

JESÚS ABEL MEJÍA MARCACUZCO

MÉTODOS NUMÉRICOS EN
RECURSOS HÍDRICOS
(Aplicaciones con MATLAB)



UNIVERSIDAD NACIONAL AGRARIA LA MOLINA

UNIVERSIDAD NACIONAL AGRARIA LA MOLINA

DR. JESÚS ABEL MEJÍA MARCACUZCO
Rector

DR. JORGE LUIS ALIAGA GUTIÉRREZ
Vicerrector Académico

MG.SC. EFRAÍN DONALD MALPARTIDA INOUYE
Vicerrector de Investigación

DR. JOSÉ CARLOS VILCAPOMA
Jefe del Fondo Editorial

Jesús Abel Mejía Marcacuzco
MÉTODOS NUMÉRICOS EN RECURSOS HÍDRICOS (Aplicaciones con MATLAB)
Lima: Fondo Editorial - UNALM. 2014; 496p.

© Jesús Abel Mejía Marcacuzco
© Universidad Nacional Agraria La Molina
Av La Universidad s/n La Molina

Derechos reservados
ISBN: N° 978-612-147-36-4
Hecho el Depósito Legal en la Biblioteca Nacional del Perú:
N° 2014-08708

Primera Edición: julio del 2014 - Tiraje: 500 ejemplares
Impreso en Perú – Printed in Peru

Coordinación editorial:
José Carlos Vilcapoma

Diseño y diagramación de carátula:
Roxana Perales Flores

Diseño, diagramación e impresión :
Q y P Impresores S.R.L
Av. Ignacio Merino 1546 Lince
E-mail: qypimpresores2005@yahoo.com

Queda terminantemente prohibida por la Ley del Perú la reproducción total o parcial de esta obra por cualquier medio, ya sea electrónico, mecánico, químico, óptico, incluyendo sistema de fotocopiado, sin autorización escrita de la Universidad Nacional Agraria La Molina y los Autores.
Todos los conceptos expresados en la presente obra son responsabilidad de los autores.

“Muchas personas deben la grandeza de sus vidas a los problemas y obstáculos que tuvieron que vencer”

SPURGEON

La elaboración de este libro no hubiera sido posible sin el interés y ánimo que nos brindaron colegas, estudiantes y amigos. En especial profesores del Departamento de Recursos Hídricos de la Facultad de Ingeniería Agrícola y estudiantes de la maestría y doctorado en recursos hídricos de la Universidad Nacional Agraria La Molina, con quienes amablemente compartimos sus puntos de vista sobre la materia y motivaron a producir un trabajo que ofreciera, en lo posible, claridad sobre los temas tratados.

Dedico este trabajo a mi madre Laura y a la memoria de mi padre Lucio, personas humildes que supieron orientar sus hijos para enfrentar la vida con dignidad y perseverancia.

A mi esposa Redina, mi hijo Luis Abel y mis hermanos Lucio, Aquilino, Aydé, José, Neli y Rene por sus consejos y apoyo permanente.

CONTENIDO

PRESENTACIÓN	13
CAPÍTULO I	
INTRODUCCIÓN	15
1.1 Definición	15
1.2 Importancia de métodos numéricos en la ingeniería	15
1.3 Fundamentos matemáticos	16
1.4 Objetivos del texto	17
1.5 Lenguajes de programación y software para cálculo numérico	18
CAPÍTULO II	
FUNDAMENTOS DEL MATLAB	21
2.1 Introducción	21
2.2 Componentes del Matlab	21
2.3 Entorno de Trabajo de Matlab	22
2.4 Variables	24
2.4.1 Cálculos con una sola variable	24
2.4.2 Variables vectoriales	25
2.4.3 Variables matriciales	27
2.5 Números	32
2.5.1 Números enteros	34
2.5.2 Funciones con números enteros y divisibilidad	34
2.5.3 Números reales	35
2.5.4 Funciones con argumento real	36
2.5.5 Números complejos	38
2.5.6 Números aleatorios	38
2.6 Operadores	39
2.6.1 Operadores aritméticos	39
2.6.2 Operadores relacionales	40
2.6.3 Operadores lógicos	40
2.7 Funciones	40
2.7.1 Funciones matemáticas elementales	40
2.7.2 Funciones matemáticas especiales	42
2.7.3 Funciones de análisis de datos y análisis estadístico básico	44
2.8 Límites y Continuidad	46

2.9	Derivadas e Integrales	46
2.10	Ecuaciones Diferenciales	47

CAPÍTULO III

REPRESENTACIÓN GRÁFICA EN MATLAB		49
3.1	Graficando Datos	49
3.2	Gráficos 2D: Barras, Sectores, Histogramas, Racimo y Flechas	49
3.3	Gráficos 2D: Funciones Explícitas, Paramétricas y Polares	53
3.4	Títulos, Etiquetas y textos en gráficos	55
3.5	Gráfico de Líneas 3D	59
3.6	Formas Geométricas Especiales 3D	61
3.7	Superficies Explícitas, Paramétricas y Líneas de Contorno	63

CAPÍTULO IV

PROGRAMACIÓN EN MATLAB		69
4.1	Introducción	69
4.2	Funciones	69
4.3	Instrucción for	75
4.4	Instrucción While	77
4.5	Instrucción If-End	77
4.6	Instrucción If-Else-End	78
4.7	Instrucción Elseif	78
4.8	Instrucciones Switch-Case	79
4.9	Instrucciones Continue y Break	80
4.10	La Instrucción Return	80
4.11	Manejo de Archivos	81
4.12	Intercambio de Datos entre Matlab y Excel	83

CAPÍTULO V

INTERFAZ GRÁFICA EN MATLAB		85
5.1	Introducción	85
5.2	Creando una Gui en Matlab	85
5.3	Propiedades de los Controles	87
5.4	Una Gui Sencilla: Hola Amigos	89
5.5	Cálculo de Derivadas, Integrales y Transformadas de Fourier	92
5.6	Gui para Conversión de Temperaturas	95
5.7	Uso de Interfaz Gráfica para la Función Seno	98
5.8	Uso de Barra de Deslizamiento	100

CAPÍTULO VI

SIMULACIÓN DE SISTEMAS CON SIMULINK		105
6.1	Introducción	105
6.2	Inicio de Simulink	105
6.3	Tipos De Librerías En Simulink:	106
6.4	Instalación y Conexión de un Bloque Operacional	109
6.5	Creación de un Modelo Sencillo:	110

6.6	Sistema de Onda Sinoidal y Modelo Integrador	111
6.7	Sistema de Integración de Señales y tren de Pulsos	112
6.8	Sistema Descrito por una Ecuación Diferencial	113

CAPÍTULO VII

APROXIMACIONES Y ERRORES EN MÉTODOS NUMÉRICOS		117
7.1	Introducción	117
7.2	Cifras Significativas	117
7.3	Exactitud y Precisión	118
7.4	Definiciones de Error	118
7.5	Errores de Redondeo	119
7.6	Errores de Truncamiento y la Serie de Taylor	120

CAPÍTULO VIII

ÁLGEBRA LINEAL NUMÉRICA		125
8.1	Introducción	125
8.2	Matrices y Vectores	125
8.3	Operaciones con Matrices y Vectores	126
8.4	Operaciones con Matrices y Vectores con Matlab	129
8.5	Determinante de una Matriz	132
8.6	Sistema Lineal de Ecuaciones	132
8.7	Eliminación de Gauss	135
8.8	Eliminación de Gauss-Jordan e Inversión de Matrices	140
8.9	Descomposición Lu	143
8.10	Valores Propios de Matrices	147
8.11	Algebra Lineal Numérica en Matlab	150

CAPÍTULO IX

INTERPOLACIÓN NUMÉRICA		155
9.1	Introducción	155
9.2	Interpolación Lineal	155
9.3	Interpolación Polinómica Como Serie de Potencia	158
9.4	Fórmula de Interpolación de Lagrange	160
9.5	Fórmulas De Interpolación De Newton	165
9.6	Polinomios De Interpolación De Hermite	176
9.7	Interpolación En Dos Dimensiones	178
9.8	Interpolación Polinómica en Matlab	180

CAPÍTULO X

SOLUCIÓN DE ECUACIONES NO LINEALES		185
10.1	Introducción	185
10.2	Método de la Bisección	186
10.3	Método de la Falsa Posición y de la Falsa Posición Modificada	191
10.4	Método de Newton	193
10.5	Método de la Secante	197
10.6	Introducción a la Solución de Sistemas de Ecuaciones No Lineales	200

10.7	Comandos de Matlab para encontrar Ceros de Funciones	200
------	--	-----

CAPÍTULO XI

INTEGRACIÓN NUMÉRICA		203
11.1	Introducción	203
11.2	Regla del Trapecio	204
11.3	Regla de Simpson	207
11.4	Fórmulas de Newton-Cotes	212
11.5	Comandos de METALAB para integración	213

CAPÍTULO XII

DIFERENCIACIÓN NUMÉRICA		217
12.1	Introducción	217
12.2	Aproximación de primeras derivadas por diferencias finitas	218
12.3	Aproximación de segundas derivadas por diferencias finitas	222
12.4	Aproximación de derivadas parciales por diferencias finitas	223
12.5	Resumen de aproximaciones por diferencias finitas	224

CAPÍTULO XIII

ECUACIONES DIFERENCIALES ORDINARIAS		227
13.1	Introducción	227
13.2	Problemas de Valor Inicial	227
13.2.1	EDO de Primer Orden	227
13.2.2	Métodos de Euler hacia adelante	228
13.2.3	Método de Euler modificado	229
13.2.4	Método de Euler hacia atrás	232
13.2.5	EDO de segundo orden	233
13.2.6	Método de Runge-Kutta de Segundo Orden	236
13.2.7	EDO de orden superior	241
13.2.8	Método de Runge-Kutta de tercer orden	245
13.2.9	Método de Runge-Kutta de cuarto orden	246
13.3	Problemas de Valor en la Frontera	251
13.3.1	Problemas de valor en la frontera unidimensionales	251
13.3.2	Solución de la Matriz Tridiagonal	255

CAPÍTULO XIV

ECUACIONES DIFERENCIALES EN DERIVADAS PARCIALES		259
14.1	Introducción	259
14.2	Ecuaciones Diferenciales Parciales Elípticas (EDPE)	259
14.3	Solución Numérica en Diferencias Finitas	260
14.4	Solución Numérica de la Ecuación de Laplace	266
14.5	Solución Numérica de la Ecuación de Poisson	274
14.6	Ecuaciones Diferenciales Parciales Parabólicas (EDPP)	279
14.7	Aproximación En Diferencias Finitas	280
14.8	Esquemas Explícitos	281
14.9	Esquemas Implícitos	282

14.10	Problemas Parabolicos Bidimensionales	283
14.11	Conducción de Calor Unidimensional en Régimen Transitorio	284
14.12	Flujo Laminar entre Dos Placas Horizontales	289
14.13	Ecuaciones Diferenciales Parciales Hiperbólicas	291
14.14	Solución Numérica de las EDPH	292
14.15	Método de las Características	293
14.16	Solución Numérica de la Ecuación de Color	295
14.17	Esquema Numérico de lax-Wendroff para las EDPH	298

CAPÍTULO XV

FLUJO PERMANENTE EN CANALES ABIERTOS		301
15.1	Introducción	301
15.2	Energía Específica y Flujo Crítico	302
15.3	Flujo Uniforme en Canales	309
15.4	Flujo Gradualmente Variado	317
15.5	Solución de la Ecuación del Flujo Gradualmente Variado	319

CAPÍTULO XVI

FLUJO TRANSITORIO EN TUBERÍAS		325
16.1	Introducción	325
16.2	Oscilaciones de Presión en Chimeneas de Equilibrio	325
	16.2.1 Ecuación de Conservación de Masa	326
	16.2.2 Ecuación de Momento o Cantidad de Movimiento	326
16.3	Golpe de Ariete	332
	16.3.1 Ecuación de Cantidad de Movimiento	332
	16.3.2 Ecuación de Continuidad	334
	16.3.3 Solución por el Método de las Características	338
	16.3.4 Ecuaciones en Diferencias Finitas	341
	16.3.5 Condiciones de Contorno Básicas	343

CAPÍTULO XVII

TRÁNSITO HIDROLÓGICO DE ONDAS DE AVENIDAS		353
17.1	Introducción	353
17.2	Fundamento Teórico del Tránsito Hidrológico	354
17.3	Función De Almacenamiento	354
17.4	Tránsito de Avenidas en Embalses	356
17.5	Transito Hidrologico en Ríos	363

CAPÍTULO XVIII

TRÁNSITO HIDRÁULICO EN CANALES		369
18.1	Introducción	369
18.2	Ecuación de Conservación de Masa	370
18.3	Ecuación de Cantidad de Movimiento	371
18.4	Solución Analítica de la Onda Cinemática	374
18.5	Solucion Numérica de la Onda Cinemática	376
18.6	Solución numérica de las ecuaciones de Saint Venant	380

18.7	Método de las Características	381
18.8	Método de las diferencias finitas	386

CAPÍTULO XIX

MODELO MATEMÁTICO DE FLUJO EN CANALES		391
19.1	Introducción	391
19.2	Ecuación Tridimensional de Conservación de Masa	391
19.3	Ecuación Tridimensional de Conservación de Momento	393
19.4	Ecuaciones de Navier-Stokes	394
19.5	Ecuación Bidimensional de Conservación de Masa	396
19.6	Ecuación Bidimensional de Conservación de Momento	399
19.7	Modelación de las Tensiones Efectivas de Corte	400
19.8	Modelación de las Tensiones de Corte de Fondo	401
19.9	Modelación de la Presión	402
19.10	Resumen de las Ecuaciones de Flujo en dos dimensiones	403

CAPÍTULO XX

MÉTODO DE VOLÚMENES FINITOS		407
20.1	Introducción	407
20.2	Ecuación de Transporte	408
20.3	Esquemas de Discretización en Volúmenes de Control	409
20.4	Esquema de Diferencias Centrales	410
20.5	Esquema Upwind	411
20.6	Solución Exacta	412
20.7	Esquema Exponencial	413
20.8	Esquema Híbrido	414
20.9	Esquema Potencial	415
20.10	Formulación General	415
20.11	Discretización en dos Dimensiones	417
20.12	Discretización en Tres Dimensiones	419
20.13	Cuidados en la Obtención de las Ecuaciones de Discretización	421
20.14	Arreglo Co-Localizado de las Variables Dependientes	421
20.15	Arreglo Escalonado De Las Variables Dependientes	423
20.16	Acoplamiento Presión - Velocidad	424
20.17	Algoritmo Simple (Semi IMPLICIT Linked Equations)	425
20.18	Algoritmo Simpler (SIMPLE Revisado)	427
20.19	Algoritmo SIMPLEC (SIMPLE Consistente)	428
20.20	Solución de las Ecuaciones Algebraicas	428
20.21	Disposición de Volúmenes de Control en la Frontera	431
20.22	Condiciones de Contorno para las Variables Dependientes	433
20.23	Criterios para el Desarrollo del Código Computacional	436

CAPÍTULO XXI

APLICACIONES DEL MÉTODO DE VOLÚMENES FINITOS		439
21.1	Conducción de Calor Bidimensional en Régimen Permanente	439
21.2	Flujo Unidimensional en Canales en Régimen Permanente	444

21.3	Flujo Bidimensional en Canales en Régimen Permanente	450
21.4	Aplicación a Flujo Plano en Canales Prismáticos	458
21.5	Flujo Unidimensional en Canales en Régimen Transitorio	462

CAPÍTULO XXII

DIFUSIÓN Y DISPERSIÓN EN CANALES		469
22.1	Introducción	469
22.2	Difusión Molecular	469
22.3	Solución de la Ecuación de Difusión	472
22.4	Difusión Advectiva	474
22.5	Difusión Turbulenta Vertical y Transversal y Dispersión Longitudinal	476
22.6	Solución Numérica de Difusión y Dispersión	488
REFERENCIAS BIBLIOGRÁFICAS		493

PRESENTACIÓN

La Universidad Nacional Agraria La Molina se complace en presentar el libro titulado “*Métodos Numéricos en Recursos Hídricos (Aplicaciones con MATLAB)*”, donde se presentan las bases de la metodología numérica que se emplea en el estudio computacional del flujo de agua en conductos cerrados y abiertos, aplicados a la Ingeniería. Se aborda el desarrollo de esta metodología mostrando cómo se implementan y se resuelven las ecuaciones de conservación (masa, momento y energía) desde los casos más sencillos y simplificados hasta su formulación más general; por lo que se considera de gran importancia para profesionales y estudiantes avanzados de ciencias e ingeniería.

Las leyes de conservación de la masa y de momento en el movimiento de los fluidos son expresadas matemáticamente en la forma diferencial o integral que no admiten soluciones generales analíticas. Por esta razón, las técnicas numéricas en Ingeniería de Recursos Hídricos, han experimentado un gran desarrollo en las últimas décadas y gracias a la evolución de los computadores ha entrado en vigencia una nueva técnica de análisis y el estudio computacional del flujo de fluidos, conocido como la Dinámica de Fluidos Computacional (CFD), generalmente empleando soluciones numéricas como los métodos de diferencias finitas (MDF) o elementos finitos (MEF). La dificultad de aplicación de estos métodos, debido a la no linealidad de las ecuaciones diferenciales, motivaron investigaciones para el desarrollo del método de los volúmenes finitos (MVF), en el cual las ecuaciones aproximadas son obtenidas a través de balances de conservación de la propiedad envuelta en el volumen elemental y con la posibilidad de asociar la interpretación física con la matemática. Por estas consideraciones, se incluye en el libro el fundamento del método de volúmenes finitos y aplicaciones problemas de transferencia de calor y al flujo de agua en canales abiertos.

El propósito del libro es que el lector adquiera los conocimientos y habilidades suficientes para desarrollar sus propios códigos computacionales, proporcionándole una visión integral de las técnicas numéricas de diferencias finitas y volúmenes finitos mediante el uso intensivo de MATLAB, software muy avanzado para el análisis numérico y la visualización gráfica. El libro surge de la experiencia del autor, Dr. Jesús Abel Mejía Marcauzco, de haber enseñado la asignatura de Métodos Numéricos en Ingeniería de Recursos Hídricos, desde el año 2000 en la maestría y desde el año 2005 en el Programa de Doctorado de Recursos Hídricos de la Universidad Nacional Agraria La Molina.

La Universidad Nacional Agraria La Molina, agradece al autor por su aporte, al hacer realidad la publicación de este excelente libro.

Dr. Jorge Luis Aliaga Gutiérrez
Vicerrector Académico
Universidad Nacional Agraria La Molina

CAPÍTULO I

INTRODUCCIÓN

1.1 Definición

Los métodos numéricos son técnicas mediante las cuales es posible formular problemas matemáticos de tal forma que puedan resolverse usando operaciones aritméticas. Aunque hay muchos tipos de métodos numéricos, comparten una característica común: invariablemente se debe realizar un buen número de tediosos cálculos aritméticos. Con el desarrollo de computadoras digitales eficientes y rápidas, el papel de los métodos numéricos en la solución de problemas de ingeniería ha aumentado en forma considerable en los últimos años.

Hoy en día, las computadoras y los métodos numéricos proporcionan una alternativa para cálculos complicados. Al usar la computadora para obtener soluciones directamente, se pueden aproximar los cálculos sin tener que recurrir a suposiciones de simplificación o a técnicas lentas. Aunque las soluciones analíticas aún son muy valiosas, tanto para resolver problemas como para proporcionar una mayor comprensión, los métodos numéricos representan alternativas que aumentan en forma considerable la capacidad para confrontar y resolver los problemas; como resultado, se dispone de más tiempo para aprovechar las habilidades creativas personales. Por consiguiente, es posible dar más importancia a la formulación de un problema, a la interpretación de la solución y a su incorporación al sistema total, o conciencia “holística”.

1.2 Importancia de métodos numéricos en la ingeniería

Desde finales de la década de los cuarenta, la multiplicación y la disponibilidad de las computadoras digitales han llevado a una verdadera explosión en el uso y desarrollo de los métodos numéricos. Al principio, este crecimiento estaba limitado por el costo de las grandes computadoras, por lo que muchos ingenieros seguían usando simples planteamientos analíticos en una buena parte de su trabajo. No es necesario mencionar que la reciente evolución de computadoras personales de bajo costo, ha dado un fácil acceso a mucha gente, a poderosas capacidades de cómputo. Además, existen muchas razones por las cuales se deben estudiar los métodos numéricos:

1. Los métodos numéricos son herramientas muy poderosas para la solución de problemas. Son capaces de manejar sistemas de ecuaciones grandes, no linealidades

y geometrías complicadas, comunes en la práctica de la ingeniería y, a menudo, imposibles de resolver analíticamente. Por lo tanto, aumentan la habilidad de quien los estudia para resolver problemas.

2. Es posible que el lector tenga la ocasión de usar software disponible comercialmente que contenga métodos numéricos. El uso inteligente de estos programas depende del conocimiento de la teoría básica en la que se basan estos métodos.
3. Hay muchos problemas que no pueden plantearse al emplear programas “hechos”. Si está versado en los métodos numéricos y es un adepto de la programación de computadoras, entonces tiene la capacidad de diseñar sus propios programas para resolver los problemas, sin tener que comprar un software costoso.
4. Los métodos numéricos son un vehículo eficiente para aprender a servirse de las computadoras. Es bien sabido que un camino efectivo para aprender programación actualmente es escribir programas de computadora. Porque la mayoría de los métodos numéricos son diseñados para implementarlos en las computadoras, por tanto, son ideales para este propósito. También hay especialistas para ilustrar el poder y las limitaciones de las computadoras. Cuando usted implemente en forma satisfactoria los métodos numéricos en computadora y aplique éstos para resolver de otra manera los problemas difíciles, usted dispondrá de una excelente demostración de cómo las computadoras pueden servir para su desarrollo profesional. Al mismo tiempo, aprenderá a reconocer y controlar los errores de aproximación que son inseparables de los cálculos numéricos a gran escala.
5. Los métodos numéricos son un medio para reforzar su comprensión de las matemáticas, ya que una de sus funciones es convertir las matemáticas superiores a operaciones aritméticas o algebraicas básicas, porque profundizan en los temas que de otro modo resultarían oscuros. Esta alternativa aumenta su capacidad de comprensión y entendimiento en la materia.

1.3 Fundamentos matemáticos

A continuación se presentan, de forma general, algunos de los temas del contenido matemático que se cubre en este libro.

1. **Raíces de ecuaciones:** Estos problemas están relacionados con el valor de una variable o de un parámetro que satisface una ecuación. Son especialmente valiosos en proyectos de ingeniería, donde con frecuencia resulta imposible despejar de manera analítica los parámetros de ecuaciones de diseño.
2. **Sistemas de ecuaciones algebraicas lineales:** En esencia, estos problemas son similares a los de raíces de ecuaciones en el sentido de que están relacionados con valores que satisfacen ecuaciones. Sin embargo, a diferencia de satisfacer una sola ecuación, se busca un conjunto de valores que satisfaga simultáneamente un conjunto de ecuaciones algebraicas. Las ecuaciones lineales simultáneas surgen en el contexto de una variedad de problemas y en todas las disciplinas de la ingeniería. En particular, se originan a partir de modelos matemáticos de sistemas grandes de elementos interrelacionados, como: estructuras, circuitos eléctricos y redes de flujo; aunque también pueden encontrarse en otras áreas de los métodos numéricos como el ajuste de curvas y ecuaciones diferenciales.

3. **Ajuste de curvas:** A menudo se presentará la oportunidad de ajustar curvas en un conjunto de datos representados por puntos. Las técnicas que se han desarrollado para este fin pueden dividirse en dos categorías generales: regresión e interpolación. La primera se emplea cuando hay un grado significativo de error asociado a los datos; con frecuencia los datos experimentales son de esta clase. Para estas situaciones, la estrategia es encontrar una curva que represente la tendencia general de los datos sin necesidad de tocar los puntos individuales. En contraste, la interpolación se maneja cuando el objetivo es determinar valores intermedios entre datos que estén, relativamente libres al error. Tal es el caso de la información tabulada. Para estas situaciones, la estrategia es ajustar una curva directamente mediante los puntos y usar esta curva para predecir valores intermedios.
4. **Integración:** Tal como se representa, una interpretación física de la integración numérica es la determinación del área bajo la curva. La integración tiene muchas aplicaciones en la práctica de la ingeniería, como el cálculo de cantidades totales basadas en conjuntos de medidas discretas. Adicionalmente, las fórmulas de integración numérica juegan un papel importante en la solución de las ecuaciones diferenciales.
5. **Ecuaciones diferenciales ordinarias:** Estas tienen una enorme significación en la práctica de la ingeniería. Esto se debe a que muchas leyes físicas están expresadas en términos de la razón de cambio de una cantidad más que en términos de su magnitud. Entre los ejemplos se observan desde los modelos de predicción hasta la aceleración de un cuerpo en descenso (razón de cambio de la velocidad). Pueden observarse dos tipos de problemas: problemas con valor inicial y valores en la frontera, además de cubrir la computación de valores propios.
6. **Ecuaciones diferenciales parciales:** Las ecuaciones diferenciales parciales son usadas para caracterizar sistemas de ingeniería, donde el comportamiento de la cantidad física es expresada en términos de rapidez de cambio con respecto a dos o más variables independientes. Se incluyen ejemplos de diferentes tipos. Dos diferencias fundamentales de aproximación son empleadas para resolver numéricamente ecuaciones diferenciales parciales. En el presente texto haremos énfasis en el método de las diferencias finitas con aproximación en una forma de puntos discretos. No obstante, también presentaremos una introducción al método de volúmenes finitos y el método de elementos finitos.

1.4 Objetivos del texto

El texto está orientado para preparar a estudiantes de ingeniería y profesionales a aventurarse en los métodos numéricos. En general, permite adquirir una noción fundamental de la importancia de las computadoras y el papel que desempeñan las aproximaciones y los errores en la implementación y el desarrollo de los métodos numéricos.

Del mismo modo permite a adquirir suficiente experiencia en computación y desarrollar sus habilidades para hacer su propio software de los métodos numéricos descritos en este texto. También permitirá desarrollar programas de computadoras bien estructurados y confiables con base en el pseudocódigo, diagramas de flujo u otras formas de algoritmo y deberá tener la capacidad de documentar sus programas para que puedan ser empleados

en forma eficiente por otros usuarios. Finalmente, además de sus propios programas, deberá estar usando los paquetes de software conocidos como MATLAB y Excel.

1.5 lenguajes de programación y software para cálculo numérico

Desde que empezó la era de la computadora, se han elaborado cientos de lenguajes de computación. Determinar cuál es el “mejor” lenguaje ha sido objeto de grandes debates. Por desgracia, muchos programadores tienen casi siempre un apego emocional a “su” lenguaje, e incluso llegan a asegurar que todos los otros son inferiores. Creemos que este tipo de actitudes cerradas pueden ser contraproducentes. Pese a que cada lenguaje tiene sus limitaciones, se puede usar con ventaja en un contexto de problemas particulares. A continuación describimos someramente los principales lenguajes de programación y software, utilizados en métodos numéricos:

Fortran: Establecido como un *traductor de fórmulas*, fue presentado comercialmente por la IBM en 1957. Como su nombre lo indica, una de sus características es que emplea una notación que facilita la escritura de fórmulas matemáticas. Debido a ello, Fortran es un lenguaje natural de computación para muchos ingenieros y científicos.

Por otro lado, para las fórmulas matemáticas, Fortran tiene otras características relevantes para los métodos numéricos. Por ejemplo, tiene capacidad para trabajar con variables de doble precisión y manejar funciones especiales matemáticas, incluyendo variables complejas. Además, es altamente propicio para la programación modular que tiene subrutinas que permiten variables locales y la transferencia de valores entre los subprogramas y el del programa principal. Así, pueden elaborarse bibliotecas numéricas orientadas a los subprogramas y obtenerse comercialmente para usarse en un código Fortran.

C: Posee muchas de las características de un lenguaje de alto nivel y, por lo tanto, puede ser empleado por ingenieros para realizar cálculos numéricos. Aunque en términos generales es inferior al Fortran 90, también tiene acceso a bajos niveles de *hardware* y *software*, como es característico del lenguaje ensamblador. Al mismo tiempo, es más genérico que el lenguaje ensamblador, el cual es usualmente un hardware específico. Esta combinación de capacidad de bajo nivel y portabilidad es lo que hace al C atractivo para quienes elaboran programas.

Aunque los ingenieros y los científicos emplearon primeramente la computación para resolver problemas, una fracción pequeña pero significativa optó por elaborar sus programas. Además, algunas universidades prefirieron enseñar a los estudiantes de ingeniería y ciencias el C como primer lenguaje de computadora. Por ello, se debe reconocer que éste podría usarse, en la próxima década, por un buen número de ingenieros y científicos.

Visual BASIC: El BASIC fue creado originalmente a mediados de la década de los sesenta, como un lenguaje instructivo para los estudiantes. A partir de entonces, se fue sofisticando cada vez más como el QUICK BASIC de Microsoft. En la actualidad, su principal implementación es el Visual BASIC. Este lenguaje está relacionado estrechamente con el sistema operativo de Windows de Microsoft, que es de hecho el ambiente de cómputo para las PC de IBM y equipos compatibles. Asimismo, es una de las

herramientas principales en la elaboración de programas para PC. Además, debido a su función de macrolenguaje para el popular programa de hoja de cálculo Excel, su utilidad está más que reconocida.

Mathcad: Es el producto estrella de MathSoft, Inc. El paquete de software es un puente que une a quienes gustan de las hojas de cálculo como Excel y garabatear en un block, lo que da como resultado una hoja de trabajo interactiva que permite al usuario realizar tareas de matemáticas, obtener datos de primera mano y elaborar gráficas propias de la ciencia y la ingeniería. La información y las ecuaciones son los datos para la hoja de trabajo, es como si se trabajara con papel y lápiz más que con un lenguaje de programación o con notación de hoja de cálculo. La primera versión del Mathcad fue presentada en 1986.

El Mathcad tiene varias funciones y operaciones que permiten llevar a cabo, en forma conveniente, muchos de los métodos numéricos desarrollados en este libro. Tales métodos se describirán en detalle en los siguientes capítulos. En virtud de que es fácil de usar, creemos que el Mathcad es particularmente útil como herramienta pedagógica.

Excel: Es una hoja de cálculo producida por Microsoft, Inc., cuyo tipo especial de software matemático permite el usuario dar y realizar cálculos en renglones y columnas de datos. Por ello, hay una versión computarizada de una hoja de trabajo de grandes cuentas, en la que se pueden realizar y destacar grandes cálculos conectados entre sí. Debido a que se puede actualizar cuando se cambia cualquier número en la hoja, la hoja de cálculo es ideal para tipos de análisis.

Excel tiene algunas capacidades numéricas inherentes, incluida la solución de ecuaciones, el ajuste de curvas y la optimización. También incluye el Visual BASIC como un macro lenguaje que puede usarse para implementar cálculos numéricos. Tiene, asimismo, varias herramientas de visualización (gráficas de superficies en tercera dimensión) que son valiosos complementos para el análisis numérico.

MATLAB: Es un software producido por Mathworks, Inc., empresa establecida por los analistas numéricos Cleve Moler y John N. Little. Como su nombre lo indica, MATLAB fue desarrollado en principio como un *laboratorio de matrices*. Hoy en día el principal elemento del MATLAB siguen siendo las matrices. La manipulación matemática de las matrices está perfectamente realizada en un ambiente interactivo, de uso fácil. Para esas manipulaciones de matrices, MATLAB tiene, además, varias funciones matemáticas, cálculos simbólicos y herramientas de visualización. Como consecuencia, la presente versión representa un ambiente técnico de cálculo.

MATLAB tiene varias funciones y operaciones que permiten realizar, de manera adecuada, muchos de los métodos numéricos empleados en este libro, los cuales serán descritos, en los siguientes capítulos. Además, los programas pueden ser escritos como argumentos y funciones, y los archivos M pueden usarse para realizar cálculos numéricos.

CAPÍTULO II

FUNDAMENTOS DEL MATLAB

2.1 Introducción

MATLAB es un software que posibilita la ejecución del cálculo numérico y simbólico de forma rápida y precisa, acompañado de características gráficas y de visualización avanzadas aptas para el trabajo científico y de ingeniería. MATLAB trabaja en un entorno interactivo y cuenta con más de 500 funciones para el trabajo en distintos campos de la ciencia. Así mismo MATLAB presenta un lenguaje de programación de muy alto nivel basado en vectores, arreglos y matrices.

La arquitectura de MATLAB es abierta y ampliamente extensible, permitiendo la relación con Excel, C, Fortran y otras aplicaciones externas muy utilizadas e importantes. También es posible con el módulo básico de MATLAB el desarrollo de algoritmos a través de un lenguaje propio de programación. MATLAB también permite la operatividad entre plataformas posibilitando trabajar con distintos sistemas operativos y relacionar el trabajo realizado en las distintas plataformas y está en continuo crecimiento y muy adaptable a los avances científicos y al trabajo en laboratorios de simulación numérica.

Materias como la Estadística, el Álgebra lineal, el Análisis matemático, el Análisis numérico, el Análisis de series temporales, las Bases de datos y la Geometría encuentran en el módulo básico de MATLAB y en sus toolboxes que son herramientas adicionales para su desarrollo.

Del mismo modo el MATLAB, a través de Simulink, permite diseñar sistemas dinámicos sencillos o complejos y realizar modelamiento y simulación mediante un lenguaje agradable basado en diagramas de bloques. Admite sistemas de tiempo continuo, sistemas de control y control inteligente, y aplicaciones de procesamiento de señal digital y comunicaciones.

2.2 Componentes del MATLAB

Una faceta muy importante de MATLAB son los toolboxes añadidos, que consisten en paquetes de ampliación al software básico y que son aplicables a determinados campos de la ciencia. A continuación se relacionan los toolboxes más interesantes del MATLAB.

Symbolic Math: Permite integrar la expresión y el cálculo simbólicos (cálculo, transformadas, álgebra lineal, ecuaciones) al entorno de cálculo y visualizaciones de MATLAB.

Extended Symbolic Math: Incluye todas las características de Symbolic Math, proporciona soporte completo para la programación en Maple y permite el acceso completo a las librerías matemáticas de Maple.

Database Toolbox: Permite desde MATLAB consultar e intercambiar datos con las bases de datos ODBC/JDBC más copulares (Oracle, Sybase SQL Server, Sybase SQL Anywhere, Microsoft SQL Server, Microsoft Access, Informix and Ingres) de forma dinámica, preservándolos durante el intercambio y simultáneamente con más de una base de datos.

Excel Link Toolbox: Integra toda la potencia de MATLAB con Microsoft Excel permitiendo la transferencia de datos en los dos sentidos, ejecutando cualquier función de MATLAB desde una fórmula de Excel o utilizando Excel desde MATLAB como editor de vectores.

Statistics Toolbox: Funciones y herramientas interactivas para el análisis de datos históricos, el modelamiento y simulación de sistemas y para el desarrollo de algoritmos estadísticos. Soporta 20 tipos de distribuciones de probabilidad, incorpora el control estadístico de procesos, el diseño de experimentos, estadística descriptiva, etc.

Optimization Toolbox: Proporciona diversos algoritmos y técnicas para solucionar problemas de optimización no lineales, tanto generales como a gran escala.

Spline Toolbox: Mediante un interface gráfico proporciona potentes funciones para el ajuste de datos, visualización, interpolación y extrapolación mediante técnicas spline.

Partial Differential Equation Toolbox: De aplicación en la solución de problemas en muchos campos de la física y la ingeniería: transferencia de calor, flujo en medios porosos, medios conductores, cálculo de esfuerzos y fatigas de estructuras, campos magnéticos, etc. Usa el método FEM (Método de Elementos Finitos que incorpora el algoritmo de triangulación de Delaunay) para solucionar ecuaciones diferenciales parciales.

Neutral Network Toolbox: Proporciona las versiones más comunes de paradigmas y algoritmos para el diseño y simulación de redes neuronales. Incluye bloques de Simulink para poder usar esta toolbox en aplicaciones de control y simulación de sistemas. Incluye ejemplos de control predictivo y control adaptativo. Entre las aplicaciones más comunes de las redes neuronales tenemos las técnicas de clasificación, predicción, filtrado, optimización, reconocimiento de patrones, aproximación a funciones, interpretación y clasificación de imágenes.

2.3 Entorno de Trabajo de MATLAB

Para comenzar en MATLAB, hasta hacer doble clic en el icono de acceso directo al programa situado en el Escritorio de Windows. La pantalla de entrada de MATLAB constituye el marco de trabajo general del programa. Los elementos más importantes de esta pantalla de inicio de MATLAB son los siguientes:

- **Command Window** (*ventana de comandos*): Ejecuta las funciones MATLAB.
- **Command History** (*historial de comandos*): Presenta una historia de todas las funciones introducidas en la ventana de comandos y permite copiarlas y ejecutarlas.

- **Launch Pad:** Ejecuta herramientas y documentación de acceso para todos los productos de Math Works instalados actualmente en el computador.
- **Current Directory (directorio actual):** Muestra archivos MATLAB y ejecuta operaciones de archivos tales como abrir y buscar contenido.
- **Help (ayuda):** Muestra y busca documentación para la familia completa de productos MATLAB
- **Workspace (espacio de trabajo):** Muestra y realiza cambios en el contenido del espacio de trabajo.
- **Array Editor:** Muestra contenido de arrays en formato de tabla y edita sus valores.
- **Editor/Debugger:** Crea, edita y comprueba Archivos-M (archivos que contienen sintaxis de funciones MATLAB). Mayor detalle sobre Archivos-M puede verse en el capítulo de Programación en Matlab.

La **ventana de comandos** es el camino principal para comunicarse con MATLAB. Aparece en el escritorio cuando se inicia MATLAB y se utiliza para ejecutar funciones y todo tipo de operaciones. Las entradas a ejecutar se escriben a continuación del prompt `>>` y, una vez completadas, se pulsa Enter.

Sobre la venta de comandos es posible evaluar los ya ejecutados anteriormente; para ello basta seleccionar la zona de sintaxis a evaluar, hacer clic con el botón derecho del mouse y elegir la opción Evaluate Selection del menú emergente resultante. La opción Open Selection de este mismo menú permite abrir en el Editor/Debugger un Archivo-M previamente seleccionado en la ventana de comandos.

MATLAB es sensible al uso de **mayúsculas** y **minúsculas**, pero permite situar o no espacios en blanco antes y después del signo menos, de los dos puntos y de los paréntesis. MATLAB también permite escribir varias entradas sobre la misma línea, pero separadas por punto y coma. Las entradas se ejecutan todas secuencialmente según están colocadas en la línea, pero sólo se ofrece la salida de la última, siempre y cuando no finalice también en punto y coma. Cualquier entrada que tenga **punto y coma** detrás se ejecuta al pulsar Enter, pero no se ve su salida.

Las entradas largas que no quepan en una línea pueden continuarse en la línea siguiente situando **puntos suspensivos** al final de la línea.

La opción **Clear Command Window** del menú *Edit* permite limpiar la ventana de comandos. El comando **clc** también realiza esta función. Del mismo modo, las opciones **Clear Command History** y **Clear Workspace** del menú *Edit* permiten limpiar la ventana historial y la del espacio de trabajo.

MATLAB dispone de un sistema de **ayuda** en línea bastante eficiente al que se accede tecleando **help** en la ventana de comandos.

Por ejemplo digitando el comando *help*, se obtiene:

```
>> help
```

HELP topics:

```
matlab\general          - General purpose commands.
```

matlab\ops	- Operators and special characters.
matlab\lang	- Programming language constructs.
matlab\elmat	- Elementary matrices and matrix manipulation.
matlab\elfun	- Elementary math functions.
matlab\specfun	- Specialized math functions.
matlab\matfun	- Matrix functions - numerical linear algebra.
matlab\datafun	- Data analysis and Fourier transforms.
matlab\audio	- Audio support.
matlab\polyfun	- Interpolation and polynomials.

.....

Vemos que el comando *help* muestra una lista de los directorios del programa y de su contenido. Si invocamos *help* sin, obtendremos ayuda sobre la función especificada:

```
>> help sin
SIN Sine.
    SIN(X) is the sine of the elements of X.
```

2.4 Variables

2.4.1 Cálculos con una sola variable

MATLAB no requiere ningún tipo de comando para declarar variables. Sencillamente crea la variable mediante asignación directa de su valor. Por ejemplo:

```
>> v=3
v =
    3
```

La variable *v* valdrá 3 mientras no se cambie su valor mediante una nueva asignación. Una vez declarada la variable podemos utilizarla en cálculos:

```
>> v^3
ans =
    27
```

El valor asignado a una variable es permanente, hasta que no se cambie de forma expresa o hasta que no se salga de la presente sesión de MATLAB

Los nombres de las variables comienzan por una letra seguida de cualquier número de letras, dígitos o subrayados, teniendo presente que MATLAB sólo se utiliza los primeros 31 caracteres del nombre de la variable. También es muy importante señalar que MATLAB es sensible a mayúsculas y minúsculas; por lo tanto, una variable con mayúsculas es distinta a la misma variable con minúsculas.

Otro ejemplo sencillo puede ser la evaluación del volumen de la esfera de radio *r*: