f(-0,6)$ y $f(-0,333333)$.
- $f(0,15)$ y verifique $f(0,25)$ y $f(0,35)$.
- $f(1,06)$ y verifique $f(1,09)$ y $f(1,12)$.

2. Aplique el *Método de las Diferencias Divididas Regresivas de Newton* para verificar los resultados obtenidos en el punto anterior.

Trazadores cúbicos

- Verifique los resultados de los ejercicios anteriores aplicando el *Método de los Trazadores Cúbicos* con frontera libre.
- Construya un polinomio interpolante mediante *Trazadores Cúbicos* de frontera libre para aproximar $f(x) = e^{-x}$ tomando los siguientes valores:

$$[0; f(0)], \quad [0,25; f(0,25)], \quad [0,5; f(0,5)], \quad [0,75; f(0,75)] \quad \text{y} \quad [1,0; f(1,0)].$$

- Repita el ejercicio anterior pero agregue como información $f'(0,0)$ y $f'(1,0)$ para construir una aproximación con *Trazadores Cúbicos* con frontera sujeta.

x	1,0	1,2	1,4	1,6
$f(x)$	1,35	1,45	1,55	1,65
$f'(x)$	-0,0004	0,0005	0,001	0

Método de Hermite

1. Obtenga el valor de $f(1,1)$ y de $f(1,35)$ aplicando el *Método de Hermite* con los siguientes datos:
2. Obtenga un polinomio interpolante para aproximar $f(x) = e^{-x}$ aplicando el *Método de Hermite* con los siguientes datos:

x	0	0,25	0,50	0,75	1,0
$f(x)$	$f(0)$	$f(0,25)$	$f(0,50)$	$f(0,75)$	$f(1,0)$
$f'(x)$	$f'(0)$	$f'(0,25)$	$f'(0,50)$	$f'(0,75)$	$f'(1,0)$

3. Con los datos del punto anterior obtenga un polinomio interpolante aplicando el *Método de Hermite Segmentado*.

Capítulo 5

Mejor aproximación y ajuste de funciones

5.1. Mejor aproximación

5.1.1. Introducción

Uno de los problemas que suele tener que resolver un ingeniero es el de armar una función que ajuste datos obtenidos experimentalmente. En el capítulo 4 (Interpolación) nos ocupamos de generar funciones polinómicas (aunque podrían haber sido de otro tipo) para representar y graficar una curva que pase por los datos mediante varios procedimientos o métodos que dependen de la cantidad de datos disponibles y de cómo están distribuidos. En todos los casos, una de las condiciones fundamentales es que los puntos x_i sean distintos. ¿Qué hacemos cuando esto no es así, cuando la cantidad de puntos exceden la capacidad de armar polinomios interpolantes o cuando los puntos que usaremos son aproximaciones de los valores reales?

Supongamos que tenemos una serie de datos empíricos obtenidos en laboratorio, tales que el conjunto de datos no cumple estrictamente que los x_i sean distintos, con lo cual para un mismo x_i tenemos varios valores de $f(x_i)$. (En realidad suele suceder que aunque los x_i sean distintos, varios x_j sean suficientemente cercanos como para considerarlos iguales.) O que la cantidad de datos disponible resulte tan grande que generar un polinomio tradicional sea muy poco práctico. Lo que necesitamos, entonces, es una curva que *ajuste* lo mejor posible los datos que disponemos, o sea, que el error entre los puntos y esa función de ajuste o aproximación sea el menor posible, *sin que la curva pase por los puntos dato*.

Existe una forma de estimar este error. Supongamos que efectivamente se cumple que los x_i sean distintos, que $x_0 < x_1 < \dots < x_n$ para los cuales conocemos $f(x_0), f(x_1), \dots, f(x_n)$. Asumamos que la aproximación la haremos con una función que definiremos de la siguiente manera:

$$g(x) = c_0\phi_0(x) + c_1\phi_1(x) + \dots + c_m\phi_m(x) = \sum_{i=0}^m c_i\phi_i(x), \quad (5.1)$$

donde $m < n$, es decir, tenemos menos funciones disponibles que puntos, y las $\phi_i(x)$ son linealmente independientes.

Dado que hemos impuesto que la función elegida no debe pasar por los puntos tomados como dato, buscaremos que el error entre los datos ($f(x_i)$) y los $g(x_i)$ de la función de ajuste sea el menor posible, plantearemos que

$$r_i = f(x_i) - g(x_i), \quad \text{para } 0 \leq i \leq n, \quad (5.2)$$

es decir, que el residuo, sea mínimo. Como se trata de un vector, una forma de analizar este caso es mediante la norma de vectores (y matrices).

5.1.2. Error y normas vectoriales

Para obtener una función que minimice este residuo, analizaremos que opciones disponemos, a saber:

1. Que la norma uno del residuo sea mínima o $\|\mathbf{r}\|_1$ sea mínima;
2. Que la norma infinita del residuo sea mínima o $\|\mathbf{r}\|_\infty$ sea mínima;
3. Que la norma dos (euclídea) del residuo sea mínima o $\|\mathbf{r}\|_2$ sea mínima.

La primera norma es buena si queremos eliminar aquellos valores considerados como desviaciones, por ejemplo, mediciones mal hechas o valores que fácilmente puede inferirse como erróneos. Consiste en minimizar la siguiente expresión:

$$\|\mathbf{r}\|_1 = \sum_{i=0}^n |r_i| = \sum_{i=0}^n |f(x_i) - y(x_i)| \quad (5.3)$$

La segunda es un caso de mínimo-máximo en la cual tenemos que:

$$\min_{c_0, c_1, \dots, c_m} \max_{0 \leq j \leq n} |f(x_j) - y(x_j)|. \quad (5.4)$$

Esto es útil cuando los valores máximos del error deben ser considerados al momento de la verificación.

Ambos casos resultan muy útiles cuando se trabaja con datos discretos, en los que tiene suma importancia verificar la exactitud de esos datos, o eventualmente, encontrar errores de medición, de transcripción, etc.

Los dos casos recién analizados, $\|\mathbf{r}\|_1$, y $\|\mathbf{r}\|_\infty$ llevan a la *programación lineal*, materia que está fuera del alcance de nuestro curso, y que resultan mucho más complejos de analizar que la última opción indicada.

Ésta consiste en minimizar la expresión:

$$\|\mathbf{r}\|_2 = \sqrt{\sum_{i=0}^n |r_i|^2} = \sqrt{\sum_{i=0}^n [f(x_i) - y(x_i)]^2}, \quad (5.5)$$

o, lo que es lo mismo,

$$\|\mathbf{r}\|_2^2 = \sum_{i=0}^n |r_i|^2 = \sum_{i=0}^n [f(x_i) - y(x_i)]^2. \quad (5.6)$$

Como nuestra función la podemos expresar así:

$$y(x) = \sum_{j=0}^m c_j \phi_j(x), \quad (5.7)$$

tendremos que la expresión a minimizar es

$$E(c_0, c_1, \dots, c_m) = \sum_{i=0}^n \left[f(x_i) - \sum_{j=0}^m c_j \phi_j(x_i) \right]^2, \quad (5.8)$$

de ahí el nombre de *método de los cuadrados mínimos*, pues lo que se minimiza es el cuadrado del residuo¹.

¹El método de los cuadrados mínimos fue inventado por *Carl Friedrich Gauss* en 1801 para estimar la trayectoria del planeta enano *Ceres*, si bien los fundamentos del mismo ya los había planteado en 1795. *Ceres* fue considerado planeta durante 50 años (entre 1800 y 1850). Luego fue considerado como un asteroide del *cinturón de asteroides* entre Marte y Júpiter. En 2006 fue incluido junto con Plutón (antes el noveno planeta) y los cuerpos más recientemente descubiertos Eris, Makemake y Haumea en la nueva categoría de *planeta enano*. Más detalles de la biografía de Gauss están disponibles en <http://www-history.mcs.st-and.ac.uk/Biographies/Gauss.html>.

5.1.3. Método de los cuadrados mínimos

Para obtener que la función $E(c_0, c_1, \dots, c_m)$ sea mínima, debemos aplicar un concepto conocido: hacer que $\frac{\partial E}{\partial c_j} = 0$, puesto que E es función de los coeficientes c_j . En consecuencia, tendremos que:

$$\begin{aligned}\frac{\partial E}{\partial c_j} &= \frac{\partial}{\partial c_j} \left[\sum_{i=0}^n \left(f(x_i) - \sum_{k=0}^m c_k \phi_k(x_i) \right)^2 \right] = 0 \\ &= \sum_{i=0}^n \frac{\partial}{\partial c_j} \left(f(x_i) - \sum_{k=0}^m c_k \phi_k(x_i) \right)^2 = 0.\end{aligned}\tag{5.9}$$

que si desarrollamos nos queda:

$$\frac{\partial E}{\partial c_j} = 2 \sum_{i=0}^n \left(f(x_i) - \sum_{k=0}^m c_k \phi_k(x_i) \right) (-\phi_j(x_i)) = 0 \text{ para } j = 0; 1; \dots; m.\tag{5.10}$$

Al distribuir el producto nos queda:

$$\begin{aligned}\sum_{i=0}^n \left(f(x_i) \phi_j(x_i) - \sum_{k=0}^m c_k \phi_k(x_i) \phi_j(x_i) \right) &= 0 \\ \sum_{i=0}^n f(x_i) \phi_j(x_i) - \sum_{i=0}^n \sum_{k=0}^m c_k \phi_k(x_i) \phi_j(x_i) &= 0 \\ \sum_{i=0}^n \sum_{k=0}^m c_k \phi_k(x_i) \phi_j(x_i) &= \sum_{i=0}^n f(x_i) \phi_j(x_i),\end{aligned}\tag{5.11}$$

para $j = 0; 1; \dots; m$. Como podemos intercambiar las sumatorias, finalmente nos queda:

$$\sum_{k=0}^m c_k \sum_{i=0}^n \phi_k(x_i) \phi_j(x_i) = \sum_{i=0}^n f(x_i) \phi_j(x_i),\tag{5.12}$$

para $j = 0; 1; \dots; m$.

Avancemos un poco más. Al desarrollar la sumatoria en i del término de la izquierda, nos queda:

$$\sum_{i=0}^n \phi_k(x_i) \phi_j(x_i) = \phi_k(x_0) \phi_j(x_0) + \phi_k(x_1) \phi_j(x_1) + \dots + \phi_k(x_n) \phi_j(x_n).\tag{5.13}$$

Lo mismo podemos hacer con la sumatoria del término de la derecha, con lo que tenemos

$$\sum_{i=0}^n f(x_i) \phi_j(x_i) = f(x_0) \phi_j(x_0) + f(x_1) \phi_j(x_1) + \dots + f(x_n) \phi_j(x_n).\tag{5.14}$$

Para facilitar la notación, definiremos lo siguiente:

$$\sum_{i=0}^n \phi_k(x_i) \phi_j(x_i) = (\phi_k, \phi_j)\tag{5.15}$$

$$\sum_{i=0}^n f(x_i) \phi_j(x_i) = (f, \phi_j).\tag{5.16}$$

Entonces, la expresión que nos queda es

$$\sum_{k=0}^m c_k (\phi_k, \phi_j) = (f, \phi_j), \quad (5.17)$$

para $j = 0; 1; \dots; m$. Ahora desarrollaremos la sumatoria en k , con lo cual obtenemos lo siguiente:

$$c_0 (\phi_0, \phi_j) + c_1 (\phi_1, \phi_j) + \dots + c_m (\phi_m, \phi_j) = (f, \phi_j). \quad (5.18)$$

Como $j = 0; 1; \dots; m$, entonces podemos armar $m + 1$ ecuaciones, lo que finalmente nos deja:

$$\begin{aligned} c_0 (\phi_0, \phi_0) + c_1 (\phi_1, \phi_0) + \dots + c_m (\phi_m, \phi_0) &= (f, \phi_0) \\ c_0 (\phi_0, \phi_1) + c_1 (\phi_1, \phi_1) + \dots + c_m (\phi_m, \phi_1) &= (f, \phi_1) \\ &\vdots \\ c_0 (\phi_0, \phi_m) + c_1 (\phi_1, \phi_m) + \dots + c_m (\phi_m, \phi_m) &= (f, \phi_m), \end{aligned} \quad (5.19)$$

que podemos escribir también en forma matricial como

$$\begin{bmatrix} (\phi_0, \phi_0) & (\phi_1, \phi_0) & \dots & (\phi_m, \phi_0) \\ (\phi_0, \phi_1) & (\phi_1, \phi_1) & \dots & (\phi_m, \phi_1) \\ \vdots & \vdots & \ddots & \vdots \\ (\phi_0, \phi_m) & (\phi_1, \phi_m) & \dots & (\phi_m, \phi_m) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{bmatrix} = \begin{bmatrix} (f, \phi_0) \\ (f, \phi_1) \\ \vdots \\ (f, \phi_m) \end{bmatrix}. \quad (5.20)$$

Esta matriz resulta ser simétrica, pues $(\phi_i, \phi_j) = (\phi_j, \phi_i)$, y definida positiva. El problema se reduce a resolver un sistema de ecuaciones lineales cuyas incógnitas son los coeficientes c_k . Obtenidos estos coeficientes, los reemplazamos en la función que hemos definido, que será la que aproxime nuestros puntos.

Existe otra forma de plantear el problema, esta vez en forma matricial desde el principio. Supongamos que representamos nuestros puntos con la función elegida. Entonces nos queda:

$$f(x_0) = \sum_{k=0}^m c_k \phi_k(x_0) = c_0 \phi_0(x_0) + c_1 \phi_1(x_0) + \dots + c_m \phi_m(x_0) \quad (5.21)$$

$$f(x_1) = \sum_{k=0}^m c_k \phi_k(x_1) = c_0 \phi_0(x_1) + c_1 \phi_1(x_1) + \dots + c_m \phi_m(x_1) \quad (5.22)$$

\vdots

$$f(x_n) = \sum_{k=0}^m c_k \phi_k(x_n) = c_0 \phi_0(x_n) + c_1 \phi_1(x_n) + \dots + c_m \phi_m(x_n) \quad (5.23)$$

Si escribimos esto en forma matricial tenemos

$$\begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{bmatrix} = \begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \dots & \phi_m(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_m(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \dots & \phi_m(x_n) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{bmatrix}, \quad (5.24)$$

que resulta ser un sistema de m incógnitas con n ecuaciones, donde $m < n$, en el cual no existe una única solución. Si hacemos

$$\mathbf{f} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{bmatrix}; \Phi = \begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \dots & \phi_m(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_m(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \dots & \phi_m(x_n) \end{bmatrix} \text{ y } \mathbf{c} = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{bmatrix}, \quad (5.25)$$

podemos decir que nos queda una ecuación del tipo $\mathbf{f} = \Phi \mathbf{c}$. Como lo que buscamos es aproximar una función, definamos el residuo como $\mathbf{r} = \mathbf{f} - \Phi \mathbf{c}$. Al igual que en el desarrollo anterior, vamos a obtener nuestra nueva función haciendo que $\|\mathbf{r}\|_2^2$ sea mínimo. En consecuencia, tenemos

$$\|\mathbf{r}\|_2^2 = \|\mathbf{f} - \Phi \mathbf{c}\|_2^2. \quad (5.26)$$

Recordemos que $\|\mathbf{r}\|_2^2 = \mathbf{r}^T \cdot \mathbf{r}$, entonces tendremos que

$$\mathbf{r}^T \cdot \mathbf{r} = (\mathbf{f} - \Phi \mathbf{c})^T \cdot (\mathbf{f} - \Phi \mathbf{c}). \quad (5.27)$$

De nuevo, para obtener que el residuo sea mínimo, anulemos la primera derivada, es decir, hagamos

$$\frac{\partial (\mathbf{r}^T \cdot \mathbf{r})}{\partial c_j} = \frac{\partial}{\partial c_j} [(\mathbf{f} - \Phi \mathbf{c})^T \cdot (\mathbf{f} - \Phi \mathbf{c})] = 0. \quad (5.28)$$

Al derivar nos queda

$$-\Phi^T \cdot (\mathbf{f} - \Phi \mathbf{c}) - (\mathbf{f} - \Phi \mathbf{c})^T \cdot \Phi = 0, \quad (5.29)$$

que desarrollada se transforma en

$$\Phi^T \cdot \mathbf{f} - \Phi^T \cdot \Phi \cdot \mathbf{c} + \mathbf{f}^T \cdot \Phi - \mathbf{c}^T \cdot \Phi^T \cdot \Phi = 0. \quad (5.30)$$

Como $\Phi^T \cdot \mathbf{f} = \mathbf{f}^T \cdot \Phi$ y $\mathbf{c}^T \cdot \Phi^T \cdot \Phi = \Phi^T \cdot \Phi \cdot \mathbf{c}$, la ecuación anterior podemos escribirla así:

$$\begin{aligned} \Phi^T \cdot \mathbf{f} - \Phi^T \cdot \Phi \cdot \mathbf{c} + \Phi^T \cdot \mathbf{f} - \Phi^T \cdot \Phi \cdot \mathbf{c} &= 0 \\ 2(\Phi^T \cdot \mathbf{f} - \Phi^T \cdot \Phi \cdot \mathbf{c}) &= 0 \\ \Phi^T \cdot \mathbf{f} - \Phi^T \cdot \Phi \cdot \mathbf{c} &= 0 \Rightarrow \\ \Phi^T \cdot \Phi \cdot \mathbf{c} &= \Phi^T \cdot \mathbf{f}, \end{aligned} \quad (5.31)$$

donde $\Phi^T \cdot \Phi$ es una matriz simétrica definida positiva, y tiene la forma

$$\begin{bmatrix} (\phi_0, \phi_0) & (\phi_1, \phi_0) & \dots & (\phi_m, \phi_0) \\ (\phi_0, \phi_1) & (\phi_1, \phi_1) & \dots & (\phi_m, \phi_1) \\ \vdots & \vdots & \ddots & \vdots \\ (\phi_0, \phi_m) & (\phi_1, \phi_m) & \dots & (\phi_m, \phi_m) \end{bmatrix}; \quad (5.32)$$

y $\Phi^T \cdot \mathbf{f}$ es un vector que tiene la forma

$$\begin{bmatrix} (f, \phi_0) \\ (f, \phi_1) \\ \vdots \\ (f, \phi_m) \end{bmatrix}. \quad (5.33)$$

Si hacemos $\mathbf{A} = \Phi^T \cdot \Phi$, $\mathbf{x} = \mathbf{c}$ y $\mathbf{B} = \Phi^T \cdot \mathbf{f}$, volvemos a tener nuestro sistema de ecuaciones lineales en la forma $\mathbf{A} \cdot \mathbf{x} = \mathbf{B}$. De nuevo, el método de los cuadrados mínimos no es otra cosa que la resolución de un sistema de ecuaciones lineales para obtener los coeficientes c_k de nuestra función de ajuste o aproximación, algo a lo que habíamos llegado mediante la deducción anterior.

Este método suele usarse para obtener la recta de regresión. Para obtenerla, basta observar que

$$y(x) = \sum_{i=0}^m c_i \cdot \phi_i(x) = c_0 + c_1 \cdot x,$$

es la recta que ajusta nuestros datos, con lo cual $\phi_0 = 1$ y $\phi_1 = x$. El siguiente paso es armar la matriz \mathbf{A} . Sabemos que

$$(\phi_k, \phi_j) = \sum_{i=0}^n \phi_k(x_i) \cdot \phi_j(x_i) \text{ y } (f, \phi_j) = \sum_{i=0}^n f(x_i) \cdot \phi_j(x_i),$$

entonces podemos escribir las componentes de \mathbf{A} y \mathbf{B} como

$$(\phi_0, \phi_0) = \sum_{i=0}^n 1 \cdot 1 = n + 1 \quad (5.34)$$

$$(\phi_1, \phi_0) = \sum_{i=0}^n x_i \cdot 1 = \sum_{i=0}^n x_i \quad (5.35)$$

$$(\phi_0, \phi_1) = (\phi_1, \phi_0) = \sum_{i=0}^n x_i \quad (5.36)$$

$$(\phi_1, \phi_1) = \sum_{i=0}^n (x_i \cdot x_i) = \sum_{i=0}^n (x_i)^2 \quad (5.37)$$

$$(f, \phi_0) = \sum_{i=0}^n (f(x_i) \cdot 1) = \sum_{i=0}^n f(x_i) \quad (5.38)$$

$$(f, \phi_1) = \sum_{i=0}^n (f(x_i) \cdot x_i), \quad (5.39)$$

y nuestro sistema quedará definido así:

$$\begin{bmatrix} n+1 & \sum_{i=0}^n x_i \\ \sum_{i=0}^n x_i & \sum_{i=0}^n (x_i)^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^n f(x_i) \\ \sum_{i=0}^n (f(x_i) \cdot x_i) \end{bmatrix}. \quad (5.40)$$

Los valores de c_0 y c_1 los obtenemos con estas dos expresiones:

$$c_0 = \frac{\sum_{i=0}^n (x_i)^2 \sum_{i=0}^n f(x_i) - \sum_{i=0}^n (f(x_i) \cdot x_i) \sum_{i=0}^n x_i}{(n+1) \sum_{i=0}^n (x_i)^2 - \left(\sum_{i=0}^n x_i \right)^2}, \quad (5.41)$$

$$c_1 = \frac{(n+1) \sum_{i=0}^n (f(x_i) \cdot x_i) - \sum_{i=0}^n x_i \sum_{i=0}^n f(x_i)}{(n+1) \sum_{i=0}^n (x_i)^2 - \sum_{i=0}^n x_i}. \quad (5.42)$$

Existen algunas variantes para este tipo de regresiones, que son:

$$\ln(y) = \ln(c_0) + c_1 \ln(x) \quad (y = c_0 x^{c_1}) \quad (5.43)$$

$$\ln(y) = \ln(c_0) + c_1 x \quad (y = c_0 e^{c_1 x}) \quad (5.44)$$

$$y = c_0 + c_1 \ln(x), \quad (5.45)$$

que permiten ajustar valores según distintas curvas. Sin embargo, las expresiones (5.43) y (5.44) no son ajustes por cuadrados mínimos en un sentido estricto. Lo correcto sería proponer una función del tipo $\sum_i c_i \phi_i(x)$ en lugar de transformar los datos. (Para más detalles, véase [3].)

Si ampliamos este esquema a una función polinómica de grado mayor o igual a 2, tendremos que

$$y(x) = \sum_{k=0}^m c_k \phi_k(x) = \sum_{k=0}^m c_k x^k = c_0 + c_1 x + c_2 x^2 + \dots + c_m x^m. \quad (5.46)$$

Al armar el sistema de ecuaciones nos quedará el sistema de ecuaciones lineales a continuación

$$\begin{bmatrix} n+1 & \sum_{i=0}^n x_i & \dots & \sum_{i=0}^n x_i^{m-1} & \sum_{i=0}^n x_i^m \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \dots & \sum_{i=0}^n x_i^m & \sum_{i=0}^n x_i^{m+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \sum_{i=0}^n x_i^{m-1} & \sum_{i=0}^n x_i^m & \dots & \sum_{i=0}^n x_i^{2(m-1)} & \sum_{i=0}^n x_i^{2m-1} \\ \sum_{i=0}^n x_i^m & \sum_{i=0}^n x_i^{m+1} & \dots & \sum_{i=0}^n x_i^{2m-1} & \sum_{i=0}^n x_i^{2m} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{m-1} \\ c_m \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^n f(x_i) \\ \sum_{i=0}^n (f(x_i) \cdot x_i) \\ \vdots \\ \sum_{i=0}^n (f(x_i) \cdot x_i^{m-1}) \\ \sum_{i=0}^n (f(x_i) \cdot x_i^m) \end{bmatrix}. \quad (5.47)$$

La matriz de coeficientes es similar a una *matriz de VanderMonde*, matriz que obtuvimos para interpolar una serie de puntos, de ahí que cualquier ajuste de curvas hecho con polinomios resulta ser un problema *mal condicionado*. Por supuesto, la mala condición de la matriz se hace cada vez más evidente a medida que m sea más grande. Es por eso que no se recomienda trabajar con polinomios de grado mayor a 4 o 5, para evitar que la mala condición de la matriz sea un problema. Aún así, trabajar con un polinomio de grado 5 conlleva trabajar con coeficientes que incluyen x^{10} , lo que resulta casi equivalente a interpolar con polinomios de grado 10. Por esto, conviene que recordemos que el ajuste polinomial, al igual que la interpolación polinomial son problemas con tendencia a ser mal condicionados.

5.2. Ajuste de funciones

5.2.1. Introducción

En el punto anterior hemos visto un método para ajustar curvas a partir de datos numéricos (discretos), con el objetivo de obtener valores de la función $f(x)$ para valores de x distintos a los datos en el intervalo dado. E

Ahora bien, existen situaciones en las cuales aún conociendo la función $f(x)$, resulta conveniente efectuar algún tipo de aproximación. Un ejemplo típico de ello es el caso de las funciones trigonométricas (por ejemplo, $\cos(x)$), para la cual es necesario realizar alguna aproximación para calcular sus valores. La más común es la hecha mediante las series de Taylor. Para estas funciones puede ser muy útil aplicar el desarrollo en series, pero no suele ser el caso general, puesto que las series de Taylor son válidas sólo en el entorno de un punto, lo que le quita generalidad.

¿Y en qué casos necesitaríamos nosotros contar con una aproximación de una función conocida? Supongamos que tenemos la siguiente función:

$$f(x) = \frac{e^x - \cos(x)}{\ln(x) \cdot \arctan(x)},$$

en un intervalo $[a, b]$. Supongamos además, que nuestro problema exige que integremos esa función $f(x)$ en el intervalo dado. Podemos ver que la situación ya no es tan fácil como parece. Si bien

disponemos de la función, hallar la primitiva puede ser todo un desafío, e incluso, imposible. Pero de alguna manera debemos salvar el escollo.

¿Que tal si en vez de hacer una integral «analítica» nos orientamos hacia una solución numérica? La idea no es tan descabellada pues lo que nosotros necesitamos generalmente es el resultado numérico y no la primitiva de la misma. Hagamos uso entonces de nuestras herramientas numéricas aprendidas anteriormente y, si es necesario, adecuemos nuestras expresiones al caso analizado.

5.2.2. Aproximación por mínimos cuadrados

Recordemos qué significa reducir al mínimo el error cuadrático entre la función y el polinomio de aproximación. Supongamos por un momento que conocemos tanto la función $f(x)$ como el polinomio de aproximación $P(x)$, en el intervalo $[a, b]$. Podemos graficar nuestra función y nuestro polinomio de manera que nos queden las curvas que se ven la figura 5.1.

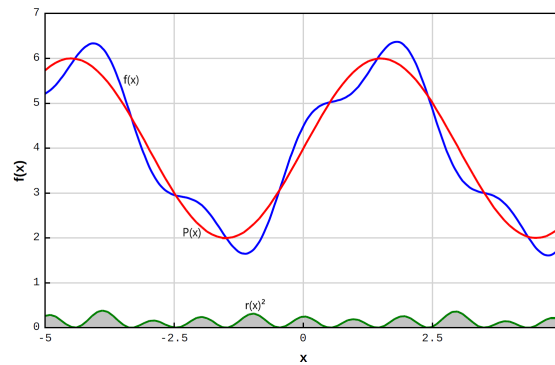


Figura 5.1: *Error cuadrático*

Si definimos que

$$E(a_k) = \int_a^b \left[f(x) - \sum_{k=0}^n a_k x^k \right]^2 dx = \|r(a_k)\|_2^2, \quad (5.48)$$

entonces podemos ver que el área bajo la curva $r(a_k)^2$ es el valor de nuestra integral. Por lo tanto, para que nuestro error cuadrático sea mínimo, deberemos buscar que la curva $r(a_k)^2$ sea lo más parecida al eje de abscisas. (Esta definición es similar a la vista para aproximación de curvas por cuadrados mínimos.)

Para ello, vamos a derivar la función $E(a_k)$ respecto de los coeficientes a_k para obtener los valores, de dichos coeficientes, que hagan mínimo el error cuadrático, tal como hicimos para el caso de un ajuste discreto. Entonces tendremos:

$$\frac{\partial E(a_0, a_1, \dots, a_n)}{\partial a_j} = 0 \Rightarrow \frac{\partial}{\partial a_j} \left\{ \int_a^b \left[f(x) - \sum_{k=0}^n a_k x^k \right]^2 dx \right\} = 0. \quad (5.49)$$

Al derivar nos queda:

$$2 \cdot \int_a^b \left[f(x) - \sum_{k=0}^n a_k x^k \right] x^j dx = 0, \quad (5.50)$$

y como el 2 no incide, nos queda:

$$\int_a^b \left[f(x) - \sum_{k=0}^n a_k x^k \right] x^j dx = 0. \quad (5.51)$$

Si distribuimos el producto dentro de la integral, conmutamos la integral y la sumatoria, y pasamos la integral que incluye $f(x)$ al otro lado de la igualdad, nos queda:

$$\sum_{k=0}^n a_k \int_a^b x^{k+j} dx = \int_a^b x^j f(x) dx \quad \text{para } j = 0; 1; \dots; n. \quad (5.52)$$

¿Qué es lo hemos obtenido? Nuevamente, como en la aproximación de puntos discretos, un sistema de ecuaciones lineales de dimensión $n+1 \times n+1$. Sin embargo, no todo es tan sencillo. Analicemos un poco más en detalle la integral que afecta a los coeficientes a_k . Tenemos que:

$$\int_a^b x^{k+j} dx = \frac{x^{k+j+1}}{k+j+1} \Big|_a^b \Rightarrow \int_a^b x^{k+j} dx = \frac{b^{k+j+1} - a^{k+j+1}}{k+j+1}. \quad (5.53)$$

Si definimos que $a = 0$ y $b = 1$, entonces la integral definida resulta en el coeficiente:

$$\frac{1}{k+j+1}. \quad (5.54)$$

La matriz que se genera a partir del coeficiente anterior es conocida como *matriz de Hilbert*, que es una matriz *mal condicionada*. Como en el caso anterior de ajuste discreto, al tener una matriz mal condicionada, el sistema es muy sensible a los cambios en los datos, o modificaciones de la matriz de coeficientes, es decir, es muy sensible a los errores inherentes.

Un segundo problema, en este caso operativo, es que si por algún motivo se desea agregar un término más al polinomio, hay que recalcular el sistema (agregar una columna y una fila), lo que significa mucho trabajo adicional. Y nada asegura que los nuevos resultados estén exentos de errores. De todos modos, contamos con método muy potente para ajustar funciones pero con inconvenientes operativos en el planteo numérico. Podemos buscar la forma de mejorarlo. Veamos como.

¿Cuál sería la mejor matriz de coeficientes para resolver un sistema de ecuaciones lineales? Evidentemente, aquella que independice cada incógnita de las otras. O sea, que la matriz de coeficientes sea una matriz diagonal. Supongamos modificar levemente la expresión del polinomio de aproximación por la siguiente:

$$P(x) = \sum_{k=0}^n a_k \phi_k(x), \quad (5.55)$$

En principio, no hemos hecho sino un cambio de notación, llamando a x^k como $\phi_k(x)$. Veamos qué ventajas nos trae esto. Por lo pronto, ahora disponemos de más posibilidades porque el método no cambia si proponemos una suma de funciones en vez de un polinomio como función de aproximación, tal como vimos para ajuste de curvas. Entonces nos queda:

$$\sum_{k=0}^n a_k \int_a^b \phi_k(x) \phi_j(x) dx = \int_a^b \phi_j f(x) dx \quad \text{para } j = 0; 1; \dots; n. \quad (5.56)$$

que conceptualmente es muy parecido a lo anterior. Pero con una diferencia: ahora podemos tomar cualquier función para definir nuestras funciones $\phi_k(x)$ y por lo tanto, también nuestros $\phi_j(x)$. Busquemos entonces que nuestra matriz de coeficientes se convierta en una matriz diagonal. ¿Y cómo lo logramos? Sencillamente estableciendo que se cumpla lo siguiente:

$$\int_a^b \phi_k(x) \phi_j(x) dx = \begin{cases} 0 & \text{si } k \neq j \\ M > 0 & \text{si } k = j \end{cases},$$

donde M es un valor cualquiera. Por supuesto, lo ideal sería que $M = 1$. Esta condición que deben cumplir las $\phi_k(x)$ asegura que las funciones sean ortogonales y que la matriz de coeficientes sea diagonal.

No hemos dicho nada aún acerca de las funciones $\phi_k(x)$. Como estamos tratando de aproximar una función cualquiera, una buena idea es proponer que esas funciones sean también polinomios. Para hallar estos polinomios ortogonales entre sí, debemos agregar una segunda condición que es agregar una función *de peso*. Esta función de peso tiene por objeto asignar diferentes grados de importancia a las aproximaciones de ciertas partes del intervalo. En esta nueva situación tenemos:

$$\frac{\partial E(a_0; a_1; \dots, a_n)}{\partial a_j} = 0 \Rightarrow \frac{\partial}{\partial a_j} \left\{ \int_a^b w(x) \left[f(x) - \sum_{k=0}^n a_k \phi_k(x) \right]^2 dx \right\} = 0, \quad (5.57)$$

con lo cual finalmente nos queda:

$$\int_a^b w(x) \left[f(x) - \sum_{k=0}^n a_k \phi_k \right] \phi_j dx = 0. \quad (5.58)$$

En este caso se debe cumplir que:

$$\int_a^b w(x) \phi_k(x) \phi_j(x) dx = \begin{cases} 0 & \text{si } k \neq j \\ M \neq 0 & \text{si } k = j \end{cases}.$$

Si definimos que $w(x) = 1$, volvemos a tener nuestra expresión original para los $\phi_k(x)$ y los $\phi_j(x)$. Y si, además, el intervalo de interpolación lo fijamos en $[-1, 1]$, el resultado es que mediante este procedimiento obtenemos los *polinomios de Legendre*. Estos polinomios los usaremos más adelante para integrar numéricamente.

5.2.3. Polinomios de Legendre

Veremos cómo se calculan los polinomios de Legendre. Antes, debemos recordar cómo se obtenía un conjunto de vectores ortogonales a partir de un conjunto de vectores no ortogonal. Esto se conseguía mediante el *proceso de Gram-Schmidt*. Adaptémoslo para el caso de funciones.

Para empezar, debemos proponer las dos primeras funciones $\phi(x)$. Estas funciones son:

$$\phi_0(x) = 1; \quad \phi_1(x) = x - B_1 \Rightarrow \phi_1(x) = (x - B_1) \phi_0(x).$$

donde B_1 es nuestra incógnita. Para obtenerla debemos plantear que:

$$\int_a^b w(x) \phi_0(x) \phi_1(x) dx = 0 \Rightarrow \int_a^b w(x) \phi_0(x) \phi_0(x) (x - B_1) dx = 0. \quad (5.59)$$

Distribuyendo en el paréntesis, obtenemos:

$$\int_a^b w(x) [\phi_0(x)]^2 x dx - B_1 \int_a^b w(x) [\phi_0(x)]^2 dx = 0 \quad (5.60)$$

y entonces B_1 podemos hallarla con:

$$B_1 = \frac{\int_a^b w(x) [\phi_0(x)]^2 x dx}{\int_a^b w(x) [\phi_0(x)]^2 dx}. \quad (5.61)$$

Para los siguientes polinomios, es decir, cuando $k \geq 2$, debemos proponer que:

$$\phi_k(x) = (x - B_k) \phi_{k-1}(x) - C_k \phi_{k-2}(x) \text{ en } [a, b]. \quad (5.62)$$

Operando algebraicamente en forma similar a la anterior obtenemos los coeficientes B_k y C_k :

$$B_k = \frac{\int_a^b x w(x) [\phi_{k-1}(x)]^2 dx}{\int_a^b w(x) [\phi_{k-1}(x)]^2 dx} \quad (5.63)$$

$$C_k = \frac{\int_a^b x w(x) \phi_{k-1}(x) \phi_{k-2}(x) dx}{\int_a^b w(x) [\phi_{k-2}(x)]^2 dx} \quad (5.64)$$

Como hemos dicho, la función de peso en el caso de los polinomios de Legendre es $w(x) = 1$ y el intervalo $[-1; 1]$, por lo que las expresiones quedan como sigue:

1. El coeficiente B_1 se obtiene con:

$$B_1 = \frac{\int_{-1}^1 x dx}{\int_{-1}^1 dx}; \quad (5.65)$$

2. Los coeficientes B_k y C_k se obtienen con las siguientes expresiones

$$B_k = \frac{\int_{-1}^1 x [\phi_{k-1}(x)]^2 dx}{\int_{-1}^1 [\phi_{k-1}(x)]^2 dx}, \quad y; \quad (5.66)$$

$$C_k = \frac{\int_{-1}^1 x \phi_{k-1}(x) \phi_{k-2}(x) dx}{\int_{-1}^1 [\phi_{k-2}(x)]^2 dx}. \quad (5.67)$$

Existe un segundo conjunto de polinomios ortogonales muy utilizados que son los *polinomios de Chebishev*. También se generan aplicando las expresiones generales ya vistas, pero con una función de peso diferente: $w(x) = \frac{1}{\sqrt{1-x^2}}$. (Más detalles de estos polinomios en [3].)

5.3. Notas finales

Tanto la aproximación discreta de curvas como el ajuste de funciones tienen un amplio uso en la ingeniería. En el primer caso, existen muchas expresiones matemáticas resultantes de aproximar valores obtenidos experimentalmente en laboratorios o mediante mediciones realizadas sobre prototipos. En la ingeniería hidráulica se tienen muchas expresiones empíricas que surgen de experiencias en laboratorio que luego resultan en fórmulas matemáticas obtenidas mediante aproximaciones discretas.

Con el ajuste de funciones ocurre algo similar. El ejemplo más interesante es el uso de polinomios de Legendre en la cuadratura de Gauss para integrar numéricamente. Estos polinomios ajustan cualquier tipo de funciones y en particular, a cualquier polinomio, lo que facilita obtener soluciones numéricas «exactas» de cualquier integral numérica que incluya funciones polinómicas, como se verá en el capítulo 6.

Ejercicios

1. A partir de los datos de la siguiente tabla, obtenga la constante de la función de ajuste propuesta: $h(x) = a_0 + a_1 x + a_2 x^2$.

Aplique un *Método Directo* para resolver el sistema de ecuaciones lineales resultante.

X	1	2	3	4	5	6
Y	3,13	3,79	6,94	12,62	20,86	31,53

X	0,5	1	1,5	2	2,5	3
Y	1,630	1,844	2,196	2,778	3,736	5,318

2. Obtenga las constantes de la función de ajuste propuesta, $h(x) = a_0 + a_1 e^x$, a partir de los siguientes datos y aplique un *Método Directo* para resolver el sistema de ecuaciones lineales resultante
3. Con los datos de la siguiente tabla y aplicando el *Método de los Cuadrados Mínimos*, obtenga:
 - a) Un polinomio de ajuste de segundo grado ($h(x) = a_0 + a_1 x + a_2 x^2$).
 - b) Un polinomio de ajuste de tercer grado ($h(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$).
 - c) Una función de ajuste $h(x) = a_0 x^{a_1}$. (Sugerencia: use $h(x) = \ln a_0 + a_1 \ln x$, transformando X en $\ln X$ y Y en $\ln Y$.)
 - d) Una función de ajuste $h(x) = a_0 e^{a_1 x}$. (Sugerencia: use $h(x) = \ln a_0 + a_1 x$, transformando Y en $\ln Y$.)

Compare todos los resultados.

X	4,0	4,2	4,5	4,7	5,1	5,5	5,9	6,3	6,8	7,1
Y	102,56	1113,18	130,11	142,05	167,53	195,14	224,87	256,73	299,50	326,72

Capítulo 6

Diferenciación e integración numérica

6.1. Diferenciación numérica

Como vimos en la introducción del capítulo 3, trabajar en forma simbólica resulta bastante complicado cuando se requiere el uso de computadoras, aún cuando existen programas que lo hagan (en parte). No siempre las soluciones analíticas son aplicables al problema que se está tratando de resolver, y peor aún, en muchos casos no hay tal solución analítica, como veremos más adelante.

Por otro lado, no siempre disponemos de las herramientas para trabajar en forma simbólica (o analítica). Cuando sólo contamos con datos obtenidos de mediciones o de cálculos previos, y no de funciones, no suele ser práctico trabajar en forma simbólica. Obtener «la derivada de una función» con datos discretos no tiene mucho sentido.

Al mismo tiempo, muchos programas de aplicación ingenieril no pueden almacenar o guardar en sus líneas de código una base de datos que incluya las derivadas de cualquier función (lo mismo se aplica al caso inverso, la integración). La cantidad de información y la aleatoriedad que puede presentar una exigencia de este tipo vuelve impracticable realizar esto en cada programa, además de llevar a construir interfaces amigables que contribuyen a aumentar los requerimientos de memoria, tanto de operación como de almacenamiento.

Veremos a continuación como encarar la diferenciación mediante métodos numéricos con ayuda de varios ejemplos, analizando las ventajas y las desventajas de cada método empleado en la discretización para luego analizar la *extrapolación de Richardson*, método que puede usarse también para otro tipo de problemas.

6.1.1. Diferencias progresivas, regresivas y centradas

La diferenciación es un tema muy conocido por los estudiantes de ingeniería. Los primeros años de la carrera consisten en estudiar en detalle cómo caracterizar y conocer a fondo una función dada, de manera que para analizar si tiene máximos o mínimos, si es convexa o cóncava, si puede aproximarse mediante un desarrollo en serie, lo primero que se aprende es el concepto de *derivada*, tanto total como parcial. Tomemos, por ejemplo, la función

$$f(x) = \text{sen} \left(\frac{2\pi}{b} x \right).$$

Hallar la derivada primera de $f(x)$ respecto de x es un procedimiento sencillo pues resulta ser

$$f'(x) = \frac{d f(x)}{dx} = \left(\frac{2\pi}{b} \right) \cos \left(\frac{2\pi}{b} x \right).$$

Si queremos conocer la derivada en el punto $x = \frac{b}{6}$ basta con reemplazar ese valor en la expresión anterior:

$$f' \left(\frac{b}{6} \right) = \left(\frac{2\pi}{b} \right) \cos \left(\frac{2\pi}{b} \frac{b}{6} \right) = \left(\frac{2\pi}{b} \right) \cos \left(\frac{\pi}{3} \right)$$

Y si finalmente le damos un valor a b , (por ejemplo, $b = 6$), el valor de nuestra derivada en $x = \frac{b}{6} = 1$ es

$$f'(1) = \left(\frac{\pi}{3} \right) \cos \left(\frac{\pi}{3} \right) = \frac{\pi}{6} \approx 0,5236$$

Supongamos ahora que queremos obtener ese mismo valor pero no recordamos cómo obtener la derivada en forma analítica. Aplicando el concepto del cual se deduce, podemos decir que

$$\frac{df(x)}{dx} = f'(x) \approx \frac{f(x + \Delta x) - f(x)}{\Delta x}, \quad (6.1)$$

que también suele escribirse como:

$$f'(x) \approx \frac{f(x + h) - f(x)}{h}. \quad (6.2)$$

Para hallar la derivada en nuestro punto $x = \frac{b}{6}$ con $b = 6$ adoptemos el valor $h = 0,1$. Así tendremos que

$$\begin{aligned} f'(1) &\approx \frac{f(1,1) - f(1)}{0,1} = \frac{\text{sen} \left(\frac{\pi}{3} 1,1 \right) - \text{sen} \left(\frac{\pi}{3} \right)}{0,1} \\ f'(1) &\approx \frac{0,9135 - 0,8660}{0,1} = 0,4750. \end{aligned}$$

Podemos ver que nuestra aproximación es razonable pero no muy buena, y que el error cometido es del orden del 10%. Como no estamos conformes con el resultado obtenido, proponemos otro algoritmo para hallar el valor buscado. Este algoritmo es:

$$f'(x) \approx \frac{f(x) - f(x - \Delta x)}{\Delta x} \quad (6.3)$$

o, como también suele escribirse

$$f'(x) \approx \frac{f(x) - f(x - h)}{h}. \quad (6.4)$$

Hallemos ahora el valor de la derivada utilizando este nuevo algoritmo. El resultados es:

$$\begin{aligned} f'(1) &\approx \frac{f(1) - f(0,9)}{0,1} = \frac{\text{sen} \left(\frac{\pi}{3} \right) - \text{sen} \left(\frac{\pi}{3} 0,9 \right)}{0,1} \\ f'(1) &\approx \frac{0,8660 - 0,8090}{0,1} = 0,5700. \end{aligned}$$

De nuevo, el valor obtenido tampoco es una aproximación muy buena, pues el error cometido del orden del 8%. Una vez más, no estamos conformes con el resultado que nos arrojó este algoritmo y proponemos este otro:

$$f'(x) \approx \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x}, \quad (6.5)$$

o también:

$$f'(x) \approx \frac{f(x + h) - f(x - h)}{2h}. \quad (6.6)$$

Si reemplazamos los valores, obtenemos:

$$f'(1) \approx \frac{f(1,1) - f(0,9)}{0,2} = \frac{\text{sen}\left(\frac{\pi}{3}1,1\right) - \text{sen}\left(\frac{\pi}{3}0,9\right)}{0,2}$$

$$f'(1) \approx \frac{0,9135 - 0,8090}{0,2} = 0,5225.$$

Evidentemente, el valor de la derivada en el punto pedido es bastante aproximado al considerado «real» o «exacto». Podemos notar que el error cometido es del orden del 0,2%. Cada una de estas aproximaciones son equivalentes a efectuar una interpolación aplicando el método de Lagrange y luego derivar el polinomio hallado. Como se tienen dos puntos, el polinomio resultante es una recta. En la figura 6.1 podemos ver las aproximaciones de la pendiente.

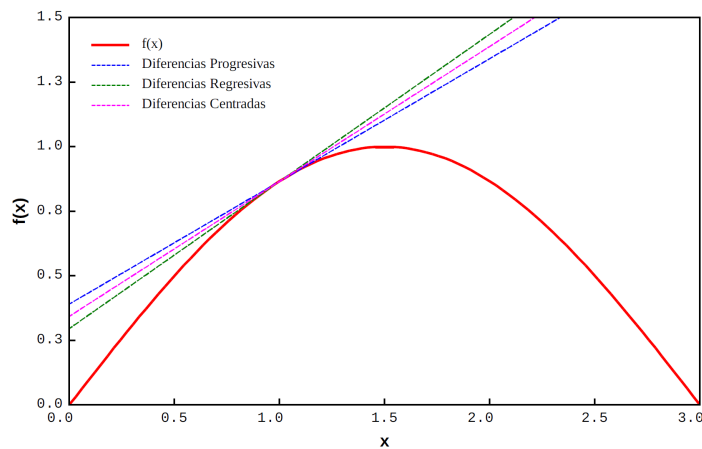


Figura 6.1: Pendiente según cada aproximación.

Hagamos una mejora escribiéndolo como:

$$f'(x) \approx \frac{f\left(x + \frac{h}{2}\right) - f\left(x - \frac{h}{2}\right)}{h}, \quad (6.7)$$

buscando mejorar la aproximación del resultado buscado. Nuevamente, si reemplazamos los valores tendremos:

$$f'(1) \approx \frac{f(1,05) - f(0,95)}{0,1} = \frac{\text{sen}\left(\frac{2\pi}{6}1,05\right) - \text{sen}\left[\frac{2\pi}{6}(0,95)\right]}{0,1}$$

$$f'(1) \approx \frac{0,8910 - 0,8387}{0,1} = 0,5230.$$

El resultado es una mejor aproximación pero no se nota una gran diferencia con respecto al anterior, puesto que el error cometido es del orden de 0,1%. Pero sin lugar a dudas, este último algoritmo es mucho mejor.

Esta forma de aproximar la derivada en un punto se conoce como *aproximación por diferencias*, y se pueden clasificar según tres tipos:

1. **Diferencias progresivas.** Cuando la derivada en un punto se aproxima según la expresión vista en primer término:

$$f'(x) \cong \frac{f(x+h) - f(x)}{h}; \quad (6.8)$$

2. **Diferencias regresivas.** Cuando la derivada en punto se aproxima según la expresión vista en segundo término:

$$f'(x) \cong \frac{f(x) - f(x-h)}{h}, \quad y \quad (6.9)$$

3. **Diferencias centradas.** Cuando la derivada en un punto se aproxima según la expresión vista en último término:

$$f'(x) \cong \frac{f(x+h) - f(x-h)}{2h}. \quad (6.10)$$

Como vimos, este último esquema es el que mejor aproxima.

Analizaremos ahora el por qué de esta mejor aproximación. Empecemos por el esquema de diferencias progresivas. Si desarrollamos por Taylor la función $f(x+h)$, tendremos que

$$f(x+h) = f(x) + f'(x)\frac{h}{1!} + f''(x)\frac{h^2}{2!} + f'''(x)\frac{h^3}{3!} + \dots, \quad (6.11)$$

de la cual podemos despejar $f'(x)$, que es

$$\begin{aligned} f'(x)h &= f(x+h) - f(x) - f''(x)\frac{h^2}{2!} - f'''(x)\frac{h^3}{3!} - \dots; \\ f'(x) &= \frac{f(x+h) - f(x)}{h} - f''(x)\frac{h}{2!} - f'''(x)\frac{h^2}{3!} - \dots \end{aligned} \quad (6.12)$$

Si nuestro h es suficientemente pequeño, entonces podemos despreciar los h^n para $n \geq 2$. Finalmente tendremos que

$$f'(x) = \frac{f(x+h) - f(x)}{h} - f''(\xi)\frac{h}{2!} = \frac{f(x+h) - f(x)}{h} + O(h), \quad (6.13)$$

con $\xi \in [x, x+h]$. En este caso, nuestra aproximación tiene un orden de convergencia $O(h)$.

De forma análoga, para el esquema de diferencias regresivas, tendremos que

$$f(x-h) = f(x) - f'(x)\frac{h}{1!} + f''(x)\frac{h^2}{2!} - f'''(x)\frac{h^3}{3!} + \dots \quad (6.14)$$

Como en el caso anterior, la expresión final es

$$f'(x) = \frac{f(x) - f(x-h)}{h} + f''(\xi)\frac{h}{2!} = \frac{f(x) - f(x-h)}{h} + O(h). \quad (6.15)$$

Al igual que en lo visto anteriormente, el orden de convergencia es $O(h)$.

Finalmente, hagamos lo mismo para el esquema de diferencias centradas. En este caso tendremos

$$f(x+h) = f(x) + f'(x)\frac{h}{1!} + f''(x)\frac{h^2}{2!} + f'''(x)\frac{h^3}{3!} + \dots \quad (6.16a)$$

$$f(x-h) = f(x) - f'(x)\frac{h}{1!} + f''(x)\frac{h^2}{2!} - f'''(x)\frac{h^3}{3!} + \dots \quad (6.16b)$$

Hagamos (6.16a) menos (6.16b):

$$f(x+h) - f(x-h) = 2f'(x)\frac{h}{1!} + 2f'''(x)\frac{h^3}{3!} + \dots \quad (6.17)$$

Observemos que a la derecha del igual hemos conseguido anular las derivadas de orden par ($f''(x)$, f^{iv} , etc.). Si despejamos $f'(x)$ de esta última expresión expresión, tenemos

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - f'''(\xi)\frac{h^2}{3!} = \frac{f(x+h) - f(x-h)}{2h} + O(h^2), \quad (6.18)$$

esta vez con $\xi \in [x - h, x + h]$.

Notemos que en este caso la convergencia es $O(h^2)$, razón por la cual la aproximación obtenida con este último método es mucho mejor respecto de los esquemas anteriores. Entonces es conveniente armar un esquema de *diferencias centradas* para aproximar una derivada en un punto dado. Además tiene otra ventaja. Como el error es proporcional a la tercera derivada, podemos obtener resultados muy precisos («exactos») para un polinomio de grado menor o igual a 2¹.

Al mismo tiempo, el hecho de que el orden de convergencia sea $O(h^2)$ nos permite inferir que si hacemos el paso (h) cada vez más chico, deberíamos tener un resultado con una mejor aproximación. Hagamos esto, y con la misma precisión del ejemplo anterior, calculemos de nuevo la derivada en el punto $x = 1$ con un nuevo paso, $h = 0,01$, para cada esquema.

$$1. \text{ Diferencias progresivas: } f'(1) = \frac{\text{sen}(\frac{\pi}{3}1,01) - \text{sen}(\frac{\pi}{3})}{0,01} = \frac{0,8712 - 0,8660}{0,01} = 0,5200$$

$$2. \text{ Diferencias regresivas: } f'(1) = \frac{\text{sen}(\frac{\pi}{3}) - \text{sen}(\frac{\pi}{3}0,99)}{0,01} = \frac{0,8660 - 0,8607}{0,01} = 0,5300$$

$$3. \text{ Diferencias centradas: } f'(1) = \frac{\text{sen}(\frac{\pi}{3}1,01) - \text{sen}(\frac{\pi}{3}0,99)}{0,02} = \frac{0,8712 - 0,8607}{0,02} = 0,5250$$

Al achicar el paso utilizado para reducir el error cometido podemos notar dos cosas. La primera es que para los esquemas progresivo y regresivo el resultado obtenido resultó ser una mejor aproximación que en el caso anterior con un paso diez veces más grande, mientras que para el esquema centrado, el resultado no fue mejor. La segunda es que hemos perdido precisión, principalmente en el esquema con diferencias centradas. La pregunta es: ¿por qué? En todo caso, ¿habremos hecho algo mal?

En realidad no hemos hecho nada incorrecto. Sucede que no hemos tomado en cuenta la incidencia del error de redondeo en nuestro algoritmo, es decir, el hecho de trabajar solamente con cuatro dígitos al representar todos los resultados. Supusimos que achicar el paso inmediatamente nos mejoraba nuestra aproximación. Pero hemos visto que la aproximación depende también de la precisión usada en los cálculos, es decir, de la representación numérica, que, como vimos, está asociada al error de redondeo².

El problema es que a medida que el paso h es cada vez más pequeño, lo mismo pasa con la operación $f(x+h) - f(x)$ o sus equivalentes. Esa diferencia se vuelve pequeña y es posible que nuestra unidad de máquina no pueda representarla correctamente. En consecuencia, debemos encontrar o desarrollar otro método para mejorar la aproximación del resultado buscado.

6.1.2. Aproximación por polinomios de Taylor

Propongamos el siguientes esquema, que se basa en tomar los intervalos $x \pm 2h$ y $x \pm h$, y el desarrollo por Taylor para cada caso:

$$f(x + 2h) = f(x) + f'(x)\frac{2h}{1!} + f''(x)\frac{4h^2}{2!} + f'''(x)\frac{8h^3}{3!} + f^{iv}(x)\frac{16h^4}{4!} + f^v(x)\frac{32h^5}{5!} + \dots; \quad (6.19)$$

$$f(x + h) = f(x) + f'(x)\frac{h}{1!} + f''(x)\frac{h^2}{2!} + f'''(x)\frac{h^3}{3!} + f^{iv}(x)\frac{h^4}{4!} + f^v(x)\frac{h^5}{5!} + \dots; \quad (6.20)$$

$$f(x - h) = f(x) - f'(x)\frac{h}{1!} + f''(x)\frac{h^2}{2!} - f'''(x)\frac{h^3}{3!} + f^{iv}(x)\frac{h^4}{4!} - f^v(x)\frac{h^5}{5!} + \dots; \quad (6.21)$$

$$f(x - 2h) = f(x) - f'(x)\frac{2h}{1!} + f''(x)\frac{4h^2}{2!} - f'''(x)\frac{8h^3}{3!} + f^{iv}(x)\frac{16h^4}{4!} - f^v(x)\frac{32h^5}{5!} + \dots \quad (6.22)$$

¹Otra forma de demostrar esta mejora del método centrado es considerar que tomamos tres puntos, $y(x-h)$, $y(x)$ y $y(x+h)$, y no dos para armar el polinomio interpolante. El resultado de esta interpolación es un polinomio de grado 2 cuyo error es proporcional a la derivada tercera de y .

²En el capítulo 1 vimos como ejemplo de la incidencia del error de redondeo en un algoritmo, el cálculo de una derivada numérica, y como, a partir de un valor del paso h , a medida que se hacía más chico, el error aumentaba.

Primero hagamos $f(x+2h) - f(x-2h)$ y $f(x+h) - f(x-h)$, con las cual obtendremos las siguientes igualdades:

$$f(x+2h) - f(x-2h) = 4f'(x)\frac{h}{1!} + 16f'''(x)\frac{h^3}{3!} + 64f^{(5)}(x)\frac{h^5}{5!} + \dots; \quad (6.23)$$

$$f(x+h) - f(x-h) = 2f'(x)\frac{h}{1!} + 2f'''(x)\frac{h^3}{3!} + 2f^{(5)}(x)\frac{h^5}{5!} + \dots \quad (6.24)$$

Si queremos mejorar la precisión de nuestros esquemas anteriores para calcular $f'(x)$, anulemos el término con h^3 . Para ello, hagamos $[f(x+2h) - f(x-2h)] - 8 \times [f(x+h) - f(x-h)]$. Así, nos queda la siguiente igualdad:

$$f(x+2h) - f(x-2h) - 8f(x+h) + 8f(x-h) = -12f'(x)h + 48f^{(5)}(x)\frac{h^5}{5!} + \dots \quad (6.25)$$

De esta última expresión podemos despejar $f'(x)$, que resulta ser:

$$f'(x) = \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h} + 4f^{(5)}(x)\frac{h^4}{5!} + \dots; \quad (6.26)$$

y si truncamos en h^4 , nos queda:

$$f'(x) = \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h} + f^{(5)}(\xi)\frac{h^4}{30}, \quad (6.27)$$

con $\xi \in [x-2h, x+2h]$ y un orden de convergencia $O(h^4)$. Con esta última expresión podemos decir que una aproximación de la primera derivada en un punto está dada por:

$$f'(x) \approx \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h}. \quad (6.28)$$

Ahora, apliquemos este nuevo esquema centrado para calcular la derivada buscada, con la misma representación numérica utilizada en los casos anteriores. Tomemos el paso $h = 0,1$ con el que obtendremos:

$$f'(x=1) = \frac{\sin\left(\frac{\pi}{3}0,8\right) - 8 \cdot \sin\left(\frac{\pi}{3}0,9\right) + 8 \cdot \sin\left(\frac{\pi}{3}1,1\right) - \sin\left(\frac{\pi}{3}1,2\right)}{12 \cdot h}$$

$$f'(x=1) = \frac{0,7431 - 6,4721 + 7,3084 - 0,9511}{12 \cdot 0,1} = \frac{0,6283}{1,2} = 0,5236.$$

El resultado obtenido es sorprendente, pues para esa representación numérica, ¡se lo puede considerar exacto! Bastó que ampliáramos el intervalo de cálculo, es decir, los puntos que usamos para armar lo que se denomina una *malla* (en inglés *mesh*), para que la aproximación sea excelente. Este algoritmo se conoce como el *método de los cinco puntos* y tiene un orden de convergencia proporcional a la derivada quinta, lo que lo vuelve muy preciso. La única desventaja es que requiere operar con cinco puntos y esa malla deberá densificarse cada vez que la representación numérica sea más *precisa*, cuidando siempre de evitar que el paso sea muy pequeño, por el riesgo de que no pueda representarse correctamente el numerador. Veremos más adelante que este tipo de mallas son muy útiles para resolver ecuaciones diferenciales y/o sistemas de ecuaciones diferenciales.

En la figura 6.2 se puede la aproximación obtenida utilizando la aproximación por polinomios de Taylor.

Con este método aproximaremos las derivadas en los extremos para el caso de los *Trazadores cúbicos con frontera sujeta* vistos en el capítulo 4. Las aproximaciones son las siguientes:

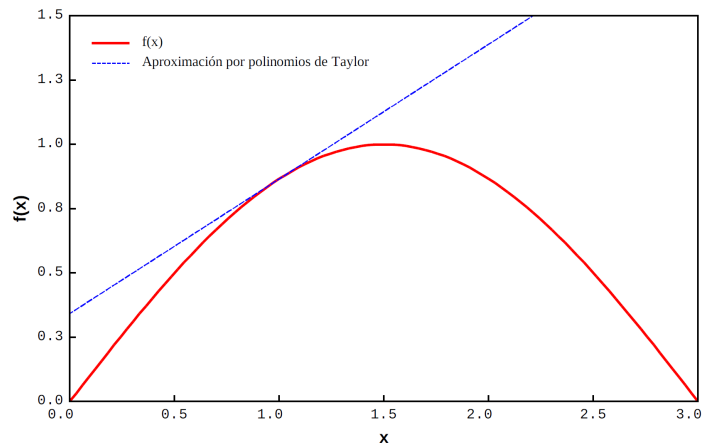


Figura 6.2: Aproximación por polinomios de Taylor.

Diferencias progresivas para

$$f'(x_0) \approx \frac{-25 y_0 + 48 y_1 - 36 y_2 + 16 y_3 - 3 y_4}{12 h}, \quad (6.29)$$

Diferencias regresivas para

$$f'(x_n) \approx \frac{25 y_n - 48 y_{n-1} + 36 y_{n-2} - 16 y_{n-3} + 3 y_{n-4}}{12 h}. \quad (6.30)$$

Estas expresiones, que tienen un orden de convergencia $O(h^4)$, resultan muy convenientes para ser incluidas en el algoritmo de los *Trazadores cúbicos con frontera sujeta*. Así, una aproximación mejor es posible sin tener los datos concretos de las derivadas primeras.

Pero nuestro interés, por ahora, es calcular en forma numérica el valor de la derivada en un punto dado con la mejor aproximación posible. ¿Existirá otra forma de obtener ese valor con un grado de aproximación similar al obtenido con el esquema anterior usando un solo punto?

6.1.3. Extrapolación de Richardson

Vimos en el punto anterior que para calcular una derivada en un punto y obtener la mejor aproximación, debemos trabajar con el esquema centrado y con un paso h pequeño, aún cuando esto trae aparejado una inestabilidad de los resultados. Tal como vimos al analizar la aproximación por polinomios de Taylor, nuestra aproximación de la derivada se puede expresar como:

$$M = N(h) + E(h), \quad (6.31)$$

donde M es el valor buscado, $N(h)$, la aproximación de M , $E(h)$, el error cometido y h , el paso. Supongamos que podemos expresar nuestra $E(h)$ de la siguiente forma:

$$E(h) = K_1 h + K_2 h^2 + K_3 h^3 + \dots \quad (6.32)$$

Análogamente al caso anterior, para un h_1 el valor buscado se podrá expresar como

$$M = N_1(h_1) + K_1 h_1 + K_2 h_1^2 + K_3 h_1^3 + \dots \quad (6.33)$$

Hagamos lo mismo pero para un h_2 tal que $q = \frac{h_1}{h_2}$. Entonces tendremos:

$$M = N_1(h_2) + K_1 h_2 + K_2 h_2^2 + K_3 h_2^3 + \dots \quad (6.34)$$

Como $h_1 = q h_2$ podemos escribir (6.33) como:

$$M = N_1(h_1) + K_1 q h_2 + K_2 (q h_2)^2 + K_3 (q h_2)^3 + \dots \quad (6.35)$$

Para mejorar el orden de aproximación de nuestro resultado, anulemos el término lineal de h , es decir, multipliquemos por q a (6.34) y luego restémosle (6.35):

$$\begin{aligned} qM - M &= qN_1(h_2) - N_1(h_1) + qK_1(h_2 - h_1) + qK_2(h_2^2 - qh_2^2) + \\ &\quad + qK_3(h_2^3 - q^2h_2^3) + \dots \\ qM - M &= qN_1(h_2) - N_1(h_1) + qK_2(h_2^2 - qh_2^2) + \\ &\quad + qK_3(h_2^3 - q^2h_2^3) + \dots, \\ (q-1)M &= qN_1(h_2) - N_1(h_2) + N_1(h_2) - N_1(h_1) + qK_2(h_2^2 - qh_2^2) + \\ &\quad + qK_3(h_2^3 - q^2h_2^3) + \dots \\ (q-1)M &= (q-1)N_1(h_2) + N_1(h_2) - N_1(h_1) + qK_2(h_2^2 - qh_2^2) + \\ &\quad + qK_3(h_2^3 - q^2h_2^3) + \dots \end{aligned} \quad (6.36)$$

Si despejamos M , tendremos la siguiente expresión:

$$M = \frac{(q-1)N_1(h_2)}{q-1} + \frac{N_1(h_2) - N_1(h_1)}{q-1} - \frac{qK_2h_2^2(q-1)}{q-1} - \frac{qK_3h_2^3(q^2-1)}{q-1} + \dots \quad (6.37)$$

en la que podemos expresar M como:

$$M = N_1(h_2) + \underbrace{\frac{N_1(h_2) - N_1(h_1)}{q-1}}_{N_2(h_1)} - qK_2h_2^2 - q(q+1)K_3h_2^3 + \dots \quad (6.38)$$

Si definimos

$$N_2(h_1) = N_1(h_2) + \frac{N_1(h_2) - N_1(h_1)}{q-1}, \quad (6.39)$$

nos queda que:

$$M = N_2(h) + K'_2 h^2 + K'_3 h^3 + \dots, \quad (6.40)$$

con

$$\begin{aligned} K'_2 &= -qK_2; \\ K'_3 &= -q(q+1)K_3; \\ &\dots \end{aligned}$$

Repitamos el proceso tomando nuevamente h_1 y h_2 . Tenemos:

$$M = N_2(h_1) + K'_2 h_1^2 + K'_3 h_1^3 + \dots, \quad (6.41)$$

$$M = N_1(h_2) + K'_2 h_2^2 + K'_3 h_2^3 + \dots \quad (6.42)$$

Al igual que en el paso anterior, impondremos que $q = \frac{h_1}{h_2}$, y reescribamos (6.41) de la siguiente forma:

$$M = N_2(h_1) + K'_2 (q h_2)^2 + K'_3 (q h_2)^3 + \dots \quad (6.43)$$

Análogamente al caso anterior, mejoremos nuestra aproximación anulando en este caso el término cuadrático de h , multiplicando por q^2 a (6.42) para luego restarle (6.43):

$$\begin{aligned} q^2M - M &= q^2N_2(h_2) - N_2(h_1) - q^2K'_2(h_2^2 - h_2^2) + q^2K'_3h_2^3(q-1) + \dots \\ q^2M - M &= q^2N_2(h_2) - N_2(h_1) + q^2K'_3h_2^3(q-1) + \dots \end{aligned} \quad (6.44)$$

De la misma forma que para el caso anterior, obtenemos una nueva aproximación para M :

$$M = \frac{(q^2 - 1)N_2(h_2)}{q^2 - 1} + \frac{N_2(h_2) - N_2(h_1)}{q^2 - 1} + \frac{q^2 K'_3 h_2^3 (q - 1)}{q^2 - 1} + \dots$$

$$M = \underbrace{N_2(h_2) + \frac{N_2(h_2) - N_2(h_1)}{q^2 - 1}}_{N_3(h_1)} + \frac{q^2 K'_3 h_2^3}{q + 1} + \dots \quad (6.45)$$

Una segunda forma de escribir esto último en función de h_1 es

$$M = \underbrace{N_2\left(\frac{h_1}{q}\right) + \frac{N_2\left(\frac{h_1}{q}\right) - N_2(h_1)}{q^2 - 1}}_{N_3(h_1)} + \frac{q^2 K'_3 \left(\frac{h_1}{q}\right)^3}{q + 1} + \dots, \quad (6.46)$$

por lo que una nueva aproximación de M es

$$M = N_3(h_1) + K''_3 h_1^3 + \dots; \quad (6.47)$$

con

$$K''_3 = \frac{K'_3}{q(q+1)}; \dots,$$

que resulta mejor que la anterior.

Finalmente, podemos generalizar el método de aproximación de la siguiente forma:

$$N_j(h) = N_{j-1}\left(\frac{h}{q}\right) + \frac{N_{j-1}\left(\frac{h}{q}\right) - N_{j-1}(h)}{q^{j-1} - 1}. \quad (6.48)$$

Este método o algoritmo para mejorar una aproximación se conoce como *Extrapolación de Richardson*.³

Un caso particular muy usado es cuando $q = 2$, cuya expresión general se define como:

$$N_j(h) = N_{j-1}\left(\frac{h}{2}\right) + \frac{N_{j-1}\left(\frac{h}{2}\right) - N_{j-1}(h)}{2^{j-1} - 1}. \quad (6.49)$$

Este algoritmo permite aproximar una derivada numérica con poco esfuerzo y teniendo en cuenta la inestabilidad del algoritmo porque no requiere dividir por números excesivamente pequeños.

Aplicemos este método al ejemplo inicial y calculemos la derivada de $f(x) = \sin\left(\frac{\pi}{3}x\right)$ en $x = 1$ con el algoritmo de diferencias progresivas.

Armemos una tabla para aplicar el algoritmo anterior de modo de visualizar fácilmente cada uno de los pasos. En primer lugar, vamos definir que la primera aproximación, es decir, $N_1(h)$ sea la derivada calculada numéricamente con h , que ocupará la primera columna. Usaremos la expresión:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} = \frac{\sin\left[\frac{\pi}{3}(x+h)\right] - \sin\left[\frac{\pi}{3}x\right]}{h}.$$

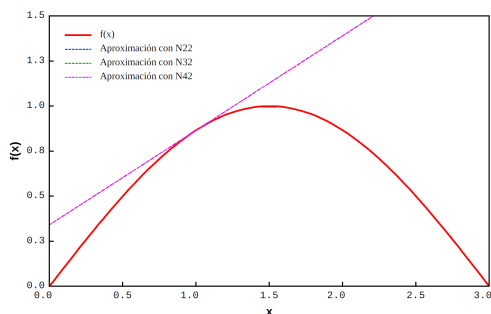
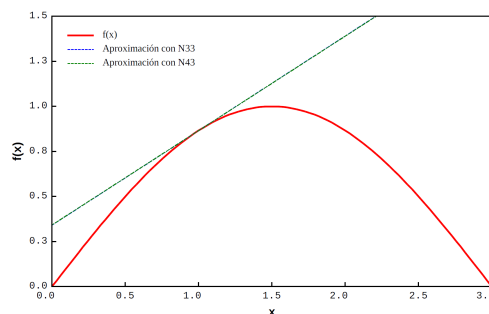
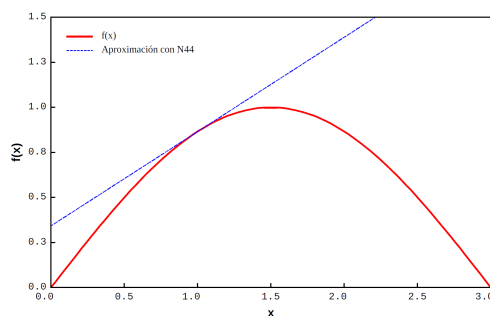
Las demás columnas serán $N_2(h)$, $N_3(h)$ y $N_4(h)$. En segundo lugar, tomaremos varios valores de h , por lo tanto tendremos varias filas con diferentes aproximaciones de la derivada buscada. Para cada caso calcularemos las aproximaciones con la fórmula de la *Extrapolación de Richardson*:

$$N_j(h) = N_{j-1}\left(\frac{h}{2}\right) + \frac{N_{j-1}\left(\frac{h}{2}\right) - N_{j-1}(h)}{2^{j-1} - 1}.$$

En la tabla 6.1 podemos ver los resultados obtenidos al aplicar la extrapolación de Richardson a nuestro ejemplo.

Tabla 6.1: *Extrapolación de Richardson*

j	h_j	$y'_j = N_{j,1}$	$N_{j,2}$	$N_{j,3}$	$N_{j,4}$
1	0,2	0,4250			
2	0,1	0,4750	0,5250		
3	0,05	0,5000	0,5250	0,5250	
4	0,025	0,5120	0,5240	0,5237	0,5235

(a) *Aproximación con N_{j2}* (b) *Aproximación con N_{j3}* (c) *Aproximación con N_{44}* Figura 6.3: *Aproximaciones con Extrapolación de Richardson.*

Analicemos rápidamente los resultados obtenidos. La primera columna contiene los resultados de aproximar la derivada con varios h diferentes. Vemos que a pesar de utilizar un h relativamente pequeño ($h = 0,025$) nuestra aproximación inicial no es muy buena.

La segunda columna es nuestra primera aplicación de la extrapolación de Richardson, usando los valores de la primera columna. A primera vista se puede observar que la aproximación es muy superior a la anterior. Algo similar ocurre en la tercera. Finalmente, en la cuarta, la aproximación final resulta ser casi el valor «exacto» para una representación de cuatro (4) decimales. Y si comparamos con la aproximación para $h = 0,025$, última fila de la primera columna, vemos que es muy superior. Si quisiéramos obtener una aproximación similar, deberíamos trabajar con más decimales, puesto que para $h = 0,01$ el valor de $f'(1)$ es 0,5200, que si bien tiene dos decimales correcto, es menos preciso que el hallado con Richardson.

En las figuras 6.3 hemos representado las aproximaciones de la pendiente en el punto dado aplicando la *Extrapolación de Richardson*.

³Veremos más adelante una aplicación de este mismo método asociado a la integración numérica.

6.1.4. Notas finales

Es evidente que la diferenciación numérica es inestable o, dicho de otro modo, es muy dependiente de la precisión utilizada. Afinar el paso h en un algoritmo dado puede conducir a resultados de menor precisión o, en términos numéricos, inservibles; en consecuencia, no es conveniente reducir el paso h suponiendo que eso mejora la aproximación buscada.

Los distintos métodos vistos en los puntos anteriores indican que es preferible mejorar el algoritmo o desarrollar uno nuevo, antes que afinar el paso de cálculo. Más aún, es mucho más efectivo aplicar el método de *Extrapolación de Richardson* a un algoritmo conocido y sencillo que desarrollar uno nuevo. En todo caso, la segunda opción sería utilizar los polinomios de Taylor o alguna aproximación polinomial que utilice la información disponible (puntos adyacentes o aledaños). Si bien esta aproximación puede ser laboriosa, queda ampliamente justificada al disminuir la incidencia del error de redondeo en los cálculos, sobre todo al no tener que dividir por un número «muy pequeño».

Los desarrollos vistos para el caso de aproximar una primera derivada pueden extrapolarse para derivadas de orden superior. Un ejemplo de ello es la aproximación centrada de la segunda derivada en un punto dado, cuya expresión es:

$$f''(x) = \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}. \quad (6.50)$$

que se obtiene de considerar los polinomios de Taylor para $x-h$ y $x+h$. Efectivamente, al desarrollar ambos polinomios tendremos

$$f(x+h) = f(x) + f'(x) \frac{h}{1!} + f''(x) \frac{h^2}{2!} + f'''(x) \frac{h^3}{3!} + \dots; \quad (6.51)$$

$$f(x-h) = f(x) - f'(x) \frac{h}{1!} + f''(x) \frac{h^2}{2!} - f'''(x) \frac{h^3}{3!} + \dots \quad (6.52)$$

Si sumamos ambos polinomios obtenemos:

$$\begin{aligned} f(x+h) + f(x-h) &= 2f(x) + f''(x)h^2 + f^{iv}(x) \frac{h^4}{12} + \dots; \\ f''(x) &= \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} - f^{iv}[\xi] \frac{h^2}{12}; \\ f''(x) &= \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} + O(h^2), \end{aligned} \quad (6.53)$$

con $\xi \in [x-h, x+h]$.

Observemos que el error cometido al calcular la derivada segunda con la expresión dada es proporcional a h^2 y a $f^{iv}(\xi)$, es decir, similar al caso de la expresión centrada para la primera derivada. Podemos asegurar que en el caso de polinomios de grado 3 o inferior, o que no exista la derivada cuarta, la derivada segunda obtenida en forma numérica, es «exacta».⁴

Mediante razonamientos análogos o similares pueden obtenerse algoritmos para calcular derivadas numéricas de orden superior. Sin embargo, a medida que aumenta el orden de la derivada, aumenta el exponente de h , y con ello, la inestabilidad del algoritmo. Calcular una derivada numérica de mayor orden puede llevar a que obtengamos valores poco satisfactorios o directamente inútiles, si el paso h elegido se demasiado pequeño. Algo de esto se verá más adelante cuando tratemos la resolución de ecuaciones diferenciales.

6.2. Integración numérica

Como en el caso de la diferenciación numérica, la integración numérica tiene la misma dificultad de trabajar con métodos simbólicos. Existen muchos programas de aplicación en la

⁴Esta aproximación de la derivada segunda la usaremos más adelante.

ingeniería que dependen de obtener integrales definidas. Como es prácticamente imposible agregar una base de datos que incluya las primitivas de cualquier función, la única manera de calcular estas integrales es mediante métodos numéricos. Un ejemplo en este sentido es la utilización del método de los elementos finitos en el análisis estructural, que calcula la matriz de rigidez mediante la integración numérica.

Veremos a continuación varios métodos numéricos para calcular integrales definidas, analizando ventajas y desventajas de cada uno de ellos.

6.2.1. Fórmulas de Newton-Cotes

Antes de desarrollar las distintas fórmulas o métodos para obtener una integral definida en forma numérica, daremos algunas definiciones.

Definición 6.1. Dada una función $f(x)$ definida en $[a, b]$, se denomina *cuadratura numérica* de la integral $I(f) = \int_a^b f(x)dx$ a una fórmula tal que:

$$Q_n(f) = \sum_{i=1}^n c_i f(x_i); \quad (6.54)$$

con $c_i \in \mathfrak{R}$ y $x_i \in [a, b]$. Los puntos x_i se denominan *puntos de cuadratura* (o raíces) y los valores c_i , coeficientes de *cuadratura o de peso*. Asimismo, se define el error de la cuadratura como $E_n(f) = I(f) - Q_n(f)$.

Definición 6.2. Una cuadratura numérica tiene grado de precisión m si $E_n(x^k) = 0$ para $k = 0; 1; \dots; m$ y $E_n(x^{m+1}) \neq 0$.

Observación 6.2.1. Si una cuadratura numérica tiene grado de precisión m , entonces $E_n(p_k) = 0$ para todo polinomio $p_k(x)$ de grado menor o igual a m ($k \leq m$).

Definición 6.3. Se denomina *fórmula cerrada* de Newton-Cotes a toda cuadratura numérica cuyos nodos incluya a los extremos del intervalo.

Definición 6.4. Se denomina *fórmula abierta* de Newton-Cotes a toda cuadratura numérica cuyos nodos no incluya a los extremos del intervalo.

6.2.2. Fórmulas cerradas de Newton-Cotes

Fórmulas simples

Supongamos que tenemos la siguiente función (o curva) y queremos hallar el área bajo la misma en el intervalo $[a, b]$, como vemos en la figura 6.4.

Para empezar, podemos hacer dos aproximaciones muy simples como las de las figuras 6.5(a) y 6.5(b).

En la aproximación de la figura 6.5(a), vemos que el área obtenida es menor que el área buscada. En cambio, en la 6.5(b), la aproximación obtenida del área es mayor que el área buscada.

Estas dos aproximaciones podemos expresarlas matemáticamente como:

$$Q_n(f) = f(a)(b - a); \quad (6.55)$$

para el caso (a) y,

$$Q_n(f) = f(b)(b - a); \quad (6.56)$$

para el caso (b).

Otra forma de aproximar el área la vemos en la figura 6.6.

Al tomar un segmento recto que une $f(a)$ y $f(b)$ la aproximación del área bajo la curva es algo mejor que la vista en los casos anterior, pero sin llegar a encontrar el valor buscado. En la figura 6.6 queda evidente que en ese caso la aproximación es por defecto.

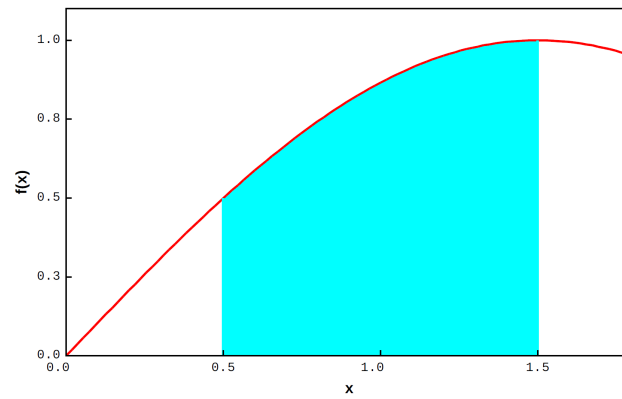


Figura 6.4: Área bajo la curva.

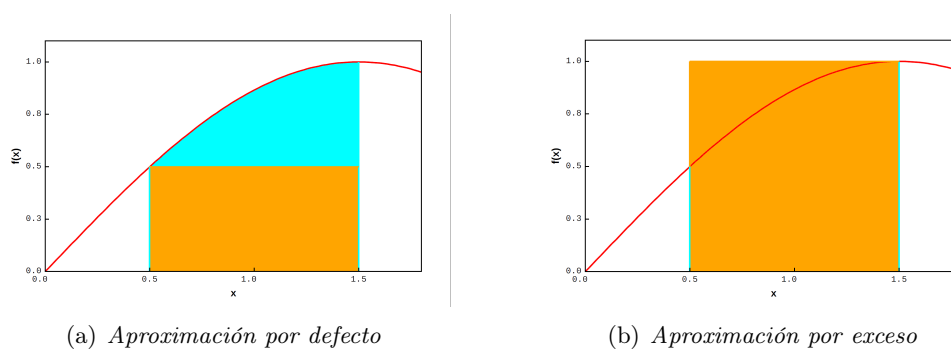


Figura 6.5: Aproximación por rectángulos.

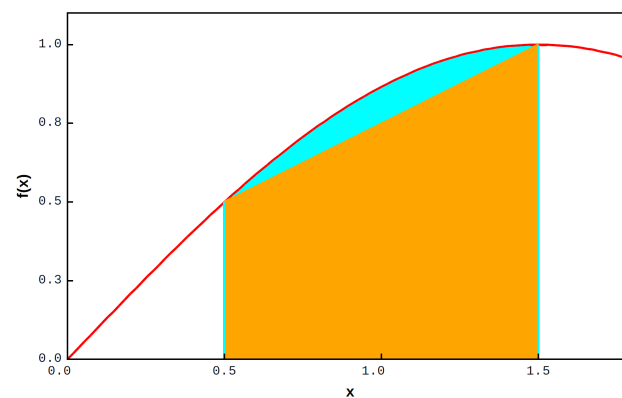


Figura 6.6: Aproximación por trapecios.

La expresión matemática para este caso es:

$$Q_n(f) = \frac{f(b) + f(a)}{2}(b - a). \quad (6.57)$$

Generalicemos estas tres expresiones. Definamos $h = b - a$, y escribamos cada una de las expresiones de la siguiente forma:

- Aproximación por rectángulos (defecto): $Q_n(f) = h \cdot f(a)$;
- Aproximación por rectángulos (exceso): $Q_n(f) = h \cdot f(b)$;
- Aproximación por trapecio: $Q_n(f) = \frac{h}{2} \cdot [f(a) + f(b)]$.

Para establecer si las aproximaciones anteriores son buenas, estimemos el error de cada una de ellas. Primeramente, analicemos cualquiera de los dos métodos que aproximan con un rectángulo. Al desarrollar $f(x)$ respecto del punto a mediante una serie de Taylor, tenemos que

$$f(x) = f(a) + f'(a)(x-a) + f''(a)\frac{(x-a)^2}{2} + \dots \quad (6.58)$$

Para obtener la integral basta con integrar la serie también. Entonces tenemos que

$$\int_a^b f(x)dx = \int_a^b f(a)dx + \int_a^b f'(a)(x-a)dx + \int_a^b f''(a)\frac{(x-a)^2}{2}dx + \dots \quad (6.59)$$

Si integramos y truncamos en el término de la derivada primera, nos queda

$$\int_a^b f(x)dx = f(a)(b-a) + f'(\xi)\frac{(b-a)^2}{2}, \quad (6.60)$$

con $\xi \in [a, b]$. Como $h = b - a$ nos queda

$$\int_a^b f(x)dx = f(a)h + f'(\xi)\frac{h^2}{2} = h \cdot f(a) + \frac{b-a}{2} \underbrace{f'(\xi)}_{O(h)} \cdot h, \quad (6.61)$$

es decir, nuestra expresión tiene un error proporcional a la derivada primera y su orden de convergencia es $O(h)$. Para el caso de usar $f(b)$ el error es análogo.

Ahora analicemos el método del trapecio. Utilicemos una interpolación entre el punto a y b aplicando el *Método de Lagrange*, con su error ⁵. En este caso tenemos que

$$f(x) = f(a)\frac{x-b}{a-b} + f(b)\frac{x-a}{b-a} + f''(\xi)\frac{(x-a)(x-b)}{2}. \quad (6.62)$$

Integremos el polinomio obtenido; así nos queda

$$\begin{aligned} \int_a^b f(x)dx &= \frac{f(a)}{a-b} \int_a^b (x-b)dx + \frac{f(b)}{b-a} \int_a^b (x-a)dx + f''(\xi) \int_a^b \frac{(x-a)(x-b)}{2}dx \\ &= \frac{f(a) + f(b)}{2} (b-a) - f''(\xi) \frac{(b-a)^3}{12} \\ &= \frac{f(a) + f(b)}{2} h - f''(\xi) \frac{h^3}{12} \\ \int_a^b f(x)dx &= \frac{f(a) + f(b)}{2} h - \frac{b-a}{2} f''(\xi) \frac{h^2}{6}, \end{aligned} \quad (6.63)$$

nuevamente con $\xi \in [a, b]$. Lo que hemos obtenido es un método cuyo error es proporcional a la derivada segunda. Hemos mejorado nuestra aproximación, incluso porque ahora ese mismo error es proporcional a h^2 ($O(h^2)$).

Analícemos ahora una segunda mejora. Supongamos que podemos calcular la función en $x = \frac{a+b}{2}$, es decir, podemos obtener $f\left(\frac{a+b}{2}\right)$. En consecuencia, tenemos ahora tres puntos que nos pueden servir para obtener el área buscada. Hagamos pasar una curva por esos tres puntos utilizando el polinomio de Taylor y definamos en este caso que $h = \frac{b-a}{2}$.⁶

⁵Otras formas son: usar el *Método de Newton* para interpolar o truncar el desarrollo por Taylor del rectángulo en el término que contiene a $f''(a)$.

⁶También podemos obtener esta curva mediante una interpolación polinómica que pase por los tres puntos.

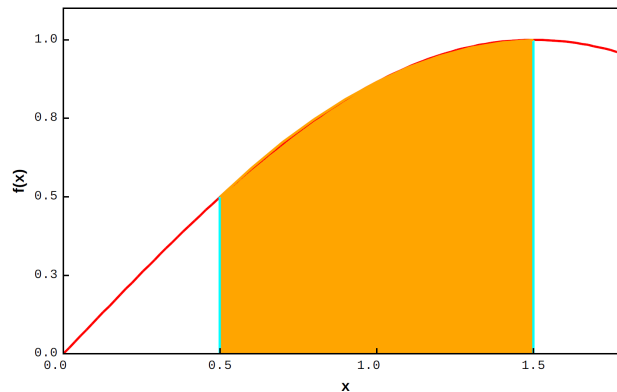


Figura 6.7: Aproximación por arcos de parábola cuadrática.

Podemos ver en la figura 6.7 que el área aproximada por esta curva que pasa por tres puntos se acerca al área buscada; es una muy buena aproximación.

Una curva que pasa por tres puntos es una parábola de segundo grado, y el método que la aplica es conocido como *Método de Simpson*, cuya expresión matemática es:

$$Q_n(f) = \frac{h}{3} \left[f(a) + f(b) + 4 \cdot f\left(\frac{a+b}{2}\right) \right]. \quad (6.64)$$

Analicemos el error cometido con esta nueva expresión. Tomemos nuevamente nuestro desarrollo de Taylor pero a partir del punto $c = \frac{a+b}{2}$ y cortemos la expresión en la derivada cuarta. Entonces nos queda

$$f(x) = f(c) + f'(c)(x-c) + f''(c)\frac{(x-c)^2}{2} + f'''(c)\frac{(x-c)^3}{6} + f^{iv}(\xi_1)\frac{(x-c)^4}{24}, \quad (6.65)$$

con $\xi_1 \in [a, b]$.

Si integramos nuevamente nos queda

$$\begin{aligned} \int_a^b f(x)dx &= \int_a^b f(c)dx + \int_a^b f'(c)(x-c)dx + \int_a^b f''(c)\frac{(x-c)^2}{2}dx + \\ &+ \int_a^b f'''(c)\frac{(x-c)^3}{6}dx + \int_a^b f^{iv}(c)\frac{(x-c)^4}{24}dx \\ &= f(c)(b-a) + f'(c)\frac{(x-c)^2}{2}\Big|_a^b + f''(c)\frac{(x-c)^3}{6}\Big|_a^b + \\ &+ f'''(c)\frac{(x-c)^4}{24}\Big|_a^b + f^{iv}(\xi_1)\frac{(x-c)^5}{120}\Big|_a^b \end{aligned} \quad (6.66)$$

Ahora tomemos que $h = b - x_1 = c - a$. Entonces nos queda

$$\int_a^b f(x)dx = f(c)2h + f''(c)\frac{h^3}{3} + f^{iv}(\xi_1)\frac{h^5}{60}. \quad (6.67)$$

Aproximemos la derivada segunda en c mediante una derivada discreta, como la vista en diferenciación numérica. Dado que esta aproximación debe tener un error de truncamiento similar a nuestra aproximación de la integral, usaremos la derivada por diferencias centradas, que también es proporcional a $f^{iv}(\xi_2)$:

$$f''(c) = \frac{f(a) - 2f(x_1) + f(b)}{h^2} - \frac{h^2}{12}f^{iv}(\xi_2), \quad (6.68)$$

con $\xi_2 \in [a, b]$.

Al reemplazarla en la fórmula de integración, nos queda

$$\begin{aligned} \int_a^b f(x)dx &= f(c)2h + \frac{f(a) - 2f(c) + f(b)}{h^2} \frac{h^3}{3} - f^{iv}(\xi_2) \frac{h^5}{36} + f^{iv}(\xi_1) \frac{h^5}{60} \\ &= \frac{h}{3} [f(a) + 4f(c) + f(b)] - f^{iv}(\xi) \frac{h^5}{90} \\ &= \frac{h}{3} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right] - \frac{b-a}{2} \frac{f^{iv}(\xi)}{90} h^4, \end{aligned} \quad (6.69)$$

con $\xi \in [a, b]$. Usualmente el término de error se define como:

$$E(h) = M h^4, \text{ con } M = \frac{b-a}{180} f^{iv}(\xi), \quad (6.70)$$

de ahí que el orden de convergencia sea $O(h^4)$.

Vemos que el error del *Método de Simpson* es proporcional a la derivada cuarta, por lo tanto, esta expresión nos da una integral «exacta» para polinomios de grado menor o igual a tres.

Unifiquemos los cuatro casos en un intervalo de integración. Si tomamos como intervalo $[a, b]$ el intervalo $[-1; 1]$, nos queda para cada método lo siguiente:

- Aproximación por rectángulo (defecto): $Q_n(x) = 2 \cdot f(-1)$.
- Aproximación por rectángulo (exceso): $Q_n(x) = 2 \cdot f(1)$.
- Aproximación por trapecios: $Q_n(x) = 1 \cdot f(-1) + 1 \cdot f(1)$.
- Aproximación por Simpson: $Q_n(x) = \frac{1}{3} \cdot f(-1) + \frac{4}{3} \cdot f(0) + \frac{1}{3} \cdot f(1)$.

Si nos fijamos en la definición de cuadratura podemos ver que hemos definido para cada caso un valor de c_i y un valor de x_i , que son los siguientes:

- Aproximación por rectángulo (defecto): $c_1 = 2, x_1 = -1$.
- Aproximación por rectángulo (exceso): $c_1 = 2, x_1 = 1$.
- Aproximación por trapecios: $c_1 = c_2 = 1, x_1 = -1, x_2 = 1$.
- Aproximación por Simpson: $c_1 = c_3 = \frac{1}{3}, c_2 = \frac{4}{3}, x_1 = -1, x_2 = 0, x_3 = 1$;

con lo cual podemos escribirlos según la forma general definida como *cuadratura numérica*:

$$Q_n(f) = \sum_{i=1}^n c_i f(x_i); \quad (6.71)$$

siendo $n = 1$ para la fórmula del *rectángulo*, $n = 2$ para la del *trapecio* y $n = 3$ para la de *Simpson*.

En los métodos vistos anteriormente, los datos que usábamos para aproximar la integral se apoyaban solamente en la función a integrar. Pero si conocemos la función, podemos conocer también su derivada primera. ¿Qué pasa si tomamos como datos para aproximar la integral definida también las derivadas primeras de la función en los puntos a y b ? Por lo pronto, y de acuerdo con lo ya visto, es posible armar una función polinómica de tercer grado, gracias a la

interpolación por el *Método de Hermite*: $P(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$. Una forma más sencilla de escribirla es mediante el *Método de Newton* adaptado al *Método de Hermite*:

$$f(x) \approx P(x) = f(a) + f'(a)(x-a) + \frac{\frac{f(b)-f(a)}{b-a} - f'(a)}{b-a}(x-a)^2 + \frac{f'(b) - 2\frac{f(b)-f(a)}{b-a} + f'(a)}{(b-a)^2}(x-a)^2(x-b). \quad (6.72)$$

Al igual que en los casos anteriores, aproximemos la integral $\int_a^b f(x)dx$ con $\int_a^b P(x)dx$:

$$\int_a^b P(x)dx = \int_a^b f(a)dx + \int_a^b f'(a)(x-a)dx + \int_a^b \frac{\frac{f(b)-f(a)}{b-a} - f'(a)}{b-a}(x-a)^2dx + \int_a^b \frac{f'(b) - 2\frac{f(b)-f(a)}{b-a} + f'(a)}{(b-a)^2}(x-a)^2(x-b)dx. \quad (6.73)$$

Al resolver cada uno de los términos de la derecha obtenemos lo siguiente:

$$\int_a^b P(x)dx = f(a)(b-a) + f'(a)\frac{(b-a)^2}{2} + [f(b) - f(a)]\frac{b-a}{3} - f'(a)\frac{(b-a)^2}{3} + f'(b)\frac{(b-a)^2}{12} + [f(b) - f(a)]\frac{b-a}{6} - f'(a)\frac{(b-a)^2}{12}. \quad (6.74)$$

Si reagrupamos todos los términos, la expresión queda:

$$\int_a^b f(x)dx = \frac{b-a}{2} [f(a) + f(b)] + \frac{(b-a)^2}{12} [f'(a) - f'(b)], \quad (6.75)$$

que se conoce como *Método del Trapecio Mejorado*. Este método tiene el mismo orden de convergencia que el *Método de Simpson* ($E(h) = (b-a)\frac{f^{iv}(\xi)h^4}{720}$), pero al necesitar las derivadas primeras en los puntos extremos, no es muy usado.

Aún cuando estas aproximaciones tienen una precisión interesante (sobre todo la última), no siempre son lo suficientemente precisas para resolver cualquier problema. Para mejorar nuestra aproximación, veremos a continuación algunas formas de mejorar la precisión de las cuadraturas.

Fórmulas compuestas

Supongamos que en lugar de utilizar la fórmula del rectángulo con el paso $h = b - a$, dividimos ese intervalo en intervalos más pequeños. Empecemos por definir un nuevo paso más chico, tomando $h = \frac{b-a}{2}$. Ahora podemos aproximar la integral con dos subintervalos, tanto por defecto como por exceso, que resultan ser $[a, a+h]$ y $[a+h, b]$, con los cuales se obtienen las siguientes aproximaciones:

$$Q_n(f) = h \cdot f(a) + h \cdot f(a+h);$$

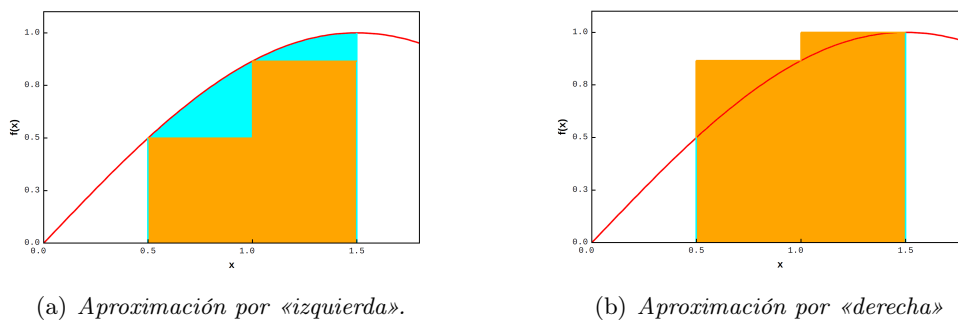
o

$$Q_n(f) = h \cdot f(a+h) + h \cdot f(b).$$

Ambas aproximaciones las podemos ver en la figura 6.8. La primera (6.8(a)) es una aproximación francamente por defecto, en cambio, en la segunda (6.8(b)) tenemos una aproximación por exceso.

Si hacemos un desarrollo similar con la fórmula del trapecio, tomando el mismo paso ($h = \frac{b-a}{2}$), y por ende, los mismos subintervalos, tendremos:

$$Q_n(f) = \frac{h}{2} [f(a) + f(a+h)] + \frac{h}{2} [f(a+h) + f(b)] = \frac{h}{2} \left[f(a) + 2f\left(\frac{a+b}{2}\right) + f(b) \right]. \quad (6.76)$$



(a) Aproximación por «izquierda».

(b) Aproximación por «derecha»

Figura 6.8: Aproximación compuesta por rectángulos.

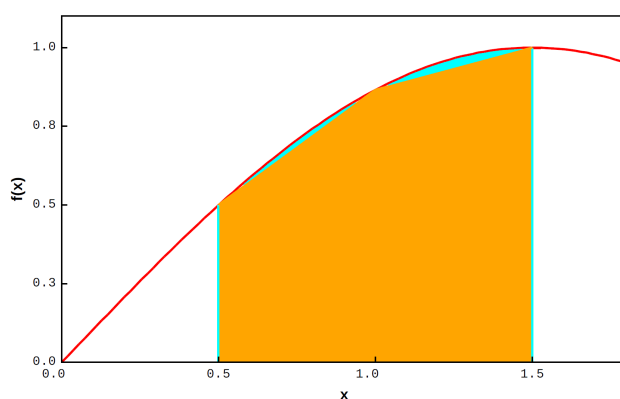


Figura 6.9: Aproximación compuesta por trapecios.

La aproximación obtenida la podemos ver en el figura 6.9.

Al igual que en los casos anteriores, podemos mejorar la aproximación del *Método de Simpson* usando la misma técnica. Si dividimos nuestro intervalo inicial en dos, de manera de trabajar con dos subintervalos y definimos $h = \frac{b-a}{4}$, tendremos la nueva aproximación:

$$Q_n(f) = \frac{h}{3} [f(a) + f(a+2h) + 4 \cdot f(a+h)] + \frac{h}{3} [f(a+2h) + f(b) + 4 \cdot f(a+3h)]. \quad (6.77)$$

Podemos simplificar la expresión para que nos quede una más general:

$$Q_n(f) = \frac{h}{3} [f(a) + f(b) + 2 \cdot f(a+2h) + 4 \cdot f(a+h) + 4 \cdot f(a+3h)]. \quad (6.78)$$

El resultado de aplicar esta fórmula, como se puede ver en la figura 6.10, muestra que la aproximación obtenida es muy precisa, y que el resultado es muy cercano al «exacto».

Podemos generalizar las expresiones de los métodos para n subintervalos:

- Rectángulos:

$$Q_n(f) = h \cdot \sum_{i=0}^{n-1} f(a + i \cdot h), \quad (6.79)$$

$$Q_n(f) = h \cdot \left[\sum_{i=1}^{n-1} f(a + i \cdot h) + f(b) \right], \quad (6.80)$$

con $h = \frac{b-a}{n}$;

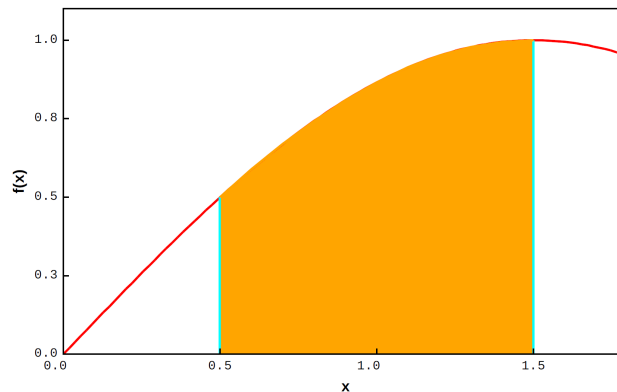


Figura 6.10: Aproximación compuesta por Simpson.

- Trapecios:

$$Q_n(f) = \frac{h}{2} \left[f(a) + f(b) + 2 \cdot \sum_{i=1}^{n-1} f(a + i \cdot h) \right], \quad (6.81)$$

también con $h = \frac{b-a}{n}$;

- Trapecios Mejorado:

$$Q_n(f) = \frac{h}{2} \left[f(a) + f(b) + 2 \cdot \sum_{i=1}^{n-1} f(a + i \cdot h) \right] + \frac{h^2}{12} [f'(a) - f'(b)], \quad (6.82)$$

con el mismo h del método anterior;

- Simpson:

$$Q_n(f) = \frac{h}{3} \left\{ f(a) + f(b) + 2 \cdot \sum_{i=1}^{n-1} f(a + 2i \cdot h) + 4 \cdot \sum_{i=1}^n f[a + (2i - 1) \cdot h] \right\} \quad (6.83)$$

con $h = \frac{b-a}{2n}$ y $n = 1; 2; 3; \dots; k$.

Estas fórmulas permiten mejorar la precisión reduciendo el paso h . En particular, en el caso del *Método Compuesto de Simpson*, el error se define como

$$E(h) = \frac{b-a}{180} f^{iv}(\mu) h^4 = M h^4, \quad \text{con } M = \frac{b-a}{180} f^{iv}(\mu), \quad (6.84)$$

con $\mu \in [a, b]$. Si bien se trata de una mejora en la precisión, el orden de convergencia sigue siendo $O(h^4)$.

Sin embargo, esta metodología tiene una desventaja. A medida que achicamos el paso aumentamos notablemente la cantidad de operaciones que debemos realizar, lo que significa más tiempo de procesamiento. Esto no siempre es práctico: por ejemplo, dividir el intervalo en 100 subintervalos para aplicar el *Método Compuesto de Simpson* representa un esfuerzo de cálculo que no siempre mejora la precisión del resultado en el mismo sentido. Puede ocurrir que nuestra representación numérica nos limite el tamaño del paso h , lo que nos impide «afinar» el paso todo lo necesario. Algo similar puede ocurrir con las otras fórmulas.

Por otro lado, toda vez que querramos afinar nuestro cálculo reduciendo el paso h , debemos calcular prácticamente todo otra vez, pues salvo los valores de la función en los extremos

del intervalo, el resto de los valores no suelen ser útiles (salvo excepciones). Cambiar el paso no suele tener «costo cero». Busquemos, en consecuencia, otra forma para obtener resultados más precisos sin tener achicar el paso, incrementar demasiado las cantidad de operaciones a realizar o repetir todos los cálculos.

Método de Romberg

Como primer paso para desarrollar un método más eficiente que mejore nuestros resultados, analicemos el error que se comete al aplicar cualquiera de las fórmulas de cuadratura vistas en los puntos anteriores. En forma general, la aproximación la podemos expresar de la siguiente forma:

$$\begin{aligned} I(f) &= \int_a^b f(x)dx = \int_a^b \sum_{i=1}^n f(x_i)L_i(x)dx + \int_a^b \frac{f^{(n)}[\xi(x)]}{n!} \prod_{i=1}^n (x - x_i)dx \\ &= \underbrace{\sum_{i=1}^n c_i f(x_i)}_{Q_n(f)} + \frac{1}{n!} \int_a^b f^{(n)}[\xi(x)] \prod_{i=1}^n (x - x_i)dx; \end{aligned} \quad (6.85)$$

y, como vimos al principio, el error está dado por:

$$E_n(f) = I(f) - Q_n(f) = \frac{1}{n!} \int_a^b f^{(n)}[\xi(x)] \prod_{i=1}^n (x - x_i)dx. \quad (6.86)$$

Para cada uno de los métodos tenemos:

Rectángulos: $E_1(f) = \frac{b-a}{2} \cdot f'(\xi) h.$

Trapecios: $E_2(f) = -\frac{b-a}{12} \cdot f''(\xi) h^2.$

Simpson: $E_3(f) = -\frac{b-a}{90} f^{iv}(\xi) h^4.$

Notemos que las aproximaciones mediante cualquiera de las fórmulas vistas se pueden expresar como:

$$\begin{aligned} M &= N(h) + K_1 \cdot h + K_2 \cdot h^2 + K_3 \cdot h^3 + \dots \\ M - N(h) &= E(h) = K_1 \cdot h + K_2 \cdot h^2 + K_3 \cdot h^3 + \dots \end{aligned} \quad (6.87)$$

lo que nos permite aplicar el método de *Extrapolación de Richardson*, visto para diferenciación numérica. En el caso particular del método compuesto del trapecio, el error puede expresarse mediante potencias pares de h ⁷:

$$E(h) = K_1 \cdot h^2 + K_2 \cdot h^4 + \dots + K_s \cdot h^{2s} + O(h^{2s+1}). \quad (6.88)$$

Esto nos induce a generar una adaptación de este método para la integración, que se conoce como *Método de Romberg*. El desarrollo para obtenerlo es el siguiente. Partamos de la fórmula compuesta del trapecio:

$$Q_n(f) = \frac{h}{2} \left[f(a) + f(b) + 2 \cdot \sum_{i=1}^{n-1} f(a + i \cdot h) \right]; \quad (6.89)$$

⁷El desarrollo algebraico de este error es bastante laborioso.

y de acuerdo con lo visto, definamos que:

$$I(f) = \frac{h}{2} \left[f(a) + f(b) + 2 \cdot \sum_{i=1}^{n-1} f(a + i \cdot h) \right] - \frac{b-a}{12} h^2 f''(\xi); \quad (6.90)$$

con $a < \xi < b$ y $h = \frac{b-a}{n}$.

Para empezar, obtengamos todas las aproximaciones para $m_1 = 1, m_2 = 2, m_3 = 4, \dots, m_n = 2^{n-1}$, con n positivo. En consecuencia, tendremos un h_k para cada valor de m_k que estará definido como $h_k = \frac{b-a}{m_k} = \frac{b-a}{2^{k-1}}$. De esta forma podemos expresar la regla del trapecio como:

$$I(f) = \frac{h_k}{2} \left[f(a) + f(b) + 2 \cdot \sum_{i=1}^{2^{k-1}-1} f(a + i \cdot h_k) \right] - \frac{b-a}{12} h_k^2 f''(\xi_k). \quad (6.91)$$

Definamos ahora que:

$$R_{k,1}(h_k) = \frac{h_k}{2} \left[f(a) + f(b) + 2 \cdot \sum_{i=1}^{2^{k-1}-1} f(a + i \cdot h_k) \right]; \quad (6.92)$$

y con esta nueva fórmula obtengamos los distintos $R_{k,1}$. En efecto, para $k = 1$ tenemos que

$$R_{1;1} = \frac{h_1}{2} [f(a) + f(b)] = \frac{b-a}{2} [f(a) + f(b)], \quad (6.93)$$

con $h_1 = b - a$. Para el caso de $k = 2$ tenemos que

$$\begin{aligned} R_{2;1} &= \frac{h_2}{2} [f(a) + f(b) + 2f(a + h_2)] \\ &= \frac{b-a}{4} \left[f(a) + f(b) + 2f\left(a + \frac{b-a}{2}\right) \right] \\ &= \frac{1}{2} \left[\underbrace{\frac{b-a}{2} (f(a) + f(b))}_{R_{1,1}} + \frac{\overbrace{b-a}^{h_1}}{2} 2f(a + h_2) \right] \Rightarrow \\ R_{2;1} &= \frac{1}{2} [R_{1,1} + h_1 f(a + h_2)], \end{aligned} \quad (6.94)$$

con $h_2 = \frac{b-a}{2}$. Análogamente, para $k = 3$, $h_3 = \frac{b-a}{4}$, con lo que nos queda

$$\begin{aligned} R_{3;1} &= \frac{h_3}{2} \left\{ f(a) + f(b) + 2[f(a + h_3) + \underbrace{f(a + 2h_3)}_{h_2}] + f(a + 3h_3) \right\} \\ &= \frac{b-a}{8} \{ f(a) + f(b) + 2f(a + h_2) + 2[f(a + h_3) + f(a + 3h_3)] \} \\ &= \frac{1}{2} \{ R_{2,1} + h_2 [f(a + h_3) + f(a + 3h_3)] \}. \end{aligned} \quad (6.95)$$

Si generalizamos para todos los k , tenemos que

$$R_{k,1} = \frac{1}{2} \left\{ R_{k-1;1} + h_{k-1} \sum_{i=1}^{2^{k-2}} f[a + (2i-1)h_k] \right\}. \quad (6.96)$$

Cada uno de estos $R_{k,1}$ son aproximaciones de nuestro valor buscado. Para refinar estos resultados podemos aplicar, ahora sí, la extrapolación de Richardson con $q = 4$. Por lo tanto tendremos que:

$$R_{k,2} = R_{k,1} + \frac{R_{k,1} - R_{k-1,1}}{4^1 - 1}, \quad (6.97)$$

con $k = 2; 3; \dots; n$. Con esta nueva expresión obtenemos un nuevo conjunto de aproximaciones, cuyo orden de convergencia es proporcional a h^4 , es decir, es similar al método de Simpson.

Si generalizamos para $j = 3; 4; \dots; n$, obtenemos la siguiente expresión:

$$R_{k,j} = R_{k,j-1} + \frac{R_{k,j-1} - R_{k-1,j-1}}{4^{j-1} - 1}; \quad (6.98)$$

con $k = 2; 3; \dots; n$ y $j = 2; 3; \dots; k$. Al aplicar este método, generamos una tabla como la 6.2, donde cada $R_{k,j}$ es una mejor aproximación del resultado, siendo la mejor el $R_{n,n}$.

Tabla 6.2: *Método de Romberg*

$R_{i,1}$	$R_{i,2}$	$R_{i,3}$	\dots	$R_{i,n}$
$R_{1,1}$				
$R_{2,1}$	$R_{2,2}$			
$R_{3,1}$	$R_{3,2}$	$R_{3,3}$		
\vdots	\vdots	\vdots	\ddots	
$R_{n,1}$	$R_{n,2}$	$R_{n,3}$	\dots	$R_{n,n}$

La ventaja de este método es que nos permite calcular una nueva fila con sólo hacer una aplicación de la fórmula compuesta del trapecio y luego usar los valores ya calculados para obtener el resto de los valores de las demás columnas de esa nueva fila; no requiere recalcular todo.

Una cuestión a tener en cuenta al aplicar este método, es que supone que la *Fórmula Compuesta del Trapecio* permite la aplicación de la *Extrapolación de Richardson*, esto es, se debe cumplir que $f(x) \in C^{2(k+1)}[a, b]$; si esto no se cumple, no tiene sentido seguir afinando el resultado hasta la iteración k . Si generalizamos, es evidente que una función $f(x)$ que cumpla con tener infinitas derivadas continuas en el intervalo $[a, b]$ es una función a la cual resulta muy conveniente aplicarle el *Método de Romberg*.

6.2.3. Fórmulas abiertas de Newton-Cotes

En los puntos anteriores hemos visto las fórmulas cerradas para integrar numéricamente. Existen también fórmulas abiertas de Newton-Cotes. La más conocida es la del punto medio. Supongamos que tomamos la fórmula del rectángulo pero en lugar de aproximar el área con los extremos, tomamos el punto medio del intervalo, es decir, $c = \frac{a+b}{2}$. En ese caso nuestra aproximación del área buscada estará dada por:

$$Q_n(f) = \underbrace{(b-a)}_h \cdot f(c) = h \cdot f(c). \quad (6.99)$$

La aproximación efectuada con esta fórmula se puede ver en la figura 6.11.

Al igual que en los casos anteriores, se puede desarrollar una fórmula compuesta, similar a la fórmula compuesta del rectángulo pero tomando los puntos medios de los subintervalos.

Sin embargo, la idea principal de las fórmulas abiertas no está relacionada con tomar puntos de un intervalo según un paso uniforme sino en determinar los puntos para efectuar la

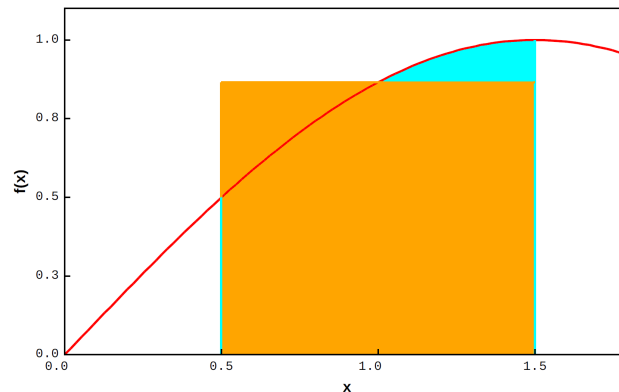


Figura 6.11: *Fórmula del punto medio.*

integración eligiéndolos de una manera diferente. ¿Qué significa esto? Analicemos brevemente la fórmula del punto medio. Al elegir dicho punto y no los extremos del intervalo, suponemos que el rectángulo que queda formado aproxima mejor la integral buscada. Si dividimos este intervalo en varios subintervalos más pequeños, tendremos una fórmula compuesta. Así y todo, estamos algo limitados.

Podríamos avanzar en la idea y desarrollar una fórmula similar para los métodos del Trapecio y de Simpson, es decir, crear una curva que no pase por los extremos y nos permita obtener una buena aproximación. Las mismas son:

- Método del Trapecio abierto:

$$\int_a^b f(x)dx = \frac{b-a}{2} [f(a+h) + f(b-h)] + (b-a) \frac{h^2}{4} f''(\xi), \quad (6.100)$$

con $h = \frac{b-a}{3}$ y $\xi \in (a, b)$;

- Método de Simpson abierto:

$$\int_a^b f(x)dx = \frac{b-a}{3} [2f(a+h) - f(a+2h) + 2f(b-h)] + (b-a) \frac{7h^4}{90} f^{iv}(\xi), \quad (6.101)$$

con $h = \frac{b-a}{4}$ y $\xi \in (a, b)$.

Pero de todas maneras tenemos la misma limitante: debemos trabajar con puntos equidistantes⁸. Esto puede llevar a que debamos utilizar las fórmulas compuestas con muchos términos para alcanzar aproximaciones razonables. Veamos en el punto siguiente un método de integración que explota la idea de las fórmulas abiertas de Newton-Cotes eligiendo puntos en la curva de manera de optimizar la aproximación de la integral buscada.

6.2.4. Cuadratura de Gauss

Recordemos la fórmula para una cuadratura:

$$Q_n(f) = \sum_{i=1}^n c_i f(x_i).$$

Supongamos ahora que elegimos una curva que pase por ciertos puntos y que aproxime la integral de la función dada, usando la fórmula de cuadratura. Curvas de ese tipo vemos en la figura 6.12.

⁸Recordemos que la base de la integración numérica es la interpolación polinómica. Disponer de muchos puntos distribuidos en forma uniforme no siempre redundará en buenos resultados al interpolar con polinomios.

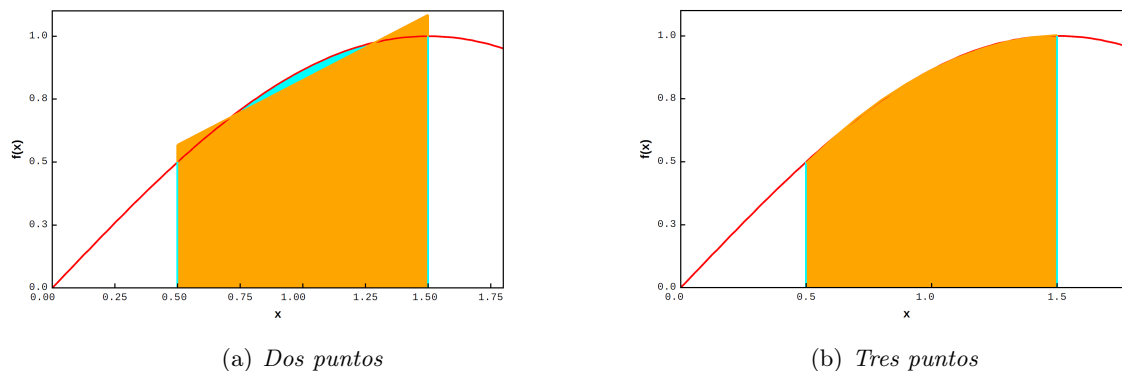


Figura 6.12: Cuadratura usando curvas de aproximación.

Elegiremos como puntos que optimizan la integral buscada a aquellos en los cuales la función se intersecta con la curva de aproximación. Entonces, nuestro problema es elegir la curva más conveniente. Por ejemplo, en la figura 6.12 aplicamos dos, una recta y una parábola. En cada caso tendremos dos y tres puntos respectivamente que intersectan a la curva. Podríamos haber utilizado una parábola cúbica, un polinomio de grado 4, etc.

En los métodos anteriores, para obtener la integral buscada, hemos utilizado puntos conocidos o que podíamos conocer a partir de definir el paso h . Por ejemplo, en el método del trapecio utilizamos dos puntos para aproximar nuestra integral, x_1 y x_2 , de manera que nuestra aproximación queda de la siguiente manera:

$$I(f) = \frac{h}{2} [f(x_1) + f(x_2)]. \quad (6.102)$$

De la fórmula de cuadratura podemos extraer que, si definimos que $h = b - a$, donde $[a, b]$ es nuestro intervalo de integración, para el caso del método del trapecio tendremos que:

$$x_1 = a, \quad x_2 = b, \quad c_1 = c_2 = \frac{b - a}{2}.$$

Supongamos ahora que definimos un intervalo fijo de integración, por ejemplo, el $[-1; 1]$. En vez de fijar los valores x_i , armemos una aproximación de nuestra integral de manera tal que dispongamos de más «variables» para aproximar nuestra integral. En esta nueva situación debemos obtener para ese intervalo los puntos x_i y los coeficientes c_i para nuestra fórmula de cuadratura, esto es, debemos buscar los puntos x_1, x_2, \dots, x_n y los coeficientes c_1, c_2, \dots, c_n que optimicen nuestra aproximación. En consecuencia, tenemos $2n$ incógnitas que debemos obtener.

Si recordamos que un polinomio de grado $2n - 1$ tiene $2n$ coeficientes (por ejemplo, un polinomio de tercer grado tiene la forma $a_0 + a_1x + a_2x^2 + a_3x^3$), podríamos decir que hallar esos parámetros para nuestra fórmula de cuadratura es equivalente a obtener los coeficientes de ese polinomio de grado $2n - 1$.

Por ejemplo, si tomamos aproximamos $f(x)$ con una parábola cúbica, tendremos que:

$$\int_a^b f(x) dx \approx c_1 f(x_1) + c_2 f(x_2) = \int_a^b (a_0 + a_1x + a_2x^2 + a_3x^3) dx. \quad (6.103)$$

Notemos que debe cumplirse que

$$\begin{aligned}
 c_1 (a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3) + c_2 (a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3) &= \int_a^b a_0 dx + \\
 &+ \int_a^b a_1 x dx + \\
 &+ \int_a^b a_2 x^2 dx + \\
 &+ \int_a^b a_3 x^3 dx,
 \end{aligned} \tag{6.104}$$

de lo que resulta un sistema de ecuaciones no lineales:

$$c_1 + c_2 = \int_a^b dx = b - a \tag{6.105}$$

$$c_1 \cdot x_1 + c_2 \cdot x_2 = \int_a^b x dx = \frac{b^2 - a^2}{2} \tag{6.106}$$

$$c_1 \cdot x_1^2 + c_2 \cdot x_2^2 = \int_a^b x^2 dx = \frac{b^3 - a^3}{3} \tag{6.107}$$

$$c_1 \cdot x_1^3 + c_2 \cdot x_2^3 = \int_a^b x^3 dx = \frac{b^4 - a^4}{4}. \tag{6.108}$$

Si obtenemos los valores de c_i y de x_i , y los reemplazamos en la función original, podemos calcular nuestra integral.

Gauss definió estos polinomios para aproximar la integral, en el intervalo $[-1; 1]$ y obtuvo los c_i y x_i para la cantidad de puntos que se deseen utilizar o, lo que es lo mismo, del grado del polinomio de aproximación. Estos polinomios son ortogonales y conocidos como *polinomios de Legendre*⁹, y son los siguientes:

$$\begin{aligned}
 P_0(x) &= 1 & P_1(x) &= x \\
 P_2(x) &= \frac{1}{2}(3x^2 - 1) & P_3(x) &= \frac{1}{2}(5x^3 - 3x) \\
 P_4(x) &= x^4 - \frac{6}{7}x^2 + \frac{3}{35} & P_5(x) &= x \left(x^4 - \frac{10}{9} + \frac{5}{21} \right) \\
 P_k(x) &= \frac{1}{2^k} \frac{d^k}{dx^k} (x^2 - 1)^k.
 \end{aligned}$$

La raíz de cada polinomio resultan ser los puntos x_i . Con éstos y la ayuda de un polinomio interpolante de Lagrange integrado en el intervalo $[-1; 1]$, obtenemos los coeficientes c_i . (En [3] se pueden ver más detalles de cómo obtener los coeficientes de peso.)

En la tabla 6.3 se dan algunos los valores de las raíces y los coeficientes, de acuerdo con la cantidad de puntos que se utilicen para aproximar la integral.

Este método es muy útil cuando lo que queremos aproximar son integrales de funciones polinómicas, puesto que los resultados son exactos cuando $g \leq 2n - 1$, donde g es el grado del polinomio a integrar y n la cantidad de puntos de Gauss. Por ejemplo, con $n = 2$, es decir, con dos puntos de Gauss, podemos aproximar cualquier integral de polinomios cuyo grado sea menor o igual a tres, pues se cumple que $g = 3 \leq 2 \cdot 2 - 1$.

⁹Hemos analizado los *polinomios de Legendre* en el capítulo 5.

Tabla 6.3: Raíces y coeficientes de la cuadratura de Gauss-Legendre

n	x_i	c_i
1	$x_1 = 0,0000000000$	$c_1 = 2,0000000000$
2	$x_1 = -\frac{1}{\sqrt{3}} = -0,5773502692$ $x_2 = \frac{1}{\sqrt{3}} = 0,5773502692$	$c_1 = 1,0000000000$ $c_2 = 1,0000000000$
3	$x_1 = -0,7745966692$ $x_2 = 0,0000000000$ $x_3 = 0,7745966692$	$c_1 = 0,5555555556$ $c_2 = 0,8888888889$ $c_3 = 0,5555555556$
4	$x_1 = -0,8611363116$ $x_2 = -0,3399810436$ $x_3 = 0,3399810436$ $x_4 = 0,8611363116$	$c_1 = 0,3478548451$ $c_2 = 0,6521451549$ $c_3 = 0,6521451549$ $c_4 = 0,3478548451$
5	$x_1 = -0,9061798459$ $x_2 = -0,5384693101$ $x_3 = 0,0000000000$ $x_4 = 0,5384693101$ $x_5 = 0,9061798459$	$c_1 = 0,2369268850$ $c_2 = 0,4786286705$ $c_3 = 0,5688888889$ $c_4 = 0,4786286705$ $c_5 = 0,2369268850$

Si el intervalo de integración no es $[-1; 1]$, basta con hacer un cambio de coordenadas. Si tenemos la siguiente integral:

$$I(f) = \int_a^b f(x) dx,$$

debemos hacer la siguiente transformación lineal para poder aproximar con cuadratura de Gauss:

$$x = \frac{b-a}{2}t + \frac{b+a}{2}; \quad I(f) = \frac{b-a}{2} \int_{-1}^1 f(t) dt. \quad (6.109)$$

Finalmente, una cuestión a tener en cuenta es el error cometido al aproximar una integral mediante cuadratura de Gauss. La expresión del error el intervalo $[-1; 1]$ está dado por

$$E = \frac{2^{2n+1}(n!)^4}{(2n+1)[(2n)!]^2} f^{2n}(\xi), \quad (6.110)$$

donde n es el número de puntos utilizados y $\xi \in [-1; 1]$. Si ampliamos el método al intervalo $[a, b]$, tenemos que el error está dado por

$$E = \frac{(b-a)^{2n+1}(n!)^4}{(2n+1)[(2n)!]^2} f^{2n}(\xi), \quad (6.111)$$

con $\xi \in [a, b]$. Vemos que en ambos casos el error cometido es proporcional a la derivada de orden $2n$. Por ejemplo, si $n = 2$, entonces el error cometido es proporcional a la derivada cuarta ($f^{iv}(\xi)$), pues tenemos

$$E = \frac{(b-a)^{2 \cdot 2+1}(2!)^4}{(2 \cdot 2+1)[(2 \cdot 2)!]^2} f^{2 \cdot 2}(\xi) = \frac{(b-a)^5(2!)^4}{5(4!)^2} f^{iv}(\xi). \quad (6.112)$$

Esto confirma que con dos puntos de Gauss ($n = 2$) obtenemos una integral «exacta» para polinomios de grado 3 o menor, pues en esos casos se cumple que $f^{iv}(x) = 0$ para cualquier x , por lo tanto, también para $f^{iv}(\xi)$ con $\xi \in [a, b]$.

Al igual que para los métodos anteriores, podemos pensar en un método compuesto para Gauss. Efectivamente, si dividimos el intervalo $[a; b]$ en subintervalos más pequeños, podemos utilizar la cuadratura de Gauss en esos subintervalos, con la correspondiente transformación lineal, e inclusive usar un aproximación con n no mayor a 3, con excelentes resultados.

Como hemos dicho, el método es muy bueno para aproximar integrales sobre todo de funciones polinómicas. El método pierde practicidad si no conocemos la función (por ejemplo, sólo conocemos puntos), y si debemos programar una base de datos con todas las raíces de los polinomios de Legendre con sus coeficientes de peso.

6.2.5. Integrales múltiples

Al igual que para el caso de integrales simples, podemos calcular en forma numérica integrales múltiples, en dos o tres dimensiones. Tomemos la siguiente integral:

$$\iint_A f(x, y) dA, \quad (6.113)$$

donde A es una región rectangular en el plano tal que

$$A = \{(x, y) | a \leq x \leq b; c \leq y \leq d\}.$$

Entonces, podemos escribir la integral de arriba como

$$\int_c^d \left[\int_a^b f(x, y) dx \right] dy. \quad (6.114)$$

Integremos respecto a x usando el método del trapecio. De esta manera obtendremos

$$\int_a^b f(x, y) dx \approx \frac{b-a}{2} [f(a, y) + f(b, y)]. \quad (6.115)$$

Reemplacemos esta expresión en la integral doble y hagamos lo mismo pero respecto a y . Entonces nos queda que

$$\begin{aligned} \int_c^d \left[\int_a^b f(x, y) dx \right] dy &\approx \int_c^d \frac{b-a}{2} [f(a, y) + f(b, y)] dy \\ &\approx \frac{b-a}{2} \int_c^d [f(a, y) + f(b, y)] dy \\ &\approx \frac{b-a}{2} \left[\int_c^d f(a, y) dy + \int_c^d f(b, y) dy \right] \end{aligned} \quad (6.116)$$

Si aplicamos a cada integral la regla del trapecio, nos queda

$$\int_c^d f(a, y) dy \approx \frac{d-c}{2} [f(a, c) + f(a, d)] \quad (6.117)$$

$$\int_c^d f(b, y) dy \approx \frac{d-c}{2} [f(b, c) + f(b, d)]. \quad (6.118)$$

Al reemplazar estas dos expresiones en (6.116) nos queda que

$$\int_c^d \int_a^b f(x, y) dx dy \approx \frac{(b-a)(d-c)}{4} [f(a, c) + f(a, d) + f(b, c) + f(b, d)]. \quad (6.119)$$

En definitiva, podemos obtener una aproximación de una integral múltiple, en este caso doble, mediante la aplicación del método del trapecio en dos dimensiones. También aplicando

el método de Simpson podemos obtener una aproximación de dicha integral. En este caso, la expresión es

$$\int_c^d \int_a^b f(x, y) \, dx \, dy \approx \frac{h_x h_y}{9} \left\{ f(a, c) + f(a, d) + f(b, c) + f(b, d) + 4 \left[f\left(a, \frac{c+d}{2}\right) + f\left(b, \frac{c+d}{2}\right) + f\left(\frac{a+b}{2}, c\right) + f\left(\frac{a+b}{2}, d\right) \right] + 16 \left[f\left(\frac{a+b}{2}, \frac{c+d}{2}\right) \right] \right\}, \quad (6.120)$$

donde $h_x = \frac{b-a}{2}$ y $h_y = \frac{d-c}{2}$. Si reemplazamos esto último en la expresión general y además definimos $x_0 = a$, $x_1 = \frac{a+b}{2}$, $x_2 = b$, $y_0 = c$, $y_1 = \frac{c+d}{2}$ e $y_2 = d$, tenemos que

$$\int_c^d \int_a^b f(x, y) \, dx \, dy \approx \frac{(b-a)(d-c)}{36} \{ f(x_0, y_0) + f(x_0, y_2) + f(x_2, y_0) + f(x_2, y_2) + 4 [f(x_0, y_1) + f(x_1, y_0) + f(x_1, y_2) + f(x_2, y_1) + 4f(x_1, y_1)] \}. \quad (6.121)$$

El error cometido por aproximar la integral mediante esta fórmula está dado por:

$$E_T = \frac{(b-a)(d-c)}{12} \left[h_x^2 \frac{\partial^2 f(\hat{\xi}, \hat{\mu})}{\partial x^2} + h_y^2 \frac{\partial^2 f(\bar{\xi}, \bar{\mu})}{\partial y^2} \right] \quad (\text{Método del trapecio}), \quad (6.122)$$

$$E_S = \frac{(b-a)(d-c)}{90} \left[h_x^4 \frac{\partial^4 f(\hat{\xi}, \hat{\mu})}{\partial x^4} + h_y^4 \frac{\partial^4 f(\bar{\xi}, \bar{\mu})}{\partial y^4} \right] \quad (\text{Método de Simpson}), \quad (6.123)$$

que, como podemos observar, son muy parecidos a los vistos para el caso de integrales simples.

Estos métodos también se pueden modificar para obtener las fórmulas compuestas, similares a las vistas anteriormente. (Para más detalles, véase [3].)

Así como hemos aplicado los métodos de trapecio y de Simpson, lo mismo podemos hacer con la cuadratura de Gauss. Si aplicamos el mismo razonamiento para integrar según x tendremos que

$$\int_a^b f(x, y) \, dx \approx \frac{b-a}{2} \sum_{i=1}^n c_i f(x_i, y). \quad (6.124)$$

Si hacemos lo mismo respecto de y , obtendremos

$$\begin{aligned} \int_c^d \int_a^b f(x, y) \, dx \, dy &\approx \int_c^d \frac{b-a}{2} \sum_{i=1}^n c_i f(x_i, y) \, dy \\ &\approx \frac{b-a}{2} \sum_{i=1}^n \int_c^d c_i f(x_i, y) \, dy \\ &\approx \frac{b-a}{2} \sum_{i=1}^n \frac{d-c}{2} \sum_{j=1}^m c_i c_j f(x_i, y_j) \\ &\approx \frac{b-a}{2} \frac{d-c}{2} \sum_{i=1}^n \sum_{j=1}^m c_i c_j f(x_i, y_j) \\ &\approx \frac{(b-a)(d-c)}{4} \sum_{i=1}^n \sum_{j=1}^m c_i c_j f(x_i, y_j), \end{aligned} \quad (6.125)$$

con

$$x_i = \frac{b-a}{2}t_i + \frac{b+a}{2} \quad (6.126)$$

$$y_j = \frac{d-c}{2}t_j + \frac{d+c}{2}, \quad (6.127)$$

donde t_i y t_j son las raíces de los polinomios de Legendre, y c_i y c_j , los coeficientes de peso. Por ejemplo, si tomamos $n = m = 2$ tenemos que $t_1 = -\frac{1}{\sqrt{3}}$, $t_2 = \frac{1}{\sqrt{3}}$ y $c_1 = c_2 = 1$, y la aproximación nos queda como

$$\int_c^d \int_a^b f(x, y) \, dx \, dy \approx \frac{(b-a)(d-c)}{4} [f(x_1, y_1) + f(x_1, y_2) + f(x_2, y_1) + f(x_2, y_2)]. \quad (6.128)$$

con

$$x_1 = -\frac{b-a}{2} \frac{1}{\sqrt{3}} + \frac{b+a}{2} \quad x_2 = \frac{b-a}{2} \frac{1}{\sqrt{3}} + \frac{b+a}{2},$$

y

$$y_1 = -\frac{d-c}{2} \frac{1}{\sqrt{3}} + \frac{d+c}{2} \quad y_2 = \frac{d-c}{2} \frac{1}{\sqrt{3}} + \frac{d+c}{2}.$$

Podemos ver que con este método solamente tenemos que evaluar la función a integrar en cuatro puntos, en cambio, con el método de Simpson debemos evaluar la misma función en nueve puntos. Este método es muy utilizado por el *Método de los Elementos Finitos* para obtener integrales dobles.

6.3. Notas finales

La integración numérica es uno de los métodos numéricos más utilizados en la ingeniería y en la ciencia en general. Inclusive, muchos programas para computadoras hacen usos de los algoritmos vistos en este capítulo. Por ejemplo, el MatLab[®] aplica el *Método de Simpson* en su función `quad` que calcula integrales definidas, en tanto que el Mathcad[®], aplica el *Método de Romberg*, entre otros.

Por otro lado, uno de los métodos numéricos más utilizados en el análisis estructural, el *Método de los Elementos Finitos*, aplica la integración numérica en forma sistemática para obtener la matriz de rigidez de un sistema estático. Más aún, para ciertos casos especiales hace uso exclusivo de la cuadratura de Gauss, como es el caso de la integración en una y dos dimensiones para elementos lineales o de superficie (elementos de barra, de viga, de estado plano y de placa) e inclusive para determinados tipos de elementos se ayuda con una «integración reducida» para evitar ciertos problemas del modelo numérico.

Ejercicios

Diferenciación numérica

1. Aproxime las derivadas primeras aplicando los *Métodos de Diferencias Progresiva y Regresiva* de orden 1 para completar la siguiente tabla de datos:

a)	x	$f(x)$	$f'(x)$	b)	x	$f(x)$	$f'(x)$
	0,5	0,4794			0,0	0,00000	
	0,6	0,5646			0,2	0,74140	
	0,7	0,6442			0,4	1,3718	

- Aproxime las derivadas primeras aplicando los *Métodos de Diferencias Progresiva, Centrada y Regresiva* (todos de orden 2) para completar la tabla de datos el ejercicio anterior.
- Aproxime las derivadas primeras aplicando los métodos necesarios para que el orden de convergencia sea 2, para completar la siguiente tabla de datos:

a)	x	$f(x)$	$f'(x)$		b)	x	$f(x)$	$f'(x)$
	1,1	9,025013				8,1	16,94410	
	1,2	11,02318				8,3	17,56492	
	1,3	13,46374				8,5	18,19056	
	1,4	16,44465				8,7	18,82091	

- Aplice el método de *Extrapolación de Richardson* iterando hasta $N_3(h)$ para aproximar la derivada de las siguientes funciones:
 - $f(x) = \ln x$, para $x_0 = 2$ y $h = 0,4$;
 - $f(x) = x + e^x$, para $x_0 = 0$ y $h = 0,4$;
 - $f(x) = e^x \sin x$, para $x_0 = 2$ y $h = 0,2$;
 - $f(x) = e^{-x} x^2$, para $x_0 = 3$ y $h = 0,1$.
- A partir de los datos de la siguiente tabla, aproxime con las fórmulas adecuadas (que utilicen todos los datos disponibles) $f'(0,4)$, $f''(0,4)$, $f'(0,6)$ y $f''(0,6)$.

x	0,2	0,4	0,6	0,8	1,0
$f(x)$	0,9798652	0,9177710	0,8080348	0,6386093	0,3843735

Integración numérica

Métodos de Newton-Cotes Cerrados

- Aproxime las siguientes integrales aplicando el *Método del Trapecio*, el *Método de Simpson* y el *Método del Trapecio Mejorado*. Compare los resultados obtenidos.

$$\begin{array}{lll}
 \text{a) } \int_0^{10} 4x \, dx & \text{b) } \int_1^5 (2x^2 + 7) \, dx & \text{c) } \int_2^6 (3x^3 + 5) \, dx \\
 \text{d) } \int_{-1}^2 (2,5x^4 + 3x) \, dx & \text{e) } \int_0^{\pi/4} \sin x \, dx & \text{f) } \int_{-2}^5 e^x \, dx
 \end{array}$$

- Aproximar nuevamente las integrales del punto anterior pero aplicando el *Método del Trapecio Compuesto*, el *Método de Simpson Compuesto* y el *Método del Trapecio Mejorado Compuesto*.
- Repita el punto anterior pero aplicando el *Método de Romberg*.

Métodos de Newton-Cotes Abiertos

- Aproxime las integrales del ítem 1 del punto anterior aplicando el *Método del Punto Medio* y la *Cuadratura de Gauss-Legendre*.

2. Aproxime las siguientes integrales mediante *Cuadratura de Gauss-Legendre*, eligiendo la cantidad de puntos necesarios para obtener un resultado «exacto»:

$$a) \int_{-1}^1 4x^7 + 5x^5 dx$$

$$b) \int_{-2,5}^{10} 10x^8 + 3x^2 dx$$

Ejemplos de aplicación práctica

1. El *Método de los Elementos Finitos* aplicado al Análisis Estructural resuelve estructuras estáticas mediante la modelación de las estructuras con una malla discreta de elementos, que genera un sistema de ecuaciones lineales, usualmente definido como $KU = R$, donde la matriz K se denomina *matriz de rigidez*. Esta matriz se obtiene por la integración de cada una de sus componentes en un dominio dado, generalmente aplicando la *Cuadratura de Gauss-Legendre*. Como ejemplo de ello, calcule las componentes de una matriz de rigidez de dimensión 4×4 de manera de que el resultado de dichas integraciones sea «exacto».

$$K_{1;1} = \int_0^2 \left(\frac{3}{2} - \frac{3x}{2} \right)^2 dx$$

$$K_{1;2} = \int_0^2 \left(\frac{3}{2} - \frac{3x}{2} \right) \left(2 - \frac{3x}{2} \right) dx$$

$$K_{1;3} = \int_0^2 \left(\frac{3}{2} - \frac{3x}{2} \right) \left(\frac{3x}{2} - \frac{3}{2} \right) dx$$

$$K_{1;4} = \int_0^2 \left(\frac{3}{2} - \frac{3x}{2} \right) \left(1 - \frac{3x}{2} \right) dx$$

$$K_{2;2} = \int_0^2 \left(2 - \frac{3x}{2} \right)^2 dx$$

$$K_{2;3} = \int_0^2 \left(2 - \frac{3x}{2} \right) \left(\frac{3x}{2} - \frac{3}{2} \right) dx$$

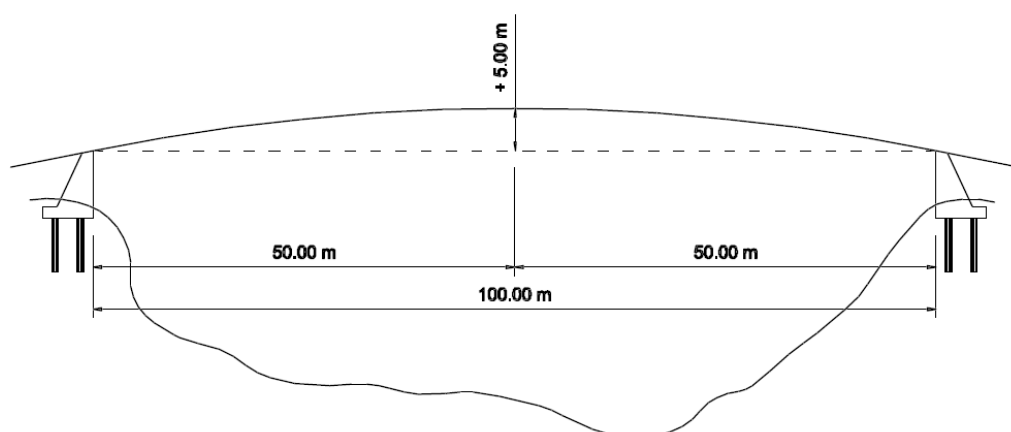
$$K_{2;4} = \int_0^2 \left(2 - \frac{3x}{2} \right) \left(1 - \frac{3x}{2} \right) dx$$

$$K_{3;3} = \int_0^2 \left(\frac{3x}{2} - \frac{3}{2} \right)^2 dx$$

$$K_{3;4} = \int_0^2 \left(\frac{3x}{2} - \frac{3}{2} \right) \left(1 - \frac{3x}{2} \right) dx$$

$$K_{4;4} = \int_0^2 \left(1 - \frac{3x}{2} \right)^2 dx$$

2. El tablero de un puente parabólico (polinomio de segundo grado) muestra una cota de +0,00 m en los extremos y una cota de +5,00 m en el centro. El puente se ha diseñado para cruzar un cañadón, para lo cual se ha considerado adecuado que la distancia entre apoyos (extremos) sea de 100 m. Determine la longitud del desarrollo del tablero.



Integrales Múltiples en forma numérica

1. Aplique el *Método de Simpson* para aproximar las siguientes integrales múltiples. Considere $m = n = 2$.

a) $\int_{2,1}^{2,1} \int_{1,2}^{1,4} x y \, dy \, dx.$

b) $\int_2^{2,2} \int_x^{2x} (x^2 + y^2) \, dy \, dx.$

c) $\int_0^{0,5} \int_0^{0,5} e^{y-x} \, dy \, dx.$

2. Aproxime las integrales del punto anterior pero considere $m = 4$ (Método de Simpson Compuesto) y $n = 2$.
3. Ídem el caso anterior pero con $n = 4$ (Método de Simpson Compuesto) y $m = 2$.
4. Ídem el caso 2 pero con $m = n = 3$.
5. Aproxime las integrales del punto 1 pero aplicando el *Método del Trapecio Compuesto* con $m = n = 4$.
6. Nuevamente aproxime las integrales del punto 1 aplicando *Cuadratura de Gauss-Legendre*:
 - a) Con $m = n = 1$.
 - b) Con $m = n = 2$.
 - c) Con $m = n = 3$.

Capítulo 7

Ecuaciones diferenciales ordinarias

7.1. Ecuaciones diferenciales ordinarias con valores iniciales

7.1.1. Introducción

Una importante proporción de los problemas que debemos resolver como ingenieros se pueden representar mediante ecuaciones diferenciales ordinarias, que son aquellas que están expresadas en derivadas totales ¹. Como ejemplos de este tipo de ecuaciones tenemos las siguientes:

- El equilibrio de una viga sometida a flexión: $\frac{dM}{dx} + p = 0$;
- Un circuito del tipo LR: $L \frac{di}{dt} + R i = V$;
- La transmisión del calor unidimensional: $q = -kA \frac{dT}{dx}$ ².

Así, buena parte de los métodos que empleamos para encarar un determinado problema resultan ser soluciones analíticas de ecuaciones diferenciales que se aplican en forma metódica y que se han obtenido a partir de ciertas condiciones, que pueden ser iniciales o de borde. Un caso bien conocido es la resolución de sistemas hiperestáticos en Estática (también los isostáticos), en los que se aplicaban métodos prácticos y numéricos (como el método de Cross) derivados de las soluciones analíticas.

Del conjunto de ecuaciones diferenciales empezaremos por las más sencilla, que son aquellas que involucran a la primera derivada, de las que basta conocer las condiciones iniciales. Si bien en cualquier curso de Análisis Matemático se aprenden métodos analíticos para obtener las soluciones de dichas ecuaciones, sabemos que no siempre son aplicables o no siempre obtendremos soluciones analíticas. Por ejemplo, y volviendo al caso de estructuras hiperestáticas, no resulta sencillo resolver la ecuación diferencial para el caso de una carga concentrada. Es en estos casos cuando los métodos numéricos se convierten en la única herramienta para obtener algún tipo de solución aproximada que nos permita resolver el problema.

Existen muchos ejemplos de ecuaciones diferenciales con condiciones iniciales, entre los cuales podemos mencionar los siguientes:

- **Dinámica de poblaciones.** El economista inglés Thomas Malthus propuso el siguiente modelo matemático para definir el crecimiento demográfico:

$$\frac{dP}{dt} = kP;$$

¹Para una mejor comprensión del tema, ver [21].

²En realidad, se trata de un sistema de ecuaciones diferenciales, pues $q = \frac{dQ}{dt}$

con $k > 0$, es decir, que la tasa de crecimiento de la población es proporcional a la población total. (Este modelo en realidad no es muy preciso, pues deja de lado otros factores como la inmigración, por ejemplo, pero en su momento daba una buena aproximación al problema demográfico.)

- **Desintegración radiactiva.** El siguiente modelo matemático es el que se aplica para el estudio de la desintegración radiactiva:

$$\frac{dA}{dt} = kA;$$

en este caso, con $k < 0$. Este modelo es la base del método de datación por Carbono 14, usado en muchas disciplinas científicas.

- **Ley de Newton del enfriamiento o calentamiento.** Isaac Newton propuso la siguiente ley matemática para el cambio de temperatura:

$$\frac{dT}{dt} = k(T - T_m);$$

con $k < 0$, donde T_m es la temperatura del medio, y T la del objeto analizado.

- **Ley de Torricelli.** El drenado de un tanque cumple con el siguiente modelo:

$$\frac{dV}{dt} = -A_h \sqrt{2gh}.$$

Si definimos $V = A_w h$, entonces la expresión anterior se puede escribir como

$$\frac{dh}{dt} = -\frac{A_h}{A_w} \sqrt{2gh}.$$

La mayoría de los libros toma el caso del péndulo como el ejemplo tradicional de las ecuaciones diferenciales ordinarias con valores iniciales. El modelo matemático que representa este fenómeno está dado por:

$$\frac{d^2\theta}{dt^2} = -\frac{g}{L} \text{sen}(\theta),$$

donde g es la aceleración de la gravedad, L , la longitud del péndulo, y θ , el ángulo del péndulo respecto de la vertical. Este ejemplo suele linealizarse para el caso de ángulos muy pequeños, pues se cumple que $\text{sen}(\theta) = \tan(\theta) = \theta$, y la ecuación diferencial queda

$$\frac{d^2\theta}{dt^2} = -\frac{g}{L}\theta,$$

modelo que en realidad está representado con una ecuación diferencial de segundo orden.

Otro ejemplo de la ingeniería civil en el ámbito del análisis estructural es la ecuación del esfuerzo normal en una barra, que se define como

$$\frac{dN}{dx} = -t(x);$$

donde $t(x)$ es una carga uniformemente distribuida en el eje de la barra.

En lo que sigue veremos, primero, las condiciones para que la solución de una ecuación diferencial ordinaria tenga solución única, y en segundo término, varios métodos para resolver numéricamente este tipo de ecuaciones.

7.1.2. Condición de Lipschitz

Una ecuación diferencial ordinaria con valor inicial está definida de la siguiente manera:

$$\frac{dy}{dt} = f(t, y) \quad \text{con } a \leq t \leq b \quad \text{e } y(a) = y_0.$$

Una función $f(t, y) \in D \subset \mathfrak{R}^2$, con D convexo, cumple con la condición de Lipschitz si satisface que

$$|f(t, y_1) - f(t, y_2)| \leq L |y_1 - y_2|,$$

o

$$\left| \frac{\partial f(t, y)}{\partial y} \right| \leq L,$$

para todo $(t, y) \in D$.

Para que una ecuación diferencial tenga solución única debe satisfacer el siguiente teorema.

Teorema 7.1. Sea $f(t, y)$ continua en D , tal que $D = \{(t, y) | a \leq t \leq b; -\infty \leq y \leq +\infty\}$. Si $f(t, y)$ satisface la condición de Lipschitz en D en la variable y , entonces el problema de valor inicial

$$\frac{dy}{dt} = f(t, y) \quad \text{con } a \leq t \leq b \quad \text{e } y(a) = y_0,$$

tiene solución única $y(t)$ para $a \leq t \leq b$.

Por lo tanto, toda ecuación diferencial con valor inicial que cumpla con la condición de Lipschitz tiene solución única.

7.1.3. Problema bien planteado

Un problema de valor inicial del tipo

$$\frac{dy}{dt} = f(t, y) \quad \text{con } a \leq t \leq b \quad \text{e } y(a) = y_0$$

se dice bien planteado si:

- El problema tiene solución única (cumple con la condición de Lipschitz);
- Para cualquier $\epsilon > 0$, existe una constante positiva $k(\epsilon)$ con la propiedad de que siempre que $|\epsilon_0| < \epsilon$, y un $\delta(t)$ que sea continuo, con $\delta(t) < \epsilon$ en $[a; b]$, el problema tiene solución única $z(t)$; es decir,

$$\frac{dz}{dt} = f(t, z) + \delta(t) \quad \text{con } a \leq t \leq b \quad \text{e } z(a) = y_0 + \epsilon_0,$$

con

$$|z(t) - y(t)| < k(\epsilon) \epsilon,$$

para toda $a \leq t \leq b$.

En definitiva, un problema está bien planteado si una perturbación (un $\delta(t)$) del problema original no cambia la esencia del mismo. El siguiente teorema define la condición de *problema bien planteado*.

Teorema 7.2. Sea $D = \{(t, y) | a \leq t \leq b; -\infty \leq y \leq +\infty\}$. Si $f(t, y)$ es continua y satisface la condición de Lipschitz en la variable y en el conjunto D , entonces el problema de valor inicial

$$\frac{dy}{dt} = f(t, y) \quad \text{con } a \leq t \leq b \quad \text{e } y(a) = y_0$$

se dice *bien planteado*.

7.1.4. Métodos de Euler explícito e implícito

Un vez definidas las condiciones que debe cumplir el problema de valor inicial para tener solución única, nos ocuparemos de los métodos para resolverlo.

Para empezar, tomemos la formulación del problema

$$\frac{dy}{dt} = f(t, y). \quad (7.1)$$

Desarrollamos por Taylor la función $y(t)$, desconocida, en un entorno $[t, t+h]$ para obtener $y(t+h)$:

$$y(t+h) = y(t) + y'(t)h + y''(t)\frac{h^2}{2} + \dots \quad (7.2)$$

Como $y'(t) = f(t, y)$ podemos escribir la expresión como sigue:

$$y(t+h) = y(t) + f(t, y)h + y''(t)\frac{h^2}{2} + \dots \quad (7.3)$$

Dado que nuestro entorno de la solución está dado por $[a, b]$, definamos el paso h como $h = \frac{b-a}{N}$, donde N es el número de intervalos. Ahora definamos que $t_{i+1} = t_i + h$. Así, nuestra expresión anterior queda:

$$y(t_{i+1}) = y(t_i) + h f[t_i, y(t_i)] + y''(t_i)\frac{h^2}{2} + \dots \quad (7.4)$$

Si truncamos en la segunda derivada, nos queda

$$y(t_{i+1}) = y(t_i) + h f[t_i, y(t_i)] + y''(\xi_i)\frac{h^2}{2}, \quad (7.5)$$

con $\xi_i \in [t_i, t_{i+1}]$.

Puesto que lo que buscamos es una aproximación de $y(t_{i+1})$, definámosla como w_{i+1} , sin considerar el término de error. Entonces nuestra expresión queda de la siguiente forma:

$$w_{i+1} = w_i + h f(t_i, w_i), \quad (7.6)$$

para $i = 0; 1; \dots; N-1$. Este método se conoce como *Método de Euler Explícito*.

Supongamos ahora que desarrollamos $y(t)$ en t_i+h para obtener $y(t_i)$. Entonces tendremos que

$$y(t_i) = y(t_i+h) - y'(t_i+h)h + y''(t_i+h)\frac{h^2}{2} + \dots; \quad (7.7)$$

y, como $y'(t_i+h) = f[t_i+h, y(t_i+h)]$, nos queda que

$$y(t_i) = y(t_i+h) - f[t_i+h, y(t_i+h)]h + y''(t_i+h)\frac{h^2}{2} + \dots \quad (7.8)$$

Nuevamente, como $t_{i+1} = t_i+h$, y despejando $y(t_{i+1})$ limitando otra vez la expresión a la segunda derivada, tenemos que

$$y(t_{i+1}) = y(t_i) + h f[t_{i+1}, y(t_{i+1})] - y''(\xi_i)\frac{h^2}{2}, \quad (7.9)$$

con $\xi_i \in [t_i, t_{i+1}]$.

En forma análoga al método anterior, lo que en realidad buscamos es una aproximación de $y(t_{i+1})$, por lo tanto tendremos la siguiente expresión:

$$w_{i+1} = w_i + h f(t_{i+1}, w_{i+1}), \quad (7.10)$$

para $i = 0; 1; \dots; N-1$. Este método se conoce como *Método de Euler Implícito*.

7.1.5. Método Predictor-Corrector de Euler

Existe otra forma de salvar esta situación. Supongamos que planteamos el siguiente sistema:

$$w_{i+1}^* = w_i + h f(t_i, w_i) \quad (7.11)$$

$$w_{i+1} = w_i + h f(t_{i+1}, w_{i+1}^*). \quad (7.12)$$

La idea es obtener una primera aproximación de w_{i+1} con el método explícito, que llamaremos w_{i+1}^* , para luego usarla en el método implícito y obtener una nueva aproximación de w_{i+1} , que «corrige» el valor antes obtenido. La combinación de estos dos métodos se conoce como *método predictor-corrector de Euler*³.

Una forma de mejorar la aproximación con este método es mediante iteraciones de la fórmula correctora. El método queda de la siguiente forma:

$$w_{i+1}^* = w_i + h f(t_i, w_i) \quad (7.13)$$

$$w_{i+1}^0 = w_i + h f(t_{i+1}, w_{i+1}^*) \quad (7.14)$$

$$w_{i+1}^{n+1} = w_i + h f(t_{i+1}, w_{i+1}^n), \quad (7.15)$$

iteración que podemos truncar cuando $|w_{i+1}^{n+1} - w_{i+1}^n| < TOL^4$.

Si bien los *Métodos de Euler* son bastante sencillos de implementar, los resultados que se obtienen no son buenas aproximaciones de nuestro problema. Se los usa solamente como introducción a los métodos numéricos y para el análisis del error.

7.1.6. Error cometido al resolver una ecuación diferencial

Para analizar el error cometido, consideremos estos dos lemas:

1. Para toda $x \geq -1$ y para cualquier m positiva, tenemos que $0 \leq (1+x)^m \leq e^{mx}$.
2. Si s y t son números reales positivos, $\{a_i\}_{i=0}^k$ es una sucesión que satisface $a_0 \geq -t/s$, y se cumple que

$$a_{i+1} \leq (1+s)a_i + t, \quad \text{para cada } i = 0; 1; 2; \dots; k,$$

entonces

$$a_{i+1} \leq e^{(i+1)s} \left(a_0 + \frac{t}{s} \right) - \frac{t}{s}.$$

A partir de estos dos lemas se tiene el siguiente teorema.

Teorema 7.3. Sea $f(t, y)$ continua, que satisface la condición de Lipschitz con la constante L en

$$D = \{(t, y) | a \leq t \leq b; -\infty \leq y \leq +\infty\},$$

y existe una constante M , tal que $|y''(t)| \leq M$ para toda $t \in [a, b]$. Si $y(t)$ es la solución única del problema de valor inicial dado por

$$\frac{dy}{dt} = f(t, y) \quad \text{con } a \leq t \leq b \quad \text{e } y(a) = y_0,$$

y los w_0, w_1, \dots, w_N son las aproximaciones a nuestra función, obtenidas por el método de Euler, entonces se cumple que

$$|y(t_i) - w_i| \leq \frac{hM}{2L} \left[e^{L(t_i-a)} - 1 \right].$$

La demostración de este teorema se puede ver [3].

³Este método no suele estar incluido en los libros de texto, posiblemente porque no mejora la aproximación de una manera significativa. Una excepción es [13].

⁴En [13] hay una demostración para algunos casos particulares, en la cual alcanza con dos iteraciones, sin necesidad de analizar si $w_{i+1}^{n+1} - w_{i+1}^n < TOL$.

Orden de convergencia

El error que acabamos de analizar es el error global, pues hemos estimado una cota del error entre el valor real (o exacto) y la aproximación por un método numérico. Sin embargo, los métodos numéricos suelen definirse según el *error local*, es decir, el error entre dos iteraciones sucesivas. Este error, en el método de Euler, está dado por:

$$e_L = \frac{y(t_{i+1}) - y(t_i)}{h} - f[t_i, y(t_i)].$$

Como vimos, el método explícito de Euler surge a partir de un desarrollo de Taylor, del cual resulta que

$$y(t_{i+1}) = y(t_i) + hy'(t_i) + \frac{h^2}{2}y''(t_i) + \dots = y(t_i) + hf[t_i, y(t_i)] + f'[t_i, y(t_i)]\frac{h^2}{2} + \dots;$$

por lo tanto,

$$\begin{aligned} \frac{y(t_{i+1}) - y(t_i)}{h} - f[t_i, y(t_i)] &= \frac{h}{2}f'[\xi, y(\xi)] \\ e_L &= \frac{h}{2}f'[\xi, y(\xi)], \end{aligned}$$

con $\xi \in [t_i, t_{i+1}]$, lo que muestra que el error local del método de Euler es $O(h)$, es decir, tiene un orden de convergencia lineal. Con un análisis similar podemos demostrar que el método implícito es del mismo orden de convergencia. Y dado que ambos métodos son de convergencia lineal, lo mismo podemos decir del predictor-corrector.

7.1.7. Métodos de Taylor de orden superior

Vimos que los *Métodos de Euler* son muy fáciles de aplicar pero su aproximación es de orden de convergencia lineal. Una forma de mejorarlo es partir otra vez del desarrollo por Taylor pero ampliando la cantidad de términos de la serie:

$$y(t_{i+1}) = y(t_i) + h y'(t_i) + \frac{h^2}{2!}y''(t_i) + \frac{h^3}{3!}y'''(t_i) + \dots + \frac{h^n}{n!}y^{(n)}(t_i) + \frac{h^{n+1}}{(n+1)!}y^{(n+1)}(\xi). \quad (7.16)$$

Como además tenemos que

$$\frac{d y(t)}{dt} = y'(t) = f(t, y), \quad y(t_i) = y_i \text{ y } y(t_{i+1}) = y_{i+1}, \quad (7.17)$$

el desarrollo por Taylor lo podemos escribir de la siguiente manera:

$$\begin{aligned} y_{i+1} = & y_i + h f(t_i, y_i) + \frac{h^2}{2!}f'(t_i, y_i) + \frac{h^3}{3!}f''(t_i, y_i) + \dots + \frac{h^n}{n!}f^{(n-1)}(t_i, y_i) + \\ & + \frac{h^{n+1}}{(n+1)!}f^{(n)}(\xi, y(\xi)). \end{aligned} \quad (7.18)$$

Podemos armar un esquema para obtener los y_{i+1} a partir de un polinomio de Taylor, calculando las derivadas totales de la función $f(t, y)$. En forma genérica, una solución aproximada por los polinomios de Taylor la podemos expresar así:

$$y_{i+1} = y_i + h \cdot T^{(n)}(t_i, y_i) + \frac{h^{n+1}}{(n+1)!}f^{(n)}(\xi, y(\xi)), \quad (7.19)$$

donde $T^{(n)}(t_i, y_i)$ representa a los términos del polinomio hasta la derivada de orden n de $y(t)$ o de orden $n - 1$ de $f(t, y)$, o sea:

$$T^{(n)}(t_i, y_i) = f(t_i, y_i) + \frac{h}{2!}f'(t_i, y_i) + \frac{h^2}{3!}f''(t_i, y_i) + \dots + \frac{h^{n-1}}{n!}f^{(n-1)}(t_i, y_i), \quad (7.20)$$

o

$$T^{(n)}(t_i, y_i) = \sum_{i=1}^n \frac{h^{(i-1)}}{i!} f^{(i-1)}(t_i, y_i). \quad (7.21)$$

Por ejemplo, para $n = 2$ tenemos:

$$T^{(2)}(t_i, y_i) = f(t_i, y_i) + \frac{h}{2} \left. \frac{\partial f(t, y)}{\partial t} \right|_{t_i, y_i} + \frac{h}{2} \left. \frac{\partial f(t, y)}{\partial y} f(t, y) \right|_{t_i, y_i}. \quad (7.22)$$

Como hemos visto, el error cometido al resolver la ecuación diferencial aplicando este esquema es el primer término que dejamos de considerar en $T^{(n)}(t, y)$:

$$E = \frac{h^{n+1}}{(n+1)!} f^{(n)}[\xi, y(\xi)] \quad \text{con } \xi \in [t_i, t_{i+1}], \quad (7.23)$$

y como el error local está dado por

$$e_L = \frac{y(t_{i+1}) - y(t_i)}{h} - f[t_i, y(t_i)]; \quad (7.24)$$

para este caso queda definido como

$$e_L = \frac{h^n}{(n+1)!} f^{(n)}[\xi_i, y(\xi_i)], \quad (7.25)$$

con $\xi \in [t_i, t_{i+1}]$. Estos métodos se conocen como *métodos de Taylor de orden superior*, pues podemos definir el orden de convergencia igual a n , siempre que al menos $f(t, y) \in C^{n-1}[a, b]$. Podemos ver que el método de Euler es un caso particular del método de Taylor para $n = 1$. (Podríamos armar métodos de Taylor de orden superior implícitos, aunque de escasa utilidad, dado que deberíamos transformar algebraicamente el algoritmo para obtener una formulación explícita.)

7.1.8. Métodos de Runge-Kutta

Los métodos de Taylor resultan muy instructivos para entender cómo mejorar nuestras aproximaciones, pero muy poco prácticos al momento de implementar un algoritmo de cálculo. El principal escollo para esto es la necesidad de calcular las derivadas de $y(t)$ (o de $f(t, y)$), algo que no siempre es fácil de hacer. Eso obligaría en muchos casos a programar algoritmos particulares según el problema que enfrentemos, lo que le quita generalidad.

Un segundo problema está relacionado directamente con la facilidad para obtener las derivadas de la función $f(t, y)$. Aún cuando se pueda probar que $f(t, y) \in C^{n-1}[a, b]$, puede ser muy complicado obtener las derivadas de mayor orden, perdiéndose la capacidad de obtener rápidamente una aproximación de la solución buscada.

Es por eso que existen otros métodos para aproximar la solución de una ecuación diferencial que consiguen órdenes de convergencia similares a los de Taylor pero que no requieren la obtención de las derivadas de la función $f(t, y)$. Son los denominados *métodos de Runge-Kutta*.

Para poder construir los métodos de Runge-Kutta, nos basaremos en el siguiente teorema.

Teorema 7.4. Sea $f(t, y) \in C^{n+1} D$ con $D = \{(t, y) | a \leq t \leq b, c \leq y \leq d\}$, y sea $(t_0, y_0) \in D$. Entonces, para toda $(t, y) \in D$, existe $\xi \in [t_0, t]$ y $\mu \in [y_0, y]$ con

$$f(t, y) = P_n(t, y) + R_n(t, y),$$

tal que

$$\begin{aligned}
 P_n(t, y) = & f(t_0, y_0) + \left[(t - t_0) \frac{\partial f(t_0, y_0)}{\partial t} + (y - y_0) \frac{\partial f(t_0, y_0)}{\partial y} \right] + \\
 & + \left[\frac{(t - t_0)^2}{2!} \frac{\partial^2 f(t_0, y_0)}{\partial t^2} + (t - t_0)(y - y_0) \frac{\partial^2 f(t_0, y_0)}{\partial t \partial y} + \right. \\
 & \left. + \frac{(y - y_0)^2}{2!} \frac{\partial^2 f(t_0, y_0)}{\partial y^2} \right] + \dots + \\
 & + \left[\frac{1}{n!} \sum_{j=0}^n \binom{n}{j} (t - t_0)^{n-j} (y - y_0)^j \frac{\partial^n f(t_0, y_0)}{\partial t^{n-j} \partial y^j} \right],
 \end{aligned}$$

y

$$R_n(t, y) = \frac{1}{(n+1)!} \sum_{j=0}^{n+1} \binom{n+1}{j} (t - t_0)^{n+1-j} (y - y_0)^j \frac{\partial^{n+1} f(\xi, \mu)}{\partial t^{n+1-j} \partial y^j}.$$

A la función $P_n(t, y)$ se la denomina *polinomio de Taylor de grado n en dos variables* para la función $f(t, y)$ alrededor de (t_0, y_0) , en tanto que $R_n(t, y)$ es el residuo o error asociado a $P_n(t, y)$.

Esto es necesario pues los *Métodos de Runge-Kutta* se basan en aproximar el polinomio de Taylor para una variable mediante polinomios de Taylor de dos variables. (Para más detalles de cómo se obtiene esta aproximación, ver [3].)

Existen varios *Métodos de Runge-Kutta* que se clasifican según del orden de convergencia. Los más sencillos son los de orden 2, los que obtenemos a partir de del método de Taylor de segundo orden si proponemos lo siguiente:

$$a_1 f(t + \alpha, y + \beta) = f(t, y) + \frac{h}{2} \frac{\partial f(t, y)}{\partial t} + \frac{h}{2} \frac{\partial f(t, y)}{\partial y} f(t, y). \quad (7.26)$$

Si desarrollamos $f(t + \alpha, y + \beta)$ por Taylor para dos variables, tenemos:

$$f(t + \alpha, y + \beta) = f(t, y) + \alpha \frac{\partial f(t, y)}{\partial t} + \beta \frac{\partial f(t, y)}{\partial y} + \dots \quad (7.27)$$

Ahora reemplacemos (7.27) por $f(t + \alpha, y + \beta)$. Nos queda lo siguiente:

$$a_1 f(t, y) + a_1 \alpha \frac{\partial f(t, y)}{\partial t} + a_1 \beta \frac{\partial f(t, y)}{\partial y} = f(t, y) + \frac{h}{2} \frac{\partial f(t, y)}{\partial t} + \frac{h}{2} \frac{\partial f(t, y)}{\partial y} f(t, y). \quad (7.28)$$

Si igualamos los términos equivalentes de cada miembro obtenemos:

$$a_1 f(t, y) = f(t, y) \Rightarrow a_1 = 1 \quad (7.29)$$

$$a_1 \alpha \frac{\partial f(t, y)}{\partial t} = \frac{h}{2!} \frac{\partial f(t, y)}{\partial t} \Rightarrow \alpha = \frac{h}{2} \quad (7.30)$$

$$a_1 \beta \frac{\partial f(t, y)}{\partial y} = \frac{h}{2} \frac{\partial f(t, y)}{\partial y} f(t, y) \Rightarrow \beta = \frac{h}{2} f(t, y). \quad (7.31)$$

Así obtenemos una expresión equivalente al término $T^{(2)}(t_i, y_i)$ de nuestra aproximación por polinomios de Taylor, sin necesidad de calcular las derivadas de $f(t, y)$:

$$T^{(2)}(t_i, y_i) = f \left[t_i + \frac{h}{2}, y_i + \frac{h}{2} f(t_i, y_i) \right], \quad (7.32)$$

y, entonces, nuestra aproximación por métodos de Taylor de orden superior podemos escribirla así:

$$\begin{aligned} y_{i+1} &\approx y_i + h \cdot T^{(2)} f(t, y) \\ y_{i+1} &\approx y_i + h \cdot f \left[t_i + \frac{h}{2}, y_i + \frac{h}{2} f(t_i, y_i) \right] \\ w_{i+1} &= w_i + h \cdot f \left[t_i + \frac{h}{2}, w_i + \frac{h}{2} f(t_i, w_i) \right], \end{aligned} \quad (7.33)$$

para $i = 0; 1; 2; \dots; n - 1$. Este método se conoce como *Método del Punto Medio*.

Si operamos de forma similar, pero proponiendo que $a_1 \cdot f(t_i, y_i) + a_2 \cdot f(t_i + \alpha, y_i + \beta) = T^{(2)}(t_i, y_i)$, podemos obtener otros métodos de orden 2. Los más conocidos son:

1. **Método de Euler Modificado.** También llamado *Método de Euler Mejorado*, su formulación es:

$$\begin{aligned} w_0 &= y_0 \\ w_{i+1} &= w_i + \frac{h}{2} \left\{ f(t_i, w_i) + f[t_i + h, w_i + h f(t_i, w_i)] \right\}, \end{aligned} \quad (7.34)$$

cuando $a_1 = a_2 = \frac{1}{2}$, $\alpha = h$ y $\beta = h \cdot f(t_i, w_i)$, para $i = 0; 1; 2; \dots; n - 1$.

2. **Método de Heun.** Su formulación es:

$$\begin{aligned} w_0 &= y_0 \\ w_{i+1} &= w_i + \frac{h}{4} \left\{ f(t_i, w_i) + 3f \left[t_i + \frac{2}{3}h, w_i + \frac{2}{3}h f(t_i, w_i) \right] \right\}, \end{aligned} \quad (7.35)$$

cuando $a_1 = \frac{1}{4}$, $a_2 = \frac{3}{4}$, $\alpha = \frac{2}{3}h$ y $\beta = \frac{2}{3}h \cdot f(t_i, w_i)$, para $i = 0; 1; 2; \dots; n - 1$.

Otro método conocido incluido en los *Métodos de Runge-Kutta de orden 2* es el *Método implícito ponderado o de Crank-Nicolson*, cuya formulación es

$$\begin{aligned} w_0 &= y_0 \\ w_{i+1} &= w_i + \frac{h}{2} \left[f(t_i, w_i) + f(t_{i+1}, w_{i+1}) \right], \end{aligned} \quad (7.36)$$

para $i = 0; 1; 2; \dots; n - 1$.

Paralelamente, los métodos del *Punto Medio* y de *Crank-Nicolson* podemos obtenerlos también integrando la función $f(t, y)$ en el intervalo $[t_i, t_{i+1}]$, si aplicamos los métodos de integración numérica del rectángulo y del trapecio respectivamente. En efecto, si partimos de la ecuación

$$dy = f(t, y)dt,$$

e integramos, obtenemos

$$\int_{y_i}^{y_{i+1}} dy = \int_{t_i}^{t_{i+1}} f(t, y)dt,$$

que podemos escribir así

$$y(t_{i+1}) = y(t_i) + \int_{t_i}^{t_{i+1}} f(t, y)dt.$$

Si aplicamos el método del rectángulo para aproximar la integral, obtenemos

$$y(t_{i+1}) = y(t_i) + h \cdot f \left[t_i + \frac{h}{2}; y(t_i) + \frac{h}{2} f(t_i, y(t_i)) \right],$$

por lo que la aproximación podemos escribirla como

$$w_{i+1} = w_i + h \cdot f \left[t_i + \frac{h}{2}, w_i + \frac{h}{2} f(t_i, w_i) \right].$$

De forma análoga, si aplicamos el método del trapecio obtenemos

$$y(t_{i+1}) = y(t_i) + \frac{h}{2} \left[f(t_i, y(t_i)) + f(t_{i+1}, y(t_{i+1})) \right],$$

y nuestra nueva aproximación podemos escribirla como

$$w_{i+1} = w_i + \frac{h}{2} [f(t_i, w_i) + f(t_{i+1}, w_{i+1})].$$

Para obtener métodos de mayor orden de convergencia, debemos aplicar el teorema 7.4. Con él obtenemos podemos obtener un método de *Runge-Kutta de orden 3*:

$$\begin{aligned} w_0 &= y_0 \\ k_1 &= h f(t_i, w_i) \\ k_2 &= h f\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1\right) \\ k_3 &= h f\left(t_i + h, w_i - k_1 + 2k_2\right) \\ w_{i+1} &= w_i + \frac{1}{6}(k_1 + 4k_2 + k_3), \end{aligned} \tag{7.37}$$

para $i = 0; 1; 2; \dots; n - 1$, y el de *Heun de tercer orden* (también incluido en los métodos de Runge-Kutta de orden 3)

$$\begin{aligned} w_0 &= y_0 \\ k_1 &= h f(t_i, w_i) \\ k_2 &= h f\left(t_i + \frac{h}{3}, w_i + \frac{1}{3}k_1\right) \\ k_3 &= h f\left(t_i + \frac{2}{3}h, w_i + \frac{2}{3}k_2\right) \\ w_{i+1} &= w_i + \frac{1}{4}(k_1 + 3k_3), \end{aligned} \tag{7.38}$$

para $i = 0; 1; 2; \dots; n - 1$.

Uno de los métodos más usados para resolver ecuaciones diferenciales ordinarias, el de *Runge-Kutta de orden 4*, cuya formulación es la siguiente:

$$\begin{aligned} w_0 &= y_0 \\ k_1 &= h f(t_i, w_i) \\ k_2 &= h f\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1\right) \\ k_3 &= h f\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2\right) \\ k_4 &= h f(t_i + h, w_i + k_3) \\ w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \end{aligned} \tag{7.39}$$

también para $i = 0; 1; 2; \dots; n - 1$.

Los métodos de Runge-Kutta de orden 3 y 4 tiene un error local de truncamiento $O(h^3)$ y $O(h^4)$, siempre que la función $y(t)$ tenga al menos cuatro y cinco derivadas continuas, respectivamente.

Este método es tan preciso, que programas como el MatLab[®] y el Mathcad[®] tienen desarrollados distintas funciones que aplican este método. Por ejemplo, Mathcad[®] cuenta con la función `rkfixed`($y, x_1, x_2, npoints, D$) que resuelve ecuaciones diferenciales de primer orden utilizando dicho método, en la cual y es el valor inicial, x_1 y x_2 son los extremos del intervalo, $npoints$ es la cantidad de intervalos, y entonces $h = \frac{x_2 - x_1}{npoints}$, y D es la función $f(x, y)$ que debemos resolver.

Este método puede asociarse a la siguiente formulación:

$$w_{i+1} = w_i + \frac{h}{6} \left[f(t_i, w_i) + 4f\left(t_{i+\frac{1}{2}}, w_{i+\frac{1}{2}}\right) + f(t_{i+1}, w_{i+1}) \right],$$

equivalente al método de Simpson de integración numérica, cuya convergencia es $O(h^4)$.

7.1.9. Métodos de paso múltiple

Los métodos anteriores se basan en obtener los valores siguientes utilizando solamente el valor anterior, sin tener en cuenta los demás valores ya calculados. Es por eso que se denominan de *paso simple*. Pero la pregunta que nos podemos hacer es: si estamos tratando de aproximar una función, tal que se cumpla que $\frac{dy}{dt} = f(t, y)$, por qué no utilizar el conjunto de los valores obtenidos, o al menos un grupo de ellos, para obtener los puntos siguientes.

Esa idea es la que domina a los denominados *métodos de paso múltiple*. El método más sencillo es el denominado *método del salto de rana*, cuya expresión es

$$\begin{aligned} w_0 &= y_0 \\ w_{i+1} &= w_{i-1} + 2h f(t_i, w_i), \end{aligned} \tag{7.40}$$

para $i = 1; 2; \dots; n - 1$. El valor de w_1 debemos calcularlo con otro método. Como las aproximaciones que obtenemos por el método del salto de rana son del mismo orden que las que se obtienen por cualquier método de Runge-Kutta de orden 2, es conveniente aproximar w_1 con alguno de esos métodos. De todos modos, el método del salto de rana es inestable para determinado tipo de ecuaciones diferenciales.

Existen otros métodos, muy utilizados, que mejoran la notoriamente la aproximación que podemos obtener.

Métodos de Adams

A partir de esta idea, de utilizar la información de los puntos calculados, se han desarrollado métodos de paso múltiple muy utilizados, los métodos de Adams. Los podemos dividir en dos grupos: los métodos explícitos, o de *Adams-Bashforth*, y los métodos implícitos, o de *Adams-Moulton*.

En ambos casos, la idea es usar los puntos $w_i, w_{i-1}, \dots, w_{i+1-p}$ para obtener el w_{i+1} , en el caso de los métodos de Adams-Bashforth, en tanto que en los de Adams-Moulton se usan los $w_{i+1}, w_i, w_{i-1}, \dots, w_{i+2-p}$, donde p es el orden de convergencia. Así, un método de Adams-Bashforth de orden 2 usa los puntos w_i y w_{i-1} , en tanto que un método de Adams-Moulton usa los puntos w_{i+1} y w_i . Veamos como obtener algunos de estos métodos.

Métodos de Adams-Bashforth

Para obtener el *Método de Adams-Bashforth de orden 2* partamos de:

$$\int_{y_i}^{y_{i+1}} dy = \int_{t_i}^{t_{i+1}} f(t, y) dt, \quad (7.41)$$

$$y(t_{i+1}) - y(t_i) = \int_{t_i}^{t_{i+1}} f(t, y) dt, \quad (7.42)$$

$$y(t_{i+1}) = y(t_i) + \int_{t_i}^{t_{i+1}} f(t, y) dt. \quad (7.43)$$

Para aproximar $\int_{t_i}^{t_{i+1}} f(t, y) dt$, armemos un polinomio interpolante, utilizando el método de Newton de diferencias divididas regresivas, que aproxime $f(t, y)$. Así, nos queda que

$$\begin{aligned} f(t, y) &\approx f(t_i, y(t_i)) + \frac{f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))}{t_i - t_{i-1}}(t - t_i) \\ &\approx f(t_i, y(t_i)) + \frac{f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))}{h}(t - t_i). \end{aligned} \quad (7.44)$$

Al integrar el polinomio interpolante obtenemos

$$\begin{aligned} \int_{t_i}^{t_{i+1}} f(t, y) dt &\approx h \cdot f(t_i, y(t_i)) + h \frac{f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))}{2}. \\ &\approx \frac{h}{2} [3f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))]. \end{aligned} \quad (7.45)$$

y si reemplazamos en la expresión inicial, tenemos

$$y(t_{i+1}) \approx y(t_i) + \frac{h}{2} [3f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))]. \quad (7.46)$$

Como siempre, lo que buscamos es una aproximación de $y(t_{i+1})$, entonces el *Método de Adams-Bashforth de orden 2* queda formulado de la siguiente manera:

$$w_{i+1} = w_i + \frac{h}{2} [3f(t_i, w_i) - f(t_{i-1}, w_{i-1})], \quad (7.47)$$

para $i = 1; 2; \dots; n - 1$. Por lo tanto, debemos calcular w_1 con algún otro método. Dado que el método es de orden de convergencia 2, lo más conveniente es usar algún *Método de Runge-Kutta de orden 2*.

Otro método es el de *Adams-Bashforth de orden 3*:

$$w_{i+1} = w_i + \frac{h}{12} [23f(t_i, w_i) - 16f(t_{i-1}, w_{i-1}) + 5f(t_{i-2}, w_{i-2})], \quad (7.48)$$

para $i = 2; 3; \dots; n - 1$. En este caso, debemos hallar w_1 y w_2 con ayuda del método de Runge-Kutta de orden 3.

También tenemos el *Método de Adams-Bashforth de orden 4*, cuya expresión es:

$$w_{i+1} = w_i + \frac{h}{24} [55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})], \quad (7.49)$$

para $i = 3; 4; \dots; n - 1$. Nuevamente, debemos hallar w_1 , w_2 y w_3 con ayuda de otro método. Al igual que en el método de orden 2, en este caso podemos usar el *Método de Runge-Kutta de orden 4*.

Métodos de Adams-Moulton

Los *Métodos de Adams-Moulton* los obtenemos de forma análoga a la aplicada para los *Métodos de Adams-Bashforth*. Por ejemplo, para obtener el de orden 2, planteemos el siguiente polinomio interpolante para aproximar $f(t, y)$:

$$\begin{aligned} f(t, y) &\approx f(t_{i+1}, y(t_{i+1})) + \frac{f(t_{i+1}, y(t_{i+1})) - f(t_i, y(t_i))}{t_{i+1} - t_i} (t - t_{i+1}) \\ &\approx f(t_{i+1}, y(t_{i+1})) + \frac{f(t_{i+1}, y(t_{i+1})) - f(t_i, y(t_i))}{h} (t - t_{i+1}). \end{aligned} \quad (7.50)$$

Al integrarlo, obtenemos:

$$\begin{aligned} \int_{t_i}^{t_{i+1}} f(t, y) dt &\approx h f(t_{i+1}, y(t_{i+1})) - h \frac{f(t_{i+1}, y(t_{i+1})) - f(t_i, y(t_i))}{2}, \\ &\approx \frac{h}{2} [f(t_{i+1}, y(t_{i+1})) + f(t_i, y(t_i))]. \end{aligned} \quad (7.51)$$

Nuevamente, al reemplazar en la expresión inicial, tenemos

$$y(t_{i+1}) = y(t_i) + \frac{h}{2} [f(t_{i+1}, y(t_{i+1})) + f(t_i, y(t_i))], \quad (7.52)$$

y como lo que buscamos es una aproximación de $y(t_{i+1})$, tenemos que

$$w_{i+1} = w_i + \frac{h}{2} [f(t_{i+1}, w_{i+1}) + f(t_i, w_i)]. \quad (7.53)$$

Resulta interesante ver que el *Método de Adams-Moulton de orden 2* es el método de Crank-Nicolson.

El *Método de Adams-Moulton de orden 3* es el siguiente:

$$w_{i+1} = w_i + \frac{h}{12} [5f(t_{i+1}, w_{i+1}) + 8f(t_i, w_i) - f(t_{i-1}, w_{i-1})], \quad (7.54)$$

para $i = 1; 2; \dots; n - 1$. En este caso debemos obtener w_1 con ayuda del método de Runge-Kutta de orden 3.

Uno de los métodos más usados es el de *Adams-Moulton de orden 4*, cuya expresión es

$$w_{i+1} = w_i + \frac{h}{24} [9f(t_{i+1}, w_{i+1}) + 19f(t_i, w_i) - 5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})], \quad (7.55)$$

para $i = 2; 3; \dots; n - 1$. Nuevamente, debemos obtener w_1 y w_2 con ayuda del *Método de Runge-Kutta de orden 4*.

En general, suelen ser más precisos los métodos de *Adams-Moulton* que los de *Adams-Bashforth*. El de *Adams-Moulton* de orden 4 entrega resultados muy parecidos, en precisión, al *Método de Runge-Kutta de orden 4*. Sin embargo, por una cuestión de sencillez al momento de programar, los paquetes de software prefieren incluir este último y no el método de *Adams-Moulton de orden 4*. Una excepción parece ser la versión 14 del Mathcad[®], que dispone de una función `Adams(init, x1, x2, npoints, D, [tol])` para resolver ecuaciones diferenciales, que de acuerdo con la ayuda del programa, utiliza métodos de Adams-Bashforth, aunque no especifica el orden. Los parámetros de la función son:

init: Valor inicial de la función;

x1, x2: Extremos del intervalo;

npoints: Cantidad de puntos a obtener (equivalente al paso h);

D: Ecuación diferencial escrita en forma vectorial, permite resolver también un sistema de ecuaciones diferenciales;

tol: Tolerancia.

También cuenta con una función para resolver *Ecuaciones Diferenciales Ordinarias* que aplica *métodos de Adams* en forma predefinida (`Odesolve([vector], x, b, [step])`). Todas estas funciones entregan, además, una curva obtenida por interpolación de los valores calculados.

Métodos de las Diferencias Regresivas

Existe otro grupo de métodos multipasos conocido como *Métodos de las Diferencias Regresivas*. Son métodos implícitos, como los *Adams-Moulton*, pero con la particularidad de que se obtienen a partir de aproximar la derivada primera en el punto t_{i+1} con ayuda de puntos anteriores y una interpolación polinomial que aproxima $y(t)$ y no $f(t, y)$ como en el caso de los *Métodos de Adams*.

Estos métodos surgieron para resolver algunos problemas de ecuaciones diferenciales en los cuales los métodos tradicionales fallaban: los problemas denominados «rígidos».

Al igual que en el caso de *Métodos de Adams*, existen varios algoritmos en función del orden de convergencia. Así, para obtener el *Método de las Diferencias Regresivas de orden uno*, $O(h)$, basta con obtener el polinomio interpolante entre los puntos t_i, y_i y t_{i+1}, y_{i+1} :

$$P_1(t) = y_i \frac{t - t_{i+1}}{t_i - t_{i+1}} + y_{i+1} \frac{t - t_i}{t_{i+1} - t_i} = -\frac{y_i}{h}(t - t_{i+1}) + \frac{y_{i+1}}{h}(t - t_i). \quad (7.56)$$

Si derivamos $P_1(t)$ obtenemos:

$$\frac{dP_1(t)}{dt} = -\frac{y_i}{h} + \frac{y_{i+1}}{h}, \quad (7.57)$$

que aproxima $f(t, y)$. Por lo tanto, podemos aproximar $f(t_{i+1}, y_{i+1})$ y obtenemos

$$f(t_{i+1}, y_{i+1}) \approx \frac{dP_1(t_{i+1})}{dt} = -\frac{y_i}{h} + \frac{y_{i+1}}{h} \quad (7.58)$$

$$y_{i+1} \approx y_i + h f(t_{i+1}, y_{i+1}), \quad (7.59)$$

que no es otro que el *Método de Euler Implícito*, cuyo término de error es:

$$E(h) = \frac{f'(\xi, y(\xi))}{2} h, \quad (7.60)$$

con $\xi \in (t_i, t_{i+1})$.

Para mejorar nuestro algoritmo, agreguemos un tercer punto, (t_{i-1}, y_{i-1}) . Con él podemos obtener un nuevo polinomio interpolante:

$$P_2(t) = y_{i-1} \frac{(t - t_i)(t - t_{i+1})}{(t_{i-1} - t_i)(t_{i-1} - t_{i+1})} + y_i \frac{(t - t_{i-1})(t - t_{i+1})}{(t_i - t_{i-1})(t_i - t_{i+1})} + y_{i+1} \frac{(t - t_{i-1})(t - t_i)}{(t_{i+1} - t_{i-1})(t_{i+1} - t_i)}. \quad (7.61)$$

Al igual que en el caso anterior, derivemos este nuevo polinomio:

$$\begin{aligned} \frac{dP_2(t)}{dt} &= \frac{y_{i-1}}{2h^2} [(t - t_i) + (t - t_{i+1})] - \frac{y_i}{h^2} [(t - t_{i-1}) + (t - t_{i+1})] + \\ &+ \frac{y_{i+1}}{2h^2} [(t - t_{i-1}) + (t - t_i)]. \end{aligned} \quad (7.62)$$

Ahora obtengamos la derivada en el punto t_{i+1} . Si tomamos un paso constante, es decir, $h = t_{i+1} - t_i = t_i - t_{i-1}$, la derivada la podemos expresar como

$$\left. \frac{dP_2(t)}{dt} \right|_{t_{i+1}} = \frac{y_{i-1}}{2h^2}(h+0) - \frac{y_i}{h^2}(2h+0) + \frac{y_{i+1}}{2h^2}(2h+h), \quad (7.63)$$

que simplificada y reagrupada nos queda así:

$$\left. \frac{dP_2(t_{i+1})}{dt} \right|_{t_{i+1}} = \frac{y_{i-1}}{2h} - \frac{2y_i}{h} + \frac{3y_{i+1}}{2h}. \quad (7.64)$$

Esta derivada no es otra cosa que la aproximación de $f(t_{i+1}, y_{i+1})$ por lo tanto podemos armar la expresión anterior de esta otra manera,

$$f(t_{i+1}, y_{i+1}) = \frac{y_{i-1}}{2h} - \frac{2y_i}{h} + \frac{3y_{i+1}}{2h}, \quad (7.65)$$

y despejar y_{i+1} :

$$y_{i+1} = -\frac{y_{i-1}}{3} + \frac{4y_i}{3} + \frac{2}{3}h f(t_{i+1}, y_{i+1}). \quad (7.66)$$

El error de la aproximación polinomial está dado por:

$$E_P(t) = \frac{y'''(\xi)}{3!}(t-t_{i-1})(t-t_i)(t-t_{i+1}). \quad (7.67)$$

Si derivamos esta expresión, obtenemos el error de nuestra aproximación de la derivada primera con el polinomio interpolante:

$$\frac{dE_P(t)}{dt} = \frac{y'''(\xi)}{3!}[(t-t_{i-1})(t-t_i) + (t-t_{i-1})(t-t_{i+1}) + (t-t_i)(t-t_{i+1})]. \quad (7.68)$$

Reemplacemos t por t_{i+1} , y tendremos el error de la derivada en ese punto. Si además recordamos que $h = t_{i+1} - t_i = t_i - t_{i-1}$, la expresión final queda de esta forma:

$$\left. \frac{dE_P(t)}{dt} \right|_{t_{i+1}} = \frac{y'''(\xi)}{3!}[(2h)(h) + (2h)(0) + (h)(0)], \quad (7.69)$$

que al agrupar en función de h , y dado que $y'''(\xi) = f''(\xi, y(\xi))$, podemos expresar como

$$E(h) = \frac{f''(\xi, y(\xi))}{3}h^2, \quad (7.70)$$

con $\xi \in (t_{i-1}, t_{i+1})$. Esto confirma que el método es de convergencia cuadrática. Por lo tanto, y como lo que estamos obteniendo son aproximación de y_i , el método lo escribimos así:

$$w_{i+1} = -\frac{w_{i-1}}{3} + \frac{4w_i}{3} + \frac{2}{3}h f(t_{i+1}, w_{i+1}). \quad (7.71)$$

Así como obtuvimos un método de convergencia cuadrática, podemos obtener métodos con órdenes de convergencia mayores. Entre ellos podemos destacar a los siguientes:

Método de orden 3:

$$w_{i+1} = \frac{2w_{i-2}}{11} - \frac{9w_{i-1}}{11} + \frac{18w_i}{11} + \frac{6}{11}h f(t_{i+1}, w_{i+1}), \quad (7.72)$$

Método de orden 4:

$$w_{i+1} = -\frac{3w_{i-3}}{25} + \frac{16w_{i-2}}{25} - \frac{36w_{i-1}}{25} + \frac{48w_i}{25} + \frac{12}{25}h f(t_{i+1}, w_{i+1}). \quad (7.73)$$

Los errores para estos métodos son:

$$E(h) = \frac{f'''(\xi; y(\xi))}{4} h^3, \text{ con } \xi \in (t_{i-2}, t_{i+1}) \text{ y;} \quad (7.74)$$

$$E(h) = \frac{f^{iv}(\xi; y(\xi))}{5} h^4 \text{ con } \xi \in (t_{i-3}, t_{i+1}); \quad (7.75)$$

respectivamente.

Estos métodos suelen ser muy precisos a pesar de que muchas veces requieren algún tipo de solución iterativa cuando la derivada $f(t, y)$ no es una función lineal.

7.1.10. Métodos predictores-correctores

Método predictor-corrector de Adams

Como hemos visto, el uso de los *Métodos de Adams-Moulton* conlleva la necesidad de reformular la expresión para convertirla en un método explícito. Como esto no siempre es posible, y gracias a la idea de los métodos predictores-correctores, una forma de aplicarlos es mediante la combinación de un *Método de Adams-Bashforth* y uno de *Adams-Moulton*, ambos del mismo orden de convergencia. Esta combinación se conoce como *Método Predictor-Corrector de Adams*. Por ejemplo, el *Método Predictor-Corrector de Adams de orden 2* es el siguiente:

$$\begin{aligned} w_{i+1}^* &= w_i + \frac{h}{2} [3f(t_i, w_i) - f(t_{i-1}, w_{i-1})] \\ w_{i+1} &= w_i + \frac{h}{2} [f(t_{i+1}, w_{i+1}^*) + f(t_i, w_i)], \end{aligned} \quad (7.76)$$

para $i = 1; 2; \dots; n - 1$ y donde el valor de w_1 debemos obtenerlo usando el *Método de Runge-Kutta de orden 2* o resolviendo en forma explícita la segunda ecuación del método. Uno de los métodos más usados es el *Predictor-Corrector de Adams de orden 4*, cuya expresión es:

$$\begin{aligned} w_{i+1}^* &= w_i + \frac{h}{24} [55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})] \\ w_{i+1} &= w_i + \frac{h}{24} [9f(t_{i+1}, w_{i+1}^*) + 19f(t_i, w_i) - 5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})], \end{aligned} \quad (7.77)$$

para $i = 3; 4; \dots; n - 1$ y donde w_1, w_2 y w_3 los obtenemos usando el *Método de Runge-Kutta de Orden 4*.

Al igual que lo visto para el *Método Predictor-Corrector de Euler*, en estos métodos también cabe la posibilidad de iterar con la fórmula correctora hasta obtener la solución buscada. Por ejemplo, el *Método Predictor-Corrector de Adams de orden 2* podemos escribirlo como:

$$\begin{aligned} w_{i+1}^* &= w_i + \frac{h}{2} [3f(t_i, w_i) - f(t_{i-1}, w_{i-1})] \\ w_{i+1}^0 &= w_i + \frac{h}{2} [f(t_{i+1}, w_{i+1}^*) + f(t_i, w_i)], \\ w_{i+1}^{n+1} &= w_i + \frac{h}{2} [f(t_{i+1}, w_{i+1}^n) + f(t_i, w_i)], \end{aligned} \quad (7.78)$$

en tanto que al *Método Predictor-Corrector de Adams de orden 4* lo podemos escribir así:

$$\begin{aligned} w_{i+1}^* &= w_i + \frac{h}{24} [55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})] \\ w_{i+1}^0 &= w_i + \frac{h}{24} [9f(t_{i+1}, w_{i+1}^*) + 19f(t_i, w_i) - 5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})], \\ w_{i+1}^{n+1} &= w_i + \frac{h}{24} [9f(t_{i+1}, w_{i+1}^n) + 19f(t_i, w_i) - 5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})]. \end{aligned} \quad (7.79)$$

En ambos casos, las iteraciones las truncamos cuando

$$|w_{i+1}^{n+1} - w_{i+1}^n| < TOL$$

o

$$\left| \frac{w_{i+1}^{n+1} - w_{i+1}^n}{w_{i+1}^{n+1}} \right| < TOL.$$

Método predictor-corrector de Milne

Existe otro método predictor-corrector multipaso muy conocido. Es el *Método Predictor-Corrector de Milne*, cuya formulación es:

$$\begin{aligned} w_{i+1}^* &= w_{i-3} + \frac{4}{3}h \cdot \left[2f(t_i, w_i) - f(t_{i-1}, w_{i-1}) + 2f(t_{i-2}, w_{i-2}) \right], \\ w_{i+1}^0 &= w_{i-1} + \frac{h}{3} \left[f(t_{i+1}, w_{i+1}^*) + 4 \cdot f(t_i, w_i) + f(t_{i-1}, w_{i-1}) \right], \\ w_{i+1}^{n+1} &= w_{i-1} + \frac{h}{3} \left[f(t_{i+1}, w_{i+1}^n) + 4 \cdot f(t_i, w_i) + f(t_{i-1}, w_{i-1}) \right]. \end{aligned} \quad (7.80)$$

en el que también truncamos las iteraciones con los criterios ya vistos. Este método es de orden cuatro, pero no es de los más usados pues los resultados no son mejores que los obtenidos con los *Métodos Predictores-Correctores de Adams*. (Para más detalles, ver [3].)

7.2. Análisis de estabilidad

Uno de los temas que tienen singular importancia en la resolución numérica de ecuaciones diferenciales es la estabilidad de los métodos. Sin embargo, en este caso, el concepto de estabilidad no está ligado al error de redondeo sino al «tamaño» del paso de cálculo.

Supongamos que tomamos una ecuación sencilla como la siguiente:

$$y'(t) = \lambda \cdot y \quad \text{con } a \leq t \leq b \quad \text{y } y(a) = y_0. \quad (7.81)$$

Planteemos el *Método de Euler Explícito* como método de resolución. El esquema quedará de esta forma:

$$y_{i+1} \approx y_i + h \lambda y_i,$$

que al agrupar queda:

$$y_{i+1} \approx (1 + h \lambda) y_i.$$

Como esto se repite para todas las iteraciones, podemos expresar cualquier iteración y_{i+1} en función del valor inicial y_0 :

$$y_{i+1} \approx (1 + h \lambda)^{i+1} y_0, \quad (7.82)$$

de manera que cualquier valor y_{i+1} es el producto de y_0 por un factor constante que depende del paso h y de λ . Como este último es dato del problema, existen dos posibilidades:

1. $\lambda > 0$: En este caso $y_{i+1} > y_i$, por lo que el error absoluto crece junto con y_{i+1} , en cambio el error relativo tiende a ser estable. En efecto, si suponemos que

$$y_{i+1} + \varepsilon_{i+1} = (1 + h \lambda) (y_i + \varepsilon_i), \quad (7.83)$$

y si simplificamos la expresión nos queda

$$\varepsilon_{i+1} = (1 + h \lambda) \varepsilon_i. \quad (7.84)$$

Como podemos ver, el crecimiento del error es similar al crecimiento de y . Y si analizamos el error relativo tenemos lo siguiente:

$$\frac{\varepsilon_{i+1}}{y_{i+1}} = \frac{(1+h\lambda)\varepsilon_i}{y_{i+1}} = \frac{(1+h\lambda)\varepsilon_i}{(1+h\lambda)y_i} = \frac{\varepsilon_i}{y_i}, \quad (7.85)$$

vemos que tiende a ser constante, lo que no trae aparejado ningún inconveniente grave.

2. $\lambda < 0$: En este caso, se cumple que $|y_{i+1}| \leq |y_i|$ y en consecuencia debe cumplirse que

$$|1 - h\lambda| \leq 1. \quad (7.86)$$

Esto podemos formularlo así:

$$-1 \leq 1 - h\lambda \leq 1 \rightarrow h\lambda \leq 2 \Rightarrow h \leq \frac{2}{\lambda}. \quad (7.87)$$

Es evidente que en este caso no podemos elegir cualquier « h »; dependerá de λ . Estamos ante un caso en que la estabilidad está condicionada.

Analicemos ahora el caso del *Método de Euler Implícito*. Análogamente, al caso anterior tendremos que

$$\begin{aligned} y_{i+1} &\approx y_i + h\lambda y_{i+1}, \\ y_{i+1} - h\lambda y_{i+1} &\approx y_i \Rightarrow \\ y_{i+1} &\approx \frac{y_i}{1 - h\lambda}. \end{aligned} \quad (7.88)$$

Nuevamente, cualquier y_{i+1} lo podemos expresar en función de y_0 , por lo tanto nos queda que

$$y_{i+1} \approx \frac{1}{(1 - h\lambda)^{i+1}} y_0. \quad (7.89)$$

También podemos efectuar un análisis de los resultados en función del λ , que resulta en dos posibilidades otra vez:

1. $\lambda > 0$: También se cumple que $y_{i+1} > y_i$, por lo que el error absoluto crece junto con y_{i+1} , y el error relativo tiende a ser estable.
2. $\lambda < 0$: También se cumple que $|y_{i+1}| \leq |y_i|$ y en consecuencia debe cumplirse que

$$\frac{1}{|1 + h\lambda|} \leq 1. \quad (7.90)$$

Es evidente que esta condición se cumple para cualquier valor de « h » (pues $h > 0$), y por lo tanto, la elección del paso no está condicionada.

Si hacemos un análisis similar para los métodos de *Euler Modificado* y de *Crank-Nicolson*, ambos *Métodos de Runge-Kutta de Orden 2*, obtenemos resultados parecidos. Para el método de *Euler Modificado*, que es explícito, si $\lambda > 0$, no tenemos inconvenientes con el paso, y cuando $\lambda < 0$, tenemos que

$$h \leq \frac{2}{\lambda},$$

es decir, el paso está condicionado al valor de λ . En el caso del método de *Crank-Nicolson*, que es implícito, la situación cuando $\lambda > 0$ es análoga al caso de *Euler Modificado*, y cuando $\lambda < 0$ tenemos que

$$h \geq -\frac{2}{\lambda} \text{ y } h \geq 0.$$

Esta condición se cumple para cualquier valor de « h ».

Si bien deberíamos hacer un análisis para cada método, con lo hecho podemos sacar algunas conclusiones importantes:

- Los métodos explícitos son más sencillos para trabajar pero el paso depende de las características de la ecuación a resolver;
- Los métodos implícitos son algo más complicados para operar, pero no tienen complicación alguna para la elección del paso.

De este análisis surge otra forma de clasificar a los métodos:

- **Métodos *condicionalmente* estables**, aquellos en los que la estabilidad está condicionada a la elección del paso, generalmente métodos explícitos, y;
- **Métodos *incondicionalmente* estables**, aquellos que no tienen ningún tipo de condicionamiento para su estabilidad, generalmente métodos implícitos.

7.3. Consistencia y convergencia

Por otro lado, también debemos verificar la *consistencia* y la *convergencia* de los métodos respecto de la solución analítica. Un método es *consistente* si

$$\lim_{h \rightarrow 0} \max_{1 \leq i \leq n} |E(h)| = 0,$$

donde $E(h)$ es el error de truncamiento del método. A su vez, la *convergencia* está definida como

$$\lim_{h \rightarrow 0} \max_{1 \leq i \leq n} |w_i - y(t_i)| = 0,$$

donde $w_i - y(t_i)$ es el error global. Como hemos visto, para los métodos de Euler tenemos que

$$|w_i - y(t_i)| \leq \frac{hM}{2L} \left[e^{L(t_i-a)} - 1 \right],$$

por lo que el método es convergente pues se cumple que

$$\lim_{h \rightarrow 0} \frac{hM}{2L} \left[e^{L(t_i-a)} - 1 \right] = 0.$$

De acuerdo con lo visto, para una ecuación diferencial de primer orden con valores iniciales definido por

$$\frac{dy}{dt} = f(t, y); \quad a \leq t \leq b; \quad y(a) = \alpha,$$

que aproximaremos con un método que definiremos así

$$\begin{aligned} w_0 &= \alpha, \\ w_{i+1} &= w_i + h \phi(t_i, w_i, h), \end{aligned}$$

con

$$h > 0 \quad \text{y} \quad \phi(t_i, w_i, h) \in C[a, b].$$

Como vemos esta definición es para cualquier *método de paso simple*. Si $\phi(t_i, w_i, h)$ cumple con la *condición de Lipshitz* en:

$$D = \{(t, w, h) | a \leq t \leq b, -\infty < w < +\infty, 0 \leq h \leq h_0\},$$

entonces:

- El método es *estable*;

- Es convergente si y solo si es consistente, es decir, se cumpla que

$$\phi(t, w, 0) = f(t, w);$$

- Si existe $E(h)$, entonces:

$$|y_i - w_i| \leq \frac{E(h)}{L} e^{(t_i - a)L}$$

En el caso de los métodos multipaso, al analizar la consistencia debemos asegurarnos además que

$$\begin{aligned} \lim_{h \rightarrow 0} |E(h)| &= 0 \quad \text{para } i = m, m+1, \dots, n \\ \lim_{h \rightarrow 0} |\alpha_j - y(t_j)| &= 0 \quad \text{para } j = 1; 2; \dots, m-1. \end{aligned}$$

donde los α_j son los valores adicionales que deben calcularse para empezar a iterar.

7.4. Ecuaciones diferenciales ordinarias de orden superior

Nos hemos ocupado de resolver numéricamente ecuaciones diferenciales ordinarias de primer orden con valores iniciales. Existen también ecuaciones similares pero de orden superior como éstas:

$$\frac{d^2 y}{dt^2} = f(t, y, y') \quad \text{con } a \leq t \leq b, \quad y(a) = \alpha, \quad y'(a) = \beta, \quad (7.91)$$

que es una ecuación de segundo orden.

La forma general de una ecuación de orden superior es:

$$\frac{d^n y}{dt^n} = f(t, y, y', \dots, y^{(n-1)}) \quad \text{con } a \leq t \leq b, \quad y(a) = y_0, \quad y'(a) = y'_0, \quad \dots, \quad y^{(n-1)}(a) = y_0^{(n-1)}.$$

El planteo y la resolución de este tipo de ecuaciones requiere un estudio casi detallado en función del orden de la ecuación. En los puntos siguientes nos concentraremos en las de segundo orden y veremos varias formas de plantear una solución numérica aproximada.

7.4.1. Aplicación de los métodos para ecuaciones diferenciales de primer orden

Una forma de resolver es aplicar cualquiera de los métodos vistos para ecuaciones de primer orden pero efectuando un cambio de variable. Así, para la ecuación de segundo orden, la transformación queda así:

$$\frac{dy}{dt} = z(t, y) \quad (7.92)$$

$$\frac{dz}{dt} = f(t, y, z). \quad (7.93)$$

con las mismas condiciones ya vistas: $a \leq t \leq b$, $y(a) = y_0$ y $z(a) = y'(a) = z_0$.

Por ejemplo, si aplicamos el *Método de Euler Explícito*, y si hacemos $w_i = y_i$ y $v_i = z_i$, la aproximación nos queda de la siguiente forma:

$$w_{i+1} = w_i + h \cdot v_i \quad (7.94)$$

$$v_{i+1} = v_i + h \cdot f(t_i, w_i, v_i). \quad (7.95)$$

Si aplicamos un *Método de Runge-Kutta de orden 2*, por ejemplo, el *método de Euler Modificado*, tenemos:

$$k_1^1 = h \cdot v_i; \quad k_1^2 = h \cdot f(t_i, w_i, v_i) \quad (7.96)$$

$$k_2^1 = h \cdot (v_i + k_1^2); \quad k_2^2 = h \cdot f(t_i + h, w_i + k_1^1, v_i + k_1^2) \quad (7.97)$$

$$w_{i+1} = w_i + \frac{1}{2} (k_1^1 + k_2^1), \quad (7.98)$$

$$v_{i+1} = v_i + \frac{1}{2} (k_1^2 + k_2^2). \quad (7.99)$$

Algo similar podemos hacer con el *Método de Runge-Kutta de orden 4*. El sistema de ecuaciones quedará de esta forma:

$$k_1^1 = h \cdot v_i; \quad k_1^2 = h \cdot f(t_i, w_i, v_i) \quad (7.100)$$

$$k_2^1 = h \cdot (v_i + \frac{1}{2}k_1^2); \quad k_2^2 = h \cdot f(t_i + \frac{1}{2}h, w_i + \frac{1}{2}k_1^1, v_i + \frac{1}{2}k_1^2) \quad (7.101)$$

$$k_3^1 = h \cdot (v_i + \frac{1}{2}k_2^2); \quad k_3^2 = h \cdot f(t_i + \frac{1}{2}h, w_i + \frac{1}{2}k_2^1, v_i + \frac{1}{2}k_2^2) \quad (7.102)$$

$$k_4^1 = h \cdot (v_i + k_3^2); \quad k_4^2 = h \cdot f(t_i + h, w_i + k_3^1, v_i + k_3^2) \quad (7.103)$$

$$w_{i+1} = w_i + \frac{1}{6} (k_1^1 + 2 \cdot k_2^1 + 2 \cdot k_3^1 + k_4^1) \quad (7.104)$$

$$v_{i+1} = v_i + \frac{1}{6} (k_1^2 + 2 \cdot k_2^2 + 2 \cdot k_3^2 + k_4^2). \quad (7.105)$$

Este mismo concepto podemos aplicarlo si queremos usar algún otro método, sea de paso simple, sea de paso múltiple. En estos últimos el esquema es más sencillo, pues lo único que hay que cuidar es la formulación para las distintas ecuaciones. Así, la aplicación del *Método de Adams-Bashforth de orden 2* resulta ser:

$$w_{i+1} = w_i + h \cdot (3 \cdot v_i - v_{i-1}) \quad (7.106)$$

$$v_{i+1} = v_i + h \cdot [3 \cdot f(t_i, w_i, v_i) - f(t_{i-1}, w_{i-1}, v_{i-1})], \quad (7.107)$$

obteniendo los valores de w_{i-1} y v_{i-1} de la misma forma a la vista para el caso de ecuaciones de primer orden.

Evidentemente, resolver ecuaciones diferenciales de orden superior con métodos explícitos no conlleva más que una complicación en la formulación algebraica del método elegido, sobre todo en el caso de los *Métodos de Runge-Kutta*.

Así como analizamos cómo aplicar los métodos explícitos, veamos brevemente qué ocurre con la aplicación de métodos implícitos. Supongamos ahora que para resolver la ecuación diferencial de segundo orden aplicamos el *Método de Euler Implícito*. El esquema quedará así:

$$w_{i+1} = w_i + h \cdot v_{i+1} \quad (7.108)$$

$$v_{i+1} = v_i + h \cdot f(t_{i+1}, w_{i+1}, v_{i+1}). \quad (7.109)$$

De nuevo, la formulación es sencilla pero ahora depende del $f(t, y, y')$ para poder resolver el sistema. En efecto, si $f(t, y, y')$ no es lineal, tendremos la misma dificultad que habíamos visto anteriormente; debemos obtener v_{i+1} de manera iterativa. Esto no siempre es fácil de hacer. Sin embargo, el sistema podemos reescribirlo así:

$$\begin{bmatrix} 1 & -h \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} w_{i+1} \\ v_{i+1} \end{bmatrix} = \begin{bmatrix} w_i \\ v_i + h \cdot f(t_{i+1}, w_{i+1}, v_{i+1}) \end{bmatrix} \quad (7.110)$$

Supongamos por un momento que $f(t_{i+1}, w_{i+1}, v_{i+1}) = t_{i+1}^2 + 2 \cdot w_{i+1} - v_{i+1}$, entonces nuestro sistema quedará de la siguiente manera:

$$\begin{bmatrix} 1 & -h \\ -2 \cdot h & 1 + h \end{bmatrix} \cdot \begin{bmatrix} w_{i+1} \\ v_{i+1} \end{bmatrix} = \begin{bmatrix} w_i \\ v_i + h \cdot t_{i+1}^2 \end{bmatrix} \quad (7.111)$$

Nuestra solución analítica será:

$$\begin{bmatrix} w_{i+1} \\ v_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & -h \\ -2 \cdot h & 1 + h \end{bmatrix}^{-1} \cdot \begin{bmatrix} w_i \\ v_i + h \cdot t_{i+1}^2 \end{bmatrix} \quad (7.112)$$

pero podemos aplicar cualquier método de resolución de ecuaciones lineales visto. En este caso particular, podemos ver que para cualquier $0 < h < 1$, la matriz es definida positiva y estrictamente diagonal dominante, así que la solución es única para cada i .

7.4.2. A partir de la serie de Taylor

Una segunda forma de encarar el problema es una vez más partir de un desarrollo en serie de Taylor para y_{i+1} :

$$y_{i+1} = y_i + y'_i h + y''_i \frac{h^2}{2!} + y'''_i \frac{h^3}{3!} + y^{iv}_i \frac{h^4}{4!} + \dots \quad (7.113)$$

Como $y''(t) = f(t, y, y')$, el desarrollo anterior podemos escribirlo así:

$$y_{i+1} = y_i + y'_i h + f(t_i, y_i, y'_i) \frac{h^2}{2!} + f'(t_i, y_i, y'_i) \frac{h^3}{3!} + f''(t_i, y_i, y'_i) \frac{h^4}{4!} + \dots \quad (7.114)$$

Al mismo tiempo, si hacemos lo mismo para $y'(t)$, tenemos,

$$y'_{i+1} = y'_i + y''_i h + y'''_i \frac{h^2}{2!} + \dots, \quad (7.115)$$

que también podemos escribir como

$$y'_{i+1} = y'_i + f(t_i, y_i, y'_i) h + f'(t_i, y_i, y'_i) \frac{h^2}{2!} + \dots \quad (7.116)$$

Ambas series representan a la función y su primera derivada. Por lo tanto, si queremos obtener una aproximación de ambas, truncamos las series de Taylor correspondientes y así obtenemos dos ecuaciones:

$$y_{i+1} = y_i + y'_i h + f(t_i, y_i, y'_i) \frac{h^2}{2!} + f'[\xi, y(\xi), y'(\xi)] \frac{h^3}{3!}, \quad (7.117)$$

$$y'_{i+1} = y'_i + f(t_i, y_i, y'_i) h + f'[\xi, y(\xi), y'(\xi)] \frac{h^2}{2!}, \quad (7.118)$$

y entonces, nos queda que:

$$y_{i+1} \approx y_i + y'_i h + f(t_i, y_i, y'_i) \frac{h^2}{2!}, \quad (7.119)$$

$$y'_{i+1} \approx y'_i + f(t_i, y_i, y'_i) h. \quad (7.120)$$

Si definimos que $y_i = w_i$ y $y'_i = v_i$, nos queda un sistema de ecuaciones:

$$w_{i+1} = w_i + v_i h + f(t_i, w_i, v_i) \frac{h^2}{2!}, \quad (7.121)$$

$$v_{i+1} = v_i + f(t_i, w_i, v_i) h. \quad (7.122)$$

Esta aproximación sencilla es de convergencia lineal, similar a los *métodos de Euler* para ecuaciones de primer orden. A partir de este tipo de formulación obtenemos dos métodos más:

- El **Método de Taylor**, que está dado por las ecuaciones

$$\begin{aligned} w_{i+1} &= w_i + v_i h + f(t_i, w_i, v_i) \frac{h^2}{2!}, \\ v_{i+1} &= v_i + h [\beta f(t_i, w_i, v_i) + (1 - \beta) f(t_{i+1}, w_{i+1}, v_{i+1})], \quad y; \end{aligned} \quad (7.123)$$

- El **Método de Newmark**, que está dado por estas dos ecuaciones

$$\begin{aligned} w_{i+1} &= w_i + v_i h + \frac{h^2}{2!} [\alpha f(t_i, w_i, v_i) + (1 - \alpha) f(t_{i+1}, w_{i+1}, v_{i+1})], \\ v_{i+1} &= v_i + h [\beta f(t_i, w_i, v_i) + (1 - \beta) f(t_{i+1}, w_{i+1}, v_{i+1})]. \end{aligned} \quad (7.124)$$

Ambos métodos tienen la particularidad de ser implícitos, el de *Taylor* sólo para la ecuación de la derivada primera, en tanto que el de *Newmark* lo es para las dos ecuaciones.

7.4.3. Aproximación de la derivada segunda

Existe una manera más de encarar la resolución de ecuaciones diferenciales de orden superior con valores iniciales. Si aproximamos la segunda derivada, como se vio en diferenciación numérica, tenemos que

$$\frac{d^2 y}{dt^2} \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}, \quad (7.125)$$

y obtenemos una nueva aproximación de y_{i+1} :

$$y_{i+1} \approx 2y_i - y_{i-1} + h^2 f(t_i, y_i, y'_i), \quad (7.126)$$

es decir,

$$w_{i+1} = 2w_i - w_{i-1} + h^2 f(t_i, w_i, v_i), \quad (7.127)$$

si hacemos nuevamente $y'_i = v_i$.

Pero ahora necesitamos calcular w_1 para poder empezar a iterar y no contamos con una forma de aproximar v_i (o sea y'_i). Esto último se puede resolver aproximando la primera derivada por el método de las diferencias centradas:

$$y'_i \approx \frac{y_{i+1} - y_{i-1}}{2h} \rightarrow v_i = \frac{w_{i+1} - w_{i-1}}{2h}, \quad (7.128)$$

con lo que evitamos tener que agregar un algoritmo para calcular dicha derivada.

Aún nos falta obtener w_i . Podemos aplicar uno de los métodos ya vistos, como por ejemplo:

$$\begin{aligned} y_1 &\approx y_0 + y'_0 h + f(t_0, y_0, y'_0) \frac{h^2}{2!} \rightarrow \\ w_1 &= w_0 + v_0 h + f(t_0, w_0, v_0) \frac{h^2}{2!}. \end{aligned} \quad (7.129)$$

Si agrupamos todo y reemplazamos en la ecuación original, tenemos:

$$\begin{aligned} w_1 &= w_0 + v_0 h + f(t_0, w_0, v_0) \frac{h^2}{2!}; \\ w_{i+1} &= 2w_i - w_{i-1} + h^2 f\left(t_i, w_i, \frac{w_{i+1} - w_{i-1}}{2h}\right); \end{aligned} \quad (7.130)$$

que se conoce como *Método de Nyström* para ecuaciones diferenciales de segundo orden que, como podemos ver, es un método implícito. Así planteado, el método tiene un orden de convergencia cuadrático, pues:

$$\frac{d^2 y}{dt^2} = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} - y^{iv}(\xi) \frac{h^2}{12}, \quad y \quad (7.131)$$

$$\frac{dy}{dt} = \frac{y_{i+1} - y_{i-1}}{2h} - y'''(\xi) \frac{h^2}{6}. \quad (7.132)$$

En algunos problemas particulares, la forma implícita no resulta conveniente, por lo que suele reemplazarse la aproximación centrada de la derivada primera por una aproximación regresiva pero perdiendo orden de convergencia.

7.5. Sistemas de ecuaciones diferenciales de primer orden

Lo visto anteriormente para ecuaciones diferenciales de orden superior podemos aplicarlo para resolver un sistema de ecuaciones diferenciales de primer orden del tipo:

$$\begin{aligned} \frac{d u_1}{d t} &= f_1(t, u_1, u_2, \dots, u_n) \\ \frac{d u_2}{d t} &= f_2(t, u_1, u_2, \dots, u_n) \\ &\vdots \\ \frac{d u_n}{d t} &= f_n(t, u_1, u_2, \dots, u_n) \end{aligned} \quad (7.133)$$

con las condiciones siguientes: $a < t < b$; $u_1(a) = \alpha_1$, $u_2(a) = \alpha_2$, \dots , $u_n(a) = \alpha_n$.

Ahora, en lugar de transformar la ecuación diferencial, lo que tenemos son varias funciones $u_i(t)$ entrelazadas. La solución de cada función $u_i(t)$ depende de las demás $u_j(t)$, que es análogo al caso anterior, donde $y(t)$ dependía de $z(t)$.

En consecuencia, podremos aplicar cualquiera de los métodos vistos para resolver ecuaciones diferenciales ordinarias de primer orden, cuidando de armar la formulación algebraica para cada $u_i(t)$. Debemos tener en cuenta que obtendremos un conjunto de valores $u_{i,j}$ con $j = 0; 1; \dots; m$ y $h = \frac{b-a}{m}$. (Para más detalles acerca de sistemas de ecuaciones diferenciales de primer orden, ver [3].)

7.6. Ecuaciones diferenciales ordinarias con condiciones de contorno

7.6.1. Introducción

En puntos anteriores hemos visto los diferentes métodos numéricos para la resolución de ecuaciones diferenciales ordinarias con valores iniciales. Estos métodos son principalmente para resolver ecuaciones diferenciales de primer orden, tanto lineales como no lineales, y que podemos adaptarlos para ecuaciones diferenciales de orden superior con valores iniciales.

Pero este caso no suele ser el más común o usual. Las ecuaciones diferenciales de orden dos o superior generalmente son de valores de contorno o frontera, es decir, no disponemos de todos los valores para $t = a$, sino que tenemos valores para $t = a$ y $t = b$. Una ecuación diferencial de orden 2 con valores de contorno está dada por:

$$\frac{d^2 y}{dt^2} = f(t, y, y'), \quad \text{en } [a, b], \quad (7.134)$$

similar a lo visto, pero con las condiciones en los extremos del intervalo:

$$y(a) = \alpha, \quad y(b) = \beta.$$

Como podemos ver, con estas condiciones no parece posible utilizar los métodos estudiados, ni siquiera transformando la ecuación diferencial de segundo orden en un sistema de ecuaciones diferenciales primer orden. Debemos buscar alguna forma que nos permita aproximar nuestra ecuación diferencial y obtener los resultados de la función $y(t)$.

¿Es necesario analizarlo con más detalle el procedimiento para resolverlas? Como dato importante, basta mencionar que buena parte de los problemas que debemos resolver los ingenieros, ya sean civiles, mecánicos, electrónicos, etc., están expresados en términos de ecuaciones diferenciales de orden superior. Un ejemplo que suele ser muy usado es el caso de la ecuación diferencial de equilibrio para una viga, dada por la expresión:

$$EI \frac{d^4 w}{dx^4} - p(x) = 0.$$

que requiere de cuatro condiciones de contorno para ser resuelta. Estas condiciones pueden ser:

1. Condiciones de borde esenciales (Dirichlet);
2. Condiciones de borde naturales (Neumann);
3. Una combinación de ambas.

Por ejemplo, para una viga doblemente empotrada, de longitud L , como se ve en la figura, las condiciones de borde son:

$$w(0) = 0, \quad w(L) = 0, \quad w'(0) = 0, \quad \text{y} \quad w'(L) = 0.$$

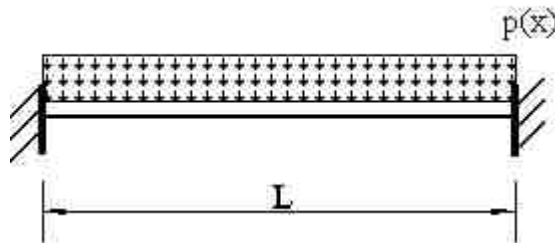


Figura 7.1: Viga doblemente empotrada

Este es el típico caso de condiciones de borde esenciales (o forzadas), puesto que las restricciones están asociadas a los desplazamientos y los giros en los extremos de la viga. Esta ecuación no es posible resolverla aplicando en forma directa los métodos mencionados anteriormente. En consecuencia, para poder aproximar una solución, debemos buscar alguna forma de adaptar los métodos vistos para tener en cuenta estas condiciones de frontera o de contorno.

Veremos a continuación dos métodos que pueden usarse para resolver este tipo de ecuaciones diferenciales. Empezaremos por el más sencillo, el método del disparo lineal, que hace uso de los métodos ya estudiados.

7.6.2. Método del tiro o disparo lineal

Supongamos que tenemos la siguiente ecuación diferencial:

$$y'' = -f(t, y), \quad t \in [0; 1], \quad (7.135)$$

que debe cumplir con las condiciones:

$$y(0) = y_0, \quad y(1) = y_1.$$

Como vemos, no tenemos dos condiciones iniciales, sino una para el valor inicial y otra para el valor final que debe tomar la función buscada.

Para encarar el problema haremos una modificación. Resolveremos el siguiente problema de valores iniciales, suponiendo que lo que buscamos es una aproximación a $y(t)$ que llamaremos $u(t_i)$. Entonces nuestro sistema quedará de la siguiente forma:

$$u'' = -f(t, u), \quad u_1(0) = y_0 \text{ y } u_1'(0) = \alpha_1, \quad (7.136)$$

donde α_1 es el primer ensayo para $u_1'(0)$. Apliquemos para ello cualquiera de los métodos vistos anteriormente, por ejemplo el de Euler. Con él obtendremos un valor para $u_1(1)$ igual a β_1 , que seguramente será distinto a y_1 .

Nuevamente, resolvamos con Euler un sistema similar pero proponiendo que $u_2(0) = y_0$ y $u_2'(0) = \alpha_2$. Obtendremos otro valor para $u(1)$, es decir, un $u_2(1) = \beta_2$, probablemente distinto a y_1 .

En consecuencia, tendremos dos aproximaciones de y_1 . Para continuar, vamos a suponer que existe una relación lineal entre $u(t_i)$, $u_1(t_i)$ y $u_2(t_i)$. Esta relación lineal estará dada por:

$$\frac{u(t_i) - u_1(t_i)}{y_1 - \beta_1} = \frac{u_2(t_i) - y_0}{\beta_2 - y_0}. \quad (7.137)$$

Para calcular $u(t)$ debemos despejarla de la expresión anterior. Así obtenemos:

$$u(t_i) = u_1(t_i) + \frac{y_1 - \beta_1}{\beta_2 - y_0} [u_2(t_i) - y_0]. \quad (7.138)$$

Para entender como opera el método, veamos un ejemplo práctico, resolviendo una ecuación diferencial de orden 2.

Ejemplo

Resolver la siguiente ecuación diferencial ordinaria con valores de frontera, aplicando el *método de Euler explícito*:

$$y'' = 4(y - x), \quad 0 \leq x \leq 1;$$

con los valores de contorno:

$$y(0) = 0, \quad y(1) = 2.$$

Para resolver la ecuación por el *método de Euler explícito* plantearemos primero que $y'(x) = z(x)$, con lo que tendremos que la ecuación diferencial se transforma en:

$$\begin{aligned} y'(x) &= z(x) \\ z'(x) &= 4(y - x) \end{aligned}$$

Si aplicamos el *método de Euler explícito*, y hacemos $u_i = y(x_i)$ tendremos las siguientes ecuaciones:

$$\begin{aligned} u_{i+1} &= u_i + h \cdot z_i \\ z_{i+1} &= z_i + h \cdot 4(u_i - x_i) \end{aligned}$$

Como vemos, debemos resolver dos ecuaciones para obtener el valor de u_{i+1} . Por ello, en primer término, vamos a resolver el sistema obteniendo, primero, valores para unas funciones $v_1(x)$ y adoptando las siguientes condiciones iniciales:

$$v_1(0) = 0, \quad z_1(0) = 0$$

Tabla 7.1: Resultados obtenidos aplicando el Método de Euler Explícito

x_i	$z_{1,i}$	$v_{1,i}$	$z_{2,i}$	$v_{2,i}$	u_i	$y(x_i)$	e
0,00	0,000	0,000	1,000	0,000	0,000	0,000	0,0
0,10	0,000	0,000	1,000	0,100	0,252	0,156	$9,7 \cdot 10^{-2}$
0,20	-0,040	0,000	1,000	0,200	0,504	0,313	$1,9 \cdot 10^{-1}$
0,30	-0,120	-0,004	1,000	0,300	0,752	0,476	$2,8 \cdot 10^{-1}$
0,40	-0,242	-0,016	1,000	0,400	0,992	0,645	$3,5 \cdot 10^{-1}$
0,50	-0,408	-0,040	1,000	0,500	1,220	0,824	$4,0 \cdot 10^{-1}$
0,60	-0,624	-0,081	1,000	0,600	1,432	1,016	$4,2 \cdot 10^{-1}$
0,70	-0,896	-0,143	1,000	0,700	1,621	1,225	$4,0 \cdot 10^{-1}$
0,80	-1,234	-0,233	1,000	0,800	1,784	1,455	$3,3 \cdot 10^{-1}$
0,90	-1,647	-0,356	1,000	0,900	1,913	1,711	$2,0 \cdot 10^{-1}$
1,00	-2,150	-0,521	1,000	1,000	2,000	2,000	0,0

por lo que el sistema a resolver será:

$$\begin{aligned}v_{1_{i+1}} &= v_{1_i} + h \cdot z_{1_i} \\z_{1_{i+1}} &= z_{1_i} + h \cdot 4(v_{1_i} - x_i)\end{aligned}$$

En segundo término, haremos lo mismo pero para las funciones $v_2(x)$ y $z_2(x) = v_2'(x)$ con los valores de contorno levemente distintos. Estos son:

$$v_2(0) = 0, \quad z_2(0) = 1,$$

y el sistema a resolver será:

$$\begin{aligned}v_{2_{i+1}} &= v_{2_i} + h \cdot z_{2_i} \\z_{2_{i+1}} &= z_{2_i} + h \cdot 4(v_{2_i} - x_i)\end{aligned}$$

Con los valores para cada una de las soluciones y por cada iteración, calcularemos los valores definitivos mediante la expresión:

$$u_i = v_{1_i} + \frac{y(1) - v_1(1)}{v_2(1) - y(0)} [v_{2_i} - y(0)]$$

En la tabla 7.1 podemos ver los resultados obtenidos.

En la penúltima columna podemos ver el valor *exacto* de la función buscada, dado que la solución analítica de la ecuación diferencial es:

$$y(x) = e^2 (e^4 - 1)^{-1} (e^{2x} - e^{-2x}) + x.$$

Los valores de $u(x_i)$ obtenidos no son muy precisos, dado que el método utilizado para resolver el sistema de ecuaciones es el de *Euler explícito*, pero igualmente sirven como demostración de la efectividad al aplicar este método. Podemos ver que la última columna muestra el error absoluto entre el valor obtenido numéricamente y el valor exacto. Observemos que el error cometido es del orden de 10^{-1} , un error razonable para este método. (Recordemos que el método de Euler explícito tiene un orden de convergencia $O(h)$.)

7.6.3. Diferencias finitas

En el punto anterior hemos resuelto una ecuación diferencial lineal con condiciones de contorno utilizando un método de resolución que transforma las condiciones de contorno en condiciones iniciales. Sin embargo, este método tiene como desventaja que es inestable en ciertas ocasiones. Por lo que su utilización se ve reducida generalmente a unos pocos casos o problemas.

Uno de los métodos más aplicados para aproximar una solución de ecuaciones diferenciales de orden mayor o igual a dos, es el que reemplaza las derivadas por diferencias finitas mediante un cociente de diferencias, tal como vimos en diferenciación numérica. La aplicación de estas *diferencias finitas* generan un sistema de ecuaciones lineales del tipo $Ax = B$, sistema que puede resolverse mediante alguno de los métodos ya vistos. Está claro que estamos limitados en la elección de nuestro intervalo h , que no puede ser muy chico. Veamos en qué consiste el método, aplicándolo a nuestro ejemplo anterior.

Para aproximar las derivadas, tomaremos el método de las *diferencias centradas*, que permiten una mejor aproximación de las derivadas. Para empezar, desarrollemos $y(x_{i+1})$ y $y(x_{i-1})$ por Taylor hasta el cuarto término, por lo que tendremos:

$$y(x_{i+1}) = y(x_i + h) = y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + \frac{h^3}{6}y'''(x_i) + \frac{h^4}{24}y^{(iv)}(\xi_i^+), \quad (7.139)$$

para alguna ξ_i^+ en (x_i, x_{i+1}) , y

$$y(x_{i-1}) = y(x_i - h) = y(x_i) - hy'(x_i) + \frac{h^2}{2}y''(x_i) - \frac{h^3}{6}y'''(x_i) + \frac{h^4}{24}y^{(iv)}(\xi_i^-), \quad (7.140)$$

para alguna ξ_i^- en (x_{i-1}, x_i) . Demás está decir que se supone que $y(x) \in C^4[x_{i-1}, x_{i+1}]$. Si sumamos ambas expresiones y despejamos $y''(x_i)$, tendremos:

$$y''(x_i) = \frac{1}{h^2} [y(x_{i+1}) - 2y(x_i) + y(x_{i-1})] - \frac{h^2}{24} [y^{(iv)}(\xi_i^+) + y^{(iv)}(\xi_i^-)]. \quad (7.141)$$

Si aplicamos el teorema del valor medio, podemos simplificar la expresión a:

$$y''(x_i) = \frac{1}{h^2} [y(x_{i+1}) - 2y(x_i) + y(x_{i-1})] - \frac{h^2}{12}y^{(iv)}(\xi_i), \quad (7.142)$$

para alguna ξ_i en (x_{i-1}, x_{i+1}) .

Reemplacemos esta última expresión en nuestra ecuación diferencial:

$$\frac{1}{h^2} [y(x_{i+1}) - 2y(x_i) + y(x_{i-1})] - \underbrace{\frac{h^2}{12}y^{(iv)}(\xi_i)}_{O(h^2)} = 4[y(x_i) - x_i]. \quad (7.143)$$

De esta manera, nuestra ecuación diferencial se transforma en:

$$[y(x_{i+1}) - 2y(x_i) + y(x_{i-1})] = 4h^2 [y(x_i) - x_i], \quad (7.144)$$

y desarrollando algebraicamente, obtenemos:

$$[y(x_{i-1}) - 2(1 + 2h^2)y(x_i) + y(x_{i+1})] = -4h^2x_i, \quad (7.145)$$

por lo tanto, para cada i tenemos una ecuación lineal. Definamos, entonces, el intervalo o *paso* h como $\frac{b-a}{N}$ siendo $N > 0$; de esta manera obtendremos N intervalos para $i \in [0; N]$. Con i y h podemos armar nuestro sistema de ecuaciones para $i \in [1; N - 1]$. La matriz resultante será:

$$A = \begin{bmatrix} 1 & -2(1 + 2h^2) & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & -2(1 + 2h^2) & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -2(1 + 2h^2) & 1 & 0 \\ 0 & \dots & 0 & 0 & 1 & -2(1 + 2h^2) & 1 \end{bmatrix}.$$

Si hacemos que $y_i = y(x_i)$ tendremos que nuestras incógnitas son:

$$y = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix}.$$

Nuestro vector de términos independientes será:

$$B = \begin{bmatrix} -4h^2x_1 \\ -4h^2x_2 \\ \vdots \\ -4h^2x_{N-2} \\ -4h^2x_{N-1} \end{bmatrix}.$$

Pero hemos armado un sistema con $N - 1$ filas y $N + 1$ incógnitas y_i . Para completar el sistema debemos recordar que $y_0 = \alpha$ y $y_N = \beta$, por lo que nuestro sistema de ecuaciones lineales quedará como:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 1 & -2(1+2h^2) & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & -2(1+2h^2) & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -2(1+2h^2) & 1 & 0 \\ 0 & \dots & 0 & 0 & 1 & -2(1+2h^2) & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{N-2} \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} \alpha \\ -4h^2x_1 \\ -4h^2x_2 \\ \vdots \\ -4h^2x_{N-2} \\ -4h^2x_{N-1} \\ \beta \end{bmatrix}.$$

Armemos el sistema definitivo con $x \in [0; 1]$, $y(0) = y_0 = 0$, $y(1) = y_N = 2$ y $N = 10$. Con estos parámetros tendremos que $h = \frac{1-0}{10} = 0,1$. Entonces, en la matriz A tendremos el coeficiente (además de 1):

$$-2 [1 + 2(0,1)^2] = -2 (1 + 0,02) = -2,04;$$

y en el vector de términos independientes:

$$-4(0 + 0,1)^2 \cdot 0,1 \cdot i = -4(0,1)^2 \cdot 0,1 \cdot i = -i \cdot 4(0,1)^3;$$

con $i \in [1, N - 1]$. El sistema definitivo quedará con la matriz de coeficientes:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2,04 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2,04 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2,04 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2,04 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2,04 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2,04 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2,04 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2,04 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2,04 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

y con el vector de términos independientes:

$$B = \begin{bmatrix} 0 \\ -0,004 \\ -0,008 \\ -0,012 \\ -0,016 \\ -0,020 \\ -0,024 \\ -0,028 \\ -0,032 \\ -0,036 \\ 2 \end{bmatrix} .$$

Al resolver el sistema de ecuaciones por alguno de los métodos numéricos que hemos estudiado en el capítulo 3, obtenemos el siguiente vector solución:

$$y = \begin{bmatrix} 0,000 \\ 0,156 \\ 0,313 \\ 0,476 \\ 0,645 \\ 0,824 \\ 1,017 \\ 1,225 \\ 1,455 \\ 1,711 \\ 2,000 \end{bmatrix} .$$

Si lo comparamos con el vector y obtenido con el *Método de Euler Explícito* (tabla 7.1), podemos observar que la solución por diferencias finitas es mucho más precisa, ya que los y obtenidos son *iguales* a los hallados por aplicación de la solución analítica.

7.7. Notas finales

Casi podría decirse que todos los problemas que debe enfrentar un ingeniero pueden formularse mediante ecuaciones diferenciales. Desde el análisis estructural hasta el diseño de un avión de pasajeros, las ecuaciones diferenciales intervienen en forma explícitas (deben ser resueltas) o en forma implícita (se aplican soluciones analíticas de dichas ecuaciones).

Hasta la mitad del siglo XX, muchas de las limitaciones en los aspectos ingenieriles estaban dados por las pocas soluciones analíticas que se podían obtener de muchas de las ecuaciones diferenciales, y en consecuencia, se dependía de los ensayos en modelos físicos o en prototipos. Con el desarrollo de las computadoras, a partir de los años '50, y principalmente, con la aparición de las computadoras personales hace 25 años, obtener soluciones aproximadas de las ecuaciones diferenciales dejó de ser un escollo en cuanto a tiempo de cálculo. Prácticamente todas las disciplinas científicas y tecnológicas basan sus soluciones en la aplicación de métodos numéricos.

Dentro del conjunto de métodos numéricos para resolver ecuaciones diferenciales, los métodos de las diferencias finitas y de los elementos finitos, y en particular este último, son los más usados para encarar soluciones aproximadas. Y en los últimos años, la gran capacidad de cálculo de las computadoras han permitido adentrarse en la resolución aproximada de problemas con ecuaciones diferenciales no lineales, permitiendo el estudio de muchos fenómenos que antes se consideraban como «imposibles» de abordar. Basta con ver el avance en el campo de los

estudios climáticos, el comportamiento de los ríos, el avance en la hidráulica marítima, etc., que han reemplazado el uso de modelos físicos (muy caros y lentos) por modelos matemáticos (más baratos y rápidos).

Ejercicios

Con valores iniciales

Métodos de paso simple

1. Aplique el *Método de Euler Explícito* para aproximar la solución de las siguientes ecuaciones diferenciales con valor inicial:

a) $y' = t e^{3t} - 2y$, con $0 \leq t \leq 1$, $y(0) = 0$ y $h = 0,5$.

b) $y' = 1 + \frac{y}{t}$, con $1 \leq t \leq 2$, $y(1) = 2$ y $h = 0,25$.

c) $y' = 1 + \frac{y}{t} + \left(\frac{y}{t}\right)^2$, con $1 \leq t \leq 3$, $y(1) = 0$ y $h = 0,2$.

d) $y' = -(y+1)(y+3)$, con $0 \leq t \leq 2$, $y(0) = -2$ y $h = 0,2$.

e) $y' = -5y + 5t^2 + 2t$, con $0 \leq t \leq 1$, $y(0) = 0,3333$ y $h = 0,1$.

2. Aproxime la solución de las ecuaciones del punto anterior aplicando el *Método de Euler Implícito*.
3. Con las soluciones analíticas de las ecuaciones del punto 1., compare los resultados obtenidos mediante la aplicación de los *Métodos de Euler* con los «exactos». Calcule el error absoluto y el error relativo de cada uno.

a) $y(t) = \frac{1}{5} t e^{3t} - \frac{1}{25} (e^{3t} - e^{-2t})$.

b) $y(t) = t \ln t + 2t$.

c) $y(t) = t \operatorname{tg}(\ln t)$.

d) $y(t) = -3 + \frac{2}{1 + e^{-2t}}$.

e) $y(t) = t^2 + \frac{e^{-5t}}{3}$.

4. Dada la siguiente ecuación diferencial con valor inicial

$$y' = \frac{1}{t^2} - \frac{y}{t} - y^2 \quad \text{con } 1 \leq t \leq 2 \text{ y } y(1) = -1,$$

cuya solución analítica es

$$y(t) = -\frac{1}{t},$$

- a) Aplicar el *Método de Euler Explícito* con $h = 0,05$ para aproximar la solución.
- b) Ídem del punto anterior pero con el *Método de Euler Implícito*.

Compare los resultados.

5. Aplique el *Método de Taylor de orden 2* para aproximar la solución de las siguientes ecuaciones diferenciales con valor inicial:

a) $y' = t e^{3t} - 2y$, con $0 \leq t \leq 1$, $y(0) = 0$ y $h = 0,5$.

b) $y' = 1 + \frac{y}{t}$, con $1 \leq t \leq 2$, $y(1) = 2$ y $h = 0,25$.

c) $y' = 1 + \frac{y}{t} + \left(\frac{y}{t}\right)^2$, con $1 \leq t \leq 3$, $y(1) = 0$ y $h = 0,2$.

d) $y' = -(y+1)(y+3)$, con $0 \leq t \leq 2$, $y(0) = -2$ y $h = 0,2$.

e) $y' = -5y + 5t^2 + 2t$, con $0 \leq t \leq 1$, $y(0) = 0,3333$ y $h = 0,1$.

Compare con los resultados obtenidos en los puntos 1 y 2.

6. Ídem el punto anterior pero con la ecuación diferencial del punto 4.
7. Aproxime las ecuaciones diferenciales del punto 1 aplicando el *Método del Punto Medio* (*Método de Runge-Kutta de orden 2*).
8. Ídem el punto anterior pero aplicando el *Método de Euler Modificado* (*Método de Runge-Kutta de orden 2*).
9. Ídem el punto anterior pero aplicando el *Método de Heun* (*Método de Runge-Kutta de orden 2*).
10. Ídem el punto anterior pero aplicando el *Método de Crank-Nicolson* (*Método de Runge-Kutta de orden 2*).
11. Ídem el punto anterior pero aplicando el *Método de Runge-Kutta de orden 3*.
12. Ídem el punto anterior pero aplicando el *Método de Runge-Kutta de orden 4*.
13. Arme un cuadro comparativo con todas las soluciones obtenidas para las ecuaciones diferenciales del punto 1.

Métodos multipasos

1. Aplique el *Método de Adams-Bashforth* de orden 2 para aproximar la solución de las siguientes ecuaciones diferenciales con valor inicial:

a) $y' = t e^{3t} - 2y$, con $0 \leq t \leq 1$, $y(0) = 0$ y $h = 0,5$.

b) $y' = 1 + \frac{y}{t}$, con $1 \leq t \leq 2$, $y(1) = 2$ y $h = 0,25$.

c) $y' = 1 + \frac{y}{t} + \left(\frac{y}{t}\right)^2$, con $1 \leq t \leq 3$, $y(1) = 0$ y $h = 0,2$.

d) $y' = -(y+1)(y+3)$, con $0 \leq t \leq 2$, $y(0) = -2$ y $h = 0,2$.

e) $y' = -5y + 5t^2 + 2t$, con $0 \leq t \leq 1$, $y(0) = 0,3333$ y $h = 0,1$.

Utilice soluciones obtenidas por algún *Método de Runge-Kutta* de orden para el valor de y_1 .

2. Ídem el punto anterior pero aplicando el *Método de Adams-Moulton* de orden 2.
3. Aproxime las ecuaciones del punto 1 pero aplicando el *Método de Adams-Bashforth* de orden 4. Utilice las soluciones aproximadas con el *Método de Runge-Kutta* de orden 4 para los valores de y_1 , y_2 y y_3 .
4. Aproxime las ecuaciones del punto 1 pero aplicando el *Método de Adams-Moulton* de orden 4. Utilice las soluciones aproximadas con el *Método de Runge-Kutta* de orden 4 para los valores de y_1 y y_2 .

Casos prácticos

1. Al mezclar dos fluidos, a veces, se originan ecuaciones diferenciales lineales de primer orden. Cuando se describe la mezcla de dos sales, se supone que la tasa con que cambia la cantidad de sal, $A'(t)$, en un tanque de mezcla, tiene una rapidez neta igual a:

$$\frac{dA}{dt} = R_I - R_O,$$

donde R_I es la rapidez con que entra una sal, y R_O es la rapidez con que sale una sal. Suponiendo que:

$$R_I = 3 \text{ kg/min}, \quad R_O = \frac{A}{1000} \text{ kg/min}, \quad A(0) = 25 \text{ kg} \quad \text{y} \quad h = 10 \text{ min},$$

obtener una solución aproximada en el intervalo $0 \leq t \leq 100$ min aplicando:

- El Método de Euler Explícito,
- El Método de Euler Implícito,
- El Método de Crank-Nicolson,
- El Método de Runge-Kutta de orden 4.

Compare los resultados obtenidos con la solución analítica:

$$A(t) = 300 - 275 e^{-\frac{t}{100}}.$$

2. Un modelo sencillo de «tsunami» está descrito por el siguiente problema de valor inicial:

$$\frac{dH}{dx} = H \sqrt{4 - 2H}, \quad \text{con} \quad H(0) = 2,$$

donde H es la altura de la ola y x es la posición del «tsunami» respecto del punto de origen. Obtenga una solución aproximada en el intervalo $0 \leq x \leq 10$ km, aplicando:

- El Método de Euler Implícito,
- El Método de Crank-Nicolson,
- El Método de Adams-Moulton de orden 4.

3. La cantidad $N(t)$ de hipermercados que usan cajas computarizadas en un país, está definida por el problema de valor inicial:

$$\frac{dN}{dt} = N(1 - 0,0005 N), \quad \text{con} \quad N(0) = 1.$$

Obtenga una solución aproximada en el intervalo $0 \leq t \leq 10$ años, aplicando:

- El Método de Euler explícito,
- El Método de Euler implícito,
- El Método de Crank-Nicolson,
- El Método de Runge-Kutta de orden 4.

4. El modelo demográfico $P(t)$ de un suburbio en una gran ciudad está descrito por el problema de valor inicial:

$$\frac{dP}{dt} = P(10^{-1} - 10^{-7} P), \quad \text{con} \quad P(0) = 5000,$$

donde t se expresa en meses. Obtenga una solución aproximada en el intervalo $0 \leq t \leq 10$ meses, aplicando:

- a) El *Método de Euler Explícito*,
 b) El *Método de Euler Mejorado*,
 c) El *Método de Runge-Kutta* de orden 4,
 d) El *Método de Adams-Bashforth* de orden 4.
5. Aplique el *Método de Euler Modificado* para aproximar las soluciones de las siguientes ecuaciones diferenciales de segundo orden con valores iniciales:
- a) $y'' = t(e^t - 1) - y + 2y'$ con $0 \leq t \leq 1$, $y(0) = 0$ y $y'(0) = 0$, tomando $h = 0,1$;
 b) $y'' = t \ln t - \frac{2}{t} \left(\frac{y}{t} - y' \right)$ con $1 \leq t \leq 2$, $y(1) = 1$ y $y'(1) = 0$, tomando $h = 0,1$.
6. Aproxime las mismas ecuaciones del punto anterior pero aplicando el *Método de Runge-Kutta* de orden 4.
7. Un sistema masa-resorte con movimiento forzado amortiguado como representado en la figura 7.2, está descrito por la siguiente ecuación diferencial de segundo orden:

$$0,2 \frac{d^2x}{dt^2} + 1,2 \frac{dx}{dt} + 2x = 5 \cos 4t,$$

en el intervalo $0 \leq t \leq 1,6$ con $x(0) = 0,5$ y $\dot{x}(0) = 0$.

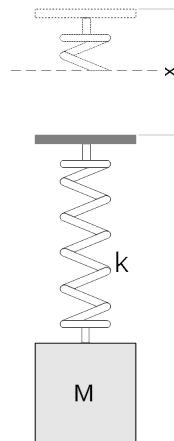


Figura 7.2: Sistema masa-resorte.

Aplique el *Método de Runge-Kutta* de orden 4 para obtener una aproximación de la solución.

Con condiciones de contorno

1. Se tienen dos esferas concéntricas de radio $r_0 = 25$ cm y $r_1 = 40$ cm, como se ve en la figura 7.3. La temperatura $T(r)$ en la región entre ambas esferas está determinada por la siguiente ecuación diferencial de segundo orden:

$$r \frac{d^2T}{dr^2} + 2 \frac{dT}{dr} = 0.$$

Las condiciones de contorno son: $T_0 = T(25 \text{ cm}) = 300$ K y $T_1 = T(40 \text{ cm}) = 280$ K. Si $h = \frac{r_1 - r_0}{10}$, aplique el *Método de las Diferencias Finitas* para aproximar una solución del problema.

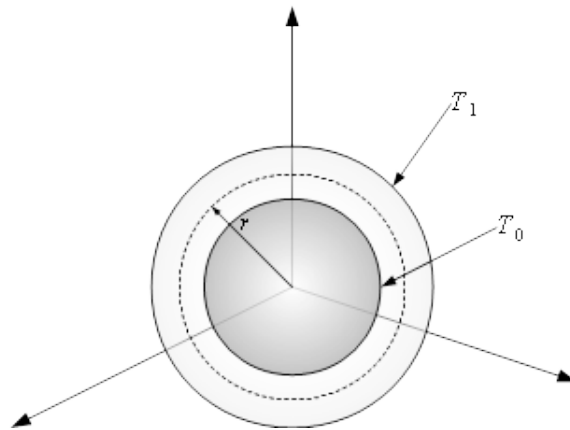


Figura 7.3: Esferas concéntricas.

2. La ecuación diferencial

$$\frac{d^2 M}{dx^2} = \frac{N \cdot M}{E \cdot I} - p \left[1 - 2 \frac{x}{L} \left(1 - \frac{x}{L} \right) \right],$$

corresponde a la ecuación de equilibrio del sistema estructural de la figura 7.4.

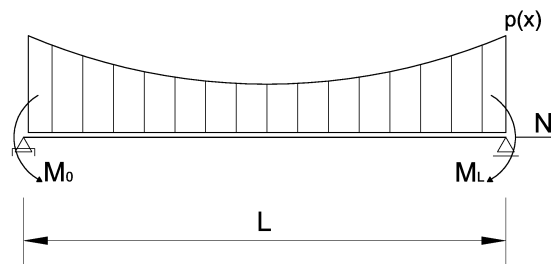


Figura 7.4: Viga simplemente apoyada.

Para $x \in [0, 10 \text{ m}]$, las siguientes condiciones de contorno, $M_0 = -0,10 \text{ MNm}$, $M_L = -0,20 \text{ MNm}$, el esfuerzo normal $N = 1,0 \text{ MN}$ y los siguientes datos del material, $E = 206 \text{ GPa}$, la sección transversal, $I = 0,001388 \text{ m}^4$ y la carga externa, $p = 0,1 \text{ MN/m}$, aproxime la solución del problema mediante el *Método de las Diferencias Finitas*, tomando $h = \frac{L}{10}$.

Apéndice A

El análisis numérico y la ingeniería

Una de las características principales de la ingeniería es la utilización de herramientas físicas y matemáticas para encarar y resolver cada uno de los problemas que enfrenta. Por algo los estudiantes de ingeniería estudian en los primeros años una gran cantidad de matemática y física (y química también), que serán la base para aprender los temas más específicos. Es común que hoy muchos estudiantes acepten que deben aprender Análisis Matemático (o Cálculo), Álgebra, Física, Química sin hacerse demasiadas preguntas al respecto, aunque en muchas ocasiones no tienen bien claro cuál es el uso de cada una de estas disciplinas en la carrera elegida. Es por eso que uno de los grandes temas que se debaten en la actualidad es cómo articular los conocimientos básicos y los conocimientos específicos que deben ser aprendidos y captados por los futuros ingenieros.

Dentro de estos conocimientos básicos es donde suele ubicarse al Análisis Numérico, asumiendo que se trata de una herramienta matemática esencial para todo ingeniero. Pero resulta que no está claro cuál es la incidencia de esta herramienta dentro de la gama de problemas que resuelve la ingeniería ni cómo interactúa con el resto de los conocimientos que ésta incluye.

El capítulo inicial comenzó con una definición del Análisis Numérico dada por el Dr. Lloyd N. Trefethen, profesor de Análisis Numérico de la Universidad de Oxford en Inglaterra, que es la que actualmente se considera como la más representativa:

Análisis numérico es el estudio de los algoritmos para resolver problemas de la matemática continua.

En su artículo «Numerical Analysis» (ver [19]), Trefethen se expresa bastante acerca de los alcances del análisis numérico, su historia, la necesidad que de ella tienen los científicos y los ingenieros, y propone o expone cuál es a su entender el futuro para esta disciplina. Pero la visión es la de un matemático orientado casi exclusivamente a una rama de la matemática aplicada, aunque hoy puede decirse que casi es una rama más de la matemática teórica.

La visión de los ingenieros (y de los estudiantes de ingeniería) probablemente no coincida con la de Trefethen pues, en general, pocas veces se dedican a estudiar cuestiones teóricas o no disponen de tiempo para refinar o proponer nuevos algoritmos o métodos de resolución para problemas conocidos o nuevos. Generalmente, están más dispuestos a aprender cómo usar los modelos matemático-numéricos, entender cómo es la aplicación práctica y eventualmente como mejorar su aplicación pero sin avanzar en mejoras explícitas de los métodos sino mediante un uso no del todo convencional, aprovechando al máximo las capacidades de los modelos. Incluso, la utilización generalizada hoy de programas que permiten manejar la matemática simbólica casi de la misma forma que en la hoja de papel, hace parecer que el entender los procesos que llevan a la construcción de los algoritmos que aplican esos modelos sea, en cierta forma, superflua o innecesaria.

Sin embargo, esa visión puede ser equivocada. ¿Y por qué? Buena parte de los métodos numéricos aplicados en la actualidad tuvo su primer desarrollo embrionario en manos de ingenieros necesitados de resolver problemas sin soluciones conocidas. Un ingeniero suele tener diferentes

problemas para encarar y resolver. Existen aquellos que ya fueron resueltos hace muchos años, que cuentan con soluciones teóricas, a veces aproximadas, que se ajusten casi perfectamente al problema y que no suele requerir demasiadas herramientas matemáticas. Hoy se pueden citar como ejemplos de esto a una gran cantidad de temas de la ingeniería estructural relacionados con la mecánica del sólido, de la ingeniería mecánica, de la electrónica, etc. Existe una gran cantidad de ingenieros que, con ayuda de las herramientas matemáticas con niveles de complejidad media, resuelven una gran cantidad de problemas, incluso con ayuda de programas que en algún sentido se pueden considerar como elementales. El uso de planillas de cálculo, hojas matemáticas (como el MathCAD o el SMath Studio, por ejemplo) o entornos como MatLab, Octave o Maxima da la idea de estar utilizando algo equivalente a la hoja de papel que utilizaban los ingenieros antes de la aparición de la computadora personal. Pero en rigor no es lo mismo.

Existe una segunda parte en la ingeniería y puede resumirse en esto: puede decirse que hay una cantidad similar de temas que no cuentan con una solución matemática conocida o por lo menos, cuya aplicación sea sencilla o práctica para el ingeniero, y en algunos casos, todavía no hay una comprensión total del tema. Lo que no impide que sean temas que deben ser resueltos de alguna forma o por lo menos analizados hasta el nivel de conocimiento actual para dejarlos en evidencia y resolverlos a medida que se manifiestan. Y tal vez, con los análisis «en directo» se pueda encontrar una solución práctica, aunque fuere circunstancial. Esta es la base de la aplicación de los modelos físicos a escala.

Y esto es así porque una de las cualidades del ingeniero es la necesidad de que todo aquello en lo que trabaja o desarrolla sea aplicado de la forma más eficiente y completa en la vida real por la sociedad en la que vive. Diseñar puentes o máquinas, por citar dos ejemplos, que solamente sean expuestas en museos o exposiciones de tecnología no es precisamente el ideal de cualquier ingeniero en ejercicio de su profesión. En todo caso, si está desarrollando algo nuevo y para ello debe primero fabricar un prototipo (o un modelo en escala), lo que busca es poder entender y resolver el problema a partir del cual no cuenta con las soluciones matemáticas tradicionales o corrientes y que ese mismo prototipo, una vez adecuado y ajustado a las condiciones reales, sea construido y puesto a disposición de la sociedad. También aspira a que los resultados de analizar varios de esos modelos físicos le permitan, a veces con gran ayuda de los matemáticos y los físicos, encontrar un modelo matemático que solucione el problema sin tener que apoyarse permanentemente en el modelo físico. Porque, ya se sabe, los modelos físicos son muy caros y difíciles de ejecutar.

Dentro de esta visión, el análisis numérico queda relegado a un conocimiento adicional pero que pocas veces aplicará en su vida profesional. ¿Pocas veces? Un ejemplo de que esto no es cierto se puede ver en la utilización de sistemas de ecuaciones lineales para resolver innumerables problemas a lo largo del período de estudio. Los estudiantes de ingeniería «pasan» muchas horas de su vida académica resolviendo cantidad de ejercicios cuya base de resolución es el planteo de un sistema de ecuaciones lineales ($Ax = B$). ¿Y cuál es el método para resolverlos? Lo más usual es que traten siempre de usar lo aprendido en Álgebra, o sea, plantear $x = A^{-1}B$. Rara vez se plantean usar un método numérico aprendido en las clases de Análisis Numérico. Esto es «ayudado» en muchos casos por la facilidad que programas como MS Excel, el MatLab o el MathCAD, entre otros, pueden invertir la matriz. Nunca, o casi nunca, se plantean cómo hacen esos programas para obtener los que les parece «elemental». Tampoco se plantean si pueden obtener un mejor resultado que el que les entrega el programa o mejor dicho, la función del programa.

Aquí podría surgir la pregunta: ¿es posible mejorar el resultado que entrega un programa? Por ejemplo, la inversión de una matriz por el MS Excel, ¿puede ser mejorada? Aquí entra el concepto del Análisis Numérico como herramienta del ingeniero. Supuestamente, no sería necesario analizar esto puesto que si el MS Excel entrega un resultado, el mismo debe ser correcto. La pregunta que un alumno o ingeniero debería hacerse no es si el resultado es correcto o no sino, ¿qué algoritmo utiliza MS Excel para obtener esa matriz inversa? Mejorar los resultados no

pasa por la «exactitud» de los resultados en pantalla solamente sino por la rapidez, confiabilidad y robustez del procedimiento, que debe asegurar, por supuesto, valores numéricos precisos.

Si los ingenieros sólo se hubieran preocupado por resultados numéricos correctos, posiblemente seguirían usando los mismos procedimientos de cálculo que en su momento usó el ing Alexandre Eiffel para calcular y luego construir la torre que lleva su nombre en París. Pero eso no fue lo único que les preocupó o interesó. Hoy ningún ingeniero estructural trabaja con los métodos gráfico-numéricos del siglo XIX y de principio del siglo XX. Tampoco utiliza los mismos criterios de cálculo que en esa época. Lo mismo puede decirse para otras ingenierías y ramas de la ciencia. (Como anécdota fuera de la ingeniería, los astrónomos y astrofísicos actuales están sorprendidos por la precisión de los cálculos astronómicos de Le Verrier, un matemático francés dedicado a la mecánica celeste, para analizar las perturbaciones de la órbita de Urano, que le permitieron pronosticar la existencia de un planeta desconocido como causa de dichas perturbaciones. Gracias a sus indicaciones, el astrónomo alemán Johann Galle, del Observatorio de Berlín, lo encontró en la noche del 23 de septiembre de 1846, apenas cinco horas después de recibir la carta enviada por Le Verrier. Ese planeta desconocido es hoy Neptuno.)

Gracias a la rápida expansión del uso de las computadoras y sus programas asociados, muchos de los problemas que la ingeniería tardaba días y meses en resolver, hoy pueden ser resueltos en cuestión de horas. Esto le permite a los ingenieros imaginar y llevar a cabo soluciones nuevas y más eficientes que se comportan y se aproximan mejor a la realidad. Y al mismo tiempo, evitan el uso de modelos físicos que son engorrosos y, en algunos casos, muy costos. Sólo apelan a ellos en momentos indispensables en los que hay que validar el diseño. (Un ingeniero aeronáutico señaló alguna vez que «nunca volaría en un avión cuyo diseño no fuera probado y calibrado en un túnel de viento».)

Por eso creemos que el análisis numérico debería ser considerado por todos los ingenieros y, particularmente, por los estudiantes de ingeniería, como la otra herramienta práctica más importantes con la que disponen para su profesión, junto con el dibujo y la representación gráfica. Sin estas dos, difícilmente un ingeniero pueda desarrollar su trabajo con eficiencia capacidad e idoneidad.

Apéndice B

Un poco de historia

Es poco o nada lo que se ve y aprende de historia de la matemática (y de las ciencias en general) en la escuela media y en la universidad. No se considera como un conocimiento que sirva para entender los procesos, el razonamiento y la lógica que sustentan esta disciplina. Esto se puede extender a todas las ramas, como la geometría, el álgebra, el análisis matemático y, por supuesto, el análisis numérico. Eso ha llevado a que no se conozca cómo fue en realidad la evolución a través del tiempo de la matemática como disciplina y herramienta de la humanidad para resolver una gran cantidad de problemas y cuestiones que resultaron en la evolución de la humanidad propiamente dicha. Sin esa evolución de la matemática no se habría alcanzado el grado de desarrollo que todos conocemos hoy.

Suponer que el análisis numérico es una disciplina moderna y que nació junto con el análisis matemático o con el álgebra, es probablemente una concepción simplista del asunto. Hay una tendencia a considerar que la primera disciplina matemática en desarrollarse fue la geometría, pues la forma más sencilla de obtener resultados numéricos en la antigüedad era mediante la representación gráfica. Lo interesante es que la segunda disciplina dentro la matemática que se comenzó a desarrollar, tal vez a la par de la geometría, fue el análisis numérico o, como se lo llamaba antes, el *cálculo numérico*.

La necesidad de calcular cuantos trabajadores era necesario para construir un canal de irrigación (como en Babilonia), o las tierras que eran necesarias para la siembra (en Egipto), o cualquier otro caso que sólo era posible representar mediante cálculos aritméticos, llevó a dos cosas que son la base de la matemática actual: la representación numérica, y la construcción de algoritmos de cálculo, esto es, el desarrollo del análisis numérico.

Cada pueblo de la antigüedad desarrolló su propio sistema de representación numérica y sus formas de calcular y resolver los diferentes problemas que el avance tecnológico les presentaba. Así, entre otras cosas, tanto los babilonios, los egipcios, los indios, los chinos, los griegos, los romanos, los árabes, los mayas y los incas, entre los más destacados, desarrollaron sus propios sistemas numéricos.

Es opinión universal que el pueblo que mayor desarrollo le impregnó a la matemática en la antigüedad fue el griego. Los matemáticos griegos de la antigüedad se caracterizaron por ser brillantes y desarrollaron una ciencia de una forma jamás vista antes, y con una proyección a futuro muy importante, tanto que sin ella la civilización actual no sería lo que es. La brillantez de los griegos se observa en que estuvieron a punto de descubrir el cálculo diferencial e integral (Arquímedes desarrolló un método numérico de integración), cuyo descubrimiento por Newton en el siglo XVII revolucionó la matemática y la física de la época, y entre otras cosas permitió confirmar el modelo heliocéntrico propuesto por Copérnico y desarrollado por Kepler¹. Hay que recordar la cantidad de teoremas adjudicados a griegos que se estudian (por ejemplo, el famoso *Teorema de Pitágoras*) y la importancia de la obra de Euclides en geometría, tanto que recién en

¹Lo notable es que Aristarco de Samos propuso un modelo heliocéntrico del sistema solar que finalmente no tuvo cabida en la ciencia «oficial» de la época, dominada por las ideas de Aristóteles y la teoría geocéntrica.

el siglo XIX se comienza a imaginar una geometría no euclidea², considerada como una gran «herejía» entre algunos matemáticos.

Por lo tanto, estamos muy acostumbrados y conocemos casi todas las contribuciones de los matemáticos griegos. Pero la matemática no se construyó sólo con los matemáticos griegos, muy por el contrario, se nutrió de muchas y fundamentales contribuciones de matemáticos no griegos. Veamos un poco cuales fueron algunos de esos aportes, alguno de los cuales fueron decisivos y cambiaron el rumbo de la matemática.

B.1. Los egipcios

Los egipcios construyeron un imperio que duró varios milenios, hasta que fueron conquistados por los macedonios (Alejandro Magno) y los griegos, para luego ser nuevamente conquistados por los romanos y luego por los turcos y los franceses y finalmente por los ingleses.

Como la idea es analizar los primeros pasos de la matemática y en particular del análisis numérico, nos ocuparemos de los egipcios del período antiguo, entre los años 3,000 a 600 antes de Cristo (A.C.). Por lo pronto, el Antiguo Egipto es muy conocido por la construcción de las pirámides, sobre todo por las tres más famosas: las de *Keops* (o Jufu, también Khufu), *Kefren* (o Jafra, también Khafra) y *Micerino* (o Menkaura), conocidas como las Pirámides de Guiza, Giza o Gizeh (en árabe, «*Al Yiza*»). Alcanza con estas construcciones para imaginar el gran conocimiento práctico que tenían en arquitectura e ingeniería, lo que lleva a considerar que también llegaron a tener algún tipo de conocimiento matemático. Hay que mencionar que fueron ellos, los egipcios, los que dividieron el año en 365 días al confeccionar un calendario solar para determinar los períodos en los cuales debían sembrar los granos para alimentar a la población.

El concepto que tenían de la matemática en realidad era de un procedimiento práctico para resolver problemas. Se han encontrado varios elementos que documentan esto, en los cuales, está el sistema de representación numérica. No se trató de un sistema posicional como el actual, sino de un sistema parecido al romano, lo que hacía que la única operación posible de hacer era la suma. La multiplicación y la división eran prácticamente imposibles.








						
1	10	100	1000	10000	100000	10 ⁶

Figura B.1: *Números egipcios en escritura jeroglífica.*

En la figura B.1 están representados los números egipcios en escritura jeroglífica. Pero los egipcios contaban con otro sistema de escritura, el hierático, que era el usado por los escribas, los funcionarios encargados de los registros y de los cálculos. Son estos números los que se usaban para los cálculos matemáticos (figura B.2).

Ambas representaciones numéricas hicieron que los cálculos fueran muy complicados, pues el sistema numérico no era posicional, tal como el actual. No obstante, se las ingenieron para resolver muchos problemas ingenieriles y contables. Existen dos documentos históricos que dan muestra de esto: el *Papiro de Rhind* y el *Papiro de Moscú*, que datan del año 1,650 A.C., si bien el primero refiere estar basado en un documento anterior del año 1,850 A.C. El *Papiro de Rhind* incluye 87 problemas en tanto que el *Papiro de Moscú* incluye 25. Todos estos problemas fueron encarados de un modo estrictamente práctico.

Uno de los grandes problemas ingenieriles resueltos fueron las pirámides, en particular, la Gran Pirámide de Guiza, la pirámide de Keops. El relevamiento de las medidas de la pirámide

²Sin la geometría no euclidea, cuyo máximo exponente fue Georg Riemann, Einstein nunca hubiese podido desarrollar su *Teoría General de la Relatividad*.

1	𐀀	10	𐀁	100	𐀂	1000	𐀃
2	𐀄	20	𐀅	200	𐀆	2000	𐀇
3	𐀈	30	𐀉	300	𐀊	3000	𐀋
4	𐀌	40	𐀍	400	𐀎	4000	𐀏
5	𐀐	50	𐀑	500	𐀒	5000	𐀓
6	𐀔	60	𐀕	600	𐀖	6000	𐀗
7	𐀘	70	𐀙	700	𐀚	7000	𐀛
8	𐀜	80	𐀝	800	𐀞	8000	𐀟
9	𐀠	90	𐀡	900	𐀢	9000	𐀣

Figura B.2: *Números egipcios en escritura hierática.*

llevó a que algunos investigadores esgrimieran que los egipcios conocieron la *proporción áurea* ($\frac{\sqrt{5}+1}{2}$) y el número π , en tanto que otros argumentan que no era ese el caso, y que los números hallados fueron una coincidencia.³ En algo que sí concuerdan es que tenían la noción geométrica del triángulo de Pitágoras de lados 3 y 4 e hipotenusa 5. (Ese triángulo cumple con el teorema, pues $3^2 + 4^2 = 5^2$.)

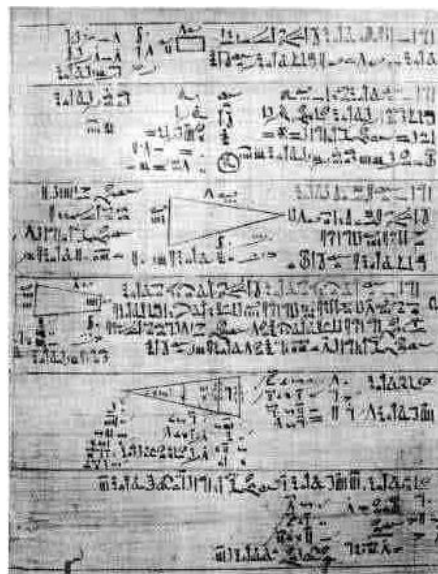


Figura B.3: *El Papiro de Rhind.*

Lo más notable es que los egipcios desarrollaron una serie de procedimientos para resolver varios de esos problemas que se encuentran en los documentos mencionados. Podemos decir que

³Como acotación a ese debate, es indiferente que conocieran o no el número π o a la proporción áurea tal como los conocemos hoy, pues lo importante es que de alguna forma los calcularon o aproximaron y los usaron en forma práctica. Con lo cual podría decirse que estaban en el camino correcto que luego derivó en la matemática abstracta de los griegos.

se dedicaron a construir algoritmos de cálculo sencillos, o lo que es lo mismo, se dedicaron a una forma incipiente del análisis numérico. Por ejemplo, en el *Papiro de Rhind* hay una forma preliminar del método de la *Regula-falsi* para resolver ecuaciones lineales.⁴ Pero los egipcios todavía están algo lejos de desarrollar algo parecido a procedimientos de cálculo sistematizados, aún cuando muchos suponen que la influencia sobre el resto de las culturas contemporáneas y posteriores es importante y notable.

B.2. Los babilonios

Los primeros babilonios fueron un pueblo que vivieron en la zona de la Mesopotamia asiática, en lo que hoy es parte de Irak, entre los años 2,000 y 1,500 A.C. Fueron tal vez los primeros en desarrollar una matemática incipiente y, particularmente, en crear los primeros pasos de la matemática que hoy conocemos. Esto se debió a que las ciudades babilónicas contaban con importantes avances tecnológicos para la época como son los sistemas de canales de irrigación, que no solo servían para transportar agua, sino también mercaderías y armas.

Para entender la capacidad de los babilonios, hay que tener en cuenta que fueron unos astrónomos extraordinarios. Ejemplo de ello es que gracias a sus observaciones astronómicas dividieron el día en 24 horas, las horas en 60 minutos y los minutos en 60 segundos. Es decir, crearon un sistema de representación numérica sexagesimal. Creo que eso basta para captar enorme inteligencia de los «matemáticos» babilónicos, pues estamos usando ese mismo sistema, ¡4,000 años después! Y no sólo lo usamos para medir el tiempo, sino que también para medir la posición geográfica y los ángulos planos.

1	∟	11	∟∟	21	∟∟∟	31	∟∟∟∟	41	∟∟∟∟∟	51	∟∟∟∟∟∟
2	∟∟	12	∟∟∟	22	∟∟∟∟	32	∟∟∟∟∟	42	∟∟∟∟∟∟	52	∟∟∟∟∟∟∟
3	∟∟∟	13	∟∟∟∟	23	∟∟∟∟∟	33	∟∟∟∟∟∟	43	∟∟∟∟∟∟∟	53	∟∟∟∟∟∟∟∟
4	∟∟∟∟	14	∟∟∟∟∟	24	∟∟∟∟∟∟	34	∟∟∟∟∟∟∟	44	∟∟∟∟∟∟∟∟	54	∟∟∟∟∟∟∟∟∟
5	∟∟∟∟∟	15	∟∟∟∟∟∟	25	∟∟∟∟∟∟∟	35	∟∟∟∟∟∟∟∟	45	∟∟∟∟∟∟∟∟∟	55	∟∟∟∟∟∟∟∟∟∟
6	∟∟∟∟∟∟	16	∟∟∟∟∟∟∟	26	∟∟∟∟∟∟∟∟	36	∟∟∟∟∟∟∟∟∟	46	∟∟∟∟∟∟∟∟∟∟	56	∟∟∟∟∟∟∟∟∟∟∟
7	∟∟∟∟∟∟∟	17	∟∟∟∟∟∟∟∟	27	∟∟∟∟∟∟∟∟∟	37	∟∟∟∟∟∟∟∟∟∟	47	∟∟∟∟∟∟∟∟∟∟∟	57	∟∟∟∟∟∟∟∟∟∟∟∟
8	∟∟∟∟∟∟∟∟	18	∟∟∟∟∟∟∟∟∟	28	∟∟∟∟∟∟∟∟∟∟	38	∟∟∟∟∟∟∟∟∟∟∟	48	∟∟∟∟∟∟∟∟∟∟∟∟	58	∟∟∟∟∟∟∟∟∟∟∟∟∟
9	∟∟∟∟∟∟∟∟∟	19	∟∟∟∟∟∟∟∟∟∟	29	∟∟∟∟∟∟∟∟∟∟∟	39	∟∟∟∟∟∟∟∟∟∟∟∟	49	∟∟∟∟∟∟∟∟∟∟∟∟∟	59	∟∟∟∟∟∟∟∟∟∟∟∟∟∟
10	∟∟∟∟∟∟∟∟∟∟	20	∟∟∟∟∟∟∟∟∟∟∟	30	∟∟∟∟∟∟∟∟∟∟∟∟	40	∟∟∟∟∟∟∟∟∟∟∟∟∟	50	∟∟∟∟∟∟∟∟∟∟∟∟∟∟		

Figura B.4: *Números babilónicas.*

Pero no se quedaron con eso. Hay que considerar que el sistema numérico de los babilonios fue el primer sistema posicional, lo cual muestra lo avanzados que estaban en términos matemáticos. Además del sistema numérico posicional que idearon, gracias a que adoptaron la escritura cuneiforme en tablas de arcilla cocida, se dedicaron a fabricar muchas tablas con ayudas numéricas. Por ejemplo, tenían tablas con los cuadrados (a^2) de 59 números y con los cubos (a^3) de 32 números, por supuesto, en el sistema sexagesimal. Esto los indujo a considerar la formulación de métodos de cálculo que facilitaran el uso de dichas tablas.

⁴Es muy interesante el análisis del Papiro de Rhind que está en [14]



Figura B.5: *Ejemplo de tablas babilónicas.*

Así, para calcular el producto de dos números ($a \cdot b$), disponían de dos procedimientos:

$$a \cdot b = \frac{(a+b)^2 - a^2 - b^2}{2} \text{ o,}$$

$$a \cdot b = \frac{(a+b)^2 - (a-b)^2}{4}.$$

Convirtieron el producto en la suma algebraica de tres cuadrados divididos por 2 en el primer caso, y de dos cuadrados divididos por 4, en el segundo. Con el segundo algoritmo, sólo tenían que buscar en la tabla los cuadrados de la suma y la diferencia de dos números, tomar un cuarto de esos resultados y restarlos. ¡Impresionante! Esto no es otra cosa que un algoritmo según el concepto moderno del mismo.

Alguien podría objetar que resolvieron el producto pero que nada hicieron con la división. Efectivamente, la división es más complicada y siempre generó dificultades en la antigüedad. Pero los babilonios encontraron que hacer $\frac{a}{b}$ es lo mismo que hacer $a \cdot \frac{1}{b}$. ¿Qué hicieron? Pues fabricaron tablas con los inversos de los números.

Con las tablas que fabricaron se dedicaron a resolver una gran cantidad de problemas concretos. Tomemos por ejemplo el siguiente: determinar los lados de un rectángulo cuya área es 0,75 y su diagonal es $1,25^2$. Escrito en álgebra actual equivale al siguiente sistema de ecuaciones:

$$x \cdot y = 0,75$$

$$x^2 + y^2 = 1,25^2.$$

La segunda ecuación no es otra cosa que ... ¡el Teorema de Pitágoras! Por supuesto, los babilonios conocían dicho teorema pero no con la formulación algebraica actual.⁵ Sabían que $\sqrt{2}$ era la diagonal de un cuadrado con lado unitario. En rigor, para ellos $\sqrt{2} = 1,4142963$, en vez de $\sqrt{2} = 1,414213562$.⁶

Pero lo más interesante es la forma en que resolvieron el sistema anterior: utilizaron un algoritmo «casi» moderno. La secuencia de procedimiento es la siguiente:

1. Calcule $2xy \rightarrow 1,50$

⁵Pitágoras tampoco lo escribió en la forma algebraica actual. El álgebra, tal como lo conocemos ahora, fue un desarrollo de los matemáticos árabes.

⁶Ver en [14] Pythagoras's theorem in Babylonian mathematics.

2. Reste $2xy$ a $x^2 + y^2 \rightarrow 1,25^2 - 1,50 = 0,0625$
3. Saque la raíz cuadrada para obtener $x - y \rightarrow \sqrt{0,0625} = 0,25$
4. Divida por 2 para obtener $\frac{x-y}{2} \rightarrow \frac{0,25}{2} = 0,125$
5. Divida $x^2 + y^2 - 2xy$ por 4 $\rightarrow \frac{0,0625}{4} = 0,015625$
6. Sume xy a este resultado $\rightarrow 0,015625 + 0,75 = 0,765625$
7. Saque la raíz cuadrada para obtener $\frac{x+y}{2} \rightarrow \sqrt{0,765625} = 0,875$
8. Sume $\frac{x-y}{2}$ a $\frac{x+y}{2}$ para obtener $x \rightarrow 0,875 + 0,125 = 1$
9. Reste $\frac{x-y}{2}$ a $\frac{x+y}{2}$ para obtener $y \rightarrow 0,875 - 0,125 = 0,75$
10. Solución: $x = 1$ y $y = 0,75$.

Es un ejemplo que hoy se usaría como pseudocódigo de programación, que se encontró en una tabla conocida como *Tabla de Tell Dhiyabi*, salvo que los números están representados en el sistema sexagesimal babilónico. ¡Un algoritmo maravilloso!

B.3. Los indios

Es más que evidente el gran aporte de los matemáticos indios a la matemática en general. Y no es otra cosa que el sistema numérico que usamos hoy en forma cotidiana para la representación nuestro sistema decimal. Se trata de los mal llamados «números arábigos», que fueron en realidad una creación de los matemáticos indios, que adoptados y modificados por los árabes, fueron introducidos en Occidente en el siglo XIII por *Leonardo de Pisa*, más conocido como «*Fibonacci*».

La cultura india tuvo varias etapas. La más antigua se remonta a alrededor del 2,500 A.C. sobrevivió hasta el 1,700 A.C. A esta etapa se la conoce también como la civilización Harappan. Aquí se destaca el uso de un sistema decimal de medidas cuya unidad de medición fue la pulgada del indo, que medía unos 3,35 cm (o 1,32 pulgadas actuales).

El siguiente avance o aporte se produjo a partir del 800 A.C., época en se escribieron los *Sulbasutras*, compilación del conocimiento matemático entre el 800 y el 200 A.C. Aquí el aporte se concentró en geometría y tenía un fin religioso, pues domina la religión védica que favoreció el desarrollo matemático asociado a la astronomía.

Esta religión fue reemplazada por el jainismo a partir del siglo VI A.C., época en la que se creyó que comenzó una etapa declinante del conocimiento matemático indio. Hoy se sabe, en cambio, que se avanzó en la teoría de los números, en operaciones aritméticas, en geometría, ecuaciones lineales, cúbicas y cuárticas, y en el cálculo combinatorio. Además desarrollaron una teoría del infinito con varias clases de infinitos (¡2,000 años antes que Cantor!) y tenían nociones del logaritmos de base 2 (sorprendente también, pues el sistema numérico era decimal).

Es a partir del 500 después de Cristo (D.C.) cuando se desarrolló lo que se conoce como el período clásico de la matemática india, la cual es también el comienzo de una nueva era en la astronomía y la matemática. Una de las obras fundamentales de esta época es la del matemático Aryabhata, que introdujo la trigonometría en la India, lo que dio un gran impulso a la ciencias ya mencionadas. Además de Aryabhata, otros matemáticos importantes fueron sus casi contemporáneos Varahamhina y Yativrsaha.

Peró es el matemático Brahmagupta a principios del siglo VII D.C. quien hizo una de las mayores contribuciones a la matemática y a los sistemas numéricos al introducir el concepto de los números negativos y el cero. A eso, que ya es probablemente lo más revolucionario en su momento, hay que agregar que contribuyó con fórmulas y métodos de interpolación que inventó

para calcular tablas con la función *seno*. Resulta más que notable que el matemático que introdujo la mayor revolución en la matemática se haya dedicado al desarrollo de métodos de interpolación, algo que hoy es casi monopolizado por el análisis numérico.

1	2	3	4	5	6	7	8	9
—	=	≡	+	॥	५	७	५	७

Números Brahmi del siglo I D.C.

1	2	3	4	5	6	7	8	9
—	=	≡	५	॥	५	७	५	७

Números Gupta del siglo IV D.C.

1	2	3	4	5	6	7	8	9	0
१	२	३	४	५	६	७	८	९	०

Números Nagari del siglo XI D.C.

Figura B.6: Evolución de los números de la India.

Si nos concentramos en los números indios, pues resultan ser la mayor contribución a la matemática, veremos en la figura B.6 la evolución de los símbolos numéricos (o numerales) a lo largo de los siglos. Observemos que en los primeros dos casos, los símbolos son bastante sencillos para los números 1 a 3, en cambio, el resto cuenta con símbolos más trabajados, aún cuando en nada se parecen a los que usamos nosotros actualmente. En cambio, en el último gráfico, notemos que los números 1 a 3 tienen una grafía similar a la nuestra, lo mismo que el cero, no así para el resto, que mantiene cierta diferencia con los actuales. Como una curiosidad, podemos ver que el símbolo que usamos para el cuatro corresponde al número cinco y que el símbolo del ocho corresponde al número cuatro, en tanto que el símbolo del nueve se parece bastante al actual aunque invertido. Resulta que si bien el origen de los números que hoy usamos es de la India, la grafía corresponde a la árabe, quienes tomaron los números y los adaptaron a su grafía.

La introducción de los números indios no sólo facilitó la representación de los números si no que agregó algo más: el sistema posicional en base diez, gracias a la introducción del cero. Si bien los babilonios ya habían desarrollado un sistema posicional pero con base sesenta, no habían conseguido desarrollar o inventar el concepto de cero. Por lo tanto, estas dos contribuciones de la matemática india fueron la gran revolución para la matemática toda, que Occidente comenzó a usar en el siglo XV y adoptó en el siglo XVI D.C. en forma definitiva.

Hay todavía cierta controversia acerca del origen del símbolo del cero en los números indios. Algunos autores estiman que el origen del cero es propiamente de la India, en tanto que otros creen ver una adaptación de un símbolo parecido que usaban los astrónomos griegos (en particular, por Tolomeo en el siglo II D.C.) y que fue adoptado por los indios en el siglo I o siglo II D.C.⁷

B.4. Los chinos

Los chinos tuvieron durante los primeros siglos o milenios de su dilatada civilización un desarrollo matemático independiente del resto del mundo o del resto de las culturas, principalmente en el mundo antiguo. Esa independencia en el desarrollo estuvo signada por las caracte-

⁷Hay un artículo interesante en [14] acerca de la historia del cero, *A History of Zero*.

rísticas geográficas de China, separada del resto del continente asiático por cadenas montañosas y mares, y por el hecho de que las conquistas que sufrieron terminaron con la asimilación de los conquistadores a la cultura china y no al revés, como usualmente ocurrió.

La matemática china se caracterizó por un desarrollo conciso, a diferencia de la matemática griega, es decir, fue pensada como herramienta para resolver problemas concretos derivados del comercio, la arquitectura, la astronomía o el cobro de impuesto.

Si bien la civilización china data del año 1,000 A.C., el avance notable de la matemática se dio alrededor del siglo IV A.C. Fueron los primeros en usar unas «pizarras o tablas de cálculo», lo que supone el uso de un sistema numérico decimal. De todas formas, hay poca información del conocimiento matemático chino anterior al año 100 A.C. Sólo se cuenta con un libro del año 180 A.C. aproximadamente, más otros posteriores, aunque no están completos. Se conoce un texto astronómico, recopilación de conocimientos entre el 100 A.C. y el 100 D.C., que incluye una versión china del *Teorema de Pitágoras*.

—	==	≡	≡	≡	⊥	⊥	⊥	⊥
1	2	3	4	5	6	7	8	9
Ⅰ	Ⅱ	Ⅲ	Ⅳ	Ⅴ	⊥	⊥	⊥	⊥
1	2	3	4	5	6	7	8	9

Figura B.7: Representación numérica utilizada en las pizarras o tablas de cálculo.

Otra de las recopilaciones chinas en materia de conocimiento matemático es el libro «*Jiuzhang suanshei*» («Prescripciones matemáticas en nueve capítulos»). Este libro contiene muchos problemas con sus respectivas soluciones. Entre los temas que incluye están las soluciones para obtener las raíces cuadrada y cúbica mediante una técnica geométrica que es equivalente al uso del triángulo de Pascal, y la solución de sistemas de ecuaciones lineales, que logra obtener soluciones numéricas explícitas manipulando los coeficientes de las ecuaciones, que no es otra cosa que uno de los métodos más utilizados en el análisis numérico, conocido como «*Eliminación de Gauss*».

Otro de los logros chinos fue la aproximación del número π , al que definieron dentro de estos límites: $3,1415926 < \pi < 3,1415927$. El matemático *Zu Chongzhi* (429-501 D.C.) -o Tsu Ch'ung Chi- recomendaba tomar el valor $\frac{355}{113}$ como una muy buena y precisa aproximación, y $\frac{22}{7}$, para una precisión normal.⁸

Por otro lado, el desarrollo matemático chino estuvo muy ligado a la astronomía. Y en la astronomía, una de las herramientas más utilizadas era la interpolación, por lo que el astrónomo *Liu Zho* introdujo la interpolación cuadrática con un método de diferencias de segundo orden (siglo VI D.C.).

Como se escribió al principio, hasta el siglo XIV D.C., se puede considerar que la matemática china avanzó en forma autónoma respecto del resto del mundo, si bien hay algunos investigadores que encontraron el uso de un símbolo para el cero muy similar al usado por los matemáticos indios. Aún así, desarrollaron métodos para hallar raíces de polinomios (usaron un método conocido hoy como de *Ruffini-Horner* para ecuaciones hasta grado 10) y un método de interpolación cúbico similar al método de las diferencias progresivas de Newton.

⁸El método utilizado por Zu Chongzhi para obtener la aproximación del número π se basó en un polígono de 3,072 lados. Arquímedes, quien calculó el valor de π en el siglo III A.C. y es considerado como el primero en calcular ese número en forma sistemática, utilizó uno de 96 lados y lo definió como $\frac{223}{71} < \pi < \frac{22}{7}$. Esta coincidencia podría suponer que la matemática china en esa época ya estaba vinculada con el resto del mundo, tal vez más de lo que se piensa. De todos modos, la mayoría de los investigadores considera poco probable que Zu Chongzhi conociera el trabajo de Arquímedes.

—	≡	≡	≡	⌘
1	2	3	4	5
↗	†) (ㄥ	丨
6	7	8	9	10
∪	∩	∩	⌘	↗
20	30	40	50	60
⊖	⊖	⊖	⊖	⊖
100	200	300	400	500
ㄗ	ㄗ	ㄗ	ㄗ	ㄗ
1000	2000	3000	4000	5000

Figura B.8: *Números chinos.*

En épocas posteriores y hasta la actualidad, la matemática china se integró a la del resto del mundo y fue influida por el conocimiento occidental, pero manteniendo su particularidad en algunos temas.

Puede decirse que uno de los grandes aportes a la aritmética es el famoso ábaco chino, tal vez la primera herramienta de cálculo mecánica y antecesor de la regla de cálculo y las calculadoras modernas.

B.5. Los árabes

Si la matemática actual tiene una deuda con los matemáticos indios, que introdujeron el sistema numérico posicional que se utiliza en casi todo el mundo, tanto que algunos investigadores la llaman «la gran deuda», entonces puede considerarse que los otros grandes aportantes no debidamente reconocidos son los matemáticos árabes, quienes aportaron conocimientos y nuevos desarrollos a la matemática actual en el mismo nivel que los considerados padres de la matemática, los griegos.

Durante mucho tiempo los matemáticos árabes fueron considerados erróneamente como «reproductores» del conocimiento matemático griego, algo que hoy se sabe, no se ajusta a la verdad. Aún no se sabe a ciencia cierta cómo se desarrolló la matemática árabe a partir del año 800 D.C. en el área que hoy ocupan Irán e Irak, y en particular, en Bagdad, pero se estima que la influencia de los matemáticos indios fue determinante en este desarrollo, producto tal vez del comercio entre árabes e indios.

Los árabes comenzaron traduciendo los viejos textos griegos, es cierto, pero aportaron a la matemática otra de las grandes ramas de la matemática actual. Entre las traducciones más importantes hechas por los árabes se encuentran las de las obras de Euclides (entre ellas, los «Elementos», fundamental en geometría), algunas obras de Arquímedes y las obras completas de Diofanto y Menelao, en matemática, y el «Almagesto» de Claudio Tolomeo⁹, el mayor astrónomo y geógrafo de la antigüedad, que vivió alrededor del siglo II D.C., que fuera fundamental como piedra angular del conocimiento astronómico y geográfico entre el siglo VIII y el siglo XV D.C. en Asia menor y Europa.¹⁰

⁹En realidad, el nombre de la obra fue *El gran tratado*, en griego *Hè Megalè Syntaxis*, cuya traducción al árabe se denominó «*Al-Majisti*», que al ser traducido al latín tomó el nombre de «*Almagesto*».

¹⁰Es probable que sin el «Almagesto» de Claudio Tolomeo, Colón nunca hubiese evaluado navegar hacia el oeste en búsqueda de las Indias Orientales. Aún con el error manifiesto en la determinación del radio del globo terrestre, los mapas y los conocimientos astronómico y geográfico incluidos en dicha obra ayudaron a definir la idea de Colón.

1	2	3	4	5	6	7	8	9	0
1	2	3	4	5	6	7	8	9	0

Números de al-Sizjī. 969 D.C.

1	2	3	4	5	6	7	8	9	0
1	2	3	4	5	6	7	8	9	0

Números de al-Biruni. 1.082 D.C.

1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9

Números de al-Bana al-Marrakushi

Figura B.9: Números árabigos.

Para tener una idea cabal de la contribución de los árabes a la matemática actual, basta saber que una de las ramas más importantes de la matemática actual, el álgebra, nace con uno de los matemáticos más brillantes de la Edad Media: *Abu Ja'far Muhammad ibn Musa al-Khwarizmi*, o simplemente, *al-Khwarizmi*, quien vivió entre los años 780 y 850 D.C., aunque no hay datos certeros al respecto. Su gran revolución consistió en cambiar el viejo concepto griego de una matemática geométrica para pasar a una matemática unificadora de todos los conocimientos disponibles hasta el momento: **el álgebra**. Su obra más importante, «*Kitab al-Mukhtasar fi hisab al-jabr w'al-muqabala*», que traducido significa «*Libro conciso de cálculo de restauración y oposición*», es justamente la que sirvió para darle el nombre a esta nueva rama de la matemática («*al-jabr*»)¹¹ y, por supuesto, es el primer libro de álgebra. No contento con esto, la latinización de su apellido en las publicaciones occidentales, que se tradujo como «Algoritmi», «Algorizmi» o «Alchorismi», llevó a la creación de la palabra que hoy es fundamental en el análisis numérico: «**algoritmo**».

No sólo «inventó» el álgebra sino que además fue el que introdujo en el imperio árabe el sistema numérico posicional y los símbolos numéricos de la matemática india, en su obra «*Kitab al-hisab al-cadad al-hindi*», o sea, «*Libro del cálculo con los números hindúes*», con todo lo que ello significa. Posteriormente, otros matemáticos le dieron la forma actual a los números, que pasaron a ser conocidos como «números árabigos», y que son los que todos conocemos y usamos, entre ellos al-Biruni (*Abu Rayhan al-Biruni*) y al-Banna al-Marrakushi (*Abu'l-Abbas Ahmad ibn Muhammad ibn Uthman al-Azdi*), como se ve en la figura B.9.

A al-Khwarizmi se suma otro de los grandes matemáticos árabes, Omar Khayyam (*Ghiyath al-Din Abu'l-Fath Umar ibn Ibrahim Al-Nisaburi al-Khayyami*), de origen persa y que vivió en el siglo XI D.C., que se ocupó de mejorar el álgebra de al-Khwarizmi, al armar una clasificación completa de las ecuaciones cúbicas con soluciones geométricas que se basaron en intersecciones de secciones cónicas.

El matemático al-Karaji (*Abu Bakr ibn Muhammad ibn al-Husayn Al-Karaji*), que vivió entre 953 y 1,029 D.C., parece ser que fue el primero en liberar al álgebra de las operaciones geométricas como solución de los problemas y las reemplazó por operaciones aritméticas. También fue quien descubrió el teorema del binomio con exponentes enteros, que fue un factor fundamental en el desarrollo del análisis numérico con el uso del sistema decimal.

¹¹El nombre *álgebra* provino de la traducción al latín hecha por Gerardo de Cremona, aproximadamente en 1,170 D.C., cuyo título fue: «*Liber Maumeti filii Moysi Alchorismi de Algebra et almuchabala*», cuya traducción al castellano es: «*Libro de Mahoma, hijo de Moisés, Alchorismi, de Álgebra y almuchabala*».

incluía al cero, que era de base 20 (no de base 10 como el nuestro). Este notable sistema no fue un sistema posicional en el sentido actual, pues en la posición tres (el equivalente nuestro a las centenas) en lugar de corresponder a 20^2 , o sea, 400, correspondía a 360, es decir, a $18 \cdot 20$, para luego continuar con posiciones en base 20. Por ejemplo, el número [8; 14; 3; 1; 12] en notación maya, convertido a notación actual es:

$$12 \cdot 20^0 + 1 \cdot 20 + 3 \cdot 20 \cdot 18 + 14 \cdot 20 \cdot 18 \cdot 20 + 8 \cdot 20 \cdot 18 \cdot 20^2 = \\ 12 + 20 + 3 \cdot 360 + 14 \cdot 7,200 + 8 \cdot 144,000 = 1,253,912.$$

La descripción de este sistema se encontró en el *Código de Dresde*, el cual contiene buena parte del conocimiento astronómico de los mayas, por lo cual algunos investigadores consideran que es muy posible que contaran con al menos dos sistemas numéricos: uno para la astronomía, el que acabamos de mostrar, y otro para las operaciones normales, ese sí posicional completo y de base 20.¹³

El conocimiento astronómico de los mayas es más que asombroso. Fueron unos extraordinarios observadores que guardaban todas las mediciones y observaciones que hicieron. Eso los llevó a definir la duración del año solar en 365,242 días de duración, que estaba dividido en 18 meses de 20 días, a los que agregaban un «mini-mes» de 5 días al final del año. Ese calendario era el *calendario civil*. Es notable esa división del año, pues da la impresión que esa es la razón del sistema posicional visto antes. Además, contaban con otro calendario, el ritual o religioso, de 260 días al año, dividido en 13 meses de 20 días (otra vez la base 20). Esa diferencia entre ambos calendarios hacía que ambos coincidieran en un mismo ciclo cada 18,980 días, o sea, cada 52 años «civiles», que era justamente la duración del ciclo sagrado. Tengamos en cuenta que hoy en día la duración del año solar está calculado en 365,242198 días, lo que muestra la asombrosa capacidad de los mayas para las mediciones y los cálculos, Así también, fueron muy precisos para calcular el mes lunar. De acuerdo con las mediciones hechas en Copán (actualmente en la frontera de Guatemala y Honduras), el mes lunar dura 29,5302, en tanto que según las mediciones hechas en Palenque (México), el lunar dura 29,5308 días. El valor actual es 29,53059, casi el promedio de ambas mediciones. Nuevamente, ¡asombroso!

Lamentablemente, no se conocen otros detalles del desarrollo matemático de los mayas, pero sí se sabe que tuvieron un conocimiento geométrico muy avanzado, que se observa en la arquitectura.

B.7. Los incas

Los incas fueron imperio que se extendió entre el norte de Ecuador y la provincia de Mendoza (Argentina) al sur, y desde el altiplano de Bolivia y Perú hasta la costa del Océano Pacífico. En relación con otros imperios, incluyen el maya y el azteca, fue de muy corta duración, pues en su máximo apogeo alcanzó a durar unos 100 años, entre 1,430 y 1,530 aproximadamente, cuando fue conquistado y destruido por los españoles, en medio de una guerra interna por la sucesión del Inca.

Una de las peculiaridades del imperio inca es que no tuvieron un sistema de escritura tradicional, por lo que la transmisión de buena parte del conocimiento se hacía en forma oral o mediante representaciones o dibujos, tanto en telas como en vasijas u otros elementos, generalmente producto de la alfarería. Esta falta del sistema escrito «formal», ha dificultado enormemente entender el alcance de los conocimientos que habían alcanzado.

Aún así, tuvieron un sistema de registro contable, si se quiere, bastante avanzado, que consistió en un conjunto de cuerdas anudadas que servían como representación de cantidades (números). Estos registros se denominaban «quipu» y estaban basado en sistema de representación numérica de base 10. Era bastante complejo y permitía representar números bastante

¹³Esta hipótesis no es tan descabellada, pues algo parecido sucede hoy con el sistema numérico posicional decimal y el sistema sexagesimal usado en mediciones geográficas y astronómicas.

grandes, en el orden de 10^5 . Tengamos en cuenta que los números romanos eran muy pobres como representación numérica, lo mismo que los griegos, en algún sentido, en tanto que los números indo-arábigos eran muy eficientes, lo mismo que el sistema de los mayas. Esta forma de representación incluía una forma de cero, mediante la ausencia de nudos.

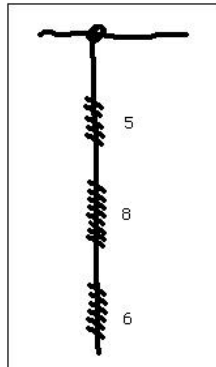


Figura B.11: *Representación del número 586 en un «quipu».*

El «quipu» era usado esencialmente como un sistema de registro contable o equivalente, lo que requería de personal capacitado para descifrarlos. Esos empleados, de alto rango, eran conocidos como «quipu-camayó». El sistema de registro permitía que de las cuerdas principales salieran cuerdas subsidiarias, lo cual pareciera darles una flexibilidad muy importante al momento de manejar esos registros.

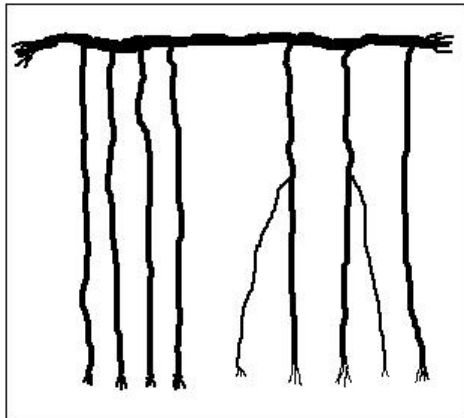


Figura B.12: *Ejemplo de un «quipu» con cuerdas subsidiarias.*

Como indicamos antes, La falta de un sistema de escritura tradicional ha dificultado el entendimiento de la matemática inca. Sí se conocen elementos que se asocian a una especie de ábaco inca o pizarra (o tabla) de cálculo, cuyo nombre es «yupana», que de acuerdo con los investigadores, era un sistema de ayuda en los cálculos numéricos. Pero también hay otros que indican que los mismos «quipus» eran usados como elementos de cálculo.

Dentro los conocimientos avanzados de los incas se encuentran la hidráulica, la ingeniería, la arquitectura y la astronomía. En los pocos años que duró el imperio, construyeron varios canales de irrigación en la zona del Cusco (o Cuzco), grandes áreas para cultivo en forma de terrazas escalonadas con canales de irrigación¹⁴, carreteras para la comunicación del imperio¹⁵ y templos

¹⁴En la Quebrada de Humahuaca quedan restos de una de ellas en la zona de Coctaca, a unos 6 km al este de la ciudad de Humahuaca.

¹⁵Lo curioso es que las carreteras no fueron hechas para vehículos sino para mensajeros a pie, los «chasquis», pues no conocieron la rueda.

y edificios que se caracterizaron por un espectacular conocimiento en el uso de la piedra como elemento base en el diseño de estructuras.

Y respecto del conocimiento astronómico, los incas usaron un calendario solar de 365 días, muy similar al usado por los mayas y los egipcios, aunque con algunas peculiaridades debidas a su sistema numérico. Existen algunos investigadores que creen que para los cálculos astronómicos lo incas contaban con un sistema numérico similar al maya, pero de base 36 y 40. Es por eso que debería seguirse la investigación acerca de la matemática inca, pues es probable que se encuentren muchos más datos y de esa forma, tener un mejor entendimiento de los conocimientos matemáticos que parecen haber sido muy importantes.

Bibliografía

- [1] Akima, H. *A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures*. J.ACM, vol. 17, no. 4, pp. 589-602, 1970.
- [2] Broyden. C. G. *A Class of Methods for Solving Nonlinear Simultaneous Equations*. Math. Comp. 19, 577-593, 1965.
- [3] Burden, R. L., Faires, J. D. & Burden, A. M. *Análisis Numérico*. Décima Edición, CENGAGE Learning, 2016.
- [4] Dennis Jr., J. E. & Moré, J. J. *Quasi-Newton methods, motivation and theory*. SIAM Review, Vol 19, No 1, pp. 46-89. January 1977.
- [5] Ezquerro, J. A., Gutiérrez, J. M., Hernández, M. A. y Salanova, M. A. *El método de Halley: posiblemente el método más redescubierto del mundo*. Universidad de La Rioja, España. 2001
- [6] Gavurin, M. K. *Conferencias sobre los métodos de cálculo*. Editorial Mir, 1973.
- [7] Goldberg, D. *What every Computer Scientist should know about Floating-Point Arithmetic*. ACM Computing Surveys, March 1991.
- [8] González, H. *Análisis Numérico, primer curso*. Primera Edición, Nueva Librería, 2002.
- [9] Grcar, J. F. *How Ordinary Elimination became Gaussian Elimination*. arXiv:0907.2397v4. 2010.
- [10] Higham, N. J. *Accuracy and Stability of Numerical Algorithms*. SIAM, 1996.
- [11] Higham, N. J. *How accurate is Gaussian Elimination*. Numerical Analysis 1989, Proceedings of the 13th Dundee Conference, volume 228 of Pitman research Notes in Mathematics.1990.
- [12] Higham, N. J. *The numerical stability of barycentric Lagrange interpolation*. IMA Journal of Numerical Analysis. 2004.
- [13] Marshall, G. *Solución numérica de ecuaciones diferenciales, Tomo I*. Editorial Reverté S.A., 1985.
- [14] O'Connor, J. J. & Robertson, E. F. *MacTutor History of Mathematics*. University of St Andrews. (<http://www-gap.dcs.st-and.ac.uk/history/Indexes/HistoryTopics.html>)
- [15] Saad, Y. *Iterative Methods for Sparse Linear Systems*. Second Edition, 2000.
- [16] Samarski, A. A. *Introducción a los métodos numéricos*. Editorial Mir, 1986.
- [17] Shewchuk, J. R. *An introduction to the Conjugate Gradient Method without the agonizing pain*. Edition 1 $\frac{1}{4}$. School of Computer Science. Carnegie Mellon University.
- [18] Trefethen, L. N. *The Definition of Numerical Analysis*. SIAM News. November 1992.

- [19] Trefethen, L. N. *Numerical Analysis*. Princeton Companion to Mathematics. 2008.
- [20] Trefethen, L. N. & Berrut, J. P. *Barycentric Lagrange Interpolation*. 2004.
- [21] Zill, D. G. *Ecuaciones diferenciales con aplicaciones de modelado*. Séptima Edición, International Thomson, 2002.