DIVISION DE ESTUDIOS SUPERIORES
FACULTAD DE INGENIERIA, UNAM.

CURSOS DE MAESTRIA Y DOCTORADO

La División de Estudios Superiores de la Facultad de Ingeniería, UNAM, ofrece las siguientes Maestrías y Doctorados:

Maestrías		Doctorados
Control	Mecánica	Estructuras
Electrónica	Mecánica de Suelos	Hidráulica
Estructuras	Petrolera	Mecánica de Suelos
Hidráulica	Potencia	Mecánica Teórica y Aplicada
Investigación de Operaciones	Planeación	Investigación de Operaciones
Mecánica teórica y Aplicada	Sanitaria	

Programa de actividades para el segundo semestre de 1976

Exámenes de admisión: 10, 11 y 12 de mayo

Inscripciones: 31 de mayo al 4 de junio

Iniciación de clases: 7 de junio

Requisitos de admisión

a) Cumplir con una de las siguientes condiciones:

1. Poseer título profesional en Ingeniería o en alguna disciplina afín a las maestrías que se ofrecen en la División, otorgado por la UNAM o por cualquier institución nacional o extranjera.

2. Ser pasante de la Facultad de Ingeniería, UNAM

b) Aprobar los exámenes de admisión que se efectuarán en las fechas señaladas arriba.

c) Presentar, dentro del período de inscripciones arriba mencionado, la documentación que se indica en el folleto de Actividades Académicas 1975 de la DESFI

Mayores informes: División de Estudios Superiores de la Facultad de Ingeniería, Apartado Postal 70-256, Ciudad Universitaria, México 20, D. F. Tel.: 548-58-77

"POR MI RAZA HABLARA EL ESPIRITU"
Cd. Universitaria, febrero 3. 1976

EL DIRECTOR DE LA FACULTAD
M. en C. ENRIQUE DEL VALLE CALDERON

EL JEFE DE LA DIVISION
DR. OCTAVIO A. RASCON CHAVEZ

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL

FECHA	DURACION	TEMA	PROFESOR
Marzo 29	16 a 20 h	INTRODUCCION COMPUTADORA DIGITAL	M. en C. Marcial Portilla Robertson
Martes 30	16 a 20 h	LENGUAJE FORTRAN IV Y DIAGRAMA DE FLUJO	M. en C. Marcial Portilla Robertson
Miércoles 31	16 a 20 h	FUNCIONES TRASCEDENTES, FUNCIONES POLINOMIALES Y ALGEBRA MATRICIAL	Ing. Armando Torres Fentanes
Jueves 1° Abril	16 a 20 h	SOLUCION SISTEMAS DE ECUACIONES LINEALES, VECTORES Y VALORES CARACTERISTICOS	Dr. Víctor Gerez Greiser
Viernes 2	16 a 20 h	PROGRAMACION LINEAL	Dr. Víctor Gerez Greiser
Sábado 3	9 a 12 h	RECORRIDO DE INSTALACIONES, EMPLEO DE EQUIPO, PROGRAMAS	Ing. Armando Torres Fentanes
Lunes 5	16 a 20 h	APROXIMACION POLINOMIAL, APROXIMACION FUNCIONAL	Ing. Heriberto Olguín Romo
Martes 6	16 a 20 h	DERIVACION E INTEGRACION NUMERICA	Ing. Heriberto Olguín Romo
Miércoles 7	16 a 20 h	ERRORES Y SOLUCION DE SISTEMAS DE ECUACIONES NO LINEALES	M. en C. Marcial Portilla R.
Jueves 8	16 a 20 h	SOLUCION DE ECUACIONES DIFERENCIALES, METODOS ESTADISTICOS	Ing. Armando Torres Fentanes
Viernes 9	16 a 20 h	OPTIMIZACION DE FUNCIONES	Ing. Armando Torres Fentanes
Sábado 10	9 a 12 h	EMPLEO DE PROGRAMAS	Ing. Armando Torres Fentanes

'edcs.

DIRECTORIO DE PROFESORES
METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL

1. DR. VICTOR GEREZ GREYSER

   Jefe del Departamento de Ingeniería
   Mecánica y Eléctrica
   Facultad de Ingeniería, UNAM
   Ciudad Universitaria
   México 20, D. F.
   Tel.: 48-99-58 y 550-00-40

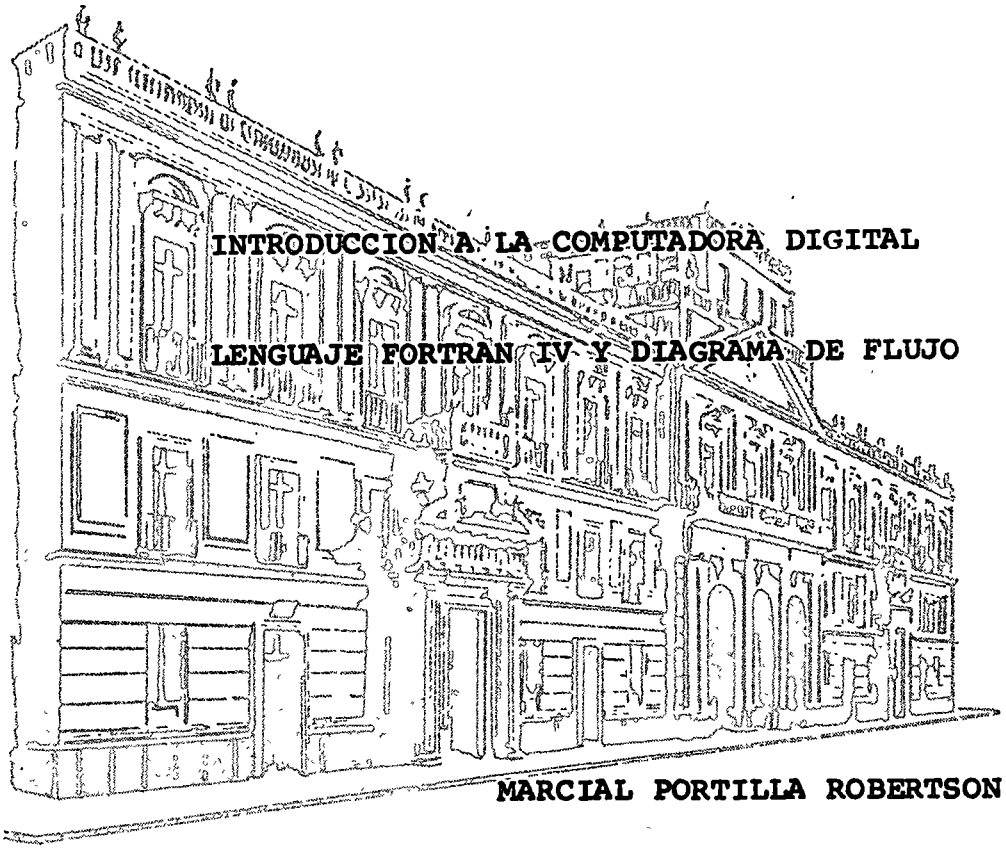
2. ING. HERIBERTO OLGUIN
   Jefe del Centro de Cálculo
   Facultad de Ingeniería, UNAM
   Ciudad Universitaria
   México 20, D. F.
   Tel.: 548-99- 58 y 548-65-00 ext. 261

3. ING. MARCIAL PORTILLA ROBERTSON
   Secretario del Departamento de
   Ingeniería Mecánica y Eléctrica
   Facultad de Ingeniería, UNAM
   Ciudad Universitaria
   México 20, D. F.
   Tel.: 548-99-58 y 550-00-40

4. ING. JOSE ARMANDO TORRES FENTANES
   Asesor de Matemáticas III
   Facultad de Ingeniería UNAM
   Ciudad Universitaria
   México 20, D. F.
   Tel.: 548-99-58 y 550-00-40

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL



INTRODUCCION A LA COMPUTADORA DIGITAL

LENGUAJE FORTRAN IV Y DIAGRAMA DE FLUJO

MARCIAL PORTILLA ROBERTSON

MARZO DE 1976.

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels: 521-40-23 521-73-35 5123-123

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

FORTRAN

INTRODUCCION.- Desde el inicio de las computadoras digitales, uno de los campos de más aplicación de estos dispositivos ha sido el científico, donde existen necesidades de cálculos y operaciones muy complicadas o largas, siendo la computadora un auxiliar poderoso en la solución de estos problemas.

Inicialmente, la programación de las computadoras se hacía a nivel de lenguaje de máquina, ésto es, instrucciones numéricas que obligan a la máquina a ejecutar una operación sencilla (sumar, restar, etc.). Esta manera de programar a la computadora exigía al usuario un profundo conocimiento de las características del equipo usado, tanto de hardware como de operación.

Por la dificultad que entrañaba el usar una computadora para un uso científico, se pensó en simplificar la programación mediante un lenguaje más parecido al lenguaje matemático, por lo tanto más sencillo de aprender, y que pudiera ser compatible con distintas marcas de computadoras. Uno de los primeros lenguajes de alto nivel (o sea que para ser ejecutados tienen que ser traducidos mediante un compilador) fue el Fortran (Fórmula Translation) que es un lenguaje para uso eminentemente científico, con instrucciones muy fáciles de entender y con compiladores en casi todas las computadoras del mercado.

Estas notas estarán orientadas a él "cómo resolver el problema utilizando la computadora". Pues durante el resto del curso la computadora será una herramienta para el dise

ño de mecanismos.

La estructura del problema tiene 4 pasos a seguir, los cuales son:

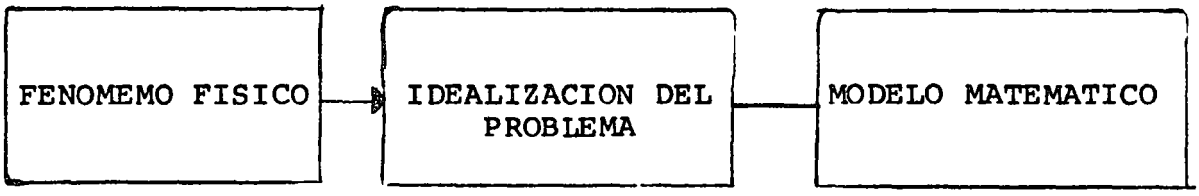
1.- La formulación precisa del problema

2.- Modelo matemático

3.- Análisis matemático

4.- Solución del problema con computadora

Resulta importante poder pasar del fenómeno físico al modelo matemático, esta relación se muestra en la siguiente figura:

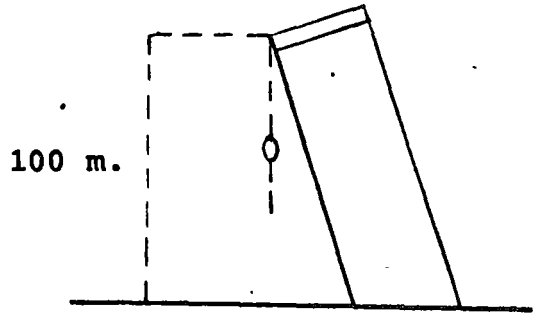


Veamos como sucede ésto en la solución de un problema científico. Por ejemplo veamos el caso de la Ley de Movimiento de Newton, la cual dice que la fuerza es igual a la masa por la aceleración:

$$f = ma$$

Supongamos que esta relación es exacta, sin tomar en cuenta la teoría de la relatividad.

Este típico problema se estudió en un curso elemental de Física conocido como el problema de la piedra que cae:



LA TORRE DE PISA

Queremos preguntarnos cuánto tarda la piedra en caer.

Si analizamos la situación vemos que la distancia d que viaja en un tiempo t con aceleración constante es:

$$d = \frac{a\,t^2}{2}$$

$$\text{si } a = 9.8 \text{ m/seg.}^2$$

$$t = \frac{9.8 \times (100)}{2}^{1/2}$$

Bien ahora preguntémonos qué tan realista fue nuestra respuesta. Notemos que NO consideramos efectos tales como:

1.- La variación de la dirección de la gravedad
2.- Variación de la gravedad dependiente de la - altura sobre el nivel del mar.
3.- Resistencia del aire (sobre la piedra)
    a) Forma de la piedra
    b) Velocidad de la piedra
    c) Densidad del aire (varía con la altitud y la temperatura).
    d) Densidad de la piedra.
4.- Atracción gravitacional entre sol y luna
5.- Vientos y corrientes de aire, etc.

Es posible incluir todos estos factores en nuestro modelo matemático, analizar estas ecuaciones y mejorar

el tiempo de caída real (pregunta original). Creo yo que hemos llevado el problema a un extremo, como conclusión podemos decir que el modelo debe ser lo suficientemente exacto para obtener de este resultados útiles, -- sin caer en el extremo anterior, donde el precio que -- hay que pagar en el análisis matemático y esfuerzos de computación, tal vez no sea lo que queremos.

Una vez que se ha hecho el diseño matemático - del modelo del problema, es necesario determinar un "algoritmo" para resolverlo, ésto es, una serie finita de pasos que nos lleve a la solución del problema.

Para el problema anterior, el algoritmo sería - el siguiente:

1.- Leer el valor de la altura (d)

2.- Hacer a = 9.8 $m/seg.^2$

3.- Hacer $t = \frac{9.8 \times d^{1/2}}{2}$

4.- Escribir el valor de t como respuesta.

Una de las técnicas más populares para describir algoritmos es por medio de diagramas de flujo, los - cuales se explicarán a continuación.

DIAGRAMAS DE FLUJO

Los diagramas de flujo son representaciones grá

ficas de los programas. Cada decisión y operación a desarrollar será colocada en una caja, la forma de la caja nos indicará el tipo de instrucción a desarrollar. Se utilizarán flechas para interconectar estas cajas, las flechas nos indicarán la secuencia de las operaciones, usualmente debemos empezar por la parte superior y bajar siguiendo las flechas (top down).

El análisis del problema se facilita utilizando diagramas de flujo, pues no son ambiguos y tienen una estructura similar a la del problema. Desafortunadamente, no existe hoy en día una convención estandard para estas representaciones, a lo largo de estas notas se usará la notación IBM, que resulta ser la más general aunque no es universal.

Las siguientes reglas serán usadas.

1.- Cada proposición será colocada en una caja. (Es válido colocar varias proposiciones en la misma caja).

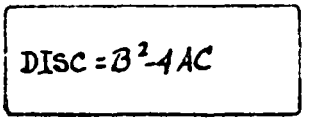
2.- La secuencia de las operaciones se indica con segmentos de línea dirigidos (flechas) entre las cajas.

3.- Se utilizan diferentes tipos de cajas según sea la proposición.

DESCRIPCION DE LOS SIMBOLOS

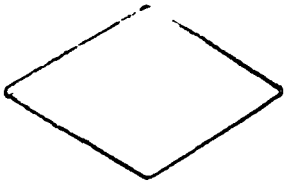
Indica inicio del programa o de un subprograma.

Operación a ejecutar por ejemplo de la figura 1.



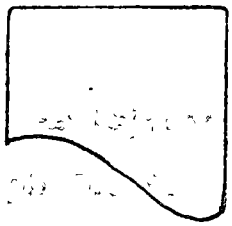
Indica que se debe de hacer la - operación:

B^2 - 4 AC y su valor asignar lo a DISC.

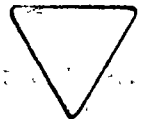


Proposición IF (# pag 7) comparación, compara lo que está dentro de la caja dependiendo de - esta comparación se podrán seguir 3 caminos. La comparación se indica con: si queremos comparar I con N, lo indicaremos (dentro de la caja) por I : N.

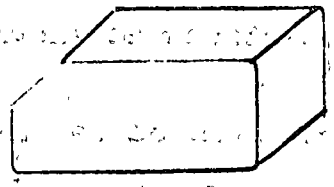
Indica que hay que leer tarjetas perforadas con datos o proposiciones.



Indica que queremos imprimir en la impresora algún mensaje o resultado.



Este símbolo es un conector



Paquete de tarjetas (usualmente un programa).

Operación DO (la cual se explicara en la siguiente sección.

Fin o alto del programa.

Cinta magnética o cinta de papel perforado.

Primer problema de clase.

Este problema trata de determinar la precipitación pluvial total y su promedio en el Distrito Federal durante un lapso, digamos un año.

Los datos con los que se cuenta son las lecturas diarias efectuadas en milímetros, (notemos que no es posible tener números negativos) o sea la cantidad de lluvia.

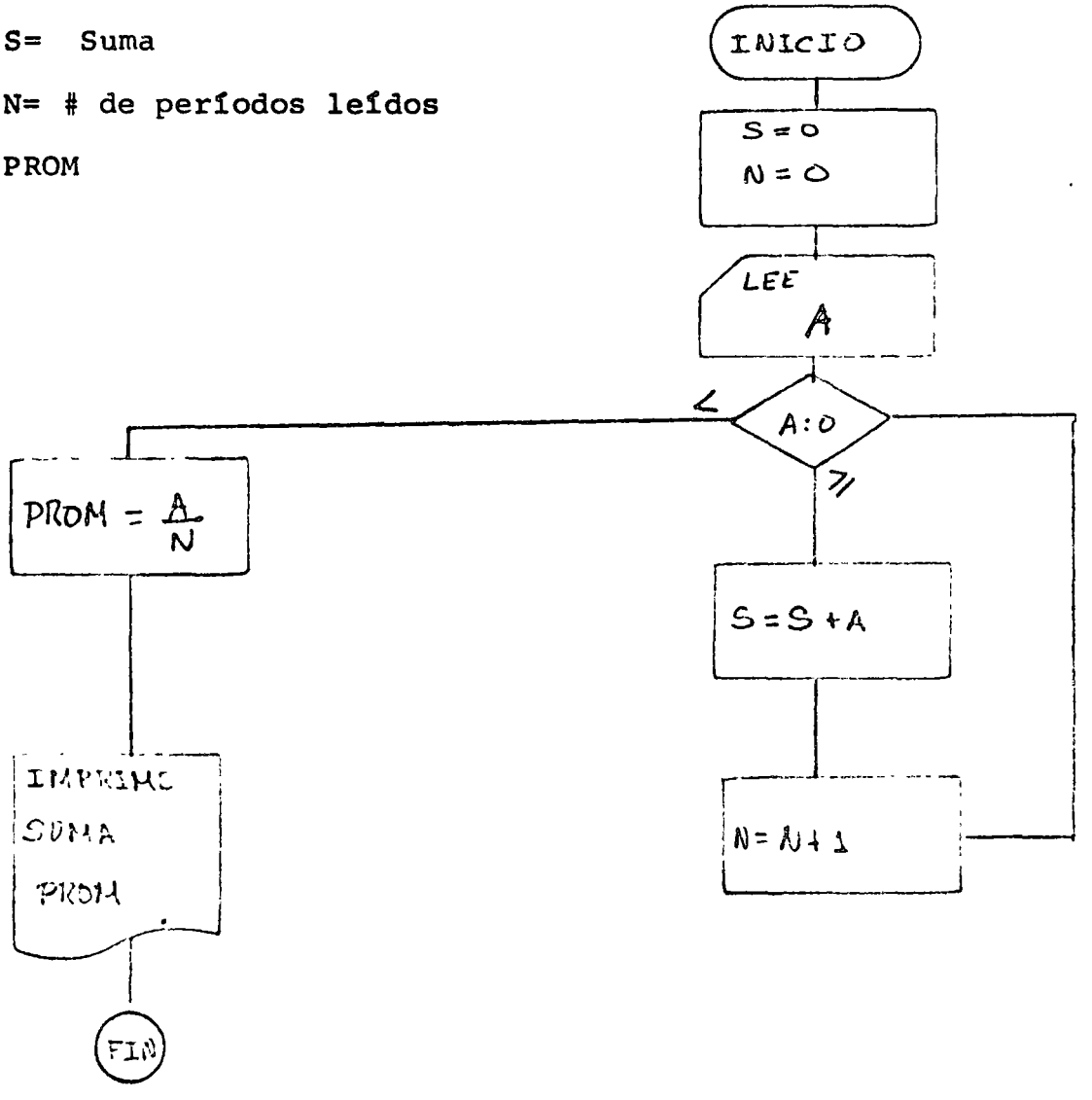
Procederemos a resolver el problema de la siguiente forma, (utilizando primero un diagrama de flujo).

A= cantidad de lluvia (agua)

S= Suma

N= # de períodos leídos

PROM



A continuación procederemos a explicar el diagrama de flujo.

La primera caja de inicio nos indica el comienzo de nuestro programa. La siguiente caja asigna el valor cero a las variables A y N donde A es el valor (cantidad de lluvia) leído, y N va a ser el número de lecturas tomadas. A continuación procedemos a leer el primer dato, notemos que no es posible tener, valor negativo de lluvia, por lo que valiéndonos de esta propiedad en la siguiente caja preguntamos si el valor leído es negativo, si es negativo calculamos el promedio de lluvia en el Distrito Federal, e imprimimos la suma y el promedio, para poder hacer esta operación en la última tarjeta de datos se pondrá un número negativo. Si el valor leído no es cero, se procede a sumar, lo cual s efectúa haciendo la asignación de la suma del valor anterior de suma más el valor leído, cosa similar ocurre en el "contador" del número de datos leídos. Notemos que este ciclo se repite hasta que leamos un número negativo, al suceder ésto, el programa imprime el resultado y termina.

CODIFICACION FORTRAN DEL PROGRAMA

```
XXX     TARJETAS DE CONTROL
C       INICIO DE LAS VARIABLES
        SUMA  =  0
        N  =  0
   1    READ ( 2, 2)  A
   2    FORMAT  (15)
        IF  (A)  3, 3, 4
   4    SUMA = SUMA + A
        N = N + 1
        GO TO 1
   3    PROM = SUMA/N
        WRITE ( 3,5 ) PROM, SUMA
   5    FORMAT ( 10X, F10.4, 5X, I5 )
        CALL EXIT
        END.
```

DESCRIPCION.

Como todo lenguaje (ya sea de programación o natural) el Fortran tiene un alfabeto, ésto es, una serie de símbolos que sirven para formar expresiones e instrucciones. El alfabeto de Fortran para B - 6500 consta de:

Letras: A, B, C. D, . . . , X, Y, Z.

Dígitos: 0, 1, 2, 3, . . . , 9 .

CARACTERES ESPECIALES.

+	MAS
-	MENOS
*	ASTERISCO (MULTIPLICA)
/	DIAGONAL (SLASH) DIVIDE
=	ASIGNA (NO CONFUNDIR CON IGUALDAD)
,	COMA (USADA COMO SEPARADOR)
(	ABRE PARENTESIS
)	CIERRA PARENTESIS
⌀	ESPACIO EN BLANCO
"	COMILLAS (UTILIZADA EN FORMATOS)
**	DOS ASTERISCOS (ELEVA AL CUADRADO)

Todo programa en Fortran contiene instrucciones de los s iguientes tipos:

a) Asignación

b) Control

c) Entrada/Salida

d) Información para el compilador

e) Funciones y subprogramas

La forma de codificar (escribir) un programa en Fortran es la siguiente.

Cada tarjeta contiene 80 columnas que deben distribuirse de la siguiente manera:

COLUMNAS	USO
1 - 5	Número de proposición (etiqueta)
6	Continuación
7 - 72	Proposición
73 - 80	Identificación o número de secuencia

Un programa completo en Fortran se vería codificado como sigue: (PROGRAMA MOSTRADO EN DIAGRAMA DE FLUJO ANTERIORMENTE).

```
    5   6   7                                   72      73      80
|       |   |                                   |       |       |
|       |   READ 10, A, B, C                    |       |       |
|   10  |   FORMAT ( 3 F10.0 )                  |       |       |
|       |   IF (A(  20, 50, 20                  |       |       |
|   20  |   DISC = B **2 - 4*A*C                |       |       |
|       |   IF (DISC) 40, 25, 25                |       |       |
|   25  |   DISC = SORT (DISC)                  |       |       |
|       |   X1 = (-B + DISC)  / (2*A)           |       |       |
|       |   X2 = (-B - DISC)  / (2*A)           |       |       |
|       |   PRINT 30, X1, X2                    |       |       |
|   30  |   FORMAT (1H, 2 F10. 3 )              |       |       |
|       |   GO TO 50                            |       |       |
|   40  |   PRINT 45                            |       |       |
|   45  |   FORMAT ("DISCRIMINANTE NEGATIVO")   |       |       |
|   50  |   CALL EXIT                           |       |       |
|       |   END                                 |       |       |
```

CONSTANTES.

Una constante en Fortran puede ser de 2 tipos:

a) Entera (Integer)

b) Real (Real)

a) Una constante entera es cualquier número - sin punto decimal. Ejemplo:

0, 91, - 173, + 327

si un entero se escribe sin signo, se supone positivo. Los valores que pueden tomar las constantes en IBM-1130 son los comprendidos en el rango:

$$-32767 \quad a \quad 32767$$

$$(-(2^{15}-1) \quad a \quad 2^{15}-1)$$

No se permite introducir comas en una constante entera. Ejemplo de constantes enteras ilegales:

3.2	tiene punto decimal
27.	
31459036	demasiado grande
5,496	contiene una coma

b) Una constante real es cualquier número con - punto decimal. Ejemplo:

0.

91.3

-145.8

5.E3

Que escribiremos como:

5.0×10^3	5. E03
$-5. \times 10^3$	-5. E03
4.1×10^0	4. E00

La maginitud de una constante real n o debe ser mayor que 2^{127} ó menor que 2^{-128}

VARIABLES. Una variable en Fortran es la representación simbólica de una cantidad que puede tomar diferentes valores.

Por ejemplo, en la instrucción

A = 5.0 + B

A y B son variables, el valor de B está determinado por alguna instrucción previa y puede cambiar. El valor de A está variando para cada nuevo valor de B.

NOMBRES DE VARIABLES.- Un nombre de una variable consiste - en una cadena de 1 a 5 caracteres alfanuméricos, excluyendo caracteres especiales, y siendo el primero una letra.

Ejemplo: de nombres de variables permisibles.

DET

AB1

I

LL4E

Ejemplo: de nombres de variables ilegales:

1LL4 Empieza con caracter no alfabetico
ABCDEFGHIJ Demasiado grande
A-B Caracter ilegal
A/B Caracter ilegal

El tipo de cantidad (real o entera) que representa se puede especificar de dos maneras: explícita e implícitamente.

La forma implícita de especificar una variable es como sigue:

a) si la primera letra del nombre de la variable es: I, J, K, L, M, N, la variable se considera como una variable entera.

b) Si la primera letra del nombre de la variable NO es: I, J, K, L, M, N, la variable se considera como una variable real.

La forma explícita de especificar una variable es usando una Declaración de tipo, la cual hace que el compilador ignore la especificación implícita, por ejemplo: si por medio de una declaración de tipo, designamos a la variable ITEM como de tipo real, será manejada por el compilador como una variable real, sin importar que su primera letra sea una I.

EXPRESIONES ARITMETICAS

Una expresión aritmética (e.a.) es una sucesión de constantes, variables y símbolos de operaciones aritméticas que siguen las reglas que a continuación se darán.

Los siguientes son los símbolos de operación aritmética:

Símbolo u operador:	Significado:
+	SUMA
-	RESTA
*	MULTIPLICACION
/	DIVISION
**	EXPONENCIACION

Ejemplo:

Expresión algebraíca	Equivalente en Fortran
$a + b$	A + B
$a - b$	A _ B
ab	A * B
$\frac{a}{b}$	A/B
a^b	A**B

REGLAS.

1.- Dos operadores no deben estar juntos. Deben estar separados por cantidades o paréntesis en la expresión por ejemplo; A +¯ B es inválida, mientras que ¯B + A ó A + (¯B) son válidas.

2.- No se pueden omitir operadores, por ejemplo:

3A NO significa 3*A.

MODOS COMPUTACIONALES.

Los cálculos aritméticos son hechos en dos formas: entera y real, dependiendo del tipo de las cantidades envueltas en el cálculo. Las constantes o variables que forman una expresión aritmética no necesitan ser del mismo tipo.

El modo de la expresión es entero, real o mixto, dependiendo de si las constantes son enteras, reales o estan -- mezcladas.

Por ejemplo:

EXPRESION	TIPO DE CANTIDAD	MODO DE LA EXPRESION
3	Constante entera	entera
I + J	Variables enteras	entero
3.0	Constante real	real
A	Variable real	real
5*JOB + ITEM	Variables enteras, Constante entera	entera
A**B	Variables reales	real
A + B/ITEM	Variables reales	mixto (el resultado se guarda como real).

Se pueden usar paréntesis en las expresiones aritméticas como en algebra, para especificar el orden en el cual se van a efectuar las operaciones aritméticas que forman la expresión.

PROPOSICION DE ASIGNACION

La forma general de esta proposición es:

(variable) = (expresión aritmética)

Ejemplo:

```
A = 5
AB2 = A* ( B ** 2 )
DISCR = B ** 2 - 4* A* C
```

El objeto de esta instrucción es el de asignar a la (variable) el valor de la (expresión aritmética), borrando el valor anterior de dicha variable.

PROPOSICIONES DE ENTRADA/SALIDA

Las proposiciones de entrada/salida permiten al programador introducir (obtener) datos al (del) programa.

La forma general de dichas instrucciones es:

READ (2 , n_f) (lista de variables)

WRITE (3 , n_f) (lista de variables

n_1 Es el número de la unidad (física) de lectura de datos: Lectora de tarjetas perforadas, lectora de cinta perforada, cinta magnética, etc. En IBM-1130 este número es el 2 para lectora de tarjetas.

n_f Es el número de una proposición de formato. Para que el programa pueda leer datos, debe tener conocimiento de la forma en que se le van a presentar. Esto se hace mediante la proposición FORMAT.

LA FORMA GENERAL DE ESTA POSICION ES

n_f FORMAT (lista de especificaciones)

La lista de Especificaciones le indica al compilador en qué forma están perforados los datos. Básicamente hay dos tipos de especificaciones

I entera

F Flotante (cant. reales)

E Exponencial

El formato I tiene la forma

Iw donde w es el ancho del campo

Este formato se utiliza para leer (o escribir) valores de variables enteros, con un ancho máximo de w dígitos.

Por ejemplo, si queremos leer un valor de la variable L de tarjeta, tendremos que escribir:

```
    READ   ( 2, 10 ) L
10 FORMAT (I4)
```

Las dos proposiciones anteriores hacen que la computadora lea un valor entero de a lo más 4 dígitos, y lo asigne a la variable L.

Para leer (o escribir) valores que corresponden a -- cantidades reales, se usa el formato F el cual tiene la forma general Fw.d, donde w es el número máximo de dígitos y d es el número de dígitos decimales. (d puede ser 0).

Por ejemplo, para leer el valor de la variable R podemos usar:

```
READ  (2,3) R
3 FORMAT (F10.3)
```

Esto le indica al compilador que va a leer un valor real de 9 digitos, de los cuales 3 son decimales.

Para la impresión de resultados, el número asignado a la impresora en línea en IBM-1130 es 3.

Existe un formato que permite mejorar la impresión de resultados, este es el fomato X, el cual hace que la impresora (y en algunos casos la lectora) se "salte" tantos espacios como lo indique la especificación, por ejemplo, la especificación 4x hará que el programa, en impresión, deje 4 espacios en blanco, libres.

A continuación veremos las conversiones en entrada-salida, de distintos valores bajo diferentes especificaciones.

(ø significa espacio en blanco)

ENTRADA

Campo de Entrada	Especificación	Valor Interno
567	I3	+ 567
- 329	I6	- 329
- 27	I7	- 27
27	I5	+ 27000
234	I7	234

SALIDA

Valor Interno	Especificación	Campo de Salida
+ 23	I4	+ 23

Valor Interno	Especificación	Campo de Salida
- 79	I 4	- 79
+ 30145	I 5	30145
- 30145	I 5	*****
+ 978	I 1	*
0	I 3	0

ENTRADA

Campo de Entrada	Especificación	Valor Interno
36725931	F8.4	+ 3672.5931
3.672593	F8.4	+ 3.672593
- 367259.	F8.4	- 367259
367259	F6.6	+ 0.367259

SALIDA

Valor Interno	Especificación	Campo de Salida
+ 36.7929	F7.3	36.763
+ 36.7934	F9.3	36.793
- 0.0316	F6.3	- 0.032
+ 579.645	F4.2	***
+ 579.645	F6.2	579.65
-579.645	F6.2	*****

En algunos de los casos, es necesario imprimir títulos o encabezados de tal manera de que los resultados del pro-- grama sean más legibles y comprensibles. Para este tipo de problemas es común poner el texto a ser impreso entre comi- las, dentro del formato que le corresponda.

Por ejemplo, si queremos imprimir los valores de tres variables A, B, C, en una forma legible, podríamos usar el siguiente formato:

```
WRITE  ( 3, 10 ) A, B, C,
10 FORMAT ('A = ', F10.4, B = ', F10.4,'C =  'F10.4)
```

Lo cual provoca que la máquina imprima lo siguiente:
(suponiendo que A = 5.6, B = - 4.85, C = 1.492)

A = _ _ _ _ 5.6000 _ B = ___ - 4.8500 C = _ _ _ _ 1. 4920

Es importante hacer notar que la primera posición de impresión (i-e 1a columna 1) sirve para el control del ca rro de impresión, por lo que es necesario tener presente este hecho siempre que se vaya a imprimir.

El control de carro posible se reduce a las siguien- tes:

PRIMERA COLUMNA

⊘	Espaciamiento sencillo antes de imprimir
0	Espaciamiento doble antes de imprimir
1	Salto a otra página antes de imprimir
+	Sobre - escritura antes de imprimir

Proposiciones de Control

Normalmente, las instrucciones de un programa en Fortran

se van ejecutando en forma secuencial, por lo que es necesario alterar (en algunos problemas) esta forma de ejecución.

Las instrucciones que alteran el flujo de las instrucciones en un programa son las instrucciones de control que son:

GO TO

IF Aritmético

IF Lógico

STOP

CONTINUE

Proposición GO TO

Esta proposición es llamada de transferencia incondicional y su forma general es:

GO TO n donde n es un número de proposición

El efecto de esta proposición es el de transferir el flujo de las instrucciones a aquella que tiene el número de proposición n. Esto se puede visualizar como un "salto" en el orden de ejecución del programa.

Por ejemplo:

En el segmento de programa siguiente, el orden de ejecución es: 1,2,3,5,6,2,3,5,6,7

```
1  A = 0                    4  C = A * B
2  B = A + 3                5  WRITE (3,8), A,B,C
3  GO TO  5                 6  GO TO  2
                            7  END
```

Proposición IF

Generalmente, en el transcurso de un programa, es necesario hacer "preguntas" sobre algunos valores de las variables del programa, y tomar "decisiones" según el resultado de la pregunta.

Esta "toma de decisiones se efectúa por medio de la proposición IF, la cual tiene las versiones formadas Aritmética Lógica.

El IF Aritmético tiene la forma

IF (exp. aritmética) n_1, n_2, n_3

Donde n_1, n_2 y n_3 son 3 etiquetas ó números de proposición.

El funcionamiento de este IF es el siguiente:

1.- Evalúa la expresión aritmética

2.- Compara el resultado con cero y toma cualquiera de las siguientes acciones:

   a) Si el resultado es 0, transfiere el control a la proposición con el número n_1

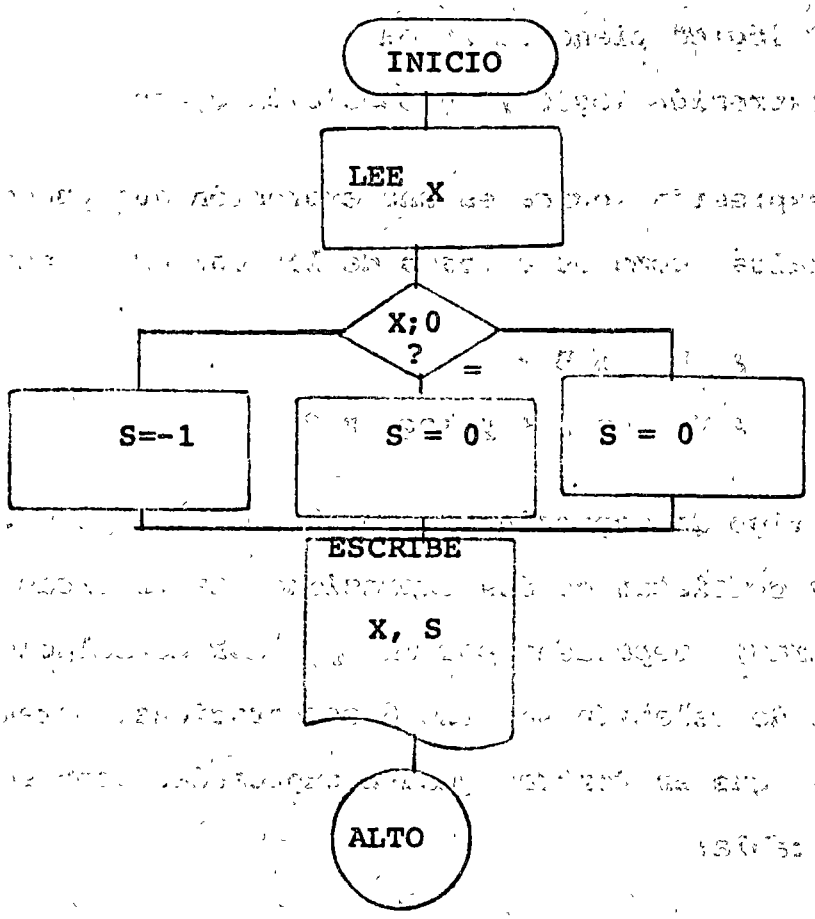
   b) Si el resultado es = 0, transfiere el control a la proposición con el número n_2

   c) Si el resultado es 0, transfiere el control a la proposición con el número n_3

Como ejemplo, veamos un programa que calcula la función SIGNUM (x), ((x)) definida como:

$$\sigma(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases}$$

Hagamos primero un diagrama de Flujo:



La codificación en Fortran IB-1130 queda como sigue:

```
1     READ (2,2 ) X
2     FORMAT ( F10.0 )
3     IF (X) 4,6,8
4     S = - 1
5     GO TO 9
6     S = 0
```

```
7    GO TO 9
8    S = 1
9    WRITE  (3,10) X,S
10   FORMAT  (F10.6, 3)
     END
```

El IF lógico tiene la forma

IF (expresión lógica) proposición ejecutable

Una expresión lógica es una expresión que puede ser cierta o falsa, como es el caso de las comparaciones:

¿ Es A B ?

¿ Es sen x + y log. z ?

Este tipo de expresionaes son llamadas expresiones de relación y consisten de dos expresiones aritméticas (reales o enteras) separadas por un operador de relación. Los operadores de relación son las 6 comparaciones matemáticas: = ≠ = = que en Fortran quedan expresadas como se muestra en la tabla:

OPERADOR DE RELACION	NOMBRE EN FORTRAN
=	.EQ.
≠	.NE.
	.LT.
=	.LE.
	.GT.
=	.GE.

Además de estos operadores, se usan tres operadores lógicos para construir expresiones más elaboradas. Estos operadores son llamados disyunción, conjunción y negación; sus nombres en Fortran son .OR., .AND., NOT. Respectivamente. Su funcionamiento es el siguiente:

Si X y Y son expresiones lógicas, entonces X.OR. Y es cierta. a menos que X o Y sean ambas falsas.

X. AND. Y es falsa, a menos que X y Y sean ambas ciertas.

NOT.X es falsa si X es cierta y viceversa

Veamos el siguiente ejemplo comparativo del uso del IF Lógico y del IF aritmético.

Supongamos que queremos hacer R = R1 solamente si S = 3 y T = 4, hay 3 maneras de hacer esto:

IF ARITMETICO	IF LOGICO
IF (S - 3) 1,2,1	a) b)
2 IF (T-4) 1,3,1	IF(S.NE.3) GO TO 1 IF(S.EQ.3.AND.
	IF (T.EQ.4) R = R1 T. EQ.4) R=R1
1 _ _ _ _	

Vemos que usando el IF aritmético necesitamos 3 etiquetas. En el IF Lógico, el funcionamiento es el siguiente:

a) Se evalúa (n) la (s) expresión (es) aritmética (s) involucrada (s) en la expresión lógica.

b) Se va evaluando el valor lógico (cierto o falso) de la expresión lógica, siguiendo un orden de prioridad de operadores lógicos. El orden es el siguiente: La más alta prioridad es dada. NOT. Después se evalúa .AND.
y finalmente .OR.

c) Una vez que se ha evaluado la expresión lógica entre los paréntesis del IF, se observa su valor y ocurre una de dos situaciones:

Si la expresión lógica es cierta, se procede a ejecutar la proposición a la derecha del If;

Si la expresión lógica es falsa, se ignora dicha proposición y se contínua con la secuencia normal del programa.

ARREGLOS.- Generalmente, en los problemas científicos, es necesario usar vectores, matrices o estructuras con más dimensiones.

Debido a que todas las variables usadas en el programa ocupan un lugar en la memoria de la computadora, los problemas deben ser "declarados" (esto es, se le debe dar información al compilador) a fin de que se les asignen localidades en la Memoria. La manera de declarar un arreglo en Fortran es por medio de la proposición "dimension" y es como sigue:

DIMENSION A (n_1, n_2), B (n_3), C(n_4, n_5, n_6)
Donde A, B, C son los nombres de los arreglos y n_1, n_2, ...n_6 son las dimensiones máximas de dichos arreglos.

Por Ejemplo:

Dimension NOM (10,10)

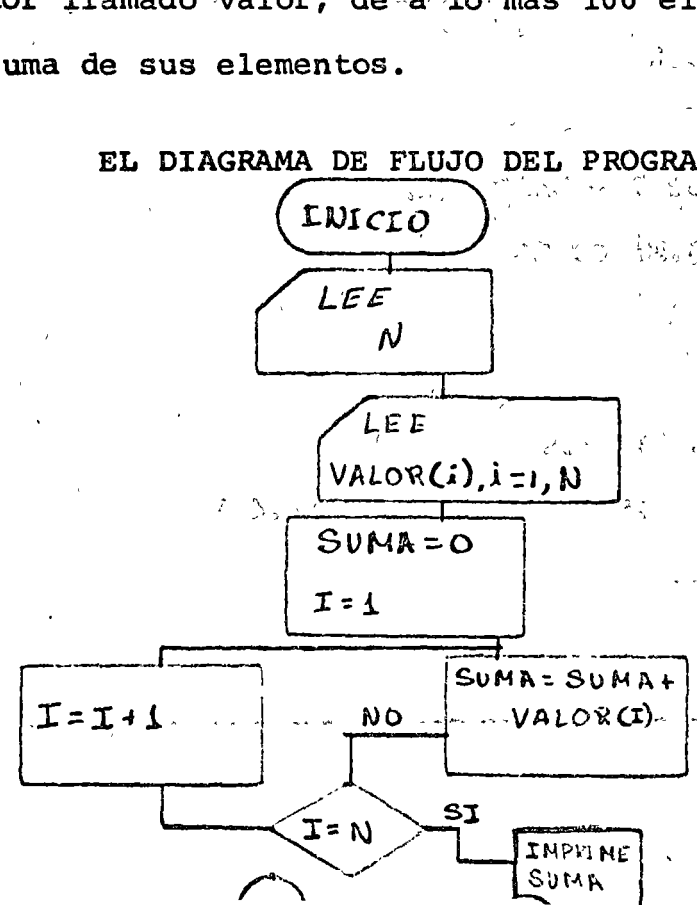
Hará que el compilador reserve 100 localidades para una matriz llamada NOM con 10 renglones y 10 columnas.

Para referirse a algún elemento de un arreglo, basta con poner el nombre del arreglo, seguido por el índice (o los índices) encerrado entre paréntesis por ejemplo, si queremos sumar los dos primeros elementos del arreglo A y guardar el resultado en el tercero, lo haríamos por medio de la siguiente proposición.

A (3) = A (1) + A (2)

En general, los índices de un arreglo son variables enteras, que van cambiando de valores en el transcurso del programa. Como ejemplo, veamos un programa que lee un vector llamado valor, de a lo más 100 elementos; y calcula la suma de sus elementos.

EL DIAGRAMA DE FLUJO DEL PROGRAMA ES:



La codificación del programa en Fortran quedaría como sigue:


```
C       SE DECLARA EL ARREGLØ
        DIMENSION VALOR (100)
C       SE LEE EL NUMERO REAL DE ELEMENTOS A PROCESAR
        READ  (2,1) N
      1 FØRMAT (13)
C       LA SIGUIENTE PROPOSICION LEE LOS ELEMENTOS DEL
        ARREGLO  MEDIANTE UNA PROPOSICION DE ITERACION
        QUE SE EXPLICARA CON MAS DETALLE
        READ (2,2) ( VALOR (I), I - 1,N )
      2 FORMAT ( 8F 10.0 )
C       DE AQUI EN ADELANTE SON LOS CALCULOS
        SUMA = 0.0
        I = 1
      3 SUMA = SUMA + VALOR (I)
        IF (I.EQ.N) GO TO 4
        I = I + 1
        GO TO 3
      4 WRITE (3,5) SUMA
      5 FORMAT  ( 1H, 'SUMA = ' , F12.4 )
        CALL EXIT
        END
```

En el programa usamos una proposición de la forma

(VALOR (I), I = 1, N)

La cual es usada con mucha frecuencia para lectura e impresión de arreglos (de cualquier dimensión) y su - funcionamiento es equivalente a escribir.

(VALOR (1), VALOR (2), VALOR (3), . . ., VALOR (N)

Este tipo de proposiciones de iteracción son las que quizá dan más poder al lenguaje, ya que es posible realizar grandes cantidades de cálculos mediante pocas proposiciones.

Una de las proposiciones de interación más usadas por los programadores en Fortran en la proposición DO.

PROPOSICION DO.

La proposición de control DO nos permite efectuar una serie de iteraciones con una sola proposición, por ejemplo: si queremos inicializar un arreglo 4 de N elementos a ceros, se puede hacer usando If's o usando una proposición Do. Veamos las dos formas:

Con If's	Con Do
I = 1	Do 1 I = 1,N
1 A(I) = 0	1 A (I) = 0
I = I + 1	

La forma general de la proposición Do es:

Do etiqueta variable entera = valor inicial, valor final, incremento

La etiqueta que aparece en la proposición DO le indica a la máquina hasta donde llegar el alcance del DO, esto es, define los límites dentro de los cuales se efectuará - la iteración.

La variable que servirá como un "contador" para el DO su valor inicial está dado en la proposición y se irá incrementando cada vez que llegue al final del DO, comparando su valor con el valor final especificado. En el caso de -- que sea mayor o igual, el ciclo termina y se ejecuta la proposición siguiente del bloque definido por el 00.

La última proposición en el bloque de un DO no puede - ser Go To, If Return o Do. En el caso en que sea necesario usar algunas veces estas proposiciones, se recurre a una proposición "muda" que es el Continue cuya única función es la de definir a una etiqueta.

Se puede dar el caso de tener varios bloques de DO's -- "anidados", esto es, uno dentro sel otro, siempre y cuando - cada uno está completamente abarcado por el más largo.

Para ejemplificar lo que se ha dicho, veamos un segmento de programa que multiplica dos matrices A, B cada una de NxN y guarda el producto en la matriz C de N x N.

Recordemos que si $C = A*B$

$$C_{ij} = \sum_{k=1}^{n} a_{ik}\ b_{kj}$$

El programa quedaría como:

```
                    .
                    .
                    .
                    DO 1 I = 1, N
                    DO 1 J = 1, N
                    C (I,J) = 0
                    DO 2 K = 1, N
2 C (I,J) = C(I,J) + A(I,K) * B(K,J)
1 CONTINUE
```

SUBRUTINAS Y FUNCIONES

Muy frecuentemente sucede que una Sección de programa, o secuencia de instrucciones es frecuentemente usada. Si tal caso sucede tal sección del programa es usualmente -- identificada como una rutina separada llamada SUBROUTINE en Fortran (PROCEDURE ALGOL).

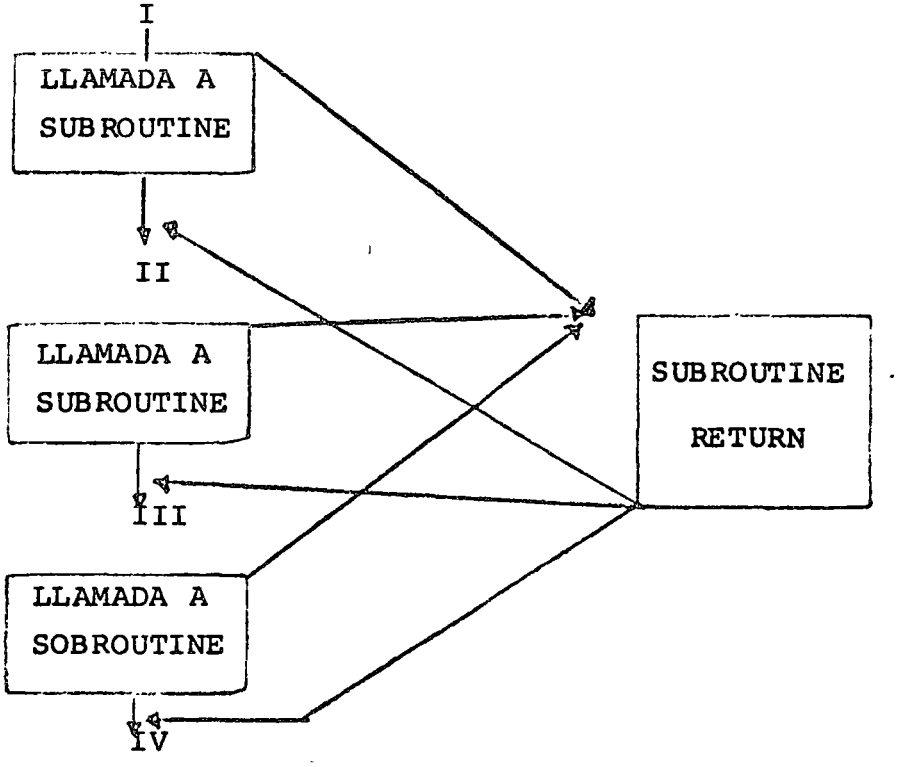
Cuando una subrutina es definida, se le dá un nombre - de identificación, y sus argumentos son identificados, estos argumentos son sus variables. A continuación estudiaremos - la naturaleza, uso y objeto de las subrutinas.

Cada subrutina debe de empezar con la proposición subrutina, su nombre, y una lista de argumentos y terminar con - las PROPOSICIONES RETURN Y END.

```
      SUBROUTINE SUMPRO ( A, B, SUM, PRO ).
      REAL A (20), B (20), SUM (20), PRO (20).
1     SUM (1) = A (I)  + B (I)
      PROD (I) =  A(I) * B(I)
      1 = I + 1
```

```
      IF  (I - 20) 1, 1, 2
2     RETURN
      END
```

La manera como se transfiere el control se ilustra en la siguiente figura. El control se transfiere a la subrutina cada vez que es llamada. El control se transfiere al programa principal (u otra subrutina) cuando encuentra la proposición RETURN. La siguiente proposición a ejecutar es la siguiente A la llamada.



La estructura de la subrutina es la siguiente:

```
# Tarjetas de control
  Real a (1000)
  .
  .
  .
  Call error
  .
  .
  .
  .
  End.
Subroutine error
  (WRITE) (6, 1)
1 Format ("un error del tipo a ocurrió")
  Return
  End
```

Cuando una subrutina es usada, algunos de los argumentos (argumentos en la subrutina de referencia) pueden ser expresiones, veamos a través de un ejemplo como son estos tratados.

```
Programa
Principal    .
             .
             Call suma ( A * A, B )
             .
             .
             .
             .
             End
```

Cuando ejecuta la preposición call suma (A * B, B), - la expresión A * A, es valuada y su valor es asignado a una variable temporal no accesible al programador llamémosle t. El segundo argumento es simplemente una variable, su nom-- bre es pasado a la subrutina suma. O sea que call suma (A* A, B) es equivalente A;

$$t = A * A$$

A esta manera de tratar argumentos se le conoce como "Llamada por valor;

En general, una subrutina admite determinados valores de entrada y "regresa" al programa principal otros valores, por ejemplo, en la subrutina SUMPRO (A, B, SUM, PRO) los valo-- res de entrada son A, B; y los valores de regreso son SUM y PRO.

Existe otro tipo de subrutinas que regresan un solo valor, por lo que son llamadas funciones. En este tipo de sub rutinas todos los argumentos representan valores de entrada y el valor de salida queda asociado al nombre de la subrutina.

Veamos un ejemplo: queremos una subrutina que admita -- tres valores A, B, C y calcule $A^2 + B^2 + C^2$ si A y B son positivos o $\frac{-C}{A + B}$ si A ó B son negativos, se procede a declarar la subrutina:

```
      FUNCION      F ( A, B, C )
      IF (A. GE. 0. OR. B. GE. 0) GO TO 2
      F = - (C/ ( A + B ) ) .
      RETURN
    2 F + SORT ( A * * 2 + B ** 2 + C ** 2 )
      RETURN
      END
```

METODO DE NEWTON

Este método sea tal vez el método más popular para encontrar los ceros (raíces) de una función de una variable f (X). Es decir se encuentra una X tal que f (X) = 0.

El método de Newton es un método iterativo que produce una secuencia de aproximación a la raíz, siempre y cuando:

a) f (X) sea continua y diferenciable en la vecindad de la raíz, y que las segundas derivadas de f (X) no lleguen a ser excesivamente grandes.

b) Se puede dar un intento inicial del valor de la raíz "bueno".

Para funciones de variables reales, el método de Newton tiene una interpretación geométrica simple somo se ilustra en la siguiente figura:

Suponga que queremos encontrar una raíz de la función f (x), es decir el punto donde f (x) corta el eje x. Supongamos que la curva tiene la forma de la figura anterior, si nuestro primer intento es x_1, x_2 será una mejor aproximación de la raíz la cual se obtiene encontrando la intersección de la tangente (x_1, f (x_1)) en el eje x. Este proceso se repite varias veces, cada vez utilizando la x_2 calculada de la x_1 anterior; hasta encontrar la raíz con la aproximación deseada.

Refiriéndose nuevamente a la figura anterior deje que f^1 (x) sea la derivada de f (x) valuada en el punto x_1, por consideraciones geométricas.

$$f^1 (x_1) = \frac{f (x_1)}{x_1 - x_2} \qquad 1$$

Y la nueva aproximación de la raíz

$$x_2 = x_1 - \frac{f (x_1)}{f^1(x_1)} \qquad 2$$

Ejemplo:

Método para calcular la raíz cuadrada.

Si la ecuación $x^2 = A$, la ecuación a resolver será:

$$f (x) = A - x^2 = 0$$

$$f^1 (x) = - 2x$$

De la fórmula 2

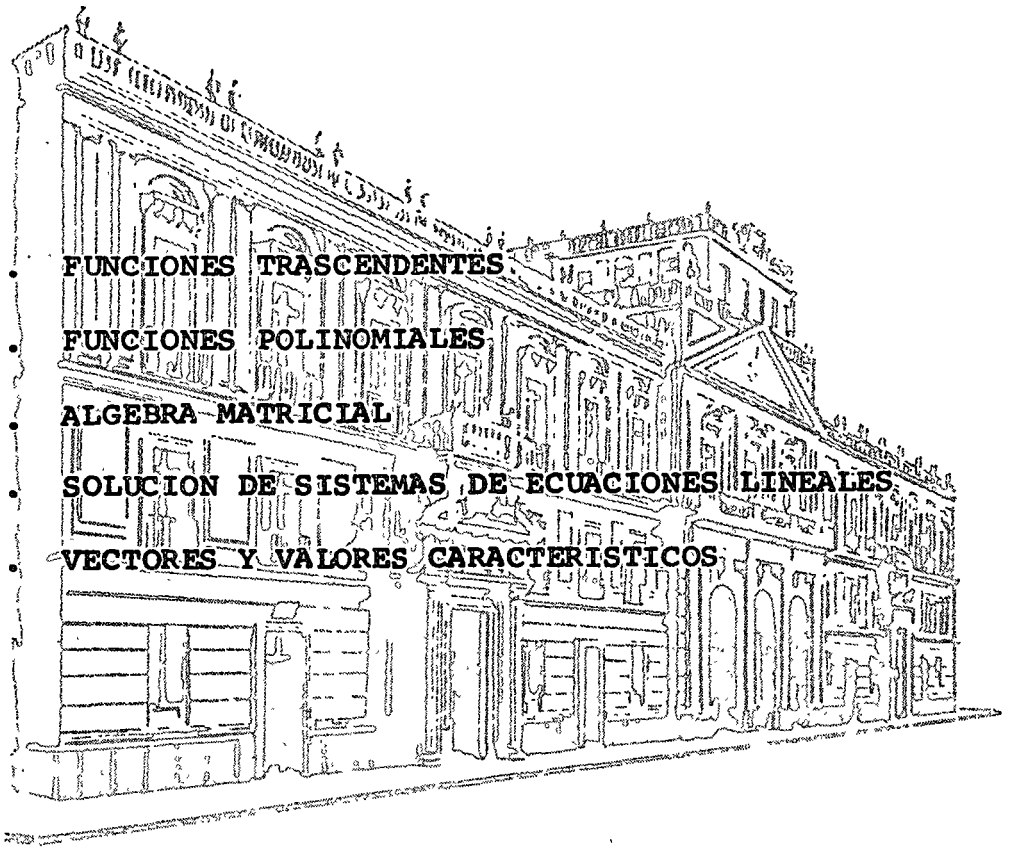
$$x_2 = x_1 \quad \frac{- A ------x1^2}{- 2 x_1} \quad \text{ESCRIBIENDO: } x_2 = \frac{1}{2} \frac{A}{x_1} + x_1$$

METODO DE NEWTON

```
c     PROGRAMA ENCONTRAR LOS CEROS DE UNA FUNCION
c
c
c     AQUI SE DEFINEN LA FUNCION  Y SU DERIVADA
c
      F (X)  =
      DF (X) =
c     LEE EL VALOR INICIAL DE LA SOLUCION Y LA TOLERANCIA
      DE ERROR
      READ (5, 1 )  XVIEJA, EPS
    1 FORMAT  ( 2 F10.0)
c     COMIENZAN LAS ITERACIONES
    2 XNUEVA = XVIEJA - F (XVIEJA)  /  DF (XVIEJA)
      DIF =  ABS  (XVIEJA  - XNUEVA)
      IF ( DIF. LT. EPS ) GO TO  3
      XVIEJA = XNUEVA
      GO TO 2
    3 WRITE (6, 4) XNUEVA, DIF
    4 FORMAT ( " X = " , F12.4, 5 X, "ERROR = ", E14.10 )
      END
```

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL



- FUNCIONES TRASCENDENTES
- FUNCIONES POLINOMIALES
- ALGEBRA MATRICIAL
- SOLUCION DE SISTEMAS DE ECUACIONES LINEALES
- VECTORES Y VALORES CARACTERISTICOS

ARMANDO TORRES FENTANES

MARZO DE 1976

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels.: 521-40-23 521-73-35 5123-123

CENTRO DE EDUCACION CONTINUA

TEMA: METODOS NUMERICOS

I) SOLUCION DE ECUACIONES

1. Funciones transcendentes

- Búsqueda por partición de intervalos.
- Método de aproximaciones sucesivas.
- Método de Newton Raphson.
- Método de Newton 2o. orden.
- Método de Von Mises

2. Funciones polinomiales:

- Teoremas.
- División sintética.
- Regla de los signos de Descartes.
- Raíces racionales (División Sintética)
- Raíces irracionales (Newton - Raphson y Newton 2o. orden)
- Método de Lin Bairstow

II) SOLUCION DE SISTEMAS DE ECUACIONES LINEALES

1. Operaciones matriciales:

- Suma y Resta
- Multiplicación.
- Obtención de la matriz inversa por el método de Gauss - Jordan

2. Solución Sist. de ecuaciones:

- Método de Gauss
- Método de Gauss-Jordan
- Método de Jacobi
- Método de Gauss - Seidel

III) VECTORES Y VALORES CARACTERISTICOS.

- Método directo
- Método de Krylov
- Método de Jacobi para obtener el mayor valor característico
- Método de Jacobi para obtener el menor valor característico.

IV) APROXIMACION POLINOMIAL

1. Interpolación con valores muestrales a espacios iguales:

- Método lineal
- Método de Newton

2. Interpolación con valores muestrales desigualmente espaciados.

- Método de Lagrange.

3. Aproximación de puntos por polinomios.

- Método de los mínimos cuadrados.

V) DERIVACION E INTEGRACION NUMERICA.

1. Derivación:

- Método de las diferencias.

2. Integración :

- Método trapezoidal

- Método de Simpson 1/3

- Método de Simpson 3/8

VI) SOLUCION ECUACIONES DIFERENCIALES ORDINARIAS.

- Método de Euler

- Método de Euler mejorado

- Método de Runge - Kutta

- Método de las diferencias finitas.

VII) SOLUCION SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS DE 1er. ORDEN.

- Método de Runge - Kutta

- Método de variación de parámetros

VIII) METODOS ESTADISTICOS Y PROBABILISTICOS

- Generación de n.a. por método de la congruencia lineal

- Método de la transformada inversa.

- Generación de v.a. gaussianas por el método polar.

- Generación de v.a. con f.d.p. exponencial

- Métodos de Monte - Carlo

IX) OPTIMIZACION DE FUNCIONES

1. Funciones unidimensionales:

- Método Aleatorio

- Método de Fibonacci.

2. Funciones multidimensionales.

- Búsqueda por Gradiente.

INTRODUCCION

El objeto del presente curso es dar un panorama general de la aplicación de la computadora para resolver problemas matemáticos y a la vez familiarizar a los asistentes al curso con el manejo de los métodos existentes, a fin de que se cuente con una base para estudios posteriores.

Por las razones antes mencionadas, durante el curso no se intentará agotar toda la materia ni profundizar demasiado en los aspectos teóricos de los métodos. En base a dicha orientación se elaboraron los apuntes, solo como guía del curso y no como un tratado profundo de la materia; cuando se desee ahondar en algún tema se recomienda acudir a la bibliografía citada.

Consciente de que pueden existir errores de impresión u otra especie, el redactor de estos apuntes se responsabiliza y disculpa por ello.

ARMANDO TORRES FENTANES

APUNTES METODOS NUMERICOS

I) SOLUCION DE ECUACIONES

Existen dos tipos básicos de ecuaciones:

- trascendentes $(e^{-x} - \text{sen } 3X = 0)$
- polinomiales $(X^4 - 3X^3 + 10X^2 + 1 = 0)$

1. Funciones trascendentes

a) Búsqueda por partición de intervalos.

La metodología a seguir es:

- trazar aprox. la curva y ver en qué intervalo se encuentra la solución.
- discretizar el intervalo y los valores de la función $F(x) = 0$
- los puntos intermedios más próximos con las siguientes características: $F(X_1) < 0$ y $F(X_2) > 0$; tomarlos como nuevos límites del intervalo y así sucesivamente según sea la precisión buscada.

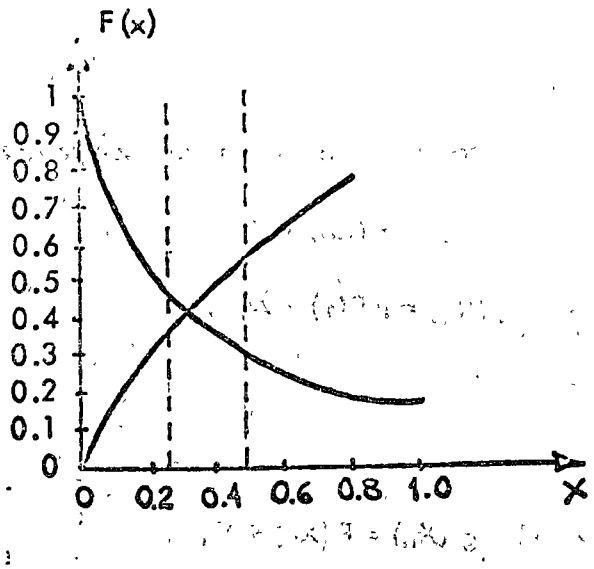
Ejemplo

Resolver la ecuación $e^{-x} - \text{sen}(\pi X/2) = 0$.

Sol.

Se traza la curva y busca el intervalo dentro del cual está la Sol.

./.



y se forma la siguiente tabla:

	Paso i	X_i	e^{-x_i} (①)	$\text{sen}\frac{\pi X_i}{2}$ (②)	① − ② +	① − ② −
	1	.5	.6085	.707		✓
I	2	.25	.7788	.3827	✓	
II	3	.375	.6873	.556	✓	
	4	.4375	.6456	.634	✓	
III	5	.46875	.6258	.6716		✓

$$\therefore \quad .437 < X < .4687$$

b) Método de aproximaciones sucesivas

Sea $F(x) = 0$, sumando X en ambos miembros :

$$F(x) + X = g(x) = X$$

si $F(a) = 0$, se concluye:

$g(a) = a$

eso sucede si la raíz es exacta, en caso contrario:

$$X_1 = g(X_o) = F(X_o) + X_o$$

$$X_2 = g(X_1) = F(X_1) + X_1$$

$$\vdots \qquad\qquad \vdots$$

$$X_{n+1} = g(X_n) = F(X_n) + X_n$$

el método se detiene cuando:

$$|X_{n+1} - X_n| < \varepsilon$$

es decir, cuando los valores de X_n y X_{n+1} son casi iguales. Este método converge si:

$$|g'(x)| < 1$$

Ejemplo

Dar la raíz negativa de $X = \sqrt{0.5}$.

Sol.

Del enunciado se tiene:

$$X^2 = 0.5$$

$$X^2 - 0.5 = 0 = F(x)$$

Sumando "X" en ambos miembros:

$$X^2 - 0.5 + X = g(x) = X$$

$$X_{n+1} = X_n^2 - 0.5 + X_n$$

Sea $X_o = -0.6$:

$$X_1 = (-0.6)^2 - 0.6 - 0.5 = -0.74$$

$$X_2 = (-0.74)^2 - 0.74 - 0.5 = -0.6924$$

$$X_3 = (-0.6924)^2 - 0.6924 - 0.5 = -0.713$$

$$X_4 = (-0.731)^2 - 0.713 - 0.5 = -0.704$$

$$X_5 = (-0.704)^2 - 0.704 - 0.5 = -0.706$$

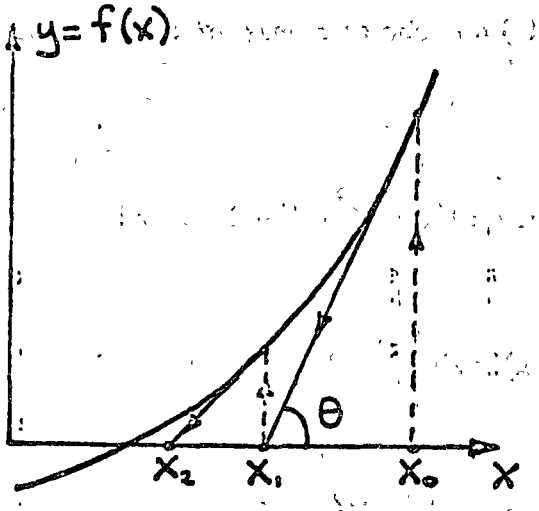
de donde

$$X \cong -0.7$$

c) Newton - Raphson

El método consiste en trazar la tangente a la curva en un punto (X_o, Y_o) y la intersección de esa recta con el eje "X" dará el nuevo valor (X_1, Y_1), el método se repite sucesivamente hasta que:

$$|X_{n+1} - X_n| < \varepsilon$$



$$f'(X_o) = tg\,\theta = \frac{f(X_o) - 0}{X_o - X_1} = \frac{f(x_o)}{X_o - X_1}$$

por lo que :

$$(X_o - X_1)\ f'(X_o) = f(X_o)$$

$$X_1 = X_o - \frac{f(X_o)}{f'(X_o)}$$

$$\vdots$$

$$X_{n+1} = X_n - \frac{f(X_n)}{f'(X_n)}$$

El método converge si:

$$|g'(X)| < 1$$

donde:

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

Si X_o es la primera aproximación, el método converge si:

- X_o está suficientemente cercano a la raíz
- $f''(x)$ no debe ser excesivamente grande
- $f'(x)$ no debe estar muy próxima a cero.

Ejemplo

Resolver la ecuación $f(x) = X^2 - C = 0,\ C = 24$

Sol.

$$f'(x) = 2X$$

por lo que:

$$X_{n+1} = X_n - \frac{X^2_n - C}{2X_n}$$

$$= \frac{1}{2}\left[X_n + \frac{C}{X_n}\right]$$

Sea $X_o = 1$:

$$X_1 = \frac{1}{2}\ (1 + 24) = 12.5$$

$$X_2 = \frac{1}{2}\ \left(12.5 + \frac{24}{12.5}\right) = 7.21$$

$$X_3 = \frac{1}{2}\ \left(7.21 + \frac{24}{7.21}\right) = 5.2693$$

$$X_4 = \frac{1}{2}\ \left(5.2693 + \frac{24}{5.2693}\right) = 4.9119$$

d) Método de Newton de 2o. Orden

Este método funciona igual que el anterior solo que converge más rápidamente y sus restricciones son iguales que en el caso anterior.

Expandiendo f(x) en series de Taylor :

$$f(X_n) = f(X_{n-1}) + \frac{(X_n - X_{n-1})}{1!} f'(X_{n-1}) + \frac{(X_n - X_{n-1})^2}{2!} f''(X_{n-1}) + \ldots$$

Si se toman los 3 primeros miembros y considera X_n como raíz:

$$f(X_n) = f(X_{n-1}) + \frac{(X_n - X_{n-1})}{1!} f'(X_{n-1}) + \frac{(X_n - X_{n-1})^2}{2!} f''(X_{n-1})$$

$$0 = f(X_{n-1}) + \frac{(X_n - X_{n-1})}{1!} f'(X_{n-1})$$

$$+ \frac{(X_n - X_{n-1})^2}{2!} f''(X_{n-1})$$

pero :

$$X_n - X_{n-1} = - \frac{f(X_{n-1})}{f'(X_{n-1})}$$

$$0 = F(X_{n-1}) + (X_n - X_{n-1}) \left[f'(X_{n-1}) - \frac{1}{2} \frac{f(X_{n-1})}{f'(X_{n-1})} f''(X_{n-1}) \right]$$

$$\frac{1}{X_n - X_{n-1}} = - \frac{f'(X_{n-1})}{f(X_{n-1})} + \frac{1}{2} \frac{f''(X_{n-1})}{f'(X_{n-1})}$$

con lo que la fórmula de recurrencia está dada por :

$$\Delta X_{n-1} = X_n - X_{n-1}$$

$$X_n = X_{n-1} + \Delta X_{n-1}$$

Ejemplo

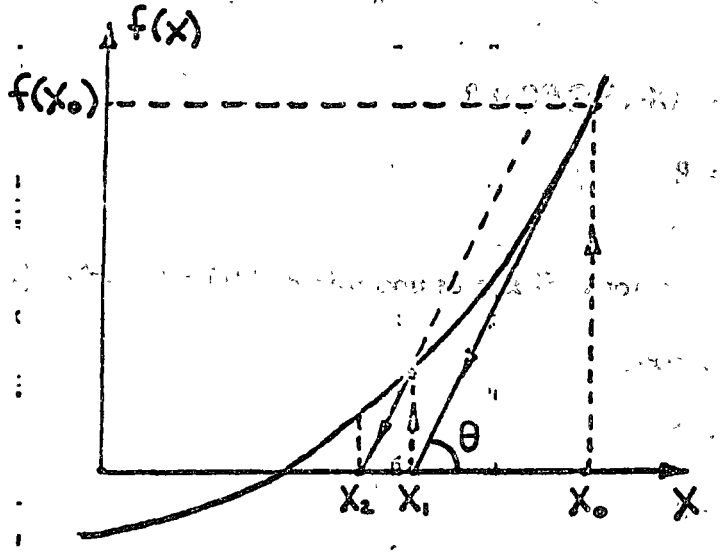
Resolver $f(x) = \text{sen } X = 0$, suponiendo $X_o = 1.165$

Sol.

X_n	f(x) = sen x	f'(x)= cos X	f''(x)=-sen X	ΔX
$X_o = 1.165$ =66.7°	.9174	.3978	-.9174	-0.6302
$X_1 = 0.534$ = 30.5 °	.508	.8611	-0.508	-0,5024
$X_2 = 0.0315$ =1.79°	.0314	.9995	-0.0314	-0.0314
$X_3 = 0.0001$ ≅ 0°	0			

e) Método de Von Mises

Consiste en tomar siempre la primera tangente como trayectoria de búsqueda. Es más lento pero reduce el inconveniente de que f'(x) quede muy próxima a cero. Se utiliza este método en los casos en que los puntos cercanos a la raíz tienen pendiente nula.



$$f'(x_o) = \frac{f(x_o) - 0}{X_o - X_1}$$

$$\vdots$$

$$f'(x_o) = \frac{f(x_{n-1})}{X_{n-1} - X_n}$$

$$\therefore \quad X_n = X_{n-1} - \frac{f(X_{n-1})}{f'(X_o)}$$

2. Funciones polinomiales

Polinomios de orden "n" son aquellos que tienen la siguiente configuración :

$$P(x) = A_n X^n + A_{n-1} X^{n-1} + \ldots + A_2 X^2 + A_1 X + A_o$$

Para resolver estas ecuaciones se emplean los métodos antes vistos pero con modificaciones que permiten realizar las operaciones más rápidamente.

a) Teoremas

-T. del resíduo : el resíduo resultante de dividir el polinomio P(x) entre el binomio X-a es igual a P(a).

$$P(x) = (X-a)\, Q(X) + R$$

$$P(a) = R$$

-T. del factor : Si x=a es una raíz de $P(x) = 0 \implies$ (x-a) es un factor de P(x).

$$P(a) = 0$$

$$\implies R = 0$$

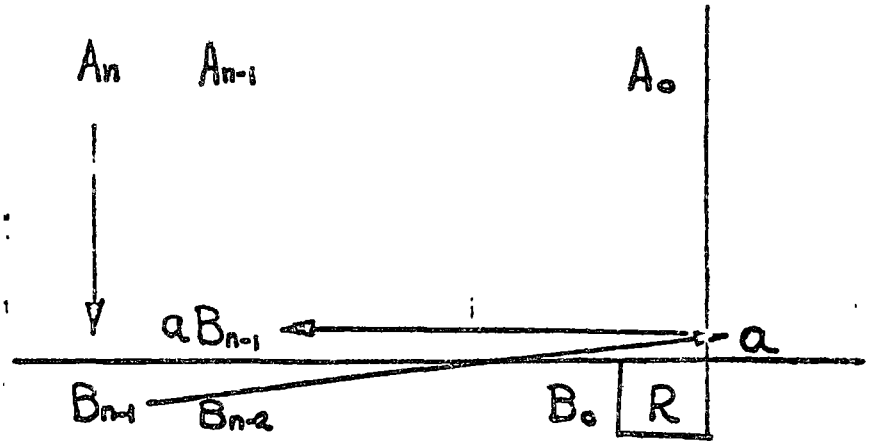
$$\therefore \quad P(x) = (x-a)\, Q(x)$$

b) División sintética

Permite efectuar numéricamente la división de P(x) por el factor x-a.

Sea:

$P(x) = A_n X^n + A_{n-1} X^{n-1} + \ldots + A_o$



$B_{n-1} = A_n$

$B_{n-2} = A_{n-1} + aB_{n-1}$

$Q(x) = B_{n-1} X^{n-1} + B_{n-2} X^{n-2} + \ldots + B_o + \frac{R}{X-a}$

Ejemplo

Si $P(x) = 3X^4 - 7X^3 + 2X^2 + 1$, encontrar Q(x) si se divide P(x) por $X + 2$ y hallar P(-2) aplicando el T. del residuo.

Sol.

3	-7	2	0	1	
	-6	26	-56	112	-2
3	-13	28	-56	113	

R.

$Q(x) = 3X^3 - 13X^2 + 28X - 56$

$R = P(-2) = 113$

Demostrar que $X = -2$ es raíz de la ecuación $X^3 - X^2 - 4X + 4 = 0$

Sol.

1	-1	-4	4	
	-2	6	-4	-2
1	-3	2	0	

$R = 0$

$P(x) = (X+2)(X^2 - 3X + 2)$

c) Regla de los signos de Descartes

Esta regla sirve para determinar el máximo número de raíces **positivas ó negativas** y sus posibles tipos (reales o imaginarias). El procedimiento es:

- Obtener todas las raíces nulas y reducir el polinomio.

$A_n X^n + \ldots + A_m X^m = 0$

$\Rightarrow$ existen m raíces nulas

$X^m \underbrace{(A_n X^{n-m} + \ldots + A_m)}_{\text{polinomio reducido para la búsqueda.}} = 0$

- Ordenar el polinomio en orden decreciente:

$$A_n X^n + A_{n-1} X^{n-1} + \ldots + A_1 X + A_o = 0$$

- El número de raíces reales positivas es igual al número de cambios de signo en P(x) ó un número menor en pares. El número de raíces reales negativas es igual al número de cambios de signo en P(-x) o un número menor en pares.

- Las raíces complejas, si existen, siempre aparecen por pares conjugados.

- Una ecuación de grado "n" tiene "n" raíces reales o complejas.

- Establecer un cuadro con las posibilidades que se tengan para las "n" raíces.

Caso / Tipo	I	II	III	...
positivas				
negativas				
complejas				
total				...

Ejemplo

Indagar los posibles tipos de raíces para el siguiente polinomio:

$$P(x) = X^3 - 7X^2 + 10X + 16 = 0$$

Sol.

$$P(-X) = -X^3 - 7X^2 + 10X + 16 = 0$$

No. de cambios en $P(x) = 2 \Rightarrow$ r.p. : 2, 0

No. de cambios en $P(-X) = 1 \Leftrightarrow$ r.n.: 1

Total de raíces : 3

	I	II
$X > 0$	2	0
$X < 0$	1	1
compleja	0	2
Total	3	3

d) Obtención de raíces racionales

Las posibles raíces racionales estarán dadas por :

$$X\text{posible} = \pm \frac{\text{múltiplos de } A_o}{A_n}$$

Se aplica división sintética en P(x) para cada raíz posible y si R=0, entonces Xposible será raíz de P(x).

Las raíces se buscan de acuerdo al cuadro obtenido por la regla de los signos de Descartes.

Ejemplo

Encontrar las raíces racionales del polinomio:

$$P(x) = X^3 - 7X^2 - 10X + 16 = 0$$

Sol.

$A_o = 16$

$A_n = 1$

múltiplos de A_o: $\pm 16, \pm 8, \pm 4, \pm 2, \pm 1$

$$X_p = \pm\frac{16}{1}, \pm\frac{8}{1}, \pm\frac{4}{1}, \pm\frac{2}{1}, \pm\frac{1}{1}$$

1	-7	-10	16	
	-2	18	-16	-2
1	-9	8	0	

$\therefore \quad X_1 = -2$

$$Q(x) = X^2 - 9X + 8$$

de donde:

$$X = \frac{9 \pm \sqrt{81 - 32}}{2} = \frac{9 \pm 7}{2}$$

$X_2 = 8$

$X_3 = 1$

e) <u>Obtención de raíces irracionales (Newton Raphson):</u>

$$X_{n+1} = X_n - \frac{P(X_n)}{P'(X_n)} \qquad (1.0)$$

sabemos que :

$$P(x) = (X - X_n)\, Q(x) + R_1 \qquad (1.1)$$

si :

$$X = X_n$$

$$\Rightarrow P(X_n) = R_1 \qquad (1.2)$$

derivando (1.1)

$$P'(x) = (X - X_n)\, Q'(x) + Q(x) \qquad (1.3)$$

si :

$$X = X_n$$

$$P'(X_n) = Q(X_n)$$

donde $Q(X_n)$ es el residuo que se obtiene al dividir $Q(x)$ por $X-X_n$:

$$Q(x) = (X - X_n)\ \overline{Q}(x) + R_2 \qquad (1.4)$$

$$P'(X_n) = Q(X_n) = R_2 \qquad (1.5)$$

substituyendo (1.5) y (1.2) en (1.0) :

$$X_{n+1} = X_n - \frac{R_1}{R_2}$$

Para aplicar el método es necesario obtener primero las cotas de la raíz mediante división sintética y esas cotas se utilizan como valores iniciales:

$$P(X_1) > 0$$

$$P(X_2) < 0$$

$$\therefore \quad X_1 \leq X \leq X_2$$

Newton 2o. orden

Se tiene :

$$\frac{1}{\Delta X_n} = -\frac{P'(X_n)}{P(X_n)} + \frac{1}{2}\ \frac{P''(X_n)}{P'(X_n)} \qquad (1.6)$$

$$X_{n+1} = X_n + \Delta X_n$$

Derivando (1.3):

$$P''(x) = (X-X_n)Q''(x) + Q'(x) + Q'(x)$$

$$= (X-X_n)\ Q''(x) + 2Q'(x)$$

Si $X = X_n$:

$$P''(X_n) = 2\,Q'(X_n) \qquad (1.7)$$

derivando (1.4)

$$Q'(x) = (x-x_n)\,\overline{Q}'(x) + \overline{Q}(x)$$

$$Q'(x_n) = \overline{Q}(x_n)$$

pero $\overline{Q}(X_n)$ es el residuo que se obtiene al dividir $\overline{Q}(X)$ por $(X-X_n)$:

$$Q'(X_n) = \overline{Q}(X_n) = R_3 \qquad (1.8)$$

substituyendo (1.8) en (1.7):

$$P''(X_n) = 2R_3 \qquad (1.9)$$

substituyendo (1.9), (1.5), (1.2) en (1.6):

$$\frac{1}{\Delta X_n} = -\frac{R_2}{R_1} + \frac{R_3}{R_2}$$

$$X_{n+1} = X_n + \Delta X_n$$

Ejemplo

Obtener para el polinomio

$$X^4 - 6X^3 - X^2 - 2X - 8 = 0$$

el cuadro de posibles soluciones, una de las raíces irracionales por los 2 métodos vistos y expresar P(x) en función de dicha raíz e indicar Q(x)

Sol.

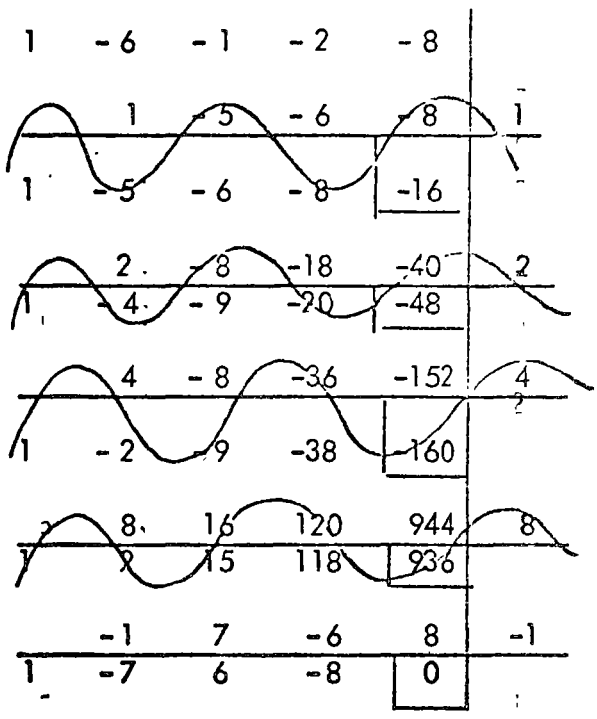
$P(x) = X^4 - 6X^3 - X^2 - 2X - 8$, r.p.: 1

$P(-x) = X^4 + 6X^3 - X^2 + 2X - 8$, r.n.: 3, 1

	I	II
$X > 0$	1	1
$X < 0$	3	1
comp.	0	2
total	4	4

$X_{pos\ rac.}$: $\pm 8, \pm 4, \pm 2, \pm 1$

aplicando división sintética:



$X_1 = -1$

$P(x) = (X+1)\ (X^3 - 7X^2 + 6X - 8)$

se trabajará con $Q(x) = X^3 - 7X^2 + 6X - 8$

de las operaciones efectuadas se observa :

$$P(4) < 0$$

$$P(8) > 0$$

$$\therefore \quad 4 \leq X \leq 8$$

Sea $X_0 = 6$:

Aplicando Newton Raphson :

1	-7	6	-8	
	6	-6	0	6
1	-1	0	-8	$= R_1$
	6	30	6	
1	5	30	R_2	

$$X_1 = X_0 - \frac{R_1}{R_2} = 6 - \frac{-8}{30} = 6.26$$

1	-7	6	-8	
	6.26	-4.63	8.58	6.26
1	-.74	1.37	.58	R_1
	6.26	34.55	6.26	
1	-5.52	35.92	R_2	

$$X_2 = 6.26 - \frac{.58}{35.92} = 6.244$$

Aplicando Newton 2o. Orden :

1	-7	6	-8	
	6	-6	0	6
1	-1	0	-8	$=R_1$
	6	30	6	
1	5	30	$=R_2$	
	6	6		
1	11	R_3		

$$\frac{1}{\Delta X_o} = -\frac{R_2}{R_1} + \frac{R_3}{R_2} = \frac{30}{-8} + \frac{11}{30} = 4.116$$

$$X_1 = X_o + \Delta X_o = 6 + .2429 = 6.242$$

$\vdots$

f) Método de Lin - Bairstow

Los métodos de Newton - Raphson y Newton 2o. orden solo dan raíces reales. Para obtener las raíces imaginarias se tienen dos procedimientos: obtener las raíces reales y después en el polinomio reducido buscar las raíces imaginarias o bien aplicar un método numérico.

Dentro de los métodos numéricos el más eficiente es el de Lin - Bairstow, del cual solo se darán sus principios básicos. Básicamente consiste en un proceso de descomposición del polinomio P(x) en formas cuadráticas y hallar la solución de esas ecuaciones.

Ejemplo

Dado el polinomio :

$P(y) = y^5 - 17\,y^4 + 124\,y^3 - 508\,y^2 + 1035\,y - 875$ encontrar sus raíces por el método de Lin - Bairstow.

Sol.

Dividir P(y) por el factor cuadrático $y^2 + py + q$:

$$P(y) = (y^2 + py + q)\ (y^3 + B_1\ y^2 + B_2\ y + B_3) + Ry + S$$

se igualan coeficientes de los términos de igual grado :

$$\left.\begin{aligned} P + B_1 &= -17 \\ B_2 + B_1 p + q &= 124 \\ B_3 + B_2\, p + B_1 q &= -508 \end{aligned}\right\}$$

$$\left.\begin{aligned} R + B_3\, p + B_2\ q &= 1035 \\ S + B_3\, q &= -875 \end{aligned}\right\}$$

de las 3 primeras ecuaciones:

$$B_1 = -17 - p$$

$$B_2 = 124 + 17\, p + p^2 - q$$

$$B_3 = -508 - 124\, p - 17\, p^2 - p^3 + 2pq + 17q$$

estos valores se substituyen en el segundo grupo de ecuaciones:

$$p^4 + 17p^3 + 124p^2 + 508\, p + 1035 - 3qp^2 -$$

$$- 3468 - 124q - q^2 = R$$

$$p^3 q + 17p^2 q + 124\, pq - 2\, pq^2 - 17q^2 +$$

$$+ 508\ q\ - 875 = S$$

el problema se resume a encontrar "p" y "q" tal que :

$$R = 0$$

$$S = 0$$

una vez hecho eso, se substituyen los valores en la forma cuadrática $y^2+py+q=0$; se encuentran sus raíces y se substituyen en B_1, B_2, B_3 teniéndose a continuación :

$$P(y) = (y^2 + py + q)\ \underbrace{(y^3 + B_1 y^2 + B_2 y + B_3)}_{Q(y)}$$

se aplica el mismo procedimiento para Q(y) y así sucesivamente.

Para el ejemplo tratado se llega a :

$$P = -4$$

$$q = 5$$

$$y^2 - 4y + 5 = 0$$

$$\therefore\ y = 2 \pm i$$

$$Q(y) = y^3 - 13 y^2 + 67 y - 303$$

II) SOLUCION DE SISTEMAS DE ECUACIONES LINEALES.

1. Operaciones matriciales.

Una matriz es un arreglo de elementos en "m" renglones y "n" columnas.

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & & & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix}$$

a) Suma ó Resta

Para sumar dos matrices se requiere que el número de renglones y columnas de una sean iguales a los de la otra.

$$A_{(m \times n)} + B_{(r \times s)}$$

es posible solo si

$$m = r$$

$$n = s$$

Si C representa la matriz suma :

$$C_{ij} = A_{ij} + B_{ij}$$

Ejemplo

Obtener la suma A + B si :

$$A = \begin{bmatrix} 3 & 1 \\ 2 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 3 & 2 \end{bmatrix}$$

Sol.

$$A + B = \begin{bmatrix} 4 & 1 \\ 5 & 2 \end{bmatrix}$$

b) Multiplicación

Dos matrices A y B se pueden multiplicar solo si el número de columnas de la primera son iguales a los renglones de la segunda, es decir :

Si $A_{(m \times n)}$

$B_{(r \times s)}$

existe AB $\Longleftrightarrow$ n = r

Si C representa la matriz producto se tendrá:

$$C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}$$

Ejemplo

Encontrar el producto AB si :

$$A = \begin{bmatrix} 3 & 1 \\ 2 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 0 \\ 3 & 2 \end{bmatrix}$$

Sol.

$$AB = \begin{bmatrix} 3 & 1 \\ 2 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0 \\ 3 & 2 \end{bmatrix} = \begin{bmatrix} (3+3) & (0+2) \\ (2+0) & (0+0) \end{bmatrix}$$

$$AB = \begin{bmatrix} 6 & 2 \\ 2 & 0 \end{bmatrix}$$

c) Matriz inversa por el método de Gauss - Jordan

La matriz inversa se denota por A^{-1} y cumple la siguiente propiedad :

$$A A^{-1} = I = A^{-1} A$$

donde I es la matriz identidad.

Solo existe inversa de una matriz A si $|A| \neq 0$, en cuyo caso A^{-1} se puede obtener por varios métodos, a continuación se describe el de Gauss - Jordan.

Representar la matriz A y la identidad I en una sola matriz:

$$[A \mid I] \qquad (II.0)$$

Transformar la matriz A en una matriz I aplicando las siguientes transformaciones a la matriz (II.0).

- multiplicación de un renglón por un escalar $\lambda \neq 0$.
- sumar a los elementos de un renglón los correspondientes de otro multiplicados por una constante (suma de equimúltiplos)
- intercambiar renglones

Ejemplo

Obtener la matriz inversa de la siguiente matriz:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 2 \end{bmatrix}$$

Sol.

$|A| = 2 - 6 \neq 0$, sí existe A^{-1}

Aplicando el método:

$$\left[\begin{array}{cc:cc} \textcircled{1} & 2 & 1 & 0 \\ 3 & 2 & 0 & 1 \end{array}\right]$$ renglón pivote

$$\left[\begin{array}{cc:cc} 1 & 2 & 1 & 0 \\ 0 & -4 & -3 & 1 \end{array}\right]$$ se divide $\div -4$

$$\left[\begin{array}{cc:cc} 1 & 2 & 1 & 0 \\ 0 & \textcircled{1} & .75 & -.25 \end{array}\right]$$ renglón pivote

$$\left[\begin{array}{cc:cc} 1 & 0 & -.5 & .5 \\ 0 & 1 & .75 & -.25 \end{array}\right]$$

$$\left[\begin{array}{c:c} I & A^{-1} \end{array}\right]$$

$$A^{-1} = \begin{bmatrix} -0.5 & 0.5 \\ .75 & -0.25 \end{bmatrix}$$

2. Solución sistemas de ecuaciones lineales.

Un sistema de ecuaciones lineales tiene la siguiente forma :

$$A_{11} X_1 + A_{12} X_2 + \ldots\ldots + A_{1n} X_n = b_1$$

$$\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots$$

$$A_{m1} X_1 + A_{m2} X_2 + \ldots\ldots + A_{mn} X_n = b_m$$

que matricialmente se puede expresar :

$$\begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & & \vdots \\ A_m & \cdots & A_{mn} \end{bmatrix} \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$$

$$\underline{A}\ \underline{X} = \underline{B}$$

para resolver dichos sistemas se tratarán algunos de los métodos existentes.

a) Gauss

Es un método de eliminación sistemática, el sistema "n" ecuaciones con "n" incógnitas se reduce a un sistema triangular que se empieza a resolver a partir del último renglón ; ó sea, se llega a un sistema:

$$A_{11} X_1 + A_{12} X_2 + \ldots\ldots + A_{1n} X_n = C$$

$$A_{22} X_2 + \ldots\ldots + A_{2n} X_n = C_2$$

$$\ddots \qquad \vdots \qquad\qquad \vdots$$

$$A_{nn}^{n-1} X_n = C_n$$

$$X_n = \frac{C_n}{A_{nn}^{n-1}}$$

Para ello se utilizan las siguientes transformaciones :

- intercambio de renglones
- suma de equimúltiplos de un renglón a otro renglón
- multiplicación de un renglón por un escalar $\lambda \neq 0$

Ejemplo

Resolver el siguiente sistema de ecuaciones :

$$X_1 + 4X_2 + X_3 = 7$$
$$X_1 + 6X_2 - X_3 = 13$$
$$2X_1 - X_2 + 2X_3 = 5$$

Sol.

Para evitar trabajar con las incógnitas se utiliza solo la matriz de coeficientes y el vector de términos independientes

$$* \begin{bmatrix} \textcircled{1} & 4 & 1 & 7 \\ 1 & 6 & -1 & 13 \\ 2 & -1 & 2 & 15 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 4 & 1 & 7 \\ 0 & 2 & -2 & 6 \\ 0 & -9 & 0 & -9 \end{bmatrix} \leftarrow \text{se divide } \div 2$$

$$* \begin{bmatrix} 1 & 4 & 1 & 7 \\ 0 & \textcircled{1} & -1 & 3 \\ 0 & -9 & 0 & -9 \end{bmatrix}$$

$$\left[\begin{array}{ccc|c} 1 & 4 & 1 & 7 \\ 0 & 1 & -1 & 3 \\ 0 & 0 & -9 & 18 \end{array}\right] \qquad \text{se divide} \div -9$$

$$X_3 = -\frac{18}{9} = -2$$

$$X_2 = 3 + X_3 = 1$$

$$X_1 = 7 - X_3 - 4X_2 = 5$$

b) Gauss - Jordan

Su proceso es el mismo que para obtener la matriz inversa, solo que en - vez de trabajar con la matriz A y la identidad, se trabaja con la matriz de coeficientes y el vector de términos independientes. Se transforma la matriz de -- coeficientes en una identidad empleando las transformaciones:

- suma de equimúltiplos de un renglón a otro.
- multiplicación de un renglón por $\lambda \neq 0$.
- intercambio de renglones

Se tiene que observar la siguiente regla :

- un renglón empleado como pivote no puede volverse a usar

Ejemplo

Resolver el siguiente sistema de ecuaciones :

$$X_1 - X_2 + X_3 = -4$$

$$5X_1 - 4X_2 + 3X_3 = -12$$

$$2X_1 + X_2 + X_3 = 11$$

Sol.

$$* \begin{bmatrix} \textcircled{1} & -1 & 1 & \vdots & -4 \\ 5 & -4 & 3 & \vdots & -12 \\ 2 & 1 & 1 & \vdots & 11 \end{bmatrix}$$

$$* \begin{bmatrix} 1 & -1 & 1 & \vdots & -4 \\ 0 & \textcircled{1} & -2 & \vdots & 8 \\ 0 & 3 & -1 & \vdots & 19 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & -1 & \vdots & 4 \\ 0 & 1 & -2 & \vdots & 8 \\ 0 & 0 & \textcircled{5} & \vdots & -5 \end{bmatrix}$$

$$* \begin{bmatrix} 1 & 0 & -1 & \vdots & 4 \\ 0 & 1 & -2 & \vdots & 8 \\ 0 & 0 & \textcircled{1} & \vdots & -1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & \vdots & 3 \\ 0 & 1 & 0 & \vdots & 6 \\ 0 & 0 & 1 & \vdots & -1 \end{bmatrix}$$

$X_1 = 3$

$X_2 = 6$

$X_3 = -1$

Los casos particulares que se presentan al aplicar el método son :

- sistema indeterminado:

$$\left[\begin{array}{ccc|c} 1 & 0 & 2 & 1 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

al presentarse esta situación hay que obtener las ecuaciones independientes que quedan y aplicar la metodología correspondiente a sistemas indetermina dos :

$$X_1 + 2X_3 = 1$$

$$X_2 + 3X_3 = 2$$

- sistema incompatible :

son sistemas que no tienen solución y al aplicar las transformaciones queda el siguiente patrón :

$$\left[\begin{array}{ccc|cc} 1 & 1 & 2 & 1r & \\ 0 & 2 & 3 & 2 & \\ 0 & 0 & 0 & \lambda & \neq 0 \end{array}\right]$$

o sea :

$$0 = \lambda \neq 0$$

lo cual es una contradicción.

c) Gauss - Jordán modificado

El método de Gauss - Jordan da una solución aproximada debida a los

redondeos, para obtener la solución más fiel posible lo que se hace es pivotear sobre los mayores elementos (en valor absoluto) que queden en la matriz de coeficientes transformada, respetando la siguiente restricción :

- un renglón que se haya empleado como pivote no puede volver a usarse.

Al terminar de aplicar el método se reordenan los renglones para obtener una matriz identidad. Este método se aplica también cuando los elementos de la diagonal principal son nulos.

Ejemplo

Resolver el siguiente sistema :

$$X_1 - X_2 + X_3 = -4$$

$$5X_1 - 4X_2 + 3X_3 = -12$$

$$2X_1 + X_2 + X_3 = 11$$

Sol.

$$\left[\begin{array}{ccc|c} 1 & -1 & 1 & -4 \\ *\,(5) & -4 & 3 & -12 \\ 2 & 1 & 1 & 11 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 0 & -1/5 & 2/5 & -8/5 \\ 1 & -4/5 & 3/5 & -12/5 \\ *\,0 & (+13/5) & -1/5 & 79/5 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 0 & 0 & 25/(13X5) & -25/(13X5) \\ 1 & 0 & 35/(13X5) & 160/(13X5) \\ 0 & 1 & -1/13 & 79/13 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 0 & 0 & 1 & -1 \\ 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 6 \end{array}\right]$$

reordenando :

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 6 \\ 0 & 0 & 1 & -1 \end{array}\right]$$

$X_1 = 3$

$X_2 = 6$

$X_3 = -1$

d) Método de Jacobi

Este método se aplica cuando la matriz de coeficientes (A) cumple los siguientes requisistos :

- **los elementos no nulos se acumulan en la diagonal principal**
- **los elementos de la diagonal principal son mayores en valor absoluto que los demás de su renglón correspondiente.**

El procedimiento se describe a continuación.

Sea el sistema :

$$\underline{A}\ \underline{X} = \underline{b}$$

donde :

$$\underline{A} = \underline{D} + \underline{R} \qquad (\underline{D} : \text{matriz diagonal})$$

por lo que :

$$(\underline{D} + \underline{R})\ \underline{X} = \underline{b}$$

$$\underline{D}\ \underline{X} = \underline{b} - \underline{R}\ \underline{X}$$

$$\underline{X} = \underline{D}^{-1}\ \underline{b} - \underline{D}^{-1}\ \underline{R}\ \underline{X} \qquad (II.1)$$

de la ecuación (II.1) se obtiene la siguiente fórmula iterativa :

$$\underline{X}_{k+1} = \underline{D}^{-1}\ \underline{b} - \underline{D}^{-1}\ \underline{R}\ \underline{X}_k \ .- \qquad (II.2)$$

En ocasiones un simple intercambio de líneas permite aplicar el método .

La ecuación (II.2) lo que indica es que de la 1a. ecuación se despeja X_1, de la 2a. X_2 y así sucesivamente:

$$(II.3) \quad \begin{cases} X_1 = \dfrac{1}{A_{11}} \left[b_1 - A_{12}\ X_2 - A_{13}\ X_3 - \ldots - A_{1n}\ X_n \right] \\ X_n = \dfrac{1}{A_{nn}} \left[b_n - A_{n1}\ X_1 - A_{n2}\ X_2 \ldots - A_{nn-1}\ X_{n-1} \right] \end{cases}$$

Para arrancar el método se establece una solución aproximada:

$$\underline{X}_o = \begin{bmatrix} X_1^o \\ X_2^o \\ \vdots \\ X_n^o \end{bmatrix}$$

lo cual se substituye en el segundo miembro de (II.3), para obtener :

$$\underline{X}_1 = \begin{bmatrix} X_1' \\ X_2' \\ \vdots \\ X_n' \end{bmatrix}$$

y así sucesivamente hasta que :

$$\left| \underline{X}_{n+1} - \underline{X}_n \right| < \varepsilon$$

Ejemplo

Resolver el sistema de ecuaciones :

$$4X_1 - X_2 = 2$$

$$-X_1 + 4X_2 - X_3 = 6$$

$$-X_2 + 4X_3 = 2$$

Sol.

Despejando :

$$X_1 = 0.5 + 0.25\, X_2$$

$$X_2 = 1.5 + .25\, X_1 + .25\, X_3$$

$$X_3 = 0.5 + 0.25\, X_2$$

lo que da la siguiente fórmula de recurrencia :

$$x_1^{(k+1)} = .5 + .25\, x_2^{(k)}$$

$$x_2^{(k+1)} = 1.5 + .25\, x_1^{(k)} + .25\, x_3^{(k)}$$

$$x_3^{(k+1)} = .5 + .25\, x_2^{(k)}$$

sea :

$$\underline{x}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

iterando :

$$x_1^{(1)} = .5 + .25(0) = .5$$

$$x_2^{(1)} = 1.5 + .25\,(0) + .25\,(0) = 1.5$$

$$x_3^{(1)} = .5 + .25\,(0) = .5$$

$$\underline{x}_1 = \begin{bmatrix} .5 \\ 1.5 \\ .5 \end{bmatrix}$$

así sucesivamente se tendría :

$$\underline{x}_2 = \begin{bmatrix} .875 \\ 1.75 \\ .875 \end{bmatrix} \qquad \underline{x}_4 = \begin{bmatrix} .985 \\ 1.97 \\ .985 \end{bmatrix}$$

$$\underline{x}_6 = \begin{bmatrix} .998 \\ 2. \\ .998 \end{bmatrix} \qquad \underline{x}_8 = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

d) Gauss Seidel

Es igual que el de Jacobi y cuenta con las mismas restricciones solo que es más rápido ya que en cuanto se obtiene $X_i^{(K+1)}$, se substituye su valor inmediatamente en las ecuaciones, o sea :

$$X_1^{(K+1)} = \frac{1}{A_{11}} \left[b_1 - A_{12} X_2^{(K)} - \ldots - A_{1n} X_n^{(K)} \right]$$

$$X_2^{(K+1)} = \frac{1}{A_{22}} \left[b_2 - A_{21} X_1^{(K+1)} - A_{23} X_3^{(k)} - \ldots - A_{2n} X_n^{(k)} \right]$$

$$X_n^{(k+1)} = \frac{1}{A_{nn}} \left[b_n - A_{n1} X_1^{(k+1)} - \ldots - A_{nn-1} X_{n-1}^{(k+1)} \right]$$

Ejemplo

Resolver el sistema de ecuaciones :

$$4X_1 - X_2 = 2$$

$$-X_1 + 4X_2 - X_3 = 6$$

$$-X_2 + 4X_3 = 2$$

Sol.

Despejando :

$$X_1 = .5 + .25 X_2$$

$$X_2 = 1.5 + .25 X_1 + .25 X_3$$

$$X_3 = .5 + .25 X_2$$

de donde :

$$X_1^{(k+1)} = .5 + .25 X_2^{(k)}$$

$$X_2^{(k+1)} = 1.5 + .25 X_1^{(k+1)} + .25 X_3 (k)$$

$$X_3^{(k+1)} = .5 + .25 X_2^{(k+1)}$$

sea :

$$\underline{X_o} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

substituyendo :

$$X_1^{(1)} = .5 + .25 (0) = .5$$

$$X_2^{(1)} = 1.5 + .25 (.5) + .25 (0) = 1.63$$

$$X_3^{(1)} = .5 + .25 (1.63) = .91$$

así sucesivamente :

$$\underline{X_3} = \begin{bmatrix} .99 \\ 2 \\ 1 \end{bmatrix} \qquad \underline{X_4} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \qquad \underline{X_5} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

$$X_1 = 1$$

$$X_2 = 2$$

$$X_3 = 1$$

III) VECTORES Y VALORES CARACTERISTICOS

Valores característicos de una matriz no singular $\underline{A}$ son los valores λ para los cuales se cumple:

$$\underline{A}\ \underline{x} = \lambda\ \underline{x}\ ,\ \underline{x} \neq \underline{0} \qquad (III.0)$$

donde $\underline{x}$ se conoce como el vector característico asociado al valor característico λ, para cada valor característico existe un vector característico. En una matriz de orden "n" hay "n" valores y vectores característicos.

Estos valores característicos se utilizan en problemas de carga y pandeo para determinar las condiciones de carga crítica, en sistemas oscilatorio, mecánicos y eléctricos para determinar las frecuencias naturales que caracterizan el comportamiento del sistema.

a) Método directo

De (III.0) se tiene :

$$(\underline{A} - \lambda\underline{I})\ \underline{X} = 0,\ \underline{X} \neq 0 \qquad (III.1)$$

de lo cual se concluye :

$$(\underline{A} - \lambda\underline{I}) = 0$$

para que exista solución diferente de la trivial :

$$|\underline{A} - \lambda\underline{I}| = 0 \qquad (III.2)$$

de la ecuación anterior se obtiene el polinomio característico de la matriz A:

$$P(\lambda) = 0$$

Las raíces de dicho polinomio son los valores característicos y substituyendo cada valor en (III.1) se obtiene un sistema de ecuaciones que permiten obtener los vectores característicos $\underline{X}$. Al resolver dichos sistemas de ecuaciones hay que asignar un valor arbitrario a una de las incógnitas.

Ejemplo

Encontrar los valores y vectores característicos de la matriz :

$$A = \begin{bmatrix} 5 & -2 \\ -2 & 1 \end{bmatrix}$$

Sol.

$|A| = 5 - 4 = 1 \neq 0 \Longrightarrow$ sí hay solución aplicando el método descrito:

$$|\underline{A} - \lambda \underline{I}| = \begin{vmatrix} 5-\lambda & -2 \\ -2 & 1-\lambda \end{vmatrix} = 0$$

$$(5-\lambda)(1-\lambda) - 4 = 0$$

$$5 - 5\lambda - \lambda + \lambda^2 - 4 = 0$$

$$P(\lambda) = \lambda^2 - 6\lambda + 1 = 0$$

- resolviendo $P(\lambda)$:

$$\lambda = \frac{6 \pm \sqrt{36 - 4}}{2} = \frac{6 \pm \sqrt{32}}{2}$$

$$\lambda = 3 \pm \sqrt{8}$$

$$\lambda_1 = 5.828$$

$$\lambda_2 = 0.171$$

- obteniendo el vector característico para λ_1 :

$$\begin{bmatrix} 5 - \lambda_1 & -2 \\ -2 & 1 - \lambda_1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} -0.828 & -2 \\ -2 & -4.828 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$-0.828\, X_1 - 2\, X_2 = 0$$

$$-2\, X_1 - 4.828\, X_2 = 0$$

$$X_1 = 1$$

$$X_2 = -\frac{.828}{2} = -0.414$$

$$\underline{X}_1 = \begin{bmatrix} 1 \\ -0.414 \end{bmatrix}$$

- obteniendo el vector característico para λ_2 :

$$\begin{bmatrix} 5 - \lambda_2 & -2 \\ -2 & 1 - \lambda_2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 4.829 & -2 \\ -2 & 0.829 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$X_2 = \frac{4.829}{2} = 2.414$$

$$\underline{X_2} = \begin{bmatrix} 1 \\ 2.414 \end{bmatrix}$$

b) Método de Krylov

Este método se emplea para obtener los coeficientes del polinomio característico en forma numérica.

Para ello se aplica el Teorema de Cayley - Hamilton que dice :

Si $P(\lambda) = 0$

$\Longrightarrow$ $P(\underline{A}) = 0$

donde λ **son los valores característicos de la matriz** $\underline{A}$.

Teniendo el polinomio característico :

$$P(\lambda) = A_n \lambda^n + A_{n-1} \lambda^{n-1} + \ldots + A_o = 0, \; A_n \neq 0$$

se transforma $P(\lambda)$ **de tal forma que el coeficiente enésimo sea unitario :**

$$F(\lambda) = \lambda^n + b_{n-1} \lambda^{n-1} + \ldots + b_o = 0 \qquad (III.3)$$

donde :

$$b_{n-1} = \frac{A_{n-1}}{A_n}$$

$$b_{n-2} = \frac{A_{n-2}}{A_n}$$

$$\vdots$$

$$b_o = \frac{A_o}{A_n}$$

aplicando el T. de Cayley - Hamilton en (III.3) :

$$F(\underline{A}) = \underline{A}^n + b_{n-1}\,\underline{A}^{n-1} + \ldots + b_1\,\underline{A} + b_0\,\underline{I} = 0 \qquad (III.4)$$

multiplicando (III.4) por un vector $\underline{Y}$, tal que $\underline{Y} \neq \underline{0}$:

$$F(\underline{A})\,\underline{Y} = A^n\,\underline{Y} + b_{n-1}\,\underline{A}^{n-1}\,\underline{Y} + \ldots + b_1\,A\underline{Y} + b_0\,\underline{Y} = 0 \qquad (III.5)$$

lo cual da un sistema de ecuaciones con incógnitas $b_1, b_2, \ldots, b_n$ que representan los coeficientes de la ecuación característica. Se resuelve dicho sistema y los valores obtenidos se substituyen en (III.3).

Ejemplo

Si, $\underline{A} = \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix}$, encontrar su ecuación característica empleando el método de Krylov.

Sol.

Sea : $\underline{Y} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

se obtendrán los siguientes términos de la ecuación (III.5) :

$$\underline{A}^3\,\underline{Y}$$
$$\underline{A}^2\,\underline{Y}$$
$$\underline{A}\,\underline{Y}$$

los cuales son :

$$\underline{A}\,\underline{Y} = \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ -2 \\ 0 \end{bmatrix}$$

$$\underline{A}^2\,\underline{Y} = \underline{A}\,\underline{A}\,\underline{Y} = \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 5 \\ -2 \\ 0 \end{bmatrix} = \begin{bmatrix} 29 \\ -16 \\ 2 \end{bmatrix}$$

$$\underline{A}^3\,\underline{Y} = A\,\underline{A}^2\,\underline{Y} = \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 29 \\ -16 \\ 2 \end{bmatrix} = \begin{bmatrix} 177 \\ -108 \\ 18 \end{bmatrix}$$

substituyendo en (iii.5) :

$$\begin{bmatrix} 177 \\ -108 \\ 18 \end{bmatrix} + \begin{bmatrix} 29 \\ -16 \\ 2 \end{bmatrix} b_2 + \begin{bmatrix} 5 \\ -2 \\ 0 \end{bmatrix} b_1 + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} b_o = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 29 & 5 & 1 \\ -16 & -2 & 0 \\ 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} b_2 \\ b_1 \\ b_o \end{bmatrix} = \begin{bmatrix} -177 \\ 108 \\ -18 \end{bmatrix}$$

sistema que se resuelve empleando alguno de los métodos antes vistos :

$$\left[\begin{array}{ccc|c} 29 & 5 & 1 & -177 \\ 16 & -2 & 0 & 108 \\ \textcircled{2} & 0 & 0 & -18 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 0 & 5 & 1 & 84 \\ 0 & \textcircled{-2} & 0 & -36 \\ 1 & 0 & 0 & -9 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 0 & 0 & 1 & -6 \\ 0 & 1 & 0 & 18 \\ 1 & 0 & 0 & -9 \end{array}\right]$$

$$\begin{bmatrix} b_2 \\ b_1 \\ b_0 \end{bmatrix} = \begin{bmatrix} -9 \\ 18 \\ -6 \end{bmatrix}$$

por lo que :

$$P(\lambda) = \lambda^3 - 9\lambda^2 + 18\lambda - 6 = 0$$

c) Método de Jacobi para obtener el mayor valor característico .

Por definición se tiene que un valor y vector característico cumplen la siguiente relación :

$$\underline{A}\,\underline{X} = \lambda\,\underline{X}, \quad \underline{X} \neq 0$$

Para aplicar el método se toma un valor aproximado de $\underline{X} = \underline{X_0}$ por el - que se multiplica la matriz $\underline{A}$, de dicho producto se extrae como factor común - el mayor elemento obteniéndose la nueva aproximación del vector característico y así sucesivamente, o sea :

$$\underline{A}\,\underline{X_0} = \lambda_1 \underline{X_1}$$

$$\underline{A}\,\underline{X_1} = \lambda_2 \underline{X_2}$$

$$\vdots$$

$$\underline{A}\,\underline{X_n} = \lambda_{n+1} \underline{X_{n+1}}$$

el método se detiene cuando :

$$|\lambda_{n+1} - \lambda_n| < \varepsilon$$

Ejemplo

$$A^2Y = AAY = \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 29 \\ -16 \\ 2 \end{bmatrix}$$

Obtener el mayor valor característico de la matriz:

$$A = \qquad A^3Y = AA^2Y = \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 29 \\ -16 \\ 2 \end{bmatrix} = \begin{bmatrix} 177 \\ -109 \\ 18 \end{bmatrix}$$

Sol.

La fórmula iterativa está dada por :

$$\begin{bmatrix} 177 \\ -109 \\ 18 \end{bmatrix} + \begin{bmatrix} 5 & -2 & 0 \\ -2 & 3 & -1 \\ 0 & -1 & 1 \end{bmatrix} b_1 \begin{bmatrix} X_1 \\ X_2 \\ 0 \end{bmatrix} = b_3 \lambda \begin{bmatrix} X_1 \\ X_2 \\ 0 \end{bmatrix}$$

sea:

$$\begin{bmatrix} 29 & 5 & 1 \\ -16 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix} \underline{X_0} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} b_1 + \begin{bmatrix} 177 \\ -109 \\ 18 \end{bmatrix} b_0$$

$$\begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ -2 \end{bmatrix} = 5 \begin{bmatrix} 1 \\ -.4 \end{bmatrix}$$

$$\begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ -.4 \end{bmatrix} = \begin{bmatrix} 5.8 \\ -3.2 \end{bmatrix} = 5.8 \begin{bmatrix} 1 \\ -.551 \end{bmatrix}$$

$$\begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ -.551 \end{bmatrix} = \begin{bmatrix} 6.102 \\ -3.653 \end{bmatrix} = 6.102 \begin{bmatrix} 1 \\ -.598 \end{bmatrix}$$

$$\begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ -.598 \end{bmatrix} = \begin{bmatrix} 6.196 \\ 3.794 \end{bmatrix} = 6.196 \begin{bmatrix} 1 \\ -.612 \end{bmatrix}$$

$$\begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ -.612 \end{bmatrix} = \begin{bmatrix} 6.224 \\ -3.836 \end{bmatrix} = 6.224 \begin{bmatrix} 1 \\ -.616 \end{bmatrix}$$

$$\begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ -.616 \end{bmatrix} = \begin{bmatrix} 6.232 \\ -3.848 \end{bmatrix} = 6.232 \begin{bmatrix} 1 \\ -.617 \end{bmatrix}$$

$$\lambda \cong 6.23$$

$$\underline{X} = \begin{bmatrix} 1 \\ -0.617 \end{bmatrix}$$

d) Método de Jacobi para obtener el menor valor característico.

Sabemos que la ecuación :

$$\underline{A}\,\underline{X} = \lambda\,\underline{X}, \quad \underline{X} \neq 0 \qquad \text{(III.6)}$$

siempre converge al mayor λ aplicando Jacobi. Multiplicando (III.6) por $\underline{A}^{-1}$:

$$\underline{A}^{-1}\,\underline{A}\,\underline{X} = \underline{A}^{-1}\,\lambda\,\underline{X}$$

$$\underline{X} = \lambda\,\underline{A}^{-1}\,\underline{X}$$

$$\underline{A}^{-1}\,\underline{X} = \frac{1}{\lambda}\,\underline{X} \qquad \text{(III.7)}$$

$$\underline{A}^{-1}\,\underline{X} = \lambda'\,\underline{X} \qquad \text{(III.8)}$$

al igual que (III.6), (III.8) convergirá al mayor $\lambda' = \frac{1}{\lambda}$, de lo cual se desprende que converge al menor λ, donde :

$$\lambda = \frac{1}{\lambda'}$$

este proceso se detiene cuando :

$$| \lambda'_{n+1} - \lambda'_n | < \varepsilon$$

Ejemplo

Encontrar el menor valor característico de :

$$\underline{A} = \begin{bmatrix} 5 & -2 \\ -2 & 3 \end{bmatrix}$$

Sol.

La inversa de $\underline{A}$ será :

$$\underline{A}^{-1} = \frac{\underline{A}^{+}}{|A|} = \frac{1}{11}\begin{bmatrix} 3 & 2 \\ 2 & 5 \end{bmatrix}$$

la fórmula recursiva :

$$\underline{A}^{-1}\underline{X} = \lambda'\underline{X}$$

$$\frac{1}{11}\begin{bmatrix} 3 & 2 \\ 2 & 5 \end{bmatrix}\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \lambda'\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$$

sea $\underline{X}_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$$\frac{1}{11}\begin{bmatrix} 3 & 2 \\ 2 & 5 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{11}\begin{bmatrix} 3 \\ 2 \end{bmatrix} = \frac{3}{11}\begin{bmatrix} 1 \\ 0.66 \end{bmatrix}$$

$$\frac{1}{11}\begin{bmatrix} 3 & 2 \\ 2 & 5 \end{bmatrix}\begin{bmatrix} 1 \\ .66 \end{bmatrix} = \frac{1}{11}\begin{bmatrix} 4.32 \\ 5.3 \end{bmatrix} = \frac{5.3}{11}\begin{bmatrix} .815 \\ 1 \end{bmatrix}$$

$$\frac{1}{11}\begin{bmatrix}3 & 2\\ 2 & 5\end{bmatrix}\begin{bmatrix}.815\\ 1\end{bmatrix} = \frac{1}{11}\begin{bmatrix}4.445\\ 6.63\end{bmatrix} = \frac{6.63}{11}\begin{bmatrix}.67\\ 1\end{bmatrix}$$

$$\frac{1}{11}\begin{bmatrix}3 & 2\\ 2 & 5\end{bmatrix}\begin{bmatrix}.67\\ 1\end{bmatrix} = \frac{1}{11}\begin{bmatrix}4.01\\ 6.34\end{bmatrix} = \frac{6.34}{11}\begin{bmatrix}.632\\ 1\end{bmatrix}$$

$$\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots$$

$$\frac{1}{11}\begin{bmatrix}3 & 2\\ 2 & 5\end{bmatrix}\begin{bmatrix}.621\\ 1\end{bmatrix} = \frac{1}{11}\begin{bmatrix}3.854\\ 6.236\end{bmatrix} = \frac{6.236}{11}\begin{bmatrix}.618\\ 1\end{bmatrix}$$

$$\therefore \lambda' = \frac{6.236}{11}$$

$$\Rightarrow \lambda = \frac{11}{6.236} = 1.763$$

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL

APENDICE
ALGEBRA MATRICIAL

VICTOR GEREZ GREISER

ABRIL DE 1976.

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels: 521-40-23 521-73-35 5123-123

A.1 Clasificacion de matrices

A.2 Operaciones con matrices

A.3 Ecuaciones lineales simultaneas y operaciones elementales

A.4 Vectores y espacios vectoriales

A.5 Transformaciones lineales, valores y vectores caracteristicos

A.6 Funciones de matrices cuadradas

A.7 Problemas

El objeto de esta serie de cuatro apéndices es resumir brevemente, conceptos de la teoría de matrices, ecuaciones diferenciales, métodos operacionales y funciones especiales que se requieren para el estudio del material de este libro. No pretenden sustituir textos especializados sobre estos tópicos, sino solamente recordar al lector conceptos que debe haber estudiado en otra parte. Algunos teoremas solamente se enuncian, mientras que en otros se incluye la demostración, cuando se considera que ésta corresponde al nivel del presente curso o es importante para entender el teorema.

En este apéndice repasamos algunos aspectos básicos de la teoría de matrices, de ecuaciones lineales y de espacios vectoriales. Definimos los principales conceptos de la teoría de matrices. Establecemos una relación entre esta teoría y la de sistemas de ecuaciones lineales algebraicas. Señalamos la utilidad de la forma normal escalonada y del rango de una matriz para la determinación de la solución de sistemas de ecuaciones lineales algebraicas. Estudiemos los conceptos de valores y vectores característicos de una matriz cuadrada y la diagonalización de una matriz cuadrada que nos permiten introducir el cálculo de $exp[A]t$. Finalmente, el teorema de Cayley-Hamilton nos sirve para calcular funciones de matrices.

A.1 Clasificación de matrices

Empecemos definiendo el concepto de matriz.

Una *matriz* es un arreglo rectangular de elementos que pueden ser: números reales, números complejos, fracciones racionales de polinomios en s, funciones del tiempo, operadores, etc.

A continuación damos algunos ejemplos de matrices:

$$\text{(A.1.1)} \quad \begin{bmatrix} 3 & -6 \\ -6 & 2 \end{bmatrix}, \quad \begin{bmatrix} 1-3j \\ 2 \\ 6-j \end{bmatrix}, \quad \frac{1}{s+3} \quad \frac{2}{s-1}$$

$$\begin{bmatrix} t \operatorname{sen} 3t & 2 \exp t \\ 3 & 2 \operatorname{sen} 3t \\ 4 \cos 2t & 0 \end{bmatrix}, \quad \begin{bmatrix} D^2 + 2D + 1, & 0 \\ 0, & D^{-1} + D^3 + 8 \end{bmatrix}$$

En la segunda matriz j representa a $\sqrt{-1}$; es decir, el coeficiente de j es la parte imaginaria de los números complejos. En la última matriz el operador diferencial D^n representa la enésima derivada, o sea

$$D^n = \frac{d^n \ldots}{dt^n},$$

y

$$D^{-1} = \int_{-\infty}^{\infty} \ldots dt$$

Una matriz está compuesta por renglones (líneas horizontales) y por columnas (líneas verticales). Se dice que una matriz con "p" renglones y "q" columnas, es de *orden* $p \times q$. Con a_{rs} designaremos al elemento del renglón r y la columna s de la matriz $[A]$, escribiendo a la matriz

$$[A] = \begin{bmatrix} a_{11} & a_{12} \ldots\ldots\ldots a_{1q} \\ a_{21} & a_{22} \ldots\ldots\ldots a_{2q} \\ \text{- - - - - - - - - - - - - - - -} \\ \text{- - - - - - - - - - - - - - - -} \\ a_{p1} & a_{p2} \ldots\ldots\ldots a_{pq} \end{bmatrix}$$

en forma más compacta como $[A] = [a_{rs}]$.

Los órdenes de las matrices que se dieron como ejemplos en (A.1.1) son, respectivamente, 2×2, 3×1, 1×2, 3×2 y 2×2.

Los elementos a_{ii} de una matriz forman la *diagonal principal* de dicha matriz.

A continuación clasificamos diversos tipos de matrices:

a) *Matriz cuadrada:*

Recibe este nombre toda matriz con igual número de columnas que de renglones. Ejemplos de matrices cuadradas son la primera y la última matriz de (A.1.1).

b) *Matriz columna:*

Toda matriz de una sola columna recibe el nombre de matriz columna y la representaremos con $A\rceil$. La segunda matriz del grupo (A.1.1) es de columna.

c) *Matriz de renglón:*

Una matriz de un solo renglón recibe el nombre de matriz renglón y se *denota* con A. La tercera matriz del grupo (A.1.1) es de este tipo.

d) *Matriz diagonal:*

Si los elementos a_{ij} de una matriz son nulos para $i \neq j$, la matriz se conoce con el nombre de matriz diagonal. La matriz de los operadores diferenciales D en (A.1.1) es diagonal.

e) *Matriz unitaria:*

Se conoce con este nombre una matriz diagonal cuadrada de cualquier orden para la cual $a_{ii} = 1$ para todo valor del índice i. Se representa con $\lceil I \rceil$ y está dada por:

$$\left[I\right] = \begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{matrix}$$

f) *Matriz nula:*

Una matriz con todos sus elementos iguales a cero recibe este nombre.

g) *Matriz simétrica*

Si los elementos de la matriz satisfacen la relación $a_{ij} = a_{ji}$ la matriz es simétrica. La primera matriz de la serie (A.1.1) es simétrica.

h) *Matriz transpuesta*

La matriz transpuesta $\left[A\right]^T$ de la matriz $\left[A\right] = \left[a_{rs}\right]$ es, por definición, la matriz que se obtiene colocando como i-ésima columna de $\left[A\right]^T$ al i-ésimo renglón de $\left[A\right]$; es decir, $\left[A\right]^T = \left[a_{sr}\right]$.

Por esta definición, la matriz transpuesta de una matriz de orden $m \times n$ será de orden $n \times m$. Es fácil demostrar que el transpuesto de una matriz simétrica es la propia matriz.

A.2 Operaciones con matrices

Empecemos definiendo la *igualdad* entre matrices:

Las dos matrices

(A.2.1) $\left[A\right] = (a_{rs})$ y $\left[B\right] = (b_{rs})$

son iguales si son del mismo orden y los elementos correspondientes de las matrices son iguales, o sea $a_{rs} = b_{rs}$ para todo valor de r y s.

Por ejemplo, la siguiente igualdad:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} = \begin{bmatrix} t & 0 \\ \sqrt{t^2} & -6 \\ 0 & t^2 \end{bmatrix}$$

implica que $a_{11} = t$, $a_{12} = 0$, $a_{21} = \sqrt{t^2}$ $a_{22} = -6$, $a_{31} = 0$, $a_{32} = t^2$.

Cuando dos matrices son del mismo orden se puede definir una tercera matriz llamada la suma de las dos matrices, o matriz suma. La matriz suma tiene elementos que son la suma de los elementos correspondientes de las matrices: es decir, dadas las matrices

$$\left[A\right] = (a_{rs}), \quad \left[B\right] = (b_{rs})$$

la matriz suma

$$\left[C\right] = \left[A\right] + \left[B\right]$$

está dada por:

$$\left[C\right] = (c_{rs}) \qquad c_{rs} = a_{rs} + b_{rs}$$

Desde luego, la suma de matrices es *conmutativa*, o sea:

(A.2.2) $[A] + [B] = [B] + [A]$

EJEMPLO A.2a Sume las dos matrices y muestre que la suma de matrices conmuta

$$\begin{bmatrix} 3 & -1 \\ 2 & -3 \\ -4 & 2 \end{bmatrix} + \begin{bmatrix} 0 & 5 \\ -9 & 3 \\ 2 & -3 \end{bmatrix} = \begin{bmatrix} 3 & 4 \\ -7 & 0 \\ -2 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 5 \\ -9 & 3 \\ 2 & -3 \end{bmatrix} + \begin{bmatrix} 3 & -1 \\ 2 & -3 \\ -4 & 2 \end{bmatrix}$$

EJEMPLO A.2b Sume las matrices

$$\begin{bmatrix} 3 & 2 \\ 4 & -1 \end{bmatrix}; \begin{bmatrix} 0 & -3 & 2 \\ 4 & 5 & 0 \end{bmatrix}$$

Solución:

Estas matrices no se pueden sumar por no ser del mismo orden.

De manera similar se define la resta de matrices.
La *multiplicación de una matriz por un escalar* es, por definición, una segunda matriz cuyos elementos son el producto de los elementos de la matriz original por el escalar; o sea, si

$$[A] = (a_{rs})$$

entonces

$$k\left[A\right] = (k\, a_{rs})$$

También esta operación es *conmutativa*

$$k\left[A\right] = \left[A\right]k$$

EJEMPLO A.2c Calcule el siguiente producto de un escalar por una matriz

$$5\begin{bmatrix} 4t & 2t+1 \\ -3 & 2t^2 \end{bmatrix} = \begin{bmatrix} 20t & 10t+5 \\ -15 & 10t^2 \end{bmatrix} \tag{A.2.3}$$

Se define el negativo de una matriz como el producto de la matriz por el escalar -1, o sea:

$$-\left[A\right] = -1 \cdot \left[A\right] \tag{A.2.4}$$

Dos *matrices* $\left[A\right]$ y $\left[B\right]$ se pueden *multiplicar*, si y solamente si, el número de columnas de la primera es igual al número de renglones de la segunda. Dos matrices con esta propiedad se llaman conformables. En este caso el elemento *(i, j)* del producto se calcula empleando la relación (A.2.5).

$$c_{ij} = \sum_{k=1}^{n} a_{ik}\, b_{kj}; \qquad \begin{matrix} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, r \end{matrix} \tag{A.2.5}$$

De acuerdo con esta fórmula, si el primer factor del producto, la matriz $[A]$, es de orden $m \times n$ y el segundo factor, la matriz $[B]$, es de orden $n \times r$, el producto, la matriz $[C]$, es de orden $m \times r$.

EJEMPLO A.2d El negativo de la matriz del ejemplo anterior es:

$$\begin{bmatrix} -4t & -2t - 1 \\ 3 & -2t^2 \end{bmatrix}$$

En general el producto de dos matrices no es *conmutativo*, es decir

$$[A][B] \neq [B][A] \tag{A.2.6}$$

EJEMPLO A.2e Calcule el producto de las siguientes matrices:

$$\begin{bmatrix} 0 & 2 & -3 \\ 6 & 4 & -1 \end{bmatrix} \cdot \begin{bmatrix} 3 & 2 \\ -9 & -1 \\ 0 & -3 \end{bmatrix}$$

$$= \begin{matrix} 0 \times 3 + 2 \times (-9) & 0 \times 2 + 2 \times (-1) \\ -3 \times 0 & +(-3) \times (-3) \\ 6 \times 3 + 4 \times (-9) & 6 \times 2 + 4 \times (-1) \\ -1 \times 0 & +(-1) \times (-3) \end{matrix}$$

$$= \begin{bmatrix} -18 & 70 \\ -27 & 11 \end{bmatrix}$$

Designemos con $\underline{A}_i$ la matriz renglón formada por el i-ésimo renglón de la matriz $[A]$ y con $B]^j$ la matriz columna formada por la j-ésima columna de $[B]$; entonces el elemento c_{ij} de la matriz producto

$$[C] = [A] \cdot [B]$$

puede calcularse también empleando la relación (A.2.7),

$$c_{ij} = \underline{A}_i \, B]^j$$

El lector puede verificar fácilmente que para cualquier terna de matrices conformables se cumplen las siguientes propiedades:

$$[D]\begin{bmatrix} A \\ A \\ A \end{bmatrix} + \begin{bmatrix} B \\ B \\ B \end{bmatrix} [C] = \begin{bmatrix} A \\ D \\ A \end{bmatrix} \begin{bmatrix} C \\ A \\ B \end{bmatrix} + \begin{bmatrix} B \\ D \\ C \end{bmatrix} \begin{bmatrix} C \\ B \end{bmatrix}$$

EJEMPLO A.2f Calcule el producto de las siguientes matrices y observe que el producto en orden inverso no puede calcularse

$$\begin{bmatrix} 0 & 3 \\ j & 0 \\ 3 & -3-j \end{bmatrix} \begin{bmatrix} 1 \\ j \end{bmatrix} = \begin{bmatrix} 3j \\ j \\ 4-3j \end{bmatrix}$$

Si se trata de efectuar la multiplicación invirtiendo el orden de los factores, habría que multiplicar una matriz 2×1 por una matriz 3×2, cosa que no se puede hacer por no ser conformables.

El ejemplo anterior ilustra la no conmutatividad del producto de dos matrices. Sin embargo, existen casos excepcionales en los que el producto de dos matrices conmuta, como vemos en el siguiente ejemplo:

EJEMPLO A.2g Muestre que los siguientes productos sí conmutan.

$$\begin{bmatrix} s & -2 \\ \frac{4}{s+1} & \frac{6}{s} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} s & -2 \\ \frac{4}{s+1} & \frac{6}{s} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s & -2 \\ \frac{4}{s+1} & \frac{6}{s} \end{bmatrix}$$

$$\begin{bmatrix} 8 & 3 \\ 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & -3 \\ -5 & 8 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & -3 \\ -5 & 8 \end{bmatrix} \begin{bmatrix} 8 & 3 \\ 5 & 2 \end{bmatrix}$$

En el primer ejemplo el producto de dos matrices conmuta por tratarse del producto $[A][I]$ que siempre satisface la relación:

$$[A][I] = [I][A] = [A]$$

En el segundo ejemplo el producto es conmutativo por tratarse del producto de una matriz por su inverso, como se verá posteriormente.

Determinante de una matriz

A continuación definiremos el determinante de una matriz cuadrada de $n \times n$ como un solo número que se representa con $det\left[A\right]$ o $|A|$. El determinante de una matriz de orden n se definirá en función del determinante de una matriz de orden $n-1$. Además, el determinante de una matriz de orden 1×1 es el elemento

$$det\left[A\right] = a_{11}$$

Para establecer la definición de determinante tendremos que introducir primero los conceptos de *menor* y cofactor.

Si en una matriz se eliminan el renglón i y la columna j, se obtiene una nueva matriz cuyo determinante se llama menor del elemento a_{ij} y que se denota con el símbolo M_{ij}.

Se llama *cofactor* del elemento a_{ij} al número

$$\alpha_{ij} = (-1) \quad |M_{ij}|$$

EJEMPLO A.2h En la matriz

$$\left[A\right] = \begin{bmatrix} 0 & -1 & -4 \\ 1 & 7 & 3 \\ -2 & 3 & 1 \end{bmatrix}$$

el *menor* del elemento $a_{12} = -1$, es el determinante

$$\begin{vmatrix} 1 & 3 \\ -2 & 1 \end{vmatrix} = +7$$

y su *cofactor* es $(-1)^{1-2}(+7) = -7$.

El valor del determinante de una matriz cuadrada de orden $n \times n$ se puede evaluar empleando cualquiera de las siguientes fórmulas

$$|A| = \sum_{j=1}^{n} a_{ij}\, \alpha_{ij} \qquad \textit{para cualquier} \quad i$$

$$|A| = \sum_{j=1}^{n} a_{ij}\, \alpha_{ij} \qquad \textit{para cualquier } j$$

Aplíquemos cualquiera de las fórmulas anteriores para calcular el determinante de una matriz de orden 2×2.

Sea

$$[A] = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

Empleando (A.2.10) y teniendo presente que

$$\alpha_{11} = a_{22}$$

$$\alpha_{12} = -a_{21}$$

obtenemos

$$|A| = a_{11}\, a_{22} - a_{12}\, a_{21}$$

para el determinante de una matriz 2×2.

EJEMPLO A.2i Obtenga el valor del siguiente determinante.

$$\begin{vmatrix} 0 & -1 & -4 \\ 1 & 7 & 3 \\ -2 & 3 & 1 \end{vmatrix}$$

$$= 0 \cdot \begin{vmatrix} 7 & 3 \\ 3 & 1 \end{vmatrix} - (-1) \begin{vmatrix} 1 & 3 \\ -2 & 1 \end{vmatrix} + 4 \begin{vmatrix} 1 & 7 \\ -2 & 3 \end{vmatrix}$$

$$= 0 \cdot (7 - 9) + 1(6 + 1) + 4(3 + 14)$$

$$= 0 + 7 + 68 = 75$$

A continuación introducimos una importante definición:

Si el determinante de una matriz cuadrada es nulo, se dice que la matriz es *singular*.

Sean α_{ij} los cofactores de la matriz cuadrada $[A]$. Se llama *matriz adjunta* $adj[A] = adj[a_{ij}]$ a la matriz cuyos elementos son α_{ji}. (Nótese la inversión en los índices de los cofactores.)

EJEMPLO A.2j Calcule el adjunto de la matriz:

$$[A] = \begin{bmatrix} 1 & 3 & -1 \\ 2 & 5 & 0 \\ 6 & +3 & 2 \end{bmatrix}$$

Los factores son:

$$\alpha_{11} = +10 \qquad \alpha_{12} = -4 \qquad \alpha_{13} = -21$$
$$\alpha_{21} = -9 \qquad \alpha_{22} = 8 \qquad \alpha_{23} = +15$$
$$\alpha_{31} = 5 \qquad \alpha_{32} = -2 \qquad \alpha_{33} = -1$$

y finalmente el $adj[A]$ es:

$$adj[A] = \begin{bmatrix} 10 & -9 & 5 \\ -4 & 8 & -2 \\ -21 & 15 & -1 \end{bmatrix}$$

La matriz adjunta satisface las siguientes relaciones:

(A.2.14) $$adj[A] \cdot [A] = [A] \cdot adj[A] = |A|[I]$$

en donde $[I]$ es la matriz unidad.

De la ec. (A.2.14) se obtiene inmediatamente:

$$[A]\frac{adj[A]}{|A|} = \frac{adj[A]}{|A|}[A]$$

Observamos que la matriz $[A]$ pre-o postmultiplicada por

$$\frac{adj[A]}{|A|}$$

nos da la matriz unitaria $[I]$. A la matriz

$$\frac{adj[A]}{|A|} = [A]^{-1} \tag{A.2.15}$$

se le llama el *inverso de la matriz* $[A]$, y se designa con $[A]^{-1}$; por lo tanto:

$$[A]\,[A] = [A][A]^{-1} = [I] \tag{A.2.16}$$

Como en (A.2.12) se divide entre $|A|$ al inverso de $[A]$ solamente existe si

$$det\ [A] \neq 0$$

o sea si la matriz es *no singular*.

A.3 Ecuaciones lineales simultaneas y operaciones elementales

El sistema de ecuaciones lineales simultáneas:

$$\begin{aligned}
a_{11}\ x_1 &+ a_{12}\ x_2 + \cdots + a_{1n}\ x_n = b_1 \\
a_{21}\ x_1 &+ a_{22}\ x_2 + \cdots + a_{2n}\ x_n = b_2 \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
a_{m1}\ x_1 &+ a_{m2}\ x_2 + \cdots + a_{mn}\ x_n = b_m
\end{aligned} \tag{A.3.1}$$

en el cual los números a_{ij}, y b_i $(i = 1,2,\ldots, m;\ j = 1,2,\ldots, n)$

son constantes conocidas y las $x_1, x_2, \ldots x_n$ son incógnitas, se puede escribir en forma matricial como

$$(A.3.2) \quad \left[A\right] \; x\right] = b\right]$$

si se definen las siguientes matrices

$$\left[A\right] = \begin{matrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{12} & a_{22} & \cdots & a_{2n} \\ \text{- - - - - - - - - - - - - - - - - -} & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{matrix} = (a_{ij})$$

$$x\right] = \begin{matrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{matrix}$$

$$b\right] = \begin{matrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{matrix}$$

Nótese que el número de incógnitas n y el número de ecuaciones m no son necesariamente iguales.

Ilustraremos primero con un ejemplo una forma sistemática para encontrar la solución de un sistema de ecuaciones lineales algebraicas simultáneas:

EJEMPLO A.3a Dado el sistema de ecuaciones algebraicas:

$$2x_1 - x_2 - 3x_3 = 2$$

$$x_1 + 2x_2 + x_3 = 1$$

Encuentre x_1, x_2 y x_3.

Solución:

De acuerdo con la notación usada en (A.3.1) $m = 2$, $n = 3$. Multipliquemos la segunda ecuación por 2

$$2x_1 - x_2 - 3x_3 = 2$$

(A.3.4)

$$2x_1 + 4x_2 + 2x_3 = 2$$

Restemos la segunda de la primera ecuación para eliminar de ella a x_1

$$2x_1 - x_2 - 3x_3 = 2$$

(A.3.5)

$$-5x_2 - 5x_3 = 0$$

Dividiendo entre 5 la segunda ecuación y restándola de la primera nos permite eliminar x_2 de la primera ecuación

$$2x_1 - 2x_3 = 2$$

(A.3.6)

$$-x_2 - x_3 = 0$$

y finalmente diviendo la primera entre 2 y multiplicando la segunda por menos uno tenemos:

$$x_1 - x_3 = 1$$

(A.3.7)

$$x_2 + x_3 = 0$$

La solución del sistema es:

$$x_1 = 1 + x_3$$

$$x_2 = -x_3$$

A la incógnita x_3 puede dársele cualquier valor; una vez fijo éste pueden calcularse x_1 y x_2. Debemos hacer notar que la solución de este sistema *no es única*, puesto que diferentes valores de x_3 nos darán diferentes x_1 y x_2.

Definiendo las matrices:

$$[A] = \begin{bmatrix} 2 & -1 & -3 \\ & & \\ 1 & 2 & 1 \end{bmatrix}$$

$$b] = \begin{bmatrix} 2 \\ \\ 1 \end{bmatrix}$$

$$x] = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

(A.3.3) puede escribirse como

(A.3.2) $$[A]x] = b]$$

Introduzcamos ahora un importante concepto: *la matriz aumentada* definida por (A.3.8)

(A.3.8) $$[\,[A] \quad b\,]$$

o sea a la matriz de coeficientes a_{ij} se le agrega como última columna la columna $b]$. Esta matriz juega un papel importante en la solución sistemática de sistemas de ecuaciones diferenciales lineales.

Usando la notación matricial (A.3.8) los pasos que llevaron a la solución del sistema del ejemplo A.3.a quedan:

(A.3.3') $$\begin{bmatrix} 2 & -1 & -3 & 2 \\ 1 & 2 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & -1 & -3 & 2 \\ 2 & 4 & 2 & 2 \end{bmatrix}$$

(A.3.4') $$\begin{bmatrix} 2 & -1 & -3 & 2 \\ 0 & -5 & -5 & 0 \end{bmatrix}$$

(A.3.6') $\begin{bmatrix} 2 & 0 & -2 & 2 \\ 0 & -1 & -1 & 0 \end{bmatrix}$

(A.3.7') $\begin{bmatrix} 1 & 0 & -1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$

Las operaciones que permitieron llegar a (A.3.7') se llaman operaciones elementales de renglón y la forma (A.3.7') se llama forma *normal escalonada*. A continuación formalizamos estas ideas.

Operaciones elementales

Se llaman operaciones elementales las siguientes operaciones aplicadas a los renglones (o columnas) de una matriz.

1. Intercambiar dos renglones (o columnas).
2. Multiplicar un renglón (o columna) por un número diferente de cero.
3. Añadir a los elementos de un renglón (o columna) k veces los elementos de otro renglón (o columna).

Reducción de una matriz a forma normal escalonada por renglones.

Antes de entrar a definir esta forma recordemos que *un vector unitario de columna* $e\rfloor_i$ tiene todos sus elementos nulos, menos el i-ésimo que es igual a la unidad.

Una matriz está en forma *normal escalonada* por renglones, si satisface las siguientes condiciones:

a) Ciertas columnas denominadas $c_1, c_2, \ldots, c_k$ son los vectores unitarios $e\rfloor_1, e\rfloor_2, \ldots e\rfloor_k$.

b) Estas columnas deben estar precisamente en el orden señalado. La primera columna entre el conjunto c_i debe ser $e\rfloor_1$, la segunda $e\rfloor_2$, etc.

c) Si una columna está a la izquierda de la columna c_1 tiene todos los elementos nulos; si está entre la c_1 y la c_{i+1} todos sus elementos después del "i-ésimo" son nulos, y toda columna situada después de la c_k tiene sus elementos posteriores al "k-ésimo" iguales a cero.

La definición anterior implica:

1) Los elementos en el triángulo inferior de posición (i, j), (i-ésimo renglón, j-ésima columna) donde $j < i$ son nulos.

2) Todo renglón posterior al "k-ésimo" es nulo. Existen k renglones diferentes de cero.

3) El primer elemento no nulo de cada renglón es 1.

Un ejemplo de una matriz en forma normal escalonada por renglones, es el siguiente:

$$\begin{bmatrix} 0 & 1 & 0 & 3 & 0 & 2 \\ 0 & 0 & 1 & 2 & 0 & 6 \\ 0 & 0 & 0 & 0 & 1 & -9 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Cualquier matriz rectangular se puede reducir a forma normal escalonada por renglones empleando operaciones elementales de renglones usando el siguiente procedimiento: Se trata primero de tener un uno en la posición (1,1). Si el elemento que está en esta posición es diferente de cero, multiplique el primer renglón por el inverso de este elemento para obtener un uno en la posición $(k,1)$. Si el elemento que está inicialmente en la posición (1, 1) es nulo, se debe intercambiar este renglón con algún otro cuyo primer elemento no sea nulo, y después realizar la operación anterior. Si todos los elementos de la primera columna son nulos, entonces se debe seguir este mismo procedimiento con la siguiente columna a su derecha que no sea nula. Una vez obtenido el vector unitario $e_{\omega}\,]_1$ en alguna columna, se procede de la misma manera a obtener el vector unitario $e\,]_2$ en la primera columna a su derecha cuyos elementos

del segundo renglón en adelante sean no nulos. Este procedimiento se sigue hasta obtener la matriz rectangular en forma normal escalonada por renglones deseada.

Como ejemplo, supongamos que partimos de la matriz $[A_1]$

$$[A_1] = \begin{bmatrix} 0 & 2 & 8 & 2 \\ 2 & 2 & 4 & 8 \\ 1 & -1 & 2 & -2 \end{bmatrix}$$

Intercambiando el primero y segundo renglones y multiplicando el primer renglón por 1/2, se obtiene la matriz $[A_2]$

$$[A_2] = \begin{bmatrix} 1 & 1 & 2 & 4 \\ 0 & 2 & 8 & 2 \\ 1 & -1 & 2 & -2 \end{bmatrix}$$

Se ha obtenido un uno en la posición (1,1) (o en su defecto, en la primera columna diferente de cero). Ahora hay que transformar la primera columna en el vector unitario $e]_1$ realizando operaciones elementales de renglón. En este caso basta restarle al renglón 3 el 1 para obtener A_3

$$[A_3] = \begin{bmatrix} 1 & 1 & 2 & 4 \\ 0 & 2 & 8 & 2 \\ 0 & -2 & 0 & -6 \end{bmatrix}$$

Ahora se procede a cambiar la matriz $[A_3]$ para transformar la columna c_2 en el vector unitario $e]_2$.

Como la posición (2,2) de la matriz $[A_3]$ ya es distinta de cero, no será necesario intercambiar renglones, simplemente se multiplica el segundo renglón por 1/2; realizando operaciones elementales de renglón se convierte la columna 2 en $e]_2$ obteniendo $[A_4]$.

$$[A_4] = \begin{bmatrix} 1 & 0 & -2 & 3 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 8 & -4 \end{bmatrix}$$

Realizando operaciones elementales de renglón, se transforma esta matriz en la matriz $[A_5]$, donde ya la tercera columna es $e]_3$.

$$[A_5] = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -\frac{1}{2} \end{bmatrix}$$

La matriz $[A_5]$ está en forma normal escalonada como el lector puede comprobar.

Para resolver el sistema de ecuaciones del ejemplo A.3a, dadas por la ecuación (A.3.3) no hicimos más que pasar la matriz aumentada (A.3.3') a la forma normal escalonada (A.3.7'). Como se vio en ese ejemplo, una vez encontrada la forma normal escalonada se tiene resuelto el problema, que es encontrar el valor de $x]$.

Los siguientes ejemplos nos servirán para ilustrar tres posibles casos:

EJEMPLO A.3b Resuelva cada uno de los siguientes sistemas de ecuaciones:

$$1)\quad \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & 2 \\ 2 & 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 9 \\ 5 \\ 18 \end{bmatrix}$$

Solución:

La matriz aumentada es:

$$\begin{bmatrix} 1 & 1 & 2 & 9 \\ 0 & 1 & 2 & 5 \\ 2 & 2 & 4 & 18 \end{bmatrix}$$

y su forma normal escalonada es:

$$\begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 2 & 5 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

o sea:

$$x_1 = 4$$

$$x_2 + 2x_3 = 5$$

La solución es, por lo tanto, *no única* y está dada por

$$x_1 = 4$$

$$x_2 = 5 - 2x_3$$

A la incógnita x_3 se le puede dar cualquier valor y de este valor dependerá x_2. El número de soluciones es infinito.

2) $$\begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 2 \\ -2 & -1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 9 \\ 6 \\ 8 \end{bmatrix}.$$

Solución:

La matriz aumentada es:

$$\begin{bmatrix} 1 & 1 & 2 & 9 \\ 2 & 1 & 2 & 6 \\ -2 & -1 & -2 & 8 \end{bmatrix}$$

y su forma normal escalonada es:

$$\begin{bmatrix} 1 & 0 & 0 & 9 \\ 0 & 1 & 2 & 12 \\ 0 & 0 & 0 & 14 \end{bmatrix}$$

o sea:

$$x_1 = 9$$

$$x_2 + 2x_3 = 12$$

$$0 = 14$$

Este sistema de ecuaciones no tiene solución, pues

3)
$$\begin{aligned} x_1 + x_2 + x_3 &= 3 \\ 2x_1 - x_2 + 3x_3 &= 12 \\ -x_1 + 3x_2 - x_3 &= -10 \\ x_1 - 2x_2 - x_3 &= 0 \end{aligned}$$

Solución:

La matriz aumentada es:

$$\begin{bmatrix} 1 & 1 & 1 & 3 \\ 2 & -1 & 3 & 12 \\ -1 & 3 & -1 & -10 \\ 1 & -2 & -1 & 0 \end{bmatrix}$$

y su forma normal escalonada es:

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

o sea

$$\begin{aligned} x_1 &= 1 \\ x_2 &= -1 \\ x_3 &= 3 \\ 0 &= 0 \end{aligned}$$

Este sistema tiene una solución única.

Este ejemplo muestra que un sistema de ecuaciones lineales algebraicas puede tener:

a) Una solución única.
b) Un número infinito de soluciones.
c) Ninguna solución.

El empleo de la matriz aumentada y su forma normal escalonada permiten determinar cuál es el tipo de solución.

Para formalizar los resultados del ejemplo anterior, tenemos que introducir una importante definición:

Se llama *rango* de una matriz al número no nulo de renglones de su forma normal escalonada.

Supongamos que un sistema tiene *no solución*. Entonces la forma normal escalonada de la matriz aumentada $[\,[A]\,|\,b]\,]$ será de la forma:

$$\begin{bmatrix} 1 & 0 & & 0 & b'_1 \\ 0 & 1 & & \vdots & \\ \vdots & 0 & & 0 & \\ \vdots & \vdots & & 1 & b'_n \\ 0 & \vdots & & & \\ 0 & \cdots\cdots & \cdots\cdots & 0 & b'_{n+1} \end{bmatrix}$$

Con $b'_{n+1} = 0$ ya que esta forma normal escalonada implica

$$\begin{aligned} x_1 &= b' \\ x_2 &= b''_2 \\ &\cdots \\ x_n &= b'_n \\ 0 &= b'_{n+1} \end{aligned}$$

El último renglón implica una contradicción. En el segundo sistema del ejemplo anterior se presenta precisamente este caso.

De acuerdo con la definición de rango, esta matriz aumentada tiene un rango de $n+1$. La forma normal escalonada de la matriz $[A]$ es:

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & \\ \vdots & \vdots & & \\ 0 & 0 & & 1 \\ 0 & \cdots & & 0 \\ \vdots & & & \\ 0 & \cdots & & 0 \end{bmatrix}$$

y su rango es n. Por lo tanto, podemos establecer el siguiente teorema:

Teorema A.3.1

Si el rango de la matriz aumentada $[[A]\,|\,b]$ es mayor que el rango de la matriz $[A]$ el sistema (A.3.2) *no tiene solución.*

De manera similar el lector puede establecer el siguiente teorema:

Teorema A.3.2

Si el rango de la matriz aumentada es igual al rango de $[A]$ el sistema (A.3.2) *tiene solución* y si el rango es igual al número de incógnitas, la solución *es única.*

Un sistema de ecs. (A.3.2) es homogéneo, si $b] = 0]$. En este caso la matriz aumentada tiene igual rango que la matriz $[A]$ siempre y cuando el teorema (A.3.1) resulte irrelevante y podemos establecer que un sistema homogéneo siempre tiene solución, pero de acuerdo con el teorema

A.3.2 si el rango de $\left[A\right]$ es igual al número de incógnitas, la solución es única e igual a $x\rangle = 0$. En caso contrario, si el rango de $\left[A\right]$ es *menor* que el número de incógnitas, existirán una infinidad de soluciones.

Puede demostrarse que si el rango de una matriz cuadrada de $n \times n$ es menor que n entonces su *determinante es nulo*.

Por lo tanto, para un sistema homogéneo $\left[A\right] x\rangle = 0\rangle$ de n ecuaciones en n incógnitas, existe una solución diferente a $x\rangle = 0\rangle$ si y solamente si el rango de $\left[A\right]$ es menor que n ó sea el determinante $|A| = 0$.

A.4 Vectores y espacios vectoriales

El lector recordará que una terna ordenada de números reales, que en general se escriben como si se tratase de una matriz de columna de 3×1, pueden representar las tres componentes, en un sistema de referencia x, y, z, de un segmento de recta dirigido.

Podemos generalizar este concepto y considerar a un vector como un n-tuplo ordenado de elementos que pueden ser reales, complejos, funciones del tiempo u operadores. Desde luego, si el número de elementos es mayor de tres ya no podemos emplear la representación geométrica usual.

Para las operaciones de suma, resta y multiplicación, los vectores se manipulan como si fuesen matrices de una columna. En general los vectores se representan con $x\rangle$, $y\rangle$, $z\rangle$, etc.

Se dice que m vectores $v\rangle_1, v\rangle_2, \ldots\ldots v\rangle_m$ *son linealmente independientes*, si la igualdad

$$a_1 v\rangle_1 + a_2 v\rangle_2 + \ldots\ldots + a_m v\rangle_m = 0 \qquad \text{(A.4.1)}$$

implica que $a_1 = a_2 = \ldots\ldots = a_m = 0$, donde $a_1, a_2, \ldots, a_m$ son escalares.

En caso contrario se dice que los vectores son linealmente dependientes.

EJEMPLO A.4a Determine si los vectores

$$x] = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} \quad y] = \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} \quad z] = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$$

son linealmente independientes:

Solución:

Si los vectores son linealmente independientes, entonces

$$a_1\ x] + a_2\ y] + a_3\ z] = 0]$$

implica

$$a_1 = a_2 = a_3 = 0$$

o sea tenemos que analizar la solución de:

$$\begin{aligned} a_1 + 0a_2 + 2a_3 &= 0 \\ 0 - 1a_2 + 2a_3 &= 0 \\ 2a_1 + 1a_2 + 2a_3 &= 0 \end{aligned}$$

Esta es una ecuación homogénea con 3 incógnitas a_1, a_2 y a_3

Pasemos la matriz

$$\begin{bmatrix} 1 & 0 & 2 \\ 0 & -1 & 2 \\ 2 & 1 & 2 \end{bmatrix}$$

a la forma normal escalonada

$$\begin{bmatrix} 1 & 0 & 2 \\ 0 & -1 & 2 \\ 0 & 1 & -2 \end{bmatrix} \quad ; \quad \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & -2 \\ 0 & 0 & 0 \end{bmatrix}$$

De acuerdo con la última matriz el rango de $[A]$ es menor que el número de incógnitas, existe una solución diferente a $a_1 = 0, a_2 = 0, a_3 = 0$. En efecto de la ecuación anterior tenemos:

$$a_1 + 2a_3 = 0$$

$$a_2 - 2a_3 = 0$$

o sea

$$a_1 = -2a_3$$

$$a_2 = 2a_3$$

Del número infinito de soluciones que tienen estas

ecuaciones podemos considerar aquella para la que $a_3 = 1$, que implica que $a_1 = -2$ y $a_2 = 2$.

El lector puede comprobar que en efecto:

$$-\ 2x\rangle\ +\ 2y\rangle\ +\ z\rangle\ =\ 0\rangle$$

por lo que los tres vectores son linealmente dependientes.

Espacio vectorial

Daremos solamente una definición apropiada a nuestras propósitos:

Un conjunto de vectores se dice que forman un espacio vectorial, si dados dos miembros del conjunto $u\rangle$ y $v\rangle$ y dos escalares cualesquiera α y β, entonces $\alpha u\rangle + \beta v\rangle$ también pertenecen al conjunto.

El espacio cartesiano de tres dimensiones es un espacio vectorial. Existen muchos otros tipos de espacios vectoriales.

Se dice que un conjunto de vectores $\{u\rangle, \ldots\ldots, u\rangle_n\}$ *expanden* un espacio vectorial si cualquier vector $v\rangle$ perteneciente a este espacio vectorial puede expresarse como una combinación lineal de ellos:

$$v\rangle = \sum_{i=1}^{n} a_i\ u\rangle_i$$

Un conjunto de vectores $\{b\rangle_1, \ldots\ldots b\rangle_n\}$ forman una base del espacio vectorial si lo expanden y son linealmente independientes.

En el espacio de tres dimensiones

$$e\rangle_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad e\rangle_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \text{y} \quad e\rangle_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

forman *una posible* base ya que cualquier vector:

$$v\rangle = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

en una combinación lineal de los $e\rangle_i$. En efecto:

$$v\rangle = a\, e\rangle_1 + b\, e\rangle_2 + c\, e\rangle_3$$

A.5 Transformaciones lineales, valores y vectores característicos

El producto de una matriz cuadrada $[A]$ de $n \times n$ y un vector $u\rangle$ de n elementos es otro vector, también de n elementos. Para visualizar la relación entre el vector $u\rangle$ y el vector $[A]u\rangle$ consideremos que $n = 2$. En la fig. A.5.1a se muestran dos vectores $u\rangle$ y $v\rangle$ y los productos

$$\omega\rangle = [A]u\rangle$$

y

$$y\rangle = [A]v\rangle$$

donde la matriz $\left[A\right]$ es:

$$\left[A\right] = \begin{matrix} \frac{5}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{4}{3} \end{matrix}$$

La fig. A.5.1b muestra la suma de los vectores $u\rangle$ y $v\rangle$, obtenida empleando la regla del paralelogramo. En la fig. A.5.1c aparece el vector producto $\left[A\right] u\rangle + u\rangle$ y finalmente la fig. A.5.1d muestra los vectores $\left[A\right] u\rangle$ y $\left[A\right] v\rangle$ y su suma, $\left[A\right] u\rangle + \left[A\right] v\rangle$. Observemos que:

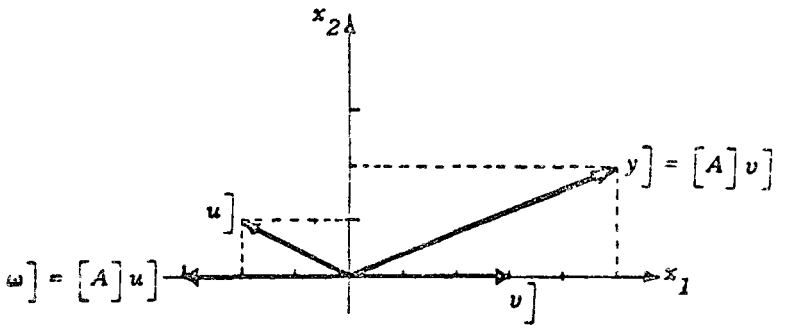
$$\left[A\right]\left(u\rangle + v\rangle\right) = \left[A\right] u\rangle + \left[A\right] v\rangle$$

En general si $u\rangle_1, u\rangle_2, \ldots u\rangle_r$ son r vectores de n elementos, $\left[A\right]$ es una matriz de $n \times n$ y $\alpha_1, \alpha_2, \ldots, \alpha_r$ son r escalares entonces

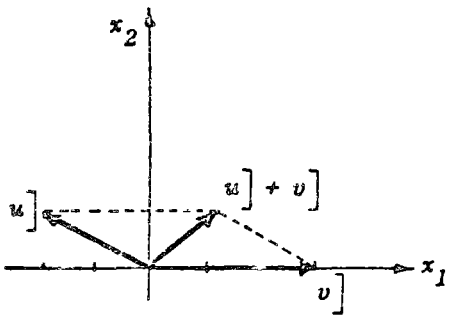
$$\left[A\right] \sum_{i=1}^{r} \alpha_i u\rangle_i = \sum_{i=1}^{r} \left[A\right] \alpha_i u\rangle_i$$

Podemos concluir, de acuerdo con el concepto de linealidad introducido en la sección 1.1 que una matriz $\left[A\right]$ de $n \times n$ es un *operador lineal* en el espacio de n dimensiones.

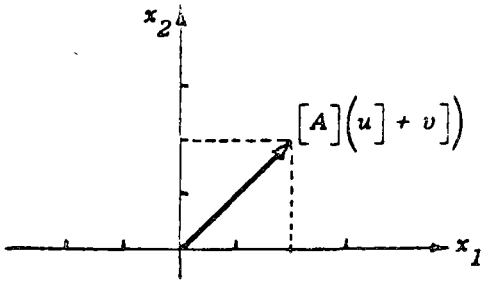
Notemos además que los vectores $\omega\rangle$ y $u\rangle$ y los vectores $y\rangle$ y $v\rangle$ tienen no solamente diferente módulo sino también diferente dirección. En general el vector $u\rangle$ y el vector



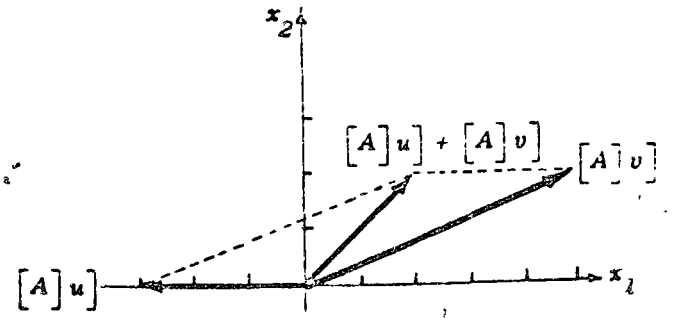
(a)



(b)



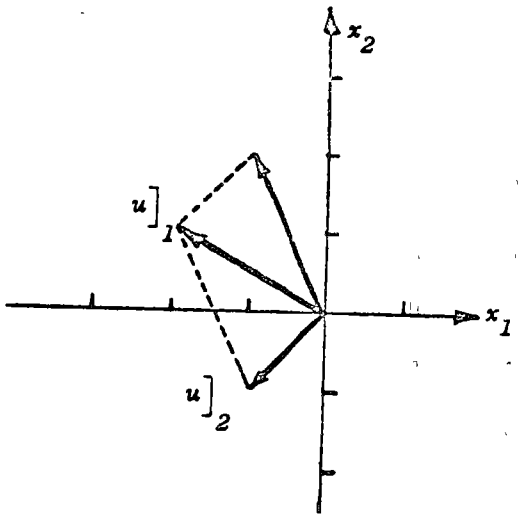
(c)



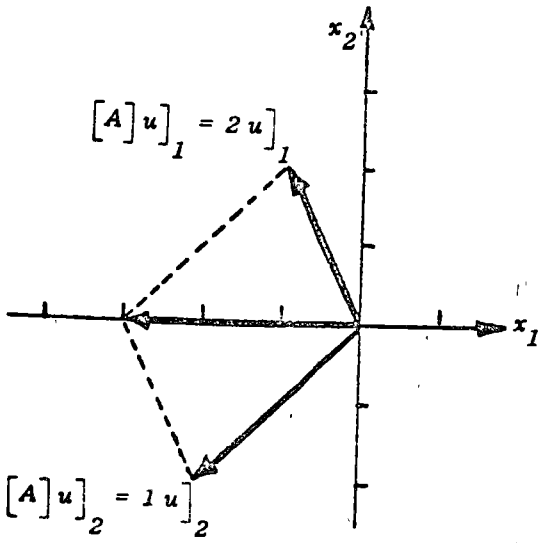
(d)

Fig. A.5.1. Ilustración del concepto de trasformación lineal

$[A]u]$, no son ni de igual módulo ni colineales. Sin embargo, en la fig. A.5.2 se observa que para dos vectores en particular, $u]_1$ y $u]_2$, los vectores producto $[A]u]_1$ y $[A]u]_2$ si tienen la misma dirección que los vectores $u]_1$ y $u]_2$. Los vectores que tienen esta propiedad, que puede expresarse como



(a)



(b)

Fig. A.5.2. Ilustración de las propiedades geométricas de los vectores característicos

$$[A]\,u] = \lambda\, u] \tag{A.5.1}$$

reciben el nombre de vectores característicos de la matriz $[A]$. Los valores (λ) se conocen con el nombre de valores característicos de la matriz $[A]$. Estos valores característicos se obtienen resolviendo el sistema de ecs. (A.5.1) que también puede escribirse de la siguiente forma:

$$\begin{aligned} &[A]\,u] - u] = 0] \\ &[A]\,u] - [I]\,u] = 0] \\ &\{[A] - \lambda[I]\}\,u] = 0] \end{aligned} \tag{A.5.2}$$

La solución de (A.5.2) nos permite encontrar los valores y vectores característicos asociados a la matriz $[A]$.

EJEMPLO A.5a Encuentre los vectores y valores característicos de la matriz:

$$[A] = \begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix}$$

Solución:

de (A.5.2)

$$\left\{\begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right\} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

donde $u] = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$\left\{ \begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \right\} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0]$$

$$\begin{bmatrix} 3-\lambda & 4 \\ 2 & 1-\lambda \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0]$$

(A.5.3)
$$\begin{aligned} (3-\lambda)x_1 + 4x_2 &= 0 \\ 2x_1 + (1-\lambda)x_2 &= 0 \end{aligned}$$

Esta ecuación homogénea tiene una solución diferente a $x] = 0$ si y solamente si se cumple

$$\begin{vmatrix} 3-\lambda & 4 \\ 2 & 1-\lambda \end{vmatrix} = 0$$

al determinante $[A] - \lambda[I]$ se le llama *polinomio característico de la matriz* $[A]$, y lo representamos con $g(\lambda)$

(A.5.4) $$g(\lambda) = [A] - \lambda[I]$$

En este ejemplo la ecuación característica es:

$$\begin{aligned} g(\lambda) = (3-\lambda)(1-\lambda) - 2.4 &= 0 \\ 3 - 4\lambda + \lambda^2 - 8 &= 0 \\ \lambda^2 - 4\lambda - 5 &= 0 \end{aligned}$$

y sus raíces son:

$$\lambda_1 = 5$$

$$\lambda_2 = -1$$

Para calcular los vectores característicos sustituimos en (A.4.4). Para $\lambda_1 = 5$

$$(3 - 5)x_1 + 4x_2 = 0$$

$$2x_1 + (1 - 5)x_2 = 0$$

o sea

$$-2x_1 + 4x_2 = 0$$

$$2x_1 - 4x_2 = 0$$

Vemos que una ecuación es el negativo de la otra de tal forma que

$$x_1 = 2x_2$$

es la solución del sistema de ecuaciones.

Dando cualquier valor a x_2 tendremos un valor de x_1, y existen un número infinito de soluciones. Si tomamos $x_1 = 2$, $x_2 = 1$ y por lo tanto: para $\lambda_1 = 5$, un posible vector característico es:

$$\begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

para el segundo vector característico

$$\lambda_2 = -1$$

$$(3 - (-1))\, x_1 + 4x_2 = 0$$

$$2x_1 + (1 - (-1))\, x_2 = 0$$

o sea

$$4x_1 + 4x_2 = 0$$

$$2x_1 + 2x_2 = 0$$

Como $x_1 = -x_2$ es la solución del sistema de ecuaciones. Una de las posibles soluciones de este sistema es:

$$u = \begin{matrix} +1 \\ -1 \end{matrix}$$

Para encontrar los valores característicos como se vió en el ejemplo, hay que encontrar las raíces de:

$$[A] - \lambda[I] = 0 \qquad \text{(A.5.5)}$$

La ec. (A.5.5) representa una ecuación algebraica de grado igual al orden de la matriz $[A]$. Esta ecuación hemos dicho que se llama *ecuación característica de* $[A]$ y su lado izquierdo se llama *polinomio característico de* $[A]$,

y representado por $g(\lambda)$. Como es una ecuación de grado n, la ecuación característica tiene n raíces (no necesariamente distintas), que determina n valores característicos. En este libro, consideraremos solamente el caso en que las raíces son distintas, pues si hay raíces repetidas, se presentan ciertas dificultades que complican la teoría.

Para cada λ_i se establece el sistema ecuaciones homogéneas (A.5.3); si su rango es menor que el número de incógnitas, su solución será diferente a $u\rbrack = 0$ y dicha solución es el *vector característico.*

La solución de cada uno de los sistemas de ecuaciones del tipo (A.5.3) (una para cada λ) permite encontrar un vector característico $u\rbrack_i$ correspondiente a cada λ; este vector característico no queda caracterizado totalmente, sino que contiene una constante arbitraria que lo multiplica. Para eliminar esta constante arbitraria, los vectores característicos pueden normalizarse, haciendo su módulo igual a la unidad. Recordemos que como módulo de un vector entendemos:

$$\text{(A.5.6)} \qquad u\rbrack = \sqrt{u_1^2 + u_2^2 + \ldots\ldots + u_n^2}$$

Para normalizar el vector característico dividimos cada componente u_i entre su módulo.

Antes de seguir adelante tenemos que estudiar un importante teorema y sus implicaciones:

Teorema A.5.1

(i) Los vectores característicos correspondientes a valores característicos distintos son linealmente independientes.

(ii) Si la matriz $[A]$ tiene n valores característicos diferentes, entonces existen exactamente n vectores característicos.

Desmostración:

Supongamos que s es el menor número de vectores linealmente dependientes, y llamémoslos $u]_1, u_2], \ldots, u_s]$. Entonces se debe cumplir que:

$$a_1\ u]_1 + a_2\ u]_2 + \ldots\ldots + a_s\ u]_s = 0] \tag{A.5.7}$$

donde, todas las a_i son diferentes a cero.

Si pre-multiplicamos (A.5.7) por $[A]$ y recordamos que

$$[A]\ u]_i = \lambda_i\ u]_i$$

se obtiene

$$a_1\ \lambda_1\ u]_1 + a_2\ \lambda_2\ u]_2 + \ldots\ldots + a_s\ \lambda_s\ u]_s = 0] \tag{A.5.8}$$

Restando (A.5.8) de λ_1 veces (A.5.7) se tiene

$$a_2 (\lambda_1 - \lambda_2)\ u]_2 + \ldots\ldots + a_s (\lambda_1 - \lambda_s)\ u]_s = 0] \tag{A.5.9}$$

Como $\lambda_1 - \lambda_i \neq 0$ para $i = 2, \ldots, s$, (A.5.9) establece una relación de dependencia lineal entre $s - 1$ vectores, resultado que contradice la suposición de que s era el menor número para el cual son linealmente dependientes.

Supongamos ahora que además de $u]_i$ existe un segundo vector característico $x]$ correspondiente a λ_i. Como los n vectores $u]_i$ por (i) son linealmente independientes, sirven como base en un espacio de n dimensiones, y el vector $x]$ puede expresarse como una combinación lineal de los $u]_i$ o sea:

$$(A.5.10) \quad x] = \sum_{i=1}^{n} \beta_i \, u]_i$$

Multiplicando por $[A]$ y recordando que :

$$[A] \; u]_i = \lambda_i \; u]_i$$

se tiene:

$$(A.5.11) \quad \lambda_i \; u]_i = \sum_{i=1}^{n} \beta_i \; \lambda_i \; u_i$$

Restando λ_i veces (A.5.11) de (A.5.10)

$$0] = \beta_1 (\lambda_1 - \lambda_i) \, u]_1 + \ldots\ldots + \beta_n (\lambda_n - \lambda_i) \, u]_n$$

Como los vectores $u]_i$ son linealmente independientes, $B_j = 0$ $(i = 1, \ldots\ldots n, \; i = j)$ de donde de (A.5.10)

$$x] = \beta_i \, u]_i$$

o sea $x\rangle$ depende de $u\rangle_i$, quedando demostrada la segunda parte del teorema, es decir a cada valor característico corresponde un solo vector característico.

El lector debe recordar que el módulo del vector característico puede ser cualquiera, por lo que $x\rangle$ y $u\rangle_i$, vectores con igual dirección pero diferente módulo, se consideran como el mismo vector característico.

Por lo estudiado anteriormente el rango de $[P]$ es n y su determinante diferente de cero, el inveros de $[P]$ existe.

EJEMPLO A.5b Se ha visto que la matriz

$$[A] = \begin{bmatrix} \frac{5}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{4}{3} \end{bmatrix}$$

tiene por valores característicos $\lambda_1 = 2, \lambda_2 = 1$. Los vectores característicos correspondientes son

$$u\rangle_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad y \quad u\rangle_2 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

Como estos dos vectores corresponden a diferentes valores característicos, son linealmente independientes y pueden servir como base en el espacio de dos dimensiones. Exprése al vector

$$v\rangle = \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

como una combinación lineal de $u]_1$ y $u]_2$ y calcule $[A]v]$ directamente y empleando las dos componentes de $v]$.

Solución:

El vector $v]$ puede expresarse como combinación lineal de $u]_1$ y $u]_2$ como sigue:

$$\begin{bmatrix} -2 \\ 1 \end{bmatrix} = a \begin{bmatrix} 1 \\ 1 \end{bmatrix} + b \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

de donde los valores de a y b son

$a = -1,$

$b = -1$

Por lo tanto, si empleamos un sistema de coordenadas con el eje de las abscisas alineado con el vector $u]_1$ y el de las ordenadas alineado con $u]_2$ las dos componentes del vector $v]$ serían -1 y -1. Se dice que el vector $v]$ en el sistema coordenado con bases

$$e]_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

y

$$e]_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

tiene como componentes

$$\left[v \right] = \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

y que en el sistema con bases

$$\left[u \right]_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

y

$$\left[u \right]_2 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

tiene como componentes

$$\left[v \right] = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

Para calcular $\left[A \right] \left[u \right]$ puede procederse de las siguientes maneras:

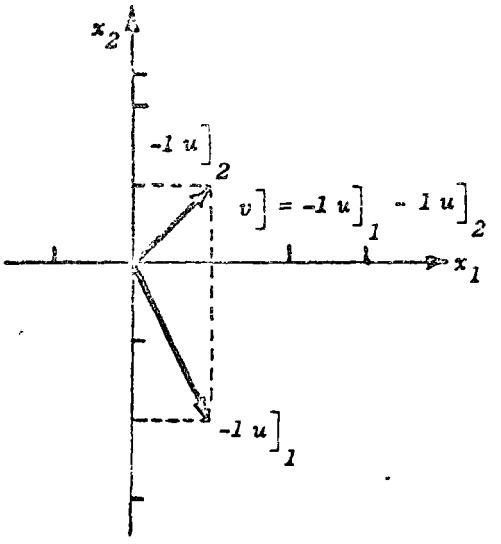
a)

$$\left[A \right] \left[v \right] = \begin{matrix} \frac{5}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{4}{3} \end{matrix} \begin{matrix} -2 \\ 1 \end{matrix} = \begin{matrix} -3 \\ 0 \end{matrix}$$

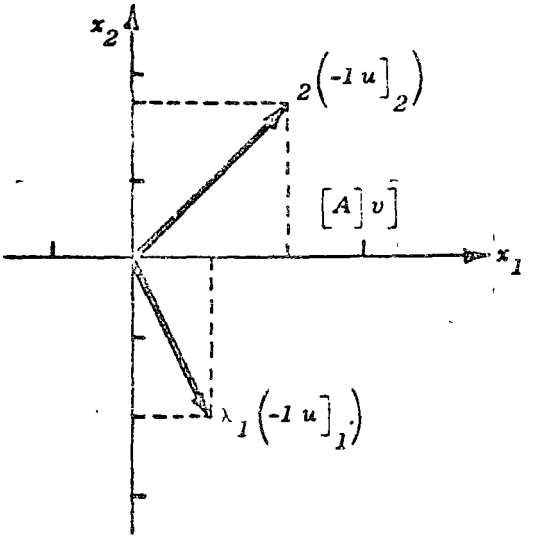
b)

$$\left[A\right] v\right] = \left[A\right]\left(a\,u\right]_1 + b\,u\right]_2\Big)$$

$$= a\left[A\right]u\right]_1 + b\left[A\right]u\right]_2$$



(a)



(b)

Fig. A.5.3. *Ejemplo de cambio de bases*

por (A.5.1) podemos escribir:

$$= a\,\lambda_1 u]_1 + b\,\lambda_2 u]_2$$

$$= \begin{matrix} -2 \\ -2 \end{matrix}\Bigg] + \begin{matrix} -1 \\ 2 \end{matrix}\Bigg] = \begin{matrix} -3 \\ 0 \end{matrix}$$

La fig. A.5.3 ilustra estas dos formas de calcular $[A]v]$.

Si la matriz $[A]$ de $n \times n$ tiene n vectores característicos linealmente independientes, éstos pueden servir como base. Todo vector de dimensión n puede expresarse en esa base. Este ejemplo ilustra cómo se calculan las componentes de un vector al cambiar de base. Además si se cumple como base, la formada por los vectores característicos, la transformación lineal $[A]v]$ se conviente en una combinación lineal de los vectores característicos.

A.6 Funciones de matrices cuadradas

Empecemos definiendo las *potencias positivas* de una matriz. Estas se definen por analogía con las potencias de escalares como:

$$[A]^2 = [A][A]; \quad [A]^3 = [A][A]^2$$

$$= [A]^2[A] \ldots,$$

$$[A]^{k-1} = [A][A]^k = [A][A]^k \ldots \quad \text{(A.6.1)}$$

Las potencias negativas se definen como potencias de la matriz inversa

$$(A.6.2) \quad [A] = \left([A]^{-1} \right)^{k}$$

Las potencias negativas, por lo tanto, existen solamente cuando $[A]^{-1}$ existe, o sea cuando la matriz es *no singular*.

La ley de los exponentes es válida cuando las potencias de las matrices existen.

Así:

$$[A]^{k} \; [A]^{p} = [A]^{k+p}$$

$$[A]^{k} \; [A]^{-p} = [A]^{k-p}$$

También tenemos:

$$[A]^{k} \; [A]^{-k} = [A]^{0} = [I]$$

Podemos definir polinomios de matrices con coeficientes escalares como sigue: si

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots$$

$$\vdots$$

$$+ a_1 x + a_0 x^0.$$

entonces para una matriz cuadrada $[A]$

$$p\left(\left[A\right]\right) = a_n \left[A\right]^n + a_{n-1} \left[A\right]^{n-1} + \ldots$$

(A.6.3)
$$+ a \left[A\right] + a_o \left[I\right]$$

Si

$$f(x) = \sum_{k=0}^{\infty} a_k x^k$$

podemos definir una serie infinita de matrices como

(A.6.4) $$f\left(\left[A\right]\right) = \sum_{k=0}^{\infty} a_k \left[A\right]$$

Las series infinitas se emplean para definir funciones como: seno, exponencial, senh, cosh, etc. de una matriz cuadrada $\left[A\right]$.

Por ejemplo:

$$exp\left(\left[A\right]\right) \sum_{k=0}^{\infty} \frac{\left[A\right]}{k!} = \left[A\right]^0 + \left[A\right]$$

(A.6.5)
$$+ \frac{\left[A\right]^2}{2} + \frac{\left[A\right]^3}{3} + \ldots$$

$$sen\left[A\right] = \left[A\right] - \frac{\left[A\right]^3}{3!}$$

(A.6.6)
$$+ \frac{\left[A\right]^5}{5!} - \frac{\left[A\right]^7}{7!} + \ldots..$$

Funciones de matrices cuadradas

Resulta muy laborioso tener que evaluar una serie infinita de matrices para encontrar el valor de la función de una matriz. Un método para calcularlas consiste en emplear teoremas derivados del teorema de Cayley-Hamilton que a continuación se enuncia:

Teorema de Cayley-Hamilton (A.6a)

Este teorema establece que cualquier matriz cuadrada satisface su propia ecuación característica. Por lo tanto si el polinomio característico de una matriz $[A]$ es $g(\lambda)$, entonces $g[A] = 0$.

A continuación estudiaremos algunas importantes implicaciones de este teorema.

Sea la ecuación característica de una matriz:

$$g(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + a_{n-2}\lambda^{n-2} + \ldots + a_1\lambda^1 + a_o = 0 \qquad \text{(A.6.7)}$$

Por el teorema de Cayley-Hamilton la matriz $[A]$ debe satisfacer a su propia ecuación característica por lo que sustituyendo en (A.6.7) λ por $[A]$ tenemos:

$$g([A]) = [A]^n + a_{n-1}[A]^{n-1} + a_{n-2}[A]^{n-2} + a_{n-3}[A]^{n-3} + \ldots + a_1[A] + a_0[I] = [0]$$

Despejando $[A]^n$ tenemos:

$$[A]^n = -a_{n-1}[A]^{n-1} - a_{n-2}[A]^{n-2} - \cdots - a_1[A] - a_0[I] \tag{A.6.8}$$

Multiplicando por $[A]$:

$$[A][A]^n = -a_{n-1}[A]^n - a_{n-2}[A]^{n-1} \cdots - a_1[A]^2 - a_0[A] \tag{A.6.9}$$

Sustituyendo (A.6.8) en (A.6.9)

$$[A]^{n+1} = -a_{n-1}\{-a_{n-1}[A]^{n-1} - \cdots - a_1[A] - a_0[I]\} - a_{n-2}[A]^{n-1} - \cdots - a_1[A]^2 - a_0[A] \tag{A.6.10}$$

Observamos que en la expresión (A.6.10) la máxima potencia de $[A]$ en el lado derecho de la ecuación es $n - 1$; podemos, por lo tanto, establecer el siguiente teorema.

Teorema (A.6b)

Si $[A]$ es una matriz cuadrada de $n \times n$ cualquier suma potencia de $[A]$ puede expresarse como una suma de po-

tencias de $[A]$ cuyo mayor exponente es cuando más $n-1$, o sea

$$(A.6.11) \quad f\left([A]\right) = \sum_{j=0}^{n-1} \alpha_j [A]^j$$

EJEMPLO A.6a Verifique el teorema de Cayley-Hamilton para la matriz $[A]$ dada en el ejemplo A.4b

$$[A] = \begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix}$$

Solución

La ecuación característica es:

$$g(\lambda) = \begin{vmatrix} (3-\lambda) & 4 \\ 2 & (1-\lambda) \end{vmatrix} = 0$$

$$g(\lambda) = \lambda^2 - 4\lambda - 5 = 0$$

En este caso, $g\left([A]\right)$ será:

$$g\left([A]\right) = \begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} - 4\begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} - 5\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

El primer término es

$$\begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 17 & 16 \\ 8 & 9 \end{bmatrix}$$

que sustituido en la ecuación da

$$g\left(\left[A\right]\right)$$

$$= \begin{bmatrix} 17 & 16 \\ 8 & 9 \end{bmatrix} - \begin{bmatrix} 12 & 16 \\ 8 & 4 \end{bmatrix} - \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$$

$$= \begin{bmatrix} (17 - 12 - 5) & (16 - 16 - 0) \\ (8 - 8 - 0) & (9 - 4 - 5) \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Por lo tanto, el teorema de Cayley-Hamilton que afirma que $g\left(\left[A\right]\right) = 0$, se satisface.

EJEMPLO A.6b Calcule:

(A.6.12) $$f\left(\left[A\right]\right) = \left[A\right]^3 + 2\left[A\right]$$

Solución:

Por el teorema de Cayley-Hamilton

$$[A]^2 - 4[A] - 5[I] = [0]$$

$$[A]^2 = 4[A] + 5[I]$$

Multiplicando por $[A]$

$$[A]^3 = 4[A]^2 + 5[A]$$

Sustituyendo el valor de $[A]^2$ dado por la ecuación característica tenemos:

$$[A]^3 = 4\left(4[A] + 5[I]\right) + 5[A]$$

$$= 21\,[A] + 20[I]$$

y sustituyendo en $f[A]$:

$$f\left([A]\right) = 23[A] + 20[I]$$

$$= 23\begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} + 20\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 69 & 92 \\ 46 & 23 \end{bmatrix} + \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}$$

$$f\left(\left[A\right]\right) = \begin{bmatrix} 89 & 92 \\ 46 & 43 \end{bmatrix}$$

Obsérvese que $f\left(\left[A\right]\right)$ que contenía originalmente una potencia cúbica de $\left[A\right]$ quedó reducido a una combinación de potencias de $\left[A\right]$ cuyo máximo exponente fue 2 - 1 = 1, como el teorema (A.6b) afirma debe suceder.

A continuación enunciaremos otro teorema:

Teorema (A.6c)

Cualquier función de una matriz $f\left(\left[A\right]\right)$ debe ser también satisfecha si $\left[A\right]$ se sustituye por alguno de los valores característicos de la matriz.

A continuación ilustramos la aplicación del teorema (A.5c) para el cálculo de $exp\left(\left[A\right]t\right)$

EJEMPLO A.6c Calcule $exp\left(\left[A\right]t\right)$ para la matriz del ejemplo A.4b, usando el teorema (A.6c).

Solución:

Por (A.5.17)

$$exp\left([A]t\right) = a_1[A] + a_0[I]$$

Por el teorema (A.6c).

$$exp(\lambda_1 t) = a_1 \lambda_1 + a_0$$

$$exp(\lambda_2 t) = a_1 \lambda_2 + a_0$$

en el ejemplo A.4b encontramos $\lambda_1 = 5$, $\lambda_2 = -1$ o sea:

$$exp(5t) = 5a_1 + a_0$$

$$exp(-t) = -a_1 + a_0$$

despejando a_1 y a_0:

$$a_1 = \frac{exp(5t) - exp(-t)}{6}$$

$$a_0 = \frac{exp(5t) + 5\,exp(-t)}{6}$$

Sustituyendo en (A.5.20)

$$exp\left([A]t\right) = a_1\begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix} + a_0\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 3a_1 + a_0 & 4a_1 \\ 2a_1 & a_1 + a_0 \end{bmatrix}$$

$$\begin{bmatrix} \frac{2}{3}\exp(5t) + \frac{1}{3}\exp(-t) & \frac{2}{3}\exp(5t) - \frac{2}{3}\exp(-t) \\ \frac{1}{3}\exp(5t) - \frac{1}{3}\exp(-t) & \frac{1}{3}\exp(5t) + \frac{2}{3}\exp(-t) \end{bmatrix}$$

A.7 Problemas

1. Ilustre que $[A][B] \neq [B][A]$ para las matrices:

a) $\begin{bmatrix} 1 & 3 \end{bmatrix}$, $\begin{bmatrix} 2 \\ 4 \end{bmatrix}$

b) $\begin{bmatrix} 1 & 0 \\ 2 & 3 \end{bmatrix}$; $\begin{bmatrix} 0 & 4 \\ 5 & -6 \end{bmatrix}$

2. Dadas las matrices

$$[A] = \begin{bmatrix} 1 & 0 & -1 \\ 2 & -3 & 0 \end{bmatrix}$$

$$[B] = \begin{bmatrix} 0 & 10 & -5 \\ 2 & -6 & 2 \end{bmatrix}$$

$$[C] = \begin{bmatrix} 0 & 2 \\ 3 & -1 \\ 4 & 2 \end{bmatrix}$$

Compruebe que: $\left([A] + [B]\right)[C] = [A][C] + [D][C]$

3. Para la matriz

$$A = \begin{bmatrix} 1 & -2 \\ -5 & 3 \end{bmatrix}$$

Demuestre que se satisface la ec. (A.2.15).

4. Empleando (A.2.17) calcule el inverso de la matriz de 2×2

$$[A] = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

y establezca una regla para calcular dichos inversos en el caso de una matriz de 2×2.

5. Establezca si los siguientes sistemas de ecuaciones tienen solución o no.

En caso afirmativo encuéntrela

a) $x_1 + x_2 + 2x_3 = 3$

$2x_1 - x_2 - x_3 = 4$

b) $2x_1 + 3x_2 + 2x_3 = 6$

$x_1 + x_2 - x_3 = 4$

$4x_1 + 2x_2 + 4x_3 = 4$

c) $$\begin{aligned} x_1 + 2x_2 - 3x_3 &= 8 \\ 2x_1 - x_2 + 2x_3 &= -2 \\ 3x_1 + 2x_2 - x_3 &= 6 \end{aligned}$$

d) $$\begin{aligned} 2x_1 + 2x_2 + 4x_3 &= 0 \\ x_1 - x_2 - x_3 &= 0 \\ -x_1 + 2x_2 + 4x_3 &= 0 \end{aligned}$$

6. Encuentre el rango de las siguientes matrices

a) $$\begin{bmatrix} -1 & -2 & 3 \\ -4 & 5 & -4 \\ 3 & 2 & -1 \end{bmatrix}$$

b) $$\begin{bmatrix} 1 & 1 & 2 \\ -2 & 1 & 1 \end{bmatrix}$$

c) $$\begin{bmatrix} 2 & 2 & 4 \\ 1 & -1 & -1 \\ 1 & 2 & 4 \end{bmatrix}$$

7. Encuentre si los conjuntos de vectores son linealmente independientes.

a) $$\begin{bmatrix} 1 \\ -1 \\ 2 \end{bmatrix} \quad \begin{bmatrix} 2 \\ 3 \\ -1 \end{bmatrix} \quad \begin{bmatrix} -4 \\ -1 \\ -3 \end{bmatrix}$$

b) $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ 0 \\ -3 \\ 2 \end{bmatrix}$

8. Encuentre los vectores y valores característicos de las siguientes matrices:

a) $\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}$

b) $\begin{bmatrix} 3 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & 2 & 0 \end{bmatrix}$

c) $\begin{bmatrix} 2 & -2 & 3 \\ 1 & 1 & 1 \\ 1 & 3 & -1 \end{bmatrix}$

9. Demuestre que $exp\left(\left[A\right]\right) \cdot exp\left(\left[B\right]\right) = exp\left(\left[A\right] + \left[B\right]\right)$ si y solamente si $\left[A\right]\left[B\right] = \left[B\right]\left[A\right]$.

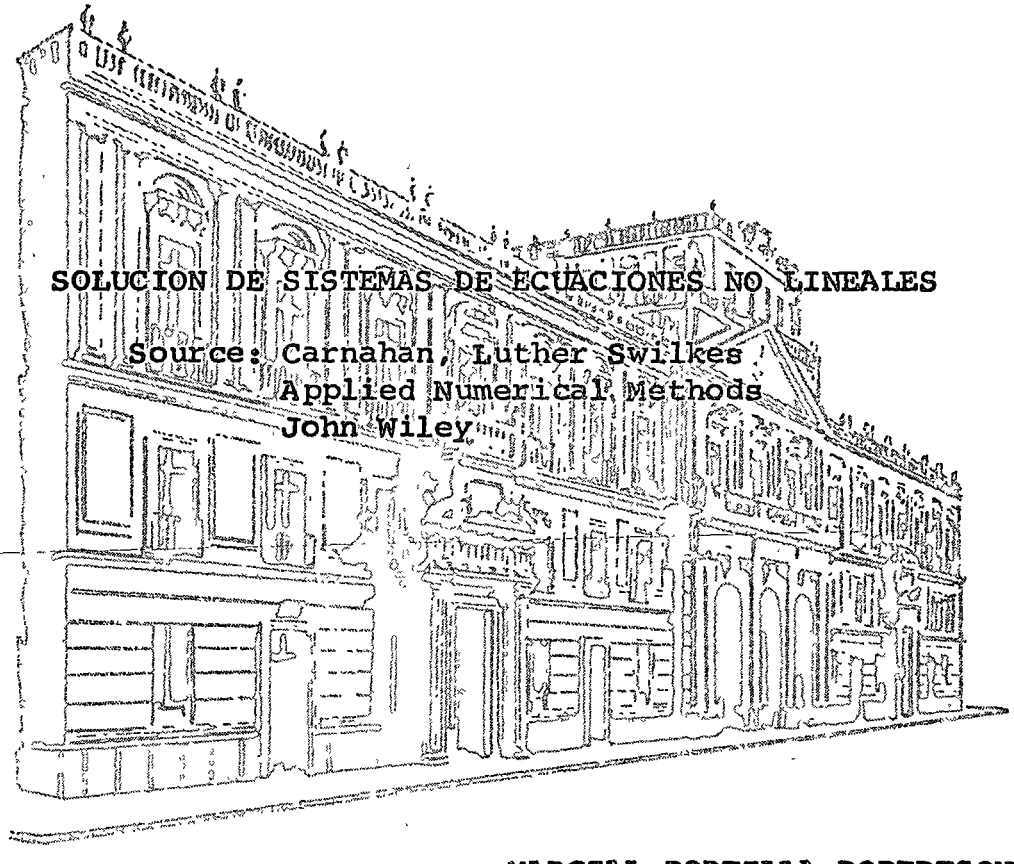
10. Calcule $exp\left(\left[A\right]t\right)$ para las matrices a) y b) del problema 8.

11. Calcule sen $\left[A\right]t$ para la matriz

$$\left[A\right] = \begin{bmatrix} -3 & 2 \\ 2 & 0 \end{bmatrix}$$

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL



SOLUCION DE SISTEMAS DE ECUACIONES NO LINEALES

Source: Carnahan, Luther Swilkes
Applied Numerical Methods
John Wiley

MARCIAL PORTILLA ROBERTSON

ABRIL DE 1976.

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels.: 521-40-23 521-73-35 5123-123

5.8 Iterative Methods for Nonlinear Equations

Sections 5.8 and 5.9 are concerned with finding the solution, or solutions, of the system

$$\begin{aligned} f_1(x_1,x_2,\cdots,x_n) &= 0,\\ f_2(x_1,x_2,\cdots,x_n) &= 0,\\ &\cdots\\ f_n(x_1,x_2,\cdots,x_n) &= 0, \end{aligned} \tag{5.33}$$

involving n real functions of the n real variables $x_1, x_2, \ldots, x_n$. Following the previous notation, $\mathbf{x} = [x_1,x_2,\ldots,x_n]^t$, we shall write $f_i(\mathbf{x}) = f_i(x_1,x_2,\ldots,x_n)$. Here, and in the subsequent development, $1 \leqslant i \leqslant n$. Then let $\boldsymbol{\alpha} = [\alpha_1,\alpha_2,\ldots,\alpha_n]^t$ be a solution of (5.33), that is, let $f_i(\boldsymbol{\alpha}) = 0$.

Let the n functions $F_i(\mathbf{x})$ be such that

$$x_i = F_i(\mathbf{x}) \tag{5.34}$$

implies $f_j(\mathbf{x}) = 0$, $1 \leqslant j \leqslant n$. Basically, the n equations (5.34) will constitute a suitable rearrangement of the original system (5.33). In particular, let

$$\alpha_i = F_i(\boldsymbol{\alpha}). \tag{5.35}$$

Let the starting vector $\mathbf{x}_0 = [x_{10},x_{20},\ldots,x_{n0}]^t$ be an approximation to $\boldsymbol{\alpha}$. Define successive new estimates of the solution vector, $\mathbf{x}_k = [x_{1k},x_{2k}\ldots,x_{nk}]^t$, $k = 1, 2, \ldots$, by computing the individual elements from the recursion relations

$$x_{ik} = F_i(x_{1,k-1},x_{2,k-1},\ldots,x_{n,k-1}) = F_i(\mathbf{x}_{k-1}). \tag{5.36}$$

Suppose there is a region R describable as $|x_j - \alpha_j| \leqslant h$, $1 \leqslant j \leqslant n$, and for $\mathbf{x}$ in R there is a positive number μ, less than one, such that

$$\sum_{j=1}^{n} \left| \frac{\partial F_i(\mathbf{x})}{\partial x_j} \right| \leqslant \mu. \tag{5.37}$$

Then, if the starting vector $\mathbf{x}_0$ lies in R, we show that the iterative method expressed by (5.36) converges to a solution of the system (5.33), that is,

$$\lim_{k\to\infty} \mathbf{x}_k = \boldsymbol{\alpha}. \tag{5.38}$$

Using the mean-value theorem, the truth of (5.38) is established by first noting from (5.35) and (5.36), that

$$\begin{aligned} x_{ik} - \alpha_i &= F_i(\mathbf{x}_{k-1}) - F_i(\boldsymbol{\alpha})\\ &= \sum_{j=1}^{n} (x_{j,k-1} - \alpha_j) \frac{\partial F_i[\boldsymbol{\alpha} + \xi_{i,k-1}(\mathbf{x}_{k-1} - \boldsymbol{\alpha})]}{\partial x_j}, \end{aligned} \tag{5.39}$$

in which $0 < \xi_{i,k-1} < 1$. That is,

$$|x_{ik} - \alpha_i| \leqslant h \sum_{j=1}^{n} \left| \frac{\partial F_i}{\partial x_j} \right| \leqslant \mu h < h,$$

showing that the points $\mathbf{x}_k$ lie in R. Also, by induction, from (5.37) and (5.39),

$$|x_{ik} - \alpha_i| \leqslant \mu \max_j (|x_{j,k-1} - \alpha_j|) \leqslant \mu^k h.$$

Therefore, (5.38) is true, and the procedure converges to a solution of (5.33). Note that if the $F_i(\mathbf{x})$ are linear, we have the Jacobi method, and the sufficient conditions of (5.37) are the same as the second set of sufficient conditions in (5.21).

For the nonlinear equations, there is also a counterpart to the Gauss-Seidel method, previously used in Sec. 5.7 for the linear case. We proceed as before, except that (5.36) is replaced by

$$x_{ik} = F_i(x_{1k},x_{2k},\ldots,x_{i-1,k},x_{i,k-1},\ldots,x_{n,k-1}).$$

That is, the most recently computed elements of the solution vector are always used in evaluating the F_i. The proof of convergence according to (5.38) is much the same as for the Jacobi-type iteration. We have

$$x_{1k} - \alpha_1 = \sum_{j=1}^{n} (x_{j,k-1} - \alpha_j) \frac{\partial F_1(\boldsymbol{\varepsilon}_{1k})}{\partial x_j},$$

where

$$\boldsymbol{\varepsilon}_{1k} = [\alpha_1 + \xi_{1k}(x_{1,k-1} - \alpha_1),\ldots,\alpha_n + \xi_{1k}(x_{n,k-1} - \alpha_n)]^t.$$

It will appear inductively that the above is true, because the various points concerned remain in R. If e_{k-1} is the largest of the numbers $|x_{j,k-1} - \alpha_j|$, then $|x_{1k} - \alpha_1| \leqslant \mu e_{k-1} < e_{k-1} < h$. It follows that

$$\begin{aligned} x_{2k} - \alpha_2 = (x_{1k} - \alpha_1) &\frac{\partial F_2(\boldsymbol{\varepsilon}_{2k})}{\partial x_1}\\ &+ \sum_{j=2}^{n} (x_{j,k-1} - \alpha_j) \frac{\partial F_2(\boldsymbol{\varepsilon}_{2k})}{\partial x_j}, \end{aligned}$$

where $\boldsymbol{\varepsilon}_{2k} = [\alpha_1 + \xi_{21}(x_{1k} - \alpha_1), \alpha_2 + \xi_{2k}(x_{2,k-1} - \alpha_2),\ldots, \alpha_n + \xi_{2k}(x_{n,k-1} - \alpha_n)]^t$. That is, $|x_{2k} - \alpha_2| \leqslant \mu e_{k-1} < e_{k-1} < h$, and, in general, $|x_{ik} - \alpha_i| \leqslant \mu e_{k-1} < e_{k-1} < h$. Therefore, $|x_{ik} - \alpha_i| \leqslant \mu^k h$, and convergence according to (5.38) is again established. Observe that the first of the sufficiency conditions of (5.28) has been reaffirmed under slightly more general circumstances.

Example. To illustrate the above techniques, choose the equations

$$f_1(x_1,x_2) = \frac{1}{2}\sin(x_1x_2) - \frac{x_2}{4\pi} - \frac{x_1}{2} = 0,$$

$$f_2(x_1,x_2) = \left(1 - \frac{1}{4\pi}\right)(e^{2x_1} - e) + \frac{ex_2}{\pi} - 2ex_1 = 0.$$

Rewrite these equations in a form which is consistent with (5.34),

$$x_1 = F_1(x_1,x_2) = \sin(x_1x_2) - \frac{x_2}{2\pi},$$

$$x_2 = F_2(x_1,x_2) = 2\pi x_1 - \left(\pi - \frac{1}{4}\right)(e^{2x_1-1} - 1),$$

and choose the starting values $x_{10} = 0.4$, $x_{20} = 3.0$. Within slide-rule accuracy, the Jacobi-type iteration gives

$$x_{11} = \sin(1.2) - \frac{3}{2\pi} = 0.455,$$

$$x_{21} = 2\pi \times 0.4 - 2.89(e^{-0.2} - 1) = 3.03;$$

$$x_{12} = \sin(1.379) - \frac{3.03}{2\pi} = 0.499,$$

$$x_{22} = 2\pi \times 0.455 - 2.89(e^{-0.09} - 1) = 3.11;$$

similarly, $x_{13} = 0.505$, $x_{23} = 3.14$; $x_{14} = 0.500$, $x_{24} = 3.14$; $x_{15} = 0.500$, $x_{25} = \pi$.

Using the same arrangement of the equations in conjunction with the same starting values, iteration of the Gauss-Seidel type gives

$$x_{11} = \sin(1.2) - \frac{3}{2\pi} = 0.455,$$

$$x_{21} = 2\pi \times 0.455 - 2.89(e^{-0.2} - 1) = 3.11;$$

$$x_{12} = \sin(1.415) - \frac{3.11}{2\pi} = 0.493,$$

$$x_{22} = 2\pi \times 0.493 - 2.89(e^{-0.014} - 1) = 3.14;$$

similarly, $x_{13} = 0.500$, $x_{23} = \pi$; $x_{14} = 0.500$, $x_{24} = \pi$, etc.

There is less risk involved in using an approximate slide-rule approach in these iterative calculations than might be supposed. Unlike the exact methods, such as Gaussian elimination for linear equations, there is no inherited round-off error from one step to the next. This follows, since the results at each stage of the iteration can be viewed as a new guess or initial approximation to the solution vector. Substantial error can be tolerated, provided that there is no gross error in the final stages of calculation. These remarks apply to the iterative solution of linear equations as well.

EXAMPLE 5.4

FLOW IN A PIPE NETWORK
SUCCESSIVE-SUBSTITUTION METHOD

Problem Statement

A network consists of a number of horizontal pipes, of specified diameters and lengths, that are joined at n nodes, numbered $i = 1, 2, \ldots, n$. The pressure is specified at some of these nodes. There is at most a single pipe connected directly between any two nodes.

Write a program that will accept information concerning the above, and that will proceed to compute: (a) the pressures at all remaining nodes, and (b) the flow rate (and direction of flow) in each pipe.

Method of Solution

For flow of a liquid from point i to point j in a horizontal pipe, the pressure drop is given by the *Fanning* equation:

$$p_i - p_j = \frac{1}{2} f_M \rho u_m^2 \frac{L}{D}. \tag{5.4.1}$$

Here, f_M is the dimensionless *Moody* friction factor, ρ is the liquid density, u_m is the mean velocity, and L and D are the length and diameter of the pipe, respectively. Since the volumetric flow rate is $Q = (\pi D^2/4)u_m$, equation (5.4.1) becomes

$$p_i - p_j = \frac{8 f_M \rho Q^2 L}{\pi^2 D^5}$$

Here, all quantities are in consistent units. However, if p_i and p_j are expressed in psi (lb_f/sq in.), ρ in lb_m/cu ft, Q in gpm (gallons/min), L in ft, and D in inches, we obtain

$$p_i - p_j = C \frac{L Q^2}{D^5}, \tag{5.4.2}$$

where

$$C = \frac{8 \times 12^5}{\pi^2 \times 144 \times 32.2 \times (7.48 \times 60)^2} f_M \rho. \tag{5.4.3}$$

Let $c_{ij} = CL_{ij}/D_{ij}^5$, where the subscripts ij now emphasize that we are concerned with the pipe joining nodes i and j. The flow rate Q_{ij} between nodes i and j is then given by

$$|p_i - p_j| = c_{ij} Q_{ij}^2, \tag{5.4.4}$$

in which Q_{ij} is plus or minus for flow from i to j or *vice versa*, respectively. In the following version, Q_{ij} will automatically have the correct sign:

$$Q_{ij} = (p_i - p_j)\sqrt{\frac{1}{c_{ij}|p_i - p_j|}}.$$

At any *free* node j, where the pressure is not specified, the sum of the flows from neighboring nodes i must be zero:

$$\sum_i Q_{ij} = \sum_i (p_i - p_j)\sqrt{\frac{1}{c_{ij}|p_i - p_j|}} = 0. \tag{5.4.5}$$

When applied at all the free nodes, equation (5.4.5) yields a system of nonlinear simultaneous equations in the unknown pressures. We shall solve this system by the successive-substitution type of method described in Section 5.8. First, note that $(p_i - p_j)$ is more sensitive than $(|p_i - p_j|)^{1/2}$ to variations in p_j. Thus an appropriate version, analogous to equation (5.3.4), is

$$p_j = \frac{\sum_i a_{ij} p_i}{\sum_i a_{ij}}, \tag{5.4.6}$$

in which

$$a_{ij} = (c_{ij}|p_i - p_j|)^{-1/2}. \tag{5.4.7}$$

Equation (5.4.6) is applied repeatedly at all free nodes until either each computed pressure p_j does not change by more than a small amount ε from one iteration to the next, or a preassigned number of iterations, *itmax*, has been exceeded. The most recently estimated values of p_i will always be used in the right-hand side of equation (5.4.6).

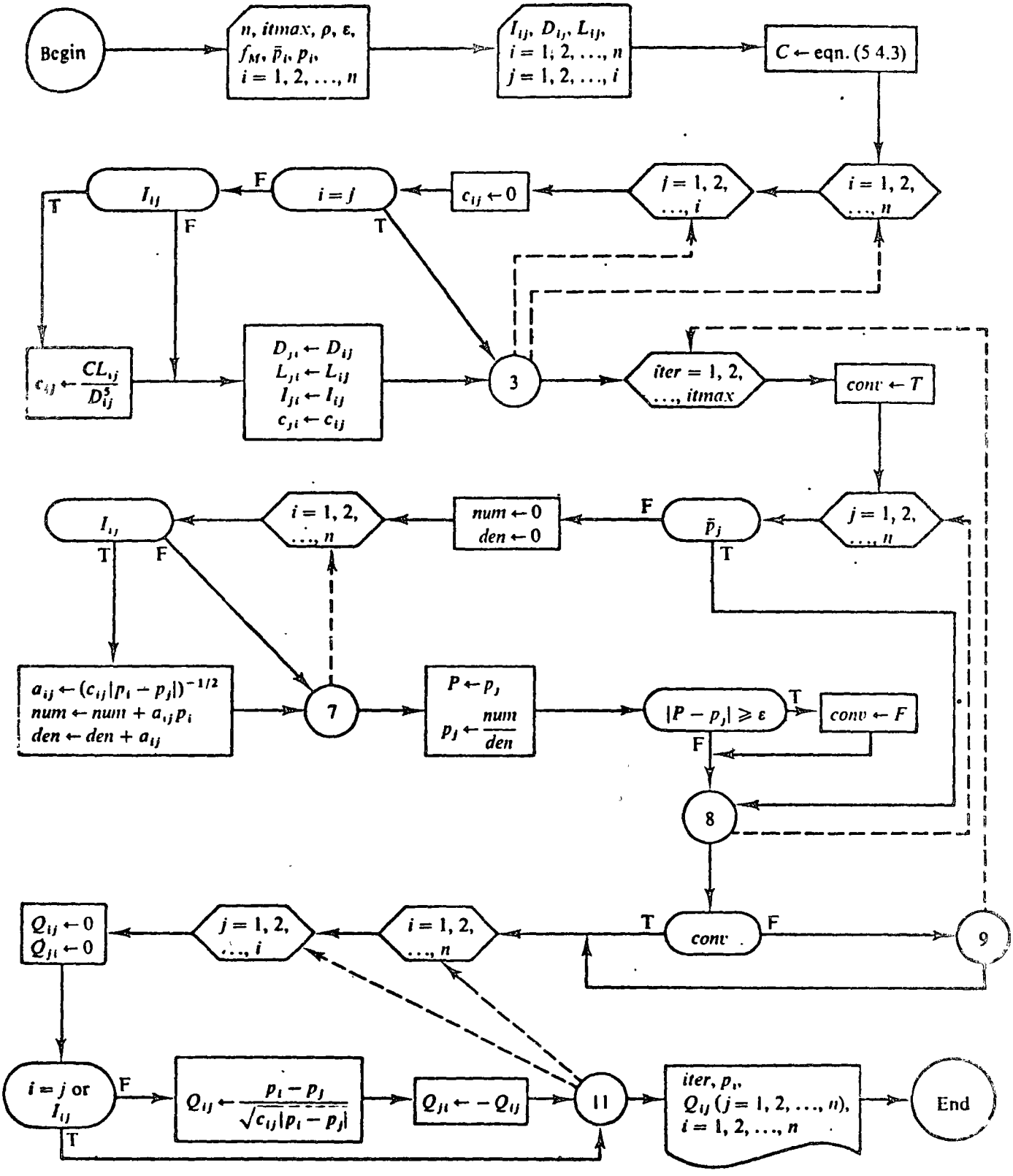
In order to implement the above, we also introduce the following:

(a) A vector of logical values, $\bar{p}_1, \bar{p}_2, \ldots, \bar{p}_n$ (PGIVEN in the program), such that $\bar{p}_j$ is true (T) if the pressure is specified at node j, and false (F) if it is not.

(b) A matrix of logical values, $I_{11} \ldots I_{nn}$ (the *incidence* matrix INCID in the program), such that I_{ij} is true if there is a pipe directly joining nodes i and j, and false if not.

Since the incidence, diameter, and length matrices are symmetric (for example, $D_{ji} = D_{ij}$), we need supply only the lower triangular portions of such matrices as data. The input data will also include a complete set of pressures, $p_1, p_2, \ldots, p_n$; some of these will be the known pressures, and the remainder will be the starting guesses at the free nodes.

Flow Diagram



FORTRAN *Implementation*

List of Principal Variables

Program Symbol	*Definition*
A	Matrix, whose elements a_{ij} are defined by (5.4.7).
C	Matrix, whose elements c_{ij} relate the flow rate to the pressure drop in the pipe joining nodes i and j.
CONV	Logical variable used in testing for convergence, *conv*.
D, L	Matrices, whose elements D_{ij} and L_{ij} give the diameter (in.) and length (ft) of the pipe joining nodes i and j.
DENOM, NUMER	Storage for the denominator and numerator of (5.4.6), *den* and *num*, respectively.
EPS	Tolerance, ε, used in testing for convergence.
F	Moody friction factor, f_M (assumed constant).
FACTOR	The constant, C, in equation (5.4.3).
I, J	Indices for representing the nodes i and j.
INCID	Matrix of logical values, I; if I_{ij} is T, there is a pipe joining nodes i and j; if F, there is not.
ITER	Counter on the number of iterations, *iter*.
IPRINT	Logical variable, which must have the value T if intermediate approximations to the pressures are to be printed.
ITMAX	Upper limit on the number of iterations, *itmax*.
N	Total number of nodes, n.
P	Vector of pressures, p_i (psi), at each node.
PGIVEN	Vector of logical values, $\bar{p}_i$, at each node. If $\bar{p}_i$ is T, the pressure is specified at node i; if F, it is not.
Q	Matrix, whose elements Q_{ij} give the flow rate (gpm) from node i to node j.
RHO	Density of the liquid in the pipes, ρ (lb_m/cu ft).
SAVEP	Temporary storage for old pressure p_j during convergence testing, P.

Program Listing

```
C       APPLIED NUMERICAL METHODS, EXAMPLE 5.4
C       FLOW IN A PIPE NETWORK - SUCCESSIVE SUBSTITUTION METHOD
C
C       THIS PROGRAM READS A DESCRIPTION OF THE TOPOLOGY OF AN
C       ARBITRARY N NODE PIPE NETWORK WITH PRESSURES SPECIFIED AT
C       PARTICULAR NODES, AND THEN COMPUTES THE PRESSURES AT THE
C       REMAINING NODES AND THE INTER-NODAL FLOW RATES USING A METHOD
C       OF SUCCESSIVE SUBSTITUTIONS.  IF INCID(I,J) IS TRUE, THEN NODES
C       I AND J ARE CONNECTED BY A PIPE SEGMENT OF DIAMETER D(I,J)
C       INCHES AND LENGTH L(I,J) FEET.  IF PGIVEN(I) IS TRUE, THE
C       PRESSURE AT NODE I, P(I) PSI, IS FIXED. OTHERWISE, P(I) ASSUMES
C       SUCCESSIVE ESTIMATES OF THE PRESSURE AT NODE I.  RHO IS THE
C       FLUID DENSITY IN LB/CU FT AND F THE PIPE FRICTION FACTOR,
C       ASSUMED IDENTICAL FOR ALL PIPE SEGMENTS.  ITER IS THE ITERATION
C       COUNTER.  ITERATION IS STOPPED WHEN ITER EXCEEDS ITMAX OR WHEN
C       NO NODAL PRESSURE CHANGES BY AN AMOUNT GREATER THAN EPS PSI
C       BETWEEN TWO SUCCESSIVE ITERATIONS.  Q(I,J) IS THE FLOW RATE IN
C       GAL/MIN BETWEEN NODES I AND J.  WHEN Q(I,J) IS POSITIVE, FLUID
C       FLOWS FROM NODE I TO NODE J.  THE MATRICES A AND C ARE
C       DESCRIBED IN THE TEXT.  WHEN IPRINT IS TRUE, INTERMEDIATE
C       APPROXIMATIONS OF THE NODAL PRESSURES ARE PRINTED.
C
      LOGICAL IPRINT, PGIVEN, INCID, CONV
      REAL  L, NUMER
      DIMENSION P(10), PGIVEN(10), A(10,10), C(10,10), D(10,10),
     1   L(10,10), INCID(10,10), Q(10,10)
C
C     ..... READ DATA .....
    1 READ (5,100)  N, ITMAX, RHO, EPS, F, IPRINT, (PGIVEN(I), I=1,N)
      READ (5,101)  (P(I), I=1,N)
      DO 2   I=1,N
      READ (5,102)  (INCID(I,J), J=1,I)
      READ (5,101)  (D(I,J), J=1,I)
    2 READ (5,101)  (L(I,J), J=1,I)
C
C     ..... SET UP UPPER TRIANGULAR PARTS OF SYMMETRIC MATRICES D, L,
C           AND INCID AND COMPUTE ELEMENTS OF C MATRIX .....
      FACTOR = 8.*12.**5*RHO*F/(3.1415926**2*32.2*(60.*7.48)**2*144.)
      DO 3   I=1,N
      DO 3   J=1,I
      C(I,J) = 0.
      IF ( I.EQ.J )   GO TO 3
      IF ( INCID(I,J) )   C(I,J) = FACTOR*L(I,J)/D(I,J)**5
      D(J,I) = D(I,J)
      L(J,I) = L(I,J)
      INCID(J,I) = INCID(I,J)
      C(J,I) = C(I,J)
    3 CONTINUE
C
C     ..... PRINT OUT INITIAL INFORMATION ABOUT NETWORK .....
      WRITE (6,200)  N,ITMAX,RHO,EPS,F,IPRINT,(I,P(I),PGIVEN(I),I=1,N)
      WRITE (6,201)
      DO 4   I=1,N
    4 WRITE (6,202)  I, I, N, (INCID(I,J), J=1,N)
      WRITE (6,201)
      DO 5   I=1,N
    5 WRITE (6,203)  I, I, N, (D(I,J), J=1,N)
      WRITE (6,201)
      DO 6   I=1,N
    6 WRITE (6,204)  I, I, N, (L(I,J), J=1,N)
```

Program Listing (*Continued*)

```
C
C     ..... COMPUTE SUCCESSIVE ESTIMATES OF PRESSURES AT NODES .....
      IF ( IPRINT )  WRITE (6,205)  (I, I=1,N)
      DO 9  ITER=1,ITMAX
      CONV = .TRUE.
      DO 8  J=1,N
      IF ( PGIVEN(J) )  GO TO 8
      NUMER = 0.
      DENOM = 0.
      DO 7  I=1,N
      IF ( .NOT.INCID(I,J) )  GO TO 7
         A(I,J) = 1.0/SQRT(C(I,J)*ABS(P(I)-P(J)))
         NUMER = NUMER + A(I,J)*P(I)
         DENOM = DENOM + A(I,J)
    7 CONTINUE
      SAVEP = P(J)
      P(J) = NUMER/DENOM
      IF ( ABS(SAVEP-P(J)).GE.EPS )  CONV = .FALSE.
    8 CONTINUE
      IF ( IPRINT )  WRITE (6,206)  ITER, (P(I), I=1,N)
      IF ( CONV )  GO TO 10
    9 CONTINUE
      WRITE (6,207)  ITMAX
C
C     ..... COMPUTE FLOWS IN INDIVIDUAL NETWORK BRANCHES .....
   10 DO 11  I=1,N
      DO 11  J=1,I
      Q(I,J) = 0.
      Q(J,I) = 0.
      IF ( I.EQ.J .OR. .NOT.INCID(I,J) )  GO TO 11
      Q(I,J) = (P(I)-P(J))/SQRT(C(I,J)*ABS(P(I)-P(J)))
      Q(J,I) = - Q(I,J)
   11 CONTINUE
C
C     ..... PRINT FINAL PRESSURES AND FLOWS .....
      WRITE (6,208)  ITER, N
      DO 12  I=1,N
   12 WRITE (6,209)  I, P(I), (Q(I,J), J=1,N)
      GO TO 1
C
C     ..... FORMATS FOR INPUT AND OUTPUT STATEMENTS .....
  100 FORMAT( 3X,I2,17X,I3,15X,F5.1,15X,E5.0/ 4X,F6.3,14X,L1 /
     1   (30X, 20(L1,1X)) )
  101 FORMAT( 30X, 5F8.3 )
  102 FORMAT( 30X,20(L1,1X) )
  200 FORMAT( 23H1FLOW IN A PIPE NETWORK/ 10H0N      = , I3/ 10H ITMAX
     1= , I3/ 10H RHO    = , F7.3/ 10H EPS    = , E10.2/ 10H F      = ,
     2   F7.3/ 10H IPRINT = , 2X, L1/ 3H0 I, 6X, 4HP(I), 4X,9HPGIVEN(I)/
     3   (1H , I2, F10.3, 6X, L1) )
  201 FORMAT( 1H0/1H0 )
  202 FORMAT( 7H0INCID(, I2, 13H, 1)...INCID(, I2,1H,,I2, 3H) =
     1   40(L1,1X)/ (1H , 29X, 40(L1,1X)) )
  203 FORMAT( 3H0D(, I2, 9H, 1)...D(, I2,1H,,I2, 1H), 9X, 1H= , 8F10.3 /
     1   (1H , 29X, 8F10.3) )
  204 FORMAT( 3H0L(, I2, 9H, 1)...L(, I2,1H,,I2, 1H), 9X, 1H= , 8F10.3 /
     1   (1H , 29X, 8F10.3) )
  205 FORMAT( 1H0/ 5H0ITER, 7X, 16HPRESSURE AT NODE/ (1H ,11X,8(I1,9X)))
  206 FORMAT( 1H , I3, 3X, 8F10.4/ (1H , 6X, 8F10.4) )
  207 FORMAT( 35H0SOLUTIONS FAILED TO CONVERGE AFTER, I3,11H ITERATIONS)
  208 FORMAT( 1H0/26H0PRESSURES AND FLOWS AFTER, I3, 15H ITERATIONS ARE/
     1   3H0 I,5X,4HP(I),7X,16HQ( I, 1)...Q( I,,I2,1H)/ 1H ,7X, 3HPSI,
     2   14X, 7HGAL/MIN/ )
  209 FORMAT( 1H , I2, F10.4, 5X, 8F10.3/ (1H , 17X, 8F10.3) )
C
      END
```

Program Listing (*Continued*)

Data

```
N = 5              ITMAX =100          RHO = 50.0          EPS = 1.E-4
F =  0.056         IPRINT = T
PGIVEN(1)...PGIVEN(5)          =  T F T F F
P(1)...P(5)                    =    50.000   20.000    0.000   40.000   30.000
INCID(1,1)                     =  F
D(1,1)                         =     0.000
L(1,1)                         =     0.000
INCID(2,1)...INCID(2,2)        =  T F
D(2,1)...D(2,2)                =     3.000    0.000
L(2,1)...L(2,2)                =   150.000    0.000
INCID(3,1)...INCID(3,3)        =  F T F
D(3,1)...D(3,3)                =     0.000    3.000    0.000
L(3,1)...L(3,3)                =     0.000  150.000    0.000
INCID(4,1)...INCID(4,4)        =  T F F F
D(4,1)...D(4,4)                =     4.000    0.000    0.000    0.000
L(4,1)...L(4,4)                =   100.000    0.000    0.000    0.000
INCID(5,1)...INCID(5,5)        =  F T F T F
D(5,1)...D(5,5)                =     0.000    4.000    0.000    4.000    0.000
L(5,1)...L(5,5)                =     0.000  100.000    0.000  100.000    0.000
```

Computer Output

```
FLOW IN A PIPE NETWORK

N      =   5
ITMAX  = 100
RHO    =  50.000
EPS    =   0.10E-03
F      =   0.056
IPRINT =   T

 I      P(I)      PGIVEN(I)
 1    50.000        T
 2    20.000        F
 3     0.0          T
 4    40.000        F
 5    30.000        F


INCID( 1, 1)...INCID( 1, 5) =F T F T F

INCID( 2, 1)...INCID( 2, 5) =T F T F T

INCID( 3, 1)...INCID( 3, 5) =F T F F F

INCID( 4, 1)...INCID( 4, 5) =T F F F T

INCID( 5, 1)...INCID( 5, 5) =F T F T F


D( 1, 1)...D( 1, 5)    =    0.0     3.000    0.0     4.000    0.0

D( 2, 1)...D( 2, 5)    =    3.000   0.0      3.000   0.0      4.000

D( 3, 1)...D( 3, 5)    =    0.0     3.000    0.0     0.0      0.0

D( 4, 1)...D( 4, 5)    =    4.000   0.0      0.0     0.0      4.000

D( 5, 1)...D( 5, 5)    =    0.0     4.000    0.0     4.000    0.0
```

Computer Output (*Continued*)

```
L( 1, 1)...L( 1, 5)  =    0.0     150.000     0.0     100.000     0.0
L( 2, 1)...L( 2, 5)  =  150.000     0.0     150.000     0.0     100.000
L( 3, 1)...L( 3, 5)  =    0.0     150.000     0.0       0.0       0.0
L( 4, 1)...L( 4, 5)  =  100.000     0.0       0.0       0.0     100.000
L( 5, 1)...L( 5, 5)  =    0.0     100.000     0.0     100.000     0.0
```

ITER	PRESSURE AT NODE 1	2	3	4	5
1	50.0000	27.4553	0.0	40.0000	31.6616
2	50.0000	30.3218	0.0	40.4145	33.1600
3	50.0000	31.9882	0.0	40.9944	34.4999
4	50.0000	33.2753	0.0	41.6180	35.7211
5	50.0000	34.3876	0.0	42.2345	36.8321
6	50.0000	35.3809	0.0	42.8204	37.8350
7	50.0000	36.2724	0.0	43.3644	38.7339
8	50.0000	37.0706	0.0	43.8616	39.5349
9	50.0000	37.7819	0.0	44.3111	40.2451
10	50.0000	38.4132	0.0	44.7139	40.8726
11	50.0000	38.9714	0.0	45.0728	41.4253
12	50.0000	39.4634	0.0	45.3909	41.9110
13	50.0000	39.8960	0.0	45.6718	42.3370
14	50.0000	40.2756	0.0	45.9191	42.7100
15	50.0000	40.6081	0.0	46.1362	43.0361
16	50.0000	40.8989	0.0	46.3266	43.3211
17	50.0000	41.1530	0.0	46.4931	43.5698
18	50.0000	41.3748	0.0	46.6387	43.7867
19	50.0000	41.5682	0.0	46.7658	43.9758
20	50.0000	41.7368	0.0	46.8767	44.1406
21	50.0000	41.8837	0.0	46.9734	44.2841
22	50.0000	42.0117	0.0	47.0576	44.4090
23	50.0000	42.1230	0.0	47.1310	44.5178
24	50.0000	42.2200	0.0	47.1949	44.6125
25	50.0000	42.3043	0.0	47.2505	44.6948
26	50.0000	42.3777	0.0	47.2989	44.7665
27	50.0000	42.4416	0.0	47.3411	44.8288
28	50.0000	42.4971	0.0	47.3777	44.8831
29	50.0000	42.5454	0.0	47.4096	44.9302
30	50.0000	42.5874	0.0	47.4373	44.9713
31	50.0000	42.6239	0.0	47.4615	45.0070
32	50.0000	42.6557	0.0	47.4825	45.0380
33	50.0000	42.6833	0.0	47.5007	45.0650
34	50.0000	42.7073	0.0	47.5166	45.0884
35	50.0000	42.7281	0.0	47.5303	45.1088
36	50.0000	42.7463	0.0	47.5424	45.1265
37	50.0000	42.7621	0.0	47.5528	45.1420
38	50.0000	42.7758	0.0	47.5619	45.1554
39	50.0000	42.7878	0.0	47.5697	45.1670
40	50.0000	42.7981	0.0	47.5766	45.1772
41	50.0000	42.8071	0.0	47.5826	45.1860
42	50.0000	42.8150	0.0	47.5878	45.1937
43	50.0000	42.8218	0.0	47.5923	45.2003
44	50.0000	42.8277	0.0	47.5962	45.2061
45	50.0000	42.8329	0.0	47.5996	45.2112
46	50.0000	42.8374	0.0	47.6026	45.2155
47	50.0000	42.8413	0.0	47.6051	45.2193
48	50.0000	42.8447	0.0	47.6074	45.2227
49	50.0000	42.8476	0.0	47.6093	45.2255
50	50.0000	42.8502	0.0	47.6110	45.2280
51	50.0000	42.8524	0.0	47.6125	45.2302
52	50.0000	42.8543	0.0	47.6138	45.2321
53	50.0000	42.8560	0.0	47.6149	45.2337
54	50.0000	42.8574	0.0	47.6158	45.2352
55	50.0000	42.8587	0.0	47.6167	45.2366

ulter Output (*Continued*)

```
 56      50.0000    42.8598     0.0       47.6174    45.2375
 57      50.0000    42.8608     0.0       47.6181    45.2385
 58      50.0000    42.8616     0.0       47.6186    45.2393
 59      50.0000    42.8624     0.0       47.6191    45.2400
 60      50.0000    42.8630     0.0       47.6195    45.2406
 61      50.0000    42.8636     0.0       47.6199    45.2411
 62      50.0000    42.8640     0.0       47.6202    45.2416
 63      50.0000    42.8644     0.0       47.6205    45.2420
 64      50.0000    42.8648     0.0       47.6207    45.2424
 65      50.0000    42.8651     0.0       47.6209    45.2427
 66      50.0000    42.8654     0.0       47.6211    45.2429
 67      50.0000    42.8656     0.0       47.6212    45.2432
 68      50.0000    42.8658     0.0       47.6214    45.2434
 69      50.0000    42.8660     0.0       47.6215    45.2435
 70      50.0000    42.8662     0.0       47.6216    45.2437
 71      50.0000    42.8663     0.0       47.6217    45.2438
 72      50.0000    42.8664     0.0       47.6218    45.2439
 73      50.0000    42.8665     0.0       47.6218    45.2440
 74      50.0000    42.8666     0.0       47.6219    45.2441

PRESSURES AND FLOWS AFTER 74 ITERATIONS ARE

 I        P(I)          Q( I, 1)...Q( I, 5)
          PSI                GAL/MIN
 1     50.0000            0.0       138.242      0.0       200.677     0.0
 2     42.8666         -138.242       0.0       338.885       0.0    -200.654
 3      0.0               0.0      -338.885      0.0         0.0       0.0
 4     47.6219         -200.677       0.0         0.0         0.0     200.664
 5     45.2441            0.0       200.654      0.0      -200.664     0.0
```

Discussion of Results

The data used above relate to the network shown in Fig. 5.4.1, with $f_M = 0.056$, $\rho = 50$ lb_m/cu ft, and two pressures fixed: $p_1 = 50$, $p_3 = 0$ psi.

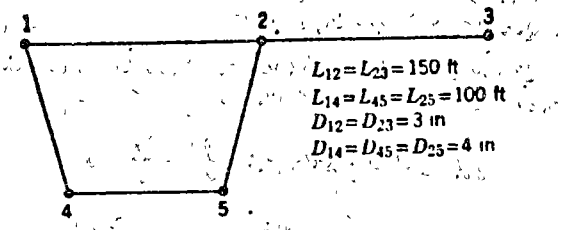


Figure 5.4.1 Pipe network for example calculation.

Although the method is computationally straightforward, it needs many iterations to give a reasonable degree of convergence. Also, referring to equation (5.4.7), we can see that a starting guess of $p_i = p_j$ for any two nodes that are directly connected would be unfortunate.

Note that the bulk of the pressure drop occurs in the pipe 2-3, and that the flow in the branch 1-4-5-2 is appreciably greater than that in the pipe 1-2, even though the latter is much shorter. Both these observations can be reconciled by noting that pressure drop is proportional to Q^2/D^5, and that pipe 2-3 must take the combined flows along 1-4-5-2 and 1-2.

The method can be extended to more complex situations, in which we could allow for (a) f_M being a function of Reynolds number and pipe roughness, instead of being treated as a constant, and (b) pumps and valves in some of the branches, etc. Also, the logical arrays used above could find a similar application in solving for the currents in a network of resistors, with known voltages applied at some of the nodes (although this would lead to a set of simultaneous *linear* equations).

5.9 Newton-Raphson Iteration for Nonlinear Equations

The equations to be solved are again those of (5.33), and we retain the nomenclature of the previous section. The Newton-Raphson process, to be described, is once more iterative in character. We first define

$$f_{ij}(\mathbf{x}) = \frac{\partial f_i(\mathbf{x})}{\partial x_j}. \tag{5.40}$$

Next define the matrix $\phi(\mathbf{x})$ as

$$\phi(\mathbf{x}) = (f_{ij}(\mathbf{x})). \qquad 1 \leqslant i \leqslant n,\ 1 \leqslant j \leqslant n. \tag{5.41}$$

Thus $\det(\phi(\mathbf{x}))$ is the *Jacobian* of the system (5.33) evaluated for the vector $\mathbf{x} = [x_1, x_2, \ldots, x_n]'$. Now define the vector $\mathbf{f}(\mathbf{x})$ as

$$\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_n(\mathbf{x})]'. \tag{5.42}$$

With these definitions in mind, and with the starting vector $\mathbf{x}_0 = [x_{10}, x_{20}, \ldots, x_{n0}]'$, let

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \delta_k, \tag{5.43}$$

where δ_k is the solution vector for the set of simultaneous linear equations

$$\phi(\mathbf{x}_k)\,\delta_k = -\mathbf{f}(\mathbf{x}_k). \tag{5.44}$$

The fundamental theorem concerning convergence is much less restrictive than those of the previous sections. We have the result that if the components of $\phi(\mathbf{x})$ are continuous in a neighborhood of a point α such that $\mathbf{f}(\alpha) = 0$, if $\det(\phi(\alpha)) \neq 0$, and if $\mathbf{x}_0$ is "near" α, then $\lim_{k\to\infty} \mathbf{x}_k = \alpha$.

An outline for a method of proof follows. By (5.42) and (5.44), since $f_i(\alpha) = 0$,

$$\delta_k = \phi^{-1}(\mathbf{x}_k)[\mathbf{f}(\alpha) - \mathbf{f}(\mathbf{x}_k)]. \tag{5.45}$$

By the mean-value theorem,

$$f_i(\mathbf{x}_k) - f_i(\alpha) = \sum_{j=1}^{n} f_{ij}(\alpha + \xi_{ik}(\mathbf{x}_k - \alpha))(x_{jk} - \alpha_j),$$

where $0 < \xi_{ik} < 1$. For the ith row of a matrix ψ use

$$[f_{i1}(\alpha + \xi_{ik}(\mathbf{x}_k - \alpha)), \ldots, f_{in}(\alpha + \xi_{ik}(\mathbf{x}_k - \alpha))].$$

Then

$$\mathbf{x}_{k+1} - \alpha = \mathbf{x}_k - \alpha + \delta_k = \phi^{-1}(\mathbf{x}_k)[\phi(\mathbf{x}_k) - \psi](\mathbf{x}_k - \alpha).$$

Since the entries in the matrix $\phi(\mathbf{x}_k) - \psi$ are differences of the type $f_{ij}(\mathbf{x}_k) - f_{ij}(\alpha + \xi_{ik}(\mathbf{x}_k - \alpha))$, they can be kept uniformly small if the starting vector $\mathbf{x}_0$ lies in an initially chosen region R describable as $|x_i - \alpha_i| \leqslant h$, $1 \leqslant i \leqslant n$. Concurrent with this is the fact that since $\det(\phi(\mathbf{x})) \neq 0$, $\det(\phi(\mathbf{x}_k))$ can be bounded from zero. The net result is that, for $0 < \mu < 1$, $|x_{ik} - \alpha_i| \leqslant h\mu^k$, $1 \leqslant i \leqslant n$. Thus the sequence $\{\mathbf{x}_k\}$ converges to α.

Example. To illustrate the procedure, the equations of the previous section are used, namely

$$f_1(x_1, x_2) = \frac{1}{2}\sin(x_1 x_2) - \frac{x_2}{4\pi} - \frac{x_1}{2} = 0$$

$$f_2(x_1, x_2) = \left(1 - \frac{1}{4\pi}\right)(e^{2x_1} - e) + \frac{e x_2}{\pi} - 2e x_1 = 0.$$

It is readily seen that

$$\frac{\partial f_1}{\partial x_1} = -\frac{1}{2} + \frac{x_2 \cos(x_1 x_2)}{2}, \qquad \frac{\partial f_1}{\partial x_2} = -\frac{1}{4\pi} + \frac{x_1 \cos(x_1 x_2)}{2},$$

$$\frac{\partial f_2}{\partial x_1} = -2e + \left(2 - \frac{1}{2\pi}\right)e^{2x_1}, \qquad \frac{\partial f_2}{\partial x_2} = \frac{e}{\pi}.$$

The increments Δx_1 and Δx_2 in x_1 and x_2 are determined by

$$\frac{\partial f_1}{\partial x_1}\Delta x_1 + \frac{\partial f_1}{\partial x_2}\Delta x_2 = -f_1,$$

$$\frac{\partial f_2}{\partial x_1}\Delta x_1 + \frac{\partial f_2}{\partial x_2}\Delta x_2 = -f_2.$$

Or, writing the determinant D of the coefficient matrix (the Jacobian),

$$D = \frac{\partial f_1}{\partial x_1}\frac{\partial f_2}{\partial x_2} - \frac{\partial f_1}{\partial x_2}\frac{\partial f_2}{\partial x_1},$$

then

$$\Delta x_1 = \left(\frac{f_2 \dfrac{\partial f_1}{\partial x_2} - f_1 \dfrac{\partial f_2}{\partial x_2}}{D}\right), \quad \Delta x_2 = \left(\frac{f_1 \dfrac{\partial f_2}{\partial x_1} - f_2 \dfrac{\partial f_1}{\partial x_1}}{D}\right).$$

For ease in verification, detailed results are tabulated in Table 5.1. Once again, calculations have been carried out by slide rule. The entries −0.0000 designate tiny negative values.

Table 5.1 Newton-Raphson Solution for $\mathbf{x}_0 = \begin{bmatrix} 0.4 \\ 3.0 \end{bmatrix}$

k	x_1	x_2	f_1	f_2	$\frac{\partial f_1}{\partial x_1}$	$\frac{\partial f_1}{\partial x_2}$	$\frac{\partial f_2}{\partial x_1}$	$\frac{\partial f_2}{\partial x_2}$	D	Δx_1	Δx_2
0	0 400	3 000	0.0272	−0.0324	0.0435	−0.0071	−1.34	0.865	0 0281	−0 831	−1.249
1	−0.431	1.751	−0.266	1.74	0.138	−0 236	−4.66	0.865	−0.982	0.186	−1.018
2	−0.245	0 733	−0.0251	0.0303	−0.139	−0.200	−4.31	0.865	−0.984	−0.016	−0.114
3	−0.261	0.619	0.0009	0 0003	−0.195	−0.208	−4.35	0.865	−1.07	0.0007	0.003
4	−0.260	0 622	0.0000	0 0000	−0.193	−0.208	−4.34	0.865	−1.07	0.0000	0.0000
5	−0.260	0 622	0.0000	0.0000							

Note that despite using the same initial value, this solution differs from that obtained in Section 5.8. However, the starting values $x_{10} = 0.6$, $x_{20} = 3.0$ do lead to the alternative solution $x_1 = 0.5$, $x_2 = \pi$. Values are given in Table 5.2.

Table 5.2 Newton-Raphson Solution for $\mathbf{x}_0 = \begin{bmatrix} 0.6 \\ 3.0 \end{bmatrix}$

k	x_1	x_2	f_1	f_2	$\frac{\partial f_1}{\partial x_1}$	$\frac{\partial f_1}{\partial x_2}$	$\frac{\partial f_2}{\partial x_1}$	$\frac{\partial f_2}{\partial x_2}$	D	Δx_1	Δx_2
1	0.600	3.000	−0.0518	−0.112	−0.841	−0.148	0.675	0.865	−0.627	−0.098	0.206
2	0.502	3.206	−0.0066	0.0549	−0.563	−0.0894	−0.411	0.865	−0.524	−0.001	−0.064
3	0.501	3.142	−0.0003	0.0000	−0.503	−0.0801	−0.426	0.865	−0.470	−0.0006	−0.0004
4	0.500	3.142	−0.0000	−0.0000	−0.500	−0.0796	−0.433	0.865	−0.467	−0.0000	−0.0000
5	0.500	3.142	−0.0000	0.0000							

EXAMPLE 5.5

CHEMICAL EQUILIBRIUM NEWTON-RAPHSON METHOD

Problem Statement

The principal reactions in the production of synthesis gas by partial oxidation of methane with oxygen are:

$$CH_4 + \tfrac{1}{2}O_2 \rightleftharpoons CO + 2H_2 \qquad (5.5.1)$$

$$CH_4 + H_2O \rightleftharpoons CO + 3H_2 \qquad (5.5.2)$$

$$H_2 + CO_2 \rightleftharpoons CO + H_2O \qquad (5.5.3)$$

Write a program that finds the O_2/CH_4 reactant ratio that will produce an adiabatic equilibrium temperature of 2200 F at an operating pressure of 20 atmospheres, when the reactant gases are preheated to an entering temperature of 1000°F.

Assuming that the gases behave ideally, so that the component activities are identical with component partial pressures, the equilibrium constants at 2200°F for the three equations are respectively:

$$K_1 = \frac{p_{CO}p_{H_2}^2}{p_{CH_4}p_{O_2}^{1/2}} = 1.3 \times 10^{11}, \qquad (5.5.4)$$

$$K_2 = \frac{p_{CO}p_{H_2}^3}{p_{CH_4}p_{H_2O}} = 1.7837 \times 10^5, \qquad (5.5.5)$$

$$K_3 = \frac{p_{CO}p_{H_2O}}{p_{CO_2}p_{H_2}} = 2.6058. \qquad (5.5.6)$$

where p_{CO}, p_{CO_2}, p_{H_2O}, p_{H_2}, p_{CH_4} and p_{O_2} are the partial pressures of CO (carbon monoxide), CO_2 (carbon dioxide), H_2O (water vapor), H_2 (hydrogen), CH_4 (methane), and O_2 (oxygen), respectively.

Enthalpies of the various components at 1000°F and 2200°F are listed in Table 5.5.1.

Table 5.5.1 Component Enthalpies in BTU/lb mole

Component	1000°F	2200°F
CH_4	−13492	8427
H_2O	−90546	−78213
CO_2	−154958	−139009
CO	−38528	−28837
H_2	10100	18927
O_2	10690	20831

A fourth reaction may also occur at high temperatures:

$$C + CO_2 \rightleftharpoons 2CO \qquad (5.5.7)$$

At 2200°F, any carbon formed would be deposited as a solid; the equilibrium constant is given by

$$K_4 = \frac{p_{CO}^2}{a_C p_{CO_2}} = 1329.5, \qquad (5.5.8)$$

where a_C is the activity of carbon in the solid state. Do not include reaction (5.5.7) in the equilibrium analysis. After establishing the equilibrium composition, considering only the homogeneous gaseous reactions given by (5.5.1), (5.5.2), and (5.5.3), determine the thermodynamic likelihood that solid carbon would appear as a result of reaction (5.5.7). Assume that the activity of solid carbon is unaffected by pressure and equals unity.

Use the Newton-Raphson method to solve the system of simultaneous nonlinear equations developed as the result of the equilibrium analysis.

Method of Solution

Because of the magnitude of K_1, the equilibrium constant for reaction (5.1.1), the first reaction can be assumed to go to completion at 2200°F, that is, virtually no unreacted oxygen will remain in the product gases at equilibrium.

Let the following nomenclature be used:

- x_1 mole fraction of CO in the equilibrium mixture
- x_2 mole fraction of CO_2 in the equilibrium mixture
- x_3 mole fraction of H_2O in the equilibrium mixture
- x_4 mole fraction of H_2 in the equilibrium mixture
- x_5 mole fraction of CH_4 in the equilibrium mixture
- x_6 number of moles of O_2 per mole of CH_4 in the feed gases
- x_7 number of moles of product gases in the equilibrium mixture per mole of CH_4 in the feed gases.

Then a system of seven simultaneous equations may be generated from three atom balances, an energy balance, a mole fraction constraint, and two equilibrium relations.

Atom conservation balances. The number of atoms of each element entering equals the number of atoms of each element in the equilibrium mixture.

$$\text{Oxygen: } x_6 = (\tfrac{1}{2}x_1 + x_2 + \tfrac{1}{2}x_3)x_7. \qquad (5.5.9)$$

$$\text{Hydrogen: } 4 = (2x_3 + 2x_4 + 4x_5)x_7. \qquad (5.5.10)$$

$$\text{Carbon: } 1 = (x_1 + x_2 + x_5)x_7. \qquad (5.5.11)$$

Energy (enthalpy) balance. Since the reaction is to be conducted adiabatically, that is, no energy is added to

or removed from the reacting gases, the enthalpy (H) of the reactants must equal the enthalpy of the products.

$$[H_{CH_4} + x_6 H_{O_2}]_{1000°F} = x_7[x_1 H_{CO} + x_2 H_{CO_2} + x_3 H_{H_2O} + x_4 H_{H_2} + x_5 H_{CH_4}]_{2200°F}. \quad (5.5.12)$$

Mole fraction constraint.

$$x_1 + x_2 + x_3 + x_4 + x_5 = 1. \quad (5.5.13)$$

Equilibrium relations.

$$K_2 = \frac{P^2 x_1 x_4^3}{x_3 x_5} = 1.7837 \times 10^5, \quad (5.5.14)$$

$$K_3 = \frac{x_1 x_3}{x_2 x_4} = 2.6058. \quad (5.5.15)$$

The relationships (5.5.14) and (5.5.15) follow directly from (5.5.5) and (5.5.6), respectively, where P is the total pressure and $p_{CO} = Px_1$, etc. In addition, there are five side conditions,

$$x_i \geq 0, \qquad i = 1, 2, \ldots, 5. \quad (5.5.16)$$

These conditions insure that all mole fractions in the equilibrium mixture are nonnegative, that is, any solution of equations (5.5.9) to (5.5.15) that contains negative mole fractions is physically meaningless. From physical-chemical principles, there is one and only one solution of the equations that satisfies conditions (5.5.16). Any irrelevant solutions may be detected easily.

The seven equations may be rewritten in the form

$$f_i(\mathbf{x}) = 0, \qquad i = 1, 2, \ldots, 7, \quad (5.5.17)$$

where $\mathbf{x} = [x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7]'$, as follows:

$$f_1(\mathbf{x}) = \frac{1}{2}x_1 + x_2 + \frac{1}{2}x_3 - \frac{x_6}{x_7} = 0 \quad (5.5.18a)$$

$$f_2(\mathbf{x}) = x_3 + x_4 + 2x_5 - \frac{2}{x_7} = 0 \quad (5.5.18b)$$

$$f_3(\mathbf{x}) = x_1 + x_2 + x_5 - \frac{1}{x_7} = 0 \quad (5.5.18c)$$

$$f_4(\mathbf{x}) = -28837x_1 - 139009x_2 - 78213x_3 + 18927x_4 + 8427x_5 + \frac{13492}{x_7} - 10690\frac{x_6}{x_7} = 0 \quad (5.5.18d)$$

$$f_5(\mathbf{x}) = x_1 + x_2 + x_3 + x_4 + x_5 - 1 = 0 \quad (5.5.18e)$$

$$f_6(\mathbf{x}) = P^2 x_1 x_4^3 - 1.7837 \times 10^5 x_3 x_5 = 0 \quad (5.5.18f)$$

$$f_7(\mathbf{x}) = x_1 x_3 - 2.6058 x_2 x_4 = 0. \quad (5.5.18g)$$

The system of simultaneous nonlinear equations has the form of (5.33), and will be solved using the Newton-Raphson method, described in Section 5.9. The partial derivatives of (5.40) may be found by partial differentiation of the seven functions, $f_i(\mathbf{x})$, with respect to each of the seven variables. For example,

$$\frac{\partial f_1}{\partial x_1} = \frac{1}{2}, \quad \frac{\partial f_1}{\partial x_4} = 0, \quad \frac{\partial f_1}{\partial x_7} = \frac{x_6}{x_7^2},$$
$$\frac{\partial f_1}{\partial x_2} = 1, \quad \frac{\partial f_1}{\partial x_5} = 0, \quad (5.5.19)$$
$$\frac{\partial f_1}{\partial x_3} = \frac{1}{2}, \quad \frac{\partial f_1}{\partial x_6} = -\frac{1}{x_7},$$

The Newton-Raphson method may be summarized as follows:

1. Choose a starting vector $\mathbf{x}_k = \mathbf{x}_0 = [x_{10}, x_{20}, \ldots, x_{70}]$, where $\mathbf{x}_0$ is hopefully near a solution α.
2. Solve the system of linear equations (5.44),

$$\phi(\mathbf{x}_k)\delta_k = -\mathbf{f}(\mathbf{x}_k),$$

where

$$\phi_{ij}(\mathbf{x}_k) = \frac{\partial f_i}{\partial x_j}(\mathbf{x}_k), \qquad \begin{matrix} i = 1, 2, \ldots, 7, \\ j = 1, 2, \ldots, 7, \end{matrix} \quad (5.5.20)$$

and

$$\mathbf{f}(\mathbf{x}_k) = [f_1(\mathbf{x}_k), f_2(\mathbf{x}_k), \ldots, f_7(\mathbf{x}_k)]', \quad (5.5.21)$$

for the increment vector

$$\delta_k = [\delta_{1k}, \delta_{2k}, \ldots, \delta_{7k}]'. \quad (5.5.22)$$

3. Update the approximation to the root for the next iteration.

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \delta_k.$$

4. Check for possible convergence to a root α. One such test might be

$$|\delta_{ik}| < \varepsilon_2, \qquad i = 1, 2, \ldots, 7. \quad (5.5.23)$$

If (5.5.23) is true for all i, then $\mathbf{x}_{k+1}$ is taken to be the root. If test (5.5.23) is failed for any i, then the process is repeated starting with step 2. The iterative process is continued until test (5.5.23) is passed for some k, or when k exceeds some specified upper limit.

In the programs that follow, the elements of the augmented matrix

$$\mathbf{A} = [\phi(\mathbf{x}_k) \,\vdots\, -\mathbf{f}(\mathbf{x}_k)] \quad (5.5.24)$$

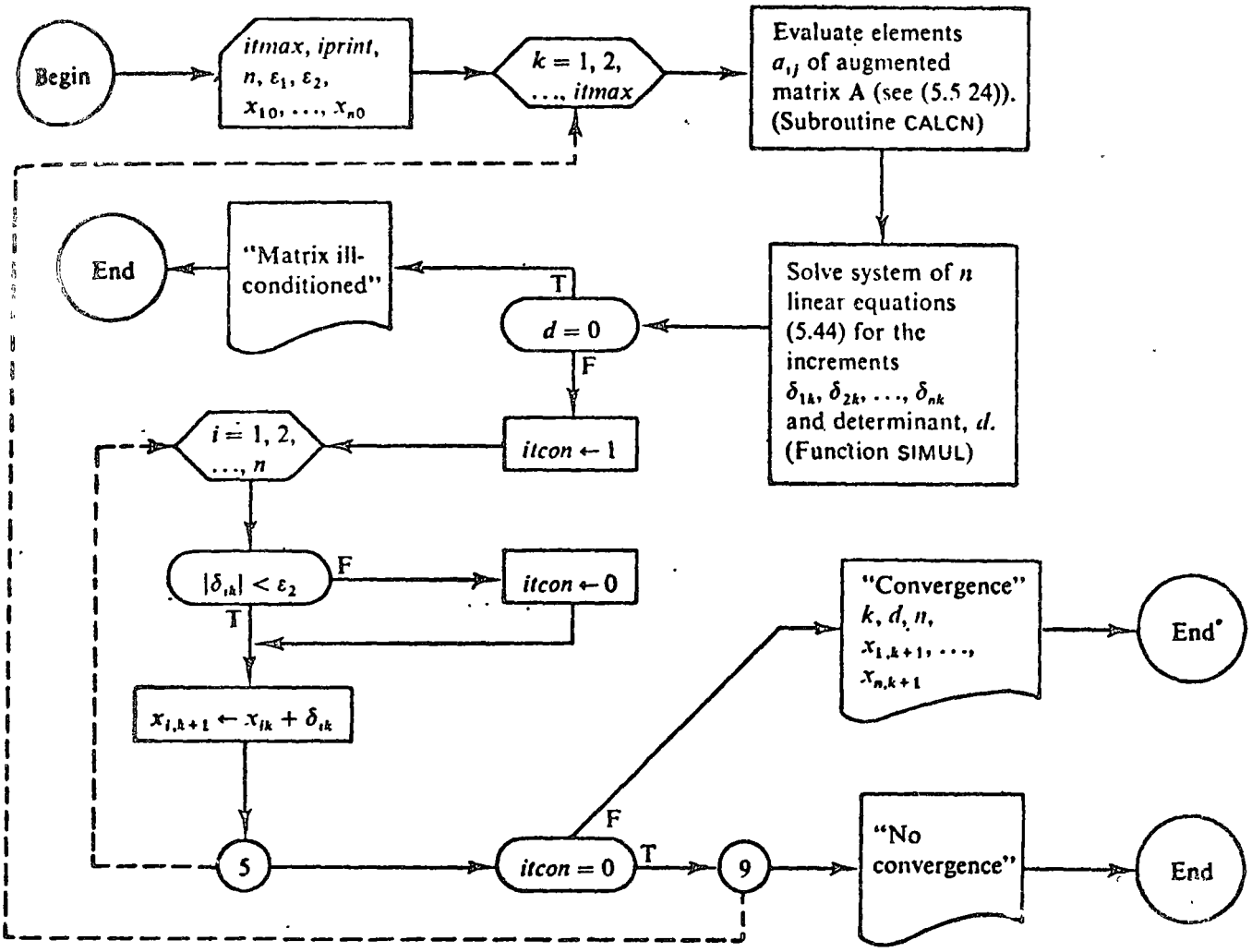
are evaluated by a subroutine named CALCN. The system of linear equations (5.44) is solved by calling on the function SIMUL, described in detail in Example 5.2.

The main program is a general one, in that it is not specifically written to solve only the seven equations of interest. By properly defining the subroutine CALCN, the main program could be used to solve any system of n

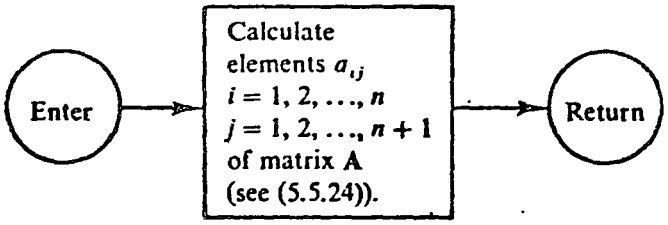
...ultaneous nonlinear equations. The main program ...ds data values for *itmax*, *iprint*, n, ε_1, ε_2, and $x_1, x_2, \ldots, x_n$. Here, *itmax* is the maximum number of Newton-Raphson iterations, *iprint* is a variable that controls printing of intermediate output, n is the number of nonlinear equations, ε_1 is the minimum pivot magnitude allowed in the Gauss-Jordan reduction algorithm, ε_2 is a small positive number used in test (5.5.23), and $x_1, x_2, \ldots, x_n$ are the initial estimates $x_{10}, x_{20}, \ldots, x_{n0}$, that is, the elements of $\mathbf{x}_0$.

Flow Diagram

Main Program



Subroutine CALCN (Arguments: $\mathbf{x}_k$, A, N)



FORTRAN *Implementation*

List of Principal Variables

Program Symbol	*Definition*
(Main)	
A	Augmented matrix of coefficients, A (see (5.5.24)).
DETER	d, determinant of the matrix ϕ (the Jacobian).
EPS1	ε_1, minimum pivot magnitude permitted in subroutine SIMUL.
EPS2	ε_2, small positive number, used in convergence test (5.5.23).
I	Subscript, i.
IPRINT	Print control variable, *iprint*. If *iprint* = 1, intermediate solutions are printed after each iteration.
ITCON	Used in convergence test (5.5.23). ITCON = 1 if (5.5.23) is passed for all i, $i = 1, 2, \ldots, n$; otherwise ITCON = 0.
ITER	Iteration counter, k.
ITMAX	Maximum number of iterations permitted, *itmax*.
N	Number of nonlinear equations, n.
XINC	Vector of increments, δ_{ik}.
XOLD	Vector of approximations to the solution, x_{ik}.
SIMUL	Function developed in Example 5.2. Solves the system of n linear equations (5.44) for the increments, δ_{ik}, $i = 1, 2, \ldots, n$.
(Subroutine CALCN)	
DXOLD	Same as XOLD. Used to avoid an excessive number of references to subroutine arguments in CALCN.
I, J	i and j, row and column subscripts, respectively.
NRC	N, dimension of the matrix A in the calling program. A is assumed to have the same number of rows and columns.
P	Pressure, P, atm.

Program Listing

Main Program

```
C       APPLIED NUMERICAL METHODS, EXAMPLE 5.5
C       CHEMICAL EQUILIBRIUM - NEWTON-RAPHSON METHOD
C
C       THIS PROGRAM SOLVES N SIMULTANEOUS NON-LINEAR EQUATIONS
C       IN N UNKNOWNS BY THE NEWTON-RAPHSON ITERATIVE PROCEDURE.
C       INITIAL GUESSES FOR VALUES OF THE UNKNOWNS ARE READ INTO
C       XOLD(1)...XOLD(N).  THE PROGRAM FIRST CALLS ON THE SUBROUTINE
C       CALCN TO COMPUTE THE ELEMENTS OF A, THE AUGMENTED MATRIX OF
C       PARTIAL DERIVATIVES, THEN ON FUNCTION SIMUL (SEE PROBLEM 5.2)
C       TO SOLVE THE GENERATED SET OF LINEAR EQUATIONS FOR THE CHANGES
C       IN THE SOLUTION VALUES XINC(1)...XINC(N).  DETER IS THE
C       JACOBIAN COMPUTED BY SIMUL.  THE SOLUTIONS ARE UPDATED AND THE
C       PROCESS CONTINUED UNTIL ITER, THE NUMBER OF ITERATIONS,
C       EXCEEDS ITMAX OR UNTIL THE CHANGE IN EACH OF THE N VARIABLES
C       IS SMALLER IN MAGNITUDE THAN EPS2 (ITCON = 1 UNDER THESE
C       CONDITIONS).  EPS1 IS THE MINIMUM PIVOT MAGNITUDE PERMITTED
C       IN SIMUL.  WHEN IPRINT = 1, INTERMEDIATE SOLUTION VALUES ARE
C       PRINTED AFTER EACH ITERATION.
C
      DIMENSION XOLD(21), XINC(21), A(21,21)
C
C     ..... READ AND PRINT DATA .....
    1 READ (5,100) ITMAX,IPRINT,N,EPS1,EPS2,(XOLD(I),I=1,N)
      WRITE (6,200) ITMAX,IPRINT,N,EPS1,EPS2,N,(XOLD(I),I=1,N)
C
C     ..... NEWTON-RAPHSON ITERATION .....
      DO 9   ITER = 1, ITMAX
C
C     ..... CALL ON CALCN TO SET UP THE A MATRIX .....
      CALL CALCN( XOLD, A, 21 )
C
C     ..... CALL SIMUL TO COMPUTE JACOBIAN AND CORRECTIONS IN XINC .....
      DETER = SIMUL( N, A, XINC, EPS1, 1, 21 )
      IF ( DETER.NE.0. )   GO TO 3
         WRITE (6,201)
         GO TO 1
C
C     ..... CHECK FOR CONVERGENCE AND UPDATE XOLD VALUES .....
    3 ITCON = 1
      DO 5   I = 1, N
      IF ( ABS(XINC(I)).GT.EPS2 )   ITCON = 0
    5 XOLD(I) = XOLD(I) + XINC(I)
      IF ( IPRINT.EQ.1 )   WRITE (6,202) ITER,DETER,N,(XOLD(I),I=1,N)
      IF ( ITCON.EQ.0 )   GO TO 9
         WRITE (6,203) ITER,N,(XOLD(I),I=1,N)
         GO TO 1
    9 CONTINUE
C
      WRITE (6,204)
      GO TO 1
C
C     ..... FORMATS FOR INPUT AND OUTPUT STATEMENTS .....
  100 FORMAT ( 10X,I3,17X,I1,19X,I3/ 10X,E7.1,13X,E7.1/ (20X, 5F10.3) )
  200 FORMAT ( 10H1ITMAX  = , I8/  10H IPRINT = , I8/  10H N      = ,
     1   I8/  10H EPS1   = , 1PE14.1/ 10H EPS2   = , 1PE14.1/
     2   26H0          XOLD(1)...XOLD(, I2, 1H)/ 1H  / (1H ,1P4E16.6) )
  201 FORMAT ( 38H0MATRIX IS ILL-CONDITIONED OR SINGULAR )
  202 FORMAT ( 10H0ITER   = ,I8/  10H DETER  = , E18.5/
     2   26H          XOLD(1)...XOLD(, I2, 1H) / (1H ,1P4E16.6) )
  203 FORMAT ( 24H0SUCCESSFUL CONVERGENCE / 10H0ITER   = , I3/
     2   26H0          XOLD(1)...XOLD(, I2, 1H) / 1H / (1H ,1P4E16.6) )
  204 FORMAT ( 15H NO CONVERGENCE )
C
      END
```

Program Listing (*Continued*)

Subroutine CALCN

```
      SUBROUTINE CALCN( DXOLD, A, NRC )
C
C        THIS SUBROUTINE SETS UP THE AUGMENTED MATRIX OF PARTIAL
C        DERIVATIVES REQUIRED FOR THE SOLUTION OF THE NON-LINEAR
C        EQUATIONS WHICH DESCRIBE THE EQUILIBRIUM CONCENTRATIONS
C        OF CHEMICAL CONSTITUENTS RESULTING FROM PARTIAL OXIDATION
C        OF METHANE WITH OXYGEN TO PRODUCE SYNTHESIS GAS.  THE PRESSURE
C        IS 20 ATMOSPHERES.  SEE TEXT FOR MEANINGS OF XOLD(1)...XOLD(N)
C        AND A LISTING OF THE EQUATIONS.  DXOLD HAS BEEN USED AS THE
C        DUMMY ARGUMENT FOR XOLD TO AVOID AN EXCESSIVE NUMBER OF
C        REFERENCES TO ELEMENTS IN THE ARGUMENT LIST.
C
      DIMENSION XOLD(20), DXOLD(NRC), A(NRC,NRC)
C
      DATA  P / 20. /
C
C     ..... SHIFT ELEMENTS OF DXOLD TO XOLD AND CLEAR A ARRAY .....
      DO 1   I = 1, 7
      XOLD(I) = DXOLD(I)
      DO 1   J = 1, 8
    1 A(I,J) = 0.
C
C     ..... COMPUTE NON-ZERO ELEMENTS OF A .....
      A(1,1) = 0.5
      A(1,2) = 1.0
      A(1,3) = 0.5
      A(1,6) = -1.0/XOLD(7)
      A(1,7) = XOLD(6)/XOLD(7)**2
      A(1,8) = -XOLD(1)/2. - XOLD(2) - XOLD(3)/2. + XOLD(6)/XOLD(7)
      A(2,3) = 1.0
      A(2,4) = 1.0
      A(2,5) = 2.0
      A(2,7) = 2.0/XOLD(7)**2
      A(2,8) = -XOLD(3) - XOLD(4) - 2.0*XOLD(5) + 2.0/XOLD(7)
      A(3,1) = 1.0
      A(3,2) = 1.0
      A(3,5) = 1.0
      A(3,7) = 1.0/XOLD(7)**2
      A(3,8) = -XOLD(1) - XOLD(2) - XOLD(5) + 1.0/XOLD(7)
      A(4,1) = -28837.
      A(4,2) = -139009.
      A(4,3) = -78213.
      A(4,4) = 18927.
      A(4,5) = 8427.
      A(4,6) = -10690./XOLD(7)
      A(4,7) = (-13492. + 10690.*XOLD(6))/XOLD(7)**2
      A(4,8) = 28837.*XOLD(1) + 139009.*XOLD(2) + 78213.*XOLD(3)
     1         -18927.*XOLD(4) - 8427.*XOLD(5) - 13492./XOLD(7) + 10690.
     2         *XOLD(6)/XOLD(7)
      A(5,1) = 1.0
      A(5,2) = 1.0
      A(5,3) = 1.0
      A(5,4) = 1.0
      A(5,5) = 1.0
      A(5,8) = 1.0 - XOLD(1) - XOLD(2) - XOLD(3) - XOLD(4) - XOLD(5)
      A(6,1) = P*P*XOLD(4)**3
      A(6,3) = -1.7837E5*XOLD(5)
      A(6,4) = 3.0*P*P*XOLD(1)*XOLD(4)**2
      A(6,5) = -1.7837E5*XOLD(3)
      A(6,8) = 1.7837E5*XOLD(5)*XOLD(3) - P*P*XOLD(1)*XOLD(4)**3
      A(7,1) = XOLD(3)
      A(7,2) = -2.6058*XOLD(4)
      A(7,3) = XOLD(1)
      A(7,4) = -2.6058*XOLD(2)
      A(7,8) = 2.6058*XOLD(4)*XOLD(2) - XOLD(1)*XOLD(3)
      RETURN
C
      END
```

gram Listing (*Continued*)

Data

```
ITMAX  =   50        IPRINT =  1          N      =   7
EPS1   =  1.0E-10    EPS2   =  1.0E-05
XOLD(1)...XOLD(5) =       0.500      0.000      0.000      0.500      0.000
XOLD(6)...XOLD(7) =       0.500      2.000
ITMAX  =   50        IPRINT =  0          N      =   7
EPS1   =  1.0E-10    EPS2   =  1.0E-05
XOLD(1)...XOLD(5) =       0.200      0.200      0.200      0.200      0.200
XOLD(6)...XOLD(7) =       0.500      2.000
ITMAX  =   50        IPRINT =  1          N      =   7
EPS1   =  1.0E-10    EPS2   =  1.0E-05
XOLD(1)...XOLD(5) =       0.220      0.075      0.001      0.580      0.125
XOLD(6)...XOLD(7) =       0.436      2.350
```

mputer Output

Results for the 1st Data Set

```
ITMAX  =        50
IPRINT =         1
N      =         7
EPS1   =         1.0E-10
EPS2   =         1.0E-05

          XOLD(1)...XOLD( 7)

    5.000000E-01     0.0              0.0              5.000000E-01
    0.0              5.000000E-01     2.000000E 00

ITER   =         1
DETER  =        -0.97077E 07
          XOLD(1)...XOLD( 7)
    2.210175E-01     2.592762E-02     6.756210E-02     4.263276E-01
    2.591652E-01     3.343250E-01     1.975559E 00

ITER   =         2
DETER  =        -0.10221E 10
          XOLD(1)...XOLD( 7)
    3.101482E-01     7.142063E-03     5.538273E-02     5.791981E-01
    4.812878E-02     4.681466E-01     2.524948E 00

ITER   =         3
DETER  =        -0.41151E 09
          XOLD(1)...XOLD( 7)
    3.202849E-01     9.554777E-03     4.671279E-02     6.129664E-01
    1.048106E-02     5.533223E-01     2.880228E 00

ITER   =         4
DETER  =        -0.22807E 09
          XOLD(1)...XOLD( 7)
    3.228380E-01     9.224802E-03     4.603060E-02     6.180951E-01
    3.811378E-03     5.758237E-01     2.974139E 00

ITER   =         5
DETER  =        -0.20218E 09
          XOLD(1)...XOLD( 7)
    3.228708E-01     9.223551E-03     4.601713E-02     6.181716E-01
    3.716873E-03     5.767141E-01     2.977859E 00

ITER   =         6
DETER  =        -0.20134E 09
          XOLD(1)...XOLD( 7)
    3.228708E-01     9.223547E-03     4.601710E-02     6.181716E-01
    3.716847E-03     5.767153E-01     2.977863E 00
```

Computer Output (*Continued*)

```
 SUCCESSFUL CONVERGENCE

 ITER   =  6

          XOLD(1)...XOLD( 7)

    3.228708E-01   9.223547E-03   4.601710E-02   6.181716E-01
    3.716847E-03   5.767153E-01   2.977863E 00
```

Results for the 3rd Data Set

```
 ITMAX  =       50
 IPRINT =        1
 N      =        7
 EPS1   =      1.0E-10
 EPS2   =      1.0E-05

          XOLD(1)...XOLD( 7)

    2.200000E-01   7.499999E-02   9.999999E-04   5.800000E-01
    1.250000E-01   4.360000E-01   2.349999E 00

 ITER   =        1
 DETER  =      -0.61808E 08
          XOLD(1)...XOLD( 7)
    6.951495E-01  -8.022028E-02   1.272939E-02   1.217132E 00
   -8.447912E-01   1.314754E 00   5.969404E 00

 ITER   =        2
 DETER  =        0.12576E 09
          XOLD(1)...XOLD( 7)
    4.958702E-01  -1.698154E-02   5.952045E-03   9.518250E-01
   -4.366657E-01   2.379797E 00   1.043425E 01

 ITER   =        3
 DETER  =        0.77199E 07
          XOLD(1)...XOLD( 7)
    4.559822E-01  -9.799302E-04  -7.583648E-04   9.107630E-01
   -3.650070E-01   2.509821E 00   1.107038E 01

 ITER   =        4
 DETER  =        0.53378E 07
          XOLD(1)...XOLD( 7)
    4.569673E-01  -4.071472E-04  -2.142648E-03   9.152630E-01
   -3.696806E-01   2.608933E 00   1.149338E 01

 ITER   =        5
 DETER  =        0.49739E 07
          XOLD(1)...XOLD( 7)
    4.569306E-01  -4.071994E-04  -2.125205E-03   9.151721E-01
   -3.695704E-01   2.610552E 00   1.150046E 01

 ITER   =        6
 DETER  =        0.49611E 07
          XOLD(1)...XOLD( 7)
    4.569306E-01  -4.071984E-04  -2.125199E-03   9.151720E-01
   -3.695703E-01   2.610549E 00   1.150045E 01

 SUCCESSFUL CONVERGENCE

 ITER   =  6

          XOLD(1)...XOLD( 7)

    4.569306E-01  -4.071984E-04  -2.125199E-03   9.151720E-01
   -3.695703E-01   2.610549E 00   1.150045E 01
```

Discussion of Results

Results are shown for the first and third data sets only. For the first two data sets, the Newton-Raphson iteration converged to the same solution, one that satisfies the equilibrium conditions (5.5.16). Results for the third data set cannot be physically meaningful, because the solution has negative mole fractions for CO_2, H_2O, and CH_4. The equilibrium compositions, reactant ratio O_2/CH_4 in the feed gases, and the total number of moles of product per mole of CH_4 in the feed are tabulated in Table 5.5.2. Thus the required feed ratio is 0.5767 moles of oxygen per mole of methane in the feed gases.

Table 5.5.2 Equilibrium Gas Mixture

x_1, mole fraction CO	0.322871
x_2, mole fraction CO_2	0.009224
x_3, mole fraction H_2O	0.046017
x_4, mole fraction H_2	0.618172
x_5, mole fraction CH_4	0.003717
x_6, feed ratio O_2/CH_4	0.576715
x_7, total moles of product	2.977863

To establish if carbon is likely to be formed according to reaction (5.5.7) at 2200°F for a gas of the computed composition, it is necessary to calculate the magnitude of

$$\bar{K} = \frac{p_{CO}^2}{a_C p_{CO_2}} = \frac{P x_1^2}{a_C x_2}. \tag{5.5.25}$$

If $\bar{K}$ is larger than K_4 from (5.5.8), then there will be a tendency for reaction (5.5.7) to shift toward the left; carbon will be formed. Assuming that $a_C = 1$,

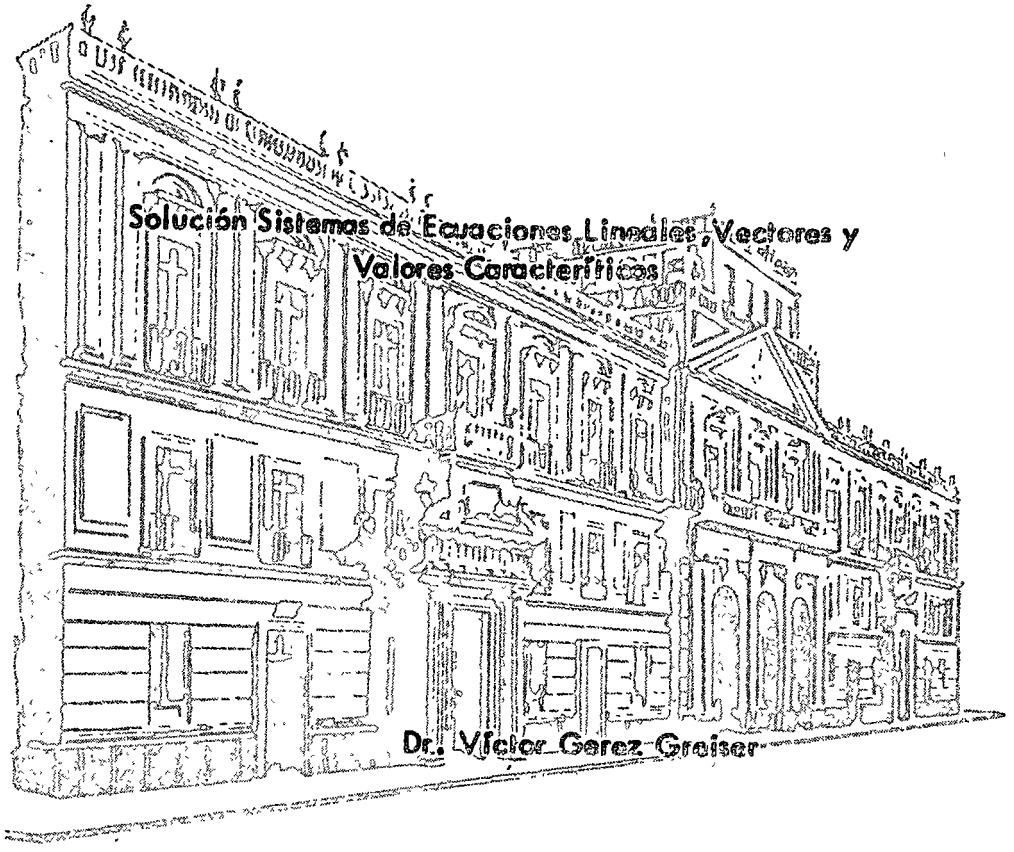
$$\bar{K} = \frac{20 \times (0.322871)^2}{1 \times 0.009224} \doteq 226.03 < K_4 = 1329.5. \tag{5.5.26}$$

Therefore there will be no tendency for carbon to form.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA

DIGITAL



Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels: 521-40-23 521-73-35 5123-123

b. Find the highest common factor of these polynomials.
c. Find the roots.

14. By direct substitution show that $y = ce^{\alpha t}$ is a solution of the differential equation

$$\frac{d^4y}{dt^4} + 3\frac{d^3y}{dt^3} - 2\frac{d^2y}{dt^2} + 3\frac{dy}{dt} + y = 0$$

if α is a root of the polynomial equation

$$\alpha^4 + 3\alpha^3 - 2\alpha^2 + 3\alpha + 1 = 0$$

If α_1, α_2, α_3, and α_4 are the four roots of this equation, show that

$$y = c_1e^{\alpha_1 t} + c_2e^{\alpha_2 t} + c_3e^{\alpha_3 t} + c_4e^{\alpha_4 t}$$

also satisfies the differential equation for any values of c_1, c_2, c_3, and c_4.

15. Show that for t sufficiently large and α_1, α_2, α_3, and α_4 all real, the value of y will be determined by the largest positive α_i.

16. Show that if $y = ce^{\alpha t}$, then y is less than or equal to c in absolute value for all $t > 0$, if the real part of α is zero or negative.

17. In a mechanical system of springs and masses, the motion of any part after a sudden impulse acceleration is governed by a differential equation of the form

$$a_1\frac{d^ny}{dt^n} + a_2\frac{d^{n-1}y}{dt^{n-1}} + \cdots + a_n\frac{dy}{dt} + a_{n+1}y = 0$$

The system will be stable, that is, will not tend to shake itself apart, if none of the solutions $y = ce^{\alpha t}$ grow very large as t increases. Show that the system will be stable if all the roots of the polynomial equation

$$a_1\alpha^n + a_2\alpha^{n-1} + \cdots + a_n\alpha + a_{n+1} = 0$$

have zero or negative real parts.

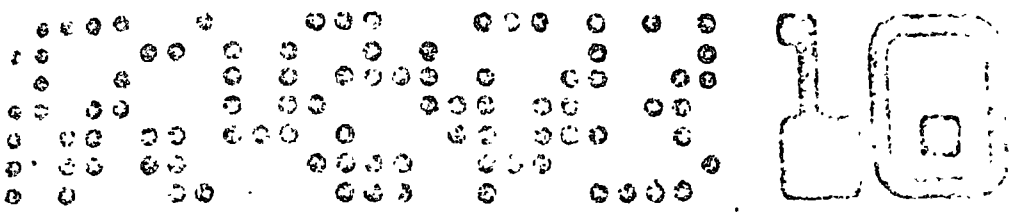
18. In an electrical circuit of resistors, capacitors, and inductances, the current at any point after a sudden initial impulse current is governed by a differential equation of the form

$$a_1\frac{d^ni}{dt^n} + a_2\frac{d^{n-1}i}{dt^{n-1}} + \cdots + a_n\frac{di}{dt} + a_{n+1} = 0$$

The system will be stable, that is, will not tend to develop very large local currents and burn out components if none of the solutions of the form $i = ce^{\alpha t}$ grow very large as t increases. Show that the system will be stable if all the roots of the polynomial equation

$$a_1\alpha^n + a_2\alpha^{n-1} + \cdots + a_n\alpha + a_{n+1} = 0$$

have zero or negative real parts.



Simultaneous Linear Equations and Matrices

10.1 INTRODUCTION

In this chapter we turn to a problem of finding the values of unknowns, x_1, x_2, etc., which satisfy systems of equations of type

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n &= b_2 \\
a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n &= b_3 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n &= b_n
\end{aligned} \qquad (10\text{-}1)$$

When the number of equations is equal to the number of unknowns, there will ordinarily be a unique solution, that is, one set of values of x_1, x_2, ..., x_n which satisfy all of the equations. At least such is the concept in the world of exact numbers and exact arithmetic. When the coefficients are approximate numbers, the concept of a solution becomes less clear, as the following example demonstrates.

Example 1. Find the solution of

$$\begin{aligned}
1.0x - 2.0y &= 1.0 \\
.5x - 1.0y &= 1.0
\end{aligned}$$

Figure 10-1 represents the solution, taking into account the approximate nature of the coefficients. Each equation is represented not by a line but by a band. Within our knowledge of the accuracy of the above numbers, any value in the band is as acceptable as any other. For example, in the first equation, when $x = 0$, y can be as small as $-1.05/1.95 \approx -.54$ or as large as $-.95/2.05 \approx -.46$. Thus at $x = 0$, the band for the first equation covers the region from $y = -.54$ to $y = -.46$. The two bands intersect not in a unique point but in a region, and any point in this region might be accepted as a solution. The nominal solution, for the above system of equations, obtained by accepting the coefficients as exact, is $x = 2/3$, $y = -1/6$, or approximately $x = .67$, $y = -.17$. However, the points $x = .86$, $y = -.12$

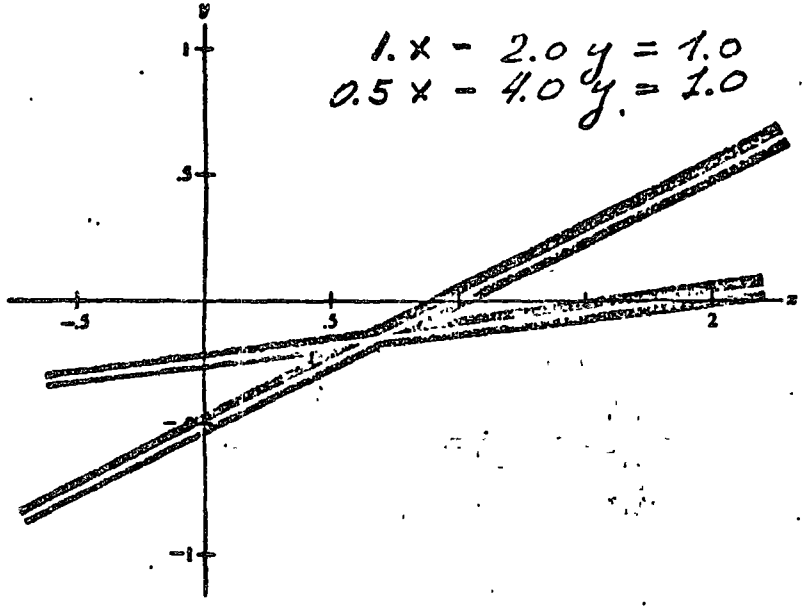


Figure 10-1

and $x = .5$, $y = -.21$ are also within the acceptable region. It is somewhat disconcerting to note that in this rather straightforward case, with the coefficients known to 10% or better, the solution is uncertain by 30% or more. It can be seen that if the equations represent lines that are nearly parallel, the region of overlap of the two bands representing the equations can be quite extended, as illustrated in Figure 10-2(a). In this case, even if the coefficients were exact, a small change in one of them can make a sizable difference in the solution, as illustrated in Figure 10-2(b) and (c). Equations having this property are termed ill-conditioned. An accurate solution can be found only by performing the computation with great care, since even small

rounding errors may influence the answer greatly. Further, in practical problems, the answer itself must be viewed with some circumspection, since any inherent inaccuracies in the values of the coefficients may cause large changes in the answers.

The above example concerned itself with two equations and two unknowns, but analogous situations exist for higher numbers of equations and unknowns.

In this chapter, three general methods of solving a set of simultaneous linear equations are discussed: direct methods, in which the solution is found by a finite number of algebraic manipulations of the coefficients; iterative methods, which produce a set of successive approximations to the solution which hopefully become very close to the solution but never actually reach it; and matrix

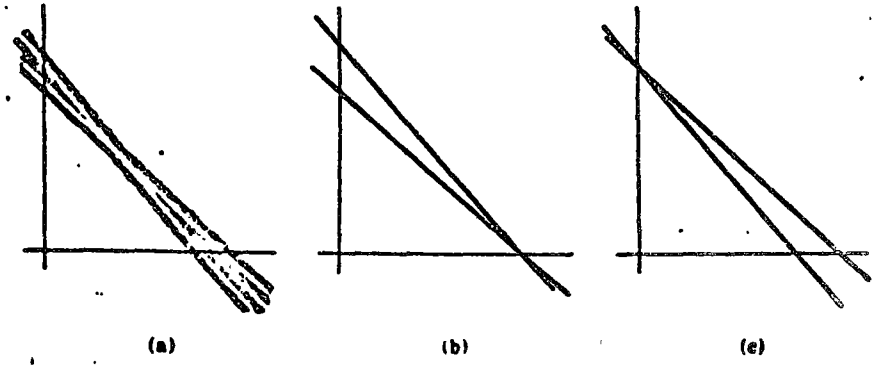


Figure 10-2

inversion methods, which are quite similar to the direct methods in numerical content but which provide conceptually more elegant bases for such methods. As was indicated in Chapter 5, no one of these methods is always best. The direct methods and matrix methods can have accuracy problems for some values of the coefficients and constant terms. The iterative methods can fail to converge to a solution. An attempt will be made to indicate the conditions under which the various methods can be expected to give satisfactory results.

10.2 THE ELIMINATION METHOD

The elimination method consists of multiplying various of the equations by appropriate constants and adding to other equations so as to obtain zero coefficients in some locations and eventually obtain equations that can be solved directly. The particular form of the elimination we shall use is that known as the Gauss-Jordan method. In this method, an appropriate multiple of the first equation is added to each of the other equations so that the resulting $n - 1$ equations have zero coefficients for the x_1 term. (If the first equation

does not have a term involving x_1, we must first interchange two equations to obtain one with an x_1 term as the first equation.) Then an appropriate multiple of the next equation is added to all equations to eliminate the x_2 term from all but one equation. The process is continued until each equation contains only one unknown, and the equations are solved. At each step, the coefficient being used to eliminate other coefficients is called the pivotal coefficient.

To demonstrate how a machine program can be organized to perform this process, we shall construct some diagrams. Equation (10-1) will be represented internally in a computer only by the stored value of the coefficients a_{11} through a_{nn} and b_1 through b_n, perhaps as subscripted variables A(I,J) and B(J). Since the plus signs, x's, and equals signs will not be stored in the computer anyway, let us omit them and write down only the constants and coefficients, arranged as in the equations but omitting the x's and algebraic symbols, thus:

$$\begin{array}{ccccccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} & b_2 \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} & b_3 \\ \vdots & & & & & \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{n4} & b_n \end{array} \tag{10-2}$$

remembering that we will mentally supply the x's and symbols where needed.

To make the notation appear more uniform, let us rename $b_1, b_2, \ldots, b_n$ as $a_{1n+1}, a_{2n+1}, \ldots, a_{nn+1}$. Then the array can be written

$$\begin{array}{ccccccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & a_{1n+1} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} & a_{2n+1} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} & a_{3n+1} \\ \vdots & & & & & \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} & a_{nn+1} \end{array} \tag{10-3}$$

As a first step in the elimination process we can divide the first equation by a_{11} to make the coefficient of x_1 become 1, and obtain the equations represented by

$$\begin{array}{ccccccc} 1 & \dfrac{a_{12}}{a_{11}} & \dfrac{a_{13}}{a_{11}} & \cdots & \dfrac{a_{1n}}{a_{11}} & \dfrac{a_{1n+1}}{a_{11}} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} & a_{2n+1} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} & a_{3n+1} \\ \vdots & & & & & \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} & a_{nn+1} \end{array}$$

Now we can eliminate the x_1 term from each of the other equations by multiplying the first equation by a_{21} and subtracting from the second, by a_{31} and subtracting from the third, etc., giving

$$\begin{array}{ccccccc} 1 & \dfrac{a_{12}}{a_{11}} & \dfrac{a_{13}}{a_{11}} & \cdots & \dfrac{a_{1n}}{a_{11}} & \dfrac{a_{1n+1}}{a_{11}} \\ 0 & a_{22}-a_{21}\left(\dfrac{a_{12}}{a_{11}}\right) & a_{23}-a_{21}\left(\dfrac{a_{13}}{a_{11}}\right) & \cdots & a_{2n}-a_{21}\left(\dfrac{a_{1n}}{a_{11}}\right) & a_{2n+1}-a_{21}\left(\dfrac{a_{1n+1}}{a_{11}}\right) \\ 0 & a_{32}-a_{31}\left(\dfrac{a_{12}}{a_{11}}\right) & a_{33}-a_{31}\left(\dfrac{a_{13}}{a_{11}}\right) & \cdots & a_{3n}-a_{31}\left(\dfrac{a_{1n}}{a_{11}}\right) & a_{3n+1}-a_{31}\left(\dfrac{a_{1n+1}}{a_{11}}\right) \\ \vdots & & & & & \\ 0 & a_{n2}-a_{n1}\left(\dfrac{a_{12}}{a_{11}}\right) & a_{n3}-a_{n1}\left(\dfrac{a_{13}}{a_{11}}\right) & \cdots & a_{nn}-a_{n1}\left(\dfrac{a_{1n}}{a_{11}}\right) & a_{nn+1}-a_{n1}\left(\dfrac{a_{1n+1}}{a_{11}}\right) \end{array}$$

At this point, we have eliminated the x_1 term from all but the first equation, using a_{11} as the pivotal coefficient. Note that in the computer, the new coefficients may as well be stored in the locations which held the old ones; that is, a_{12}/a_{11} simply replaces a_{12}, etc. If this is done, the above array becomes

$$\begin{array}{ccccccc} 1 & a_{12} & a_{13} & \cdots & a_{1n} & a_{1n+1} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} & a_{2n+1} \\ 0 & a_{32} & a_{33} & \cdots & a_{3n} & a_{3n+1} \\ \vdots & & & & & \\ 0 & a_{n2} & a_{n3} & \cdots & a_{nn} & a_{nn+1} \end{array}$$

and the process which gives this array from the original one can be described by

$$\begin{aligned} a_{1j}/a_{11} &\to a_{1j} && \text{for } j = 2, \ldots, n+1 \\ a_{ij} - a_{i1}a_{1j} &\to a_{ij} && \text{for } i = 2, \ldots, n \\ & && \quad j = 2, \ldots, n+1 \end{aligned}$$

Note that these steps will not actually put $a_{11} = 1$ and $a_{i1} = 0$ for $i > 1$, that is, will not set the first column to one and zeros. Since we know they should be there, we can simply remember the fact, and not force the computer to take the extra steps to actually put them there.

Now we need to eliminate x_2 from equations 3 through n and from equation 1 by an analogous process. The steps are described by

$$\begin{aligned} a_{2j}/a_{22} &\to a_{2j} && \text{for } j = 3, \ldots, n+1 \\ a_{ij} - a_{i2}a_{2j} &\to a_{ij} && \text{for } i = 3, \ldots, n, \text{ and } i = 1 \\ & && \quad j = 3, \ldots, n+1 \end{aligned}$$

and produce an array of the form

$$\begin{matrix} 1 & 0 & a_{13} & \cdots & a_{1n} & a_{1n+1} \\ 0 & 1 & a_{23} & \cdots & a_{2n} & a_{2n+1} \\ 0 & 0 & a_{33} & \cdots & a_{3n} & a_{3n+1} \\ \vdots & & & & & \\ 0 & 0 & a_{n3} & \cdots & a_{nn} & a_{nn+1} \end{matrix}$$

If the process is continued, we eventually obtain the array

$$\begin{matrix} 1 & 0 & 0 & \cdots & 0 & a_{1n+1} \\ 0 & 1 & 0 & \cdots & 0 & a_{2n+1} \\ 0 & 0 & 1 & \cdots & 0 & a_{3n+1} \\ \vdots & & & & & \\ 0 & 0 & 0 & \cdots & 1 & a_{nn+1} \end{matrix} \qquad (10\text{-}4)$$

and so $x_1 = a_{1n+1}$, $x_2 = a_{2n+1}$, etc. The process can be summarized in flow chart form as in Figure 10-3. A remote-terminal routine which would perform the process for systems up to 10 by 10 can be written as follows:

```
 1    DIMENSION A(10,11)
 2  1 PRINT, "NUMBER OF EQUATIONS"
 3    INPUT, N
 4    NN=N+1
 5    PRINT, "A(1,1),A(1,2),,,A(1,N),B(1),A(2,1),ETC"
 6    INPUT, ((A(I,J),J=1,NN),I=1,N)
 7    DO 3 K=1,N
 8    KK=K+1
 9    DO 3 J=KK,NN
10    A(K,J)=A(K,J)/A(K,K)
11    DO 3 I=1,N
12    IF(K-I)2,3,2
13  2 A(I,J)=A(I,J)-A(I,K)*A(K,J)
14  3 CONTINUE
15    PRINT, "SOLUTION",(A(I,NN),I=1,N)
16    GO TO 1
17    END
```

Example 1. Show all inputs and machine responses for running the above program to solve the set of equations

$$2x_1 + 3x_2 + 5x_3 = 5$$
$$3x_1 + 4x_2 + 7x_3 = 6$$
$$x_1 + 3x_2 + 2x_3 = 5$$

The inputs and responses would appear as follows:

```
RUN
*N
? 3
  A(1,1),A(1,2),,,A(1,N),B(1),A(2,1),ETC
? 2,3,5,5,3,4,7,6,1,3,2,5
  SOLUTION  -3.000000        2.000000        1.000000
```

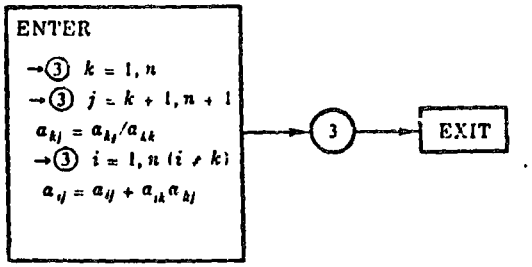


Figure 10-3: Flow chart for Gauss-Jordan method

The program given above will run into trouble if any of the coefficients A(K,K) are zero, since it will attempt to divide by zero. One way to avoid this problem is to rearrange the equations any time a zero element on the diagonal is encountered.

Another way, not much more difficult to execute, is to rearrange the equations at each step so that the pivotal coefficient at each step is not only nonzero but is actually the largest coefficient. This approach not only avoids division by zero but also tends to enhance accuracy by minimizing round off error. It has the disadvantage that the rearrangement will cause the unknowns to be scrambled at the end of the process. Suppose, for example, that initially the largest coefficient is a_{32}. Then we would like to arrange the equations as

$$a_{32}x_2 + a_{31}x_1 + a_{33}x_3 + \cdots + a_{3n}x_n = b_3$$
$$a_{22}x_2 + a_{21}x_1 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2$$
$$a_{12}x_2 + a_{11}x_1 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1$$
$$a_{42}x_2 + a_{41}x_1 + a_{43}x_3 + \cdots + a_{4n}x_n = b_4$$
$$\vdots$$
$$a_{n2}x_2 + a_{n1}x_1 + a_{n3}x_3 + \cdots + a_{nn}x_n = b_n$$

In terms of the original set of equations, (10-1), we have interchanged the first and third equations and have also interchanged the positions of x_1 and x_2 in all equations. In terms of the array of coefficients (10-3) we have inter-

changed the first and third rows and the first and second columns. If we continued the process to the end with no further rearrangement, the final value in $a_{1,n+1}$ when we reach the stage represented by (10-4) is not x_1 but x_2. Thus when we interchange rows or columns to obtain a large pivotal coefficient, we must also keep track of which unknown is represented by a particular column. This can be done by storing an identification number, ID, for each column which indicates the number of the unknown represented by that column. For example, in the rearrangement shown above, the information that the variable x_2 was now in the first column would be indicated by setting ID(1) = 2.

A separate subroutine can be written to handle the exchange of rows and columns to make the largest element appear at location A(K,K). The subroutine given below would suffice for this purpose.

```
      SUBROUTINE EXCH(A,N,NN,K,ID)
      DIMENSION A(20,21),ID(20)
      NROW=K
      NCOL=K
      B=ABS(A(K,K))
      DO 2 I=1,N
      DO 2 J=1,NN
      IF(ABS(A(I,J)-B))2,2,21
   21 NROW=I
      NCOL=J
      B=ABS(A(I,J))
    2 CONTINUE
      IF(NROW-K)3,3,31
   31 DO 32 J=K,NN
      C=A(NROW,J)
      A(NROW,J)=A(K,J)
   32 A(K,J)=C
    3 CONTINUE
      IF(NCOL-K)4,4,41
   41 DO 42 I=1,N
      C=A(I,NCOL)
      A(I,NCOL)=A(I,K)
   42 A(I,K)=C
      I=ID(NCOL)
      ID(NCOL)=ID(K)
      ID(K)=I
    4 CONTINUE
      RETURN
      END
```


In this subroutine, the statements up to number 2 locate the element having the largest absolute value and identify its location as NROW,NCOL. The statements from 2 to 3 interchange rows K and NROW if they are not the same row. The statements from 3 to 4 interchange columns K and NCOL if they are not the same column, and also interchange the ID numbers to record this fact. Using this subroutine, one to solve the set of linear equations can be written as follows:

```
      SUBROUTINE ELIM(AA,N,BB,X)
      DIMENSION AA(20,20),BB(20),A(20,21),X(20),ID(20)
      NN=N+1
      DO 100 I=1,N
      A(I,NN)=BB(I)
      ID(I)=I
      DO 100 J=1,N
  100 A(I,J)=AA(I,J)
      K=1
    1 CALL EXCH(A,N,NN,K,ID)
    2 IF(A(K,K))3,999,3
    3 KK=K+1
      DO 4 J=KK,NN
      A(K,J)=A(K,J)/A(K,K)
      DO 4 I=1,N
      IF(K-I)41,4,41
   41 A(I,J)=A(I,J)-A(I,K)*A(K,J)
    4 CONTINUE
      K=KK
      IF(K-N)1,2,5
    5 DO 10 I=1,N
      DO 10 J=1,N
      IF(ID(J)-I)10,6,10
    6 X(I)=A(J,NN)
   10 CONTINUE
      RETURN
  999 PRINT 1000
      RETURN
 1000 FORMAT(19H NO UNIQUE SOLUTION)
      END
```

In this subroutine, the input coefficients are identified as AA(I,J) and the input constants as BB(I). The statements up to 100 reidentify these quantities as A(I,J), so the original values will not be destroyed by the subroutine. Statement 1 calls subroutine EXCH to make the largest coefficient the pivotal

coefficient. If the largest coefficient is zero, the message "NO UNIQUE SOLUTION" is printed and an exit is taken. Otherwise, statements 3 through 4 solve the equations as in the remote-terminal program given earlier. Statements 5 through 10 use the identification numbers to unscramble the unknowns and return them in proper order.

10.3 GAUSS-SEIDEL METHOD

Another and quite different method of solving a system of linear equations is the so-called Gauss-Seidel method, in which equations (10-1) are rewritten in the following form:

$$\begin{aligned}
a_{11}x_1 &= b_1 - a_{12}x_2 - a_{13}x_3 - \cdots - a_{1n}x_n \\
a_{22}x_2 &= b_2 - a_{21}x_1 - a_{23}x_3 - \cdots - a_{2n}x_n \\
a_{33}x_3 &= b_3 - a_{31}x_1 - a_{32}x_2 - a_{34}x_4 - \cdots - a_{3n}x_n \\
&\vdots \\
a_{nn}x_n &= b_n - a_{n1}x_1 - a_{n2}x_2 - \cdots - a_{nn-1}x_{n-1}
\end{aligned} \tag{10-5}$$

In words, in each of the equations all but one unknown is taken to the right-hand side of the equation. We then guess a set of values for $x_2, x_3, \ldots, x_n$ and substitute these in the right-hand side of the first equation and solve for x_1. Then we substitute this value and the original values of $x_3, \ldots, x_n$ in the right-hand side of the second equation and solve for x_2. We discard the old value of x_2 and keep this as a better one. We then substitute in the right-hand side of the third equation and obtain a new value for x_3. After we have proceeded through all the equations in this fashion, we have a new set of values $x_1, x_2, \ldots, x_n$. (We must first arrange the equations so that none of the $a_{ii} = 0$.) We then start again with the first equation and find a new x_1, then a new x_2, etc. Each time through this process gives us a new, and, we hope, better set of values for $x_1, x_2, \ldots, x_n$. When the new values obtained agree with the previous set to within the accuracy we desire, we have the solution. This is an iteration process similar in nature to those discussed in Chapter 8. It is not absolutely certain that this process will converge, that is, that the differences between succeeding sets of values will get smaller and smaller. We shall discuss the convergence problem more fully a little later. It is not certain, either, how many multiplications will be required to obtain the solution to a desired accuracy. Each trip through the set of equations, or iteration, requires n^2 multiplications. If $(1/3)n$ iterations happen to be required, then the method will take about as long as the elimination method. It may take more or less time, depending entirely on the speed of convergence and accuracy required.

Example 1. Solve the system

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

by the Gauss-Seidel method.

We write the equations as

$$x_1 = 1 + 2x_2 \tag{10-6}$$

$$x_2 = 1 - x_1/4 \tag{10-7}$$

Let us take as starting values $x_1 = x_2 = 0$.

Putting $x_2 = 0$ in equation (10-6), we obtain

$$x_1 = 1$$

Putting $x_1 = 1$ in equation (10-7), we obtain

$$x_2 = 3/4$$

At the end of the first iteration, then, we have

$$x_1 = 1, \qquad x_2 = 3/4$$

Putting $x_2 = 3/4$ in equation (10-6), we have

$$x_1 = 5/2$$

Putting $x_1 = 5/2$ in equation (10-7), we have

$$x_2 = 3/8$$

At the end of the second iteration, then, we have

$$x_1 = 5/2, \qquad x_2 = 3/8$$

We can continue this process. The results for the first several steps, starting from the beginning, are

x_1	x_2
0	0
1	.75
2.5	.375
1.75	.5625
2.125	.46875
1.9375	.515625
2.03125	.4921875
1.984375	.51390625

It is easily verified from the equation that the correct solution is $x_1 = 2$, $x_2 = 1/2$. This solution is slowly converging toward those values.

Example 2. Solve the system

$$x_1 + 4x_2 = 4$$
$$x_1 - 2x_2 = 1$$

by the Gauss-Seidel method.

This is the same problem as Example 1, with the equations reversed. We write the equations as

$$x_1 = 4 - 4x_2$$
$$x_2 = -1/2 + x_1/2$$

Then the successive interations give the following values:

x_1	x_2
0	0
4	1.5
−2	−1
8	3.5
−10	−5.5
26	12.5
−46	−23.5

It is clear that the process is diverging, and the solution will not be obtained.

Example 3. Apply the Gauss-Seidel method to Example 1, Section 10.2.

The equations are

$$2x_1 + 3x_2 + 5x_3 = 5$$
$$3x_1 + 4x_2 + 7x_3 = 6$$
$$x_1 + 3x_2 + 2x_3 = 5$$

We write them as

$$2x_1 = 5 - 3x_2 - 5x_3$$
$$4x_2 = 6 - 3x_1 - 7x_3$$
$$2x_3 = 5 - x_1 - 3x_2$$

Successive iterations give (to four decimal places)

	x_1	x_2	x_3
	0	0	0
	2.5	−.375	1.8125
	−1.4688	.5703	2.3789
	−4.3027	.5640	3.8054
	−7.8595	.7352	5.3270
	−11.9203	1.1180	6.7831

In Section 10.2 we found that the solution to this system was

$$x_1 = -3, \quad x_2 = 2, \quad x_3 = 1$$

Our iteration scheme is not converging toward those values.

10.31 Convergence of the Gauss-Seidel Method

Some insight into the convergence problem can be obtained by following Examples 1 and 2, Section 10.3, in graphical form. Figure 10-4 illustrates the scheme followed in Example 1. Starting at the point P_0, we change x_1 (that is, move horizontally) to arrive on the line $x_1 - 2x_2 = 1$, and then change x_2 (that is, move vertically) to arrive on the line $x_1 + 4x_2 = 4$, bringing us to the point P_1. This is the point given by the first iteration. On the second iteration we move horizontally, then vertically to arrive at P_2. On the third we move horizontally, then vertically to arrive at P_3, etc. It is clear from the figure that this process is bringing us closer and closer to the true point of intersection.

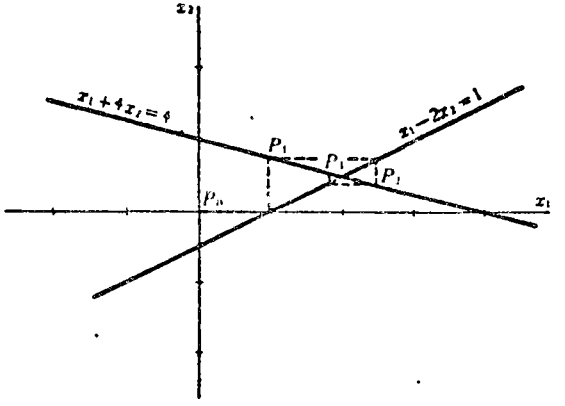


Figure 10-4

Figure 10-5 illustrates the scheme followed in Example 2. The same two lines are involved, but this time we always move horizontally to reach the line $x_1 + 4x_2 = 4$ and vertically to reach the line $x_1 - 2x_2 = 1$. The points $P_0, P_1, P_2, \ldots$ are the results of the successive iterations in this case.

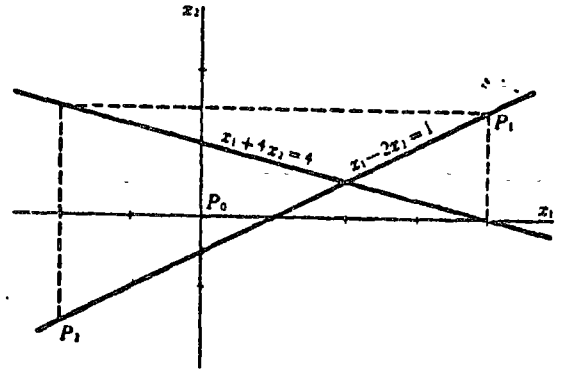


Figure 10-5

It appears that graphically the Gauss-Seidel method for two equations in two unknown consists of following the above boxlike pattern about the point of intersection of the two lines: If this pattern is followed in the correct direction the intersection will be approached, but if it is followed in the wrong direction the process will diverge from the intersection. This is the case if the slopes of the lines have opposite signs. If the signs of the slopes are the same, the situation is a little different, as depicted in Figure 10-6. The sequence of points P_0, P_1, P_2 is part of a convergent process, in which we proceed horizontally to line (b), then vertically to line (a). The points P_0,

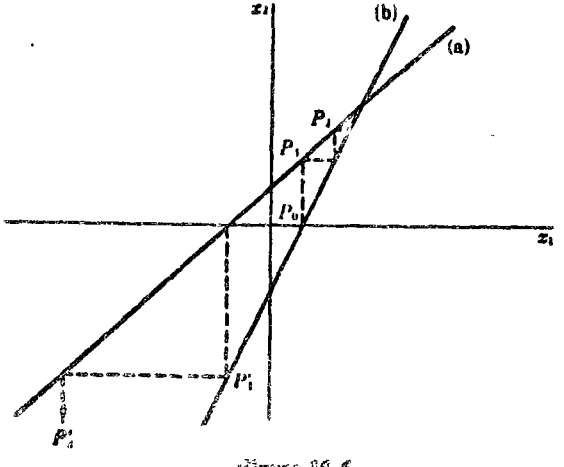


Figure 10-6

P_1', P_2' are part of a divergent process, in which we proceed horizontally to line (a) then vertically to line (b).

As indicated by the above figures, the situation regarding convergence for the Gauss-Seidel method for two equations in two unknowns is as follows: The process will converge for the equations arranged in one order and diverge for the equations arranged in the opposite order. The only exception occurs when the equations represent perpendicular lines, in which case the process will not converge for either arrangement. It is interesting to note that, contrary to our experience with iteration methods in the preceding chapters, the convergence or nonconvergence for these linear equations does *not* depend on choice of initial estimate.

For larger systems of equations the situation becomes much more complex. The necessary and sufficient conditions for convergence are known but are not easily expressed in a very usable form. Sometimes a rearrangement of the equations will produce convergence, but this is not at all guaranteed. The likelihood of convergence is usually increased if the equations are rearranged so that the coefficients $a_{11}, a_{22}, a_{33}, \ldots, a_{nn}$ which appear on the left-hand side in the system as written in Section 10.3 are the largest coefficients in absolute value. In fact, convergence is assured in this case if in each equation the absolute value of the coefficient a_{ii} is larger than the sum of the absolute values of the remaining coefficients. This condition is not often met. In fact, as in Example 3, Section 10.3, it is often impossible even to write all the equations with largest terms on the left-hand side.

10.32 Flow Chart and Program for the Gauss-Seidel Method

The flow chart in Figure 10-7 describes the Gauss-Seidel method. This flow chart uses the equations arranged just as they are, with no attempt to rearrange the equations to increase the likelihood of convergence. If desired, it could be preceded by another section of flow chart which would rearrange the equations in attempt to enhance the likelihood of convergence. In order to cut down on the number of divisions required, each of the equations is first divided through by the coefficient a_{ii}, so that in the set of new coefficients, c_{ij}, the c_{ii}'s are all one. This flow chart computes at each iteration a quantity

$$E = \sum_{i=1}^{n} |x_i^{\text{new}} - x_i^{\text{old}}|$$

and when this quantity becomes smaller than the given number d, the iteration stops. Note that, in the way the expression for P is written, P is precisely $x_i^{\text{new}} - x_i^{\text{old}}$.

The FORTRAN subroutine below uses the Gauss-Seidel method to solve an N by N system of linear equations, following the flow chart. Again, if a rearrangement of the equations were desired, it could be accomplished by using a subroutine for that purpose just prior to using the one given below.

```
      SUBROUTINE GAUSID(A,N,B,X,ERR)
      DIMENSION A(20,20), B(20),C(20,21),X(20)
      K=0
      NN=N+1
      DO 11 I=1,N
      IF(A(I,I))12,6,12
   12 X(I)=1.
      C(I,NN)=B(I)/A(I,I)
      DO 11 J=1,N
   11 C(I,J)=A(I,J)/A(I,I)
    1 CONTINUE
      E=0.
      DO 3 I=1,N
      P=C(I,NN)
      DO 2 J=1,N
      P=P-C(I,J)*X(J)
    2 CONTINUE
      X(I)=X(I)+P
      E=E+ABSF(P)
    3 CONTINUE
      IF(E-ERR)4,4,5
    4 RETURN
    5 K=K+1
      IF(100-K)6,1,1
    6 PRINT 1000
      RETURN
 1000 FORMAT(25H GAUSID DOES NOT CONVERGE)
      END
```

EXERCISE 26

1. Following the remote-terminal program of Section 10.2, solve the following systems of equations.

   a. $x - y = 2$
      $x - y = 4$

   b. $x + 2y = 7$
      $4x + y = 5$

   c. $2x + 3y + z = 2$
      $x + 2y - 4z = 3$
      $4x - 2y - z = -2$

   d. $x - y + z = 4$
      $3x - y - z = 1$
      $x + 2y - z = 5$

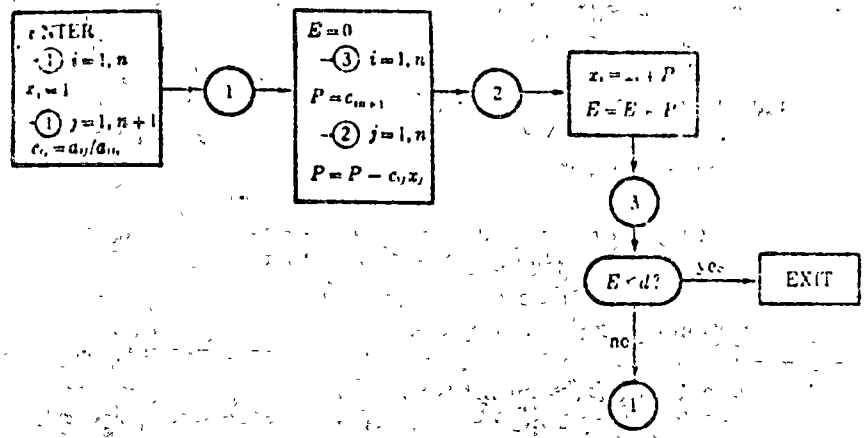


Figure 10-7: Solution of linear equations—Gauss-Seidel method

2. Following the flow chart, Figure 10-7, perform the first four iterations for the following systems of equations.

   a. $2x + y = 3$
      $x + 2y = 3$

   b. $4x - y = 6$
      $x + 3y = -5$

   c. $3x + 2y + z = 5$
      $2x + 5y + 4z = 8$
      $x + 4y + 6z = 4$

   d. $x + 2y + 4z = 6$
      $3x + y + 2z = 5$
      $2x + 4y + z = 4$

3. Write a FORTRAN program that will input a system of linear equations up to size 20 by 20, call subroutine ELIM of Section 10.2 to solve them, and print the result.

4. a. Write a FORTRAN subroutine which will rearrange a set of linear equations, for use of subroutine GAUSID of Section 10.3, so that after rearrangement

   $$a_{ii} \geq a_{ki} \qquad \text{for } k > i$$

   b. Show that your subroutine will correctly arrange the equations

   $$x_1 + 8x_2 + x_3 = 10$$
   $$x_1 + x_2 + 7x_3 = 9$$
   $$9x_1 + x_2 + x_3 = 11$$

   so that the Gauss-Seidel method will converge.

   c. Explain what may go wrong with this method of arrangement if some of the coefficients are zero.

10.4 MATRICES

In all the methods of solving linear equations by computer, we have seen that only the coefficients and the constants appear within the machine. The

formalism of writing down the unknowns x_1, x_2, etc., when we write the equations longhand, merely serves to identify the proper locations of the coefficients and constants. In other words, the solution of the system of equations

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n \end{aligned}$$

is determined completely by the array of coefficients

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

and the array of constants

$$\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

If we are given any two such arrays, we can write the set of equations they represent. If we were to change the numerical value of any number in one of these arrays, a different set of equations would be represented. Further, if we were to interchange the position of any two of the numbers, a still different set of equations would be represented. All this suggests that it may be useful to consider these arrays of numbers as separate entities, establish rules for manipulating them, and perhaps free ourselves somewhat of the repetitious writing of the basically nonessential symbols $x_1 +$, $x_2 +$, $x_3 +$, etc. Considerations such as these have led to the definition of a matrix as an array of numbers, and to the development of an "algebra" of matrices, a set of rules for combining matrices to form other matrices. Once developed, matrix algebra has come to have far-reaching applications, completely apart from systems of linear equations.

10.5 DEFINITIONS AND ELEMENTARY OPERATIONS

A matrix is a rectangular array of quantities or numbers, such as

$$\begin{matrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{matrix}$$

In order to distinguish a matrix from a determinant, which also frequently *looks like* an array of numbers, it is customary to enclose a matrix in brackets, or large parentheses, or double bars, as

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}, \quad \text{or} \quad \begin{Vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{Vmatrix}$$

A determinant is usually written between single bars, as

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

This determinant only looks like an array. Really the symbol only stands for a single quantity which is obtained by multiplying and adding the individual a_{ij}'s in the manner described in Section 10.54. The matrix, on the other hand, has no single numerical value but is instead the entire array. We shall be using a single letter or symbol to stand for a matrix, such as

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

When we do this, it is important to remember that A is not a number, and does not act like a number; that is, it does not obey the ordinary laws of algebra.

Occasionally, we will be interested in the value of a determinant made up of exactly the same elements as some square matrix A. When we do we shall refer to it as the determinant of the matrix A.

A matrix of m rows and n columns is an m by n matrix. If $m = n$, the matrix is a square matrix of order n.

The sum of the diagonal elements of a square matrix is called the "trace" of the matrix, i.e. $A = a_{11} + a_{22} + \cdots + a_{nn}$.

If a matrix consists of a single column it is called a column matrix, or sometimes a column vector.

If the elements in the main diagonal of a square matrix are ones and all the other elements are zeros, the matrix is called a unit matrix, or identity matrix. Thus

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

is a unit matrix of order 3. Unit matrices of any order are usually denoted by the symbol I.

If all the elements are zero, the matrix is called a zero matrix.

Two matrices A and B are said to be equal if:

(1) They have the same number of rows.
(2) They have the same number of columns.
(3) Each pair of corresponding elements are equal.

10.51 Addition and Subtraction of Matrices

The operations of addition and subtraction are defined for two matrices A and B if:

(1) They have the same number of rows.
(2) They have the same number of columns.

The sum of two matrices is the matrix obtained by adding corresponding pairs of elements. Thus, if

$$A=\begin{pmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{pmatrix} \quad \text{and} \quad B=\begin{pmatrix}b_{11} & b_{12} & b_{13}\\ b_{21} & b_{22} & b_{23}\\ b_{31} & b_{32} & b_{33}\end{pmatrix}$$

then

$$A+B=\begin{pmatrix}a_{11}+b_{11} & a_{12}+b_{12} & a_{13}+b_{13}\\ a_{21}+b_{21} & a_{22}+b_{22} & a_{23}+b_{23}\\ a_{31}+b_{31} & a_{32}+b_{32} & a_{33}+b_{33}\end{pmatrix}$$

The difference $A-B$ is the matrix obtained by subtracting the elements of B from the corresponding elements of A.

$$A-B=\begin{pmatrix}a_{11}-b_{11} & a_{12}-b_{12} & a_{13}-b_{13}\\ a_{21}-b_{21} & a_{22}-b_{22} & a_{23}-b_{23}\\ a_{31}-b_{31} & a_{32}-b_{32} & a_{33}-b_{33}\end{pmatrix}$$

Example 1. Find $A+B$ and $A-B$, where

$$A=\begin{pmatrix}3 & 0 & -2\\ 1 & 3 & 1\end{pmatrix}, \qquad B=\begin{pmatrix}2 & 1 & 2\\ -1 & 3 & -2\end{pmatrix}$$

SOLUTION:

$$A+B=\begin{pmatrix}5 & 1 & 0\\ 0 & 6 & -1\end{pmatrix}, \qquad A-B=\begin{pmatrix}1 & -1 & -4\\ 2 & 0 & 3\end{pmatrix}$$

Example 2. Find $A+B$ and $A-B$, where

$$A=\begin{pmatrix}3 & 0 & -2\\ 1 & 3 & 2\end{pmatrix}, \qquad B=\begin{pmatrix}2 & -1\\ 1 & 3\\ 2 & -2\end{pmatrix}$$

Since there are not the same number of rows or columns in A and B, they cannot be added or subtracted. The symbols $A+B$ and $A-B$ are meaningless in this case.

Example 3. Given two matrices A and B, each with N columns and M rows, write FORTRAN statements which would form the sum, $C=A+B$

The matrix A can be represented by a single subscripted variable A(I,J), where I runs from 1 to M and J runs from 1 to N. The same is true for B and C. Then the required FORTRAN statements are

```
      DO 20 I=1M,
      DO 20 J=1,N
   20 C(I,J)=A(I,J)+B(I,J)
```

A total of $N \times M$ additions are required to obtain C

As a direct extension of addition, it would be natural to be able to say

$$A+A=2A$$

This leads to the definition of multiplication of a matrix by a constant as follows: A constant times a matrix is the matrix obtained by multiplying *all* elements of the original matrix by the constant.

10.52 Multiplication of Matrices

At first acquaintance, the operation of multiplication of two matrices seems to be defined in a most peculiar way. There are very good reasons for choosing to call this seemingly awkward process "multiplication," and these will appear shortly.

The product AB of two matrices, A and B, is defined only if the number of *columns* in A is equal to the number of *rows* in B. In all other cases the product is undefined. If the number of columns in A is equal to the number of rows in B, then A and B are said to be "conformable" in the order AB.

The product AB of two conformable matrices is itself a matrix, whose elements are found according to the following rule: The element in the ith row and the jth column of the product is the sum of the products by pairs of the elements of the ith row of A and jth column of B.

Example 1. If

$$A = \begin{pmatrix} 1 & 2 \\ 3 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 3 & -2 \\ 2 & 1 \end{pmatrix}$$

find AB.

Since A has 2 columns and B has 2 rows, A and B are conformable in the order AB, so the product is indeed defined. To find the element in the first row, first column of the product matrix, we take the first row of A, which is

$$1 \quad 2$$

and the first column of B, which is

$$\begin{matrix} 3 \\ 2 \end{matrix}$$

and form the sum of the products by pairs:

$$1 \times 3 + 2 \times 2 = 7$$

Hence 7 is the element in the first row, first column of the product.

In like manner, the element in the first row and second column of the product is obtained from combining the first row of A with the second column of B, thus:

$$1 \times (-2) + 2 \times 1 = 0$$

and for the second row, first column,

$$3 \times 3 + (-1) \times 2 = 7$$

and the second row, second column,

$$3 \times (-2) + (-1) \times 1 = -7$$

Hence the product is

$$\begin{pmatrix} 1 & 2 \\ 3 & -1 \end{pmatrix} \begin{pmatrix} 3 & -2 \\ 2 & 1 \end{pmatrix} = \begin{pmatrix} 7 & 0 \\ 7 & -7 \end{pmatrix}$$

Example 2. If

$$A = \begin{pmatrix} 1 & 3 & 1 \\ -2 & 1 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

find AB.

Since A has 3 columns and B has 3 rows, they are conformable in the order AB. We can expedite the process of finding the product by writing the two matrices side by side, and then going across a row of A and down a column of B forming products by pairs, thus:

$$\begin{pmatrix} 1 & 3 & 1 \\ -2 & 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 1 \times 1 + 3 \times 2 + 1 \times 3 \\ -2 \times 1 + 1 \times 2 - 1 \times 3 \end{pmatrix} = \begin{pmatrix} 10 \\ -3 \end{pmatrix}$$

Example 3. For the matrices A and B of Example 2, find BA.

Since B has 1 column and A has 2 rows, they are not conformable in the order BA. The product BA is not defined!

Example 4. If

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}, \quad B = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1l} \\ b_{21} & b_{22} & \cdots & b_{2l} \\ \vdots & & & \\ b_{n1} & b_{n2} & \cdots & b_{nl} \end{pmatrix}$$

and

$$AB = C$$

write a formula for finding c_{ij}, the element in the ith row and jth column of C.

The ith row of A is

$$a_{i1} \quad a_{i2} \quad \cdots \quad a_{in}$$

and the jth column of B is

$$\begin{matrix} b_{1j} \\ b_{2j} \\ \vdots \\ b_{nj} \end{matrix}$$

and the sum of the products by pairs gives

$$c_{ij} = a_{i1} b_{1j} + a_{i2} b_{2j} + \cdots + a_{in} b_{nj}$$

or, in more abbreviated form,

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

Example 5. Given matrix A with M rows and N columns and matrix B with N rows and L columns, write a set of FORTRAN statements which will form the product $C = AB$.

A suitable set of statements is

```
      DO 10 I=1,M
      DO 10 J=1,L
      C(I,J)=0.
      DO 10 K=1,N
   10 C(I,J)=C(I,J)+A(I,K)*B(K,J)
```

We note that statement 10 is in three DO loops, and will be performed $N \times M \times L$ times, or $N \times M \times L$ multiplications are required to find the product matrix C.

Example 6. If

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}, \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

write the product Ax.

SOLUTION:

$$Ax = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 \end{pmatrix}$$

Note that this product Ax is actually a column vector, having three elements

Example 7. Write the system of linear equations

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}$$

in matrix form

From Example 6, if we define

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}, \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

then the left-hand sides of the equations above are just the three elements

of the column vector Ax. Now let us define the column vector

$$b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

We recall that two matrices are equal if and only if every pair of corresponding elements are equal. Thus, the statement

$$Ax = b$$

is a matrix equation. The expressions on each side of the equals sign are matrices. The equation means that

(1) The first element of Ax, that is $a_{11}x_1 + a_{12}x_2 + a_{13}x_3$, is equal to b_1.

(2) The second element of Ax, that is, $a_{21}x_1 + a_{22}x_2 + a_{23}x_3$, is equal to b_2.

(3) The third element of Ax, that is, $a_{31}x_1 + a_{32}x_2 + a_{33}x_3$, is equal to b_3.

Hence the matrix equation

$$Ax = b$$

says exactly the same thing as the system of linear equations above

We see from Examples 6 and 7 that any system of linear equations, with any number of unknowns, can be represented by a matrix equation

$$Ax = b$$

where A is a matrix and x and b are column vectors of the correct order. This simple expression is one of the several happy results of the seemingly odd definition of multiplication.

10.53 Laws of Matrix Algebra

We have defined three operations with matrices and have given them the names "addition," "subtraction," and "multiplication"—names we use in the ordinary algebra of numbers. Actually this is a little dangerous, since it suggests that these new matrix operations will obey the same rules as the ordinary arithmetic operations, and we really have no right to expect that they will do so.

The fundamental laws of ordinary algebra are the following

(1). Addition is *commutative*. $a + b = b + a$; that is, if we add b to a, or a to b, we will get the same result

(2). Addition is *associative*. $(a + b) + c = a + (b + c)$; that is, if we add $a + b$, and then add c to this sum, we get the same result as if we add b and c first, and then add a to the sum.

(3). Multiplication is *distributive* with respect to addition. $a(b + c) = ab + ac$; that is, if we add b to c and then multiply by a, we get the same result as if we multiply a by b, multiply a by c, and then add the result.

(4). Multiplication is *commutative*. $ab = ba$; that is, if we multiply a by b or b by a, we get the same result.

(5). Multiplication is *associative*. $(ab)c = a(bc)$; that is, if we take the product ab and multiply by c we get the same answer as if we take the product bc and multiply by a.

When these laws for the algebra of numbers are investigated for matrices, it is found that they all hold *except* law 4, the commutative law for multiplication. As was seen in Examples 2 and 3, Section 10.52, it is possible to have two matrices whose product AB could be found but whose product BA was not even defined.

In summary, then, we can say that, in expressions involving sums, differences, and products of matrices, we can use the same laws for combining these operations as for ordinary numbers except that the order of any two matrices in a product cannot be reversed. In a matrix equation, we may add the same matrix to both sides or subtract the same matrix from both sides without changing the equality. We also may *multiply* both sides by the same matrix, provided that:

(1) The matrix is conformable with those by which it is to be multiplied.

(2) The order of multiplication is made the same on both sides of the equation.

Example 1. If A, B, and C are square matrices of order n, and if

$$A + B = C$$

solve for A.

Subtracting B from both sides, we have

$$A = C - B$$

Example 2. If A, B, C, and D are square matrices of order n, and if $A = B + C$, find AD and DA.

Multiplying the equation

$$A = B + C$$

on the right by D, we have

$$AD = (B + C)D = BD + CD$$

Multiplying the above equation on the left by D, we have

$$DA = D(B + C) = DB + DC$$

10.54 Determinants

The determinant of a square matrix A is defined to be the number obtained in the following manner: From the elements of A, we form all possible products containing exactly one element from each row and column in A. To each such term we assign a plus or minus sign in accordance with a rule to be stated shortly. The sum of these terms is the value of the determinant. The sign to be assigned to a term is determined by the following procedure. The factors in the term are arranged in order according to the row from which each factor was chosen:

$$a_{1k_1}a_{2k_2}a_{3k_3}\cdots a_{nk_n}$$

We then rearrange these factors so that they are in order according to the column from which each was chosen, that is, so that the subscripts $k_1, k_2, \ldots, k_n$ are in their natural order, and count the number of interchanges required to do this. We assign the term a plus sign if the number of interchanges was even and a minus sign if it was odd. For a 2 by 2 determinant, then,

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{21}a_{12}$$

For a 3 by 3 system,

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31}$$

It is clear that, by utilizing the programming methods of the earlier chapters, we can cause a computer to perform such calculations and provide the solution to a system of equations. It is not so obvious, but it can be shown that such a procedure is quite inefficient in machine time, particularly for systems involving a very large number of unknowns. According to the rule just stated for evaluating a determinant, an n by n determinant is the sum of $n!$ terms, each of which is the product of n numbers. If we were to calculate the value of a determinant by the most direct method, then, about $n \times n!$ multiplications would be required. For even a 10 by 10 determinant, several

million multiplications would be required, and for a 20 by 20 determinant, over 10^{18} multiplications would be needed. This would require over 100,000 years even on the fastest computers.

There is another method of evaluation of a determinant that is very much faster than the brute-force approach. If all elements on one row of a determinant are changed by adding or subtracting a constant multiple of the corresponding elements of another row, the value of the determinant is unchanged. By repeated application of this rule, we can reduce a determinant to a "triangular" form, in which all elements below the main diagonal are zero. For example,

$$\begin{vmatrix} b_{11} & b_{12} & b_{13} & \cdots & b_{1n} \\ 0 & b_{22} & b_{23} & \cdots & b_{2n} \\ 0 & 0 & b_{33} & \cdots & b_{3n} \\ 0 & 0 & 0 & b_{44} \cdots & b_{4n} \\ & \vdots & & & \\ 0 & 0 & 0 & \cdots & b_{nn} \end{vmatrix}$$

The value of a determinant when written in this form turns out to be just the product of the diagonal elements, $b_{11}b_{22}b_{33}\cdots b_{nn}$, since all other terms formed in accordance with the definition of a determinant's value contain at least one factor whose value is zero. Hence, after a determinant is written in triangular form, only $n-1$ multiplications are required to find its value.

The process is quite similar to that used in Section 10.2 to solve a system of linear equations by the elimination method. We start out with the array

$$\begin{matrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & & & & \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{matrix}$$

and perform the operations

$$a_{ij} - \frac{a_{ik}a_{kj}}{a_{kk}} \rightarrow a_{ij} \qquad \text{for } i \text{ and } j = k+1, k+2, \ldots, n$$
$$k = 1, \ldots, n$$

Figure 10-8 is a flow chart of the process

A calculation based on the flow chart, Figure 10-8, could run into trouble if a_{kk} ever becomes zero, since there is a division by this quantity. This problem can be avoided by taking the additional precaution of checking to see if a_{kk} is zero and if so interchanging two rows to obtain a nonzero value for a_{kk}. Since interchanging two rows in a determinant changes the sign of the determinant's value, we must also change the signs of the elements in one of the rows to correct this.

There can also be accuracy problems associated with evaluating a determinant using the above flow chart, particularly for determinants of large order. These problems tend to be alleviated if the rows and columns are rearranged at each step so that a_{kk} is not only nonzero but is actually the largest element in absolute value. SUBROUTINE EXCH of Section 10.2

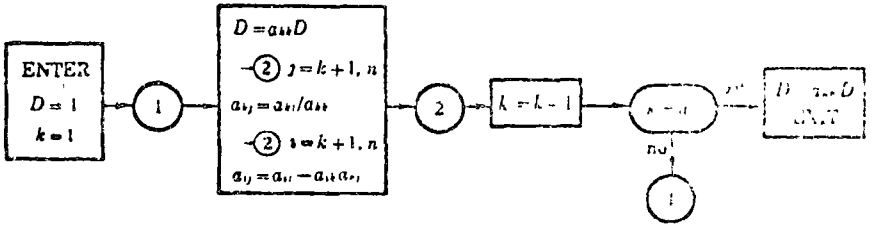


Figure 10-8: Evaluation of a determinant

provided this service for the elimination method, and with [illegible] modifications it can be made to work in the present case. The [illegible] that interchanging rows or columns in a determinant changes the sign of the determinant. We can correct for this by changing statement 32 to read

32 A(K,J) = −C

and statement 42 to read

42 A(I,K) = −C

We also need the dimension statement to read A(20,20) instead of A(20,21). We do not need the quantity ID as output, so we can eliminate the three statements following statement 42 and change the first statement to read

SUBROUTINE EXCH2(A,N,NN,K)

We changed the name as well, to ensure that the old routine of Section 10.2 is not used by mistake.

Another step which is useful to avoid undetected machine errors is in connection with the computation

$$a_{ij} = a_{ij} - a_{ik}a_{kj}$$

If the result of this subtraction is supposed to be zero, then [illegible] will be subject to the trouble mentioned many times earlier [illegible]

of accuracy caused by introduction of leading zeros. The method of protection against this trouble is the same one used in division of polynomials in Section 9.43. We check the result of the subtraction, and if the difference is much smaller than the numbers being subtracted, we set the difference equal to zero. The operation can be described by a section of flow chart (as in Figure 10-9) in which a_{ij} is set equal to zero if more than four significant figures have been lost in the subtraction.

The FORTRAN subroutine below evaluates the Nth-order determinant

$$\begin{vmatrix} A(1,1) & A(1,2) & \cdots & A(1,N) \\ A(2,1) & A(2,2) & \cdots & A(2,N) \\ \vdots & & & \\ A(N,1) & A(N,2) & \cdots & A(N,N) \end{vmatrix}$$

for values of N to 20. In the first statement, the determinant is given the name AA, and the statements up to 100 redefine the elements so that the original determinant will not be destroyed during the calculation. State-

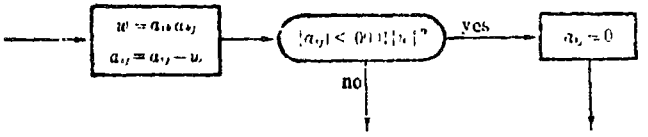


Figure 10-9

ment 1 calls EXCH2 to interchange rows and columns if necessary to move the largest element to location A(K,K). Statements 3 through 4 perform the actual calculation required in the main part of the flow chart, Figure 10-8.

```
      SUBROUTINE DETERM(AA,N,D)
      DIMENSION AA(20,20),A(20,20)
      DO 100 I=1,N
      DO 100 J=1,N
  100 A(I,J)=AA(I,J)
      D=1.
      K=1
    1 CALL EXCH2(A,N,N,K)
      D=A(K,K)*D
      IF(A(K,K))3,10,3
    3 KK=K+1
      DO 4 J=KK,N
      A(K,J)=A(K,J)/A(K,K)
```



```
      DO 4 I=KK,N
      W=A(I,K)*A(K,J)
      A(I,J)=A(I,J)-W
      IF(ABS(A(I,J))-.0001*ABS(W))42,4,4
   42 A(I,J)=0.
    4 CONTINUE
      K=KK
      IF(K-N)1,9,10
    9 D=A(N,N)*D
   10 RETURN
      END
```

10.55 Matrix Inversion

We have given definitions and rules for the addition, subtraction, and multiplication of matrices which parallel to some extent the rules of ordinary algebra. As yet we have not mentioned division, for the very good reason that division as such is not defined for matrices. There is, however, an operation which serves a somewhat analogous purpose, however. That operation is the "inversion" of a matrix.

In ordinary algebra, b/a stands for the number which, when multiplied by a, gives b. Thus, if $ax = b$, we can say that $x = b/a$. Instead of defining division in this manner, we could define an "inverse" of a number. For any number a, the inverse, a^{-1}, is that number which, when multiplied by a gives 1. Every nonzero number has a unique inverse, for example the inverse of 2 is .5, and .5 is the *only* inverse of 2. Then if we have $ax = b$, we do not even have to have a process of division in order to find x, for we can multiply both sides of the equation by a^{-1}, giving

$$a^{-1}ax = a^{-1}b \qquad \text{or} \qquad x = a^{-1}b$$

For square matrices, we define the inverse in a manner analogous to that above. For a square matrix A of order n, the inverse matrix, A^{-1}, is that matrix which when multiplied by A gives the identity matrix of order n, that is,

$$AA^{-1} = I$$

Example 1. Show that the inverse of

$$\begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix}$$

$$A^{-1} = \begin{pmatrix} 1 & 1 \end{pmatrix}$$

We form the product

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Since the product is the identity matrix, the second matrix is indeed the inverse of the first.

It can be shown that any square matrix A has a unique inverse if and only if its determinant is different from zero. It can also be shown that A commutes with its inverse; that is,

$$AA^{-1} = A^{-1}A = I$$

The inverse is not defined for nonsquare matrices.

A formula for the inverse of a matrix A can be found as follows. Consider the set of linear equations

$$\begin{aligned} y_1 &= a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ y_2 &= a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \\ &\cdots \\ y_n &= a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n \end{aligned} \tag{10-8}$$

connecting one set of variables $x_1, x_2, \ldots, x_n$ with another set $y_1, y_2, \ldots, y_n$. In matrix form we can write this set of equations as

$$y = Ax$$

where

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

If we multiply this set of equations by A^{-1}, we obtain

$$A^{-1}y = A^{-1}Ax$$

or

$$A^{-1}y = \left(\frac{1}{n}\right)Ix = \left(\frac{1}{n}\right)x$$

Hence the elements of A^{-1} are just the coefficients of the y's if we solve the set of equations (10-8) for the x's in terms of the y's.

A rather efficient way of solving for the x's in terms of the y's is to proceed as in the elimination method, Section 10.2. Instead of the n by $n+1$ array of constants shown at (10-3) we start with the n by $2n$ array of the form

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & 1 & 0 & 0 & \cdots & 0 \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} & 0 & 1 & 0 & \cdots & 0 \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} & 0 & 0 & 1 & \cdots & 0 \\ & & & & & & & & & \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} & 0 & 0 & 0 & \cdots & 1 \end{pmatrix} \tag{10-9}$$

Then we proceed exactly as in Section 10.2, and end up with an array of the form

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & a_{1n+1} & a_{1n+2} & a_{1n+3} & \cdots & a_{12n} \\ 0 & 1 & 0 & \cdots & 0 & a_{2n+1} & a_{2n+2} & a_{2n+3} & \cdots & a_{22n} \\ 0 & 0 & 1 & \cdots & 0 & a_{3n+1} & a_{3n+2} & a_{3n+3} & \cdots & a_{32n} \\ & & & & & & & & & \\ 0 & 0 & 0 & \cdots & 1 & a_{nn+1} & a_{nn+2} & a_{nn+3} & \cdots & a_{n2n} \end{pmatrix}$$

The solution, instead of being the single column $a_{1n+1}, \ldots, a_{nn+1}$, is the entire right-hand side of the above array. If we have interchanged any rows in the process, the unscrambling is just as indicated in Section 10.4, except that the present case the whole rows must be unscrambled. The solution can be written immediately as an almost direct paraphrase of SUBROUTINE ELIM.

```
      SUBROUTINE MATINV(AA,N,AINV)
      DIMENSION AA(20,20),AINV(20,20),A(20,40),B(20)
      NN=N+1
      N2=2*N
      DO 100 I=1,N
      B(I)=0.
      DO 100 J=1,N
  100 A(I,J)=AA(I,J)
      DO 200 I=1,N
      DO 200 J=NN,N2
```

```
200 A(I,J)=0.
    DO 300 I=1,N
300 A(I,N+1)=1.
    K=1
  1 CALL EXCH3(A,N,N2,K,ID)
  2 IF(A(K,K))3,999,3
  3 KK=K+1
    DO 4 J=KK,N2
    A(K,J)=A(K,J)/A(K,K)
    DO 4 I=1,N
    IF(K-I)41,4,41
 41 W=A(I,K)*A(K,J)
    A(I,J)=A(I,J)-W
    IF(ABS(A(I,J))-.0001*ABS(W))42,4,4
 42 A(I,J)=0.
  4 CONTINUE
    K=KK
    IF(K-N)1,2,5
  5 DO 10 J=1,N
    DO 10 J=1,N
    IF(ID(J)-I)10,8,10
  8 DO 10 K=1,N
    AINV(I,K)=A(J,N+K)
 10 CONTINUE
    RETURN
999 PRINT 1000
    RETURN
1000 FORMAT(19H MATRIX IS SINGULAR)
    END
```

In this subroutine the statements through 300 move the quantities to working storage to form the array depicted by (10-9). Statement 1 calls a version of SUBROUTINE EXCH given in Section 10.2. It is called EXCH3, to indicate that it must be a modified version of that subroutine with dimension statement changed to read

```
DIMENSION A(20,40)
```

Statements down to statement 4 parallel SUBROUTINE ELIM, except that the accuracy flag shown in Figure 10-9 has been inserted at statement 42. In the loops terminating on statement 10, instead of setting individual values of the X(I)'s, the subroutine sets entire rows of the inverse matrix AINV(I,K).

Once an inverse matrix has been obtained, an improved accuracy version can be obtained in a relatively straightforward manner. Let A be the matrix

to be inverted, and let D_1 be the approximate inverse produced by the above routine. Then, because of inaccuracies,

$$AD_1 \neq I$$

but instead

$$I - AD_1 = F_1$$

where F_1 is a matrix which, if D_1 was a reasonably good estimate, has small elements. If all the elements of F_1 are less than one in absolute value, then the matrix D_2 defined by

$$D_2 = D_1(I + F_1)$$

is an improved estimate of A^{-1}. If the error matrix $F_2 = I - AD_2$ still has elements which are too large, then the matrix D_3 defined by $D_3 = D_2(I + F_2)$ is a still better estimate, and so on. Thus repetition of a process involving some matrix multiplications can be used to improve the accuracy of the inverse to the extent desired, within the limits imposed by the usual problems of approximate arithmetic on computers.

EXERCISE 27

1. Given the following matrices

$$A = \begin{pmatrix} 2 & 2 \\ 1 & -1 \\ -2 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & -1 \\ 2 & 3 \end{pmatrix}, \quad C = \begin{pmatrix} 2 \\ 1 \\ -2 \end{pmatrix}$$

$$D = \begin{pmatrix} 4 & 1 & 3 \\ 2 & -1 & 1 \\ -3 & 2 & 1 \end{pmatrix}, \quad E = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad F = \begin{pmatrix} 1 & -1 & 2 \\ 2 & -3 & 1 \end{pmatrix}$$

evaluate the following expressions, or, if the expression is meaningless, so state

a. AB b. DC c. BE
d. BA e. ADE f. $DA + A$
g. $FA + B$ h. $FC + BE$ i. $FDABE$
j. $AF + D$

2. Using the method of Section 10.55, invert the following matrices

a. $\begin{pmatrix} 3 & 2 \\ 4 & 3 \end{pmatrix}$ b. $\begin{pmatrix} 1 & -3 \\ 2 & 4 \end{pmatrix}$

c. $\begin{pmatrix} 2 & 3 & 1 \\ 1 & -1 & 2 \\ -3 & 1 & -1 \end{pmatrix}$ d. $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 2 \\ 0 & 4 & 3 \end{pmatrix}$

3. Find A^{-1}, then solve $Ax = b$ by multiplying both sides by A^{-1}, if

a. $A = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix}$, $b = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$

b. $A = \begin{pmatrix} 2 & 4 & -1 \\ -1 & -3 & 1 \\ 3 & -1 & 2 \end{pmatrix}$, $b = \begin{pmatrix} 4 \\ -2 \\ 6 \end{pmatrix}$

4. Write a FORTRAN subroutine INVIMP that will take the trial inverse obtained from MATINV and use the method described at the end of Section 10.55 to improve the inverse until all the elements of the error matrix F_n are less than .001 in absolute value

10.6 OVERDETERMINED AND UNDERDETERMINED SYSTEMS OF LINEAR EQUATIONS

In several of the preceding sections, methods were discussed for solving systems of linear equations. In all these discussions it was assumed that there was a unique solution and that there were just as many equations as unknowns. Further, it was tacitly assumed that the equations were nonhomogeneous, that is, not all the constant terms were zero, and also that the determinant of the coefficients was not zero. With these conditions satisfied there *is* a unique solution. In many important cases, however, these conditions are not all satisfied—yet there may still be a unique solution, or there may be no solution or an infinite number of solutions. In this section we will discuss a method for finding which situation prevails and for completely describing the solutions when there is an infinite number of them.

10.61 Rank of a Matrix

As a tool for further study of systems of equations we will need the concept of rank of a matrix.

Definition. The rank of a matrix is the order of the highest-order nonvanishing determinant within the matrix.

By a "determinant within the matrix" we mean any determinant that can be made by crossing out rows or columns in the matrix.

Example 1. Find the rank of the matrix

$$\begin{pmatrix} -1 & 1 & 2 \\ -3 & 3 & 1 \end{pmatrix}$$

The largest-order determinant we can construct is second order, so the rank is 2 or less. To see if it is 2, we must check all second-order determinants. If we cross out the third column, we can construct the determinant

$$\begin{vmatrix} -1 & 1 \\ -3 & 3 \end{vmatrix}$$

which has the value zero. Since this one vanishes, we must check other second-order determinants. Crossing out the second column in the matrix, we obtain the determinant

$$\begin{vmatrix} -1 & 2 \\ -3 & 1 \end{vmatrix}$$

which has the value 5. Since there is a nonvanishing second-order determinant, the rank is 2.

Example 2. Find the rank of the matrix

$$\begin{pmatrix} 1 & 2 & 3 \\ -1 & -2 & -3 \\ 2 & 4 & 6 \end{pmatrix}$$

The largest-order determinant we can construct is third order, so the rank is 3 or less. The only third-order determinant is

$$\begin{vmatrix} 1 & 2 & 3 \\ -1 & -2 & -3 \\ 2 & 4 & 6 \end{vmatrix} = 0$$

so the rank is not 3. If we cross out the third row and third column, we have the determinant

$$\begin{vmatrix} 1 & 2 \\ -1 & -2 \end{vmatrix} = 0$$

Similarly, if we check all other second-order determinants, we find that they all vanish.

Hence the rank is less than 2. If we cross out the second and third rows, and the second and third columns, we can form the determinant $|1| = 1$. Since the highest-order nonvanishing determinant is first order, the rank of the matrix is 1.

It is seen from the above examples that finding the rank of a matrix is a straightforward process. For matrices of higher order, however, the process

as just demonstrated is extremely laborious, sometimes involving the evaluation of many determinants. Fortunately, however, a less laborious method is available, based on the following theorem:

Theorem 1. *The rank of a matrix is unchanged if any multiple of the elements of one row (or columns) is added to the corresponding elements of another row (or column).*

This theorem means that we can proceed, just as in evaluating a determinant, to combine rows or columns to obtain zeros where we choose.

Example 3. Find the rank of

$$\begin{pmatrix} 1 & -1 & -1 & -2 \\ 2 & 1 & -2 & 2 \\ 4 & 3 & -4 & 6 \end{pmatrix}$$

Using Theorem 1, we may proceed as follows:

$$\text{rank}\begin{pmatrix} 1 & -1 & -1 & -2 \\ 2 & 1 & -2 & 2 \\ 4 & 3 & -4 & 6 \end{pmatrix} = \text{rank}\begin{pmatrix} 1 & -1 & -1 & -2 \\ 0 & 3 & 0 & 6 \\ 4 & 3 & -4 & 6 \end{pmatrix}$$ (twice first row subtracted from second)

$$= \text{rank}\begin{pmatrix} 1 & -1 & -1 & -2 \\ 0 & 3 & 0 & 6 \\ 0 & 7 & 0 & 14 \end{pmatrix}$$ (four times first row subtracted from third)

$$= \text{rank}\begin{pmatrix} 1 & -1 & -1 & -2 \\ 0 & 3 & 0 & 6 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$ (7/3 times second row subtracted from third)

It is obvious in this last matrix all third-order determinants are zero, but at least one second-order determinant,

$$\begin{vmatrix} 1 & -1 \\ 0 & 3 \end{vmatrix}$$

is not zero. Hence the rank of the original matrix is 2.

Note that in the above example, we have *not* said the *matrices* obtained at each step are equal, but only that the *ranks* are equal. Each step has created a new matrix, one differing from the preceding in many respects, but having the rank in common.

It is seen that the method of determining rank as demonstrated in Example 3 is closely akin to the method of evaluation of a determinant given in Section

10.54. Minor modifications to the program given there will give a program for finding the rank of a matrix with no more effort than that involved in evaluating the largest determinant in the matrix.

The FORTRAN subroutine below finds the rank, K, of a matrix having N rows and M columns, where neither N nor M exceed 20.

```
      SUBROUTINE MARANK(AA,N,M,K)
      DIMENSION AA(20,20),A(20,20)
      DO 100 I=1,N
      DO 100 J=1,M
  100 A(I,J)=AA(I,J)
      K=1
    1 CALL EXCH2(A,N,M,K)
      IF(A(K,K))2,10,2
    2 IF(K-N)3,11,11
    3 IF(K-M)40,11,11
   40 KK=K+1
      DO 4 J=KK,M
      A(K,J)=A(K,J)/A(K,K)
      DO 4 I=KK,N
      W=A(I,K)*A(K,J)
      A(I,J)=A(I,J)-W
      IF(ABS(A(I,J))-.0001*ABS(W))42,4,4
   42 A(I,J)=0.
    4 CONTINUE
      K=KK
      GO TO 1
   10 K=K-1
   11 RETURN
      END
```

10.62 Consistent and Inconsistent Equations

A set of linear equations

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1m}x_m &= b_1 \\ a_{21}x_1 + \cdots + a_{2m}x_m &= b_2 \\ &\vdots \\ a_{n1}x_1 + \cdots + a_{nm}x_m &= b_n \end{aligned}$$

is said to be consistent if there exists at least one solution

if there is no solution. We are now in a position to give a criterion for determining whether a set of equations is consistent or inconsistent. We will refer to the matrix

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & & & \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix}$$

as the *coefficient* matrix, and to the matrix

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2m} & b_2 \\ \vdots & & & & \\ a_{n1} & a_{n2} & \cdots & a_{nm} & b_n \end{pmatrix}$$

as the *augmented* matrix. Then the following theorem applies:

Theorem 2. *A set of linear equations is consistent if and only if the coefficient matrix and augmented matrix have the same rank.*

Example 1. Determine if the following equations are consistent:

$$\begin{aligned} x + 3y &= 4 \\ 2x + 6y &= 2 \end{aligned}$$

The coefficient matrix is

$$\begin{pmatrix} 1 & 3 \\ 2 & 6 \end{pmatrix}$$

which has rank 1.

The augmented matrix is

$$\begin{pmatrix} 1 & 3 & 4 \\ 2 & 6 & 2 \end{pmatrix}$$

which has rank 2.

Hence the system is inconsistent.

Example 2. Determine if the following equations are consistent:

$$\begin{aligned} x + 2y &= 3 \\ 2x - y &= 2 \\ 3x + y &= 5 \end{aligned}$$

The coefficient matrix is

$$\begin{pmatrix} 1 & 2 \\ 2 & -1 \\ 3 & 1 \end{pmatrix}$$

which has rank 2.

The augmented matrix is

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & -1 & 2 \\ 3 & 1 & 5 \end{pmatrix}$$

which has rank 2.

Hence the equations are consistent, and there is a solution, despite the fact that there are more equations than unknowns! Upon closer scrutiny, it will be observed that the third equation is merely the sum of the first two.

The last example illustrates an important principle, that consistency or inconsistency cannot be ascertained merely from the numbers of equations and unknowns. A system with more equations than unknowns can be consistent, and a system with more unknowns than equations can be inconsistent. The subroutine for finding rank given in Section 10.62 is the tool needed to investigate consistency in the larger systems.

10.63 Linear Independence of Vectors

Consistent systems of linear equations may have infinitely many solutions. It is possible, however, to investigate these solutions systematically and to characterize them completely. To do so we need first the concept of linear dependence and independence. Consider the set of column vectors

$$\mathbf{u}_1 = \begin{pmatrix} u_{11} \\ u_{21} \\ \vdots \\ u_{n1} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} u_{12} \\ u_{22} \\ \vdots \\ u_{n2} \end{pmatrix}, \ldots, \quad \mathbf{u}_r = \begin{pmatrix} u_{1r} \\ u_{2r} \\ \vdots \\ u_{nr} \end{pmatrix}$$

If $c_1, c_2, \ldots, c_r$ are any constants, the expression

$$c_1\mathbf{u}_1 + c_2\mathbf{u}_2 + \cdots + c_r\mathbf{u}_r$$

is called a "linear combination" of the vectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$. If there is some set of constants $c_1, \ldots, c_r$, not all zero, such that

$$c_1u_1 + c_2u_2 + \cdots + c_ru_r = 0$$

then the vectors are said to be "linearly dependent." If, on the other hand, every linear combination of the vectors $u_1, \ldots, u_r$ is nonzero except for the case $c_1 = c_2 = \cdots = c_r = 0$, then the vectors are said to be "linearly independent."

Example 1. Are the vectors

$$\begin{pmatrix}0\\1\end{pmatrix} \quad \text{and} \quad \begin{pmatrix}1\\0\end{pmatrix}$$

linearly independent?

The sum

$$c_1\begin{pmatrix}0\\1\end{pmatrix} + c_2\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}c_2\\c_1\end{pmatrix}$$

is zero only if both c_1 and c_2 are zero. Hence they are linearly independent.

Example 2. Are the vectors

$$\begin{pmatrix}1\\-1\\2\end{pmatrix}, \quad \begin{pmatrix}2\\1\\1\end{pmatrix}, \quad \text{and} \quad \begin{pmatrix}3\\0\\3\end{pmatrix}$$

linearly independent?

The sum

$$c_1\begin{pmatrix}1\\-1\\2\end{pmatrix} + c_2\begin{pmatrix}2\\1\\1\end{pmatrix} + c_3\begin{pmatrix}3\\0\\3\end{pmatrix} = \begin{pmatrix}c_1 + 2c_2 + 3c_3\\-c_1 + c_2\\2c_1 + c_2 + 3c_3\end{pmatrix}$$

is zero if $c_1 = 1$, $c_2 = 1$, $c_3 = -1$. Hence the vectors are not linearly independent.

10.64 Complete Solution of Systems of Linear Equations

The following theorem gives a complete picture of the situation regarding solutions for systems of linear equations.

Theorem 3. *Let $Ax = b$ be a consistent system having m unknowns, and let the rank of A be r. Then:*

(1) If $r = m$, there is a unique solution vector x.

(2) If $r < m$, then there is at least one solution vector x. In addition, $m - r$ linearly independent vectors $u_1, u_2, \ldots, u_{m-r}$ can be found which are solutions to the set of homogeneous equations $Ax = 0$. The vector x plus any linear combination of these is also a solution of the given equation, and there are no other solutions. If $b = 0$, the vector x can be taken as $x = 0$.

Hereafter we will refer to the vector x described in this theorem as a *particular* solution.

A method of obtaining all these solutions in a systematic fashion is illustrated by the example below.

Example 1. Solve the system

$$\begin{pmatrix}4 & 2 & -1 & 1\\1 & -1 & 2 & -1\\3 & 3 & -3 & 2\\2 & -2 & 4 & -2\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\end{pmatrix} = \begin{pmatrix}6\\1\\5\\2\end{pmatrix}$$

We will proceed as in the elimination method as illustrated in Section 10.5. Dividing the first equation by 4 and using it to eliminate x_1 from the remaining equations,

$$\begin{pmatrix}1 & .5 & -.25 & .25\\0 & -1.5 & 2.25 & -1.25\\0 & 1.5 & -2.25 & 1.25\\0 & -3 & 4.5 & -2.5\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\end{pmatrix} = \begin{pmatrix}1.5\\-.5\\.5\\-1\end{pmatrix}$$

Rearranging to make the largest element to be in the proper position,

$$\begin{pmatrix}1 & -.25 & .5 & .25\\0 & 4.5 & -3 & -2.5\\0 & -2.25 & 1.5 & 1.25\\0 & 2.25 & -1.5 & -1.25\end{pmatrix}\begin{pmatrix}x_1\\x_3\\x_2\\x_4\end{pmatrix} = \begin{pmatrix}1.5\\-1\\.5\\-.5\end{pmatrix}$$

Dividing the second equation by 4.5 and using it to eliminate x_3 from the other equations,

$$\begin{pmatrix}1 & 0 & 1/3 & 1/9\\0 & 1 & -2/3 & -5/9\\0 & 0 & 0 & 0\\0 & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\x_3\\x_2\\x_4\end{pmatrix} = \begin{pmatrix}13/9\\-2/9\\0\\0\end{pmatrix}$$

At this point we see that the rank of A is 2 and that the system now has

two equations. If the system had been inconsistent, there would be more than two nonzero elements remaining on the right-hand side of the equation at this point.

Since there are four unknowns and the rank of A is 2, Theorem 2 tells us that the complete solution is made up of a particular solution and any linear combination of two linearly independent solution vectors.

We can find the particular solution by setting $x_2 = x_4 = 0$. Then the system becomes

$$x_1 = 13/9$$
$$x_3 = -2/9$$

Hence the particular solution is

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 13/9 \\ 0 \\ -2/9 \\ 0 \end{pmatrix}$$

To find two linearly independent solution vectors, we take the homogeneous equation

$$\begin{pmatrix} 1 & 0 & 1/3 & -1/9 \\ 0 & 1 & -2/3 & -5/9 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

and choose arbitrary values for x_2 and x_4.

Taking $x_2 = 1, x_4 = 0$, we have

$$x_1 + 1/3 = 0$$
$$x_3 - 2/3 = 0$$

which has the solution

$$x_1 = -1/3, \qquad x_3 = 2/3$$

so one of the linearly independent vectors is

$$u_1 = \begin{pmatrix} -1/3 \\ 1 \\ 2/3 \\ 0 \end{pmatrix}$$

Taking $x_2 = 0, x_4 = 1$, we have

$$x_1 + 1/9 = 0$$
$$x_3 - 5/9 = 0$$

which has the solution

$$x_1 = -1/9, \qquad x_3 = 5/9$$

and so the other solution is

$$u_2 = \begin{pmatrix} -1/9 \\ 0 \\ 5/9 \\ 1 \end{pmatrix}$$

and the general solution is

$$x = \begin{pmatrix} 13/9 \\ 0 \\ -2/9 \\ 0 \end{pmatrix} + c_1 \begin{pmatrix} -1/3 \\ 1 \\ 2/3 \\ 0 \end{pmatrix} + c_2 \begin{pmatrix} -1/9 \\ 0 \\ 5/9 \\ 1 \end{pmatrix}$$

where c_1 and c_2 are arbitrary constants.

For convenience in organizing a computer solution, we note that the vectors (apart from a constant multiple of -1 in some cases) can be read from the last set of equations by the following somewhat artificial steps.

(1) Add -1's down the last two columns of the diagonal of the coefficient matrix so that it becomes

$$\begin{pmatrix} 1 & 0 & 1/3 & -1/9 \\ 0 & 1 & -2/3 & -5/9 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

(2) Rearrange these last two columns and the corresponding rows so that they were ordered just as the x's are

$$\begin{pmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{pmatrix}$$

and needed to be correctly ordered. They become

$$\begin{pmatrix} 1/3 & 1/9 & 13/9 \\ -1 & 0 & 0 \\ -2/3 & -5/9 & -2/9 \\ 0 & -1 & 0 \end{pmatrix}$$

The column of constants has become the particular solution and the other two columns two linearly independent vectors that can be used to form the complete solution.

The method just demonstrated is a general one, and can be used for computer solution of larger systems. It requires only a few modifications and extensions of the elimination method given in Section 10.2.

The FORTRAN subroutine given below solves a system of N equations in M unknowns, where N and M are both 20 or less. Inputs are AA, the coefficient matrix; BB, the constant vector; and NI and M, the dimensions of the system. Outputs are: X, a particular solution vector, K, the number of linearly independent solution vectors for the homogeneous system, and U, a set of linearly independent solution vectors.

```
      SUBROUTINE LINEQ(AA,NI,M,BB,X,K,U)
      DIMENSION AA(20,20),BB(20),A(20,21),X(20),ID(20),U(20,20)
      N=NI
      MM=M+1
      DO 100 I=1,N
      A(I,MM)=BB(I)
      DO 100 J=1,M
  100 A(I,J)=AA(I,J)
      K=1
      IF(N-M)200,1,1
  200 NP=N+1
      N=M
      DO 300 I=NP,M
      DO 300 J=1,MM
  300 A(I,J)=0.
      K=1
    1 CALL EXCH(A,M,MM,K,ID)
      IF(A(K,K))2,5,2
    2 KK=K+1
      DO 3 J=KK,MM
      A(K,J)=A(K,J)/A(K,K)
      DO 3 I=1,N
      IF(I-K)31,3,31
   31 W=A(I,K)*A(K,J)
      A(I,J)=A(I,J)-W
      IF(ABS(A(I,J))-.0001*ABS(W))32,3,3
```



```
   32 A(I,J)=0.
    3 CONTINUE
      K=KK
      IF(K-M)1,2,7
    5 DO 6 J=K,M
      A(J,J)=-1.
      DO 7 I=K,N
      IF(A(I,MM))999,7,999
    7 CONTINUE
      DO 10 I=1,M
      DO 10 J=1,M
      IF(ID(J)-I)10,8,10
    8 X(I)=A(J,MM)
      IF(K-MM)9,10,10
    9 KM=K-1
      DO 10 IP=K,M
      U(I,IP-KM)=A(J,IP)
   10 CONTINUE
      K=M-K
      RETURN
  999 PRINT 1000
      RETURN
 1000 FORMAT(27H EQUATIONS ARE INCONSISTENT
      END
```

10.7 EIGENVALUES AND EIGENVECTORS

A surprisingly large number of problems in physics and engineering can be reduced to the following mathematical problem: Given a square matrix A, find a nonzero vector x and a constant λ such that

$$Ax = \lambda x$$

That is, find a vector x such that Ax is simply a multiple of x itself. We can rewrite this equation as

$$Ax - \lambda x = 0$$

or

$$(A - \lambda I)x = 0$$

In this form, the equation appears as a set of homogeneous

for $x_1, x_2, \ldots, x_n$. The matrix of coefficients is $(A - \lambda I)$, and the augmented matrix is the same with a column of zeros added, so by Theorem 2 of Section 10.62 the equations are consistent. By Theorem 3 of Section 10.63 there is a unique solution if the rank of the coefficient matrix is n. We already know that solution; it is $x_1 = x_2 = \cdots = x_n = 0$. Hence there is a nonzero vector x only if the rank of $(A - \lambda I)$ is less than n. This will be true if

$$\det(A - \lambda I) = 0 \tag{10-11}$$

If this determinant is zero, then by Theorem 3 of Section 10.64 there are one or more linearly independent solution vectors that can be used to describe the complete solution. Thus we are interested in the values of λ for which

$$\begin{vmatrix} a_{11} - \lambda & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} - \lambda & & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} - \lambda \end{vmatrix} = 0$$

In Section 10.54 it was stated that the value of a determinant could be obtained by forming all possible terms containing as factors exactly one element from each row and each column. If we were to attempt to do this with the determinant above, we would find that the various terms would contain different powers of λ. If we were to collect the terms having like powers, we would obtain an expression of the form

$$(-1)^n[\lambda^n - p_1\lambda^{n-1} - p_2\lambda^{n-2} - \cdots - p_n] \tag{10-12}$$

where the constants $p_1, p_2, \ldots, p_n$ are numbers resulting from some very complicated manipulations of the numbers a_{ij} in the determinant.

From Chapter 9, there are exactly n values of λ (not necessarily distinct) which will make (10-12) be equal to zero. These values are called the "eigenvalues" (or "characteristic roots," or "latent roots," or "proper values") of the matrix A. For any eigenvalue λ_i, the vector x which satisfies equation (10-10) is called the "eigenvector" (or "characteristic vector," or "latent vector," or "proper vector") corresponding to λ_i. The polynomial (10-12) is called the "characteristic polynomial" of the matrix A, and the equation

$$\lambda^n - p_1\lambda^{n-1} - p_2\lambda^{n-2} - \cdots - p_n = 0 \tag{10-13}$$

is called the "characteristic equation."

Example 1. Find the eigenvalues and eigenvectors for the matrix

$$\begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix}$$

To find the eigenvalues, we set

$$\begin{vmatrix} 1 - \lambda & 3 \\ 2 & 2 - \lambda \end{vmatrix} = 0$$

Expanding, we obtain the characteristic equation

$$(1 - \lambda)(2 - \lambda) - 6 = \lambda^2 - 3\lambda - 4 = 0$$

This factors into

$$(\lambda - 4)(\lambda + 1) = 0$$

so the eigenvalues are

$$\lambda_1 = 4, \qquad \lambda_2 = -1$$

To find the eigenvector corresponding to λ_1, we set

$$\begin{pmatrix} 1 - \lambda_1 & 3 \\ 2 & 2 - \lambda_1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 0$$

or

$$\begin{pmatrix} -3 & 3 \\ 2 & -2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 0$$

Since there are two unknowns and the coefficient matrix has rank 1, Theorem 3 of Section 10.63 tells us that these equations have one linearly independent vector solution U_1, and that all other solutions are multiples of this one. We see by inspection that the vector

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

is a solution, and hence is an eigenvector corresponding to λ_1. All other solutions are of the form

$$c_1\begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

where c_1 is an arbitrary constant. Hence the eigenvector is really determined only up to an arbitrary constant multiple.

To find the eigenvector corresponding to λ_2, we set

$$\begin{pmatrix} 1-\lambda_2 & 3 \\ 2 & 2-\lambda_2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 0$$

or

$$\begin{pmatrix} 2 & 3 \\ 2 & 3 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 0$$

Again there is one linearly independent solution vector. We see by inspection that

$$\begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

is a solution. All solutions are of the form

$$c_2\begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

where c_2 is an arbitrary constant.

Hence the eigenvalues are

$$4, \quad -1$$

and the corresponding eigenvectors are

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

(We ordinarily ignore the arbitrary constant multiple when writing an eigenvector.)

Example 2. Find the eigenvalues and eigenvectors for the matrix

$$\begin{pmatrix} 3 & 2 & 4 \\ 1 & 4 & 4 \\ -1 & -2 & -2 \end{pmatrix}$$

To determine the eigenvalues, we set

$$\begin{vmatrix} 3-\lambda & 2 & 4 \\ 1 & 4-\lambda & 4 \\ -1 & -2 & -2-\lambda \end{vmatrix} = 0$$

Expanding, we obtain the characteristic equation

$$-\lambda^3 + 5\lambda^2 - 8\lambda + 4 = 0$$

which has the roots

$$\lambda_1 = 1, \qquad \lambda_2 = \lambda_3 = 2$$

To find the eigenvector corresponding to λ_1, we set

$$\begin{pmatrix} 3-\lambda_1 & 2 & 4 \\ 1 & 4-\lambda_1 & 4 \\ -1 & -2 & -2-\lambda_1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 0$$

or

$$\begin{pmatrix} 2 & 2 & 4 \\ 1 & 3 & 4 \\ -1 & -2 & -3 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 0$$

The coefficient matrix has rank 2, so this system has one [illegible] dent vector solution. If we solve by the method of Sect[illegible] that the eigenvector is

$$\begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$

To find the eigenvector corresponding to λ_2, we set

$$\begin{pmatrix} 3-\lambda_2 & 2 & 4 \\ 1 & 4-\lambda_2 & 4 \\ -1 & -2 & -2-\lambda_2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 0$$

or

$$\begin{pmatrix} 1 & 2 & 4 \\ 1 & 2 & 4 \\ -1 & -2 & -4 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 0$$

The coefficient matrix has rank 1, so this system has two [illegible] dent vector solutions. Solving by the method of Section 10[illegible] linearly independent eigenvectors,

$$\begin{pmatrix} -4 \\ 0 \\ 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix}$$

The root λ_3, being the same as λ_2, has the same eigenvectors. Hence we have a single root, 1, with its eigenvector

$$\begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$

and a double root, 2, with two eigenvectors:

$$\begin{pmatrix} -4 \\ 0 \\ 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix}$$

Example 3. Find the eigenvalues and eigenvectors for the matrix

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -8 & -12 & -6 \end{pmatrix}$$

We set

$$\begin{vmatrix} -\lambda & 1 & 0 \\ 0 & -\lambda & 1 \\ -8 & -12 & -6-\lambda \end{vmatrix} = 0$$

and obtain the characteristic equation

$$-\lambda^3 - 6\lambda^2 - 12\lambda - 8 = 0$$

which has the roots

$$\lambda_1 = -2, \qquad \lambda_2 = -2, \qquad \lambda_3 = -2$$

To find the eigenvectors, we set

$$\begin{pmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ -8 & -12 & -4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 0$$

Since the coefficient matrix has rank 2, there is only one linearly independent eigenvector. It turns out to be

$$\begin{pmatrix} 1 \\ -2 \\ 4 \end{pmatrix}$$

Since all roots are the same, we can obtain no more eigenvectors. Hence in this case we have a triple eigenvalue, -2, and only one eigenvector (which might be considered an eigenvector of multiplicity 3):

$$\begin{pmatrix} 1 \\ -2 \\ 4 \end{pmatrix}$$

The above examples have illustrated all the possibilities concerning real eigenvalues and their corresponding eigenvectors. These possibilities can be summarized in the following theorem.

Theorem 4. *An nth-order square matrix has n eigenvalues. If these are discrete, there is one eigenvector corresponding to each eigenvalue. If an eigenvalue is of multiplicity r, it may have from one to r linearly independent eigenvectors associated with it.*

10.71 Program for Largest Eigenvalue and Eigenvector

Suppose that the matrix A has one eigenvalue λ_1 which is larger than all others in absolute value, and y is any nonzero column vector conformable with A. Let the vectors y_1, y_2, etc., be defined by

$$\begin{aligned} y_1 &= Ay \\ y_2 &= Ay_1 \\ &\vdots \\ y_n &= Ay_{n-1} \end{aligned} \tag{10-14}$$

The vectors y_i defined in this manner can lead to the value of λ_1 and to x_1, the eigenvalue corresponding to λ_1. The method of obtaining the eigenvalue and eigenvector will be illustrated without proof.* In order to program the illustration, let us first write a remote-terminal program to perform the computations indicated by expression (10-14). A suitable program is

* See, for example, J. G. Herriot, *Methods of Mathematical Analysis and Computation*, John Wiley & Sons, Inc., New York, 1963.

```
 1    DIMENSION A(10,10),Y(10),YN(10)
 2    PRINT, "INPUT N, TEN OR LESS"
 3    INPUT, N
 4    PRINT, "INPUT A(1,1)A(1,2),,,A(N,N)"
 5    INPUT, ((A(I,J),J=1,N),I=1,N)
 6    PRINT, "INPUT Y(1),Y(2),,,Y(N)"
 7    INPUT, (Y(I),I=1,N)
 8  1 DO 2 I=1,N
 9    YN(I)=0.
10    DO 2 J=1,N
11  2 YN(I)=YN(I)+A(I,J)*Y(J)
12    PRINT, (YN(I),I=1,N)
13    INPUT, Q
14    DO 3 I=1,N
15  3 Y(I)=YN(I)
16    GO TO 1
17    END
```

In this program, the statements at lines 2 through 7 allow the user to input an initial matrix A and vector y of order up to 10. The statements at lines 8 through 12 compute and print the vector $y_1 = Ay$. At line 13, the user is allowed to specify whether another step of the process is required. If the typed entry is the letter S, the program will terminate. If the entry is any number whatsoever, the program will cause y_1 to replace y, and will repeat lines 8 through 12, thereby computing and printing $y_2 = Ay_1$, and so on.

Example 1. Write all user inputs and machine responses for running the above program with the matrix

$$\begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix} \quad \text{and the vector} \quad \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

continuing until vectors through y_n have been generated.

The inputs and responses are

```
 RUN
 INPUT N, TEN OR LESS
? 2
 INPUT A(1,1),A(1,2),,,A(N,N)
? 1,3,2,2
 INPUT Y(1),Y(2),,,Y(N)
? 1,0
        1.000000      2.000000
? 0
        7.000000      6.000000
```



```
? 0
        25.00000      25.00000
? 0
        103.0000      102.0000
? 0
        409.0000      410.0000
? 0
        1639.000      1638.000
? 0
        6553.000      6554.000
? 0
        .2621500E5    .2621400E5
? S
STOP
```

The above program was Example 1, Section 10.6, which had a largest eigenvalue of 4 and corresponding eigenvector of $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Looking at the vectors printed out above, we see that the vectors generated were

$$\begin{pmatrix} 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 7 \\ 6 \end{pmatrix}, \begin{pmatrix} 25 \\ 26 \end{pmatrix}, \begin{pmatrix} 103 \\ 102 \end{pmatrix}, \begin{pmatrix} 409 \\ 410 \end{pmatrix}, \begin{pmatrix} 1639 \\ 1638 \end{pmatrix}, \begin{pmatrix} 6553 \\ 6554 \end{pmatrix}, \begin{pmatrix} 26215 \\ 26214 \end{pmatrix}$$

and that after the first few steps the components are always very nearly equal, that is, the vectors themselves are very nearly simple multiples of the vector $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Since eigenvectors are determined only up to a constant multiple, we can say that the vector y_n of (10-14) is actually approaching the eigenvector x. Now since

$$Ax = \lambda x$$

then if y_n is x, then y_{n+1} will be λx. We note without surprise, then, that in each of the vectors computed in the above example, the components are very nearly four times those of the preceding vector.

It appears, then, that the above program can be used almost directly to find the largest eigenvalue and corresponding eigenvector. Some improvement can be made by replacing the statement at line 15 by

```
15  3 Y(I)=YN(I)/YN(1)
```

This will serve to keep the components from growing at each stage, and further will cause the first component to approach the actual value of the eigenvalue. If this had been done for Example 1, the printouts would have been

```
   1.000000  2.000000
? 0
   7.000000  6.000000
? 0
   3.571429  3.714286
? 0
   4.120000  4.080000
? 0
   3.970874  3.980583
? 0
   4.007335  4.004890
? 0
   3.998109  3.998595
? 0
   4.000453  4.000305
? S
STOP
```

From these results, the eigenvalue 4 and the eigenvector $\begin{pmatrix}1\\1\end{pmatrix}$ are apparent.

10.72 Complex Eigenvalues

From Chapter 9 it is known that the characteristic equation may have complex roots, occuring in conjugate pairs. In this case, the eigenvectors are also complex, and the equation

$$(A - \lambda I)x = 0$$

instead of being n equations in n unknowns, is really $2n$ equations in $2n$ unknowns, for both the real and imaginary part of x must satisfy the equation. Let

$$\lambda = \alpha + \beta i$$

be a complex eigenvalue, and let the eigenvector be

$$x = \begin{pmatrix} x_1 + y_1 i \\ x_2 + y_2 i \\ \vdots \\ x_n + y_n i \end{pmatrix}$$

If we substitute these in the above equation and separate real and imaginary parts, the result can be written in the form

$$\begin{pmatrix} a_{11}-\alpha & a_{12} & \cdots & a_{1n} & \beta & 0 & \cdots & 0 \\ a_{21} & a_{22}-\alpha & \cdots & a_{2n} & 0 & \beta & \cdots & 0 \\ \vdots & & & & & & & \\ a_{n1} & a_{n2} & \cdots & a_{nn}-\alpha & 0 & 0 & \cdots & \beta \\ -\beta & 0 & \cdots & 0 & a_{11}-\alpha & a_{12} & \cdots & a_{1n} \\ 0 & -\beta & \cdots & 0 & a_{21} & a_{22}-\alpha & \cdots & a_{2n} \\ \vdots & & & & & & & \\ 0 & 0 & \cdots & -\beta & a_{n1} & a_{n2} & \cdots & a_{nn}-\alpha \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = 0$$

These are $2n$ real equations in $2n$ real unknowns which can be solved for the x_i's and y_i's. The eigenvector corresponding to the complex conjugate of λ is the complex conjugate of the eigenvector for λ, so the process needs to be done only once for each pair of complex roots.

Example 1. Find the eigenvalues and eigenvectors of

$$\begin{pmatrix} -1 & -5 \\ 1 & 3 \end{pmatrix}$$

We set

$$\begin{vmatrix} -1-\lambda & -5 \\ 1 & 3-\lambda \end{vmatrix} = 0$$

and obtain the characteristic equation:

$$\lambda^2 - 2\lambda + 2 = 0$$

which has roots:

$$\lambda_1 = 1 + i, \qquad \lambda_2 = 1 - i$$

To find the eigenvector corresponding to λ_1, using the method described above, we write

$$\begin{pmatrix} -2 & -5 & 1 & 0 \\ 1 & 2 & 0 & 1 \\ -1 & 0 & -2 & -5 \\ 0 & -1 & 1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{pmatrix} = 0$$

Applying the method of Section 13.34, this reduces to

$$\begin{pmatrix} 1 & 0 & -.2 & .4 \\ 0 & 1 & .4 & .2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ y_1 \\ x_1 \end{pmatrix} = 0$$

This has two linearly independent solution vectors

$$\begin{pmatrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 0 \\ .2 \\ 1 \\ -.4 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 1 \\ -.4 \\ 0 \\ -.2 \end{pmatrix}$$

These vectors of themselves are not of interest to us, except to use the numbers in them to construct the complex of vectors

$$\begin{pmatrix} x_1 + y_1 i \\ x_2 + y_2 i \end{pmatrix} = \begin{pmatrix} i \\ .2 - .4i \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} x_1 + y_1 i \\ x_2 + y_2 i \end{pmatrix} = \begin{pmatrix} 1 \\ -.4 - .2i \end{pmatrix}$$

These two vectors are not linearly independent at all, for if we multiply the second by i, we obtain the first. Hence we have really obtained only one eigenvector corresponding to the eigenvalue $1 + i$, and that is

$$\begin{pmatrix} i \\ .2 - .4i \end{pmatrix}$$

The eigenvector corresponding to $1 - i$ is the conjugate of this:

$$\begin{pmatrix} -i \\ .2 + .4i \end{pmatrix}$$

10.73 Determination of All Eigenvalues and Eigenvectors

The method described in Section 10.71 will provide the largest eigenvalue and corresponding eigenvector. Frequently it is necessary to find all eigenvalues and eigenvectors. From the discussions of Section 10.7, it is clear that this can be done by accomplishing the following three steps:

(1) Find the characteristic polynomial.
(2) Solve the characteristic equation for its roots.
(3) Solve sets of linear equations for the eigenvectors.

Chapter 9 gave methods for solving polynomial equations, so we already have computer methods for step (2). Section 10.64 gave a computer method for solving systems of linear equations which is satisfactory for step (3). Hence the only thing really required is a computer method for generating the characteristic polynomial. In the examples above, we have used very small matrices and found the characteristic polynomial by brute-force expansion of the determinant, but this process is the worst for large-order matrices. A more efficient method is the Leverrier-Faddeev method, which proceeds as follows:

$$\begin{array}{ll} \text{Let } A_1 = A & \text{and } p_1 = \operatorname{tr} A \\ \text{Let } A_2 = A(A_1 - p_1 I), & \text{and } p_2 = (1/2) \operatorname{tr} A_2 \\ \text{Let } A_3 = A(A_2 - p_2 I), & \text{and } p_3 = (1/3) \operatorname{tr} A_3 \\ \quad \vdots & \\ A_n = A(A_{n-1} - p_{n-1} I) & \text{and } p_n = (1/n) \operatorname{tr} A_n \end{array}$$

The numbers $p_1, p_2, \ldots, p_n$ are the required coefficients in the characteristic equation

$$\lambda^n - p_1 \lambda^{n-1} - p_2 \lambda^{n-2} - \cdots - p_n = 0$$

In addition, as a bonus side product of this process, it can be shown that the inverse of A is given by

$$A^{-1} = (1/p_n)(A_{n-1} - p_{n-1} I) \tag{10-15}$$

and also, as a sometimes helpful check,

$$A_n - p_n I = 0 \tag{10-16}$$

Example 1. Find the characteristic equation of

$$\begin{pmatrix} 1 & 3 & 2 \\ -2 & 1 & . \\ 1 & -2 & -1 \end{pmatrix}$$

Using the above procedure, we have

$$A_1 = \begin{pmatrix} 1 & 3 & 2 \\ -2 & 1 & 1 \\ 1 & -2 & -1 \end{pmatrix}, \qquad p_1 = 1 + 1 - 1 = 1$$

$$A_2 = \begin{pmatrix} 1 & 3 & 2 \\ -2 & 1 & 1 \\ 1 & -2 & -1 \end{pmatrix}\begin{pmatrix} 0 & 3 & 2 \\ -2 & 0 & 1 \\ 1 & -2 & -2 \end{pmatrix} = \begin{pmatrix} -4 & -1 & 1 \\ -1 & -8 & -5 \\ 3 & 5 & 2 \end{pmatrix}$$

$$p_2 = (1/2)(-4 - 8 + 2) = -5$$

$$A_3 = \begin{pmatrix} 1 & 3 & 2 \\ -2 & 1 & 1 \\ 1 & -2 & -1 \end{pmatrix}\begin{pmatrix} 1 & -1 & 1 \\ -1 & -3 & -5 \\ 3 & 5 & 7 \end{pmatrix} = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix}$$

$$p_3 = (1/3)(4 + 4 + 4) = 4$$

As a check, we see that

$$A_3 - p_3 I = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} - \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} = 0$$

Hence the characteristic equation is

$$\lambda^3 - \lambda^2 + 5\lambda - 4 = 0$$

The flow chart, Figure 10-10, describes this process for an nth-order matrix. According to equation (10-16), the matrix A_n is simply the identity matrix multiplied by p_n, so only the first element of A_n need be calculated to give p_n. The value of p_n is, in fact, the determinant of A, so that if p_n is zero, the matrix is singular. If p_n is not zero, the inverse of A is easily calculated from equation (10-15), and the flow chart includes this calculation. The elements of A^{-1} are the last values obtained for f_{ij}.

The FORTRAN subroutine given below will generate coefficients in accordance with the flow chart, for matrices up to order 20. However, in order that the subscripts will match the notation in the subroutines of Chapter 9, the characteristic equation is written as

$$Q(1)\lambda^N + Q(2)\lambda^{N-1} + \cdots + Q(N+1) = 0$$

The relationship between the p_k of the flow chart and the Q(K) of the subroutine is given by

$$Q(1) = 1, \; Q(K+1) = -p_k \qquad \text{for } k = 1, \ldots, n$$

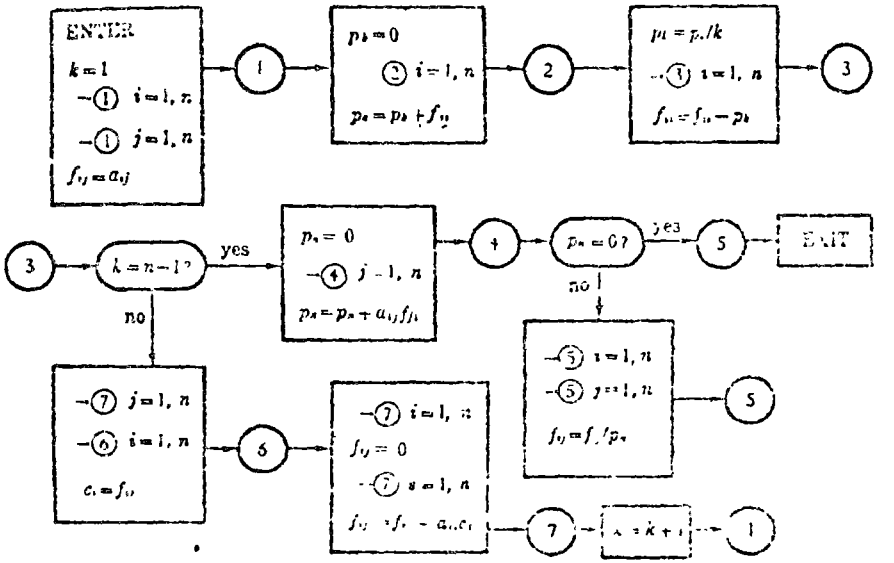


Figure 10-10: Generation of characteristic polynomial.

As in the flow chart, the subscripted variable F(I,J) is the inverse matrix unless Q(N+1) happens to be zero.

```
      SUBROUTINE CHARLQ(A,N,Q,F)
      DIMENSION A(20,20),F(20,20),Q(21),C(20)
      Q(1)=1.
      K=1
      DO 11 I=1,N
      DO 11 J=1,N
   11 F(I,J)=A(I,J)
    1 CONTINUE
      Q(K+1)=0.
      DO 2 I=1,N
    2 Q(K+1)=Q(K+1)+F(I,I)
      CONTINUE
      FK=K
      Q(K+1)=-Q(K+1)/FK
      DO 3 I=1,N
      F(I,I)=F(I,I)+Q(K+1)
    3 CONTINUE
      IF(K-N+1)71,41,71
   71 DO 7 J=1,N
      DO 6 I=1,N
      C(I)=F(I,J)
```

```
    6 CONTINUE
      DO 7 I=1,N
      F(I,J)=0.
      DO 7 IS=1,N
      F(I,J)=F(I,J)+A(I,IS)*C(IS)
    7 CONTINUE
      K=K+1
      GO TO 1
   41 Q(N+1)=0.
      DO 4 J=1,N
      Q(N+1)=Q(N+1)-A(1,J)*F(J,1)
    4 CONTINUE
      IF(Q(N+1))51,5,51
   51 DO 52 I=1,N
      DO 52 J=1,N
   52 F(I,J)=-F(I,J)/Q(N+1)
    5 RETURN
      END
```

With the above subroutine and subroutines of Chapter 9 and Section 10.6, eigenvalues and eigenvectors can be found in a systematic way. There are also methods which, under some conditions, can be used to find all eigenvalues directly from the matrix itself, without generating the characteristic equation first. These methods are available in the literature and will not be reported here.

EXERCISE 23

1. Determine the rank of the following matrices

   a. $\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ b. $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 6 \end{pmatrix}$

   c. $\begin{pmatrix} 1 & 2 & 3 \\ -2 & 1 & -2 \\ -1 & 3 & 1 \end{pmatrix}$ d. $\begin{pmatrix} 2 & -1 & 3 & 4 \\ 1 & -2 & -2 & -1 \\ 0 & 3 & 7 & 6 \end{pmatrix}$

2. Determine whether the following systems are consistent or inconsistent.

   a. $x+2y+z=4$, $-2x-4y-2z=3$ b. $x+2y=6$, $x+3y=8$

   c. $x+3y=7$, $2x-y=4$, $4x+5y=18$ d. $x+2y=8$, $3x-y=2$, $2x+y=6$

3. Solve completely the following systems of equations.

   a. $x+2y=0$, $-2x-4y=0$ b. $x+y-z=2$, $x-y+z=3$

   c. $x+3y-z=4$, $2x-y+2z=3$, $3x+2y+z=7$ d. $x+2y+z=1$, $2x-y+z=2$, $3x-y+4z=3$

*4. Find all eigenvalues and eigenvectors for the following matrices.

   a. $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ b. $\begin{pmatrix} 1 & 3 \\ -2 & -4 \end{pmatrix}$ c. $\begin{pmatrix} 1 & 2 \\ -2 & -3 \end{pmatrix}$

   d. $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix}$ e. $\begin{pmatrix} 2 & 1 & 2 \\ -1 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$

5. Find the number of multiplications required to find the rank of a 10 by 15 matrix using SUBROUTINE MARANK of Section 10.61, if the rank turns out to be 3.

6. Write a program that will input a system of linear equations up to 20 by 20, call subroutine LINEQ of Section 10.64 to obtain all solutions, and print the result.

7. Write a program that will input a square matrix up to 19 by 19, call subroutine CHARLQ of Section 10.73 to obtain the characteristic equation, call the appropriate subroutine from Chapter 9 to find the largest real root, call subroutine LINEQ of Section 10.64 to find the corresponding eigenvector, and print the result.

I N D I C E

the problems of the third chapter are characterized by sets of simultaneous ordinary differential equations with prescribed *initial* conditions, the problems of the fourth and fifth chapters are characterized by ordinary or partial differential equations with closed *boundary* conditions, and the problems of the sixth chapter are characterized by partial differential equations with open boundary conditions. This survey of numerical procedures thus amounts to a catalogue of practical methods for the solution of algebraic, ordinary differential, and partial differential equations. In Chaps. 1, 3, 4, and 6 both linear and nonlinear problems are considered. The discussion of eigenvalue problems in Chaps. 2 and 5 is limited to linear systems.

All six chapters have the same structure. At the beginning of each chapter several representative problems are presented. These serve to identify the class of problems under consideration. The process of formulating a mathematical model is illustrated for each of these problems.

Before leaving these formulations they are each cast into *dimensionless* form. This is an extremely useful organizational tool[1] of the analyst. In connection with numerical calculations it removes all unnecessary symbols, leaving the basic problem in its simplest form.

Then, before surveying numerical procedures applicable to this class of problems, a brief résumé of the classical mathematical theory is given. A complete mathematical development has not been attempted but an effort has been made to describe clearly the properties of the well-behaved or regular system. The possibilities of irregular behavior are hinted at by means of simple counterexamples. Enough theory is presented to provide a background for the explanation of the success (and limitations) of the numerical procedures which follow.

After these preliminaries the actual survey of numerical procedures begins. Illustrative examples are drawn from the problems formulated at the beginning of the chapter. At the end of each section there is a set of exercises for the reader. A few of these are of the nature of drill problems but the majority represent interesting extensions or alternative developments of the text material. Answers or hints for the solution are given in most cases.

The numerical procedures described here are those which in the judgment of the author are of most potential interest to the engineering analyst. Methods for both hand and machine computation are given.

Throughout the text there are references to books and papers having direct bearing on the matter at hand. For the reader's convenience a number of selected general references are grouped together in the Bibliography at the end of the book.

[1] See, for example, H. L. Langhaar, "Dimensional Analysis and Theory of Models," John Wiley & Sons, Inc., New York, 1951.

CHAPTER 1

EQUILIBRIUM PROBLEMS IN SYSTEMS WITH A FINITE NUMBER OF DEGREES OF FREEDOM

The state of a physical system can often be described with adequate precision by giving the magnitudes of a finite number of state variables. This chapter deals with numerical procedures for determining steady states of such systems. The chapter begins with a preliminary examination of several particular problems. The general problem of this type is then formulated mathematically as a set of simultaneous algebraic equations. There is a review of the classical results from the theory of such systems, including a discussion of the relationship of extremum principles to equilibrium problems. Numerical procedures, both exact and approximate, are then described and illustrated by applying them to the particular problems set up at the beginning of the chapter.

1-1. Particular Examples

We begin with an assortment of examples of how mathematical formulations are set up for particular physical problems. The examples are taken from a variety of fields and, in general, have been chosen for their simplicity despite the fact that the really significant contributions of numerical procedures occur in problems of extended complexity.

It is generally recognized that the most difficult step in the whole process of engineering analysis is that in which a mathematical model is substituted for a real physical system. It is here that judgment, experience, and ingenuity of the highest order are required of the analyst. It is here that the really gross approximations and simplifications are made. In this text the basic structure of the various physical problem types and the corresponding mathematical models is emphasized.

The general equilibrium problem in a lumped-parameter system has the following structure: The given system is made up by interconnecting a number of simple elements. The equilibrium or steady-state requirements for each individual element are known. As examples we have the sine-tangent law for elastic elements, Ohm's law for electrical resistances, and the pressure-flow relation for hydraulic resistance. In addition to satisfying the equilibrium requirements of the individual elements it is

also necessary to satisfy certain interconnection requirements. Thus in elastic systems we must have geometric fit and balance of forces at all joints; in electric networks we must satisfy both of Kirchhoff's laws; and in hydraulic networks we must have conservation of flow and uniqueness of pressure at every interconnection. The over-all equilibrium problem then consists in finding the state of a system which simultaneously satisfies the equilibrium requirements of the individual elements together with the interconnection requirements.

To serve as concrete illustrations of this general statement and to provide illustrative examples for the numerical procedures which follow, we here consider the following five particular equilibrium problems:

1-1. Elastic spring system.
1-2. D-c network.
1-3. A-c network.
1-4. Continuous beam.
1-5. Hydraulic network.

In each case the problem is cast into nondimensional form, with particular data assumed, in preparation for numerical solution. In most cases complementary forms of the problem are considered. The first four problems are linear, while the fifth represents an example of a practically important nonlinear problem.

Problem 1-1. Elastic Spring System. In Fig. 1-1 a system of four

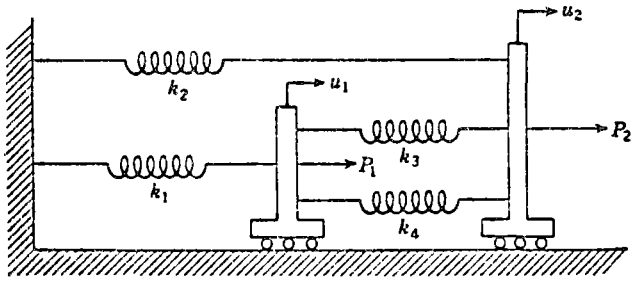


FIG. 1-1 Elastic system of interconnected springs subjected to loads P_1 and P_2.

linear springs is shown. Assume that when P_1 and P_2 are zero then u_1 and u_2 are both zero and that all springs are in their natural positions. The problem here is to find the displacements u_1 and u_2 and the forces f_1, f_2, f_3, and f_4 in the four springs when the loads P_1 and P_2 are applied. The fundamental requirements are:

1. Spring force = k(spring elongation) for each spring.
2. Forces should balance on each movable cart.
3. Spring elongations should be compatible with the displacements of the carts.

A standard method of solution is to choose unknown variables in such a way that requirement 3 above is automatically satisfied. In our problem this is done by taking u_1 and u_2 as unknowns and expressing the spring elongations in terms of them (e.g., elongation of spring 4 is $u_2 - u_1$). Next the spring forces are expressed in terms of u_1 and u_2 by introducing the spring constants. Finally, writing the force-balance conditions for each cart gives us the following equations for u_1 and u_2:

$$\begin{aligned} k_1u_1 - k_3(u_2 - u_1) - k_4(u_2 - u_1) &= P_1 \\ k_2u_2 + k_3(u_2 - u_1) + k_4(u_2 - u_1) &= P_2 \end{aligned} \tag{1-1}$$

A complete solution of our problem would require the solution of these simultaneous equations. We stop at this point, however, since we are here concerned only with the formulation of the problem. Summarizing, we limited ourselves to geometrically compatible states as soon as we took u_1 and u_2 as unknowns; requiring that force balance should also hold gave us (1-1).

A *complementary* method of solution for the same problem is to choose unknown variables in such a way that requirement 2 above is automatically satisfied. This may be done by taking the spring forces f_2 and f_3 as unknown and expressing the other spring forces, f_1 and f_4, in terms of them by means of the force-balance conditions.

$$\begin{aligned} f_4 &= P_2 - f_2 - f_3 \\ f_1 &= P_1 + f_3 + f_4 = P_1 + P_2 - f_2 \end{aligned} \tag{1-2}$$

Next the spring elongations are expressed in terms of f_2 and f_3 by introducing the spring constants. Finally we obtain the following equations for f_2 and f_3 by requiring that the spring elongations be compatible with unique displacements of the carts:

$$\begin{aligned} \frac{f_2}{k_2} &= \frac{P_1 + P_2 - f_2}{k_1} + \frac{P_2 - f_2 - f_3}{k_4} \\ \frac{f_3}{k_3} &= \frac{P_2 - f_2 - f_3}{k_4} \end{aligned} \tag{1-3}$$

The second of these expresses the fact that the elongations of springs 3 and 4 should be the same. The first expresses the fact that the elongation of spring 2 must be the same as the sum of the elongations of springs 1 and 4. Again a complete solution would require the simultaneous solution of (1-3), but we stop at this point. Reiterating our logic, we limited ourselves to self-balancing states when we took f_2 and f_3 as unknowns and used (1-2) for the other forces. Among these self-balancing states the true state is selected by (1-3), which requires that the spring elongations should be compatible with the given interconnections of the system.

For future use we now specialize the above problem to the case where

$$\begin{aligned} k_1 &= 3k \\ k_2 &= 2k \\ k_3 &= k \\ k_4 &= k \end{aligned} \qquad \begin{aligned} P_1 &= P \\ P_2 &= 2P \end{aligned} \tag{1-4}$$

Substituting these values in (1-1) and (1-3), we obtain

$$\begin{aligned} 5ku_1 - 2ku_2 &= P \\ -2ku_1 + 4ku_2 &= 2P \end{aligned} \tag{1-5}$$

as the equations for the displacements and

$$\begin{aligned} \tfrac{11}{6}f_2 + f_3 &= 3P \\ f_2 + 2f_3 &= 2P \end{aligned} \tag{1-6}$$

as the complementary equations for the forces. These formulations can be simplified even further by introducing dimensionless variables. If we define the nondimensional displacements

$$x_1 = \frac{u_1}{P/k} \qquad x_2 = \frac{u_2}{P/k} \tag{1-7}$$

the displacement equations (1-5) can be written in the following form:

$$\begin{aligned} 5x_1 - 2x_2 &= 1 \\ -2x_1 + 4x_2 &= 2 \end{aligned} \tag{1-8}$$

Similarly, in terms of the nondimensional forces

$$y_1 = \frac{f_2}{P} \qquad y_2 = \frac{f_3}{P} \tag{1-9}$$

the force equations (1-6) become

$$\begin{aligned} \tfrac{11}{6}y_1 + y_2 &= 3 \\ y_1 + 2y_2 &= 2 \end{aligned} \tag{1-10}$$

Problem 1-2. D-C Network. We consider the problem of determining the voltages and currents in the network shown in Fig. 1-2. The resistances and battery emfs are given in the figure in terms of R and E. The equilibrium or steady-state conditions are Ohm's law for each individual resistor plus the interconnection requirements which are the two laws of Kirchhoff.[1] We can obtain *complementary* formulations of the problem in the following manner: If we represent the state of the system by a set of independent currents such that Kirchhoff's first law is automatically satisfied, we then obtain equations for determining these currents by requiring that the second law be satisfied. Alternatively if the state of the system is represented by a set of independent voltages such that Kirchhoff's second law is automatically satisfied, equations can then be obtained for determining these voltages by requiring the satisfaction of the first law.

[1] See, for example, C. L. Dawes, "Electrical Engineering," 3d ed., vol. I, McGraw-Hill Book Company, Inc., New York, 1937, p. 72.

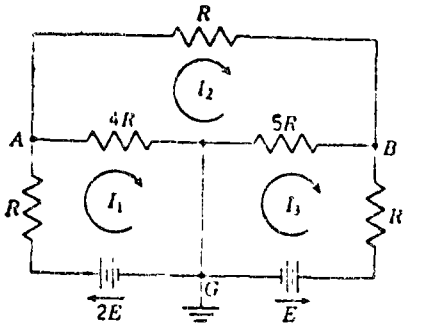


FIG. 1-2. Network of resistors and batteries

In accordance with the first procedure the state of the system is represented by the three loop currents I_1, I_2, and I_3. The net current flow into any junction is always zero for any values of I_1, I_2, and I_3. Ohm's law together with the requirement that the net voltage drop in any closed loop should vanish yields the following equations:

$$\begin{aligned} 2E - RI_1 - 4R(I_1 - I_2) &= 0 \\ -RI_2 - 5R(I_2 - I_3) - 4R(I_2 - I_1) &= 0 \\ -RI_3 - 5R(I_3 - I_2) - E &= 0 \end{aligned} \tag{1-11}$$

When the currents which satisfy (1-11) are found, any desired network emf is easily obtained by an elementary application of Ohm's law.

Following the second procedure, the state of the system can be represented by the potentials e_1 and e_2 of the nodes A and B with respect to G. This ensures that the voltage drop around any closed loop vanishes. The requirement that there should be no net current flow into the nodes A and B results in the following equations:

$$\begin{aligned} \frac{2E - e_1}{R} - \frac{e_1}{4R} + \frac{e_2 - e_1}{R} &= 0 \\ \frac{E - e_2}{R} - \frac{e_2}{5R} + \frac{e_1 - e_2}{R} &= 0 \end{aligned} \tag{1-12}$$

When the voltages e_1 and e_2 which satisfy (1-12) have been found, any desired network current may be obtained by a simple application of Ohm's law.

The complete solution can thus be obtained by solving either (1-11) or (1-12). Note that here the number of degrees of freedom is not the same for the two analyses. Before leaving this problem, we cast the equations into nondimensional form. Dimensionless current and volt-

ages are defined as follows:

$$x_1 = \frac{I_1}{E/R} \qquad x_2 = \frac{I_2}{E/R} \qquad x_3 = \frac{I_3}{E/R} \qquad y_1 = \frac{e_1}{E} \qquad y_2 = \frac{e_2}{E} \tag{1-13}$$

The current equations (1-11) then become

$$\begin{aligned} 5x_1 - 4x_2 &= 2 \\ -4x_1 + 10x_2 - 5x_3 &= 0 \\ -5x_2 + 6x_3 &= -1 \end{aligned} \tag{1-14}$$

while the voltage equations (1-12) take the following form:

$$\begin{aligned} 2.25y_1 - y_2 &= 2 \\ -y_1 + 2.20y_2 &= 1 \end{aligned} \tag{1-15}$$

These last two sets of equations constitute complementary dimensionless formulations of Prob. 1-2.

Problem 1-3. A-C Network The equilibrium problem here is to determine the steady-state currents in the network of Fig. 1-3. The impedances of the branches at the frequency of the voltage source are indicated in the usual[1] complex notation in terms of R. Complementary formulations of this problem can be obtained in the same manner as in Prob. 1-2. We consider here only the equations for the currents. If we take I_1 and I_2 as the state variables, Kirchhoff's first law is automatically satisfied and the second law yields the following equations:

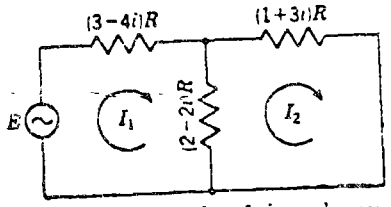


FIG. 1-3. Network of impedances connected to alternating-voltage source.

$$\begin{aligned} E - (3 - 4i)RI_1 - (2 - 2i)R(I_1 - I_2) &= 0 \\ -(2 - 2i)R(I_2 - I_1) - (1 + 3i)RI_2 &= 0 \end{aligned} \tag{1-16}$$

A nondimensional formulation is obtained by introducing the dimensionless variables

$$I_1' = \frac{I_1}{E/R} \qquad I_2' = \frac{I_2}{E/R} \tag{1-17}$$

into (1-16) as follows:

$$\begin{aligned} (5 - 6i)I_1' - (2 - 2i)I_2' &= 1 \\ -(2 - 2i)I_1' + (3 + i)I_2' &= 0 \end{aligned} \tag{1-18}$$

The quantities I_1' and I_2' are expected to be complex. Although procedures exist for the direct solution of sets of equations such as (1-18), it is sometimes useful to transform the complex equations into their real equivalents. To illustrate this process for the present example, we define the real quantities $x_1, \ldots, x_4$ as follows:

[1] See, for example, C. I. Dawes, "A Course in Electrical Engineering," 4th ed., vol. II, McGraw-Hill Book Company, Inc., New York, 1947, p. 70. The symbol i stands for the imaginary unit $\sqrt{-1}$.

$$\begin{aligned} I_1' &= x_1 + ix_2 \\ I_2' &= x_3 + ix_4 \end{aligned} \tag{1-19}$$

When these are substituted in (1-18), each equation can be separated into two: one obtained from the real terms and one from the imaginary terms. We thus obtain the following four real equations, which are equivalent to the two complex equations of (1-18):

$$\begin{aligned} 5x_1 + 6x_2 - 2x_3 - 2x_4 &= 1 \\ 6x_1 - 5x_2 - 2x_3 + 2x_4 &= 0 \\ -2x_1 - 2x_2 + 3x_3 - x_4 &= 0 \\ -2x_1 + 2x_2 - x_3 - 3x_4 &= 0 \end{aligned} \tag{1-20}$$

Problem 1-4. Continuous Beam. In Fig. 1-4 a uniform elastic beam is shown. It is simply supported at A, B, and C and clamped at D. Equilibrium problems for such systems consist in determining the bending moments and deflections resulting from assigned loads. We consider the particular problem of Fig. 1-4, where the load is the single moment M applied at A. The flexural stiffness of the beam is EI, and the span lengths are given in terms of a.

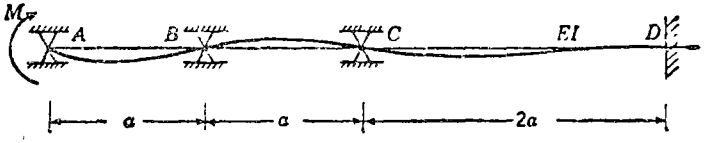


FIG. 1-4. Continuous beam freely supported at A, B, and C, clamped at D, and subjected to external moment M applied at A.

This system may be treated as a lumped parameter system by considering each span as a single element. The total equilibrium problem then involves satisfying the elastic requirements within each span, together with the interconnection requirements at the joints. These interconnection requirements are that adjacent spans should have the same inclination and the same bending moment at their common junction. The internal elastic requirements for a single span are one stage more complicated than the corresponding single-element relations in the foregoing examples. Here each span is itself a two-degree-of-freedom system described by two geometric quantities (the inclinations at the ends) and by two force quantities (the bending moments at the ends). The relations between these which represent the elastic requirements[1] are shown in Fig. 1-5. Clockwise angles have been called positive. Bending moments which tend to stretch the bottom fibers and compress the top fibers have been called positive. A formulation of the equilibrium

[1] See, for example, L. C. Maugh, "Statically Indeterminate Structures," John Wiley & Sons, Inc., New York, 1946, p. 49.

problem may be obtained by using either inclinations or bending moments to represent the state of the system. Thus a set of independent angles which satisfy the compatibility requirements might be chosen. With the aid of the elastic relations bending moments could then be expressed in terms of these angles, and finally, by writing the conditions for moment

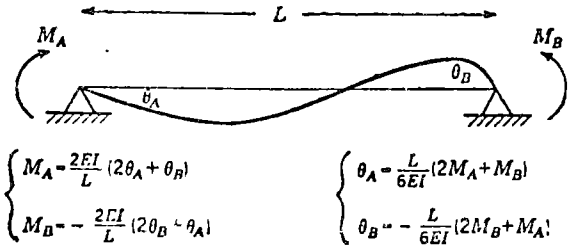


FIG. 1-5. Elastic relationships for a span whose ends are restrained from translation and which is subjected to end moments

balance, a set of equations for determining the angles would be obtained. Alternatively a set of independent bending moments which satisfy the requirements of moment balance could be used to represent the state of the system. The compatibility requirements together with the elastic relations would then furnish equations for determining these moments.

Adopting the former procedure, the state of the system of Fig. 1-4 can

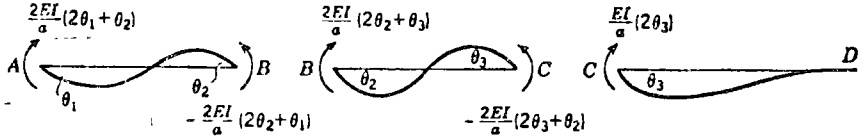


FIG. 1-6. Representation of the beam of Fig. 1-4 in terms of the displacements θ_1, θ_2, and θ_3.

be represented by the clockwise inclinations of the beam at A, B, and C. These angles are denoted by θ_1, θ_2, and θ_3, respectively. Making use of the elastic relations of Fig. 1-5, the terminal bending moments in each span are as indicated in Fig. 1-6.

Governing equations for the angles are now obtained by writing the conditions for moment balance at the supports A, B, and C

$$\begin{aligned} M &= \frac{2EI}{a}(2\theta_1 + \theta_2) \\ -\frac{2EI}{a}(2\theta_1 + \theta_1) &= \frac{2EI}{a}(2\theta_2 + \theta_1) \\ -\frac{2EI}{a}(2\theta_3 + \theta_2) &= \frac{EI}{a}(2\theta_3) \end{aligned} \tag{1-21}$$

These may be cast into nondimensional form by introducing the following dimensionless inclinations:

$$x_1 = \frac{\theta_1}{Ma/2EI} \qquad x_2 = \frac{\theta_2}{Ma/2EI} \qquad x_3 = \frac{\theta_3}{Ma/2EI} \tag{1-22}$$

We thus obtain the following formulation of the equilibrium problem:

$$\begin{aligned} 2x_1 + x_2 &= +1 \\ x_1 + 4x_2 + x_3 &= 0 \\ x_2 + 3x_1 &= 0 \end{aligned} \tag{1-23}$$

A complementary formulation may be obtained in terms of the bending moments M_1, M_2, and M_3 at B, C, and D, respectively, in the beam of Fig. 1-4. It is left as an exercise for the reader to show that in terms of the dimensionless moments

$$y_1 = \frac{M_1}{M} \qquad y_2 = \frac{M_2}{M} \qquad y_3 = \frac{M_3}{M} \tag{1-24}$$

the governing equations are as follows:

$$\begin{aligned} 4y_1 + y_2 &= -1 \\ y_1 + 6y_2 + 2y_3 &= 0 \\ 2y_2 + 4y_3 &= 0 \end{aligned} \tag{1-25}$$

Problem 1-5. Hydraulic Network. We consider the problem of determining the steady flow of an incompressible fluid in a network of branched pipes under the assumption that the pressure drop in a single

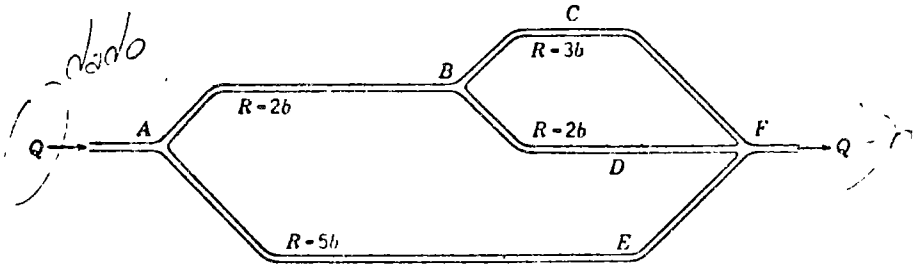


FIG. 1-7. Schematic diagram of hydraulic network passing a total flow Q

branch is proportional to the square of the rate of flow through that branch. Figure 1-7 shows the plan of a particular pipe network. The total rate of flow, in at A and out at F, is Q. For a single branch the

pressure drop in the direction of flow is given[1] by the following resistance law,

$$\Delta p = Rq^2 \tag{1-26}$$

where q is rate of flow through the branch and R is a resistance coefficient. The resistance coefficient of each branch in Fig. 1-7 is given in terms of b.

The equilibrium problem consists in determining the pressure and flow distribution in the steady state. To make the problem definite, we assume that Q is given and that the pressure at F is zero. The governing requirements are that the pressure at each junction should be single-valued, that the rate of flow into any junction should equal the rate of flow out of that junction, and that in each separate branch the resistance law (1-26) should be satisfied. A formulation of the problem can be made in terms of either junction pressures or branch flow rates. Thus the state of the system can be represented by p_1 and p_2, the pressures at A and B, respectively. In terms of these the flow rates in the individual branches are given by (1-26).

$$\begin{aligned} q_{AB} &= \left(\frac{p_1 - p_2}{2b}\right)^{\frac{1}{2}} \\ q_{BCF} &= \left(\frac{p_2}{3b}\right)^{\frac{1}{2}} \\ q_{BDF} &= \left(\frac{p_2}{2b}\right)^{\frac{1}{2}} \\ q_{AEF} &= \left(\frac{p_1}{5b}\right)^{\frac{1}{2}} \end{aligned} \tag{1-27}$$

The requirement of continuity of flow at the junctions A and B provides the following governing equations:

$$\begin{aligned} Q &= \left(\frac{p_1 - p_2}{2b}\right)^{\frac{1}{2}} + \left(\frac{p_1}{5b}\right)^{\frac{1}{2}} \\ \left(\frac{p_1 - p_2}{2b}\right)^{\frac{1}{2}} &= \left(\frac{p_2}{3b}\right)^{\frac{1}{2}} + \left(\frac{p_2}{2b}\right)^{\frac{1}{2}} \end{aligned} \tag{1-28}$$

A nondimensional formulation may be obtained by introducing dimensionless pressures

$$x_1 = \frac{p_1}{bQ^2} \qquad x_2 = \frac{p_2}{bQ^2} \tag{1-29}$$

[1] See, for example, H. W. King, C. O. Wisler, and J. G. Woodburn, "Hydraulics," John Wiley & Sons, Inc., New York, 1948, 5th ed., p. 220. Strictly speaking we should consider Δp and q as directed quantities and write $\Delta p = [\text{sign } (q)]Rq^2$. If we use (1-26), it is incumbent on us to check that all pressure drops are actually in the direction of flow in any proposed solution.

In terms of these (1-28) may be cast into the following form:

$$\begin{aligned} 0.4472x_1^{\frac{1}{2}} + 0.7071(x_1 - x_2)^{\frac{1}{2}} &= 1 \\ 0.7071(x_1 - x_2)^{\frac{1}{2}} - 1.2845x_2^{\frac{1}{2}} &= 0 \end{aligned} \tag{1-30}$$

A *complementary* formulation may be obtained in terms of branch flow rates. Continuity of flow will be preserved in Fig. 1-7 if the flow rates q_1 and q_2 in the branches AB and BCF, respectively, are independent provided the flow rates in the remaining branches are taken as follows.

$$\begin{aligned} q_{BDF} &= q_1 - q_2 \\ q_{AEF} &= Q - q_1 \end{aligned} \tag{1-31}$$

With the aid of (1-26) the requirement of single-valued pressures at A and B leads to the following governing equations:

$$\begin{aligned} 2bq_1^2 + 2b(q_1 - q_2)^2 &= 5b(Q - q_1)^2 \\ 3bq_2^2 &= 2b(q_1 - q_2)^2 \end{aligned} \tag{1-32}$$

Introducing the dimensionless flow rates

$$y_1 = \frac{q_1}{Q} \qquad y_2 = \frac{q_2}{Q} \tag{1-33}$$

we obtain a nondimensional formulation as follows:

$$\begin{aligned} 10y_1 - y_1^2 - 4y_1y_2 + 2y_2^2 &= 5 \\ +2y_1^2 - 4y_1y_2 - y_2^2 &= 0 \end{aligned} \tag{1-34}$$

EXERCISES

1-1. The lengths and cross-sectional areas of the bars of a plane pinned truss are indicated in Fig. 1-8. The bars are joined by frictionless pins, and each one satisfies Hooke's law, $f/A = E\delta/L$, where f is the tensile force and δ is the elongation. The

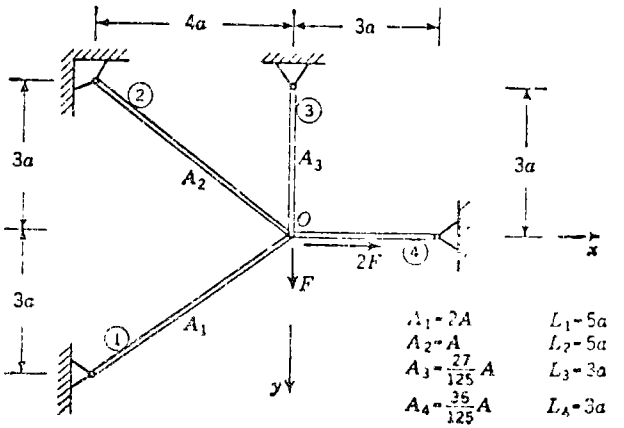


FIG. 1-8 Exercise 1-1.

CHAPTER 2

EIGENVALUE PROBLEMS FOR SYSTEMS WITH A FINITE NUMBER OF DEGREES OF FREEDOM

Equilibrium problems involve the determination of system configurations under prescribed loading conditions. An eigenvalue problem may also involve the determination of system configurations, but of greater importance is the determination of the critical loading conditions under which these configurations are possible. A parameter which describes such a critical condition is called an eigenvalue. As examples we have the *natural frequencies* in oscillating systems and the *buckling loads* in elastic-stability problems.

We consider only *linear* eigenvalue problems. Matrix notation is used because it facilitates the theoretical discussion and because it provides a useful system for laying out the actual computations. The necessary rules are briefly reviewed in Sec. 2-2.

2-1. Particular Examples

Two examples are used to illustrate the formulation of eigenvalue problems from physical systems:

2-1. Three-mass vibrating system.

2-2. Buckling of a structure

In both cases the formulations are left in ordinary algebraic form. Matrix formulations will be given in Sec 2-2.

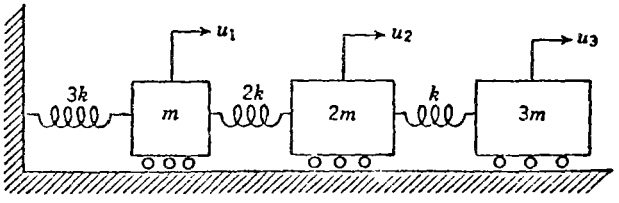


FIG. 2-1. Vibrational system with three degrees of freedom.

Problem 2-1. Three-mass Vibrating System. The system is shown in Fig. 2-1. The displacements of the three masses from the unstrained configuration are measured by u_1, u_2, and u_3. The equations of motion may be written by imagining the system disturbed from equilibrium and

applying Newton's second law to each mass. Neglecting friction, we obtain

$$\begin{aligned} -3ku_1 + 2k(u_2 - u_1) &= m\frac{d^2u_1}{dt^2} \\ -2k(u_2 - u_1) + k(u_3 - u_2) &= 2m\frac{d^2u_2}{dt^2} \\ -k(u_3 - u_2) &= 3m\frac{d^2u_3}{dt^2} \end{aligned} \tag{2-1}$$

For a *natural* vibration we would have

$$\begin{aligned} u_1 &= x_1 \sin(\omega t + \varphi) \\ u_2 &= x_2 \sin(\omega t + \varphi) \\ u_3 &= x_3 \sin(\omega t + \varphi) \end{aligned} \tag{2-2}$$

where x_1, x_2, and x_3 represent the amplitudes of vibration, ω is the natural frequency, and φ is a phase angle. If we substitute (2-2) into (2-1) and set

$$\frac{m\omega^2}{k} = \lambda \tag{2-3}$$

we find the following equations as the conditions for determining the amplitudes and frequency:

$$\begin{aligned} 5x_1 - 2x_2 &= \lambda(x_1) \\ -2x_1 + 3x_2 - x_3 &= \lambda(2x_2) \\ -x_2 + x_3 &= \lambda(3x_3) \end{aligned} \tag{2-4}$$

The parameter λ is a dimensionless measure of the frequency. An *eigenvalue* is a value of λ for which there are nonzero amplitudes which satisfy (2-4). A configuration of amplitudes which meets these requirements is called a *natural mode*. The corresponding frequency is called a *natural frequency*. A complete solution would involve finding all the natural frequencies and their associated natural modes. In technical problems it may not be of interest to obtain the complete solution. Sometimes only the lowest natural frequency is desired; sometimes just the lowest frequency and the corresponding mode or just the two lowest frequencies are desired.

Problem 2-2. Buckling of a Structure. A system of rigid weightless links hinged together and supported by springs is shown in Fig. 2-2*a*. In this position all three links are exactly vertical, and there is no force in any of the springs. We consider the stability of this system when subjected to a vertical load P applied at B. For small loads the three links will remain vertical, moving down as a unit against the springs k_1. For large loads the links will buckle; that is, B and C will undergo transverse displacements as shown in Fig. 2-2*b*. Our problem is to determine

the stability limit for the vertical position. We want to know the value of P for which an equilibrium position with transverse displacements first becomes possible.

To obtain a quantitative analysis, we assume that the desired critical buckling load is holding the system in equilibrium and find the equilibrium conditions by applying the principle of minimum potential energy. A geometrically compatible state can be represented by arbitrary (small) values of y_1, x_1, and x_2 if we take y_2 as

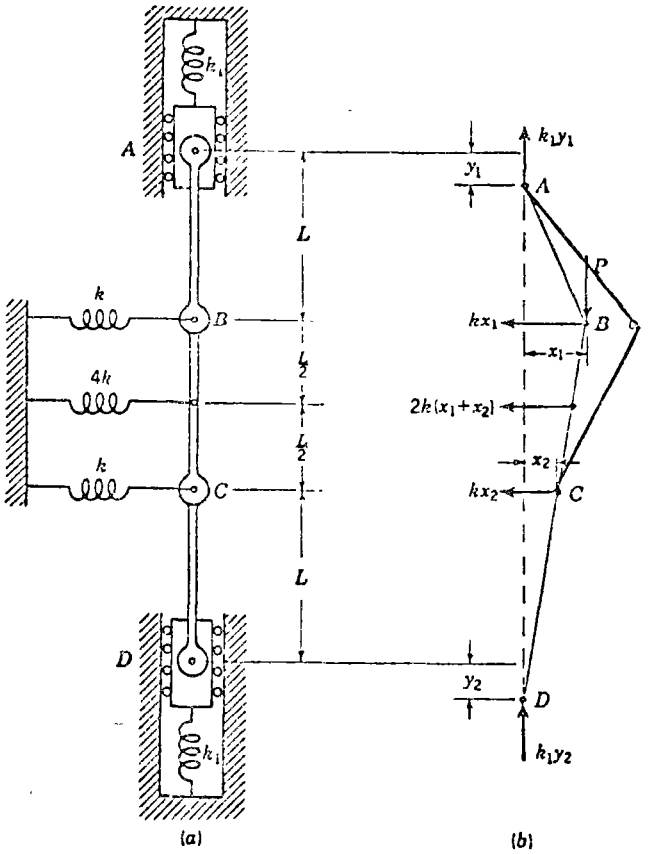


FIG 2-2. Buckling of a system of spring-supported rigid links

$$y_2 = y_1 - \frac{x_1^2}{2L} - \frac{(x_1 - x_2)^2}{2L} - \frac{x_2^2}{2L} \tag{2-5}$$

The usual small-angle approximations, $1 - \cos\theta \approx \frac{1}{2}\theta^2$ and $\sin\theta \approx \theta$, have been made here. By adding the strain energy of the springs to the

potential energy of the load P we have the total potential energy

$$\Phi = \tfrac{1}{2}k_1y_1^2 + \tfrac{1}{2}kx_1^2 + \tfrac{1}{2}k(x_1 + x_2)^2 + \tfrac{1}{2}kx_2^2 + \tfrac{1}{2}k_1y_2^2 - P\left(y_1 - \frac{x_1^2}{2L}\right) \tag{2-6}$$

where y_2 is understood to take the value (2-5). The equilibrium equations are the conditions for stationary potential energy.

$$\begin{aligned}
\frac{\partial\Phi}{\partial y_1} &= k_1y_1 + k_1y_2 - P = 0 \\
\frac{\partial\Phi}{\partial x_1} &= kx_1 + k(x_1 + x_2) + k_1y_2\left(-\frac{x_1}{L} - \frac{x_1 - x_2}{L}\right) + \frac{Px_1}{L} = 0 \\
\frac{\partial\Phi}{\partial x_2} &= k(x_1 + x_2) + kx_2 + k_1y_2\left(\frac{x_1 - x_2}{L} - \frac{x_2}{L}\right) = 0
\end{aligned} \tag{2-7}$$

One solution of this system is $x_1 = x_2 = 0$ and

$$y_2 = y_1 = \frac{P}{2k_1} \tag{2-8}$$

which is obtained from (2-5) and the first of (2-7). This is the unbuckled equilibrium position.

If x_1 and x_2 do not vanish, we obtain

$$\begin{aligned}
y_1 &= \frac{P}{2k_1} + \frac{1}{4L}[x_1^2 + (x_1 - x_2)^2 + x_2^2] \\
y_2 &= \frac{P}{2k_1} - \frac{1}{4L}[x_1^2 + (x_1 - x_2)^2 + x_2^2]
\end{aligned} \tag{2-9}$$

by solving (2-5) and the first of (2-7). We next insert the second of (2-9) into the last two relations of (2-7) to get a pair of simultaneous equations in x_1 and x_2. These equations contain linear and cubic terms. Since we are interested in the first appearance of buckling, we need consider only such small values of x_1 and x_2 that the cubic terms may be neglected in comparison with the linear terms. The linearized equations for x_1 and x_2 then appear as follows:

$$\begin{aligned}
kx_1 + k(x_1 + x_2) + \frac{P}{2L}(-2x_1 + x_2) + \frac{Px_1}{L} &= 0 \\
k(x_1 + x_2) + kx_2 + \frac{P}{2L}(x_1 - 2x_2) &= 0
\end{aligned} \tag{2-10}$$

By introducing the dimensionless parameter

$$\lambda = \frac{2kL}{P} \tag{2-11}$$

we obtain

$$\begin{aligned}
-x_2 &= \lambda(2x_1 + x_2) \\
-x_1 + 2x_2 &= \lambda(x_1 + 2x_2)
\end{aligned} \tag{2-12}$$

as our formulation of the eigenvalue problem.

An *eigenvalue* is a value of λ for which the equations permit nonvanishing displacements. Such a configuration of displacements is called a *buckling mode*.

A complete solution of an eigenvalue problem involves finding all possible eigenvalues with their associated modes. In technical buckling problems a complete solution is not of interest. Very often the magnitude of the smallest buckling load is all that is required. Sometimes the corresponding buckling mode is of interest in order to assist in the design of stiffening reinforcement.

The present system has the interesting feature that if the load P is reversed (i.e., applied vertically upward) there is still the possibility of buckling. In such cases both the smallest positive and smallest negative buckling loads are of practical interest.

EXERCISES

2-1. Show that the eigenvalue problem for determining the natural frequencies and modes of torsional vibration of the system of Fig. 2-3 may be formulated as follows:

$$\begin{aligned}
x_1 - x_2 &= \lambda x_1 \\
-x_1 + 2x_2 - x_3 &= \lambda x_2 \\
-x_2 + 2x_3 - x_4 &= \lambda x_3 \\
-x_3 + \tfrac{3}{2}x_4 - \tfrac{1}{2}x_5 &= \lambda x_4 \\
-\tfrac{1}{2}x_4 + \tfrac{1}{2}x_5 &= 4\lambda x_5
\end{aligned}$$

where $\lambda = \omega^2J/k$ and k is the torsional stiffness of a shaft and J is the moment of

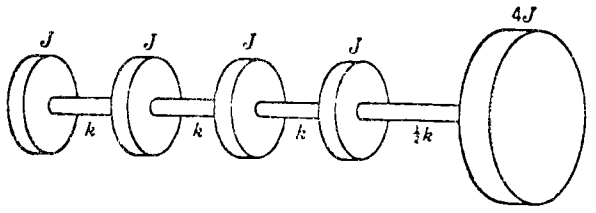


Fig. 2-3 Exercise 2-1

inertia of a disk. The system is so supported on frictionless bearings that it is free to rotate without any bending of the shafts.

2-2. At resonance let the currents in Fig. 2-4 be

$$\begin{aligned}
I_1 &= x_1 \sin(\omega t + \varphi) \\
I_2 &= x_2 \sin(\omega t + \varphi)
\end{aligned}$$

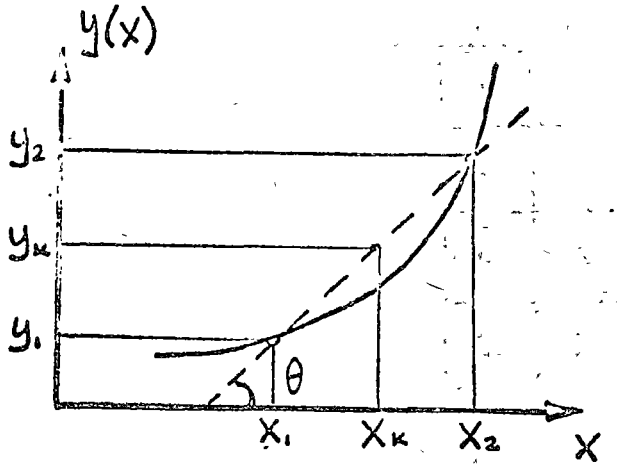
centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL

ARMANDO TORRES FENTANES

ABRIL DE 1976

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels: 521-40-23 521-73-35 5123-123



$$X_2 = X_1 + h$$

$$X_3 = X_2 + h$$

$$m = tg\,\theta = \frac{Y_2 - Y_1}{X_2 - X_1} \qquad (IV.0)$$

$$= \frac{Y_k - Y_1}{X_k - X_1} \qquad (IV.1)$$

$$= \frac{Y_k - Y_2}{X_k - X_2} \qquad (IV.2)$$

de las ecuaciones (IV.0), (IV.1), (IV.2) :

$$Y_k = Y_1 + \left[\frac{Y_2 - Y_1}{X_2 - X_1}\right][X_k - X_1]$$

$$Y_k = Y_2 + \left[\frac{Y_2 - Y_1}{X_2 - X_1}\right][X_k - X_2]$$

$$X_1 \leq X_k \leq X_2$$

Ejemplo

Si se tiene la siguiente tabla de puntos muestrales :

K	X	Y
0	0	1
1	2	4
2	4	9
3	6	12

encontrar el valor de Y para $X = 0.5$

Sol.

$$Y(0.5) = Y_o + \left[\frac{Y_1 - Y_o}{X_1 - X_o}\right]\left[X_k - X_o\right]$$

$$= 1 + \left[\frac{3}{2}\right]\left[\frac{1}{2}\right]$$

$$= 1 + \frac{3}{4}$$

$$= 7/4$$

Método de Newton

Este procedimiento es más exacto y su demostración cae fuera de los propósitos del curso. Se dice que para una serie de puntos muestrales, el valor Y_k correspondiente a X_k está dado por :

$$Y_k = Y_o + K A_o + \frac{K(K-1)}{2!} b_o + \frac{(K)(K-1)(K-2)}{3!} C_o \qquad (IV.3)$$

donde :

$Y_o = Y(X_o)$

X_o : valor inmediato anterior de X_k

$$K = |X_k - X_0|$$

A_o : primeras diferencias de X_o

b_o : segundas diferencias de X_o

C_o : terceras diferencias de X_o

⋮ ⋮

Las diferencias se obtienen en la siguiente forma :

X_i	Y_i	ΔY_i	$\Delta^2 Y_i$	$\Delta^3 Y_i$	...
X_o	Y_o	$a_o = Y_1 - Y_o$			...
$X_1 = X_o + h$	Y_1	$a_1 = Y_2 - Y_1$	$b_o = a_1 - a_o$	$C_o = b_1 - b_o$	...
$X_2 = X_1 + h$	Y_2	$a_2 = Y_3 - Y_2$	$b_1 = a_2 - a_1$		...
$X_3 = X_2 + h$	Y_3	⋮	⋮		...
⋮	⋮	⋮	⋮	⋮	...
$X_n = X_{n-1} + h$	Y_n				...

La aproximación será más exacta entre mayor cantidad de términos se utilice.

Ejemplo

Hallar y (0.5) para la siguiente tabla muestral :

X_i	Y_i
0	1
2	4
4	9
6	12

Sol :

X_i	Y_i	Y_i (a)	^2Y_i (b)	^3Y_i (c)
0	1			
		3		
2	4		2	
		5		-4
4	9		-2	
		3		
6	12			

1/2 ←

$$K = X_k - X_o = 0.5$$

$$X_o = 0$$

$$Y_o = 1$$

$$Y_k = Y_o + K A_o + \frac{K(k-1)}{2!} b_o + \frac{K(k-1)(k-2)}{3!} C_o$$

$$Y(0.5) = 1 + \frac{1}{2}(3) + \frac{1}{2}\left(-\frac{1}{2}\right)\frac{1}{2}(2) + \frac{1}{2}\left(-\frac{1}{2}\right)\left(-\frac{3}{2}\right)\frac{1}{6}(-4)$$

$$= 1 + \frac{3}{2} - \frac{1}{4} - \frac{1}{4}$$

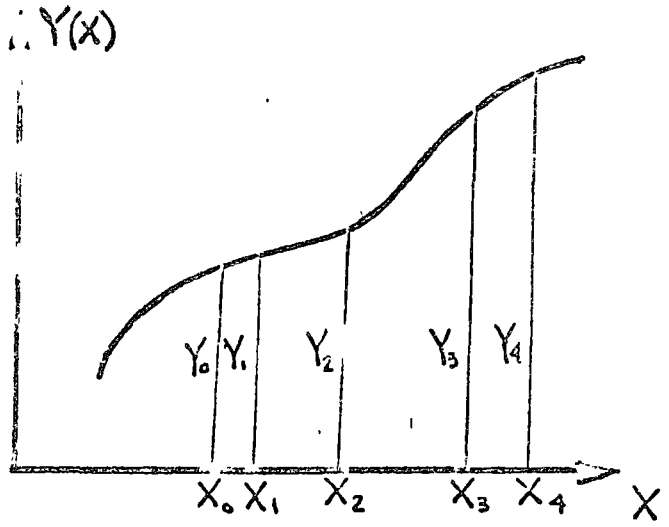
$$= 1 + \frac{3}{2} - \frac{2}{4} = 1 + \frac{3}{2} - \frac{1}{2}$$

$$Y(0.5) = 2$$

b) Interpelación con valores muestrales desigualmente espaciados.

Método de Lagrange

En este caso se cuenta con una serie de valores muestrales del siguiente tipo:



Se supone que dichos puntos se pueden aproximar por un polinomio de orden "n-1", donde "n" son la cantidad de puntos muestrales, ó sea :

$$Y(X) = A_{n-1} X^{n-1} + A_{n-2} X^{n-2} + \ldots + A_o$$

este polinomio se puede representar en la siguiente forma :

$$\begin{aligned} Y(X) = {} & A_1 (X-X_2)(X-X_3) \cdots (X-X_n) + \\ & + A_2 (X-X_1)(X-X_3) \cdots (X-X_n) + \ldots \\ & + A_n (X-X_1)(X-X_2) \cdots (X-X_{n-1}) \end{aligned} \qquad (IV.4)$$

donde $A_1 \ldots A_n$ se determinan en forma tal que el polinomio satisfaga los puntos muestrales, en base a lo anterior y empleando (IV.4) se tiene :

$$Y_1 = A_1 (X_1-X_2) (X_1-X_3) \ldots . (X_1-X_n)$$

$$A_1 = \frac{Y_1}{(X_1-X_2) (X_1-X_3) \ldots .(X_1-X_n)}$$

$$Y_2 = A_2 (X_2 - X_1) (X_2-X_3) \ldots . (X_2 - X_n)$$

$$A_2 = \frac{Y_2}{(X_2-X_1)(X_2 - X_3)\ldots . (X_2-X_n)}$$

$$\vdots$$

$$Y_n = A_n (X_n-X_1) (X_n-X_2) \ldots . (X_n-X_{n-1})$$

$$A_n = \frac{Y_n}{(X_n-X_1) (X_n-X_2) \ldots . (X_n-X_{n-1})}$$

substituyendo estos valores en (IV.4) :

$$Y(X) = \frac{(X-X_2) (X-X_3)\ldots .(X-X_n)}{(X_1-X_2) (X_1-X_3) \ldots (X_1-X_n)} Y_1 + \ldots .+$$

$$+ \frac{(X - X_1) (X-X_2) \ldots . (X-X_{n-1})}{(X_n-X_1) (X_n-X_2) \ldots (X_n-X_{n-1})} Y_n$$

La fórmula anterior permite evaluar Y(X) aun en el caso de que los puntos estén igualmente espaciados.

Ejemplo

Determinar y(0.5) para la siguiente tabla muestral :

n	X	Y
1	0	1
2	2	4
3	4	9
4	6	12

Sol.

$$Y(0.5) = \frac{(.5-2)(.5-4)(.5-6)}{(0-2)(0-4)(0-6)}(1) + \frac{(.5-0)(.5-4)(.5-6)}{(2-0)(2-4)(2-6)}(4) + \frac{(.5-0)(.5-2)(.5-6)}{(4-0)(4-2)(4-6)}(9) + \frac{(.5-0)(.5-2)(.5-4)}{(6-0)(6-2)(6-4)}(12)$$

$$Y(0.5) = 0.602 + 2.406 - 2.320 + .65625$$

$$Y(0.5) = 1.344$$

c) Método de los mínimos cuadrados.

Este procedimiento se utiliza para aproximar una serie de "n" puntos a un polinomio de orden "m", en forma tal que pase lo más cercano posible a todos los puntos. Para ello se minimiza la suma de los cuadrados de los errores.

Sea f(x) el polinomio aproximado de la función que da los valores muestrales :

$$f(x) = A_m X^m + A_{m-1} X^{m-1} + \ldots + A_1 X + A_o \qquad (IV.5)$$

El valor de la variable dependiente que corresponde a la variable independiente X_i es Y_i, por lo que el error estará dado por :

$$e_i = f(x_i) - y_i$$

$$e_i = A_m X_i^m - A_{m-1} X_i^{m-1} + \ldots + A_o - Y_i$$

y la suma de los cuadrados de los errores considerando todos los puntos será :

$$\sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} [A_o + A_1 X_i + A_2 X_i^2 + \ldots + A_m X_i^m - Y_i]^2$$

para obtener el mínimo se deriva la expresión (IV.6) con respecto a los parámetros A_j y dichas expresiones se igualan a cero :

$$\frac{\partial}{\partial A_j} \sum_{i=1}^{n} e_i^2 = \frac{\partial}{\partial A_j} \sum_{i=1}^{n} [A_o + A_1 X_i + \ldots + A_m X^m - Y_i]^2$$

$$= \sum_{i=1}^{n} 2 [A_o + A_1 X_i + \ldots + A_m X_i^m - Y_i] X_i^j = 0$$

lo cual se cumple solo si :

$$A_o \sum_{i=1}^{n} X_i^j + A_1 \sum_{i=1}^{n} X_i^{j+1} + \ldots + A_m \sum_{i=1}^{n} X_i^{j+m} = \sum_{i=1}^{n} X_i^j Y_i \qquad (IV.7)$$

al evaluar (IV.7) para todas las "j" se tiene :

$$n A_o + A_1 \Sigma X + A_2 \Sigma X^2 + \ldots + A_m \Sigma X^m = \Sigma Y$$

$$A_o \Sigma X + A_1 \Sigma X^2 + A_2 \Sigma X^3 + \ldots + A_m \Sigma X^{m+1} = \Sigma XY$$

$$\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots$$

$$A_o \Sigma X^m + A_1 \Sigma X^{m+1} + A_2 \Sigma X^{m+2} + \ldots + A_m \Sigma X^{m+m} = \Sigma X^m Y$$

dicho sistema se resuelve para obtener los parámetros A_j, los cuales se substituyen en la expresión (IV.5).

Ejemplo

Aproximar mediante una recta, empleando el método de mínimos cuadrados, los siguientes puntos :

X	Y
1	2
3	7
4	8
5	10
6	11

Sol.

La ecuación será de la forma :

$$Y(x) = A_o + A_1 X \qquad \text{(IV.7)}$$

por lo que se tiene que resolver el siguiente sistema :

$$\left.\begin{aligned} n A_o + A_1 \Sigma X &= \Sigma Y \\ A_o \Sigma X + A_1 \Sigma X^2 &= \Sigma XY \end{aligned}\right\} \qquad \text{(IV.8)}$$

para resolverlo se construye la siguiente tabla :

	X	Y	X^2	XY
	1	2	1	2
	3	7	9	21
	4	8	16	32
	5	10	25	50
	6	11	36	66
$\sum$	19	38	87	171

$n = 5$

substituyendo en (IV.8)

$$5 A_o + 19 A_1 = 38$$

$$19 A_o + 87 A_1 = 171$$

resolviendo el sistema :

$$A_o = \frac{38 - 19 A_1}{5}$$

$$\frac{19}{5} \left[38 - 19 A_1 \right] + 87 A_1 = 171$$

$$144.4 - 72.2 A_1 + 87 A_1 = 171$$

$$A_1 = \frac{171 - 144}{87 - 72.2} = 1.824$$

$$A_o = \frac{38 - 34.662}{5} = 0.667$$

por lo que :

$$Y(x) = 0.667 + 1.824 X$$

V) DERIVACION E INTEGRACION NUMERICA

1. Derivación

Los métodos de derivación numérica son aplicables en funciones bien comportadas y contínuas.

Si se tiene una tabulación de puntos, una forma de obtener el valor de la derivada para un punto dado es aproximar los puntos mediante un polinomio de orden "m", derivarlo y evaluar la derivada en el punto considerado. El otro método aprovecha la posibilidad de expandir una función en series de Taylor si dicha función es contínua.

a) Método de las diferencias

En este método se expande la función mediante una serie de Taylor alrededor del punto considerado y se llega a una serie de expresiones correspondientes a la primera, segunda, tercera derivada, etc.

Una versión modificada es aceptar que se pueden aproximar los puntos por un polinomio aplicando el método de Newton y derivar dicha expansión.

Sea:

$$F(x) = Y_0 + K\Delta Y_0 + \frac{K(K-1)}{2!}\Delta^2 Y_0 + \frac{K(K-1)(K-2)}{3!}\Delta^3 Y_0 + \ldots \qquad (V.0)$$

donde :

$$K = \frac{X - X_o}{h}$$

$$h = X_i - X_{i-1}$$

ΔY_o : primeras diferencias

ΔY^2 : segundas diferencias

ΔY^3 : terceras diferencias

$\vdots$

de lo anterior se concluye :

$$\frac{dk}{dx} = \frac{1}{h} \qquad (V.1)$$

derivando (V.1) con respecto a "X" :

$$-\frac{dF}{dX} = \frac{d}{dK}\left[Y_o + K\Delta Y_o + \frac{K^2-K}{2!}\Delta^2 Y_o + \frac{K^3-3K^2+2K}{3!}\Delta^3 Y_o + \ldots\right]\frac{dK}{dX}$$

$$-\frac{dF}{dX} = \frac{1}{h}\left[\Delta Y_o + \frac{2K-1}{2}\Delta^2 Y_o + \frac{3K^2-6K+2}{6}\Delta^3 Y_o + \ldots\right] \qquad (V.2)$$

derivando (V.2) con respecto a X :

$$\frac{d^2F}{dX^2} = \frac{1}{h}\frac{d}{dK}\left[\Delta Y_o + \frac{2K-1}{2!}\Delta^2 Y_o + \ldots\right]\frac{dK}{dX}$$

$$= \frac{1}{h^2}\left[\Delta^2 Y_o + (K-1)\Delta^3 Y_o + \ldots\right] \qquad (V.3)$$

derivando V.3 :

$$\frac{d^3 F}{dX^3} = \frac{1}{h^3} \left[\Delta^3 Y_0 + \dots \right]$$

las fórmulas anteriores dan las expresiones para cada una de las derivadas y - según se tomen los términos de primer, segundo, tercer orden, etc. se habla - de fórmulas de diferencias de primer, segundo, tercer orden, etc. Como muestra tenemos :

- diferencias de primer orden

$$F'(X_0) \triangleq \frac{1}{h} \Delta y_0 = \frac{Y_1 - Y_0}{h}$$

$$y'(x) \triangleq \frac{1}{h} \left[\underline{-1} \quad \underline{1} \right]$$

donde $\underline{1}$ significa que se trata del coeficiente del valor y(x) .

- diferencias de segundo orden

$$F'(X) \doteq \frac{1}{h} \left[\Delta Y_0 + \frac{2K-1}{2} \Delta^2 Y_0 \right]$$

$$F''(X) \doteq \frac{1}{h^2} \left[\Delta Y_0 \right]$$

$$\Delta Y_0 = Y_1 - Y_0$$

$$\Delta^2 Y_0 = Y_2 - 2Y_1 + Y_0$$

$$F'(X) = \frac{1}{h} \left[Y_1 - Y_0 + \frac{2K-1}{2} (Y_2 - 2Y_1 + Y_0) \right]$$

$$F''(X) = \frac{1}{h^2} \left[Y_2 - 2Y_1 + Y_0 \right]$$

si $X = X_0 \Rightarrow K = 0$

$$F'(X_0) = \frac{1}{2h}\left[-\underline{3} \quad 4 \quad -1\right]$$

$$F''(X_0) = \frac{1}{h^2}\left[\underline{1} \quad -2 \quad 1\right]$$

} diferencias hacia adelante

si $X = X_1 = X_0 + h \Rightarrow K = 1$

$$F'(X_1) = \frac{1}{2h}\left[-1 \quad \underline{0} \quad 1\right]$$

$$F''(X_1) = \frac{1}{h^2}\left[1 \quad \underline{-2} \quad 1\right]$$

} diferencias centrales

si $X = X_2 = X_0 + 2h \Rightarrow K = 2$

$$F'(X_2) = \frac{1}{2h}\left[1 \quad -4 \quad \underline{3}\right]$$

$$F''(X_2) = \frac{1}{h^2}\left[1 \quad -2 \quad \underline{1}\right]$$

} diferencias hacia atrás

De igual forma se obtienen las fórmulas para diferencias de mayor orden. A continuación se da una tabla de fórmulas de derivación hasta tercer orden, indicando cual es el elemento para el cual se está evaluando la derivada. Las que mayor exactitud proporcionan son las fórmulas de diferencias centrales.

Tabla de fórmulas para derivación numérica

primer orden

$$Y'(X) = \frac{1}{h}\left[-\underline{1} \quad 1\right]$$

$$Y'(X) = \frac{1}{h}\left[-1 \quad \underline{1}\right]$$

segundo orden

$$Y'(x)=\frac{1}{2h}\left[\underline{\underline{-3}}\ \ 4\ -1\right]$$

$$Y'(x)=\frac{1}{2h}\left[-1\ \ \underline{\underline{0}}\ \ 1\right]$$

$$Y'(x)=\frac{1}{2h}\left[1\ -4\ \ \underline{\underline{3}}\right]$$

$$Y''(x)=\frac{1}{h^2}\left[\underline{\underline{1}}\ -2\ \ 1\right]$$

$$Y''(x)=\frac{1}{h^2}\left[1\ \ \underline{\underline{-2}}\ \ 1\right]$$

$$Y''(x)=\frac{1}{h^2}\left[1\ -2\ \ \underline{\underline{1}}\right]$$

tercer orden

$$Y'(x)=\frac{1}{6h}\left[\underline{\underline{-11}}\ \ 18\ -9\ \ 2\right]$$

$$Y'(x)=\frac{1}{6h}\left[-2\ \ \underline{\underline{-3}}\ \ 6\ -1\right]$$

$$Y'(x)=\frac{1}{6h}\left[1\ -6\ \ \underline{\underline{3}}\ \ 2\right]$$

$$Y'(x)=\frac{1}{6h}\left[-2\ \ 9\ -18\ \ \underline{\underline{11}}\right]$$

$$Y''(x) = \frac{1}{h^2}\left[\underline{\underline{2}} \quad -5 \quad 4 \quad -1\right]$$

$$Y''(x) = \frac{1}{h^2}\left[1 \quad \underline{\underline{-2}} \quad 1 \quad 0\right]$$

$$Y''(x) = \frac{1}{h^2}\left[0 \quad 1 \quad \underline{\underline{-2}} \quad 1\right]$$

$$Y''(x) = \frac{1}{h^2}\left[-1 \quad 4 \quad -5 \quad \underline{\underline{2}}\right]$$

$$Y'''(x) = \frac{1}{h^3}\left[\underline{\underline{-1}} \quad 3 \quad -3 \quad 1\right]$$

$$Y'''(x) = \frac{1}{h^3}\left[-1 \quad \underline{\underline{3}} \quad -3 \quad 1\right]$$

$$Y'''(x) = \frac{1}{h^3}\left[-1 \quad 3 \quad \underline{\underline{-3}} \quad 1\right]$$

$$Y'''(x) = \frac{1}{h^3}\left[-1 \quad 3 \quad -3 \quad \underline{\underline{1}}\right]$$

Para obtener el valor de las derivadas de mayor orden correspondientes a las diferencias de un orden lo que se hace es correr operadores de dos derivadas de orden menor, es decir, si se quiere 3a. derivada de 2o. orden, hay que aplicar la 1a. y 2a. derivadas de dicho orden en la siguiente forma:

$$\begin{array}{lll} -1 & \left[1 \quad \underline{\underline{-2}} \quad 1 \quad\quad\right] & \\ \underline{\underline{0}} & \left[\quad 1 \quad \underline{\underline{-2}} \quad 1 \quad\right] & \\ 1 & \left[\quad\quad 1 \quad \underline{\underline{-2}} \quad 1\right] & \end{array}$$

$$Y'''(x) = \frac{1}{2h}\left(\frac{1}{h^2}\right)\left[-1 \quad 2 \quad \underline{\underline{0}} \quad -2 \quad 1\right]$$

Ejemplo

Obtener la primera y segunda derivada en X = 3, empleando diferencias de 2o. orden y comparar con los resultados exactos, para los valores de la siguiente tabla :

X	Y
1	0
2	1
3	1.414
4	1.732
5	2

$Y^2 = X - 1$

Sol.

Calculando las derivadas exactas :

$$Y'(X) = \frac{1}{2}(X-1)^{-1/2}\Big|_{X=3} = \frac{1}{2\sqrt{2}} = 0.353$$

$$Y''(X) = -\frac{1}{4}(X-1)^{-3/2}\Big|_{X=3} = -\frac{1}{4\sqrt{2^3}} = -0.088$$

Aplicando fórmulas de diferencias de 2o. orden :

$$Y'(X) \doteq \frac{1}{2h}[-1 \quad \underline{0} \quad 1]$$

$$Y''(X) \doteq \frac{1}{h^2}[1 \quad \underline{-2} \quad 1]$$

$h = 1$

$$Y'(3) = \frac{1}{2}[-1 + 1.732] = 0.366$$

$$Y''(3) = [1 - 2\sqrt{2} + 1.732] = -0.096$$

2. Integración numérica.

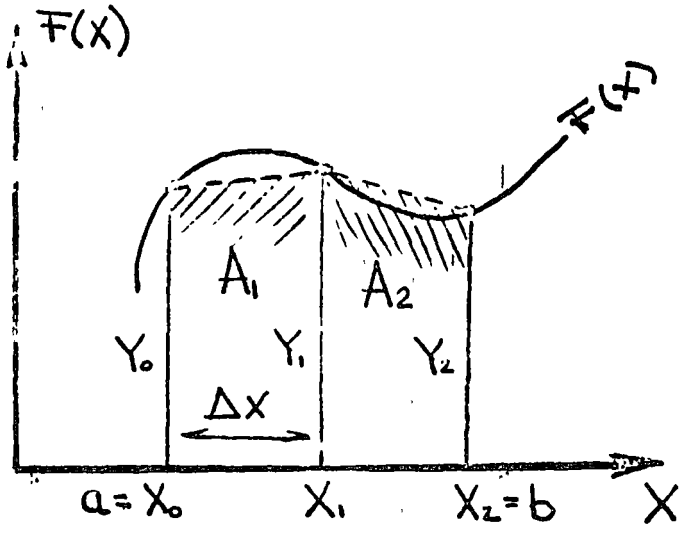
El integrar una curva $y = f(x)$ consiste en encontrar el área bajo dicha curva. En ocasiones es imposible el encontrar la integral exacta de una función dada o en otras no se cuenta con la expresión analítica de la curva; en ambos casos es necesario acudir a los métodos de integración numérica.

Solo trataremos tres de dichos métodos :

Trapezoidal, Simpson de 1/3 y Simpson de 3/8

a) Método Trapezoidal

Debido a que la integral de una función es el área bajo la curva, este método lo que hace es dividir el intervalo de integración en "n" puntos equidistantes y aproxima la curva original por una serie de rectas en cada uno de los "n-1" subintervalos, finalmente se encuentra el área de cada trapezoide y la suma de dichas áreas da la integral en la totalidad del intervalo.



numéricamente se tendrá :

$$\int_a^b F(x)\,dx = \sum_{i=1}^{2} A_i$$

$$A_1 = \frac{\Delta x}{2}\,(Y_0 + Y_1)$$

$$A_2 = \frac{\Delta x}{2}\,(Y_1 + Y_2)$$

para "n" puntos :

$$A_n = \frac{\Delta x}{2}\,(Y_{n-1} + Y_n)$$

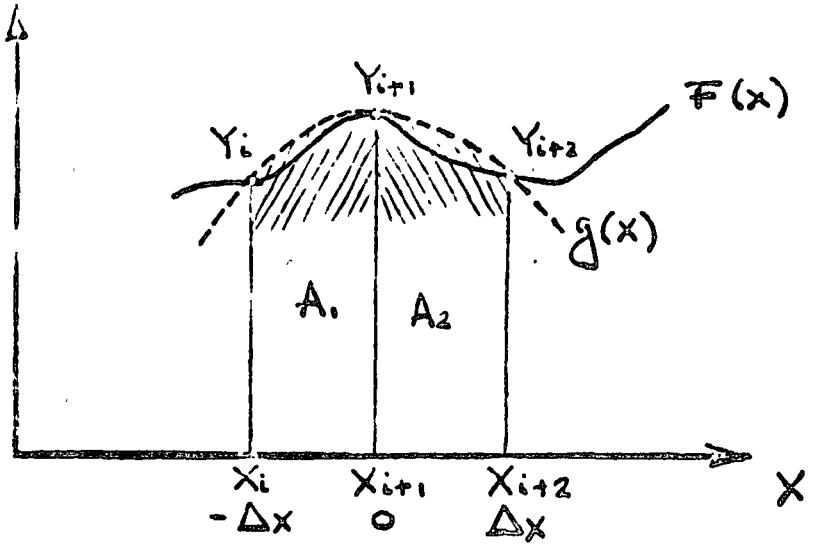
$$\int F(x)\,dx \doteq \frac{\Delta x}{2}\,[Y_0 + 2Y_1 + 2Y_2 + \ldots + 2Y_{n-1} + Y_n]$$

$$\doteq \frac{\Delta x}{2}\,[Y_0 + Y_n + 2\,\Sigma \text{ resto ordenadas}]$$

Para aplicar el método se requiere que el incremento Δx sea lo más - pequeño posible para reducir el error al mínimo. Se puede demostrar que el error producido es del orden de Δx^2.

b) Método de Simpson de 1/3

Este método lo que hace es aproximar 3 puntos sucesivos del intervalo mediante una parábola y calcular el área que se encuentra debajo de esta - curva. El procedimiento se repite para todos los puntos del intervalo (igualmente espaciados) de 3 en 3 y al final se obtiene la suma de todas las áreas.



Numéricamente se tendría :

$$Y = F(x) \doteq g(x)$$

$$g(x) = AX^2 + BX + C \qquad (V.4)$$

$$\int_{-\Delta x}^{\Delta x} F(x)\,dx \doteq \int_{-\Delta x}^{\Delta x} g(x)\,dx = \int_{-\Delta x}^{\Delta x} (AX^2 + BX + C)\,dx$$

$$= \frac{AX^3}{3} + \frac{BX^2}{2} + CX \Big]_{-\Delta x}^{\Delta x}$$

$$= \frac{2A}{3}\Delta x^3 + 2C\,\Delta x \qquad (V.5)$$

para evaluar a, b, c se utiliza (V.4), la cual debe satisfacer todos los puntos del intervalo :

$$\left.\begin{aligned} Y_i &= A\Delta X^2 - B\Delta X + C \\ Y_{i+1} &= C \\ Y_{i+2} &= A\Delta X^2 + B\Delta X + C \end{aligned}\right\} \qquad (V.6)$$

de (V.6) se obtiene :

$$\left.\begin{aligned} A &= \frac{Y_i + 2Y_{i+1} + Y_{i+2}}{2\Delta X^2} \\ B &= \frac{Y_{i+2} - Y_i}{2\Delta X} \\ C &= Y_{i+1} \end{aligned}\right\} \qquad (V.7)$$

substituyendo (V.7) en (V.5) :

$$\int_{-\Delta X}^{\Delta X} F(X)\,dX \cong \frac{\Delta X}{3}\left[Y_i + 4Y_{i+1} + Y_{i+2}\right]$$

para el total del intervalo de integración :

$$A_1 = \frac{\Delta X}{3}\left[Y_0 + 4Y_1 + Y_2\right]$$

$$A_2 = \frac{\Delta X}{3}\left[Y_2 + 4Y_3 + Y_4\right]$$

$$\vdots$$

$$A_{n/2} = \frac{\Delta X}{3}\left[Y_{n-2} + 4Y_{n-1} + Y_n\right]$$

como $\int_a^b F(X)\,dX = \sum_{i=1}^{n/2} A_i$, se tiene :

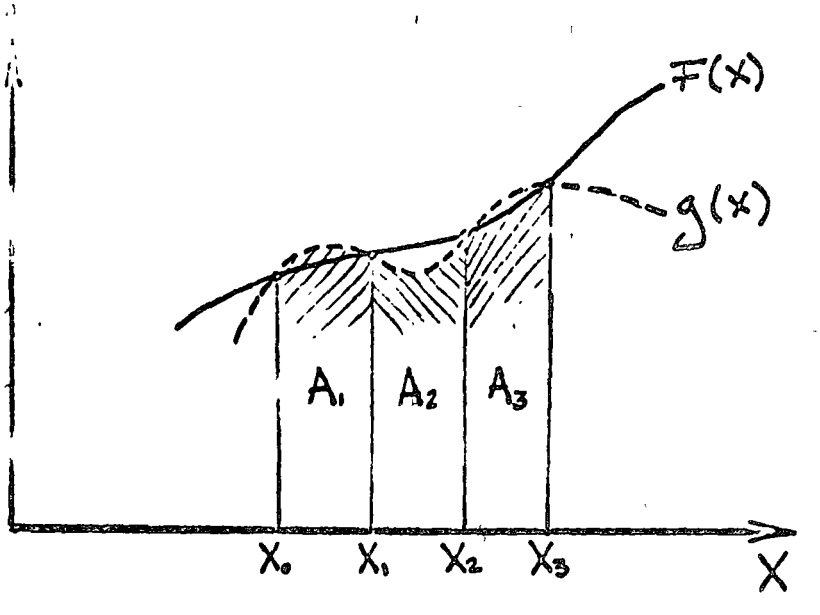
$$\int_a^b F(x)dx \doteq \left[Y_0 + Y_n + 2(Y_2 + Y_4 + \ldots) + 4(Y_1 + Y_3 + \ldots)\right]$$

$$= \frac{\Delta x}{3}\left[Y_0 + Y_n + 2\sum_{i=1}^{n-1} \text{ord. pares} + 4\sum_{i=1}^{n-1} \text{ord. nones}\right]$$

Para aplicar este método se requiere que el número de puntos muestrales sea non o que "n" sea par, si n = 0, 1, 2, ...; en caso contrario se usa una cantidad non de puntos para aplicar este método y el resto del intervalo se integra por el método trapezoidal. Se puede demostrar que el error producido por el método es del orden de Δx^4.

c) Método de Simpson de 3/8

En este caso se conectan 4 puntos del intervalo mediante un polinomio de tercer orden y se evalua la integral bajo dicho polinomio. La integral bajo todo el intervalo será la suma de las áreas encontradas.



Aplicando lo antes dicho se tiene :

$$\int_{X_0}^{X_3} F(x)dx \doteq \int_{X_0}^{X_3} g(x)dx = \sum_{i=1}^{3} A_i$$

$$g(x) = AX^3 + BX^2 + CX + D \qquad (V.8)$$

$$\int_{X_0}^{X_3} F(x)dx \doteq \int_{X_0}^{X_3} (AX^3 + BX^2 + CX + D)dX$$

$$= \frac{AX^4}{4} + \frac{BX^3}{3} + \frac{CX^2}{2} + DX \Big]_{X_0}^{X_3}$$

$$= \frac{A}{4}(X_3 - X_0)^4 + \frac{B}{3}(X_3 - X_0)^3 + \frac{C}{2}(X_3 - X_0)^2 +$$

$$+ D(X_3 - X_0) \qquad (V.9)$$

pero :

$$X_3 - X_0 = 3 \ \Delta \ X$$

substituyendo en (V.9)

$$\int_{X_0}^{X_3} F(x)dx \doteq \frac{A}{4}(3\Delta X)^4 + \frac{B}{3}(3\Delta X)^3 + \frac{C}{2}(3\Delta X)^2 + D(3\Delta X) \qquad (V.10)$$

de la ecuación (V.8) se obtienen A, B, C, D al plantear un sistema de ecuaciones como en el método anterior y esos valores obtenidos se substituyen en (V.10) para obtener la siguiente expresión :

$$A_1 = \int_{X_0}^{X_3} F(x)dx \doteq \frac{3}{8}\Delta X \left[Y_0 + 3Y_1 + 3Y_2 + Y_3 \right]$$

para todo el intervalo se suman las áreas obtenidas llegándose a la siguiente expresión para un intervalo completo :

$$\int_a^b F(x)dx \doteq \frac{3\Delta x}{8}\left[Y_0 + Y_n + 2\sum_{i=2}^{n-1} \text{ord. mult. de 3} + 3\sum_{i=2}^{n-1} \text{resto ord.}\right]$$

Para aplicar el método se requiere que "n" sea múltiplo de 3, donde - n = 0, 1, 2, ... En caso contrario se procede igual que en el método anterior. El error que produce esta fórmula es del orden de Δx^4.

En términos generales, cuando se desee integrar una función con la mayor exactitud posible se debe tratar de aplicar lo más que se pueda los métodos descritos con anterioridad ateniéndose a la siguiente jerarquía :

- Simpson 3/8
- Simpson 1/3
- Trapezoidal.

Ejemplo

Encontrar el área bajo la curva, aplicando los métodos vistos, para los siguientes valores muestrales de una parábola y compararlos con la integral exacta.

n	x	y
0	0	0
1	0.5	0.25
2	1	1
3	1.5	2.25
4	2	4
5	2.5	6.25
6	3	9

$y = x^2$

$\Delta x = 0.5$

Sol.

Aplicando Simpson 3/8 :

$$\int_0^3 f(x)\,dx = \frac{3}{8}(0.5)\left[0 + 9 + 2(2.25) + 3(.25+1+ \right.$$
$$\left. + 4 + 6.25)\right]$$
$$= 9$$

Aplicando Simpson 1/3 :

$$\int_0^3 f(x)\,dx = \frac{0.5}{3}(0 + 9 + 2(1+4) + 4(.25 + 2.25 +$$
$$+ 6.25)$$
$$= 9$$

Aplicando Trapezoidal :

$$\int_0^3 f(x)\,dx = \frac{0.5}{2}\left[0 + 9 + 2(.25 + 1 + 2.25 + \right.$$
$$\left. + 4 + 6.25)\right.$$
$$= 9.125$$

Solución exacta :

$$\int_0^3 f(x)\,dx = \left.\frac{x^3}{3}\right]_0^3 = \frac{27}{3} = 9$$

VI) SOLUCION ECUACIONES DIFERENCIALES ORDINARIAS.

Las ecuaciones diferenciales ordinarias son aquellas en las que la variable dependiente es función de una sola variable independiente :

* $Y^{(n)} = (X, Y, Y', \ldots, Y^{(n-1)})$

a) Método de Euler

Se tratará el caso de ecuaciones diferenciales ordinarias de primer orden :

$$dy = y' dx$$

Substituyendo por los incrementos en la expresión anterior se tiene :

$$\Delta y = y' \Delta X \qquad (VI.0)$$

Tomando un punto inicial para arrancar y conservando un incremento constante Δx se obtiene la siguiente fórmula iterativa :

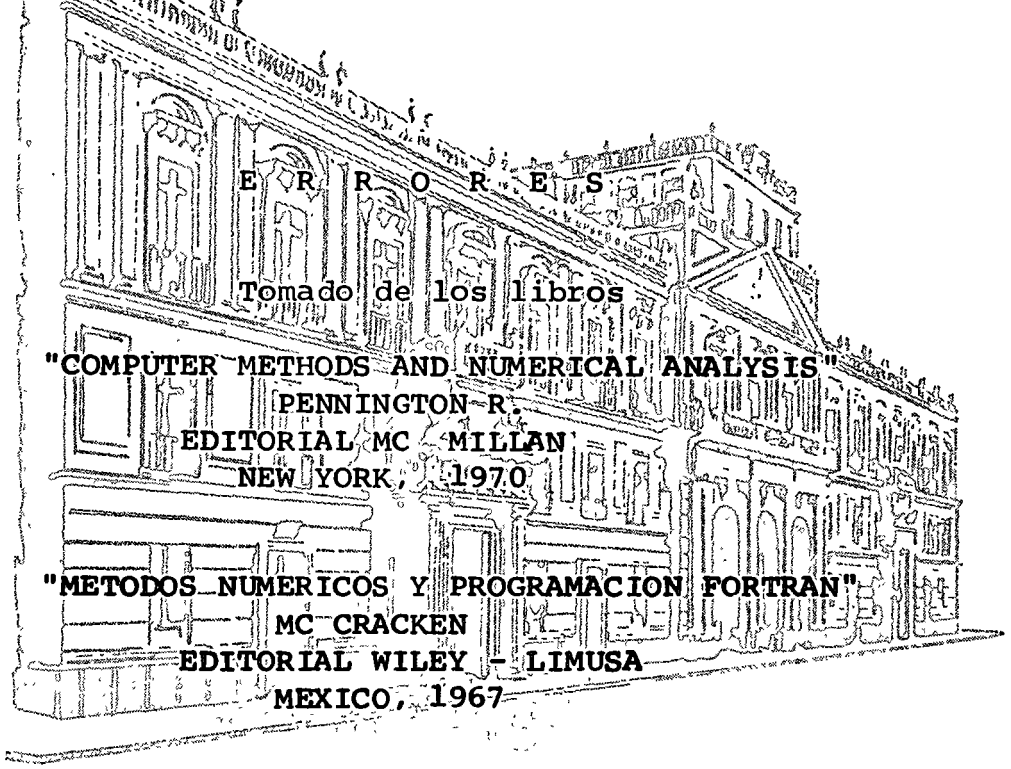
$$\begin{aligned} Y_1 &= Y_o + Y' \,\Big|_{(X_o, Y_o)}^{\Delta x} \\ Y_2 &= Y_1 + Y' \,\Big|_{(X_1, Y_1)}^{\Delta x} \\ &\vdots \\ Y_{n+1} &= Y_n + Y' \,\Big|_{(X_n, Y_n)}^{\Delta x} \end{aligned} \qquad (VI.1)$$

---

* $Y^{(n)} \equiv \dfrac{d^n Y}{dX^n}$

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL



E R R O R E S

Tomado de los libros

"COMPUTER METHODS AND NUMERICAL ANALYSIS"
PENNINGTON R.
EDITORIAL MC MILLAN
NEW YORK, 1970

"METODOS NUMERICOS Y PROGRAMACION FORTRAN"
MC CRACKEN
EDITORIAL WILEY - LIMUSA
MEXICO, 1967

MARCIAL PORTILLA ROBERTSON

ABRIL DE 1976.

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels.: 521-40-23 521-73-35 5123-123

CENTRO DE SANCIONES COMERCIALES

Universidad de Belgrano
Facultad de Ingeniería
Manuel Belgrano

didos entre 1000 cps y 3500 cps en incrementos de 100 cps. Los resultados obtenidos se muestran en la figura 1.12.

Cuando $2\pi FL = 1/(2\pi FC)$, el término en paréntesis en el radical es cero, y se dice que el circuito está en *resonancia*. En los resultados impresos es evidente la presencia de un amplio pico de resonancia correspondiente a una frecuencia resonante de unos 2250 cps.

CAPITULO

2

Errores

2.1 Introducción

El análisis del error en un resultado numérico es fundamental para cualquier computación inteligente, sea hecha a mano o con una computadora. Los datos de entrada rara vez son exactos, ya que a menudo se basan en experimentos o son estimados, y los procesos numéricos a su vez introducen errores de varios tipos. Antes de iniciar nuestro estudio del tema de errores observemos en unos pocos ejemplos cuan importante es. En el ejercicio 18 al final de este capítulo se pide encontrar una lista de las raíces de la ecuación $x^2 + 0.1002x + 0.00003 = 0$, usando aritmética de punto flotante de cuatro dígitos. Utilizando la conocida fórmula

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

obtenemos un resultado de $-0.0001$5. Esta fórmula se presenta usualmente en cursos de álgebra sin ninguna discusión de su precisión, sin embargo, la aritmética de punto flotante de cuatro dígitos introduce errores que hacen el resultado erróneo en 25%, la raíz real, determinada con aritmética de ocho dígitos es -0.0002.

En este caso la culpa fue de la aritmética de cuatro dígitos, pero no se piense que los números de punto flotante de ocho dígitos resolverán todos los problemas. Considérese la serie de Taylor para el seno:

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

La serie se describe usualmente como válida para cualquier ángulo fi- [illegible] error por truncamiento cometido al suspender la serie después [illegible] número finito de términos se dice que es menor en valor al último

que el primer término despreciado. Estos postulados serían verdaderos *si hubiera alguna forma de conservar un número finito de dígitos en cada resultado aritmético.* Veremos en el caso particular 3 del capítulo 3 que la serie es, de hecho, totalmente inútil para ángulos grandes. Supóngase, por ejemplo, que tratamos de valuar el seno de 1470° (= 25.7 radianes, aproximadamente), usando aritmética de punto flotante de ocho dígitos y calculando los términos hasta encontrar uno que sea menor que 10^{-8} en valor absoluto. El resultado calculado será 24.25401855, que aparentemente tiene gran precisión, pero que por supuesto, carece de sentido. Aún si usamos aritmética de punto flotante de 16 dígitos, el seno de 2550° resulta 29.5.

Las dificultades en estos ejemplos se deben a la representación finita de los números. Este no es el único problema. Considérese las dos ecuaciones simultáneas siguientes:

$$5x - 331y = 3.5$$
$$6x - 397y = 5.2$$

Una respuesta "exacta" se determina fácilmente sin problemas del tipo que se encontró anteriormente: $x = 331.7$, $y = 5\,000$. Aparentemente estos resultados contienen cuatro dígitos significativos ¿Los tienen realmente? Veamos primeramente que sucede a las respuestas si la constante de la segunda ecuación se cambia a 5.1, es decir, sufre una variación de un 2%. Se obtiene ahora $x = 298.6$, $y = 4.5$. Esto da que pensar: un cambio de 2% en uno de los datos cambia los resultados en un 10%. Motivo de mayor preocupación es que si sustituimos $x = 358.173$, $y = 5.4$ en las ecuaciones, el valor redondeado de los primeros miembros es exactamente igual a los segundos miembros. Concluimos que los valores calculados de x y y tienen a lo sumo un dígito significativo.

Esto no se debió a la aritmética; todos los resultados eran exactos. El problema radica en la naturaleza de los datos; el determinante del sistema es pequeño, o dicho en forma geométrica, las dos líneas representadas por las ecuaciones son casi paralelas.

Como un ejemplo final, el valor de la integral

$$\int_{e^{-1}}^{1} \frac{dx}{x}$$

se determina exactamente igual a 1. Sin embargo, la integración con la fórmula trapecial, usando 10 intervalos, da un resultado de 5.3. Aún si se usan 40 intervalos obtenemos 4.13, que tiene un error de 3%.

En este caso el problema estriba en la naturaleza del integrando que es muy grande para valores pequeños de x, y en el proceso numérico. Con datos exactos y operaciones exactas obtenemos [illegible]

muy grande debido a la naturaleza de la función y a la técnica numérica empleada.

Sin aumentar más los ejemplos, debe estar claro que sin un análisis de los errores en una computación, realmente no sabemos gran cosa acerca de los resultados. Por supuesto a veces puede ocurrir que con una inspección cuidadosa de las operaciones se puede decir que no habrá problemas especiales; en los dos primeros casos particulares del capítulo 4, por ejemplo, observamos inmediatamente que no existe problema respecto a la precisión de los cálculos Sin embargo, es claro que esto no es cierto siempre.

El material presentado en este capítulo debe resultar interesante y útil en sí mismo para analizar los resultados de operaciones aritméticas simples. Es también fundamental para efectuar el análisis de los errores en los procedimientos numéricos que se van a discutir en los capítulos siguientes. El análisis del error es un punto de partida adecuado para nuestro estudio de métodos numéricos.

2.2 Errores relativos y errores absolutos

Para empezar estableceremos una distinción entre errores relativos y errores absolutos. El *error absoluto* en una cantidad es la diferencia entre el valor verdadero, suponiendo que se conoce, y una aproximación al valor verdadero. La notación ordinaria consiste en indicar el valor aproximado mediante una barra sobre el símbolo de la cantidad, y el error se indica mediante la letra e con un subíndice. Entonces, si x es el valor verdadero, escribiríamos

$$x = \bar{x} + e_x$$

En esta expresión e_x es el error absoluto, que, repetimos, se define como la diferencia entre el valor real y la aproximación:

$$e_x = x - \bar{x}$$

El *error relativo* es el cociente del error absoluto entre la aproximación. Parecería más razonable definirlo como el error absoluto dividido entre el valor *verdadero*, pero generalmente no conocemos éste. Todo lo que tenemos generalmente es un valor aproximado y una [illegible] del error o un *límite* al tamaño máximo del error. Si el error es pequeño, la diferencia en la definición no tiene una influencia muy grande en el valor numérico del error relativo

El error absoluto y el error relativo son aproximadamente iguales para [illegible] cercanos a 1. Para números no cercanos a 1 puede haber una [illegible] Por ejemplo, si tenemos un valor verdadero de 0.000006

y una aproximación de 0.00005, el error absoluto es sólo 10^{-5}, pero el error relativo es 0.2, es decir, 20%. Por otra parte, si tenemos un valor verdadero de 100,500 y una aproximación de 100,000, el error absoluto es 500 pero el error relativo es sólo 0.005, o sea, 0.5%.

Obviamente es necesario en cualquier caso indicar si nos referimos al error absoluto o al relativo, a menos que el significado esté claramente determinado con la notación o con el contenido de la frase.

2.3 Errores inherentes

Existen tres tipos básicos de errores en una computación numérica: inherentes, por truncamiento, y por redondeo. Cada uno se puede expresar en forma absoluta o en forma relativa.

Los *errores inherentes* son errores que existen en los valores de los datos, causados por incertidumbre en las mediciones, por verdaderas equivocaciones, o por la naturaleza necesariamente aproximada de la representación, mediante un número finito de dígitos, de cantidades que no pueden representarse exactamente con el número de dígitos permisible.

Una medición física, tal como una distancia, un voltaje, o un período de tiempo, no puede ser exacta. Si la medición se da con muchos dígitos, tal como un voltaje de 6.4837569, podemos estar seguros de que al menos algunos de los dígitos de la extrema derecha no tienen ningún sentido, porque los voltajes no pueden medirse con esta precisión. Si la medición se da con unos cuantos dígitos, tal como un intervalo de tiempo de 2.3 segundos, podemos estar bastante seguros de que hay algún error inherente, porque sólo accidentalmente el intervalo de tiempo sería de *exactamente* 2.3 seg.* En tales casos podemos conocer algunos límites razonables del error inherente, por ejemplo decir que el tiempo es 2.3 $\pm$ 0.1 seg.

A menudo se supone que cuando se da una medición física sin ninguna declaración referente a la precisión de los dígitos, se entiende que la precisión de esa medición corresponde a media unidad en la última posición. Así si una distancia se da como 5.63 cm, se entendería que no es menor que 5.625 ni mayor que 5.635. Esta convención no es universalmente aceptada. Cuando los límites de precisión son importantes, es mucho mejor indicarlos explícitamente, escribiendo por ejemplo 5.63 $\pm$ 0.005.

Independientemente del número de dígitos usado para representar una cantidad, ésta puede contener un error de cualquier clase. Estos errores pueden variar de

* [illegible]

como copiar mal los datos o leer equivocadamente una escala, a errores "sofisticados" debidos a un entendimiento incompleto de las leyes físicas.

Muchos números no pueden ser representados exactamente en un número dado de dígitos decimales. Si necesitamos usar π en un cálculo, podemos escribirlo como 3.14, 3.14159265, o 3.14159265358979 3. En cualquiera de los casos no tenemos una representación *exacta* de π, que es un número irracional y por lo tanto no tiene una representación decimal exacta finita. En muchos casos aún una fracción simple no tiene representación decimal exacta, por ejemplo $\frac{1}{3}$, que puede escribirse solamente como una sucesión infinita de números 3.

Sucede también que muchas fracciones que tienen una representación finita en algún sistema numérico no la tienen en otro sistema. Por ejemplo, el número $\frac{1}{10}$, cuya representación en el sistema decimal es simplemente 0.1, se representa en el sistema binario en una forma repetitiva infinita, 0.0001100110011001100... Entonces, si se efectúa la suma de 10 números, cada uno de los cuales es una aproximación binaria a la cantidad decimal 0.1, el resultado no será exactamente 1.0. Los que trabajan por primera vez con computadoras binarias han resultado a veces frustrados por su primer encuentro con esta peculiaridad de la naturaleza. El problema es inevitable, su solución no es difícil una vez que se le ha reconocido, como veremos en algunos de los casos particulares.

2.4 Errores por truncamiento

Los errores inherentes son errores en los datos con los que la computadora efectúa algún proceso numérico. Los otros dos tipos de errores, por truncamiento y por redondeo, se refieren a errores debidos a la manera de efectuar los procesos numéricos.

La conocida serie infinita de Taylor

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

se puede usar para calcular el seno de cualquier ángulo x expresado en radianes.* Por supuesto no podemos usar todos los términos de la serie en un cálculo, porque la serie es infinita, de modo que después de calcular un número finito de términos [illegible] Los términos omitidos (que son los [illegible] en los resultados calculados. Este error se [illegible] que es debido al truncamiento de [illegible]

* [illegible]

Muchos de los procedimientos usados en cálculos numéricos son infinitos, así que el problema del error por truncamiento adquiere una gran importancia. Lo discutiremos en detalle en capítulos futuros en relación con el tópico al que se aplica.

2.5 Errores por redondeo

Aún si suponemos que los datos de entrada no tienen errores inherentes y aplicamos procesos de computación que son finitos y no tienen, por tanto, error por truncamiento, podemos introducir otra clase de errores al efectuar aritmética simple: error por redondeo. Supóngase por el momento que tenemos una computadora en la cual cada número contiene cinco dígitos y que deseamos sumar 9.2654 y 7.1625, los que suponemos exactos. La suma es 16.4279, que tiene seis dígitos y por lo tanto no puede ser almacenada en nuestra computadora hipotética. La computadora debe entonces *redondear* el resultado de seis dígitos a 16.428 y al hacerlo introduce un *error por redondeo*. Como el trabajo de una computadora se efectúa con cantidades que tienen algún número fijo de dígitos, la necesidad de redondear ocurre con frecuencia.

El redondeo en FORTRAN es de interés solamente en lo referente a números de punto flotante. En FORTRAN los números de punto fijo son enteros. La suma, resta, o multiplicación de dos enteros es siempre un entero; si un resultado es demasiado grande y no cabe en una dirección de la computadora, se considera que el programa tiene un error; *no* se redondea y se almacenan sólo los dígitos posibles. El cociente de dos enteros no siempre es un entero, así que podría pensarse que el redondeo constituye un problema, pero en la práctica la aritmética de punto fijo no se aplica en forma tal que de ordinario necesitemos redondear un cociente. (En la gran mayoría de los cálculos de ingeniería no se usa la división de punto fijo.)

Como nos interesa primeramente el redondeo de punto flotante, revisaremos brevemente la forma de representación de un número de punto flotante y estableceremos una notación. Recordamos que cada número está representado por una fracción, generalmente llamada mantisa, la cual está multiplicada por una potencia del número base, llamada generalmente el exponente. Tenemos números como los siguientes

$$\begin{array}{ll} .7392 \cdot 10^{4} & (= 7392) \\ .3246 \cdot 10^{2} & (= 32.46) \\ .1627 \cdot 10^{-3} & (= .001627) \end{array}$$

Se dice que un número de punto flotante está normalizado si el primer dígito de la mantisa es diferente de cero. Si un número no está normalizado, se puede normalizar desplazando la mantisa hacia la izquierda y disminuyendo el exponente en una unidad por cada dígito que se desplace. Así, los números

$$\begin{array}{ll} .0001627 \cdot 10^{0} & (= .001627) \\ .0032460 \cdot 10^{4} & (= 32.46) \end{array}$$

no están normalizados. Normalizados, son

$$\begin{array}{ll} .1627 \cdot 10^{-2} & (= .001627) \\ .3246 \cdot 10^{2} & (= 32.46) \end{array}$$

que todos los números de punto flotante han quedado en cierta forma normalizados.

Si representamos la mantisa de un número x de punto flotante mediante la letra f y el exponente mediante la letra e, la forma generalizada de un número de punto flotante (de base decimal) es

$$x = f \cdot 10^{e}$$

El valor de f no puede ser menor que $\frac{1}{10}$, pues suponemos números normalizados, y no puede llegar a ser 1, porque la mantisa es una fracción propia.

El resultado de una operación aritmética consta en general de dos partes, una más significativa y otra menos significativa. Por ejemplo, supóngase que vamos a sumar los dos números de punto flotante siguientes, en una computadora en la que la mantisa tiene cuatro dígitos y el exponente tiene un dígito.

$$\begin{array}{ll} .1624 \cdot 10^{3} & (= 162.4) \\ .1769 \cdot 10^{1} & (= 1.769) \end{array}$$

Dijimos que cuando realiza aritmética de punto flotante la computadora automáticamente se encarga de los problemas de colocación del punto decimal. Lo que esto significa en la suma, es que los exponentes de los dos números son comparados para ver cuantos lugares debe ser desplazada hacia la derecha la mantisa del número que tiene menor exponente, para "alinear" los puntos decimales supuestos. En el ejemplo, el resultado sería

$$\begin{array}{ll} .1624 & \cdot 10^{3} \\ .001769 & \cdot 10^{3} \end{array}$$

En otras palabras, la mantisa del número que tiene menor exponente es desplazada hacia la derecha tantos lugares como indique la diferencia entre los exponentes. Entonces se pueden sumar directamente las dos mantisas.

Obviamente, la mantisa de la suma tiene más de cuatro dígitos. Antes de redondear, el resultado se puede mostrar como dos cantidades de punto flotante:

$$\begin{array}{rl} & .1624 \cdot 10^{3} \\ + & .001769 \cdot 10^{3} \\ \hline & .164169 \cdot 10^{3} = .1641 \cdot 10^{3} + .6900 \cdot 10^{-1} \end{array}$$

Cualquiera de las cuatro operaciones aritméticas producirá un resultado que se puede indicar en general (antes de redondear) en dos mitades en la forma siguiente

$$y = f_y \cdot 10^e + g_y \cdot 10^{e-t}$$

en que f_y tiene t dígitos. Como hemos visto, el intervalo de valores posibles de f_y, es $\frac{1}{10} \leqslant |f_y| < 1$. El caso de g_y es diferente, porque no podemos garantizar que g_y estará normalizada; de hecho, g_y puede ser cero. El intervalo de variación es $0 \leqslant |g_y| < 1$.

Llegamos a dos asuntos de importancia primordial en esta discusión: ¿cómo se va a tomar en consideración g_y para modificar f_y, y para cada caso cuál es el error máximo que resulta en $\bar{y}$?

Generalmente "redondear" implica afectar de alguna manera a f_y, dependiendo del valor de g_y. Sin embargo, una definición más general de redondeo debe incluir el caso en el que g_y se ignora, lo cual significa que f_y nunca se modifica. Esta regla se llama "truncar" el resultado; pero nosotros preferimos llamarla "redondeo truncado", que es una terminología alterna aceptable, para evitar confusión con el error por truncamiento que se comete al considerar sólo una parte de un proceso infinito —lo cual es una cosa enteramente diferente.

Un gran número de los compiladores FORTRAN que se encuentran en operación cuando se está escribiendo este libro preparan el programa objeto de manera que use redondeo truncado. Como veremos más adelante esta clase de redondeo introduce mayor error que la regla más conocida. Por otra parte, el uso de esta última regla de redondeo desperdicia tiempo de computadora si se usa en *cada* operación aritmética incluyendo aquellos lugares del programa en que no es realmente necesaria. Evidentemente muchos diseñadores de compiladores han tomado la decisión económica de que el redondeo truncado no causa problemas tales que justifiquen el costo de una regla más sofisticada de redondeo.

Se puede determinar fácilmente un límite al error relativo máximo que puede ocurrir en un resultado aritmético obtenido con redondeo truncado. El error relativo máximo ocurre cuando g_y es grande y f_y es pequeño. El valor máximo posible de g_y es menor que 1.0; el valor mínimo de f_y es 0.1. Por lo tanto el valor absoluto del error relativo es

$$\left|\frac{e_y}{\bar{y}}\right| = \left|\frac{g_y \cdot 10^{e-t}}{f_y \cdot 10^e}\right| < \frac{1 \cdot 10^{e-t}}{0.1 \cdot 10^e} = 10^{-t+1}$$

Recordando que t es el número de dígitos en la mantisa de *cualquier* número de punto flotante, obtenemos un resultado interesante: el máximo error relativo por redondeo en el resultado de una operación aritmética de punto flotante no depende en ninguna manera del tamaño de las cantidades. Esto nos da un enfoque significativo para el error en cálculos de punto flotante.

El tipo más conocido de redondeo, que se denomina generalmente *redondeo simétrico*, puede describirse como sigue: dadas las dos partes de un resultado como en el caso anterior, la aproximación redondeada a y está dada por

$$|\bar{y}| = \begin{cases} |f_y| \cdot 10^e \text{ si } |g_y| < 1/2 \\ |f_y| \cdot 10^e + 10^{e-t} \text{ si } |g_y| \geqslant 1/2 \end{cases}$$

en que $\bar{y}$ tiene el mismo signo que f_y. La adición de 10^{e-t} en el segundo renglón de la ecuación corresponde a sumar 1 al último dígito retenido si el primer dígito que se pierde es igual o mayor que 5. Se escriben los símbolos de valor absoluto para indicar que las mismas fórmulas se aplican a cantidades positivas y negativas.

Si $g_y < 1/2$, el error absoluto es

$$|e_y| = |g_y| \cdot 10^{e-t}$$

Si $|g_y| \geqslant 1/2$, el error absoluto es

$$|e_y| = |1 - g_y| \cdot 10^{e-t}$$

De cualquier manera, tenemos 10^{e-t} multiplicado por un factor cuyo valor absoluto no es mayor que $1/2$. El valor absoluto del error absoluto es, por lo tanto

$$|e_y| \leqslant 1/2 \cdot 10^{e-t}$$

y el valor absoluto del error relativo es entonces

$$\left|\frac{e_y}{\bar{y}}\right| \leqslant \left|\frac{1/2 \cdot 10^{e-t}}{f_y \cdot 10^e}\right| \leqslant \left|\frac{1/2 \cdot 10^{e-t}}{0.1 \cdot 10^e}\right| = 5 \cdot 10^{-t} = 1/2 \cdot 10^{-t+1}$$

A veces se usa una regla ligeramente más refinada para tomar en consideración el caso en que g_y es exactamente un medio: f_y se deja inalterado si su último dígito es par y se redondea si su último dígito es impar. Esto se hace rara vez porque complica el diseño y la operación de la computadora.

En adelante supondremos que la regla adecuada para redondeo simétrico es aquella con la que estamos más familiarizados, sin ninguna provisión especial para el caso en que $g_y = 1/2$.

Para un ejemplo de la diferencia entre las dos reglas de redondeo, considérese el siguiente resultado de alguna operación aritmética:

$$y = .7324 \cdot 10^3 + .8261 \cdot 10^{-1}$$

Para "redondeo truncado"

$$\bar{y} = .7324 \cdot 10^3$$

y

$$\left|\frac{e_y}{\bar{y}}\right| = \frac{.8261 \cdot 10^{-1}}{.7324 \cdot 10^3} \simeq 1.1 \cdot 10^{-4}$$

($\simeq$ significa "aproximadamente igual a").

Para la operación que llamamos redondeo simétrico,

$$\bar{y} = .7325 \cdot 10^3$$

$$e_y = -.1739 \cdot 10^{-1}$$

y

$$\left|\frac{e_y}{\bar{y}}\right| = \frac{.1739 \cdot 10^{-1}}{.7325 \cdot 10^3} \simeq .24 \cdot 10^{-4}$$

Entonces en este ejemplo el error por redondeo simétrico es considerablemente menor que el error por redondeo truncado. El error por redondeo simétrico nunca excede al error por redondeo truncado y la mitad de las veces es menor que éste.

Ninguno de los errores es tan grande como su límite superior correspondiente, que es 10.10^{-4} para el truncado y $5 \cdot 10^{-4}$ para el simétrico. Y puede suceder, por circunstancias especiales o buena suerte, que el error por redondeo sea cero. La situación típica es que conocemos un *límite* al tamaño del error en un cálculo, pero no el error *real*. Para estar del lado de la seguridad, supondremos siempre lo peor, es decir, que el error pudiera ser tan grande como su límite. Un camino más satisfactorio sería tomar alguna clase de error "promedio" y usar técnicas estadísticas para encontrar el valor más probable del error total en una computación. Sin embargo, tales técnicas están más allá del alcance de este libro.

Estos resultados han sido establecidos en términos de números flotantes *decimales*. Muchas computadoras para cálculos científicos operan en sistema flotante *binario*, es decir, con números de base 2 en lugar de números de base 10. En dicho sistema, cada número de punto flotante se representa como una fracción (expresada por supuesto en sistema binario) multiplicada por una potencia de 2:

$$\bar{x} = f \cdot 2^e \qquad {}^1/_2 \leqslant |f| < 1$$

Un análisis similar al efectuado previamente conduce a un límite en el error relativo de $2 \cdot 2^{-t}$ para redondeo truncado y 2^{-t} para redondeo simétrico.

Algunas computadoras son *hexadecimales*, es decir, trabajan con números de base 16. En ellas los límites al error relativo son $16 \cdot 16^{-t}$ para redondeo truncado y $8 \cdot 16^{-t}$ para redondeo simétrico.

En lo que sigue trataremos el error por redondeo en cantidades de punto flotante en términos de números decimales para poder trabajar con un sistema que nos es familiar. Debe notarse, sin embargo, que los resultados serán ligeramente diferentes en otros sistemas.

2.6 Propagación del error

De mucha importancia en análisis numérico es la forma en que un error en algún punto de una computación *se propaga*, es decir, determinar si su efecto aumenta o disminuye al efectuar operaciones subsiguientes. La resta de dos cantidades aproximadamente iguales es un caso extremo: aunque los dos números tengan errores pequeños, el error relativo en la diferencia puede ser muy grande. Este error relativo grande será propagado por operaciones aritméticas posteriores.

Nuestro primer paso en este importantísimo estudio es encontrar expresiones para el error absoluto y el error relativo en el resultado de cada una de las cuatro operaciones aritméticas en función de los operandos y sus errores. Después, en la sección siguiente desarrollaremos una técnica para determinar un límite al error total en un cálculo que contenga un número cualquiera de operaciones aritméticas.

Suma

Se tienen dos aproximaciones, $\bar{x}$ y $\bar{y}$, a dos valores verdaderos, x y y, junto con sus errores respectivos, e_x y e_y. Tendremos entonces

$$x + y = \bar{x} + e_x + \bar{y} + e_y = (\bar{x} + \bar{y}) + (e_x + e_y)$$

El error en la suma, que indicaremos mediante e_{x+y}, es por tanto

$$e_{x+y} = e_x + e_y$$

Resta

De una manera semejante obtenemos

$$e_{x-y} = e_x - e_y$$

Multiplicación

En este caso se tiene

$$\begin{aligned} x \cdot y &= (\bar{x} + e_x) \cdot (\bar{y} + e_y) \\ &= \bar{x}\bar{y} + \bar{x}e_y + \bar{y}e_x + e_x e_y \end{aligned}$$

Suponemos que los errores son mucho más pequeños que las aproximaciones, e ignoraremos el producto de los errores. Entonces

$$x \cdot y \simeq \bar{x}\bar{y} + \bar{x}e_y + \bar{y}e_x$$

$$e_{x \cdot y} \simeq \bar{x}e_y + \bar{y}e_x$$

División

Tenemos

$$\frac{x}{y} = \frac{\bar{x} + e_x}{\bar{y} + e_y}$$

Multiplicando el denominador por $\bar{y}/\bar{y}$ y reacomodando términos obtenemos

$$\frac{x}{y} = \frac{\bar{x} + e_x}{\bar{y}} \left(\frac{1}{1 + e_y/\bar{y}} \right)$$

El factor en paréntesis puede desarrollarse en serie mediante una división:

$$\frac{x}{y} = \frac{\bar{x} + e_x}{\bar{y}} \cdot \left(1 - \frac{e_y}{\bar{y}} + \left(\frac{e_y}{\bar{y}} \right)^2 - \cdots \right)$$

Efectuando la multiplicación y despreciando todos los términos que contienen productos o potencias de orden superior al primero de e_x y e_y, tenemos

$$\frac{x}{y} \simeq \frac{\bar{x}}{\bar{y}} + \frac{e_x}{\bar{y}} - \frac{\bar{x}}{\bar{y}^2} e_y$$

Por lo tanto

$$e_{x/y} \simeq \frac{1}{\bar{y}} e_x - \frac{\bar{x}}{\bar{y}^2} e_y$$

Para un ejemplo simple del significado de estas fórmulas, considérese la suma de dos logaritmos de cuatro cifras. Como podemos suponer que los logaritmos están correctos hasta la cuarta cifra, sabemos que el error en cada uno no es mayor que 0.00005. El error en la suma no puede ser mayor que 0.0001. Naturalmente, no sabemos que sea *tan* grande, sino que *podría serlo*.

Debe observarse que rara vez conocemos el signo de un error. Por ejemplo, no se debe inferir que la suma incrementa siempre el error y que la resta siempre lo disminuye simplemente porque los errores se suman en la adición y se restan en la substracción. Si los errores tienen signos diferentes ocurrirá precisamente lo contrario.

Como tenemos ahora fórmulas para la propagación de errores absolutos en las cuatro operaciones aritméticas básicas, podemos usarlas de dar y obtener los errores relativos. Para la suma y la resta

tados han sido reacomodados para mostrar explícitamente el efecto de los errores en los operandos.

Suma

$$\frac{e_{x+y}}{\bar{x} + \bar{y}} = \frac{\bar{x}}{\bar{x} + \bar{y}} \left(\frac{e_x}{\bar{x}} \right) + \frac{\bar{y}}{\bar{x} + \bar{y}} \left(\frac{e_y}{\bar{y}} \right)$$

Resta

$$\frac{e_{x-y}}{\bar{x} - \bar{y}} = \frac{\bar{x}}{\bar{x} - \bar{y}} \left(\frac{e_x}{\bar{x}} \right) - \frac{\bar{y}}{\bar{x} - \bar{y}} \left(\frac{e_y}{\bar{y}} \right)$$

Multiplicación

$$\frac{e_{x\,y}}{\bar{x} \cdot \bar{y}} = \frac{e_x}{\bar{x}} + \frac{e_y}{\bar{y}}$$

División

$$\frac{e_{x/y}}{\bar{x}/\bar{y}} = \frac{e_x}{\bar{x}} - \frac{e_y}{\bar{y}}$$

Es importante comprender claramente el significado de estas fórmulas de propagación. Partimos de dos valores aproximados, $\bar{x}$ y $\bar{y}$, que contienen los errores e_x y e_y. Los errores pueden ser de cualquier tipo. Los valores de $\bar{x}$ y $\bar{y}$ pueden ser resultados experimentales que contienen errores inherentes; pueden ser el resultado de algún cálculo previo efectuado mediante un proceso infinito y por lo tanto pueden contener errores por truncamiento; pueden ser resultado de operaciones aritméticas previas y por tanto contener errores por redondeo. También pueden con suma facilidad ser una combinación de los tres tipos que se han enumerado.

Entonces las fórmulas anteriores dan el error en el resultado de cada una de las operaciones aritméticas en función de $\bar{x}$, $\bar{y}$, e_x y e_y *suponiendo que no hay error por redondeo* en la operación. Si como ocurre con frecuencia queremos saber ahora como se propaga el error en este resultado a otras operaciones aritméticas, *debemos agregar explícitamente el error por redondeo*.

A menudo escribimos x sin la barra superior, aunque para ser completamente precisos debería escribirse la barra. Del texto puede inferirse en la mayoría de los casos si se trata de una aproximación y no del verdadero valor.

La situación puede aclararse con un ejemplo. Supongase que empezamos una computación con tres cantidades, x, y, y z, y por simplicidad consideramos que son exactas, es decir, que no tienen errores de ninguna clase. Supóngase que calculamos

$$u = (x + y) \cdot z$$

Por la forma en que se escribió la expresión, debe efectuarse primero la suma. Se supuso que ambos operandos no tienen error, así que el error *propagado* por la suma es cero; sin embargo, al efectuar ésta se introduce un error por redondeo. Este error por redondeo puede considerarse como un error inherente en la suma cuando procedemos a ejecutar la multiplicación. Si acordamos llamar e_{x+y} al error total en la suma, incluyendo cualquier error propagado y el redondeo, se tiene

$$\left| \frac{e_{x+y}}{x + y} \right| \leq 5 \cdot 10^{-t}$$

que es simplemente el límite en el error por redondeo en cualquier operación aritmética, suponiendo siempre redondeo simétrico. Nuevamente estamos suponiendo una computadora en la que los números de punto flotante tiene una parte fraccionaria que consta de t dígitos decimales.

Sabemos que el error relativo en un producto es la suma de los errores relativos de los dos factores, mas el error por redondeo que se introduce en la multiplicación. Como el resultado de la multiplicación es u, que es nuestra aproximación a u, podemos escribir

$$\frac{e_u}{u} = \frac{e_{x+y}}{x + y} + \frac{e_z}{z} + r_m$$

en que e_z/z es el error relativo en z, y r_m es el error por redondeo en la multiplicación. Como supusimos nulo el error en z, y como

$$\left| \frac{e_u}{u} \right| = \left| \frac{e_{x+y}}{x + y} + r_m \right| \leq \left| \frac{e_{x+y}}{x + y} \right| + |r_m|$$

(La última desigualdad se denomina la *desigualdad triangular:* la igualdad se cumple si $e_{x+y}/(x + y)$ y r_m tienen signos iguales, y la desigualdad si tienen signos diferentes.) Entonces tenemos

$$\left| \frac{e_u}{u} \right| \leq 5 \cdot 10^{-t} + 5 \cdot 10^{-t}$$

Como al final del cálculo conocemos u, podemos fácilmente obtener el límite del error absoluto:

$$|e_u| \leq \bar{u} \cdot 10^{-t+1}$$

2.7 Gráficas de procesos

Tenemos expresiones que nos permiten conocer la propagación de los errores que existen en los operandos de las operaciones aritméticas y vimos en un ejemplo la manera de determinar el error total en una

tación. Necesitamos ahora una forma más conveniente de manejar el problema de la propagación de los errores en un cálculo completo.

Una *gráfica de procesos* * es una representación pictórica de la secuencia en la que se efectúan las operaciones aritméticas en una computación, y un esquema para identificar las flechas que aparecen en la gráfica de manera que sea fácil determinar el error total en el resultado final. El método también facilita determinar la contribución al error total de un error cualquiera en cualquier lugar de la secuencia.

La figura 2.1 es la gráfica de proceso del ejemplo de la sección precedente, $u = (x + y) \cdot z$. Una gráfica de proceso debe leerse de abajo hacia arriba, siguiendo las flechas. Primero se efectúan todas las operaciones en un nivel horizontal dado, después todas las operaciones del nivel superior siguiente, y así sucesivamente. En la figura 2.1 se ve explícitamente que la suma de x y y se efectúa primero, y que el resultado se multiplica por z.

Hasta el momento tenemos sólo una representación pictórica del orden de las operaciones aritméticas, lo cual es interesante pero no es el propósito principal. Agregamos ahora identificaciones a cada una de las flechas, de acuerdo con las reglas siguientes, para indicar la manera en que se propagan los errores.

Suma

Considérese que las dos flechas que llegan a un círculo de adición provienen de dos círculos cuyos resultados son a_1 y a_2. (Estos "resultados" pueden en efecto ser el resultado de otras operaciones, o pueden ser datos de entrada como en nuestro caso.) La flecha que va de a_1 a $\oplus$ se identifica con la etiqueta $a_1/(a_1 + a_2)$ y la flecha que va de a_2 a $\oplus$ con la etiqueta $a_2/(a_1 + a_2)$.

Resta

Si la operación es $a_1 - a_2$, las flechas correspondientes pueden identificarse como $a_1/(a_1 - a_2)$ y $-a_2/(a_1 - a_2)$.

Multiplicación

Las dos flechas que conducen a una multiplicación llevan la identificación + 1.

* El uso de las gráficas de procesos para este fin fue sugerido primeramente a los autores por el Dr. Kenneth M. King de la Universidad Columbia.

División

Si la división es a_1/a_2, la flecha que va de a_1 a (/) se identifica con + 1, y la flecha que va de a_2 a (/) lleva la identificación — 1.

El objeto de todo esto aparece en la regla siguiente: *El error relativo en el resultado de cualquier operación (círculo) aparece en el resultado de la siguiente operación multiplicado por el término que identifica la flecha que une ambas operaciones.*

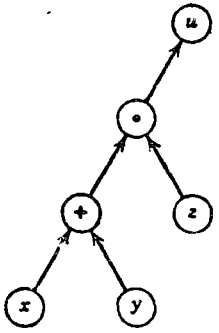


Fig. 2.1 Gráfica de proceso de la operación $u = (x + y) \cdot z$.

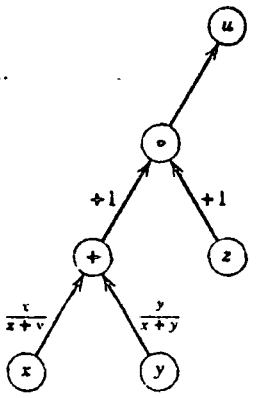


Fig. 2.2 Gráfica de proceso de la figura 2.1, con las flechas identificadas para indicar la propagación de los errores.

Por ejemplo, considérese la figura 2.2, que es igual a la figura 2.1, pero con las flechas debidamente identificadas.

Supongamos ahora que las tres cantidades de la figura 2.1 tienen errores inherentes relativos por redondeo llamados i_x, i_y, e i_z, y veamos cómo se aplica la regla. Considérese primeramente la suma. Tenemos un error relativo i_x en la cantidad x, éste aparece en el resultado de la operación siguiente (la suma) multiplicado por el término que identifica la flecha que une (x) con ⊕:

$$\frac{x}{x+y}\, i_x$$

Hemos omitido las barras en x y en y, pero debe sobrentenderse que éstas son aproximaciones a los valores verdaderos. De la misma manera el error en y, i_y, aparece en el resultado de la operación anterior multiplicado por el término que identifica la flecha que une (y) con ⊕:

Hay finalmente un error por redondeo en la suma, al que llamamos r_1, y el error relativo total en el resultado de la adición es

$$\frac{e_{x+y}}{x+y} = \frac{x}{x+y}\, i_x + \frac{y}{x+y}\, i_y + r_1$$

La regla puede aplicarse ahora a la multiplicación. Uno de los factores es la suma de x y y, que tiene un error que se acaba de indicar; éste aparece como error inherente en el resultado de la multiplicación, de acuerdo con la regla, multiplicado por + 1. El error inherente por redondeo en z, i_z, aparece en el resultado de la multiplicación multiplicado también por + 1. La multiplicación tendrá un error por redondeo que llamamos r_2, y el error total después de efectuar la multiplicación, que es el error total en u, es

$$\frac{e_u}{u} = \frac{x}{x+y}\, i_x \cdot 1 + \frac{y}{x+y}\, i_y \cdot 1 + r_1 \cdot 1 + i_z \cdot 1 + r_2$$

Si todos los resultados están correctamente redondeados (de acuerdo con el método de redondeo convenido), ninguno de los errores por redondeo será mayor que $5 \cdot 10^{-t}$. Entonces tenemos

$$\left|\frac{e_u}{u}\right| \leq \left(\left|\frac{x}{x+y}\right| + \left|\frac{y}{x+y}\right| + 3\right) \cdot 5 \cdot 10^{-t}$$

Si tanto x como y son no negativos, entonces

$$\left|\frac{x}{x+y}\right| + \left|\frac{y}{x+y}\right|$$

no puede ser mayor que 1, y finalmente tenemos

$$\left|\frac{e_u}{u}\right| \leqslant 20 \cdot 10^{-t} = 2 \cdot 10^{-t+1}$$

2.8 Ejemplos

Apliquemos ahora la técnica de gráficas de procesos a tres ejemplos, para ver lo que significa la propagación del error en términos de computaciones prácticas. Las conclusiones que derivemos serán directamente utilizables en muchas situaciones de los capítulos siguientes. Estos ejemplos también ilustran de una manera atractiva los problemas especiales que resultan de trabajar con una computadora digital, en especial los [illegible] resultados no son lo que esperaríamos de acuerdo con nuestro conocimiento en matemáticas clásicas.

Ejemplo 1

Adición de números positivos acomodados en orden ascendente.

Considérese el problema de sumar cuatro números positivos:

$$y = x_1 + x_2 + x_3 + x_4$$

en que

$$0 < x_1 < x_2 < x_3 < x_4$$

La gráfica de proceso se muestra en la figura 2 3. Supongamos que no hay errores inherentes en las x_i, y sean r_1, r_2, y r_3 los errores relativos por redondeo en cada una de las operaciones de abajo hacia arriba. La aplicación sistemática de la regla para determinar el error total en una gráfica de proceso da

$$\frac{e_y}{y} = r_1 \frac{x_1 + x_2}{x_1 + x_2 + x_3} \cdot \frac{x_1 + x_2 + x_3}{x_1 + x_2 + x_3 + x_4} + r_2 \frac{x_1 + x_2 + x_3}{x_1 + x_2 + x_3 + x_4} + r_3$$

Cancelando la suma $x_1 + x_2 + x_3$ en el primer término y multiplicando toda la ecuación por $y = x_1 + x_2 + x_3 + x_4$ nos da

$$e_y = r_1(x_1 + x_2) + r_2(x_1 + x_2 + x_3) + r_3(x_1 + x_2 + x_3 + x_4)$$

Multiplicando y reacomodando los términos se obtiene

$$|e_y| \leqslant (3x_1 + 3x_2 + 2x_3 + x_4) \cdot 5 \cdot 10^{-t}$$

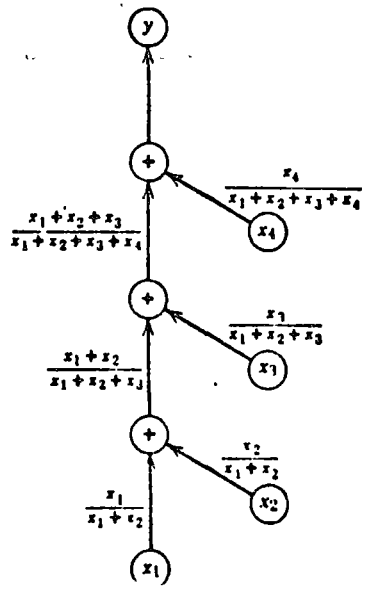


Fig 2.3 Gráfica de proceso para la adición $x_1 + x_2 + x_3 + x_4$, en que $0 < x_1 < x_2 < x_3 < x_4$

Es obvio que el límite en el error total (absoluto o relativo) debido a redondeo se minimiza reacomodando los términos de manera que los números más pequeños sean los que primeramente se sumen.

Este resultado es un poco sorprendente, ya que todo nuestro entrenamiento matemático ha estado basado en la suposición —a menudo implícita— de que la suma es asociativa y conmutativa. La diferencia se debe a que no estamos operando con precisión infinita, que tácitamente se supone en las matemáticas clásicas. Cada resultado en una computadora debe expresarse en un número finito de dígitos y esta restricción aparentemente sencilla [illegible] bía completamente muchas [illegible] matemáticas "estándar".

La fórmula para el límite del error total en la suma de n números que no tienen errores inherentes es

$$|e_y| \leqslant [(n-1)x_1 + (n-1)x_2 + (n-2)x_3 + \ldots + 2x_{n-1} + x_n] \cdot 5 \cdot 10^{-t}$$

Como un ejemplo numérico supóngase que necesitamos efectuar las sumas de los siguientes números:

$$\begin{array}{l} 0.2897 \cdot 10^0 \\ 0.1976 \cdot 10^0 \\ 0.2488 \cdot 10^1 \\ 0.7259 \cdot 10^1 \\ 0.1638 \cdot 10^2 \\ 0.6249 \cdot 10^2 \\ 0.2162 \cdot 10^3 \\ 0.5233 \cdot 10^3 \\ 0.1103 \cdot 10^4 \\ 0.5291 \cdot 10^4 \end{array}$$

Si sumamos en orden ascendente, las sumas parciales sucesivas son las siguientes (La primera suma parcial es la suma de los dos primeros números, la segunda suma parcial es la suma de la primera suma parcial y el tercer número, etc.) Téngase en mente que estamos suponiendo una computadora en la que cada mantisa tiene cuatro dígitos, cada suma parcial que excede cuatro dígitos debe ser redondeada. Este hecho, por supuesto, es básico para la explicación total, aunque ocho dígitos serían más usuales en números de computadora.

$$\begin{array}{l} 0.7873 \cdot 10^0 \\ 0.3275 \cdot 10^1 \\ 0.1053 \cdot 10^2 \\ 0.2691 \cdot 10^2 \\ 0.8940 \cdot 10^2 \\ 0.3056 \cdot 10^3 \\ 0.8289 \cdot 10^3 \\ 0.2232 \cdot 10^4 \\ 0.7523 \cdot 10^4 \end{array}$$

Si por otra parte sumamos los números en el orden inverso, de mayor a menor, las sumas parciales son

$$\begin{array}{l} 0.6694 \cdot 10^4 \\ 0.7217 \cdot 10^4 \\ 0.7433 \cdot 10^4 \\ 0.7495 \cdot 10^4 \\ 0.7511 \cdot 10^4 \\ 0.7518 \cdot 10^4 \\ 0.7520 \cdot 10^4 \\ 0.7520 \cdot 10^4 \\ 0.7520 \cdot 10^4 \end{array}$$

La suma correcta a ocho cifras se puede encontrar conservando todos los dígitos en cada suma. Es $0.75229013 \cdot 10^4$. Entonces el error en la suma ascendente es $-0.1 \cdot 10^0$, mientras que el error en la suma descendente es $2.9 \cdot 10^0$, que es aproximadamente 30 veces mayor.

Los límites en los errores son del orden de $5.5 \cdot 10^0$ para la suma ascendente y $33 \cdot 10^0$ para la descendente. En ambos casos los errores actuales son considerablemente menores que el error máximo posible. El error máximo, dado por los límites, ocurre cuando el redondeo de cada suma parcial requiere despreciar una parte de menor significado que es aproximadamente 1/2, lo cual ocurre raras veces.

Nótese que si se descartan los dos números más pequeños la suma ascendente se convierte en $0.7522 \cdot 10^4$, que es ligeramente diferente, pero la suma descendente permanece inalterada como $0.7520 \cdot 10^4$. Lo que sucede es que los dos números más pequeños en la suma descendente son demasiado pequeños para afectar el último dígito de la suma parcial cuando se agregan separadamente. En la suma ascendente, por otra parte, son sumados primero, y su suma es lo suficientemente grande para afectar el último dígito de las sumas parciales mayores.

Ejemplo 2

Adición de cuatro números positivos aproximadamente iguales. Supóngase que estamos sumando cuatro números positivos, pero que ahora son aproximadamente iguales. Podemos escribir

$$x_i = x_0 + \delta_i, \qquad i = 1, 2, 3, 4$$

en que

$$|\delta_i| \ll x_0$$

(El símbolo $\ll$ significa "es *mucho* menor que".) La aplicación directa de la regla al resultado de la suma de cuatro números da

$$|e_y| \leqslant (9x_0 + 3|\delta_1| + 3|\delta_2| + 2|\delta_3| + |\delta_4|) \cdot 5 \cdot 10^{-t}$$

Como $|\delta_i|$ es pequeña comparada con x_0, tenemos aproximadamente

$$|e_y| \leqslant 4.5 \cdot 10^{-t+1} \cdot x_0$$

Este resultado se basa en calcular las sumas parciales como se indica en la gráfica de proceso de la figura 2.3. Considérese una forma alterna de efectuar la suma, como se indica en la gráfica de proceso de la figura 2.4. En ella $y = (x_1 + x_2) + (x_3 + x_4)$, efectuando primero las operaciones encerradas en paréntesis.

Si llamamos r_1, r_2 y r_3 a los errores por redondeo en [illegible] con los subíndices indicando el orden en que se efectúan [illegible]

$$\frac{e_y}{y} = r_3 \cdot \frac{x_1 + x_2}{x_1 + x_2 + x_3 + x_4} + r_2 \cdot \frac{x_3 + x_4}{x_1 + x_2 + x_3 + x_4} + r_3$$

Reacomodando, tenemos

$$|e_y| \leqslant (2x_1 + 2x_2 + 2x_3 + 2x_4) \cdot 5 \cdot 10^{-t}$$

Haciendo nuevamente $x_i = x_0 + \delta_i$ y despreciando los términos en $|\delta_i|$ comparados con la x_0, obtenemos finalmente

$$|e_y| \leqslant 4 \cdot 10^{-t+1} \cdot x_0$$

Comparando con el límite del error para la gráfica de proceso de la figura 2.3, vemos que este arreglo da un límite ligeramente menor, lo cual no es intuitivamente obvio.

En general, si deseamos sumar n^2 números positivos de aproximadamente igual magnitud, el error total por redondeo se reduce si se suman

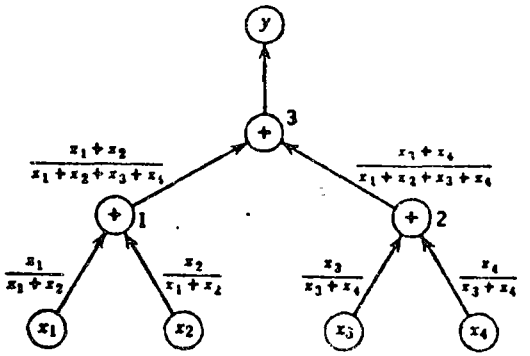


Fig. 2.4 Diferente gráfica de proceso para la suma de cuatro números. Los números cercanos a los círculos de adición indican la secuencia de las adiciones.

en n grupos de n elementos cada uno y después se suman las n sumas parciales. Para un valor grande de n, el límite en el error es sólo $1/n$ del límite correspondiente a la suma de los n^2 términos en una sola "faja" (ver figura 2.3).

Como ejemplo numérico considérense estos cuatro números:

$$x_1 = 0.5243 \cdot 10^0$$
$$x_2 = 0.5262 \cdot 10^0$$
$$x_3 = 0.5226 \cdot 10^0$$
$$x_4 = 0.5278 \cdot 10^0$$

Podemos hacer $x_0 = 0.5200$ y $t = 4$ como antes. Sumando uno a continuación del otro y redondeando correctamente en cada adición obtenemos $y = 0.2102 \cdot 10^1$. Sumando separadamente $x_1 + x_2 = 0.1051 \cdot 10^1$ y

$x_3 + x_4 = 0.1050 \cdot 10^1$, obtenemos $y = 0.2101 \cdot 10^1$. La suma exacta $0.21009 \cdot 10^1$.

El programador inexperto pudiera preguntarse si valen la pena estas pequeñas mejoras. Debe tenerse en mente siempre que estamos presentando ejemplos que requieren sólo unas cuantas operaciones. Más adelante veremos procesos que requieren cientos y a veces hasta miles de operaciones aritméticas; en estas situaciones más reales un error pequeño puede multiplicarse considerablemente en operaciones futuras. Lo que estamos discutiendo tiene entonces importancia práctica bien definida.

Ejemplo 3

Substracción de dos números aproximadamente iguales. Supóngase que tenemos $z = x - y$. De las fórmulas de la página 69 se obtiene entonces

$$\frac{e_z}{z} = \frac{x}{x-y}\left(\frac{e_x}{x}\right) - \frac{y}{x-y}\left(\frac{e_y}{y}\right)$$

Supóngase ahora que x y y son números positivos correctamente redondeados, de manera que

$$\left|\frac{e_x}{x}\right| \leqslant 5 \cdot 10^{-t} \quad y \quad \left|\frac{e_y}{y}\right| \leqslant 5 \cdot 10^{-t}$$

Si $x - y$ es pequeño, el error relativo en z puede ser grande, aunque el error absoluto sea pequeño. Como los errores relativos son los que se propagan en computaciones de punto flotante, esto puede tener un efecto drástico en los resultados finales.

Como ejemplo simple supóngase que tenemos

$$x = 0.5628 \cdot 10^1$$

$$y = 0.5631 \cdot 10^1$$

Entonces

$$z = -0.0003 \cdot 10^1$$

Como conocemos x y y, podemos escribir

$$\left|\frac{e_x}{x}\right| \leqslant \frac{0.5}{0.5628} \cdot 10^{-4} \leqslant 10^{-4} = 0.01\%$$

$$\left|\frac{e_y}{y}\right| \leqslant \frac{0.5}{0.5631} \cdot 10^{-4} \leqslant 10^{-4} = 0.01\%$$

que son errores relativos pequeños. Sin embargo

$$\left|\frac{e_z}{z}\right| \leqslant \frac{10^1}{3} \cdot 10^{-4} = \frac{1}{3} \quad 33\%$$

que es un error relativo grande. Este error relativo grande en $x - y$ se propaga a través de todas las computaciones que siguen. Si la siguiente operación fuera multiplicar por $0.7259 \cdot 10^1$, e imprimiéramos el resultado, éste sería $0.2178 \cdot 10^1$ que aparentemente tiene una precisión de cuatro dígitos. Sin embargo, sólo uno de los dígitos es correcto.

2.9 Lista de recomendaciones para lograr mayor precisión

Algunas de las ideas presentadas en este capítulo pueden resumirse en una pequeña lista de sugestiones para computaciones prácticas. Algunos de los ejercicios que siguen ilustran estos puntos y se indica en la lista la referencia correspondiente a ellos.

1. Cuando se van a sumar y/o restar números, trabajar siempre con los números más pequeños primero (Ejercicio 13).
2. De ser posible, evitar la substracción de dos números aproximadamente iguales. Una expresión que contenga dicha substracción puede a menudo ser reescrita para evitarla (Ejercicios 12, 14 y 18).
3. Una expresión del tipo $a(b - c)$ puede reescribirse en la forma $ab - ac$, y $(a - b)/c$ puede reescribirse como $a/c - b/c$. Si hay números aproximadamente iguales en el paréntesis, *ejecutar la resta antes que la multiplicación*. Esto evitará complicar el problema con errores de redondeo adicionales (Ejercicios 16 y 17).
4. Cuando no se aplica ninguna de las reglas anteriores, minimizar el número de operaciones aritméticas (Ejercicios 6 y 7).

Ejercicios

*1. La corriente pasa a través de una resistencia de 10 ohmios cuya precisión está dentro del 10%. La corriente es 2.0 amperios, medida con una aproximación de ± 0.1 amp. Según la ley de Ohm, la caída de potencial a través de la resistencia es el producto de la resistencia por la corriente. ¿Cuales son los errores relativo y absoluto en el voltaje calculado? Despreciar errores por redondeo.

2. La distancia aérea media entre Nueva York y San Francisco es 4300 kilómetros, pero puede ser 320 kilómetros más larga o más corta debido a variaciones en la ruta. La velocidad de crucero de un avión dado es 960 kph, pero puede variar hasta en 100 kph en exceso o en defecto a causa de los vientos. ¿Cuales son los límites mínimo y máximo de duración del vuelo?

3. La reactancia de un condensador está dada por

$$X_c = \frac{1}{2\pi f C}$$

en que X_c = reactancia capacitiva, ohmios
f = frecuencia, ciclos por segundo
C = capacitancia, faradios

* ¿Cuales son los límites de variación de X_c para $f = 400 \pm 1$ cps y $C = 10^{-6}$ faradios ± 10%?

4. La posición S de un cuerpo que cae libremente en el vacío está dada por

$$S = \tfrac{1}{2}gt^2$$

en que g = aceleración de la gravedad, metros/seg^2
t = tiempo de caída, seg.

Supóngase que g es exactamente 9.81 m/seg^2, pero que t puede medirse solamente dentro de $\pm$ 0.1 seg. Demuestre que al aumentar t, aumenta el error absoluto en el valor calculado de S, pero disminuye el error relativo.

*5. Supóngase que a es un número positivo propiamente redondeado, y que el número 2 puede representarse exactamente en una computadora. Dibuje gráficas de procesos y determine los límites en los errores relativos máximos para demostrar que son iguales para $u = a + a$ y para $v = 2a$.

*6. Con las mismas hipótesis que en el ejercicio 5, demuestre que el límite en el máximo error relativo para $u = a + a + a$ es mayor que para $v = 3a$. Ilustre esto con $a = 0.6992$, conservando sólo cuatro dígitos después de cada operación aritmética.

7. Supóngase que a y b son números positivos propiamente redondeados. Dibuje gráficas de procesos y derive las expresiones para el límite del error, con las que se demuestre que el límite en el error relativo de $u = 3(ab)$ es menor que el que corresponde a $v = (a + a + a)b$. Ilústrelo para $a = 0.4299$ y $b = 0.6824$.

*8. Supóngase que x es un número correctamente redondeado. Dibuje gráficas de procesos y derive expresiones para el límite del error con las que se demuestre que $u = x \cdot (x \cdot (x \cdot x))$ y $v = (x^2)^2$ tienen el mismo límite de error.

9. Supóngase que x es un número propiamente redondeado. Dibuje gráficas de procesos y derive expresiones para el límite del error con las que se demuestre que $u = x \cdot (x \cdot (x \cdot (x \cdot (x \cdot (x \cdot x))))))$ y $v = ((x^2)^2)^2$ tienen el mismo límite de error.

10. Demuestre que en decimal flotante 10./10. = 10 *(1./10.) y 2./2. = 2 *(1./2.) pero 3./3. $\neq$ 3.*(1./3.).

*11. Supóngase que a, b, y x son números positivos exactos. Dibuje gráficas de procesos y derive expresiones para el límite del error para demostrar que los límites del error relativo por redondeo para $u = ax + bx^2$ y para $v = x(a + bx)$ son iguales. Use $a = 0.7625$, $b = 0.6917$, y $x = 0.4302$ para demostrar que aunque los límites son iguales, los errores reales, que generalmente son menores que los límites, no tienen que ser iguales.

12. Supóngase que a y b son positivos y exactos, y que $a > b$. Demuestre que aunque en un sistema de precisión infinita $a + b = (a^2 - b^2)/(a - b)$, los errores por redondeo pueden hacer que el segundo miembro resulte considerablemente diferente del primero. Demuestre que el caso peor ocurre cuando los errores por redondeo incurridos al calcular a^2 y b^2 son próximos al máximo, pero de signo contrario. Ilústrese con $a = 0.3525$ y $b = 0.3411$, usando aritmética de punto flotante de cuatro dígitos.

13. Supóngase que a es un número positivo propiamente redondeado y que el número 1 puede ser representado exactamente. Considérese las expresiones $u = (1 + a)^3$ y $v = 1 + (2a + a^2)$. Demuestre que cuando a crece indefinidamente los límites para el error relativo en u y en v son aproximadamente iguales, pero cuando a disminuye indefinidamente, el límite del error relativo en u tiende al triple del límite del error relativo en v. Ilústrese con $a = 0.2635$.

14. Dibuje una gráfica de procesos y obtenga una expresión para el límite del error relativo en la operación $(a + b) - b$. Ilústrelo con $a = 0.9911$ y $b = 0.9919$, y con $a = 0.201$ y $b = 0.2017$.

15. Considérese la expresión $5a + b$. Demuestre que en el resultado el error relativo inherente en a influye cinco veces más que el error relativo inherente en b.

*16. Considérense las expresiones $u = (a - b)/c$ y $v = a/c - b/c$. Supóngase que a, b, y c son todos positivos y no tienen errores inherentes y que $a \simeq b$. Demuestre que el error relativo por redondeo en v puede ser mucho mayor que el error relativo por redondeo en u. Ilustre esto con $a = 0.41$, $b = 0.36$, y $c = 0.70$, usando aritmética de punto flotante de dos dígitos.

17. Considérense las expresiones $u = a \cdot (b - c)$ y $v = ab - ac$, en las que suponemos que $a > 0$, $b > 0$, $c > 0$, $b > c$, y $b \simeq c$. Demuestre que en las condiciones citadas u tiene mucho mayor precisión relativa que v. Demuestre que con $a = 0.9364$, $b = 0.6392$, y $c = 0.6375$, $u = 0.1592 \cdot 10^{-3}$, lo cual es una expresión propiamente redondeada de la respuesta exacta, pero $v = 0.1500 \cdot 10^{-3}$.

*18. Supóngase que los coeficientes en la ecuación cuadrática $ax^2 + bx + c = 0$, son todos positivos y exactos, y que $b^2 \gg 4ac$. Demuestre primeramente que con precisión infinita la menor de las dos raíces está dada por

$$x_1 = \frac{-b + \sqrt{b - 4ac}}{2a}$$

o por

$$x_1' = \frac{-2c}{b + \sqrt{b^2 - 4ac}}$$

Demuestre después que para las condiciones especificadas x_1' da mucho mejor precisión relativa. Demuestre que con $a = 0.1000 \cdot 10^1$, $b = 0.4002 \cdot 10^1$, y $c = 0.8000 \cdot 10^{-1}$, $x_1 = -0.1500 \cdot 10^{-1}$ y $x_1' = -0.2000 \cdot 10^{-1}$. La última es la raíz exacta. (Puede también mostrar que en una gráfica de proceso la raíz cuadrada es un círculo al que llega un solo operando. El error relativo inherente en el operando aparece en la raíz cuadrada multiplicado por $\frac{1}{2}$, y la flecha que une el operando con el círculo de la raíz cuadrada puede identificarse con dicha marcha. La raíz cuadrada contiene un error relativo por redondeo adicional que en la mayoría de los sistemas FORTRAN no excede 10^{-6+1}.)

19. Considérese las ecuaciones simultáneas

$$ax + by = c$$
$$dx + ey = f$$

y la solución por la regla de Cramer

$$x = \frac{ce - bf}{ae - bd}$$
$$y = \frac{af - cd}{ae - bd}$$

Demuestre que si $ae - bd$ es pequeño, la precisión de la solución puede ser deficiente, aunque los coeficientes no tengan errores inherentes. Ilustre lo anterior mostrando que la solución del sistema

$$0.2038x + 0.1218y = 0.2011$$
$$0.1071x + 0.2436y = 0.4038$$

obtenida con aritmética de punto flotante de cuatro dígitos es $x = -1.714$, $y = 1.226$ (muestre esto). La solución exacta, que puede obtenerse con aritmética de punto flotante de ocho dígitos, es $x = 2.000$, $y = 5.000$. Si los

coeficientes en sí son inexactos, como ocurre en la mayoría de los casos, la "solución" de este sistema puede carecer totalmente de significado

20. El siguiente problema, sugerido por Richard V. Andree, demuestra efectivamente que el redondeo no es el único problema en computación numérica. Considérese el sistema

$$x + 5.0y = 17.0$$

$$1.5x + 7.501y = 25.503$$

Demuestre que si se conservan suficientes dígitos para hacer cero todos los "errores" por redondeo el sistema tendrá solución única, $x = 2$, $y = 3$. Demuestre después que si el término constante de la segunda ecuación se convierte en 25.501, que es una modificación de una parte en 12,000, se obtiene una solución mucho muy diferente

Si los coeficientes y los términos independientes fueran resultados experimentales con la duda correspondiente acerca de sus valores exactos, la "solución" carecería totalmente de sentido.

CAPITULO

3

Valuación práctica de funciones

3.1 Introducción

Vimos en el capítulo 1 que las funciones elementales comunes – seno, coseno, logaritmo, etc. – están disponibles en FORTRAN con escribir simplemente el nombre adecuado. Estas funciones suministradas automáticamente son adecuadas para muchos fines. Otras veces, sin embargo, pueden no ser suficientemente rápidas; pueden desperdiciar tiempo calculando las funciones con mucha mayor precisión de la que es necesaria, o puede necesitarse una función de la que no se dispone en la lista estándar.

Por estas razones principiamos nuestro estudio de métodos de computación numérica con el tema de valuación de funciones. La presentación se desarrolla mediante una función conocida, el seno, pero hay que notar que los mismos métodos se aplican a cualquier función que pueda desarrollarse en serie de Taylor. La representación de una función mediante una serie de Taylor, con lo que el lector debe estar familiarizado por sus estudios de Cálculo, es el punto de partida para valuar cualquier función por los métodos que aquí se explican.

Además de presentar métodos de valuación de funciones, este capítulo trata el importante tema de cómo evaluar mejor un polinomio y continúa el desarrollo de la idea fundamental de análisis del error.

3.2 Series de potencias

Cuando se trabaja con cualquier representación en serie de una función lo primero ha hacer, es reducir si es posible el rango del argumento para el que se requiere calcular la función. Esto reducirá considerablemente el error por redondeo. La determinación matemática de una en términos de la conocida serie de potencias Taylor es comple-

tamente válida para *todos* los valores del argumento, esto es, si fuera posible conservar un número infinito de dígitos en cada operación aritmética. *Computacionalmente,* la serie simple del seno es inútil para ángulos grandes y produce resultados que no tienen ninguna cifra significativa válida.

Afortunadamente esto no es problema en el caso de la serie del seno. Recuérdese que si n es entero

$$\operatorname{sen}(n\pi + y) = \operatorname{sen} n\pi \cos y + \cos n\pi \operatorname{sen} y = (-1)^n \operatorname{sen} y,$$

$$-\frac{\pi}{2} \leqslant y \leqslant \frac{\pi}{2}$$

Así, restando un múltiplo adecuado de π podemos reducir el problema de determinar el seno de cualquier ángulo a la determinación del seno de un ángulo comprendido entre $-\pi/2$ y $\pi/2$. Finalmente, si hacemos la sustitución

$$y = \frac{\pi x}{2}, \quad \operatorname{sen} y = \operatorname{sen} \frac{\pi x}{2}$$

es suficiente con considerar $\operatorname{sen} \pi x/2$ para $-1 \leqslant x \leqslant 1$.

En la práctica la reducción no se hace mediante resta repetida. En vez de ello el ángulo original se divide entre π, con la división preparada dentro de la máquina de manera que el cociente es un entero. El residuo será entonces un ángulo comprendido entre 0 y π. Si el residuo está comprendido entre $\pi/2$ y π, una resta final de π produce un ángulo comprendido entre $-\pi/2$ y $\pi/2$. El cociente entero se usa solamente para determinar si el signo del resultado final debe alterarse.

Estas operaciones preliminares en el ángulo modifican su error inherente. El valor de π usado en la división contiene un error inherente por redondeo, ya que π es un número irracional; la disminución de la magnitud del ángulo aumenta su error relativo aunque el error absoluto es el mismo; la sustitución $y = \pi x/2$ introduce un error por redondeo adicional. Un análisis completo del error debe considerar todos estos factores. En la práctica, sin embargo, tales efectos serán oscurecidos por la incertidumbre en el valor del ángulo original y por el error cometido al truncar la serie usada para calcular el valor de la función. Esto último es nuestro principal interés en este capítulo.

Los primeros cinco términos de la serie de Taylor para el seno son

$$(3.1)\quad \operatorname{sen}\frac{\pi x}{2} \simeq \frac{\pi x}{2} - \frac{1}{3!}\left(\frac{\pi x}{2}\right)^3 + \frac{1}{5!}\left(\frac{\pi x}{2}\right)^5 - \frac{1}{7!}\left(\frac{\pi x}{2}\right)^7 + \frac{1}{9!}\left(\frac{\pi x}{2}\right)^9$$

$$= 1.5707963x - 0.6459641x^3 + 0.0796926x^5 - 0.0046817541x^7 + 0.0001604411x^9$$

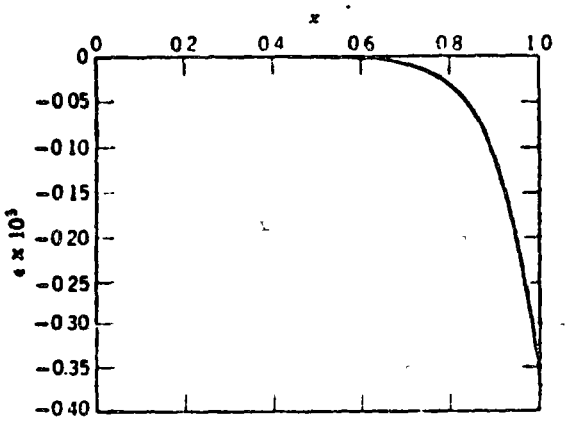


Fig. 3.1 Curva del error para la serie truncada de Taylor (3.1)

La serie completa tiene un número infinito de términos, así que hemos introducido un error por truncamiento. Se puede demostrar que para cualquier serie alternante convergente este error por truncamiento no es mayor que el primer término despreciado:

$$(3.2)\qquad |e_T| \leq \frac{1}{11!}\left(\frac{\pi x}{2}\right)^{11} \leq 0.0000035988 \simeq 3.6 \cdot 10^{-6}$$

(x es a lo sumo, 1.)

La figura 3.1 es una gráfica del error total incurrido al usar la serie (3.1). Este error total incluye los errores por truncamiento y por redondeo, pero el error por truncamiento predomina en este caso. Nótese que aunque el error es esencialmente cero para $|x| < \frac{1}{2}$, aumenta rápidamente cuando x tiende a 1. El error máximo es $3.54 \cdot 10^{-6}$, lo cual está de acuerdo con el límite dado en la expresión (3.2).

3.3 Series de Chebyshev

Interesa ver si hay alguna manera de reducir la magnitud del error cerca de $x = 1$. Podemos lograrlo, pero solamente a expensas de aumentar el error en algún otro lugar. La técnica que vamos a explorar ahora es las series de Chebyshev, distribuye el error en todo el intervalo.

Con este fin definimos los polinomios de Chebyshev $T_n(x)$ como sigue:

$$T_n(x) = \cos n\theta$$

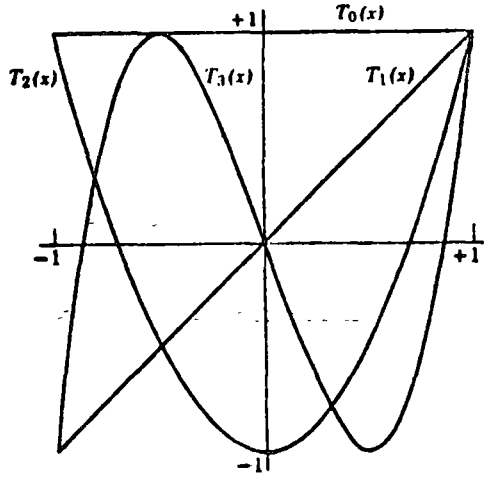


Fig. 3.2 Gráficas de los polinomios de Chebyshev $T_0(x)$, $T_1(x)$, $T_2(x)$, y $T_3(x)$.

en que $x = \cos\theta$. En otras palabras,

$$T_n(x) = \cos(n \arccos x)$$

Por ejemplo,

(3.4) $T_0(x) = \cos 0 = 1$

(3.5) $T_1(x) = \cos\theta = x$

(3.6) $T_2(x) = \cos\theta = \cos^2\theta \quad \operatorname{sen}^2\theta = x^2 - (1 - x^2) = 2x^2 - 1$

Podríamos continuar usando identidades trigonométricas para encontrar tantos elementos $T_n(x)$ como quisiéramos, pero en vez de ello vamos a establecer una fórmula de recurrencia que definirá *cualquier* $T_{n+1}(x)$ en función de $T_n(x)$ y de $T_{n-1}(x)$.

$$T_{n+1}(x) = \cos(n\theta + \theta) = \cos n\theta \cos\theta - \operatorname{sen} n\theta \operatorname{sen}\theta$$

$$T_{n-1}(x) = \cos(n\theta - \theta) = \cos n\theta \cos\theta + \operatorname{sen} n\theta \operatorname{sen}\theta$$

Sumando estas dos ecuaciones, obtenemos

$$T_{n+1}(x) + T_{n-1}(x) = 2\cos n\theta \cos\theta = 2xT_n(x)$$

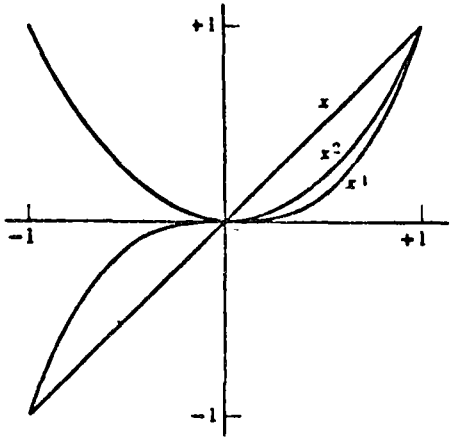


Fig. 3.3 Gráficas de los tres primeros potencias de x

y

(3.7) $$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

Mediante las ecuaciones (3.4), (3.5), (3.6) y (3.7) podemos encontrar cualquier polinomio de Chebyshev. Por ejemplo, haciendo $n = 2$ en (3.7) obtenemos

$$T_3(x) = 2xT_2(x) - T_1(x)$$

y usando (3.5) y (3.6)

$$T_3(x) = 2x(2x^2 - 1) - x = 4x^3 - 3x$$

En la parte A del apéndice 2 se incluye una lista de los 12 primeros polinomios de Chebyshev, junto con las primeras 11 potencias de x expresadas en términos de los polinomios de Chebyshev.

Los primeros cuatro polinomios de Chebyshev se grafican en la figura 3.2. Los siguientes $T_n(x)$ siguen oscilando entre ± 1, con las oscilaciones cada vez más frecuentes conforme n aumenta.

Como contraste, y para mostrar la razón de nuestro interés en los polinomios de Chebyshev, en la figura 3.3 se grafican las tres primeras potencias de x. Comparando las dos figuras, vemos que un cambio en los coeficientes de las funciones usadas en una serie de Taylor (1, x, x^2, x^3, ...) tendrán un efecto mucho mayor para $x = 1$ que cerca del cero, mientras que el efecto de un cambio en los coeficientes de una serie cuyos términos son los polinomios de Chebyshev se distribuirá en el intervalo completo ± 1.

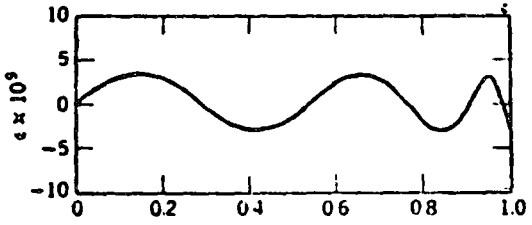


Fig. 3.4 Curva del error para la aproximación (3.8). Nótese el cambio de escala con respecto a la figura 3.1.

El problema de la determinación de los coeficientes en la serie de Chebyshev es algo complejo y no se discute en este libro.* El resultado para una serie que determina el seno mediante polinomios de Chebyshev hasta de grado 9 y por lo tanto resulta en una expresión de potencias impares de x hasta de grado 9 es el siguiente.**

$$(3.8)\quad \operatorname{sen}\left(\frac{\pi x}{2}\right) = 1.5707963x - 0.64596336x^3 + 0.079688475x^5 - 0.0046722203x^7 + 0.00015081716x^9$$

La curva del error para esta aproximación se muestra en la figura 3.4. Debe ser comparada con la serie de Taylor de la figura 3.1, que contiene las mismas potencias de x, pero con diferentes coeficientes. (Nótese el cambio en la escala.) En esta figura vemos las características de la expansión en polinomios de Chebyshev: el error máximo es menor que con la serie de Taylor, los puntos de error máximo están distribuidos a lo largo del intervalo, y los signos de los errores máximos se alternan.

La diferencia entre los dos tipos de aproximación se muestra más claramente en la figura 3.5, en la que se presenta la función seno con las dos aproximaciones. Por supuesto el error se muestra muy exagerado.

La "mejor" aproximación, que es la que tiene un valor mínimo para el valor máximo del error en el intervalo $-1 \leqslant x \leqslant 1$, se llama *aproximación de Chebyshev*. Esto es *diferente* de lo que hicimos anteriormente, que fue una expansión en polinomios de Chebyshev. Por supuesto, existen métodos para obtener la aproximación de Chebyshev, pero el trabajo adicional requerido para obtenerla rara vez se justifica, dada la pequeña

* El lector interesado puede consultar "Mathematical Tables, Chebyshev Series for Mathematical Functions", C. W. Clenshaw *Nat. Phys. Lab. (Gran Bretaña)*, 1962, para tener una explicación detallada del proceso.

** Tabla 9 de "Chebyshev Approximations of Some Exponential Functions for Use in Digital Computing", A. W. Duyvestijn y A. J. DeBoer, *Phil. Res. Rept., 16* (Nord, 1961). Este reporte también contiene series de Chebyshev para muchas funciones elementales (log, seno, arctan, exp, et

reducción relativa en el error.* (Muchos centros de cálculo tienen programas para determinar la aproximación de Chebyshev para una función cualquiera. En ese caso la determinación de la aproximación es casi un proceso de rutina.)

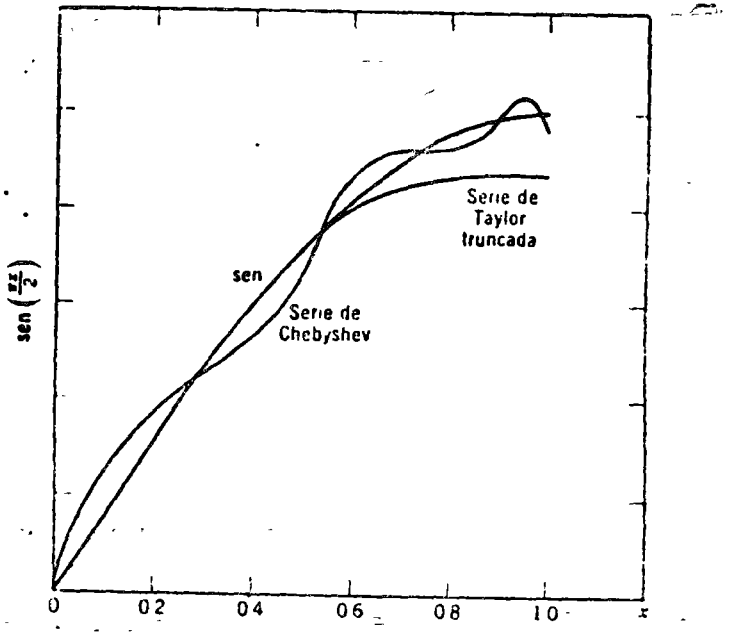


Fig. 3.5 La función seno, la aproximación de Chebyshev, y la aproximación mediante la serie de Taylor truncada. Los errores están considerablemente exagerados

3.4 Acortamiento de series de potencias

Los polinomios de Chebyshev proporcionan una buena aproximación a una función, ya que el error máximo es pequeño, pero es algo difícil calcular la aproximación. A menudo vale la pena desarrollar una rutina de computadora que se utiliza con frecuencia por muchos programadores, pero el esfuerzo que requiere de un programador el desarrollar para su uso personal una expansión en polinomios de Chebyshev, es generalmente superior a su utilidad.

Existe un método relativamente fácil de mejorar una serie de Taylor. La determinación de los coeficientes mejorados no es difícil, aunque el

* Ver, por ejemplo, T. D. Muraschina y J. D. Wrench, "The Determination of Chebyshev Approximating Polynomial for a Differential Function", *Math. Comp., 13*, 185-193 (1959).

método, llamado *economización* o *acortamiento* de una serie de potencias, cae dentro del rango de aplicabilidad en computaciones de uso frecuente.

Considérese nuevamente la serie de Taylor para el seno, pero ahora incluyendo términos hasta x^{11}.

$$\operatorname{sen}\frac{\pi x}{2} = 1.5707963x - 0.64596410x^3 + 0.079692626x^5 - 0.0046817541x^7 + 0.00016044118x^9 - 0.0000035988432x^{11}$$

De la parte B del apéndice 2 tenemos

$$x^{11} = \frac{1}{1024}(462T_1 + 330T_3 + 165T_5 + 55T_7 + 11T_9 + T_{11})$$

Reemplazando ahora T_1, T_3, T_5, T_7 y T_9 con las expresiones dadas en la parte A del apéndice 2, se obtiene

$$x^{11} = \frac{1}{1024}(11x - 220x^3 + 1232x^5 - 2816x^7 + 2816x^9 + T_{11})$$

Reemplazando el valor de x^{11} dado en esta ecuación en la serie de Taylor, obtenemos

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) = 1.5707962x - 0.64596332x^3 + 0.079688296x^5 - 0.0046718573x^7 + 0.00015054436x^9 - 0.000000000351T_{11} \tag{3.9}$$

Como $T_{11}(x)$ nunca es mayor que 1 en valor absoluto, el último término es

$$|e_T| < 3.51 \cdot 10^{-9}$$

siempre y cuando hayamos determinado los coeficientes de (3.9) con precisión infinita. Como este no es el caso, el error al utilizar (3.9) es considerablemente mayor. De hecho, evaluando (3.9) para varios valores de x, se obtiene el error máximo

$$\max|e_T| = 8.0 \cdot 10^{-9}$$

Este es menor que el error por truncamiento de la serie de Taylor del mismo grado [ver (3.2)].

La curva del error para (3.9) se muestra en la figura 3.6. Nótese que la aproximación es mejor que la que se obtiene con la serie de Taylor no acortada (figura 3.1, notando el cambio de escala) pero no tan buena como la obtenida con la serie de Chebyshev (figura 3.4), especialmente para $0.7 \leq |x| \leq 1.0$.

El proceso de acortamiento puede continuarse: x^9 puede ser reemplazada por un polinomio de grado 7 y así T_9. Naturalmente el

truncamiento sería entonces mayor. El proceso de acortamiento puede continuarse mientras el error permanezca dentro de los límites aceptables. En la parte C del apéndice 2 se dan fórmulas para el acortamiento de todas las potencias de x hasta la x^{11}.

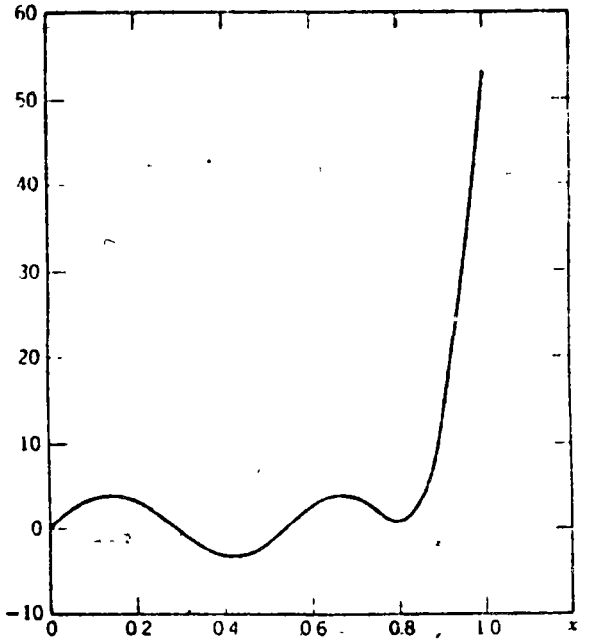


Fig. 3.6 Curva del error para la serie comprimida del seno (3.9).

3.5 Valuación de series

Independientemente del tipo de serie usada para representar una función —serie de Taylor, de Chebyshev, o comprimida— el analista tiene que encarar eventualmente el problema de determinar el valor numérico de un polinomio de la forma

$$p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{n-1}x^{n-1} + a_nx^n \tag{3.10}$$

Este polinomio puede ser reacomodado en una forma que no solo es más rápida de calcularse, sino también más precisa en muchos casos de interés práctico. Llevaremos a cabo una derivación detallada de la técnica de reacomodación aunque el resultado es intuitivamente obvio con el objeto de emplear los hechos para aplicaciones futuras, especialmente las del [illegible] que dividimos $f(x)$ entre $(x - x_0)$. Obtendre-

mos un cociente que es un polinomio de grado $n-1$ y un residuo constante:

$$p(x) = (x - x_0)(b_1 + b_2x + \cdots + b_nx^{n-1}) + b_0 \tag{3.11}$$

Nótese que $p(x_0) = b_0$, así que si podemos encontrar b_0 tendremos una manera de valuar $p(x_0)$. Esto se puede hacer fácilmente. Efectuando la multiplicación en el segundo miembro de (3.11) e igualando potencias iguales de x:

$$\begin{aligned} a_n &= b_n \\ a_{n-1} &= b_{n-1} - x_0 b_n \\ &\vdots \\ a_j &= b_j - x_0 b_{j+1} \\ &\vdots \\ a_0 &= b_0 - x_0 b_1 \end{aligned}$$

o

$$\begin{aligned} b_n &= a_n \\ b_j &= a_j + x_0 b_{j+1}, \quad j = n-1, \ldots, 0 \end{aligned}$$

Por lo tanto, podemos calcular b_n, b_{n-1}, b_{n-2}, ..., y finalmente b_0 en ese orden.

Por ejemplo, sea $n = 5$ y nótese que las b's sucesivas son las siguientes,

$$\begin{aligned} b_5 &= a_5 \\ b_4 &= a_4 + x_0 a_5 \\ b_3 &= a_3 + x_0(a_4 + x_0 a_5) \\ b_2 &= a_2 + x_0(a_3 + x_0(a_4 + x_0 a_5)) \\ b_1 &= a_1 + x_0(a_2 + x_0(a_3 + x_0(a_4 + x_0 a_5))) \\ b_0 &= a_0 + x_0(a_1 + x_0(a_2 + x_0(a_3 + x_0(a_4 + x_0 a_5)))) \end{aligned}$$

Como no dimos ninguna restricción a x_0, puede ser *cualquier* x, y podemos abandonar el subíndice cero.

Este método de valuar el polinomio (3.10) se conoce como la *regla de Horner*, y se puede representar en general en la forma

$$p(x) = a_n + x(a_{n-1} + x(a_{n-2} + \cdots + x(a_{n-1} + x(a_n)) \cdots))$$

Las a's en esta ecuación son las mismas que en (3.10). Queda claro que en la computación se efectúan primero los paréntesis más interiores. En realidad, no existe otra manera de efectuar la computación [illegible]

birla totalmente. A causa de la apariencia de la fórmula, la regla de Hörner se denomina a menudo *proceso de anidamiento*.

La evaluación de un polinomio general mediante la regla de Horner requiere de n multiplicaciones y n adiciones. El número de multiplicaciones en la evaluación de (3.10) es $n(n+1)/2$ si cada potencia de x se obtiene mediante multiplicaciones sucesivas por x, es decir $x^k = x \cdot x^{k-1}$, etc.

Para la mayoría de las aplicaciones la regla de Horner es suficiente y se usa con frecuencia. Para polinomios especiales que deben ser valuados gran número de veces con diferentes argumentos, se han creado métodos que reducen considerablemente el número total de operaciones aritméticas.*

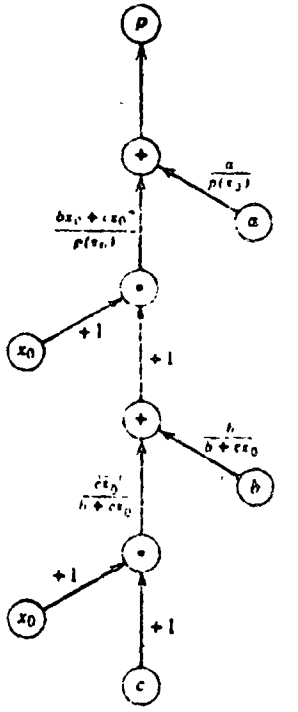


Fig. 3.7 Gráfica de proceso para valuar $p(x) = a + bx + cx^2$ mediante la regla de Horner, $p(x) = a + x(b + xc)$

Naturalmente el método de evaluación de un polinomio tiene una influencia considerable en la propagación de los errores inherentes y de los que se introducen por redondeo. Por ejemplo, supóngase que necesitamos valuar el polinomio de segundo grado

$$p(x) = a + bx + cx^2$$

La gráfica de proceso para la regla de Horner se muestra en la figura 3.7. Las flechas están identificadas en la forma descrita en el capítulo 2, pág. 70.

Podemos entonces determinar el efecto en $p(x)$ de los errores inherentes y de los errores por redondeo. Sean m_1 y m_2 los errores relativos en la primera y segunda multiplicaciones respectivamente, a_1 y a_2 los errores relativos por redondeo en las dos adiciones. Finalmente, sea Δ el error inherente en x_0, y sean δ_a, δ_b y δ_c los errores inherentes en a, b y c, respectivamente. Entonces

$$\begin{aligned} e_p = {} & \delta_c \frac{cx_0^2}{p(x_0)} + \delta_b \frac{bx_0}{p(x_0)} + \delta_a \frac{a}{p(x_0)} + \Delta\left(\frac{cx_0^2}{p(x_0)} + \frac{bx_0 + cx_0^2}{p(x_0)}\right) \\ & + m_1 \frac{cx_0^2}{p(x_0)} + m_2 \frac{bx_0 + cx_0^2}{p(x_0)} + a_1 \frac{bx_0 + cx_0^2}{p(x_0)} + a_2 \end{aligned}$$

* Ver "Evaluation of Polynomials by Computer", Donald E. Knuth, Communications of the Association for Computing Machinery, [illegible]

El error absoluto en $p(x_0)$ es

$$e_p = p(x_0) = cx_0^2(\delta_c - 2\Delta + m_1 + m_2 + \alpha_1 + \alpha_2) + bx_0(\delta_b + \Delta + m_2 + \alpha_1 + \alpha_2) + a(\delta_a + \alpha_2)$$

Para una computadora con t dígitos decimales en la mantisa de cada cantidad de punto flotante y para $|x_0| \leq 1$ tenemos

$$|e_p - p(x_0)| \leq 5 \cdot 10^{-t}(7|c| + 5|b| + 2|a|) \qquad E_H$$

Considérese ahora otra forma de valuar el polinomio, a saber, la evaluación directa en la forma en que está escrito. Podemos representar la secuencia de operaciones reagrupando los términos de la manera siguiente; las operaciones dentro de paréntesis se ejecutan primero, después las encerradas en paréntesis rectangulares y finalmente la operación encerrada en llaves.

$$p(x_0) = \{[a + (b \cdot x_0)] + c \cdot (x_0 \cdot x_0)\}$$

La gráfica de proceso se presenta en la figura 3.8. Sean α_1 y α_2 los errores relativos por redondeo en las dos adiciones indicadas por los subíndices, y sean m_1, m_2 y m_3 los errores relativos por redondeo en las multiplicaciones indicadas con los mismos índices. Entonces

$$e_p = p(x_0) = cx_0^2(\delta_c + 2\Delta + m_2 + m_3 + \alpha_2) + bx_0(\delta_b + \Delta + m_3 + \alpha_1 + \alpha_2) + a(\delta_a + \alpha_1 + \alpha_2)$$

y nuevamente para el caso de $|x_0| \leq 1$

$$|e_p - p(x_0)| \leq 5 \cdot 10^{-t}(6|c| + 5|b| + 3|a|) = E_*$$

Por lo tanto

$$E_* - E_H = 5 \cdot 10^{-t}(|a| - |c|)$$

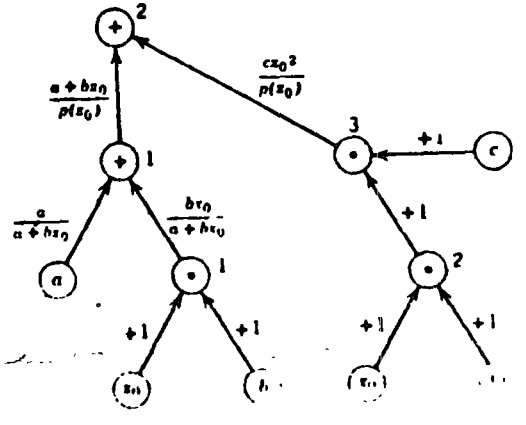


Fig. 3.8 Gráfico de proceso para $p(x_0)$

Si $|a| > |c|$. La regla de Horner da un límite menor por efecto de los errores inherentes y de redondeo. En los casos que se tratan en este capítulo se satisface esta condición: las series son series truncadas convergentes, y los coeficientes se reducen para potencias más altas de x.

Entonces en esta y en muchas otras aplicaciones prácticas la regla de Horner no sólo ahorra tiempo de computación por requerir menor número de operaciones aritméticas, sino también produce un menor error absoluto por redondeo.

Para un polinomio general, como el de la ecuación (3.10), el límite del error obtenido mediante la regla de Horner es

$$\text{error} \leq 5 \cdot 10^{-t}\left[\sum_{j=0}^{n}(3j+2)|a_j| - |a_n|\right]$$

En el caso de series convergentes, los a_j disminuyen con j, y en la expresión del límite del error los coeficientes más grandes están multiplicados por los números más pequeños.

Vale la pena hacer notar nuevamente que estos son los límites superiores para los errores inherentes y de redondeo. Los errores reales son considerablemente más pequeños.

3.6 Aproximaciones racionales y fracciones continuadas

Algunas funciones no pueden ser desarrolladas convenientemente en términos de polinomios. En otras ocasiones se dispone de un desarrollo polinomial preciso, pero converge muy lentamente. Por estas razones recurrimos a otra forma de representación de funciones: *aproximación racional*, en la cual trabajamos con el cociente de dos polinomios.

Nuevamente empezamos a partir de una expansión en serie de Taylor

$$(3.12) \qquad f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6 + a_7x^7 + \cdots$$

A continuación escribimos $f(x)$ como el cociente de dos polinomios de tercer grado

$$(3.13) \qquad f(x) = \frac{b_0 + b_1x + b_2x^2 + b_3x^3}{1 + c_1x + c_2x^2 + c_3x^3}$$

La constante + 1 que aparece en el denominador no indica pérdida de generalidad, ya que cualquier otra constante que apareciera ahí podría ser simplificada dividiendo entre ella numerador y denominador. Igualando los segundos miembros de (3.12) y (3.13) y simplificando las fracciones tenemos

$$b_0 + b_1x + b_2x^2 + b_3x^3 = (1 + c_1x + c_2x^2 + c_3x^3)(a_0 + a_1x + \ldots + a_7x^7)$$

Multiplicando y agrupando potencias iguales de x,

$$\begin{aligned} b_0 &= a_0 \\ b_1 &= a_1 + a_0c_1 \\ b_2 &= a_2 + a_1c_1 + a_0c_2 \\ b_3 &= a_3 + a_2c_1 + a_1c_2 + a_0c_3 \\ 0 &= a_4 + a_3c_1 + a_2c_2 + a_1c_3 \\ 0 &= a_5 + a_4c_1 + a_3c_2 + a_2c_3 \\ 0 &= a_6 + a_5c_1 + a_4c_2 + a_3c_3 \end{aligned}$$

(Las tres últimas ecuaciones resultan de haber supuesto una forma de representación en la que los coeficientes de potencias de x de orden superior al tercero en el numerador son cero.)

Tenemos ahora siete ecuaciones en las siete incógnitas b_0, b_1, b_2, b_3, c_1, c_2, y c_3. Estas siete ecuaciones simultáneas pueden ser resueltas por los métodos que se indican en el capítulo 8.

Estimamos el error en esta formulación considerando la magnitud del coeficiente de b_7 si estuviera incluido, dividido por el valor del denominador:

$$\frac{(a_7 + a_6c_1 + a_5c_2 + a_4c_3)x^7}{1 + c_1x + c_2x^2 + c_3x^3}$$

Esta es una aproximación únicamente al error por truncamiento y no incluye el error por redondeo, pero el error por truncamiento generalmente será mucho mayor que el error por redondeo.

Generalmente las aproximaciones racionales no se valúan como se expresa en la ecuación (3.13), sino mediante el uso de una *fracción continuada* equivalente. Podemos ver cómo se lleva a cabo esto considerando por última vez la función seno

$$\text{sen}\left(\frac{\pi x}{2}\right) = \frac{\pi x}{2} - \frac{1}{3!}\left(\frac{\pi x}{2}\right)^3 + \frac{1}{5!}\left(\frac{\pi x}{2}\right)^5 - + \qquad (3.14)$$

Buscamos una aproximación racional de la forma *

* La selección de una forma apropiada para una aproximación racional es en parte ciencia, en parte arte, y en parte buen juicio guiado por la experiencia. No podemos dar reglas explícitas pero al menos podemos justificar las características generales de este ejemplo. No necesitamos un término constante en el denominador, si incluyéramos uno éste tendría que ser cero, porque el seno de cero y de cero. El lector puede verificar que la existencia de un término en x^3 en el numerador y un término en x en el denominador es redundante porque, si el seno es una función impar, es decir, sen $(-x) = -\text{sen}\,x$, la división debe tener una forma independiente en consecuencia con la propiedad de la función seno.

$$\text{sen}\left(\frac{\pi x}{2}\right) = \frac{ax + bx^3}{1 + cx^2} \qquad (3.15)$$

y

$$(1 + cx^2)\left(\frac{\pi}{2}x - \left(\frac{\pi}{2}\right)^3\frac{1}{3!}x^3 + \left(\frac{\pi}{2}\right)^5\frac{1}{5!}x^5 - \left(\frac{\pi}{2}\right)^7\frac{1}{7!}x^7\right) = ax + bx^3$$

Por lo tanto

$$a = \frac{\pi}{2} \qquad \text{(potencias de } x\text{)}$$

$$c\cdot\frac{\pi}{2} - \left(\frac{\pi}{2}\right)^3\frac{1}{3!} = b \qquad \text{(potencias de } x^3\text{)}$$

$$-c\left(\frac{\pi}{2}\right)^3\frac{1}{3!} + \left(\frac{\pi}{2}\right)^5\frac{1}{5!} = 0 \qquad \text{(potencias de } x^5\text{)}$$

Determinando el valor de c en la última ecuación, tenemos

$$\left\{\begin{array}{l} c = \dfrac{1}{20}\left(\dfrac{\pi}{2}\right)^2 = 0.12337\,0055 \\ \text{Entonces} \\ b = -\dfrac{7}{60}\left(\dfrac{\pi}{2}\right)^3 = -0.45217\,4868 \\ \text{y} \\ a = \dfrac{\pi}{2} = 1.57079\,633 \end{array}\right. \qquad (3.16)$$

El error *aproximado* por truncamiento es

$$e_T = \frac{\left(\dfrac{\pi}{2}\right)^7\dfrac{1}{7!}x^7}{1 + \dfrac{1}{20}\left(\dfrac{\pi}{2}\right)^2x^2} = \frac{0.0046815x^7}{1 + 0.12337x^2}$$

Nótese que (3.15) es aproximadamente equivalente a una serie de potencias de tres términos, como la indicada en (3.14).

Esta aproximación racional tiene el mismo orden de precisión que una serie de Taylor de cinco términos:

$$\text{sen}\left(\frac{\pi x}{2}\right) \simeq \frac{\pi x}{2} - \frac{1}{3!}\left(\frac{\pi x}{2}\right)^3 + \frac{1}{5!}\left(\frac{\pi x}{2}\right)^5$$

* Esta expresión se valúa mediante la regla de Horner, con evaluación de x^2, se requieren cuatro multiplicaciones y dos sumas.

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) = x(a - x^2(b - cx^2))$$

Si la función racional (3.15) se valúa aplicando la regla de Horner al numerador y al denominador, se requieren cuatro multiplicaciones, dos sumas y una división. Es decir, no se ahorra trabajo con respecto a la serie de Taylor.

Por tanto convertimos (3.15) en una fraccion continuada equivalente. Primeramente reescribimos (3.15) en la forma

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) = -\frac{7\pi}{6}\left[\frac{x^3 - \dfrac{60}{7}\left(\dfrac{2}{\pi}\right)^2 x}{x^2 + 20\left(\dfrac{2}{\pi}\right)^2}\right]$$

y dividimos el numerador entre el denominador para obtener un cociente igual a x, y un residuo igual a

$$-\frac{200}{7}\left(\frac{2}{\pi}\right)^2 x:$$

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) = -\frac{7\pi}{6}\left[x - \frac{\dfrac{200}{7}\left(\dfrac{2}{\pi}\right)^2 x}{x^2 + 20\left(\dfrac{2}{\pi}\right)^2}\right]$$

ó

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) = -\frac{7\pi}{6}\left\{x - \frac{\dfrac{200}{7}\left(\dfrac{2}{\pi}\right)^2}{\left[\dfrac{x^2 + 20\left(\dfrac{2}{\pi}\right)^2}{x}\right]}\right\}$$

Tomamos ahora la fracción encerrada en paréntesis rectangular y dividimos su numerador entre su denominador para obtener

$$\frac{x^2 + 20\left(\dfrac{2}{x}\right)^2}{x} = x + \frac{20\left(\dfrac{2}{\pi}\right)^2}{x}$$

y finalmente

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) = -\frac{7\pi}{6}\left[x - \frac{\dfrac{200}{7}\left(\dfrac{2}{\pi}\right)^2}{x + \dfrac{20\left(\dfrac{2}{\pi}\right)^2}{x}}\right]$$

o

$$\operatorname{sen}\frac{\pi x}{2} = -3.66519143\left(x - \frac{11.47221432}{x + \dfrac{8.03055026}{x}}\right) \tag{3.17}$$

La evaluación de (3.17) requiere dos divisiones, dos sumas y una multiplicación, lo cual es un ahorro considerable de trabajo en comparación con la función racional (3.15) y con el polinomio (3.14). En este ejemplo, si el tiempo de división de la computadora es menor que ½ del tiempo de multiplicación, la fracción continuada (3.17) es más rapida que el polinomio (3.14). (En bastantes computadoras de uso común se satisface esta condición de tiempo; en algunas máquinas la multiplicación y la división consumen el mismo tiempo.)

Las fracciones continuadas pueden ser comprimidas análogamente a lo que se hizo con las series de potencias. Véase, por ejemplo, Hans J. Maehly, "Methods for Fitting Rational Approximations. Part I: Telescoping Procedures for Continued Fractions", *Communications* de la Asociación de Maquinaria de Computación, 7, 150-162 (abril 1960).

3.7 Funciones elementales

Vimos en la Sección 3.1 que no es necesario poder valuar un seno para un argumento arbitrario: siempre es posible reducir el argumento a $-\pi/2 \leqslant x \leqslant \pi/2$. Esto es un ejemplo de la manera en que puede simplificarse la computación de funciones elementales. Podemos ahora considerar técnicas similares para algunas otras funciones comunes.

Coseno

Nunca se requiere un programa separado para calcular un coseno porque podemos emplear la identidad

$$\operatorname{sen}(x + \pi/2) = \operatorname{sen} x \cos \pi/2 + \cos x \operatorname{sen} \pi/2 = \cos x$$

Por consiguiente, se requiere solamente agregar $\pi/2$ al ángulo y usar la [illegible] seno

Funciones hiperbólicas

Suponiendo que se dispone de una rutina para calcular funciones exponenciales, el seno y el coseno hiperbólicos pueden determinarse a partir de

$$\operatorname{senh} x = \tfrac{1}{2}(e^x - e^{-x})$$

$$\cosh x = \tfrac{1}{2}(e^x + e^{-x})$$

Nótese, sin embargo, que e^x y e^{-x} son aproximadamente iguales para valores de x cercanos a cero. Entonces al calcular senh (x) estamos restando dos números aproximadamente iguales, y ya hemos visto en la sección 2.8 que esto disminuye la precisión relativa. Si la precisión relativa es importante para valores pequeños de x, es mucho mejor usar la serie de potencias:

$$\operatorname{senh} x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \ldots$$

Para pequeños valores de x basta tomar unos cuantos términos.

Logaritmos

La rutina del logaritmo que se llama a ejecución cuando escribimos LOGF en un programa FORTRAN fue clasificada en el capítulo 1 como un logaritmo natural, es decir, un logaritmo de base e. Algunos sistemas proporcionan una función adicional para obtener el logaritmo ordinario (de base 10) en los casos en que se usa con mucha frecuencia, pero esto se hace solamente para simplificar la preparación de programas y no es esencial.

Si se conoce el logaritmo de base e de algún número y se desea obtener su logaritmo de base b, se procede de la manera siguiente. Recordemos que si

$$\log_e x = k$$

entonces

$$e^k = x$$

por definición. Por lo tanto

$$\log_b x = \log_b (e^k) = k \log_b e = (\log_b e)(\log_e x)$$

Entonces, para obtener el logaritmo de un número en alguna base diferente de e, basta multiplicar el logaritmo natural por el logaritmo del número e en la nueva base; éste es una constante fija. En particular, $\log_{10} e = 0.4342944$.

En la parte D del apéndice 2, se presentan aproximaciones polinomiales para algunas de las funciones elementales, incluyendo el logaritmo natural.

3.8 Caso particular 3: Errores en la evaluación directa de la serie del seno

La serie de Taylor para el seno

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

es válida teóricamente para cualquier valor de x, como hemos indicado. En realidad es casi inútil para valores grandes de x. Será instructivo investigar por qué ocurre esto.

Escribiremos un programa que valúa la serie directamente, es decir, comenzando con el primer término y procediendo término a término hasta encontrar alguno que es menor en valor absoluto que alguna cantidad, digamos 10^{-6}. Sabemos que el error por truncamiento es entonces menor que el primer término despreciado, así que debería ser posible calcular el seno con una aproximación de 10^{-6} con simplemente tomar suficientes términos. Veremos que esto no es prácticamente posible debido a problemas extremos de redondeo.

El programa requerirá una interesante estratagema para evitar producir resultados intermedios que sean demasiado grandes como variables de punto flotante. El mayor ángulo que consideraremos será de aproximadamente 50 radianes; si tratáramos de elevar 50 a las elevadas potencias requeridas excederíamos con mucho las magnitudes máximas permitidas para variables de punto flotante casi en todos los sistemas FORTRAN. Por tanto el método que seguiremos consistirá en calcular cada nuevo término en la serie a partir del precedente. La relación de recurrencia no es complicada. Dado el primer término x, podemos obtener el siguiente término multiplicando por $-x^2$ y dividiendo por $2 \cdot 3$. Una vez obtenido el segundo término podemos obtener el tercero multiplicando por $-x^2$ y dividiendo entre $4 \cdot 5$. Brevemente, dado el término precedente, podemos obtener el siguiente multiplicando por $-x^2$ y dividiendo entre el producto de los dos enteros siguientes.

El diagrama de bloque se muestra en la figura 3.9. Está preparado de manera que lea tarjetas, cada una de las cuales contiene un ángulo en grados, hasta llegar a una "tarjeta centinela" con un ángulo de cero grados. Los ángulos en grados se convierten primeramente en radianes mediante la división entre $180/\pi$, el resultado se denomina X. Necesitamos ahora poner en movimiento el proceso de recurrencia. Estaremos agregando continuamente un término a una suma que eventualmente constituye el seno, una vez que se hayan calculado suficientes términos. Para empezar hacemos esta suma igual a X; el primer término

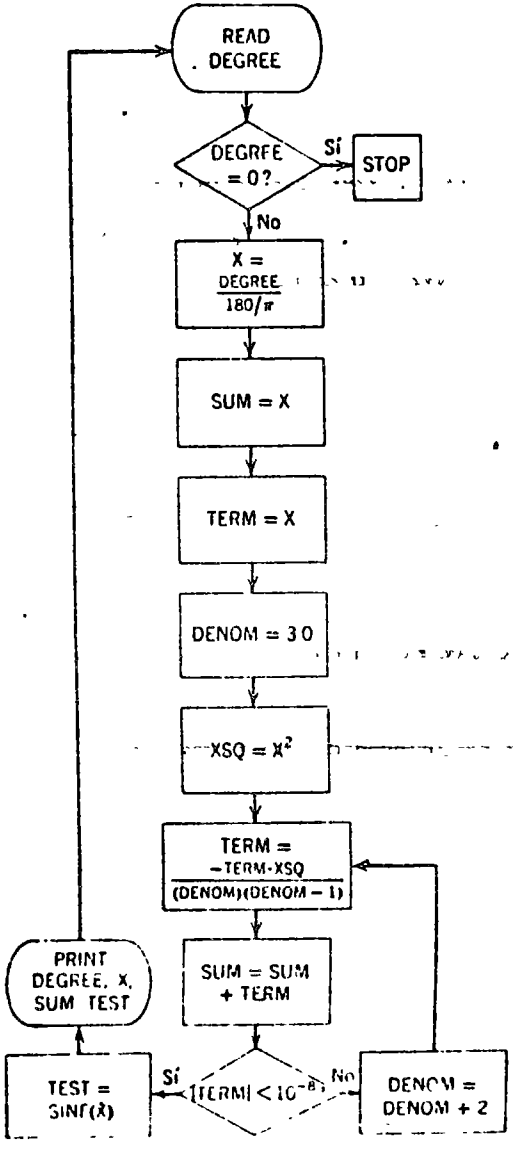


Fig. 3.9 Diagrama de bloque de un método para calcular el seno (Caso particular 3)

que se calcula por el método de recurrencia será $-x^3/3!$. Entonces el término precedente es también X. Para obtener los enteros sucesivos entre los que hay que dividir, damos el valor 3 a una variable llamada DENOM. Para evitar recalcular repetidas veces x^2, lo calculamos una vez antes de entrar al ciclo repetitivo y le damos el nombre XSQ.

La obtención de un nuevo término es ahora simplemente la operación de multiplicar el término precedente por —XSQ y dividir el producto por el resultado de multiplicar DENOM por DENOM — 1.0. Este

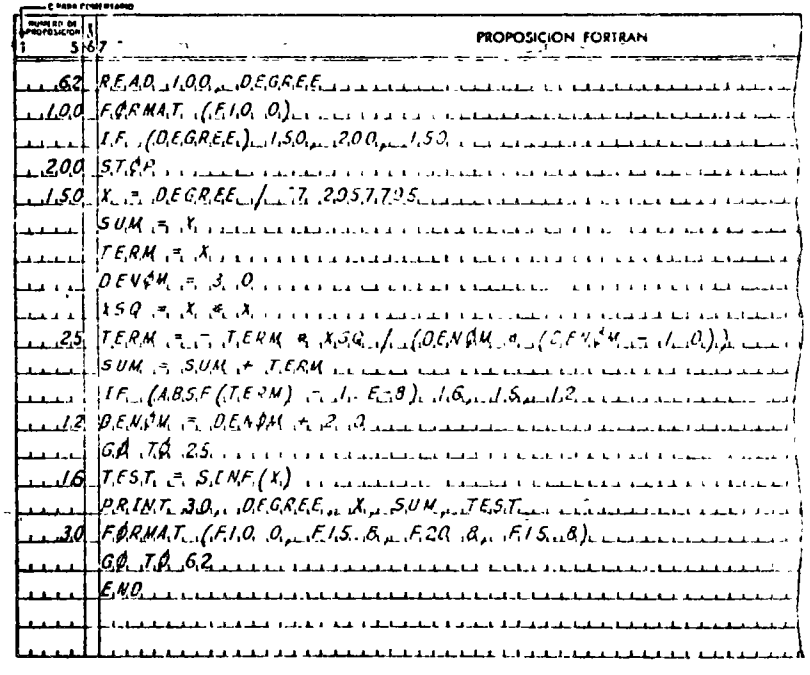


Fig. 3.10 Programa para calcular el seno (Caso particular 3)

nuevo término reemplaza al término precedente y se agrega a la suma. En este punto debemos determinar si ya se han calculado suficientes términos. Para esto preguntamos si el valor absoluto del término que acabamos de calcular es menor o igual que 10^{-8}. Si lo es, podemos imprimir el resultado y proceder a la lectura de la siguiente tarjeta. Si no lo es, debemos incrementar en 2 unidades el valor de DENOM antes de proceder a calcular otro término.

Además de imprimir el valor del seno calculado por este método, sería interesante compararlo con el valor calculado por la función seno

suministrada por el sistema FORTRAN. Esto se efectúa inmediatamente antes de imprimir.

El programa que se muestra en la figura 3.10 sigue los pasos del diagrama de bloque y no introduce nuevos conceptos de FORTRAN. Las dimensiones de las especificaciones de campo F en la proposición 30 se seleccionaron para poder acomodar el tamaño esperado de los resultados.

Los resultados se presentan en la figura 3.11 para ángulos iguales a 30° más múltiplos de 360°. Por lo tanto, el resultado exacto en cada caso debería ser 1/2. El resultado para 30° es tan aproximado como podríamos razonablemente esperar, ya que el error en este caso es exactamente la tolerancia de 10^{-8}. El error se incrementa para ángulos más grandes. Para 390° el valor es aceptable; los valores empiezan a deteriorarse; y a los 1470° el sistema se desintegra. Angulos mayores producen valores del seno que también carecen de sentido.

Consideremos el primer valor para el que el método falla completamente (1470°) para ver si podemos averiguar lo que ha sucedido. Mediante un programa independiente que no se muestra aquí, se imprimieron los valores de cada uno de los términos de la serie. El primer término es simplemente el valor de x, 25.656340 radianes. El segundo término, a ocho dígitos, es −2789.0481; cuando se suman estos dos términos conservando ocho dígitos, la suma es −2763.3921. En la adición se perdieron los dos últimos dígitos del primer término. Obviamente esos dos términos nunca van a poder ser reincorporados en el cálculo cuando la suma se haya reducido a un valor menor que 1. El tercer término es 89849.610; al sumarlo con el resultado anterior para obtener 87086.218, se pierde el último dígito de la suma precedente. Se han perdido hasta el momento tres dígitos del primer término. El cuarto término es −1362035.9; en la adición para obtener −1274949.7, se pierden dos dígitos de la suma previa. Se han perdido los últimos cinco dígitos del primer término y el esquema debe aparecer claro. El último término de la serie es $55037680 \cdot 10^2$; después de sumarlo a la suma previa, se han perdido *todos* los dígitos del primer término, junto con algunos dígitos de los otros términos. En el siguiente renglón de la figura 3.10, correspondiente a 1830°, el término más grande es aproximadamente $2.7 \cdot 10^{12}$, lo que causa la pérdida de todos los dígitos del primero y segundo términos.

Claramente, el problema más serio en este caso es que las adiciones se realizan en una secuencia que está muy lejos de la ideal, como vimos en el capítulo 2 que es mucho mejor sumar números más pequeños, o más generalmente, combinar los términos pequeñas como sea posible.

Pero este no es el único problema. Este ejemplo se ejecutó en una computadora binaria en la que las variables de punto flotante se representan con el equivalente de unas ocho cifras decimales. Considérese un término como $0.26553689 \cdot 10^{13}$. Se escribiría en la forma 2,665,368,900,000 en la que los ceros carecen de significado: sirven solamente para localizar el punto decimal en esta manera de escribir el número. Obviamente, los ocho dígitos significativos son una aproximación, y los ceros representan los dígitos que no podemos conservar en el sistema de computación. En otras palabras, esta aproximación podría diferir del valor verdadero hasta en 50,000 unidades. Esta clase de error hace imposible esperar algún significado de un valor final que nunca es mayor que 1.

30.	0.52359878	0.49999999	0.49999999
390.	6.80678415	0.49999993	0.50000005
750.	13.08996952	0.50013507	0.50000010
1110.	19.37315488	0.51658490	0.50000016
1470.	25.65634012	24.25401855	0.50000010
1830.	31.93952560	14380.23767090	0.50000025
2190.	38.22271109	25902480.00000000	0.50000040
2550.	44.50589609	-130402508.00000000	0.50000013
2910.	50.78908157	-83272283.00000000	0.50000029

Fig. 3.11 Resultados del programa de la figura 3.10 (Caso particular 3)

30.	0.52359878	0.49999999	0.49999999
390.	6.80678415	0.50000006	0.50000005
750.	13.08996952	0.50000011	0.50000010
1110.	19.37315488	0.50000016	0.50000016
1470.	25.65634012	0.50000143	0.50000010
1830.	31.93952560	-0.49953845	0.50000025
2190.	38.22271109	0.79868912	0.50000040
2550.	44.50589609	29.53991437	0.50000013
2910.	50.78908157	-142982.02734375	0.50000029

Fig. 3.12 Resultados del programa de la figura 3.10, modificado para realizar los cálculos en doble precisión (Caso particular 3)

Recurriendo a la *doble precisión* podemos demostrar que en este caso el problema estriba realmente en el significado limitado de las variables de punto flotante. Veremos en qué consiste este método.

En el método de doble precisión cada variable se representa con el doble de dígitos que se usarían normalmente, y las operaciones aritméticas se preparan de manera que se tomen en cuenta todos los dígitos. En la versión de FORTRAN usada para este ejemplo si se coloca una D en la columna 1 de una proposición aritmética, toda la aritmética de esa proposición se ejecuta con doble precisión. Los resultados obtenidos al correr el mismo programa con este cambio se presentan en la figura 3.12. Vemos que para valores hasta de 1830° para los que el programa original falló completamente, los resultados son aproximada-

mente correctos. Sin embargo, para ángulos muy grandes, aún la doble precisión no es suficiente.

Podemos brevemente observar cómo se comportó la función seno suministrada por el sistema FORTRAN. En todos los casos cuando menos seis dígitos son correctos. La disminución en la precisión para ángulos grandes es debida a una pérdida de cifras significativas al reducir el ángulo original a un ángulo menor que $\pi/2$. Por ejemplo, cuando convertimos a radianes el ángulo 2190°, obtenemos 38.22271109. La computadora ha impreso 10 dígitos como el equivalente decimal del valor binario flotante, pero no puede haber ahí más que ocho dígitos significativos. Cuando esta aproximación de ocho dígitos es reducida a un ángulo menor que $\pi/2$, hemos de hecho restado dos números casi iguales, lo cual, como hemos visto, reduce la aproximación relativa. En otras palabras, no calculamos realmente el seno de 2190°, sino de algún ángulo ligeramente diferente.

Por razones obvias los senos de ángulos grandes nunca se calculan por el método presentado en este caso particular. Se espera que el lector haya aprendido a conocer algunos de los problemas que se tienen que encarar cuando se trabaja con una computadora, lo que obviamente encierra aproximaciones. Los usuarios ingenuos de las computadoras tienen una tendencia a suponer que si el "cerebro gigante" imprime ocho dígitos, éstos tienen que significar algo. Confiamos que el estudio de este caso particular muestre la falta de fundamento de esta suposición.

Ejercicios

1. En seguida se muestran algunas funciones y la serie de Taylor correspondiente. Para el valor de x que se indica estime el número de términos requeridos para producir un valor de la función cuyo error por truncamiento sea menor que $5 \cdot 10^{-6}$. Para cada caso estime también el número de términos requeridos para un error por truncamiento menor que $5 \cdot 10^{-9}$.

   °a. $\operatorname{sen} x = x - \dfrac{x^3}{3!} + \dfrac{x^5}{5!} - \dfrac{x^7}{7!} + \cdots \qquad x = 1$

   b. $\operatorname{sen} x = x - \dfrac{x^3}{3!} + \dfrac{x^5}{5!} - \dfrac{x^7}{7!} + \cdots \qquad x = 3$

   °c. $\arctan x = \dfrac{\pi}{2} - \dfrac{1}{x} + \dfrac{1}{3x^3} - \dfrac{1}{5x^5} + \dfrac{1}{7x^7} + \cdots \qquad x = 2$

   (La serie es válida para $x > 1$.)

   d. $\log_e x = (x-1) - \dfrac{(x-1)^2}{2} + \dfrac{(x-1)^3}{3} - \dfrac{(x-1)^4}{4} + \cdots \qquad x = 3$

   (La serie es válida para $0 < x \leq 2$.)

   e. $e^x = 1 + x + \dfrac{x^2}{2!} + \dfrac{x^3}{3!} + \dfrac{x^4}{4!} + \cdots \qquad x = 1$

2. Demuestre que para $|x| \leq 1$ los polinomios de Chebyshev satisfacen las desigualdades $|T_n(x)| \leq 1$.
3. Demuestre que

$$\int_{-1}^{1} \frac{T_m(x)\, T_n(x)\, dx}{\sqrt{1-x^2}} = 0 \qquad m \neq n$$

Por tener esta propiedad, se dice que los polinomios de Chebyshev son *ortogonales* en el intervalo $(-1, 1)$ con la función de peso $1/\sqrt{1-x^2}$. (*Sugestión* Use la definición de $T_n(x)$ y reemplace x por θ.)

4. Demuestre que

$$\int_{-1}^{1} \frac{[T_m(x)]^2\, dx}{\sqrt{1-x^2}} = \begin{cases} \pi & m = 0 \\ \dfrac{\pi}{2} & m = 1, 2, \end{cases}$$

5. Con base en los resultados de los Ejercicios 3 y 4, demuestre que los coeficientes a_i en una serie de Chebyshev para $f(x)$

$$f(x) = \sum_{i=0}^{\infty} a_i\, T_i(x)$$

se pueden obtener a partir de

$$a_0 = \frac{1}{\pi} \int_{-1}^{1} \frac{f(x)\, dx}{\sqrt{1-x^2}}$$

$$a_n = \frac{2}{\pi} \int_{-1}^{1} \frac{f(x)\, T_n(x)\, dx}{\sqrt{1-x^2}} \qquad n \neq 0$$

En la práctica estas fórmulas se usan raramente para calcular los coeficientes a_i, debido a la dificultad en calcular las integrales.

°6. A partir de los resultados del ejercicio 5 determine los primeros cinco coeficientes de la serie de Chebyshev para

$$f(x) = x$$

Puede usar las siguientes integrales definidas:

$$\int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}} = \pi \qquad \int_{-1}^{1} \frac{x^2\, dx}{\sqrt{1-x^2}} = \frac{\pi}{2} \qquad \int_{-1}^{1} \frac{x^4\, dx}{\sqrt{1-x^2}} = \frac{3\pi}{8}$$

$$\int_{-1}^{1} \frac{x\, dx}{\sqrt{1-x^2}} = 0 \qquad \int_{-1}^{1} \frac{x^3\, dx}{\sqrt{1-x^2}} = 0 \qquad \int_{-1}^{1} \frac{x^5\, dx}{\sqrt{1-x^2}} = 0$$

7. Sabiendo que el sexto coeficiente, a_5, de la serie de Chebyshev para $f(x) = x$ debe ser cero, deduzca que

$$\int_{-1}^{1} \frac{x^6\, dx}{\sqrt{1-x^2}} = \frac{5\pi}{16}$$

[ ] Las integrales citadas en el ejercicio 6.

*8. a. Encuentre los cinco primeros coeficientes de la serie de Chebyshev para

$$f(x) = \sqrt{1 - x^3}$$

b. Escriba los cinco primeros términos de la serie en potencias de x.
c. Calcule la serie de cinco términos en x para $x = 0.5$, compare el resultado con el valor correcto y con el valor dado por la serie de Taylor de cinco términos desarrollada con respecto a $x = 0$.

9. a. Usando las integrales de los ejercicios 6 y 7, encuentre los cinco primeros coeficientes de la serie de Chebyshev para

$$f(x) = |x|$$

b. Escriba los cinco términos de la serie de Chebyshev en potencias de x.
c. Valúe el resultado obtenido en la parte b para $x = +0.5$.

*10. Los *polinomios desplazados de Chebyshev* se pueden definir como

$$T_n^*(x) = T_n(2x - 1)$$

Los polinomios desplazados, T_n^*, se usan en el intervalo $0 \leq x \leq 1$ exactamente en la misma forma en que los polinomios T_n se usan en $-1 \leq x \leq 1$. Determine los cuatro primeros polinomios desplazados de Chebyshev.

11. Demuestre que los polinomios desplazados de Chebyshev son ortogonales en el intervalo $(0, 1)$ con una función de peso $(x - x^2)^{-1/2}$; es decir,

$$\int_0^1 \frac{T_n^*(x)\, T_m^*(x)\, dx}{\sqrt{x - x^2}} = 0 \qquad m \neq n$$

*12. Comprima la aproximación

$$e^x \simeq 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!} + \frac{x^6}{6!} + \frac{x^7}{7!}$$

a una aproximación que incluya hasta x^6.

13. Comprima la serie que resulte del ejercicio 12 a una aproximación que incluya términos hasta x^5.

14. a. Encuentre los cinco primeros coeficientes de la expansión de Chebyshev para

$$f(x) = 6x^4 - 2x^3 + x^2 - x + 4 \qquad |x| \leq 1$$

b. Utilizando dos veces la técnica de acortamiento, aproxime $f(x)$ mediante un polinomio de segundo grado.

15. Demuestre que la aproximación

$$e^x \simeq 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!}$$

puede reescribirse en la forma

$$e^x \simeq 1 + x\left(1 + \frac{x}{2}\left(1 + \frac{x}{3}\left(1 + \frac{x}{4}\left(1 + \frac{x}{5}\right)\right)\right)\right)$$

16. Demuestre que la aproximación

$$\tan^{-1} x \simeq x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \frac{x^9}{9}$$

puede reescribirse en la forma

$$\tan^{-1} x \simeq x(1 - x^2(\tfrac{1}{3} - x^2(\tfrac{1}{5} - x^2(\tfrac{1}{7} - x^2(\tfrac{1}{9})))))$$

17. Encuentre una factorización semejante a las de los ejercicios 15 y 16 para la función

$$\operatorname{sen}^{-1} x \simeq x + \frac{x^3}{2 \cdot 3} + \frac{1 \cdot 3 \cdot x^5}{2 \cdot 4 \cdot 5} + \frac{1 \cdot 3 \cdot 5 \cdot x^7}{2 \cdot 4 \cdot 6 \cdot 7} + \frac{1 \cdot 3 \cdot 5 \cdot 7 \cdot x^9}{2 \cdot 4 \cdot 6 \cdot 8 \cdot 9}$$

18. Encuentre una factorización similar a las de los ejercicios 15 y 16 para la función

$$J_0(x) \simeq 1 - \frac{(x/2)^2}{1^2} + \frac{(x/2)^4}{1^2 \cdot 2^2} - \frac{(x/2)^6}{1^2 \cdot 2^2 \cdot 3^2} + \frac{(x/2)^8}{1^2 \cdot 2^2 \cdot 3^2 \cdot 4^2}$$

*19. Si se conoce de antemano el número de términos que se van a retener en una serie truncada convergente infinita, el mejor procedimiento de computación es la regla de Horner. Escriba una proposición usando la regla de Horner para valuar la serie de Taylor para e^x que incluya hasta x^6:

a. Usando los valores de los coeficientes en forma decimal, es decir,

$$e^x \simeq 1 + x + 0.5x^2 + 0.166667x^3 + 0.0416667x^4 + 0.00833333x^5 + 0.00138889x^6$$

b. Sin usar constantes excepto los enteros de 1 a 6, siguiendo el método del ejercicio 15

*20. Considere la siguiente secuencia de computación:

1. Haga la variable E igual a 1.
2. Haga la variable D igual a 5
3. Reemplace E por $1 + (EX/D)$.
4. Si D es igual a 1, detenga el proceso; si no lo es, reste 1 de D y repita el paso 3. Demuestre que cuando el proceso termina

$$E = 1 + x + \frac{x^2}{2} + \frac{x^3}{2 \cdot 3} + \frac{x^4}{2 \cdot 3 \cdot 4} + \frac{x^5}{2 \cdot 3 \cdot 4 \cdot 5} \simeq e^x$$

Dibuje un diagrama de bloque para el proceso y escriba un segmento de programa en el que se supone que X ha recibido un valor dado en una proposición previa.

21. Siguiendo el método del ejercicio 20, dibuje un diagrama de bloque y escriba un segmento de programa para evaluar la serie de Taylor para la función sen x incluyendo términos hasta x^{15} (Vea el ejercicio 17).

22. Escriba una rutina para evaluar el seno de x para $-3 \leq x \leq 3$ con un error por truncamiento menor que $5 \cdot 10^{-8}$, usando una modificación adecuada al método del ejercicio 20. Si $|x| \leq 1$, haga $D = 7$ antes de entrar al ciclo de computación; si $|x| > 1$, haga D igual al resultado del Ejercicio 11b.

23. Si *no* se conoce de antemano el número de términos que se van a retener, no se puede usar la regla de Horner; la serie debe ser valuada "desde el "frente". Sin embargo, si x es grande, digamos en el rango de 10 a 20, al elevar x a una potencia grande se puede exceder la magnitud permisible para números de punto flotante aunque el término completo no sea demasiado grande. Una solución es efectuar las divisiones entre los números del denominador al mismo tiempo que se eleva x a la potencia deseada. (Ver caso particular 3.) Escriba una rutina para valuar e^x, comenzando por el principio y continuando hasta encontrar un término que sea menor que 10^{-8} en valor absoluto.

24. Escriba una rutina para valuar las siguientes funciones por el método del caso particular 3. Comience desde el principio y continúe hasta encontrar un término que sea menor en valor absoluto que 10^{-8}.

a. $J_0(x) = 1 - \frac{(x/2)^2}{1^2} + \frac{(x/2)^4}{1^2 \cdot 2^2} - \cdots$

b. $\tan^{-1} x = \frac{\pi}{2} - \frac{1}{x} + \frac{1}{3x^3} - \frac{1}{5x^5} + \cdots \quad x > 1$

c. $J_3(x) = \frac{x^3}{2^3 \cdot 2!} - \frac{x^4}{2^4 \cdot 1! \cdot 3!} + \frac{x^5}{2^5 \cdot 2! \cdot 4!} - \frac{x^6}{2^6 \cdot 3! \cdot 5!} + \cdots$

25. Recuerde que si $p(x)$ es un polinomio de grado n, entonces

$$p(x) = (x - x_0)\, q(x) + b_0$$

en que

$$q(x) = b_1 + b_2 x + \cdots + b_n x^{n-1}$$

y las b_j pueden ser calculadas en forma recurrente mediante

$$b_n = a_n$$

$$b_j = a_j + x_0 b_{j+1}, \qquad j = n - 1, \ldots, 0$$

a. Demuestre que

$$\frac{dp}{dx} = q(x)$$

b. Encuentre una fórmula simple de recurrencia para determinar la derivada de $p(x)$ para $x = x_0$ en términos de los b_j. (Sugestión Note que $q(x)$ es un polinomio de grado $n - 1$ en x.)

*26 Escriba una proposición aritmética para efectuar las operaciones de (3.17).

27. Escriba proposiciones aritméticas para efectuar las operaciones de las cinco aproximaciones presentadas en la Parte D del Apéndice 2.

*28 Escriba

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) \simeq \frac{\pi}{2} x - \left(\frac{\pi}{2}\right)^3 \frac{x^3}{6} + \left(\frac{\pi}{2}\right)^5 \frac{x^5}{120} - \left(\frac{\pi}{2}\right)^7 \frac{x^7}{5040}$$

Encuentre los coeficientes de una aproximación racional de la forma

$$\operatorname{sen}\left(\frac{\pi x}{2}\right) \simeq \frac{b_1 x + b_3 x^3}{1 + c_2 x^2 + c_4 x^4}$$

Haga uso del hecho que el seno es una función impar [sen $(-x) = -\text{sen}\,(x)$] para explicar por qué se puede hacer $b_0 = b_2 = c_1 = c_3 = 0$ en la forma supuesta de la aproximación racional.

29. Determine los coeficientes de una aproximación racional de cos $(\pi x/2)$ en la forma

$$\cos\left(\frac{\pi x}{2}\right) \simeq \frac{b_0 + b_2 x^2}{1 + c_2 x^2}$$

30. Dada la serie

$$\tan x \simeq x + \frac{1}{3} x^3 + \frac{2}{15} x^5 + \frac{17}{315} x^7$$

encuentre los coeficientes de una aproximación racional de la forma

$$\tan x \simeq \frac{b_1 x + b_3 x^3}{1 + c_2 x^2 + c_4 x^4}$$

*31. Dada una aproximación racional de la forma

$$f(x) = \frac{a + bx + cx^2}{1 + dx}$$

determine la fracción continuada correspondiente de la forma

$$f(x) = k_1 + \cfrac{x}{k_2 + \cfrac{x}{k_3 + \cfrac{x}{k_4}}}$$

32. Dada una aproximación racional de la forma

$$f(x) = \frac{a + bx + cx^2}{1 + dx + ex^2}$$

determine la fracción continuada correspondiente de la forma

$$f(x) = k_1 + \cfrac{x}{k_2 + \cfrac{x}{k_3 + \cfrac{x}{k_4 + \cfrac{x}{k_5}}}}$$

33. Dada una aproximación racional de la forma

$$f(x) = \frac{a + bx}{1 + dx + ex^2}$$

determine la fracción continuada correspondiente de la forma

$$f(x) = \cfrac{1}{k_1 + \cfrac{x}{k_2 + \cfrac{x}{k_3 + \cfrac{x}{k_4}}}}$$

34. Partiendo del teorema del binomio

$$(1 - x)^{1/2} \simeq 1 - \frac{1}{2} x - \frac{1}{8} x^2 - \frac{3}{24} x^3 - \frac{15}{384} x^4 - \frac{105}{3840} x^5$$

a. Comprímala a una serie de términos que incluyan hasta x^3.

b. De la serie comprimida, determine una aproximación racional de la forma

$$(1 - x)^{1/2} \simeq \frac{a + bx + cx^2}{1 + dx}$$

10. Modify the program of problem 2 to set all coefficients of the Chebyshev expansion which are less than 5×10^{-8} equal to zero, before calling SUBROUTINE POWR. Now input $K = 13$ and the coefficients of $\sin(\pi/2x)$ from Example 1, Section 6.33. Compare the answers with those obtained in Example 1, Section 6.33. Can you trust your routine to do automatic telescoping of power series?

6.4 RATIONAL APPROXIMATIONS

In using the Taylor series for $\tan^{-1} x$ in Section 6.25, it was found that faster convergence was obtained if $\tan^{-1} x$ was first multiplied by a polynomial in x. This actually had the effect of representing $\tan^{-1} x$ as a rational function, that is, as a quotient of two polynomials in x. In general, rational functions can be found which give better accuracy than polynomials for the same number of terms, and which give much better accuracy than any of the approximations discussed in the preceding sections. The so-called rational Chebyshev approximations are of the form

$$f(x) \approx R_{mk}(x) = \frac{\sum_{j=0}^{m} a_j x^j}{\sum_{j=0}^{k} b_j x^j}.$$

where the a_j's and b_j's are chosen to minimize the maximum error in estimating $f(x)$ over some range of values for x. The process for determining the coefficients is too involved to reproduce here.*

The rational Chebyshev approximations are usually used for the intrinsic functions such as SIN, COS, EXP, etc., included in FORTRAN compilers, but are sufficiently difficult to generate that they are not of particular value to the analyst who desires to generate an approximating function that will receive only limited use.

6.5 ERROR ACCUMULATION IN EVALUATING POLYNOMIALS

In all the approximations discussed in the preceding sections, the actual computation involved in obtaining a function value is the evaluation of a polynomial (or possibly the quotient of two polynomials) of the form

$$P(x) = a_1 + a_2 x + \cdots + a_{n+1} x^n \tag{6-49}$$

* See, for example, Anthony Ralston and H. S. Wolf, *Digital Computers*, Vol. II, John Wiley & Sons, Inc., New York.

The degree of this polynomial was determined by consideration of the error involved in truncating a series and neglecting higher-order terms. In these error considerations, it was assumed that this truncation error was the only source of error; that is, the a_i's and x were exact numbers and the arithmetic was performed exactly. In fact, this is not the case. In a computer calculation, the a_i's and x will be approximate numbers and the arithmetic will be performed approximately. As was seen in Chapter 3, the errors from these sources can be of importance and must be considered. Indeed, it was shown that on a computer the value computed for an expression such as (6-49) could be different depending upon the order in which the terms are combined. The normal way of performing the calculation is to group the terms as

$$P(x) = a_1 + x(a_2 + x(a_3 + x(a_4 + \cdots + x(a_{n-1} + x(a_n + a_{n+1}x)) \quad))) \tag{6-50}$$

This grouping has two advantages. First, it requires a near-minimum number of multiplications and additions, so it is fast. Second, it tends to add the smallest numbers first, since the higher powers of x in the Taylor or Chebyshev series tend to have small coefficients. It was seen in Chapter 3 that this tends to work for improved accuracy.

The error propagation in evaluating (6-50) can be inferred from the rules of Section 3.6. The required calculation can be represented by the steps

$$S_{n+1} = a_{n+1} \tag{6-51}$$

$$S_i = a_i + xS_{i+1} \qquad \text{for } i = n, n-1, n-2, \ldots, 2, 1 \tag{6-52}$$

Let ΔS_i be the absolute error in S_i, r the roundoff error in the a_i's and the machine arithmetic, and Δx the absolute error in x. Then by the rules of Section 3.6, the relative error in the product xS_{i+1} is

$$\frac{\Delta x}{|x|} + \frac{\Delta S_{i+1}}{|S_{i+1}|} + r$$

and the relative error in S_i is

$$\frac{\Delta S_i}{|S_i|} \leq \frac{|a_i|}{|S_i|} r + \frac{|xS_{i+1}|}{|S_i|}\left(\frac{\Delta x}{|x|} + \frac{\Delta S_{i+1}}{|S_{i+1}|} + r\right) + r$$

$$\Delta S_i \leq |a_i| r + |x|\,|S_{i+1}| + |x|\,\Delta S_{i+1} + r|x|\,|S_{i+1}| + r|S_i| \tag{6-53}$$

Let us first apply this relationship for $i = 1$, obtaining a relation for ΔS_1, which is ΔP, the error in the final polynomial

$$\Delta P = \Delta S_1 = r(|a_1| + |S_1| + |x||S_2|) + \Delta x|S_2| + |x|\,\Delta S_2$$

Now let us apply it again, with $i = 2$, to the $|x|\,\Delta S_2$ term in this expression, obtaining

$$\begin{aligned}\Delta P = {} & r(|a_1| + |S_1| + |x||S_2|) + \Delta x\,|S_2| \\ & + r\,|x|(|a_2| + |S_2| + |x||S_3|) + |x|\,\Delta x\,|S_3| + |x|^2\,\Delta S_3\end{aligned}$$

and again with $i = 3$, $i = 4$, etc., until we finally obtain

$$\begin{aligned}\Delta P = {} & r(|a_1| + |S_1| + |x||S_2|) + \Delta x\,|S_2| \\ & + r\,|x|(|a_2| + |S_2| + |x||S_3|) + |x|\,\Delta x\,|S_3| \\ & + r\,|x|^2(|a_3| + |S_3| + |x||S_4|) + |x|^2\,\Delta x\,|S_4| \\ & + \cdots \\ & + r\,|x|^{n-1}(|a_n| + |S_n| + |x||S_{n+1}|) + |x|^{n-1}\,\Delta x\,|S_{n+1}| + |x|^n\,\Delta S_{n+1}\end{aligned}$$

Now

$$|S_1| \leqslant |a_1| + |a_2||x| + |a_3||x^2| + \cdots + |a_{n+1}||x^n|$$

and

$$\Delta S_{n+1} = r\,|a_{n+1}|$$

Using these relations and regrouping terms in the above relations, we have

$$\begin{aligned}\Delta P \leqslant {} & r(2\,|S_1| + 2\,|x||S_2| + \cdots + 2\,|x|^{n-1}|S_n| + |x^n||S_{n+1}|) \\ & + \Delta x(|S_2| + |x||S_3| + \cdots + |x|^{n-1}|S_{n+1}|)\end{aligned}$$

If we arbitrarily add in a term $|x^n|S_{n+1}$ and a term $\dfrac{\Delta x}{|x|}|S_1|$ to the right-hand side,

$$\Delta P \leqslant \left(2r + \frac{\Delta x}{|x|}\right)(|S_1| + |x||S_2| + |x^2||S_3| + \cdots + |x^n||S_{n+1}|)$$

Now

$$|S_1| \leqslant |a_1| + |a_2||x| + |a_3||x^2| + \cdots + |a_{n+1}||x^n|$$

and

$$|S_2| \leqslant |a_2| + |a_3||x| + \cdots + |a_{n+1}||x^{n-1}|$$

etc., and if S_1, S_2, and so on, are replaced by these approximations, the above becomes

$$\Delta P \leqslant \left(2r + \frac{\Delta x}{|x|}\right)(|a_1| + 2|a_2 x| + 3|a_3 x^2| + \cdots + (n+1)|a_{n+1}x^n|)$$

$$\frac{\Delta P}{|P|} \leqslant \left(2r + \frac{\Delta x}{|x|}\right)\frac{|a_1| + 2|a_2 x| + 3|a_3 x^2| + \cdots + (n+1)|a_{n+1}x^n|}{|a_1 + a_2 x + a_3 x^2 + \cdots + a_{n+1}x^n|} \tag{6-54}$$

It is seen that the relative error introduced by roundoff is dependent on the sizes and the signs of the a_i, and the number of terms. Consider the case where x is subject to the same sort of roundoff errors as the a_i, so that $\Delta x/|x| = r$ (in actual application x may be subject to additional errors from other sources, which is why it was treated separately in developing the formula). Also, let us limit our consideration for the moment to the case where all a_i's are positive. For this case, the above formula can be written

$$\frac{\Delta P}{P} \leqslant 3r\,\frac{(d/dx)(xP(x))}{P(x)} \tag{6-55}$$

or

$$\frac{\Delta P}{P} \leqslant 3r\left(1 + \frac{xP'(x)}{P(x)}\right) \tag{6-56}$$

If, for example, the polynomial $P(x)$ is approximating the exponential function, then $P(x) \approx e^x$, $P'(x) \approx e^x$, and we have

$$\frac{\Delta P}{P} \approx 3r(1 + x)$$

If we restrict ourselves to values of x less than one, the relation error in P [illegible] about one place less accuracy than the coefficients

If the a_i's are not all positive, relation (6-56) is not guaranteed to give an upper bound on the relative error. It will give a smaller value than relation (6-54). Sometimes it can be informative to apply (6-56) to cases where the a_i's are not all positive. If it indicates a large error, then (6-54) would indicate an even larger error, and we know that a problem situation exists. If it indicates a small error, then we cannot be convinced that there is no problem, however. Consider the case where $P(x)$ is approximating $\sin x$. Then application of (6-56) would give $P(x) \approx \sin x$, $P'(x) \approx \cos x$, and

$$\frac{\Delta P}{P} \leqslant 3r(1 + x \cot x)$$

The maximum value of $x \cot x$ is one, so that we obtain

$$\frac{\Delta P}{P} \leqslant 6r$$

indicating no serious accuracy problem in computing $\sin x$ from the polynomial expression. As we pointed out, this is not an actual upper bound, so it does not provide positive assurance that the error is small. In this particular case, an actual upper bound for the error in computing $\sin x$ can be obtained by noting that

$$\sinh x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \cdots$$

so that the expansion for the hyperbolic sine is the same as that for $\sin x$, except that all signs are positive. Hence we can say with rigor that for the approximation of $\sin x$ by a polynomial $P(x)$, relation (6-54) can be written

$$\frac{\Delta P}{P} \leqslant 3r \frac{(d/dx)(x \sinh x)}{\sin x}$$

$$= 3r \frac{\sinh x + x \cosh x}{\sin x}$$

The value of this expression increases as x increases. If we restrict our attention to values of x less than $\pi/2$, then the largest value, at $x = \pi/2$, is $(\pi/2)\cosh(\pi/2)$, or about 4. Hence

$$\frac{\Delta P}{P} \leqslant 12r$$

Hence the relative error in the sine can be [illegible]

that of the coefficients, or the values of the sine can have one less correct significant figure than the input coefficients used.

As mentioned earlier, several overestimates were made in deriving (6-54) as an upper bound for the error. A closer bound for the error can be found by using (6-53) directly. The error estimate can be included in a FORTRAN program right with the computation of the value itself. Let $A(1), A(2), \ldots, A(N+1)$ be the coefficients for the polynomial

$$P = A(1) + A(2)x + A(3)x^2 + \cdots + A(N+1)x^N$$

and let R be the relative error in the coefficient and DX the absolute error in X. Then the statements

```
      P=A(N+1)
      AX=ABS(X)
      AP=ABS(P)
      DP=R*AP
      DO 10 J=1,N
      I=N+1-J
      APOLD=AP
      P=A(I)+P*X
      AP=ABS(P)
   10 DP=R*(ABS(A(I))+AP+AX*APOLD)+DX*APOLD+AX*DP
```

will compute P, the value of the polynomial, and DP, the maximum error in this value.

EXERCISE 20

1. The Taylor series for $\ln(1 + x)$ is

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

Using relation (6-54), estimate the error in using 25 terms of this series to estimate $\ln(1 + x)$ for $0 < x < 1/2$, if seven-place arithmetic is used.

2. Write a program which will

a. Input the quantities A(1) = A(3) = A(5) = A(7) = A(9) = A(11) = 0, A(2) = 1, A(4) = −1/3!, A(6) = 1/5!, A(8) = −1/7!, A(10) = 1/9!, A(12) = −1/11!

b. Compute P and DP from the FORTRAN statements given in Section 6.5.

c. Use a library sine routine to compute sin x

d. [illegible] print sin(x), P(x), (sin x − P(x)), and DP(x) for x = 1, 1.5

e. [illegible]e about the error estimate DP compared to actual [illegible] in this case?

3. Perform problem 2, using inputs of

$$A(1) = 1,\ A(I) = \frac{1}{(I-1)!} \qquad \text{for } I = 2, 3, \ldots, 11.$$

Compare with the library routine EXP(x) for $x = .1, .2, \ldots, 1.0$.

6.6 ERROR PROPAGATION THROUGH FUNCTIONS

Thus far in this chapter we have concentrated on two sources of error in the evaluation of functions. The first was the truncation error in discarding the high-order terms in the approximating polynomial and the second was the roundoff error associated with the use of approximate values for coefficients and approximate arithmetic. Let us now assume that adequate measures have been taken to make these errors acceptably small, and concern ourselves with a different source of error, the intrinsic error in the inputs to a calculation. That is, assume that given x, we can find $f(x)$ accurately, and we wish to know the error in $f(x)$ when x is in error. To make the considerations more general, consider the case where we have several input quantities $u_1, u_2, u_3, \ldots, u_n$, which are to be used to calculate some quantity N. We can indicate this relationship by writing

$$N = f(u_1, u_2, u_3, \ldots, u_n)$$

Now if small changes are made in u_1, u_2, etc., by amounts Δu_1, Δu_2, etc., we can calculate a quantity called the differential of N by the relation

$$dN = \frac{\partial f}{\partial u_1}\Delta u_1 + \frac{\partial f}{\partial u_2}\Delta u_2 + \frac{\partial f}{\partial u_3}\Delta u_3 + \cdots + \frac{\partial f}{\partial u_n}\Delta u_n \qquad (6\text{-}57)$$

This quantity dN is approximately equal to ΔN, the error in N when u_1 is replaced by $u_1 + \Delta u_1$, u_2 by $u_2 + \Delta u_2$, etc. To demonstrate the meaning of this formula, we will use it to rederive the error rules for addition, subtraction, multiplication, and division given in Chapter 3.

For addition, we have

$$N = u_1 + u_2$$

so that

$$dN = \Delta u_1 + \Delta u_2$$

the situation expressed in Section 3.51.

For subtraction, we have

$$N = u_1 - u_2$$

so that

$$dN = \Delta u_1 - \Delta u_2$$

We must remember that Δu_1 and Δu_2 can be either positive or negative, so if we are interested in the maximum error, it is $|\Delta u_1| + |\Delta u_2|$.

For multiplication,

$$N = u_1 u_2$$

or, if we take logarithms, we can write

$$\ln N = \ln u_1 + \ln u_2$$

If now we take the total differential, we have

$$dN/N = \Delta u_1/u_1 + \Delta u_2/u_2$$

This rule corresponds to the statement in Section 3.53 concerning relative errors.

For division, if

$$N = u_1/u_2$$

then

$$\ln N = \ln u_1 - \ln u_2$$

and

$$dN/N = \Delta u_1/u_1 - \Delta u_2/u_2$$

As in subtraction, to obtain an estimate of the maximum possible error, we must allow for the case where Δu_1 and Δu_2 are of opposite sign, so we must consider

$$|\Delta u_1/u_1| + |\Delta u_2/u_2|$$

as an expression for estimating the error.

For more complicated expressions, equation (6-57) can be applied directly to give an expression for the error, remembering that in each case signs of the individual errors Δu_1, Δu_2, etc., should be chosen in such a way as to give the maximum result.

6.61 Error Accumulation for the Exponential Function

As a demonstration of the problem of error accumulation, let us apply relation (6-57) to the function

$$y = e^x$$

Applying relation (6-57) with $f(x) = e^x$, we have

$$dy = e^x \, \Delta x$$

where dy is the absolute error in y. The relative error is

$$dy/y = \Delta x$$

Hence the *relative* error in the computed value of e^x is equal to the *absolute* error in x itself. The disturbing feature of this result can be seen from the following example:

Example 1. Suppose $x = 100$, to three correct significant figures. What is the relative error in $y = e^x$?

The limit of the absolute error in x is

$$\Delta x = .5$$

Hence the relative error in y is .5, or 50%. The value of y has *no* significant figures!

The above example demonstrates that, even though our subroutines may be designed to compute to many correct significant digits, the problem of error accumulation is still with us when we use these subroutines.

6.62 Error Estimate by Formula

The example of Section 6.61 was indicative of the problem associated with the evaluation of any function of one or more independent quantities or approximate numbers. No calculation of this sort can be considered complete u[nless] some sort of assessment of the error has been made. For functions w[hich] are not too complex, the relation of Section 6.6 can be used for this pur[pose]. Further examples will be given to illustrate its use

Example 1. The function $y = a \sin b$ is to be calculated, where $a = 30.0$ and $b = .45$, the numbers being correct to the number of significant digits shown. Find the absolute and relative errors in y

By the formula of Section 6.6,

$$\begin{aligned} dy &= \frac{\partial y}{\partial a}\Delta a + \frac{\partial y}{\partial b}\Delta b \\ &= \sin b \, \Delta a + a \cos b \, \Delta b \\ &= (.435)(.05) + (30.0)(.900)(.005) \\ &= .022 + .14 = .16 \end{aligned}$$

or the absolute error is .16.

Since $y = (30.0)(.435) = 13.05$, the relative error is about .16/13, or roughly 1%.

Example 2. The function $y = a \sin b$ is to be calculated, where $a = 30.0$ and $b = \pi/6$, the number a being correct to three significant digits and the number b being exact. Find the absolute and relative errors in y

As before, we may write

$$dy = \frac{\partial y}{\partial a}\Delta a + \frac{\partial y}{\partial b}\Delta b$$

but since b is exact, $\Delta b = 0$, so the term $(\partial y/\partial b)\,\Delta b$ will drop out. This points up the fact that, whenever the function under consideration involves *exact* numbers, they can be treated as constants throughout, and the expression need not be differentiated with respect to them. All quantities which may be in error, whether constants or variables, should be treated as variables in applying the error formula of Section 6.6. (As indicated earlier, even the constants are subject to machine roundoff error. In the present case, and in many cases, the truncation error involved in roundoff is so small compared to other sources of error that it can safely be ignored.)

For the present problem, then,

$$\begin{aligned} dy &= \frac{dy}{da}\Delta a \\ &= \sin b \, \Delta a \\ &= (.5)(.05) \\ &\approx .025 \end{aligned}$$

[The] absolute error is .025 and the relative error is 025/15, or about 0.2%.

Example 3. The function $y = 2.0 \sin x + 3 \ln x$ is to be evaluated for $x = 1.26$. The constant 2.0 and the value of x are correct only to the number of significant digits shown. The constant 3 is exact. Find the absolute and relative errors in y.

Since the number 2.0 may be in error, it is best to replace it by a symbol before applying the error formula. Thus

$$\begin{aligned} y &= a \sin x + 3 \ln x \\ dy &= \Delta a \sin x + (a \cos x + 3/x)\,\Delta x \\ &= (.05)(.952) + [(2.0)(.306) + 3/1.26](.005) \\ &= .048 + [.612 + 2.38](.005) \\ &= .048 + .015 = .063 \end{aligned}$$

The absolute error is .063. Since

$$\begin{aligned} y &= (2.0)(.952) + 3(.231) \\ &= 1.90 + .69 = 2.59 \end{aligned}$$

the relative error is .063/2.59, or about 2%.

Example 4. Perform the calculation of Example 3 for $x = .65$.

Substituting in the formula of the previous exercise, we have

$$\begin{aligned} dy &= (.05)(.605) + [(2.0)(.796) + 3/.65](.005) \\ &= .030 + [1.59 + 4.62](.005) \\ &= .030 + .031 = .062 \end{aligned}$$

Again, the absolute error is about .062

However, since

$$\begin{aligned} y &= (2.0)(.605) + (3)(-.431) \\ &= 1.21 - 1.29 = -.08 \end{aligned}$$

the relative error is about .06/.08 = .75!

Although all the numbers used in this case were accurate to 2% or better, the final result had a 75% error! Closer inspection shows that this error came from the operation remarked as dangerous in Chapter 3, the subtraction of two nearly equal quantities. For $x = .65$, $\ln x$ is negative and the quantities $2 \sin x$ and $3 \ln x$ are very nearly equal in absolute value. The subtraction involved in finding y above resulted in loss of the two significant figures.

Example 5. The function $y = ke^{-\mu x}/x^2$ is to be evaluated for 100 values of x, ranging from 100 to 5000, and subject to an experimental error of one unit. The constants are $\mu = 3.0 \times 10^{-3}$ and $k = 1.3 \times 10^7$, each accurate to the number of significant digits indicated. Find the absolute and relative errors in y for a low, medium, and high value of x (use $x = 100$, 700, and 5000).

For functions such as this, where only multiplications, divisions, and powers are involved, it is convenient to take logarithms and then differentiate, thus obtaining relative error directly. Thus

$$\begin{aligned} \ln y &= \ln k - \mu x - 2 \ln x \\ dy/y &= \Delta k/k - \mu\,\Delta x - x\,\Delta\mu - 2\,\Delta x/x \end{aligned}$$

For $x = 100$,

relative error

$$= \frac{dy}{y} = \frac{.05 \times 10^7}{1.3 \times 10^7} - (3.0 \times 10^{-3})(-1) - (100)(-.05 \times 10^{-3}) - \frac{2(-1)}{100}$$

(signs of Δk, Δu, and Δx were chosen to maximize the error)

$$\begin{aligned} &= .038 + .003 + .005 + .02 \\ &= .066 \quad \text{or} \quad 7\% \end{aligned}$$

Since

$$y = 1.3 \times 10^7 e^{-(3.0 \times 10^{-3})(100)}/(100)^2$$

$$= 9.6 \times 10^2 \quad \text{or} \quad 960$$

the absolute error is

$$(.07)(9.6 \times 10^2) = .7 \times 10^2 \quad \text{or} \quad 70$$

For $x = 700$,

relative error

$$= \frac{dy}{y} = \frac{.05 \times 10^7}{1.3 \times 10^7} - (3.0 \times 10^{-3})(-1) - (700)(-.05 \times 10^{-3}) - \frac{2(-1)}{700}$$

$$= .038 + .003 + .035 + .003 = .079 \quad \text{or} \quad 8\%$$

$$y = 1.3 \times 10^7 e^{-(3.0 \times 10^{-3})(700)}/(700)^2 = 3.3$$

so the absolute error is

$$(.08)(3.3) \approx .3$$

For $x = 5000$,

relative error

$$= \frac{dy}{y} = \frac{.05 \times 10^7}{1.3 \times 10^7} - (3.0 \times 10^{-3})(-1) - (5000)(-.05 \times 10^{-3}) - \frac{2(-1)}{5000}$$

$$= .038 + .003 + .25 + .0004$$

$$= .29 \quad \text{or} \quad 29\%$$

$$y = 1.3 \times 10^7 e^{-(3.0 \times 10^{-3})(5000)}/(5000)^2$$
$$= 1.6 \times 10^{-7}$$

so the absolute error is about

$$(.29)(1.6 \times 10^{-7}) = .5 \times 10^{-7}$$

Comparing the values of the errors for the three different values of x gives us some feel for the errors throughout the range of values of x. The error is between 5 and 10% for the smaller values of x, and increases to about 30% at the extremely large values of x. We cannot be sure that the percentage error remains in the ranges indicated for all values of x, since we have studied only three particular values. If we wish a surer picture of the behavior of the error throughout the entire range of x, we can look at the expression for the relative error with numerical values substituted for all quantities except x:

$$\frac{dy}{y} = \frac{.05 \times 10^7}{1.3 \times 10^7} - (3.0 \times 10^{-3})(-1) - x(-.05 \times 10^{-3}) - 2(-1)/x$$

$$= .038 + .003 + .00005x + 2/x$$
$$= .041 + .00005x + 2/x$$

We can now study this expression as a function of x, making a plot of it if desired, and thus obtain a more complete picture of the relative error throughout the range of values of x. Figure 6-5 indicates this behavior. The relative error is .066 for $x = 100$, decreases to .061 at $x = 200$, then increases continuously to .29 at $x = 5000$.

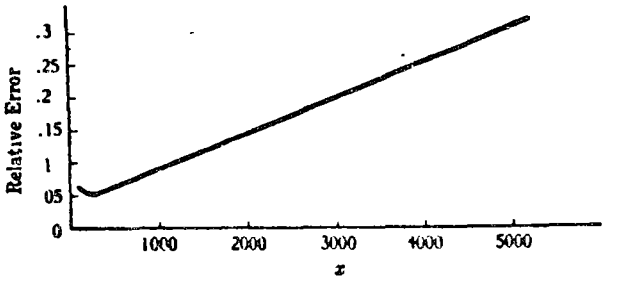


Figure 6-5

6.63 Error Estimate by Computer Trial

The application of the error formula (6-57) of Section 6.6 is straightforward as long as the functions involved are simple enough to be differentiated easily. For extremely complex functions, however, the process may be impracticable because of the difficulties involved in finding the derivatives or in evaluating the derivatives once found. In such cases the process of estimating the errors is often ignored completely. This is indeed unfortunate, since these are just the cases in which error accumulation is most likely to have some unexpected effect on the accuracy of the final answer. Instead of neglecting the problem, one should attempt to estimate the error by other methods. One method, quite adaptable for computer use, is to perform the calculation several times, each time varying one or more of the quantities which may be in error, and observing the effect on the final answer. Used properly, this method can give a more valid index of the error than does the error formula of Section 6.6. Again, as in that section, assume that the quantities $u_1, u_2, u_3, \ldots, u_n$ are to be combined to form some resulting number N, where

$$N = f(u_1, u_2, \ldots, u_n) \tag{6-58}$$

Suppose now that small changes $\Delta u_1, \Delta u_2, \ldots, \Delta u_n$ are made in the quantities $u_1, u_2, \ldots, u_n$. Then N will be changed to a new value $N + \Delta N$, given by

$$N + \Delta N = f(u_1 + \Delta u_1, u_2 + \Delta u_2, \ldots, u_n + \Delta u_n) \tag{6-59}$$

The Taylor expansion for a function of one variable given in Section 6.2 has its analogue for functions of several variables, the chief difference being that the ordinary derivatives are replaced by partial derivatives. This

expansion applied to expression (6-58) gives

$$N + \Delta N = f(u_1, u_2, \ldots, u_n) + \Delta u_1 \frac{\partial f}{\partial u_1} + \Delta u_2 \frac{\partial f}{\partial u_2} + \cdots + \Delta u_n \frac{\partial f}{\partial u_n}$$
$$+ \frac{1}{2}\left[(\Delta u_1)^2 \frac{\partial^2 f}{\partial u_1^2} + \cdots + (\Delta u_n)^2 \frac{\partial^2 f}{\partial u_n^2} + 2\Delta u_1 \Delta u_2 \frac{\partial^2 f}{\partial u_1 \partial u_2} + \cdots\right]$$
$$+ \frac{1}{3!}\left[(\Delta u_1)^3 \frac{\partial^3 f}{\partial u_1^3} + \cdots\right] + \cdots \qquad (6\text{-}60)$$

If the errors $\Delta u_1, \Delta u_2, \ldots, \Delta u_n$ are so small that we can neglect their squares, products, and higher powers, we can write (6-60) as

$$N + \Delta N \approx f(u_1, u_2, \ldots, u_n) + \Delta u_1 \frac{\partial f}{\partial u_1} + \Delta u_2 \frac{\partial f}{\partial u_2} + \cdots + \Delta u_n \frac{\partial f}{\partial u_n} \qquad (6\text{-}61)$$

or, subtracting (6-58) from (6-61),

$$\Delta N \approx \Delta u_1 \frac{\partial f}{\partial u_1} + \Delta u_2 \frac{\partial f}{\partial u_2} + \cdots + \Delta u_n \frac{\partial f}{\partial u_n} \qquad (6\text{-}62)$$

This is just the error formula (6-57) of Section 6.6. Thus that error formula is merely an approximation to the value of ΔN defined by relation (6-59). Direct application of relation (6-59) should give a better estimate of the error, since it does not neglect squares or products of errors. The task, once the computer program for evaluating the function f is prepared, is quite straightforward in concept. We merely run the calculation twice, once with input values $u_1, u_2, \ldots, u_n$, and once with input values $u_1 + \Delta u_1, u_2 + \Delta u_2, \ldots, u_n + \Delta u_n$. The difference in the results is then the absolute error. There is one difficulty, however. To obtain the maximum error we must choose the signs of the errors $\Delta u_1, \Delta u_2, \ldots, \Delta u_n$ so as to combine in the worst possible way. It is not usually possible to do this by inspection. Consequently, it is necessary first to change each one separately and observe how much N is increased or decreased. In a sense, this procedure is somewhat analogous to applying relation (6-62). Since by the definition of a partial derivative

$$\frac{\partial f}{\partial u_1} = \lim_{\Delta u_1 \to 0} \frac{f(u_1 + \Delta u_1, u_2, u_3, \ldots, u_n) - f(u_1, u_2, \ldots, u_n)}{\Delta u_1}$$

then, for well-behaved functions,

$$\frac{\partial f}{\partial u_1} \Delta u_1 \approx f(u_1 + \Delta u_1, u_2, u_3, \ldots, u_n) - f(u_1, u_2, \ldots, u_n)$$

Hence the difference between the value of N when $u_1 + \Delta u_1, \ldots$ are used in the calculation and that when $u_1, u_2, \ldots$

$(\partial f/\partial u_1)\,\Delta u_1$. If we do this for each variable and then add the absolute values of the resulting errors in N from all the calculations, we have an error estimate of the same type as is given by relation (6-62). To determine if the higher-order terms contained in relation (6-60) but ignored in relation (6-62) are important, it is usually wise to make a final calculation changing all the variables simultaneously. In each of the calculations in which only one variable has been changed, we observe whether N is increased or decreased, and then make a final calculation in which all variables are changed in directions chosen to produce the same direction change in N. The following set of steps outline the procedure:

(1) Calculate $N = f(u_1, u_2, \ldots, u_n)$.

(2) Calculate $N_i = f(u_1, u_2, \ldots, u_i + \Delta u_i, \ldots, u_n)$ for $i = 1$ to n.

(3) Calculate $N + \Delta N = f(u_1 + a_1 \Delta u_1, u_2 + a_2 \Delta u_2, \ldots, u_n + a_n \Delta u_n)$, where $a_i = +1$ if $N_i > N$ and $a_i = -1$ if $N_i < N$.

A word of caution should be given in connection with the use of the above procedure. The computation of the values of N and each N_i, and $N + \Delta N$, will be subject to the normal errors associated with the use of approximate numbers. If too small a value is chosen for the Δu_i, the change in N caused by this deliberate alteration may be disguised by the change induced by different roundoff errors. The value of the Δu_i must be chosen large enough that its effect is not lost in the "noise" of roundoff errors.

Example 1. Use the method just described to estimate the error for Example 3, Section 6.62.

In order to make the procedure clearer, the problem will be rewritten in the notation used in the description above. We wish to find

$$N = f(u_1, u_2)$$

where

$$f(u_1, u_2) = u_1 \sin u_2 + 3 \ln u_2$$

and

$$u_1 = 2.0 \qquad u_2 = 1.26$$
$$\Delta u_1 = .05 \qquad \Delta u_2 = .005$$

Following the steps above, we calculate:

(1) $N = f(u_1, u_2) = 2.0 \sin 1.26 + 3 \ln 1.26 = 2.59$.

(2) $N_1 = f(u_1 + \Delta u_1, u_2) = 2.05 \sin 1.26 + 3 \ln 1.26 = 2.64$,
$N_2 = f(u_1, u_2 + \Delta u_2) = 2.0 \sin 1.265 + 3 \ln 1.265 = 2.61$.

(3) Since $N_1 > N$, $a_1 = +1$. Since $N_2 > N$, $a_2 = +1$. Hence $N + \Delta N = f(u_1 + a_1 \Delta u_1, u_2 + a_2 \Delta u_2) = 2.05 \sin 1.265 + 3 \ln 1.265 = 2.66$, so that $\Delta N = 2.66 - 2.59 = .07$.

Example 2. The expression $y = \ln(a + \sqrt{b + e^{c \tan^{-1} x}})$ is to be calculated for values of x from 0 to 10 in order to make a graph. The values of the constants are $a = 2.0 \pm .1$, $b = 3.5 \pm .2$, $c = 1.0 \pm .1$. Draw a flow chart for the calculation, which includes an error estimate for each value of x.

Since in this problem we are allowed to choose the values of x, we may assume them to be precise, so that no error in x need be considered. The quantities a, b, and c are subject to errors $\Delta a = .1$, $\Delta b = .2$, and $\Delta c = .1$. Figure 6-6 shows the flow chart only for a single value of x. Additions to the chart to cause the calculation to be performed for a sequence of values of x are left to the reader.

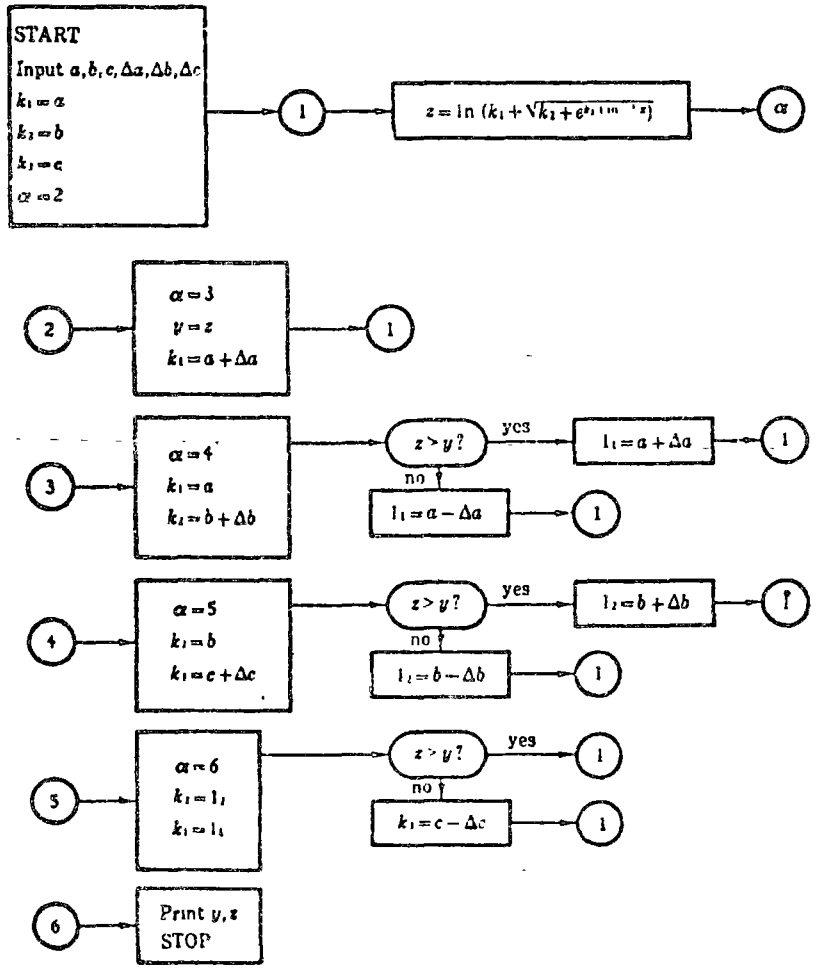


Figure 6-6

In the discussion of flow charts it was pointed out that the charts could be made detailed or crude as the occasion required. In the example just given, the calculation of the very complex function y was relegated to a single box. Most of the chart is devoted to outlining the selection of the values of k_1, k_2, and k_3 to be used in the calculation. The variable connector symbol was used to good advantage in this chart to indicate the reuse of the basic formula for y several times with different values for k_1, k_2, and k_3.

The FORTRAN program given below follows the flow chart rather closely but does include using a sequence of values of x:

```
      READ 101,A,B,C,DA,DB,DC,DX
      X=0
      L=10./DX
      DO 14 I=1,L
      FK1=A
      FK2=B
      FK3=C
      J=1
    1 Z=LOGF(FK1+SQRTF(FK2+EXPF(FK3*ATANF(X))))
      GO TO (2,3,6,9,13),J
    2 J=2
      Y=Z
      FK1=A+DA
      GO TO 1
    3 J=3
      FK1=A
      FK2=B+DB
      IF(Z-Y)4,4,5
    4 FL1=A-DA
      GO TO 1
    5 FL1=A+DA
      GO TO 1
    6 J=4
      FK2=B
      FK3=C+DC
      IF(Z-Y)7,7,8
    7 FL2=B-DB
      GO TO 1
    8 FL2=B+DB
      GO TO 1
    9 J=5
      IF(Z-Y)10,10,11
   10 FL3=C-DC
      GO TO 12
```

```
   11 FL3=C+DC
   12 FK1=FL1
      FK2=FL2
      FK3=FL3
      GO TO 1
   13 PRINT 101,Y,Z
   14 X=X+DX
      STOP
  101 FORMAT(7E10.4)
      END
```

This estimate by computer trial can be done much more simply in an interactive fashion from a remote terminal. In the language of Section 5.42, a suitable remote-terminal program is

```
1  1 PRINT, "A,B,C,X"
2    INPUT, A,B,C,X
3    Z=ALOG(A+SQRT(B+EXP(C*ATAN(X))))
4    PRINT, "Z=",Z
5    GO TO 1
6    END
```

Specific values of A, B, C, and X can be run as desired, using the computer as a supercalifragilistic desk calculator.

6.64 Limitations on Validity of the Error Estimate

It might seem that the procedure outlined above and illustrated in the example would provide an absolutely certain means of obtaining an upper estimate on the error. Surprisingly enough, such is *not* the case. It is possible, although unusual, for the error to be much larger than indicated by the error estimate. If the function f happens to behave in a sufficiently erratic fashion, it may be that the quantities

$$N = f(u_1, u_2, \ldots, u_n)$$

and

$$N + \Delta N = f(u_1 + \Delta u_1, u_2 + \Delta u_2, \ldots, u_n + \Delta u_n)$$

may be nearly equal, but that for some set of values of the u's intermediate between $u_1, u_2, \ldots, u_n$ and $u_1 + \Delta u_1, u_2 + \Delta u_2, \ldots, u_n + \Delta u_n$, the [illegible] has a very different value. More advanced studies show that the [illegible]

ΔN given above can be depended upon, that is, N will change only slowly as the u's change, if the following conditions are satisfied:

(1) All the partial derivatives $\partial f/\partial u_i$ exist and are continuous at the point $(u_1, u_2, \ldots, u_n)$.

(2) The errors $\Delta u_1, \Delta u_2, \ldots, \Delta u_n$ are sufficiently small. (We shall not try to define what is meant by "sufficiently." This is properly a subject for an advanced calculus course.)

An example will illustrate what may happen when these conditions are not satisfied.

Example 1. Compute $y = (1/16)\ln(\tan\sqrt{1 + x^2})^2$ for $x = 1.211$, and estimate the error, where the constants in the expression are exact, and x is accurate to the number of digits shown.

Let us attempt to estimate the error by computing y for the value x and for the value $x + \Delta x$, where $\Delta x = .0005$. For $x = 1.211$, $y = 1.02$. For $x + \Delta x = 1.2115$, $y + \Delta y = 1.14$. Hence we would be led to believe that the maximum error is $\Delta y = .12$. However, if we believe this, we are badly misled. For example, if the true value of x were 1.2113633, the value of y would be 1.39, a value differing from our original value by .37. Hence the maximum error is clearly more than .12. As a matter of fact, there is a value of x between 1.211 and 1.2115 for which y is infinite, so the error in y may be infinite! We can see this as follows: Since $\tan \pi/2 = \infty$, y is infinite when $\sqrt{1 + x^2} = \pi/2$. This is true when $x = \sqrt{-1 + (\pi/2)^2}$. The exact value of this number is between the two approximate numbers 1.2113633 and 1.2113634. The peculiar nature of this function in the region of interest is apparent from its graph, Figure 6-7. It has a vertical asymptote at the value $x = \sqrt{-1 + (\pi/2)^2}$, and the two values of x, 1.211 and 1.2115, happen to give nearly equal values of y on opposite sides of this asymptote. In this example the difficulty arises from the fact that the derivative dy/dx becomes

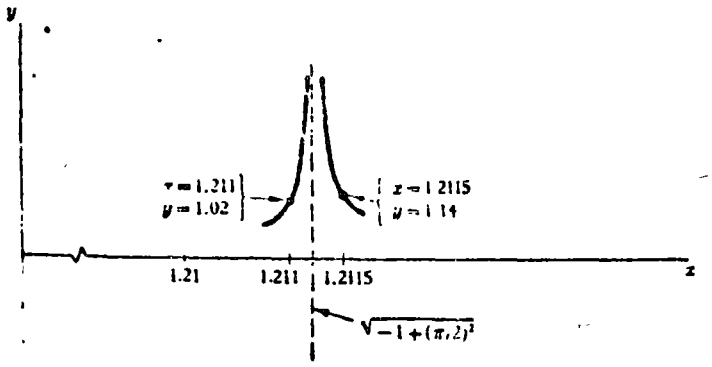


Figure 6-7

infinite within the interval x to $x + \Delta x$. This gives some idea as to what is meant by errors "sufficiently small" in condition (2). Δx must be small enough that the derivative dy/dx does not become inordinately large in the interval x to $x + \Delta x$.

The above example was included to demonstrate that calculation of an error estimate is no sure defense against the accidental acceptance of an answer grossly in error. The only sure protection is a detailed knowledge of the behavior of the function involved. This is no excuse, however, for failure to attempt to evaluate the effects of errors in the constants and variables involved in a calculation, both theoretically and experimentally by additional computer runs varying the values of any uncertain quantities. Stated another way, *part of the performance of any calculation is the testing of the sensitivity of the results to variations in the parameters involved in the problem.* In some cases, as in the preceding example, this testing of the sensitivity may be misleading, but such cases are fortunately rare. (They do however, tend to follow Gumperson's law, which, stated roughly, is: Those events which have a low probability of occurence tend to occur at the least opportune time. This law has been cited as the reason for the ringing of the telephone when one is in the bathtub, or failure of the car to start when one is about to drive to an important engagement. Gumperson reportedly met his death by being struck by an automobile. He was walking down the left side of the road in order to face traffic but was struck down from behind by a car driven by a visiting foreigner who was accustomed to driving on the left-hand side.)

6.65 Detailed Error Bounding in the Program

In any program it is possible to include error-estimation equations based on Table I of Section 3.6, and carry an error estimate right along with the calculation. This can be done as follows. Use an input parameter, say R, to represent the roundoff error. If the machine is doing seven-place arithmetic the value assigned to R would be R = 5.E−7. For each variable in the program, assign an associated variable, its absolute error (relative error could be chosen instead, the choice is immaterial). For example, if the variables are X, Y, and Z, carry also the variables DX, DY, and DZ. Each FORTRAN statement involving X, Y, and Z would be accompanied by one involving DX, DY, and DZ. For example,

```
Z=X+Y     or     Z=X-Y
```

would be accompanied by

```
DZ=DX+DY+R*ABS(Z)
```


and

```
Z=X*Y     or     Z=X/Y
```

would be accompanied by

```
DZ=ABS(Z)*(DX/ABS(X)+DY/ABS(Y))+R
```

Use of an implicit function, such as

```
Z=SIN(X)
```

would be accompanied by a statement based on equation (6-57), or in this case,

```
DZ=ABS(COS(X))*DX
```

Statements involving combinations of these quantities would be broken down into the individual operations and treated as above.

Example 1. Write a program with error bounding for the evaluation of the expression

$$y = ax + b\cos x + c$$

A FORTRAN expression for the calculation of y is

```
Y=A*X+B*COS(X)+C
```

The compiler will create a program which will evaluate this from left to right, first finding ax, then $b\cos x$ and adding, then adding c. The error bounding should be done in the same fashion, and can be done by a series of FORTRAN statements rather than a single one. A suitable program is

```
READ 101,A,B,C,X
READ 101,DA,DB,DC,DX,R
U=A*X
DU=ABS(U)*(DA/ABS(A)+DX/ABS(X)+R)
V=COS(X)
DV=ABS(SIN(X))*DX
W=B*V
DW=ABS(W)*(DB/ABS(B)+DV*ABS(V)+R)
V=U+W
DV=DU+DW+R*ABS(V)
```

```
      Y=V+C
      DY=DV+DC+R*ABS(Y)
      PRINT 101,Y,DY
  101 FORMAT(5E10.4)
      STOP
      END
```

Clearly, this program is longer, slower, and more painful to prepare than a simple one which computes y by a single FORTRAN statement and prints the answer. The time and trouble involved would be warranted only where one had reason to suspect serious accuracy problems.

6.7 LOSS OF SIGNIFICANT DIGITS IN SUBTRACTION

In Chapter 3 it was pointed out that the primary cause of loss of accuracy in calculations was the introduction of leading zeros in subtraction of two nearly equal numbers. In that chapter it was stated that, whenever such an event might occur, special programming precautions must be taken to avoid the difficulty or at least to make the programmer aware that a dangerous point in the calculation has arisen.

An error-bounding program as described in Section 6.6 will ordinarily detect the problem, although, as indicated there, such a technique is expensive in machine time and effort and is not guaranteed to flag the accuracy problems. In some cases it is possible to anticipate where accuracy loss during subtraction may occur, and make provision at those points to provide protection without encumbering the entire program with additional FORTRAN statements for error bounding. Some techniques for accomplishing this will be discussed.

6.71 Programmed Warning of Accuracy Loss

The first problem in protecting against accuracy loss in subtraction is to recognize when such an error may occur in a program. Any subtraction command (or addition command, since the machine adds algebraically) may be guilty if the numbers being handled happen to be of the right size. A program may work beautifully for certain sets of input and yet produce worthless answers for other sets because of loss of leading digits in subtractions. Sometimes it is possible to recognize during programming that such a danger exists, and in other cases it may be virtually impossible to recognize a danger spot. When a potential danger spot in the program can be recognized, programming to provide warning of accuracy loss may be advisable.

Suppose, for example, that a part of our program contains the statement

$$Y=A-B$$

Suppose further that we know that this part of the program will work satisfactorily for most sets of input numbers, but we suspect that in some cases the values of A and B at this point may be nearly equal. We fear that if as many as four leading zeros are produced by the subtraction, our final answer will not be trustworthy. We would like the machine to warn us if four figures are lost at this point in the calculation. It is an easy matter to write a section for the program which will accomplish this. We note first that, if four digits are lost, then the difference obtained as the result of the subtraction is roughly 10^{-4} times the minuend or subtrahend. The following statements will test for this occurrence and print out a warning if it does happen:

```
    2 Y=A-B
      IF(ABS(Y)-.0001*ABS(A))9,3,3
    9 PRINT 101
  101 FORMAT(26H1 ACCURACY LOSS,STATEMENT 2)
    3 (continuation of program)
```

After the calculation of $A-B$, a test is inserted which will determine if the difference is less than 10^{-4} times A. If it is, the print command is executed. If there is no excessive accuracy loss, the program continues without the printout. A section of flow chart which describes this operation might appear as in Figure 6-8.

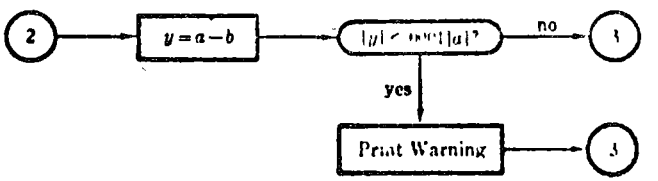


Figure 6-8

Example 1. The program given below will compute the third side a of a triangle, given sides b and c and the included angle A, the angle being denoted by the FORTRAN variable AA. The formula used is the law of cosines. Rewrite this program to give warning when subtraction results in the loss of two or more digits.

```
      READ 101,B,C,AA
      A=SQRT(B*B+C*C-2.*B*C*COS(AA))
      PRINT 101,A
  101 FORMAT(3E12.4)
      STOP
      END
```

A program which will give warning is

```
      READ 101,B,C,AA
      U=B*B+C*C
      V=2.*B*C*COS(AA)
      IF(ABS(U-V)-.01*ABS(U))2,2,3
    2 PRINT 102
      STOP
    3 A=SQRT(U-V)
      PRINT 101,A
      STOP
  101 FORMAT(3E12.4)
  102 FORMAT(29H ACCURACY LOSS IN COMPUTING A)
      END
```

In this example we arbitrarily settled on the loss of two leading digits as the danger point, the point at which we desire warning. It is fair to ask how such a requirement might come about. In a practical problem the loss of digits we could tolerate would be determined by accuracy of our knowledge of the input numbers b, c, and A and the required accuracy of the result. A careful error analysis using the general error formula of Section 6.6 would be quite difficult, but a loose line of reasoning following the accuracy theorems of Chapter 3 is sufficient to indicate how the error in the final result will depend on that of the input numbers. For example, suppose b, c, and A are each known to 1%. Then b^2, c^2, and bc have a relative error of about 2%. The absolute error in $\cos A$ is $\sin A\ \Delta A$, so the relative error in $\cos A$ is $\tan A\ \Delta A$. A little study of the expression $b^2 + c^2 - 2bc \cos A$ discloses that the quantities $(b^2 + c^2)$ and $(2bc \cos A)$ are nearly equal only when A is near zero and b is nearly equal to c. When A is near zero, $\tan A$ is small, so the relative error in $\cos A$ is small. Hence the term $2bc \cos A$ has a relative error of about 2%. From these values of relative error, we see that the terms $(b^2 + c^2)$ and $(2bc \cos A)$ each have about two significant figures. If one is lost in the subtraction, a^2 has one significant figure, or is accurate to about 10% (which means a is accurate to about 5%). If two significant digits are lost, a^2 may have no significant digits.

An error analysis of the type just given, while not at all precise, is usually sufficient to give guidance as to the acceptability of loss of leading significant digits in subtraction.

6.72 Programming to Avoid Accuracy Loss in Subtraction

In Section 6.71 it was demonstrated that it is sometimes possible to program a machine to give automatic warning in the event of serious accuracy loss in subtraction. It would be desirable to have the machine take some

corrective action instead of merely issuing a warning. This can be done in many cases. Let us first consider the case in which we have only one uncertain input number, x. Suppose we are evaluating the expression

$$y = f_1(x) - f_2(x)$$

where f_1 and f_2 are functions calculable to a high degree of accuracy by standard computer subroutines. Then by formula (6-57) of Section 6.6 the absolute error in y is

$$dy \leq |f_1'(x)\,\Delta x| + |f_2'(x)\,\Delta x|$$

and the relative error is

$$\frac{dy}{y} \leq \frac{|f_1'(x)| + |f_2'(x)|}{|f_1(x) - f_2(x)|}\,\Delta x$$

The relative error will be large due to loss of leading significant digits in subtraction when $f_1(x)$ and $f_2(x)$ are nearly equal. This will ordinarily occur near some value of x for which $f_1(x)$ and $f_2(x)$ are exactly equal. For example, suppose that for $x = a$

$$f_1(a) = f_2(a)$$

Then for values of x near $x = a$, say, for example, $x = a + h$ where h is small

$$f_1(a + h) \approx f_2(a + h)$$

and for small values of h we have subtraction problems. Now by Taylor's formula

$$f_1(a + h) = f_1(a) + hf_1'(a) + \text{terms involving higher powers of } h$$

and

$$f_2(a + h) = f_2(a) + hf_2'(a) + \text{terms involving higher powers of } h$$

Since h is small, we do not ordinarily need to carry these expansions past the first power in h to achieve sufficient accuracy. Thus, for h small, that is, for x near a, we have

$$y = f_1(x) - f_2(x) \approx h[f_1'(a) - f_2'(a)]$$

If the first-order terms were equal, $f_1'(a) = f_2'(a)$, it would be necessary to take the terms involving second powers of h in order to have a useful approximation for y. In cases in which a painstaking error analysis is warranted, the complete Taylor formula with remainder as given in Section 6.2 should be used.

Example 1. The function $y = 1 - e^{x-1}$ is to be calculated for values of x very near 1. Write an approximation which will have good accuracy for x sufficiently near 1.

Let us consider

$$y = f_1(x) - f_2(x)$$

where

$$f_1(x) = 1$$
$$f_2(x) = e^{x-1}$$

Let

$$x - 1 = h$$

Then

$$h[f_1'(1) - f_2'(1)] = h[0 - 1] = -h$$

so that the approximation is

$$y = 1 - x$$

Example 2. If an observer takes horizontal sighting over a smooth sea-level surface, how high is the line of sight at a distance of x miles from the observer? (The distance x is to be measured along the curved surface.) Write a program which will perform this calculation.

It can be seen from Figure 6-9 that the correct formula is

$$H = a \sec x/a - a$$

where a is the radius of the earth. In order to use the standard functions of

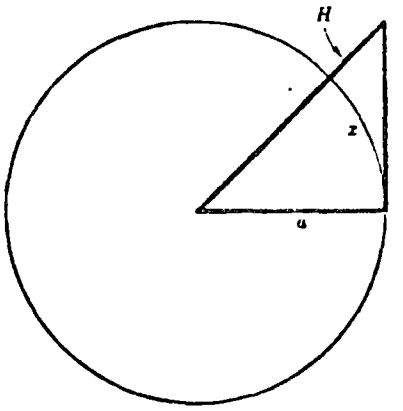


Figure 6-9

Chapter 4, we could write this as

$$H = \frac{a}{\cos x/a} - a$$

This formula appears quite straightforward, yet in using it we are apt to have accuracy difficulties. For example, using 4000 miles for the radius of the earth, consider the case where $x = 4$ miles. Then to seven correct significant digits,

$$\cos x/a = .9999995$$

The values of the various quantities involved, then, as they would be carried inside the computer, are

Quantity	Value Stored
a	$.4000000 \times 10^4$
$\dfrac{a}{\cos x/a}$	$.4000002 \times 10^4$
$\dfrac{a}{\cos x/a} - a$	$.2??????? \times 10^{-2}$

The six leading digits are lost in the subtraction, leaving at most one correct significant digit.

Note that in this case the accuracy problem results from the fact that the computer has only seven significant digits available, and not necessarily from inaccurate input data. It would not be at all unreasonable to ask for a program that would produce better accuracy for values of x on the order of

a few miles, and this can easily be arranged if we use the method described above, taking

$$f_1(x) = a \sec x/a, \qquad f_2(x) = a$$

When $x = 0$, we have $f_1 = f_2$, so we use the first nonzero term of a Taylor expansion about $x = 0$. We have

$$f_1(x) = a + a(x/a)^2/2$$
$$f_2(x) = a$$

Therefore,

$$H_1 = f_1(x) - f_2(x) = x^2/2a$$

A use of the remainder term to calculate the error would show that we obtain a more accurate value for H with this formula than with the original formula when $x < .01a$, or when $x < 40$ miles. Figure 6-10, then, indicates a good way of setting up the problem to work for all reasonable values of x.

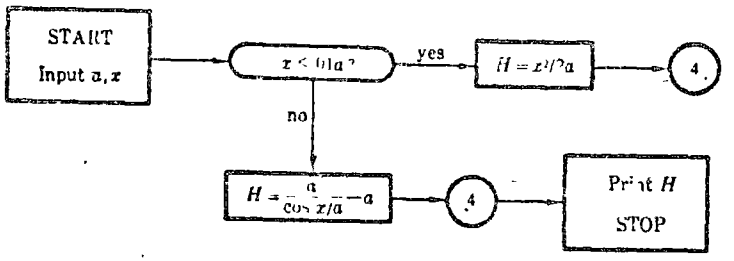


Figure 6-10

A FORTRAN program for this calculation is

```
      A=4000.
    1 READ 101,X
      IF(ABS(X)-.01*ABS(A))2,3,3
    2 FH=X*X/(2.*A)
      GO TO 4
    3 FH=A/COS(X/A)-A
    4 PRINT 102,X,FH
  101 FORMAT(E12.4)
  102 FORMAT(2E12.4)
      STOP
      END
```


An analogous procedure can be followed when several variables are involved. For Example 1, Section 6.61, we can derive an approximate relation for use when subtraction error is a problem as follows:

We need the approximation when A is nearly zero and b and c are nearly equal. Let us write

$$A = 0 + \Delta A$$
$$b = d + \Delta b$$
$$c = d + \Delta c$$

and assume henceforth that ΔA, Δb, and Δc are small. Then

$$a^2 = (d + \Delta b)^2 + (d + \Delta c)^2 - 2(d + \Delta b)(d + \Delta c)\cos \Delta A$$

Expanding, and discarding all terms having powers higher than the *second* in small quantities (all first-order terms drop out in this case, so we must keep the second-order terms):

$$\begin{aligned} a^2 &\approx d^2 + 2d\,\Delta b + \Delta b^2 + d^2 + 2d\,\Delta c + \Delta c^2 \\ &\quad -2(d^2 + d\,\Delta b + d\,\Delta c + \Delta b\,\Delta c)\left(1 - \frac{\Delta A^2}{2}\right) \\ &\approx \Delta b^2 + \Delta c^2 - 2\Delta b\,\Delta c + d^2\,\Delta A^2 \\ &= (\Delta b - \Delta c)^2 + d^2\,\Delta A^2 \end{aligned}$$

or, adding and subtracting d within the parentheses, and using $A = \Delta A$,

$$a^2 = (b - c)^2 + d^2 A^2$$

This relation says that, when b and c are nearly equal and A is small, we can determine a by considering it to be the hypotenuse of a right triangle, one of whose legs is $b - c$ and the other of whose legs is dA (or, to the same order of accuracy, bA or cA). Figure 6-11 illustrates this approximation.

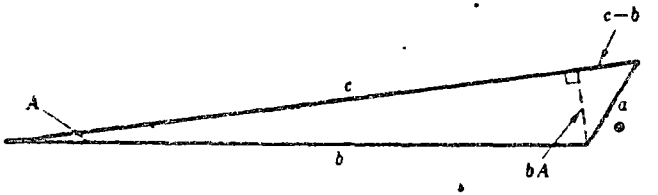


Figure 6-11

EXERCISE 21

1. Find the absolute and relative error in y for the following functions. The constants are accurate to the number of digits shown.

a.	$y = 1.00 + \ln x$	$x = 2.71 \pm .01$
b.	$y = 2.00 \cos x + 3.00 x^3$	$x = 4.00 \pm .005$ (the exponent is exact)
c.	$y = x^{-1.2}$	$x = 1, 10, 100, 1000$ (exact values)
d.	$y = e^x \sin x$	$x = 6.3 \pm .05$
e.	$y = 1.00 \cos x + .60 \cos 2x + .30 \cos 3x$	$x = 1.0 \pm .05$ $1.5 \pm .05$ $2.4 \pm .05$

2. In a circle of radius a, a chord is drawn which subtends a central angle θ. Write the expression for the distance from the center of the chord to the edge of the circle, x. Draw a flow chart for a program that will calculate to four significant digits for *all* values of θ less than $\pi/2$.

3. Draw a flow chart for a calculation that will find y accurate to four significant figures for *all* values of x between 0 and 1. Write a FORTRAN program to calculate and print y for 1000 equally spaced values of x.

a. $y = x - \sin x$ b. $y = \tan x - \sin x$

c. $y = e^x - \cos x$ d. $y = \cos x - 2\ln(1 + x)$

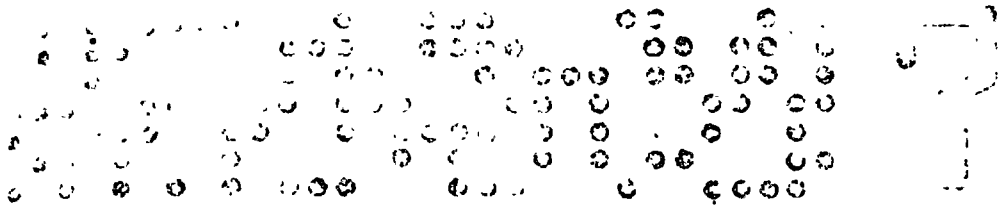
e. $y = \dfrac{\tan x - 2\ln(1 + x)}{\sin x - \ln(1 + x)}$ f. $y = \dfrac{\cos \pi x/2 - 1}{x - 1}$

4. Write a program for Example 2, Section 6.63, with detailed error bounding.

5. Write a program for determining a side of a triangle from the cosine law

$$a^2 = b^2 + c^2 - 2bc \cos \theta$$

with detailed error bounding. Compute a and Δa from this program for $b = 1.$, $\Delta b = .1$, $c = 1$, $\Delta c = .1$, $\theta = .1$, $\Delta\theta = .01$.



Quadrature

7.1 INTRODUCTION

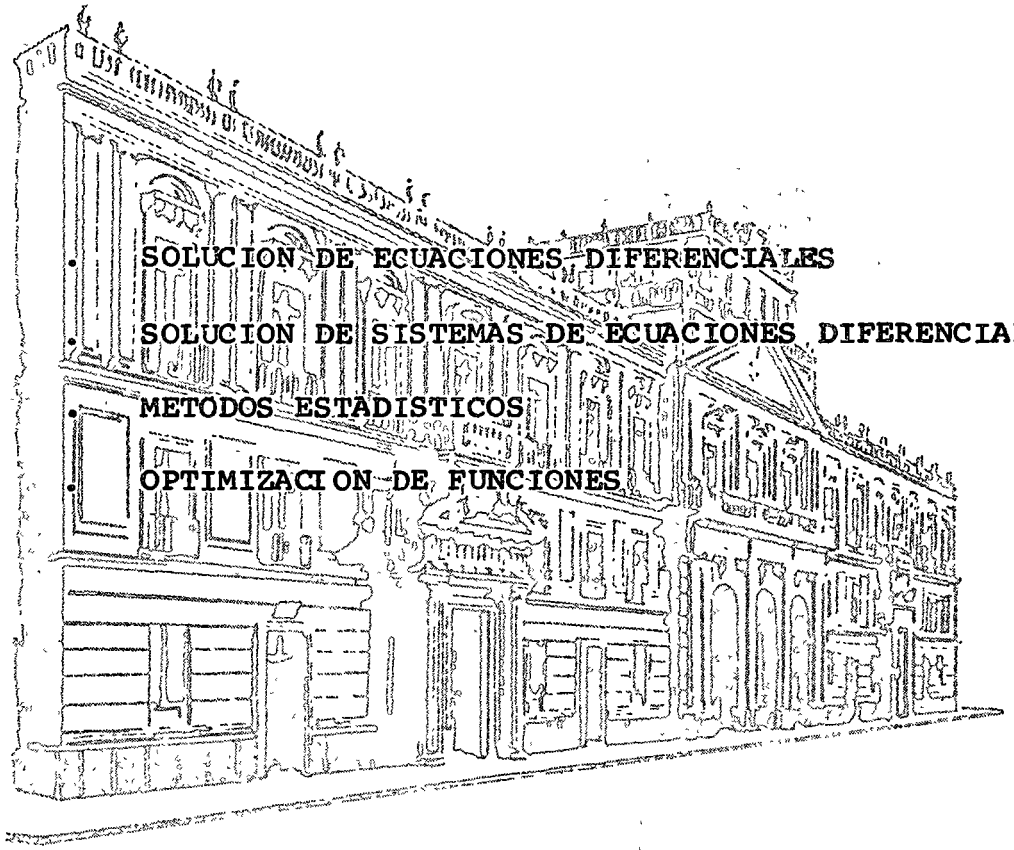
The preceding chapters have been devoted largely to describing the digital computer and the types of operations it can perform. The remaining chapters are devoted to topics ordinarily treated in a numerical analysis course.

Strangely enough, it seems proper to make quadrature, or integration, the first such topic to be covered. There are two reasons for doing this. First, quadrature as ordinarily done on the computer is a very direct extension of the material of Chapters 5 and 6. Second, quadrature is one of the fields of applied mathematics most markedly affected by the advent of the computer.

In elementary calculus the methods for differentiation and integration of various functions are taught. Generally speaking, differentiation turns out to be the more easily performed of the two operations. Physicists and engineers, then, sometimes find it strange that mathematicians usually consider integration to be the "nicer" process. In particular, the mathematician is inclined to regard a problem as solved once he presents the answer in terms of a quadrature, that is, a definite integral of a known function, between known limits. After all, such an integral merely represents a number. To the physicist or engineer, however, the numerical value of this number may be a matter of considerable concern. Before the advent of the computer, the task of evaluating any but the most simple definite integrals was imposing, to say the least, and was insurmountable in many cases. The digital computer has produced a marked change in this situation. Numerical evaluation of large classes of definite integrals is a process well within the capabilities of even the slower computers. However, there are still problems involving quadrature in two or more dimensions which would require inordinate amounts of time on even the fastest of present day computers.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL



- SOLUCION DE ECUACIONES DIFERENCIALES
- SOLUCION DE SISTEMAS DE ECUACIONES DIFERENCIALES
- METODOS ESTADISTICOS
- OPTIMIZACION DE FUNCIONES

ARMANDO TORRES FENTANES

ABRIL DE 1976.

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels: 521-40-23 521-73-35 5123-123

La ecuación VI.1 nos da la fórmula recursiva de Euler. Para aplicar el método se requiere que Δx sea pequeño y además contar con un punto de inicio (X_o, Y_o). El error producido es del orden de ΔX^2.

b) Euler modificado

El procedimiento básico es el mismo solo que para cada Y_{i+1} se hace una serie de iteraciones con los valores obtenidos sucesivamente de Y_{i+1} a - fin de obtener el valor más exacto de Y_{i+1}.

Al tener :

$$Y_{i+1} = Y_i + Y' \Big|_{(X_i, Y_i)} \Delta X \qquad (VI.2)$$

se efectúan las siguientes iteraciones :

$$Y'_{i+1} = \mp(X_{i+1}, Y_{i+1})$$

$$\hat{Y}_{i+1} = Y_i + \frac{(Y'|_i + Y'|_{i+1})}{2} \Delta X$$

$$\hat{Y}'_{i+1} = \mp(X_{i+1}, \hat{Y}_{i+1})$$

$$\hat{\hat{Y}}_{i+1} = Y_i + \frac{(Y'|_i + \hat{Y}'|_{i+1})}{2} \Delta X$$

y así sucesivamente hasta que :

$$\left| \hat{\hat{Y}}_{i+1} - \hat{Y}_{i+1} \right| < \varepsilon \qquad (VI.3)$$

al cumplirse, se procede a obtener Y_{i+2} y así sucesivamente.

Al igual que en el método anterior es necesario emplear incrementos (Δx) pequeños. El error producido es del orden Δx^3.

c) Método de Runge - Kutta

Este método utiliza las fórmulas de integración antes vista para llegar a la obtención de su propia fórmula recursiva. Dicho proceso es bastante laborioso por lo que no se tratará.

La solución para una ecuación diferencial de primer orden $Y' = f(x, y)$ está dada por :

$$Y_{n+1} = Y_n + \Delta Y_n$$

donde :

$$\Delta Y_n = \frac{\Delta X}{6} (K_0 + 2K_1 + 2K_2 + K_3)$$

$$K_0 = f (X_n, Y_n)$$

$$K_1 = f (X_n + \frac{\Delta X}{2} , Y_n + \frac{K_0 \Delta X}{2})$$

$$K_2 = f (X_n + \frac{\Delta X}{2} , Y_n + \frac{K_1 \Delta X}{2})$$

$$K_3 = f (X_n + \Delta X, Y_n + K_2 \Delta X)$$

La fórmula anterior es la de Runge-Kutta de 4o. orden, hay otras - fórmulcs con mayor cantidad de términos que se obtienen empleando diferencias de mayor orden al deducir la fórmula.

Los parámetros K_i representan la pendiente de la función en los -- puntos en que se está evaluando. El método da un error del orden de ΔX^5 y es uno de los más precisos.

Ejemplo

Obtener la solución de la ecuación diferencial $Y' = 1 - X + 4y$ para - 5 puntos consecutivos empleando los métodos de Euler, Euler mejorado y Runge - Kutta usando un incremento $\Delta x = 0.1$. Comparar dichos valores con la solución real si $X_o = 0$, $Y_o = 1$.

Sol.

La solución exacta está dada por :

$$Y' - 4Y = 1 - X$$

$$Y_h = Ce^{4X}$$

$$Y_p = A + BX$$

$$\therefore \; -4A - 4BX = 1 - X$$

$$A = -\frac{1}{4} \quad ; \quad B = \frac{1}{4}$$

$$Y(X) = Ce^{4X} - \left(\frac{1}{4} + \frac{1}{4}X\right)$$

$$Y(0) = 1 = C - \frac{1}{4}$$

$$C = \frac{5}{4}$$

$$Y(x) = \frac{5}{4} e^{4x} - \frac{1}{4} + \frac{1}{4} x$$

las fórmulas de solución para los métodos son :

$$Y_n = Y_{n-1} + Y']_{n-1} \Delta x \qquad \text{(Euler)}$$

$$\left.\begin{aligned} Y_n &= Y_{n-1} + Y']_{n-1} \Delta x \\ \hat{Y}_n &= Y_{n-1} + \frac{(Y']_{n-1} + Y']_n}{2} \Delta x \end{aligned}\right\} \text{Euler mejorado}$$

$$Y_n = Y_{n-1} + \frac{\Delta x}{6} [K_1 + 2K_2 + 2K_3 + K_4]$$

$$K_1 = f(x_{n-1}, Y_{n-1})$$

$$K_2 = f\left(x_{n-1} + \frac{\Delta x}{2}, Y_{n-1} + \frac{K_1}{2} \Delta x\right) \qquad \text{Runge Kutta}$$

$$K_3 = f\left(x_{n-1} + \frac{\Delta x}{2}, Y_{n-1} + \frac{K_2}{2} \Delta x\right)$$

$$K_4 = f(x_{n-1} + \Delta x, Y_{n-1} + K_3 \Delta x)$$

las soluciones se muestran en la siguiente tabla:

K	X_k	Euler	E. Mej.	R.Kutta	Real
0	0	1.	1.	1.	1.
1	0.1	1.5	1.595	1.608	1.609
2	0.2	2.19	2.463	2.505	2.505
3	0.3	3.146	3.737	3.829	3.830
4	0.4	4.774	5.609	5.792	5.794
5	0.5	6.324	8.369	8.709	8.712

d) Diferencias finitas

Este método se emplea cuando se tienen problemas con valores en la frontera.

El procedimiento consiste en lo siguiente : dividir el intervalo de integración en "n" espacios iguales, emplear las fórmulas de derivación de diferencias finitas en la ecuación diferencial (todas las diferencias deben ser del mismo orden), substituir las condiciones de frontera y por último resolver el sistema de ecuaciones planteado. Se tiene que aplicar el operador diferencial a todos los pivotes del intervalo.

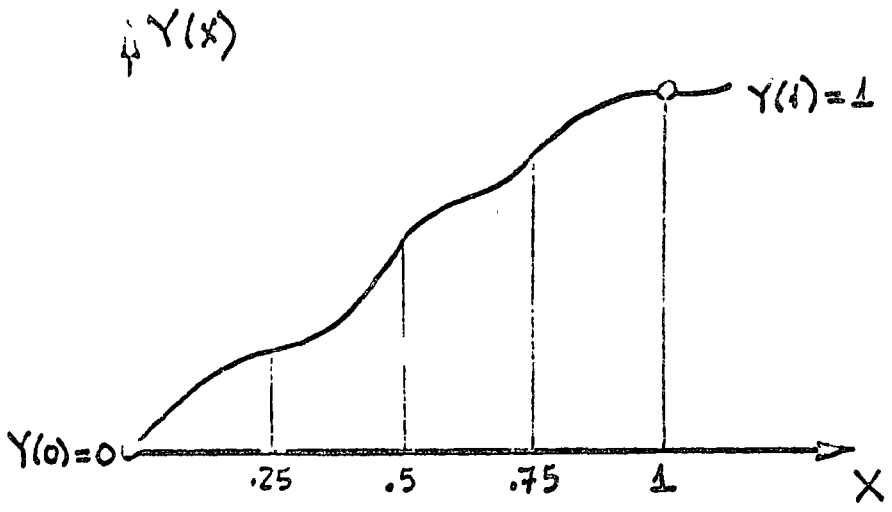
Ejemplo

Resolver la ecuación diferencial $\frac{d^2 y_2}{d X^2} - y = 0$, en el intervalo (0,1) si $y(0) = 0$, $y(1) = 1$.

Sol.

Se divide el intervalo en "n" partes iguales, sean 4 en este caso :

$$\Delta X = \frac{1 - 0}{4} = 0.25$$



empleando diferencias de 2o. orden :

$$Y_i'' = \frac{1}{(\Delta X)^2} \left[Y_{i-1} - 2 Y_i + Y_{i+1} \right]$$

substituyendo en la ecuación diferencial :

$$\frac{1}{\Delta X^2} \left[Y_{i-1} - 2 Y_i + Y_{i+1} \right] - Y_i = 0$$

$$Y_{i-1} - 2.0625\ Y_i + Y_{i+1} = 0 \qquad \text{(VI.4)}$$

las condiciones de frontera son :

$$Y_o = 0$$

$$Y_4 = 1$$

aplicando VI.4 en los pivotes :

$$X_1 = 0.25$$

$$Y_o - 2.0625\, Y_1 + Y_2 = 0$$

$$-2.0625\, Y_1 + Y_2 = 0 \qquad \text{(VI.5)}$$

$$X_2 = 0.5$$

$$Y_1 - 2.0625\, Y_2 + Y_3 = 0 \qquad \text{(VI.6)}$$

$$X_3 = 0.75$$

$$Y_2 - 2.0625\, Y_3 + Y_4 = 0$$

$$Y_2 - 2.0625\, Y_3 = -1 \qquad \text{(VI.7}$$

el sistema de ecuaciones es :

$$\begin{aligned} -2.062\, Y_1 + Y_2 &= 0 \\ Y_1 - 2.062\, Y_2 + Y_3 &= 0 \\ Y_2 - 2.062\, Y_3 &= -1 \end{aligned}$$

de donde :

$$Y_1 = 0.216$$

$$Y_2 = 0.445$$

$$Y_3 = 0.701$$

NOTA: Cuando se trata de ecuaciones diferenciales de mayor orden y se - cuenta como condiciones $y'' = 0$, etc., hay que substituir en dichas ecuaciones las fórmulas de diferencias y despejar de ahí las condiciones de frontera - desconocidas.

VII) SOLUCION SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS DE PRIMER ORDEN.

Una ecuación diferencial de orden "n" de la forma :

$$A_n Y^{(n)}(x) + A_{n-1} Y^{(n-1)}(x) + \ldots + A_1 Y'(x) + A_o Y(x) = f(x)$$

se puede descomponer en un sistema de "n" ecuaciones diferenciales ordinarias de primer orden mediante el siguiente cambio de variables.

$$Y_1 = Y$$

$$Y_2 = \dot{Y}_1$$

$$Y_3 = \dot{Y}_2$$

$$\vdots$$

$$\dot{Y}_n = \frac{1}{A_n}\left[f(x) - A_{n-1} Y_{n-1} - \ldots - A_o Y_1\right]$$

es por ello que solo se hablará de la solución de sistemas de ecuaciones diferenciales ordinarias de primer orden.

a) Runge - Kutta

Con este método solo trataremos la solución de sistemas de 2 ecuaciones diferenciales y la metodología es similar a la empleada para resolver una ecuación diferencial de primer orden. Como la demostración cae fuera de los propósitos del curso solo se dará la metodología.

Sea un sistema de dos ecuaciones diferenciales de primer orden:

$$\frac{dy}{dx} = f(x, y(x), z(x))$$

$$\frac{dz}{dx} = g(x, y(x), z(x))$$

con las condiciones iniciales :

X_o, $Y(X_o)$, $Z(X_o)$.

Empleando el método de Runge - Kutta de 4o. orden :

$$Y_{i+1} = Y_i + \frac{\Delta X}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$

$$K_1 = f(X_i, Y_i, Z_i)$$

$$K_2 = f\left(X_i + \frac{\Delta X}{2}, Y_i + \frac{K_1 \Delta X}{2}, Z_i + \frac{q_1 \Delta X}{2}\right)$$

$$K_3 = f\left(X_i + \frac{\Delta X}{2}, Y_i + \frac{K_2 \Delta X}{2}, Z_i + \frac{q_2 \Delta X}{2}\right)$$

$$K_4 = f(X_i + \Delta X, Y_i + K_3 \Delta X, Z_i + q_3 \Delta X)$$

$$Z_{i+1} = Z_i + \frac{\Delta X}{6}(q_1 + 2q_2 + 2q_3 + q_4)$$

$$q_1 = g(X_i, Y_i, Z_i)$$

$$q_2 = g\left(X_i + \frac{\Delta X}{2}, Y_i + \frac{K_1 \Delta X}{2}, Z_i + \frac{q_1 \Delta X}{2}\right)$$

$$q_3 = g\left(X_i + \frac{\Delta X}{2}, Y_i + \frac{K_2 \Delta X}{2}, Z_i + \frac{q_2 \Delta X}{2}\right)$$

$$q_4 = g(X_i + \Delta X, Y_i + K_3 \Delta X, Z_i + q_3 \Delta X)$$

Ejemplo

Transformar la siguiente ecuación diferencial ordinaria de cuarto orden en un sistema de 4 ecuaciones diferenciales de primer orden:

$$4y'''' + 3y''' + \frac{1}{2}y'' + 2y' - 3y = 5\cos t$$

$$Y_1 = Y$$

$$Y_2 = \dot{Y}_1$$

$$Y_3 = \dot{Y}_2$$

$$Y_4 = \dot{Y}_3$$

$$\dot{Y}_4 = \frac{1}{4}\left[5\cos t - 3Y_4 - \frac{1}{2}Y_3 - 2Y_2 + 3Y_1\right]$$

b) Variación de parámetros

Dado un sistema de ecuaciones diferenciales de primer orden con coeficientes constantes de la forma :

$$\dot{\underline{X}} = \underline{A}\,\underline{x} + \underline{B}\,\underline{u}, \quad \underline{X} = \underline{X}(t)$$

la solución está dada por :

$$\underline{X}(t) = e^{\underline{A}(t-t_o)}\underline{X}_o + \int_{t_o}^{t} e^{\underline{A}(t-t')}\underline{B}\,\underline{U}(t')\,dt' \qquad \text{(VII.0)}$$

donde $e^{\underline{A}(t-t_o)}$ se conoce como la matriz de transición y es igual a :

$$e^{\underline{A}(t-t_o)} = \underline{I} + \frac{\underline{A}}{1!}(t-t_o) + \frac{\underline{A}^2}{2!}(t-t_o)^2 + \ldots \qquad \text{(VII.1)}$$

Si se consideran incrementos constantes de la variable independiente (Δt) y se aproximan las funciones $\underline{u}(t)$ por paralelogramos o trapezoides, se puede obtener una serie similar a VII.1 para el término :

$$\int_{t_o}^{t} e^{\underline{A}(t-t')} \underline{B}\,\underline{U}(t')\,dt'$$

Dichas series son fáciles de obtener en computadora y para su convergencia se pide que todos los elementos de las matrices obtenidas en dos iteraciones sucesivas, sean menores que una cierta ε prefijada de antemano.

Este método es bastante exacto si se fija un incremento Δt muy pequeño y se establece un buen criterio de convergencia ε para las series.

Para obtener manualmente la matriz de transición $e^{\underline{A}t}$ hay dos métodos:

- en el dominio de la transformada de Laplace :

$$e^{\underline{A}t} = L^{-1}\left\{\left[\underline{I}\,S - \underline{A}\right]^{-1}\right\} \quad *$$

- en el dominio del tiempo, se necesita obtener los valores característicos de la matriz $\underline{A}$, la cual en términos generales es de orden $n \times n$, y - aplicar teoremas de matrices para llegar al siguiente sistema de ecuaciones con incógnitas α_i :

---

* $L^{-1}(\phi)$, indica la antitransformada de Laplace de ϕ.

$$e^{\underline{A}(t)} = \sum_{i=0}^{n-1} \alpha_i \underline{A}^i$$

$$e^{\lambda_1 t} = \sum_{i=0}^{n-1} \alpha_i \lambda_1^{i+1}$$

$$e^{\lambda_2 t} = \sum_{i=0}^{n-1} \alpha_i \lambda_2^{i+1}$$

$$\vdots$$

$$e^{\lambda_n t} = \sum_{i=0}^{n-1} \alpha_i \lambda_n^{i+1}$$

VIII) METODOS ESTADISTICOS Y PROBABILISTICOS

a) Generación de números aleatorios por el método de la congruencia lineal multiplicativa.

Números aleatorios son aquellos que se generan al azar y que se utilizan para efectuar simulaciones, predicciones, etc. Una característica que deben cumplir los números aleatorios creados para dichos fines es que sean reproducibles para efectos de comprobar resultados. Uno de los métodos más eficaces es de la congruencia lineal, el cual se describe a continuación.

Elegir 4 parámetros :

X_o valor inicial ó semilla, $X_o \geq 0$

A multiplicador, $A \geq 0$

c incremento , $C \geq 0$

m módulo , $m > X_o$
$m > A$
$m > C$

utilizar la ecuación iterativa :

$$X_{n+1} = (A X_n + C) \bmod m$$

donde

$Z \bmod m$ = residuo de dividir $Z \div m$

todos los parámetros elegidos deben ser números enteros.

La longitud del ciclo de números aleatorios depende del módulo m, por lo tanto es aconsejable elegirlo en la siguiente forma :

$m = p^b$ $\begin{cases} p = 2, \text{ en computadoras binarias} \\ p = 10, \text{ en computadoras decimales} \\ b = \text{número de bits por palabra de la computadora.} \end{cases}$

Para mayor facilidad se suele emplear $c = o$ y en casos de computadoras binarias se obtienen mejores resultados si:

$a = 8t + 3$, $t = o, 1, 2, \ldots$

a debe ser comparable con m

X_o debe ser entero impar no divisible $\div$ 5

Los números aleatorios que se obtienen con este método quedan comprendidos en el intervalo (o, m); pero la mayoría de las veces se necesitan números aleatorios uniformemente distribuidos (r_n) entre 0 y 1, para ello se hace lo siguiente:

$$r_n = \frac{X_n}{m}$$

Ejemplo

Genere v.a. en el intervalo (0, 1) si:

$X_o = 3$

$a = 4$

$c = 5$

$m = 12$

Sol.

La fórmula recursiva es:

$$X_n = (A X_{n-1} + C) \bmod m$$

aplicando queda :

$$X_1 = (12 + 5) \bmod 12$$

$$= 17 \bmod 12$$

$$= 5$$

$$r_1 = 5/12 = 0.4166$$

$$X_2 = (20 + 5) \bmod 12$$

$$= 25 \bmod 12$$

$$= 1$$

$$r_2 = 1/12 = 0.0833$$

$$X_3 = (4 + 5) \bmod 12$$

$$= 9 \bmod 12$$

$$= 9$$

$$r_3 = 9/12 = 0.75$$

$$X_4 = (36 + 5) \bmod 12$$

$$= 41 \bmod 12$$

$$= 5$$

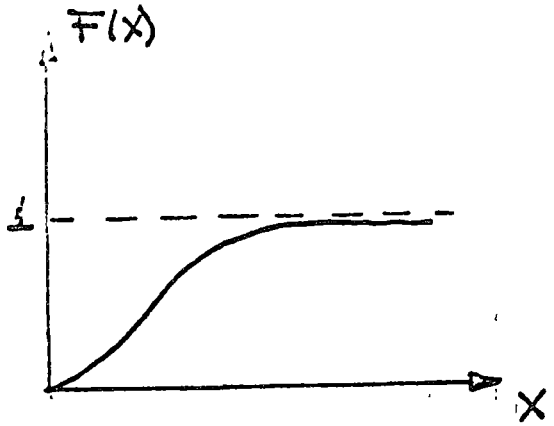
$$r_4 = 5/12 = 0.4166$$

b) Método de la transformada inversa

Este proceso se emplea para obtener la configuración de la función densidad de probabilidad (f.d.p.) de una variable aleatoria (v.a.) X cuando solo se conoce su función de densidad acumulada (f.d.a.) ya sea en forma discreta

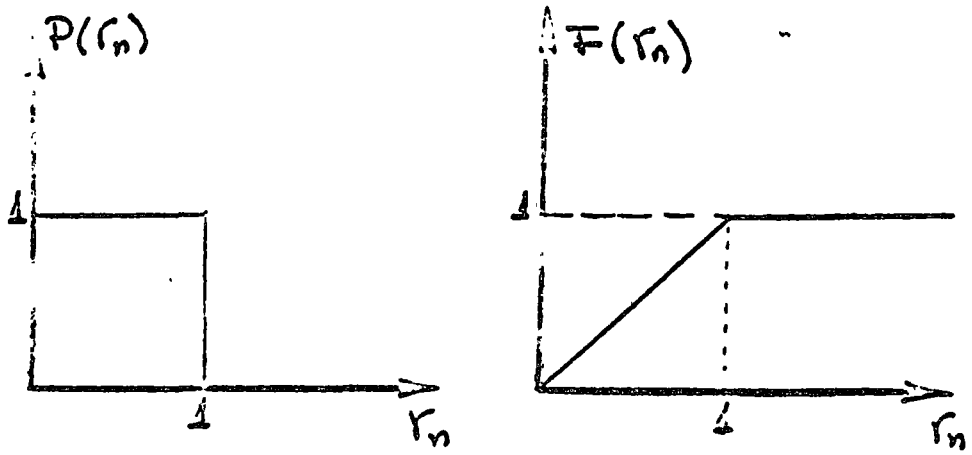
ó contínua . Numéricamente el método trabaja con valores discretos, por lo que hay que discretizar f.d.a. si esta es contínua.

Las características de una f.d.a. son :



$$F(X) = \int f(X)\, dX$$

y las de una variable aleatoria uniformemente distribuida entre (0, 1) son :



Analíticamente lo que se hace es :

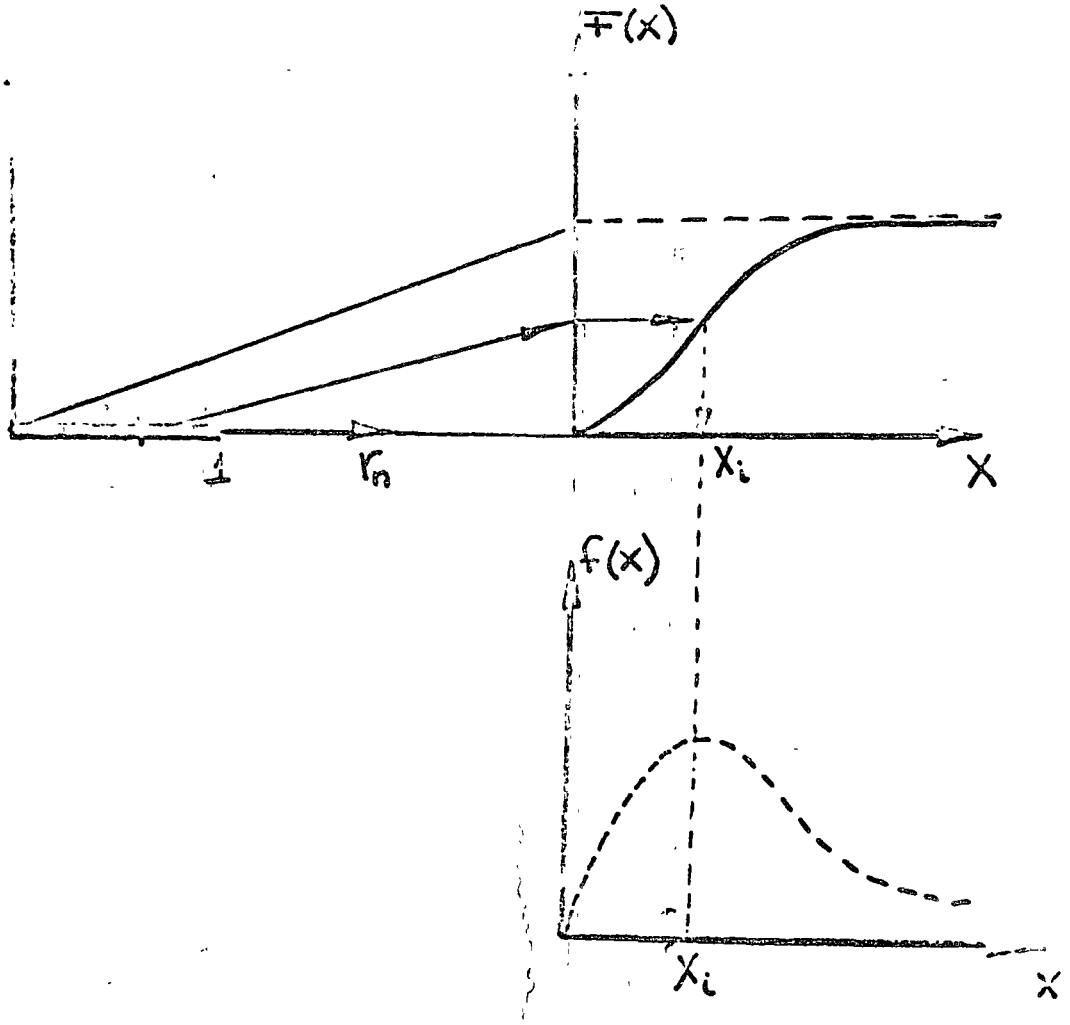
$$\int_0^{r_n} 1 \cdot d\, r_n = \int_0^{x} f(x)\, d_x$$

$$r_n = F(x)$$

$$F^{-1}(r_n) = F^{-1} F(x) = f(x)$$

Numéricamente el procedimiento es el siguiente :

se generan v.a. unit . dist. (r_n) las cuales se proyectan sobre la f.d.a. de la v.a. "x", se observa a que valor de "x" corresponde $F(r_n)$; esto se hace multitud de veces y se archiva la frecuencia con que se cae en las variables "x", - finalmente se traza un histograma de dichas frecuencias el cual corresponderá a la forma de la f.d.p. de la v.a. "x".



c) Método polar para generar v.a. gaussianas

Una v.a. gaussiana tiene una f.d.p. con las siguientes características:

$$f(x) = \frac{1}{\sigma_x \sqrt{2\pi}} \exp\left\{ -\frac{1}{2} \frac{(X-\mu_x)^2}{\sigma_x^2} \right\}$$

Se puede demostrar (ref. 5) que las ecuaciones :

$$X_1 = \sigma_x (-2 \ln r_1)^{1/2} \cos(2\pi r_2) + \mu_x$$

$$X_2 = \sigma_x (-2 \ln r_1)^{1/2} \operatorname{sen}(2\pi r_2) + \mu_x$$

donde :

r_1, r_2 = v.a. unif. dist. e indep. entre si

dan variables aleatorias X_1, X_2 con f.d.p. gaussiana.

Este método es muy útil cuando se cuenta con una computadora que calcule eficientemente las funciones ln , cos, sen.

Ejemplo

Generar 2 v.a. con f.d.p. gaussiana si :

$r_1 = 0.892$

$r_2 = 0.072$

$\sigma_x = 2.5$

$\mu_x = 5$

Sol.

$$X_1 = 2.5 (-2 \ln 0.892)^{1/2} \cos(2\pi 0.072) + 5 = 5.5407$$

$$X_2 = 2.5 (-2 \ln 0.892)^{1/2} \operatorname{sen}(2\pi 0.072) + 5 = 5.0094$$

d) Obtención de v.a. con f.d.p. exponencial

La f.d.p. exponencial tiene la siguiente forma :

$$f(t) = \alpha e^{-\alpha t}$$

donde α se define según las características del problema en estudio :

$$\alpha = \frac{1}{\mu_t}$$

Aplicando el método de la transformada inversa se tiene :

$$\int_0^{r_n} 1 \cdot dr = \int_0^{Y} f(Y)\,dY$$

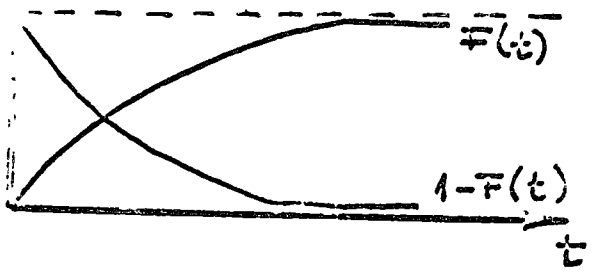
$$r_n = F(Y)$$

si: $$f(t) = \alpha e^{-\alpha t}$$

$$r_n = \int_0^t \alpha e^{-\alpha t}\,dt = -e^{-\alpha t} + 1$$

$$r_n = F(t) = 1 - e^{-\alpha t}$$

graficando F(t) y 1 - F(t) :



se observa que son simétricas y da lo mismo proyectar r_n sobre cualquiera de ellas :

$$r_n = F(t)$$

$$r_n = 1 - F(t) = e^{-\alpha t}$$

$$\ln r_n = -\alpha t$$

$$t = -\frac{1}{\alpha} \ln r_n$$

$$t = -\mu_t \ln r_n$$

Ejemplo

Generar v.a. con f.d.p. exponencial y media 2 seg.

Sol.

Sean :

$r_1 = 0.898$

$r_2 = 0.175$

$r_3 = 0.533$

por lo que :

$$t_1 = -2 \ln 0.898 = 0.215 \quad S$$

$$t_2 = -2 \ln 0.175 = 3.485 \quad S$$

$$t_3 = -2 \ln 0.533 = 1.258 \quad S$$

e) Métodos de Monte Carlo

Por métodos de Monte Carlo se conocen todas las simulaciones en las cuales están involucrados números aleatorios ó v.a. con cualquier tipo de distribución. Este tipo de simulaciones es muy útil cuando resulta imprác-tico, costoso ó imposible el simular una actividad.

Para efectuar estas simulaciones se requiere únicamente conocer las medias, desviaciones estandar y f.d.p. de las v.a. inmiscuidas.

Entre los procesos que se pueden simular de esta manera se cuentan : fenómenos de teoría de colas, simulaciones de accidentes, eventos de frecuencia, etc.

Para ilustrar el procedimiento se resolverá un ejemplo.

Ejemplo

Las llamadas telefónicas por minuto que entran a una central telefónica tienen una f.d.p. de Poisson con media 5. Simular la cantidad de llamadas que entrarán en 10 minutos si la f.d.p. de Poisson es :

$$P(k) = \frac{\mu^k e^{-\mu}}{k!}, \quad k = 0, 1, 2, \ldots$$

Sol.

K	P(k)	$P_{x\leq}(k)$
0	0.00674	0.00674
1	0.0336	0.0404
2	0.0842	0.124
3	0.140	0.264
4	0.175	0.439
5	0.1754	0.6144
6	0.1462	0.7606
7	0.104	0.865
8	0.0652	0.93
9	0.0362	0.966
10	0.0181	0.984
11		1.0

De una tabla de números aleatorios se escogen 10 (correspondientes a c/ minuto) :

1) 0.621
2) 0.861
3) 0.824
4) 0.708
5) 0.395
6) 0.165
7) 0.709
8) 0.937
9) 0.630
10) 0.859

estos números se consideran como los f.d.a. para c/ día y con este valor se entra a la tabla para ver cuántas llamadas corresponden :

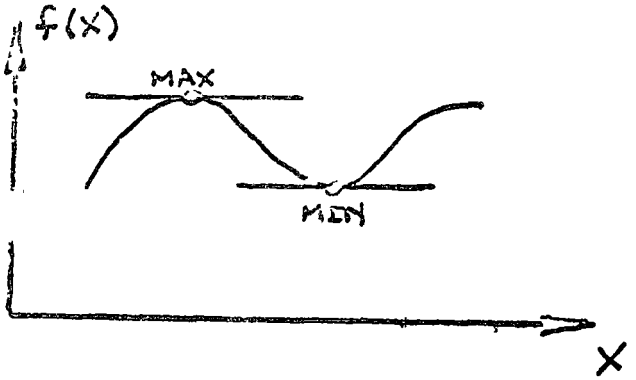
min	f. d. a.	llamadas
1	0.621	6
2	0.861	7
3	0.824	7
4	0.708	6
5	0.395	4
6	0.165	3
7	0.709	6
8	0.987	11
9	0.630	6
10	0.859	7
TOTAL	10	63

IX) OPTIMIZACION DE FUNCIONES

1. Funciones unidimensionales

Dada una función de una variable $y = f(x)$, se denomina como puntos críticos de la misma aquéllos que dan un valor máximo ó mínimo de la función en un intervalo considerado, esto se cumple cuando :

$$Y' = f'(X) = 0$$



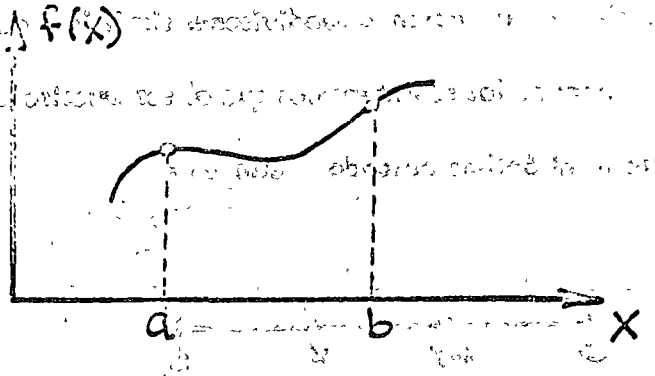
o sea, que la pendiente de la tangente a la curva en el punto considerado es nula. Dicho proceso es el que se conoce como optimización.

El método analítico solo es válido cuando la función es contínua en el intervalo considerado, esta es una de las causas por las que existen métodos numéricos para resolver dicho problema, la otra es que una computadora digital no puede efectuar el proceso analítico.

a) Método de búsqueda aleatoria

Este método es aplicable para funciones unidimensionales contínuas o discontínuas, es decir, para todo tipo de funciones. El procedimiento a seguir es :

-Fijar un intervalo de búsqueda (a, b) :



- Generar un número aleatorio uniformemente distribuido (r_i) y proyectarlo en el intervalo de búsqueda (a, b) :

$$X_i = A + r_i\,(b - a)$$

- Evaluar la función en el punto encontrado X_i

- Generar otro número aleatorio y evaluar la función en el punto X_{i+1}.

- Comparar los valores de la función obtenidos en dos iteraciones sucesivas e ir guardando la variable que arroje un valor óptimo :

$$\max\,(f(x_i),\ f(X_{i+1}))$$

ó

$$\min\,(f(x_i),\ f(x_{i+1}))$$

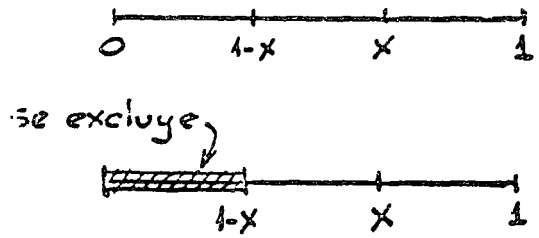
- Detener el proceso cuando :

$$|f(x_i) - f(x_{i+1})| < \epsilon$$

b) Método de Fibonacci

Este método solo es aplicable en funciones unimodales y unidimensionales para el intervalo de búsqueda, a fin de que el método sea convergente al valor óptimo.

Básicamente consiste en ir haciendo subdivisones simétricas del intervalo de búsqueda e ir eliminando los subintervalos que al ser muestreados dan un valor en sentido contrario al óptimo buscado, esto es :



Al efectuar el cociente de dos particiones para encontrar la razón ó factor al cual se está haciendo la partición se llega a una serie de números que se conocen como números de Fibonacci (Fn), los cuales se generan en la siguiente forma :

$$F_o = 1$$

$$F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2}$$

Números de Fibonacci

n	0	1	2	3	4	5	6	7	8	9	10	11
F_n	1	1	2	3	5	8	13	21	34	55	89	144

Numéricamente para aplicar el método se procede como se indica a continuación.

Dada la función f(x) se establece un intervalo de búsqueda (a, b) con lo cual la longitud inicial del intervalo es :

$$L_1 = b - a$$

el primer incremento de partición será :

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_n}$$

con lo que los primeros puntos límites del subintervalo son :

$$X_1 = a + \Delta_2$$

$$X_2 = b - \Delta_2$$

se comparan los valores $f(X_1)$ y $f(X_2)$ y de acuerdo al resultado, ya sea que se esté minimizando ó maximizando, se rechaza cualquiera de los siguientes intervalos :

$[a,\ a + \Delta_2]$, si $f(X_1)$ no sirve

$[b - \Delta_2,\ b]$, si $f(X_2)$ no sirve

el intervalo restánte se subdivide nuevamente y así sucesivamente, el procedimiento se repite hasta alcanzar el grado de exactitud deseado, es decir :

$$\left| f(X_i) - f(X_{i+1}) \right| < \varepsilon$$

Después de la primera iteración la longitud del intervalo restante será :

a X_1 X_2 b

$$L_2 = b - X_1 = b - (a + \Delta_2) = b - a - \Delta_2$$

$$= L_1 - \Delta_2$$

$$L_2 = X_2 - a = b - \Delta_2 - a = b - a - \Delta_2$$

$$= L_1 - \Delta_2$$

En términos generales se llega a establecer que el valor del incremento y longitud del intervalo para la "i - ésima" iteración son :

$$L_i = L_{i-1} - \Delta_i$$

$$L_i = L_1 \left(\frac{F_{n-i+1}}{F_n} \right)$$

$$\Delta_{i+1} = L_i \left[\frac{F_{n-(i+1)}}{F_{n-(i-1)}} \right]$$

n : es la max. cantidad de números de Fibonacci - que se deben usar para el grado de exactitud - deseado.

La ventaja de este método es que solo se tiene que evaluar la función una sola vez en cada iteración debido a que en el intervalo restante una de las nuevas subdivisiones cae en el mismo lugar que una de las subdivisiones de la iteración anterior lo cual se demuestra a continuación.

Se tenía :

$$L_2 = L_1 - \Delta_2 \qquad (IX.0)$$

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_n} \qquad (IX.1)$$

substituyendo (IX.1) en (IX.0) :

$$L_2 = L_1 \frac{(F_n - F_{n-2})}{F_n} \qquad (IX.2)$$

por otro lado :

$$F_n = F_{n-1} + F_{n-2} \qquad (IX.3)$$

substituyendo (IX.3) en (IX.2) :

$$L_2 = L_1 \frac{F_{n-1}}{F_n}$$

y se define el nuevo incremento :

$$\Delta_3 = L_2 \frac{F_{n-3}}{F_{n-1}}$$

veamos la relación de X_1 y X_2 con Δ_3 :

$$X_2 - X_1 = b - \Delta_2 - (a - \Delta_2)$$

$$= L_1 - 2\Delta_2$$

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_{n-1}}$$

$$X_2 - X_1 = L_1 \left(1 - 2\frac{F_{n-2}}{F_n}\right)$$

$$= L_2 \left(\frac{F_n - F_{n-2} - F_{n-2}}{F_n}\right)$$

$$= L_1 \left(\frac{F_{n-1} - F_{n-2}}{F_n}\right)$$

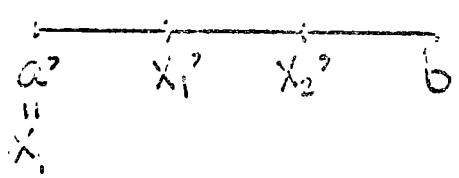
$$X_2 - X_1 = L_1 \left(\frac{F_{n-3}}{F_n}\right) \qquad (IX.4)$$

$$L_1 = L_2 \frac{F_n}{F_{n-1}} \qquad (IX.5)$$

substituyendo (IX.5) en (IX.4) :

$$X_2 - X_1 = L_2 \frac{F_{n-3}}{F_{n-1}} = \Delta_3 \qquad (IX.6)$$

analizando (IX.6) geométricamente :



se desechó $[a, X_1]$

$$X_1 = a + \Delta_2$$

$$X_2 = b - \Delta_2$$

$$X_1' = a' + \Delta_3$$

$$= X_1 + \Delta_3$$

$$= a + \Delta_2 + \Delta_3$$

$$= a + \Delta_2 + X_2 - X_1$$

$$= a + \Delta_2 + X_2 - a - \Delta_2$$

$$X_1' = X_2$$

por lo que :

$$f(X_1') = f(X_2)$$

y ya no hay que evaluar . Se llega a un resultado semejante si se rechaza el intervalo $(b - \Delta_2, b)$.

Para ver la cantidad de números de Fibonacci que se deben usar analicemos las longitudes de los subintervalos :

$$L_2 = L_1 \frac{F_{n-1}}{F_n}$$

$$L_3 = L_1 \frac{F_{n-2}}{F_n}$$

$$\vdots$$

$$L_n = L_1 \frac{F_o}{F_n} \qquad (IX.7)$$

de (IX.7) :

$$\frac{L_n}{L_1} = \frac{F_o}{F_n} \qquad (IX.8)$$

(IX.8) nos da la relación de longitudes del primero y último intervalo, de aquí se deduce la cantidad de números de Fibonacci de acuerdo a la precisión deseada, o sea; si se quiere $L_n = 0.5$ y $L_1 = 50$, se tendría :

$$\frac{L_n}{L_1} = \frac{0.5}{50} = \frac{1}{100} = \frac{F_o}{F_n}$$

$$\therefore F_n = 100\, F_o = 100$$

esto se cumple solo si n = 11 de acuerdo a la tabla de números de Fibonacci:

Ejemplo

Encontrar el punto para el cual la función $y = 5X^2 + 5X$ alcanza un mínimo escogiendo como intervalo de búsqueda (- 5, 0) y si se desea una aproximación de 0.1 .

Sol .

Determinemos "n" de acuerdo a la aproximación pedida :

$$0.1 = \frac{F_o}{F_n}$$

$$F_n = 10\, F_o = 10$$

$$n = 6$$

Aplicando el método se tiene :

$L_1 = 0 - (-5) = 5$

$\Delta_2 = L_1 \frac{F_{n-2}}{F_n} = L_1 \frac{F_4}{F_6} = 5 \left(\frac{5}{13}\right)$

$= 1.92$

$X_1 = -5 + 1.92 = -3.076$

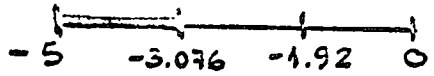
$X_2 = 0 - 1.92 = -1.92$

$f(X_1) = 31.928$

$f(X_2) = 8.832$

$f(X_1) > f(X_2)$

se rechaza (-5, -3.076)



$L_2 = L_1 - \Delta_2 = L_1 \left(\frac{F_{n-1}}{F_n}\right) = 5 \frac{F_5}{F_6}$

$= 5 - 1.92 = 3.08$

$\Delta_3 = L_2 \frac{F_3}{F_5} = 3.08 \left(\frac{3}{8}\right) = 1.155$

$X_1' = -3.076 + 1.155 = -1.92 = X_2$

$X_2' = 0 - 1.155 = -1.155$

$f(X_2') = 0.895125$

$f(X_1') = f(X_2) > f(X_2')$

se rechaza (-3.076, - 1.92)

$$L_3 = L_2 - \Delta_3 = 3.08 - 1.155$$
$$= 1.923$$

$$\Delta_4 = L_3 \frac{F_2}{F_4} = 1.92 \left(\frac{2}{5}\right) = 0.768$$

$$X_1'' = -1.92 - 0.768 = -1.152 = X_2$$

$$X_2'' = 0 - 0.768 = -0.768$$

$$f(X_2'') = -0.89 \quad f(X_1'')$$

se rechaza (- 1.92, - 1.155)

$$L_4 = L_3 - \Delta_4 = 1.923 - 0.768 = 1.15$$

$$\Delta_5 = L_4 \frac{F_1}{F_3} = 1.15 \left(\frac{1}{3}\right) = 0.383$$

$$X_1''' = -1.15 + 0.383 = -0.767$$

$$X_2''' = -0.383$$

$$f(X_2''') = -1.181 \quad f(X_1''')$$

se rechaza (-1.15, - 0.767)

$$L_5 = L_4 - \Delta_5 = 1.15 - 0.383 = 0.769$$

$$\Delta_6 = L_5 \frac{F_0}{F_2} = 0.769 \left(\frac{1}{2}\right) = 0.384$$

$$X_1'''' = -0.767 + 0.384 = -0.384$$

$$X_2'''' = -0.384$$

por lo tanto :

$$X = -0.384$$

$$f(x) = -1.182$$

2. Funciones multidimensionales

Una función multidimensional es aquella en la que la variable dependiente es función de varias variables independientes, es decir :

$$f(\underline{x}) = f(x_1, x_2, \ldots, x_n)$$

Para encontrar los puntos críticos de dicha función, el procedimiento analítico consiste en derivar $f(\underline{x})$ parcialmente con respecto a cada una de las variables independientes e igualar dichas derivadas a cero; del sistema de ecuaciones formado de esa manera se despejan los valores que arrojan un punto crítico de la función $f(\underline{x})$:

$$\frac{\partial f}{\partial x_1} = 0$$
$$\frac{\partial f}{\partial x_2} = 0$$
$$\vdots$$
$$\frac{\partial f}{\partial x_n} = 0$$

En muchas ocasiones el evaluar las derivadas es prácticamente imposible o muy laborioso, así como el resolver el sistema de ecuaciones que se forma; además en la computadora no se puede proceder en la forma antes descrita; en todos estos casos el método de elección es una solución numérica .

a) Búsqueda por gradiente.

Esta técnica sirve para maximizar o minimizar una función multidimensional $f(\underline{x})$. El procedimiento consiste en ir escalando la curva de niveles --

eligiendo para ello la trayectoria que permita alcanzar el punto crítico lo más rápidamente posible.

Es un método de aproximaciones sucesivas y como todos ellos requiere de un valor inicial de la solución con el cual iniciar el proceso :

$$\underline{X}_o = (X_1^o, X_2^o, \ldots, X_n^o)$$

a continuación se busca la dirección de la trayectoria óptima de búsqueda para el punto crítico de la función $f(\underline{x})$, lo cual se hace en base al gradiente de la función $\nabla f(\underline{x})$ mismo que sirve para calcular los incrementos $\underline{\Delta x}$ que dan el siguiente valor aproximado de la solución $\underline{X_1}$; esto se hace sucesivamente hasta que se llega a :

$$\nabla f(\underline{x}) \doteq 0$$

ó $\left| f(x_n) - f(x_n + 1) \right| < \epsilon$

El método presenta la inconveniencia de que si la función presenta puntos críticos relativos se puede llegar a $\nabla f(\underline{x}) = 0$ sin que se haya alcanzado el punto óptimo, para evitar estos errores se recomienda efectuar una revisión alrededor del punto para el cual $\nabla f(\underline{x}) \doteq 0$ hasta asegurarse de que no hay otras trayectorias que conduzcan a un valor más exacto de la solución. A continuación se ilustra el proceso numérico:

Sea la función :

$$F(X_1, X_2, \ldots, X_n) \qquad (IX.9)$$

se escoje un valor inicial de la solución :

$$\underline{X_o} = (X_1^o, X_2^o, \ldots, X_n^o, \qquad (IX.10)$$

se evalua el gradiente de la función :

$$\nabla F(\underline{X}) = \left(\frac{\partial F}{\partial X_1}, \frac{\partial F}{\partial X_2}, \ldots, \frac{\partial F}{\partial X_n} \right) \qquad (IX.11)$$

se evaluan las derivadas parciales del gradiente, las cuales empleando computadora se obtienen de la siguiente forma :

$$\left.\begin{aligned} \frac{\partial F}{\partial X_1} &= \frac{F(X_1+\Delta X_1, X_2, \ldots, X_n) - F(X_1, X_2, \ldots X_n)}{\Delta X_1^o} \\ &\vdots \\ \frac{\partial F}{\partial X_2} &= \frac{F(X_1, X_2+\Delta X_2, \ldots, X_n) - F(X_1, X_2, \ldots, X_n)}{\Delta X_n^o} \end{aligned}\right\} \qquad (IX.12)$$

los incrementos $\underline{\Delta X^o}$ son diferentes de los incrementos de la variable para caminar sobre la trayectoria de búsqueda, estos incrementos solo se utilizan para evaluar las derivadas parciales y deben ser lo más pequeño posible para que dichas derivadas sean lo más aproximadas posible.

Se evalua el gradiente de la función en $\underline{X_o}$:

$$\nabla F(\underline{X})\Big|_{\underline{X_o}} = \left(\frac{\partial F}{\partial X_1}, \frac{\partial F}{\partial X_2}, \ldots, \frac{\partial F}{\partial X_n} \right)\Big|_{\underline{X_o}} \qquad (IX.13)$$

a continuación se obtiene el incremento de la función, el cual por el cálculo diferencial está dado por :

$$\Delta F(\underline{X}) = \frac{\partial F}{\partial X_1}\Big|_{\underline{X_o}} \Delta X_1 + \frac{\partial F}{\partial X_2}\Big|_{\underline{X_o}} \Delta X_2 + \ldots + \frac{\partial F}{\partial X_n}\Big|_{\underline{X_o}} \Delta X_n$$

$$\Delta F(\underline{X}) = \nabla F\Big|_{\underline{X_o}} \underline{\Delta X} \qquad (IX.14)$$

con lo que :

$$\underline{X}_1 = \underline{X}_0 + \underline{\Delta X}_0$$

$$F(\underline{X}_1) = F(\underline{X}_0) + \Delta F(\underline{X}_0) \qquad (IX.15)$$

El incremento $\underline{\Delta X}$ se debe calcular de forma en que se obtenga el mejor incremento o decremento de la función, según el problema que se esté tratando. Del cálculo se sabe que la función variará más rápidamente si la variable independiente se incrementa en la dirección del gradiente :

$$\max_{\min} \Delta F\Big|_{\underline{X}_0} = \max_{\min} \nabla F\Big|_{\underline{X}_0} \underline{\Delta X}$$

para ello se buscará el valor óptimo en un entorno circular alrededor de $\underline{X}_0$:

$$|\underline{X} - \underline{X}_0| = r$$

$$\underline{\Delta X} = \underline{X} - \underline{X}_0$$

$$|\underline{\Delta X}| = r$$

$$|\underline{\Delta X}|^2 = r^2 \qquad (IX.16)$$

$$|\underline{\Delta X}|^2 = \Delta X_1^2 + \Delta X_2^2 + \ldots + \Delta X_n^2 \qquad (IX.17)$$

el punto óptimo se obtendrá aplicando el método de los multiplicadores de -- Lagrange :

$$L(X_1, X_2, \ldots, X_n, \lambda) = \nabla F\Big|_{\underline{X}_0} \underline{\Delta X} - \lambda \left\{ |\underline{\Delta X}|^2 - r^2 \right\}$$

$$= \frac{\partial F}{\partial X_1}\Big|_{\underline{X}_0} \Delta X_1 + \ldots + \frac{\partial F}{\partial X_n}\Big|_{\underline{X}_0} \Delta X_n -$$
$$- \lambda \left(\Delta X_1^2 + \ldots + \Delta X_n^2 - r^2 \right)$$

$$\left.\begin{aligned} \frac{\partial L}{\partial \Delta X_1} &= \frac{\partial F}{\partial X_1}\Big|_{\underline{X}_0} - 2\lambda \Delta X_1 = 0 \\ \frac{\partial L}{\partial \Delta X_2} &= \frac{\partial F}{\partial X_2}\Big|_{\underline{X}_0} - 2\lambda \Delta X_2 = 0 \\ &\vdots \\ \frac{\partial L}{\partial \Delta X_n} &= \frac{\partial F}{\partial X_n}\Big|_{\underline{X}_0} - 2\lambda \Delta X_n = 0 \end{aligned}\right\} \qquad (IX.18)$$

$$\frac{\partial L}{\partial \lambda} = -\Delta X_1^2 - \Delta X_2^2 - \ldots - \Delta X_n^2 + r^2 = 0 \qquad (IX.19)$$

de las ecuaciones (IX.18)

$$\Delta X_1 = \frac{1}{2\lambda} \frac{\partial F}{\partial X_1}\Big|_{\underline{X}_0} = \rho \frac{\partial F}{\partial X_1}\Big|_{\underline{X}_0}$$

$$\vdots$$

$$\Delta X_n = \frac{1}{2\lambda} \frac{\partial F}{\partial X_n}\Big|_{\underline{X}_0} = \rho \frac{\partial F}{\partial X_n}\Big|_{\underline{X}_0}$$

o sea :

$$\underline{\Delta X} = \rho \nabla F\Big|_{\underline{X}_0} \qquad (IX.20)$$

estos valores se substituyen en la función $F(\underline{x})$:

$$F(\underline{X_o} + \Delta \underline{X_o}) = F(\underline{X_o} + \rho \nabla F \Big|_{\underline{X_o}}) \qquad (IX.21)$$

con lo que se obtiene una función unidimensional en " ρ ", la cual se optimiza en función de " ρ " y el valor obtenido se substituye en (IX.20) y esto a su vez en :

$$\underline{X_1} = \underline{X_o} + \Delta \underline{X_o}$$

se compara $F(\underline{X_1})$ y $F(\underline{X_o})$ y de acuerdo al resultado se detiene el proceso o se vuelve a repetir la búsqueda de una nueva trayectoria hasta obtener :

$$\left| F(\underline{X_n}) - F(\underline{X_{n+1}}) \right| < \hat{\varepsilon}$$

El proceso de optimización de $F(\rho)$ en la computadora se puede efectuar eficientemente con el método de búsqueda aleatoria.

Ejemplo

Determinar el mínimo de la función :

$$F(X_1, X_2) = X_1^2 + X_2^2 + X_1 X_2 + X_1$$

por el método analítico y el numérico.

Sol.

Analíticamente :

$$\frac{\partial F}{\partial X_1} = 2X_1 + X_2 + 1 = 0$$

$$\frac{\partial F}{\partial X_2} = 2X_2 + X_1 = 0$$

$$2X_1 + X_2 = -1$$

$$X_1 + 2X_2 = 0$$

$$X_1 = -2X_2$$

$$-4X_2 + X_2 = -1$$

$$-3X_2 = -1$$

$$X_2 = 1/3 = 0.333$$

$$X_1 = -2/3 = -0.666$$

Numéricamente se tendría :

$$F(\underline{X}) = X_1^2 + X_2^2 + X_1 X_2 + X_1$$

eligiendo el valor inicial de la solución :

$$\underline{X_o} = (X_1^o, X_2^o) = (0,0)$$

obteniendo el gradiente :

$$\frac{\partial F}{\partial X_1} = 2X_1 + X_2 + 1$$

$$\frac{\partial F}{\partial X_2} = 2X_2 + X_1$$

evaluando en $\underline{X_o}$:

$$\nabla F \Big|_{\underline{X_o}} = (1,0)$$

los incrementos $\underline{\Delta X}$ serán :

$$\underline{\Delta X} = -\rho \nabla F|_{\underline{X}}$$

$$\underline{X_1} = \underline{X_0} + \underline{\Delta X_0}$$

substituyendo en $F(\underline{\ })$:

$$F(\underline{X}) = F(\rho) = \rho^2 + \rho$$

$$\frac{dF}{d\rho} = 2\rho + 1 = 0$$

$$\rho^* = -1/2$$

se tendrá :

$$\underline{X_1} = (-1/2, 0)$$

efectuando los mismos cálculos para las siguientes iteraciones :

$$\nabla F|_{\underline{X_1}} = (0, -1/2)$$

$$\underline{\Delta X_1} = (\rho \cdot \nabla F|_{\underline{X_1}}) = (0, -1/2\,\rho)$$

$$\underline{X_2} = \underline{X_1} + \underline{\Delta X_1} = (-1/2, -\rho/2)$$

$$F(\rho) = 1/4 + \rho^2/4 + \rho/4 - 1/2$$

$$\frac{dF}{d\rho} = \rho/2 + 1/4 = 0$$

$$\rho^* = -1/2$$

$$\underline{X_2} = (-1/2, 1/4)$$

$$\nabla f|_{\underline{X_2}} = (1/4, 0)$$

$$\underline{\Delta X_2} = \rho \nabla f|_{\underline{X_2}} = (\rho/4, 0)$$

$$\underline{X_3} = \underline{X_2} + \underline{\Delta X_2} = (-1/2 + \rho/4, 1/4)$$

$$f(\rho) = 1/4 - \rho/4 + \rho^2/16 + 1/16 + \rho/16 - 1/8 - 1/2 + \rho/4$$

$$f(\rho) = \rho^2/16 + \rho/16 + C$$

$$\frac{d}{d\rho} F(\rho) = \rho/8 + 1/16 = 0$$

$$\rho^* = -1/2$$

$$\underline{X_3} = (-1/2 - 1/8, 1/4)$$

$$= (-5/8, 1/4)$$

$$\underline{X_3} \doteq (-0.625, 0.25)$$

BIBLIOGRAFIA

1. Canchon B., Luther H., Wilkes J., "Applied Numerical Methods", John Wiley, 1969.

2. Gerez G.V., Grijalva L.M. "Enfoque de Ingeniería de Sistemas", Apuntes C.E.C.S.A. y Limusa Wiley (en imprenta), 1975.

3. Hamming R., "Numerical Methods for Scientists and Engineers", -- Mc. Graw Hill, 1962.

4. James M., Smith G., Wolford J., "Applied Numerical Methods for Digital Computation with FORTRAN", International Textboo Co., 1967.

5. Knuth D.F., "The Art of Computer Programming", Adisson - Wesley, 1971.

6. Kuo S., "Computer Applications of Numerical Methods", Adisson-Wesley, 1972.

7. Naylor T H., Balintfy J.L., Burdick,Chung, "Técnicas de simulación en computadoras", Limusa Wiley, 1973.

8. Oliveira S.A., "Apuntes de Métodos Numéricos", Fac. de Ingeniería, -- U.N.A.M., 1972.

APENDICE CON PROGRAMAS

DEL PAQUETE "SSP"

(SCIENTIFIC SUBROUTINE PACKAGE)

PARA LA COMPUTADORA

IBM - 1130

INTRODUCTION

The IBM 1130 Scientific Subroutine Package makes available a mathematical and statistical subroutine library. The user may supplement or modify the collection to meet his needs. This library includes a wide variety of subroutines to perform the functions listed below, but is not intended to be exhaustive in terms of either functions performed or methods used.

AREAS OF APPLICATION

Individual subroutines, or a combination of them, can be used to carry out the listed functions in the following areas:

Statistics

- Analysis of variance (factorial design)
- Correlation analysis
- Multiple linear regression
- Polynomial regression
- Canonical correlation
- Factor analysis (principal components, varimax)
- Discriminant analysis (many groups)
- Time series analysis
- Data screening and analysis
- Nonparametric tests
- Random number generation (uniform, normal)

Matrix Manipulation

- Inversion
- Eigenvalues and eigenvectors (real symmetric case)
- Simultaneous linear algebraic equations
- Transpositions
- Matrix arithmetic (addition, product, etc.)
- Partitioning
- Tabulation and sorting of rows or columns
- Elementary operations on rows or columns

Other Mathematical Areas

- Integration of given or tabulated functions
- Integration of first-order differential equations
- Fourier analysis of given or tabulated functions
- Bessel and modified Bessel function evaluation
- Gamma function evaluation
- Legendre function evaluation
- Elliptic, exponential, sine, cosine, Fresnel integrals
- Finding real roots of a given function
- Finding real and complex roots of a real polynomial
- Polynomial arithmetic (addition, division, etc.)
- Polynomial evaluation, integration, differentiation

CHARACTERISTICS

Some of the characteristics of the Scientific Subroutine Package are:

- All subroutines are free of input/output statements.
- Subroutines do not contain fixed maximum dimensions for the data arrays named in their calling sequences.
- All subroutines are written in 1130 FORTRAN.
- Many matrix manipulation subroutines handle symmetric and diagonal matrices (stored in economical, compressed formats) as well as general matrices. This can result in considerable saving in data storage for large arrays.
- The use of the more complex subroutines (or groups of them) is illustrated in the program documentation by sample main programs with input/output.
- All subroutines are documented uniformly.

SUBROUTINES

GENERAL REMARKS

Below are listed the subroutines of SSP/1130, grouped into related functional areas. In the case of six statistical entries (Multiple Linear Regression to Factor Analysis) the abstract gives the sequence of several SSP subroutines needed to perform the statistical function.

A tabulation of the subroutines of SSP, with detailed characteristics, is given in the appendices.

STATISTICS

Data Screening

TALLY--totals, means, standard deviations, minimums, and maximums

BOUND--selection of observations within bounds

SUBST--subset selection from observation matrix

ABSNT--detection of missing data

TAB1--tabulation of data (1 variable)

TAB2--tabulation of data (2 variables)

SUBMX--build subset matrix

Elementary Statistics

MOMEN--first four moments

TTSTT--tests on population means

Correlation

CORRE--means, standard deviations, and correlations

Multiple Linear Regression

Abstract (CORRE, ORDER, MINV, MULTR in sequence)

ORDER--rearrangement of intercorrelations

MULTR--multiple regression and correlation

Polynomial Regression

Abstract (GDATA, ORDER, MINV, MULTR in sequence)

GDATA--data generation

Canonical Correlation

Abstract (CORRE, CANOR, MINV, NROOT, EIGEN in sequence)

CANOR--canonical correlation

NROOT--eigenvalues and eigenvectors of a special nonsymmetric matrix

Analysis of Variance

Abstract (AVDAT, AVCAL, MEANQ in sequence)

AVDAT--data storage allocation

AVCAL--Σ and Δ operation

MEANQ--mean square operation

Discriminant Analysis

Abstract (DMATX, MINV, DISCR in sequence)

DMATX--means and dispersion matrix

DISCR--discriminant functions

Factor Analysis

Abstract (CORRE, EIGEN, TRACE, LOAD, VARMX in sequence)

TRACE--cumulative percentage of eigenvalues

LOAD--factor loading

VARMX--varimax rotation

Time Series

AUTO--autocovariances

CROSS--crosscovariances

SMO--application of filter coefficients (weights)

EXSMO--triple exponential smoothing

Nonparametric Statistics

CHISQ--x^2 test for a contingency table

UTEST--Mann-Whitney U-test

FWOAV--Friedman two-way analysis of variance

QTEST--Cochran Q-test

SRANK--Spearman rank correlation

KRANK--Kendall rank correlation

WTEST--Kendall coefficient of concordance

RANK--rank observations

TIE--calculation of ties in ranked observations

Random Number Generators

RANDU--uniform random numbers

GAUSS--normal random numbers

MATRIX MANIPULATION

MINV--Matrix inversion

EIGEN--eigenvalues and eigenvectors of a real, symmetric matrix

SIMQ--solution of simultaneous linear, algebraic equations

GMADD--add two general matrices

GMSUB--subtract two general matrices

GMPRD--product of two general matrices

GMTRA--transpose of a general matrix

GTPRD--transpose product of two general matrices

MADD--add two matrices

MSUB--subtract two matrices

MPRD--matrix product (row into column)

MTRA--transpose a matrix

TPRD--transpose product

MATA--transpose product of matrix by itself

SADD--add scalar to matrix

SSUB--subtract scalar from a matrix

SMPY--matrix multiplied by a scalar

SDIV--matrix divided by a scalar

RADD--add row of one matrix to row of another matrix

CADD--add column of one matrix to column of another matrix

SRMA--scalar multiply row and add to another row

SCMA--scalar multiply column and add to another column

RINT--interchange two rows

CINT--interchange two columns

RSUM--sum the rows of a matrix

CSUM--sum the columns of a matrix

RTAB--tabulate the rows of a matrix

CTAB--tabulate the columns of a matrix

RSRT--sort matrix rows

CSRT--sort matrix columns

RCUT-partition row-wise

CCUT--partition column-wise

RTIE--adjoin two matrices row-wise

CTIE--adjoin two matrices column-wise

MCPY--matrix copy

XCPY--copy submatrix from given matrix

RCPY--copy row of matrix into vector

CCPY--copy column of matrix into vector

DCPY--copy diagonal of matrix into vector

SCLA--matrix clear and add scalar

DCLA--raplace diagonal with scalar

MSTR--storage conversion

MFUN--matrix transformation by a function

RIN--reciprocal function for MFUN

LOC--location in compressed-stored matrix

AR --vector storage--double dimensioned storage conversion

OTHER MATHEMATICAL AREAS

Integration

QSF--integral of tabulated function by Simpson's Rule

QATR--integral of given function by trapezoidal rule using Romberg's extrapolation method

RK1--integral of first-order differential equation by Runge-Kutta method

RK2--tabulated integral of first-order differential equation by Runge-Kutta method

RKGS--solution of a system of first-order differential equations by Runge-Kutta method

Fourier Analysis

FORIF--Fourier analysis of a given function

FORIT--Fourier analysis of a tabulated function

Special Operations and Mathematical Functions

GAMMA--gamma function

LEP--Legendre polynomial

BESJ--J Bessel function

BESY--Y Bessel function

BESI--I Bessel function

BESK--K Bessel function

ELI--elliptic integral of the first kind

ELI2--elliptic integral of the second kind

EXPI--exponential integral

SICI--sine cosine integral

CI--cosine integral

Roots of Nonlinear Equations

RTWI--refine estimate of root by Wegstein's iteration

RTMI--determine root within a range by Mueller's iteration

RTNI--refine estimate of root by Newton's iteration

Roots of Polynomial

POLRT--real and complex roots of a real polynomial

Polynomial Operations

PADD--add two polynomials

PADDM--multiply polynomial by constant and add to another polynom

PCLA--replace one polynomial by another

PSUB--subtract one polynomial from another

PMPY--multiply two polynomials

PDIV--divide one polynomial by another

PQSD--quadratic synthetic division of a polynomial

PVAL--value of a polynomial

PVSUB--substitute variable of polynomial by another polynomial

PCLD--complete linear synthetic division

PVLD--evaluate polynomial and its first derivative

PDER--derivative of a polynomial

PINT--integral of a polynomial

PGCD--greatest common divisor of two polynomials

PNORM--normalize coefficient vector of a polynomial

VERALL RULES OF USAGE

SUBROUTINE USAGE

All subroutines in the Scientific Subroutine Package (SSP) are entered by means of the standard FORTRAN CALL statement. These subroutines are purely computational in nature and do not contain any references to input/output devices. The user must therefore furnish, as part of his program, whatever input/output and other operations are necessary for the total solution of his problem. In addition, the user must define by DIMENSION statements all matrices to be operated on by SSP subroutines as well as those matrices utilized in his program. The subroutines contained in SSP are no different from any user-supplied subroutine. All of the normal rules of FORTRAN concerning subroutines must, therefore, be adhered to with the exception that the dimensioned areas in the SSP subroutine are not required to be the same as those in the calling program.

MATRIX OPERATIONS

Special consideration must be given to the subroutines that perform matrix operations. These subroutines have two characteristics that affect the format of the data in storage — variable dimensioning and data storage compression.

Variable Dimensioning

These subroutines that deal with matrices can operate on any size array limited, in most cases, only by the available core storage and numerical analysis considerations. The subroutines do not contain fixed maximum dimensions for data arrays named in their calling sequence. The variable dimension capability has been implemented in SSP by using a vector storage approach. Under this approach, each column of a matrix is immediately followed in storage by the next column. Vector storage and two-dimensional storage result in the same layout of data in core, so long as the number of rows and columns in the matrix are the same as those in the user's dimension statement. If, however, the matrix is smaller than the dimensioned area, the two forms of storage are not compatible. A subroutine called ARRAY is available in SSP to change from one form of storage to the other. In addition, a subroutine called LOC is available to assist in referencing elements in an array stored in the vector mode.

Storage Compression

Many subroutines in SSP can operate on compressed forms of matrices, as well as the normal form. Using this capability, which is called "storage mode", considerable savings in matrix storage can be obtained for special forms of large arrays. The three modes of storage are termed general, symmetric, and diagonal. In this context, general mode is one in which all elements of the matrix are in storage. Symmetric mode is one in which only the upper triangular portion of the matrix is retained column-wise in sequential locations in storage. (The assumption is made that the corresponding elements in the lower triangle have the same value.) Diagonal mode is one in which only the diagonal elements of the matrix are retained in sequential locations in storage. (The off-diagonal elements are assumed to be zero.) This capability has been implemented using the vector storage approach.

A special set of matrix subroutines is included in SSP. These subroutines (GMADD, GMSUB, GMPRD, GMTRA, and GTPRD) execute faster than their counterparts (MADD, MSUB, MPRD, MTRA, and TPRD) because they do not have the storage mode capability.

SAMPLE PROGRAMS

Distributed with the subroutines of SSP are 15 sample main programs with input/output, control (parameter) cards, and sample data. These sample main programs serve two purposes. First, they demonstrate input/output and the use of sequences of subroutines to carry out higher level functions. Secondly, many of the sample programs are useful as they stand. The user need only substitute his own data (in similar format).

There are sample main programs to do each of the following operations (the code names of the main programs are enclosed in parentheses):

1. Data screening (DASCR)
2. Regression (REGRE)
3. Polynomial regression (POLRG)
4. Canonical correlation (MCANO)
5. Analysis of variance, factorial design (ANOVA)
6. Discriminant analysis, many groups (MDISC)
7. Factor analysis (FACTO)
8. Exponential smoothing, third order (EXPON)
9. Matrix addition (ADSAM)

Subroutines - Random Number Generators

RANDU

Purpose:
Computes uniformly distributed random floating point numbers between 0 and 1.0 and integers in the range 0 to 2**15.

Usage:
CALL RANDU(IX, IY, YFL)

Description of parameters:
IX - For the first entry this must contain any odd positive integer less than 32 768. After the first entry, IX should be the previous value of IY computed by this subroutine.
IY - A resultant integer random number required for the next entry to this subroutine. The range of this number is from zero to 2**15.
YFL - The resultant uniformly distributed, floating point, random number in the range 0 to 1.0.

Remarks:
This subroutine is specific to the IBM 1130. This subroutine should not repeat its cycle in less than 2 to the 13th entries.
Note. If random bits are needed, the high order bits of IY should be chosen.

Subroutines and function subprograms required:
None.

Method:
Power residue method discussed in IBM manual Random Number Generation and Testing (C20-8011).

GAUSS

This subroutine computes a normally distributed random number with a given mean and standard deviation.

An approximation to normally distributed random numbers Y can be found from a sequence of uniform random numbers* using the formula

$$Y = \frac{\sum_{i=1}^{K} X_i - \frac{K}{2}}{\sqrt{K/12}} \qquad (1)$$

where X_i is a uniformly distributed random number, $0 < X_i < 1$

K is the number of values X_i to be used

Y approaches a true normal distribution asymptotically as K approaches infinity. For this subroutine, K was chosen as 12 to reduce execution time. Equation (1) thus becomes:

$$Y = \sum_{i=1}^{12} X_i - 6.0$$

The adjustment for the required mean and standard deviation is then

$$Y' = Y * S + AM \qquad (2)$$

where Y' is the required normally distributed random number

S is the required standard deviation

AM is the required mean

* R. W. Hamming. Numerical Methods for Scientists and Engineers, McGraw-Hill, N.Y., 1962, pages 34 and 389.

Subroutine GAUSS

Purpose:

Computes a normally distributed random number with a given mean and standard deviation.

Usage:

CALL GAUSS(IX, S, AM, V)

Description of parameters:

- IX - IX must contain an odd positive integer less than 32,768. Thereafter it will contain a uniformly distributed integer random number generated by the subroutine for use on the next entry to the subroutine.
- S - The desired standard deviation of the normal distribution.
- AM - The desired mean of the normal distribution.
- V - The value of the computed normal random variable.

Remarks:

This subroutine uses RANDU which is machine specific.

Subroutines and function subprograms required:

RANDU

Method:

Uses 12 uniform random numbers to compute normal random numbers by central limit theorem. The result is then adjusted to match the given mean and standard deviation. The uniform random numbers computed within the subroutine are found by the power residue method.

```
      SUBROUTINE GAUSS(IX,S,AM,V)                                       GAUSS   1
      V=0.0                                                             GAUSS   2
      DO 10 I=1,12                                                      GAUSS   3
      CALL RANDU(IX,FV)                                                 GAUSS   4
      IX=IY                                                             GAUSS   5
   10 V=V+FV                                                            GAUSS   6
      V=(V-6.0)*S+AM                                                    GAUSS   7
      RETURN                                                            GAUSS   8
      END                                                               GAUSS   9
```

Mathematics - Special Matrix Operations

MINV

Purpose:

Invert a matrix.

Usage:

CALL MINV(A, N, D, L, M)

Description of parameters:

- A - Input matrix, destroyed in computation and replaced by resultant inverse.
- N - Order of matrix A.
- D - Resultant determinant.
- L - Work vector of length N.
- M - Work vector of length N.

Remarks:

Matrix A must be a general matrix.

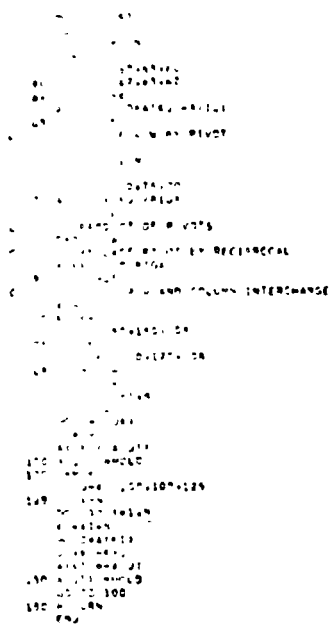
Subroutines and function subprograms required:

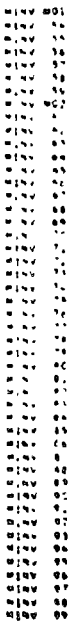
None.

Method:

The standard Gauss-Jordan method is used. The determinant is also calculated. A determinant with absolute value less than 10**(-20) indicates singularity. The user may wish to change this.

```
      SUBROUTINE MINV(A,N,D,L,M)                                        MINV    1
      DIMENSION A(1),L(1),M(1)                                          MINV    2
C     SEARCH FOR LARGEST ELEMENT                                        MINV    3
      D=1.0                                                             MINV    4
      NK=-N                                                             MINV    5
      DO 80 K=1,N                                                       MINV    6
      NK=NK+N                                                           MINV    7
      L(K)=K                                                            MINV    8
      M(K)=K                                                            MINV    9
      KK=NK+K                                                           MINV   10
      BIGA=A(KK)                                                        MINV   11
      DO 20 J=K,N                                                       MINV   12
      IZ=N*(J-1)                                                        MINV   13
      DO 20 I=K,N                                                       MINV   14
      IJ=IZ+I                                                           MINV   15
   10 IF(ABS(BIGA)-ABS(A(IJ)))15,20,20                                  MINV   16
   15 BIGA=A(IJ)                                                        MINV   17
      L(K)=I                                                            MINV   18
      M(K)=J                                                            MINV   19
   20 CONTINUE                                                          MINV   20
C     INTERCHANGE ROWS                                                  MINV   21
      J=L(K)                                                            MINV   22
      IF(J-K) 35,35,25                                                  MINV   23
   25 KI=K-N                                                            MINV   24
      DO 30 I=1,N                                                       MINV   25
      KI=KI+N                                                           MINV   26
      HOLD=-A(KI)                                                       MINV   27
      JI=KI-K+J                                                         MINV   28
      A(KI)=A(JI)                                                       MINV   29
   30 A(JI)=HOLD                                                        MINV   30
C     INTERCHANGE COLUMNS                                               MINV   31
   35 I=M(K)                                                            MINV   32
      IF(I-K) 45,45,38                                                  MINV   33
   38 JP=N*(I-1)                                                        MINV   34
      DO 40 J=1,N                                                       MINV   35
      JK=NK+J                                                           MINV   36
      JI=JP+J                                                           MINV   37
      HOLD=-A(JK)                                                       MINV   38
      A(JK)=A(JI)                                                       MINV   39
   40 A(JI)=HOLD                                                        MINV   40
C     DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS           MINV   41
C     CONTAINED IN BIGA)                                                MINV   42
   45 IF(ABS(BIGA)-1.E-20)46,46,48                                      MINV   43
   46 D=0.0                                                             MINV   44
      RETURN                                                            MINV   45
   48 DO 55 I=1,N                                                       MINV   46
      IF(I-K) 50,55,50                                                  MINV   47
   50 IK=NK+I                                                           MINV   48
      A(IK)=A(IK)/(-BIGA)                                               MINV   49
   55 CONTINUE                                                          MINV   50
C     REDUCE MATRIX                                                     MINV   51
      DO 65 I=1,N                                                       MINV   52
      IK=NK+I                                                           MINV   53
```





ALGEN

This subroutine computes the eigenvalues and eigenvectors of a real symmetric matrix.

Given a symmetric matrix A of order N, the eigenvalues are to be developed in the diagonal elements of the matrix. A matrix of eigenvectors is also to be generated.

An identity matrix is used as a first approximation of R.

The initial off-diagonal norm is computed

$$\nu_I = \left\{ \sum_{i \le k} 2A_{ik}^2 \right\}^{1/2} \qquad (1)$$

ν_I = initial norm

A = input matrix (symmetric)

This norm is divided by N at each stage to produce the threshold.

The final norm is computed:

$$\nu_F = \frac{\nu_I \times 10^{-6}}{N} \qquad (2)$$

This final norm is set sufficiently small that the requirement that any off-diagonal element A_{ij} should be smaller than ν_F in absolute magnitude controls the convergence of the process.

An indicator is initialized. This indicator is later used to determine whether any off-diagonal elements have been found that are greater than the present threshold.

Each off-diagonal element is selected in turn and a transformation is performed to annihilate the off-diagonal (pivotal) element as shown by the following equations:

$$\lambda = -A_{lm} \qquad (3)$$

$$\mu = 1/2 \, (A_{ll} - A_{mm}) \qquad (4)$$

$$\omega = \text{sign}(\mu) \frac{\lambda}{\sqrt{\lambda^2 + \mu^2}} \qquad (5)$$

$$\sin \theta = \frac{\omega}{\sqrt{2(1 + \sqrt{1 - \omega^2})}} \qquad (6)$$

$$\cos \theta = \sqrt{1 - \sin^2 \theta} \qquad (7)$$

MSUB

Purpose:

Subtract two matrices element by element to form resultant matrix.

Usage:

CALL MSUB(A, B, R, N, M, MSA, MSB)

Description of parameters:

A - Name of input matrix.
B - Name of input matrix.
R - Name of output matrix.
N - Number of rows in A, B, R.
M - Number of columns in A, B, R.
MSA - One digit number for storage mode of matrix A:
0 - General.
1 - Symmetric.
2 - Diagonal.
MSB - Same as MSA except for matrix B.

Remarks:

None.

Subroutines and function subprograms required:

LOC

Method:

Structure of output matrix is first determined. Subtraction of matrix B elements from corresponding matrix A elements is then performed. The following table shows the storage mode of the output matrix for all combinations of input matrices:

A	B	R
General	General	General
General	Symmetric	General
General	Diagonal	General
Symmetric	General	General
Symmetric	Symmetric	Symmetric
Symmetric	Diagonal	Symmetric
Diagonal	General	General
Diagonal	Symmetric	Symmetric
Diagonal	Diagonal	Diagonal

```
      SUBROUTINE MS(A(1),B(1),N,M,MSA,MSB)                              MSUB   1
      DIMENSION A(1),B(1),R(1)                                          MSUB   2
C     DETERMINE STORAGE MODE OF OUTPUT MATRIX                           MSUB   3
      IF(N-M)5,5,7                                                      MSUB   4
    5 CALL LOC(N,N,N,M,MSA)                                             MSUB   5
      GO TO 100                                                         MSUB   6
    7 MTEST=MSA+MSB                                                     MSUB   7
      GO TO(10,20,10                                                    MSUB   8
   10 MSA=1                                                             MSUB   9
   20 IF(MTEST-2) 30,30,30                                              MSUB  10
   30 MSA=2                                                             MSUB  11
C     SET UP ELEMENTS AND PERFORM SUBTRACTION                           MSUB  12
   35 DO 60 I=1,N                                                       MSUB  13
      DO 60 J=1,M                                                       MSUB  14
      CALL LOC(I,J,N,M,MSA,                                             MSUB  15
     1 IF(MSA-2) 40,50,40                                               MSUB  16
   40 CALL LOC(I,J,N,M,MSB,                                             MSUB  17
      R(L)=A(I)                                                         MSUB  18
      IF(L-1) 55,60,55                                                  MSUB  19
   50 R(L)=A(I)                                                         MSUB  20
   60 CALL LOC(I,J,N,M,MSB,                                             MSUB  21
      R(L)=A(I)                                                         MSUB  22
      IF(MSA-2) 70,80,70                                                MSUB  23
   70 R(L)=R(L)+B(I)                                                    MSUB  24
   80 R(L)=A(I)+B(L)                                                    MSUB  25
   90 CONTINUE                                                          MSUB  26
      RETURN                                                            MSUB  27
C     SUBTRACT MATRICES FOR OTHER CASES                                 MSUB  28
  100 DO 110 I=1,NM                                                     MSUB  29
  110 R(I)=A(I)-B(I)                                                    MSUB  30
      RETURN                                                            MSUB  31
      END                                                               MSUB  32
```

MPRD

Purpose:

Multiply two matrices to form a resultant matrix.

Usage:

CALL MPRD(A, B, R, N, M, MSA, MSB, L)

Description of parameters:

A - Name of first input matrix.
B - Name of second input matrix.
R - Name of output matrix.
N - Number of rows in A and R.
M - Number of columns in A and rows in B.
MSA - One digit number for storage mode of matrix A:
0 - General.
1 - Symmetric.
2 - Diagonal.
MSB - Same as MSA except for matrix B.
L - Number of columns in B and R.

Remarks:

Matrix R cannot be in the same location as matrices A or B.
Number of columns of matrix A must be equal to number of rows of matrix B.

Subroutines and function subprograms required:

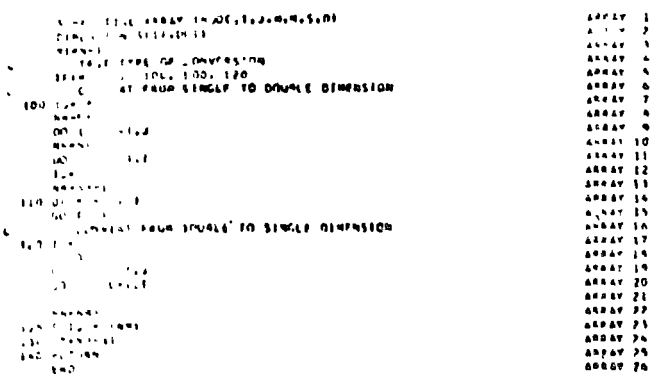
LOC

Method:

The M by L matrix B is premultiplied by the N by M matrix A and the result is stored in the N by L matrix R. This is a row into column product.
The following table shows the storage mode of the output matrix for all combinations of input matrices:

A	B	R
General	General	General
General	Symmetric	General
General	Diagonal	General
Symmetric	General	General
Symmetric	Symmetric	General
Symmetric	Diagonal	General
Diagonal	General	General
Diagonal	Symmetric	General
Diagonal	Diagonal	Diagonal

```
      SUBROUTINE MPRD(A,B,R,N,M,MSA,MSB,L)                              MPRD   1
      DIMENSION A(1),B(1),R(1)                                          MPRD   2
C     SPECIAL CASE FOR DIAGONAL BY DIAGONAL                             MPRD   3
      MSA=MSA+MSB                                                       MPRD   4
      IF(MSA-2) 30,10,30                                                MPRD   5
   10 DO 20 I=1,N                                                       MPRD   6
   20 R(I)=A(I)*B(I)                                                    MPRD   7
      RETURN                                                            MPRD   8
C     ALL OTHER CASES                                                   MPRD   9
   30 IA=0                                                              MPRD  10
      DO 90 K=1,L                                                       MPRD  11
      DO 90 J=1,M                                                       MPRD  12
      IB=(J-1)*N                                                        MPRD  13
      DO 40 I=1,N                                                       MPRD  14
      IF(MSA-2) 40,60,40                                                MPRD  15
   40 CALL LOC(I,J,N,M,MSA)                                             MPRD  16
      CALL LOC(I,K,N,L,MSB)                                             MPRD  17
      IF(I-1) 50,60,50                                                  MPRD  18
   50 IF(I-N) 70,80,70                                                  MPRD  19
   60 IA=IA+1                                                           MPRD  20
      IB=IB+1                                                           MPRD  21
   70 R(IR)=R(IR)+A(IA)*B(IB)                                           MPRD  22
   80 CONTINUE                                                          MPRD  23
   90 IR=IR+1                                                           MPRD  24
      RETURN                                                            MPRD  25
      END                                                               MPRD  26
```



Mathematics — Integration and Differentiation

QSF

This subroutine performs the integration of an equidistantly tabulated function by Simpson's rule. To compute the vector of integral values:

$$z_i = z(x_i) = \int_a^{x_i} y(x)\,dx \quad \left.\begin{array}{l} (i = 1,2,\ldots,n) \\ \text{with } x_i = a + (i-1)\,h \end{array}\right\}$$

for a table of function values y_i $(i = 1,2,\ldots,n)$, given at equidistant points $x_i = a + (i-1)\,h$ $(i = 1,2,\ldots,n)$, Simpson's rule together with Newton's 3/8 rule or a combination of these two rules is used. Local truncation error is of the order h^4 in all cases with more than three points in the given table. Only z_2 has a truncation error of the order h^4 if there are only three points in the given table. No action takes place if the table consists of less than three sample points.

The function is assumed continuous and differentiable (three or four times, depending on the rule used).

Formulas used in this subroutine (z_j are integral values, y_j function values) are:

$$z_j = z_{j-1} + \frac{h}{3}(1.25\,y_{j-1} + 2y_j - 0.25\,y_{j+1}) \qquad (1)$$

$$z_j = z_{j-2} + \frac{h}{3}(y_{j-2} + 4y_{j-1} + y_j) \qquad (2)$$

(Simpson's rule)

$$z_j = z_{j-3} + \frac{3}{8}h\,(y_{j-3} + 3y_{j-2} + 3y_{j-1} + y_j) \qquad (3)$$

(Newton's 3/8 rule)

$$z_j = z_{j-5} + \frac{h}{3}(y_{j-5} + 3.875\,y_{j-4} + 2.625\,y_{j-3} + 2.625\,y_{j-2} + 3.875\,y_{j-1} + y_j) \qquad (4)$$

[combination of (2) and (3)]

Sometimes formula (2) is used in the following form:

$$z_j = z_{j+2} - \frac{h}{3}(y_j + 4\,y_{j+1} + y_{j+2}) \qquad (5)$$

Local truncation errors of formulas (1)...(4) are, respectively:

$$R_1 = \frac{1}{24}h^4 y'''(\xi_1) \quad (\xi_1 \in [x_{j-1}, x_{j+1}])$$

$$R_2 = -\frac{1}{90}h^5 y''''(\xi_2) \quad (\xi_2 \in [x_{j-2}, x_j])$$

$$R3 = -\frac{3}{80}h^5 y''''(\xi_3) \quad (\xi_3 \in [x_{j-3}, x_j])$$

$$R_4 = -\frac{1}{144}h^5 y''''(\xi_4) \quad (\xi_4 \in [x_{j-3}, x_j])$$

However, these truncation errors may accumulate.

For reference see:

(1) F.B. Hildebrand, Introduction to Numerical Analysis. McGraw-Hill, New York/Toronto/London, 1956, pp. 71-76.

(2) R. Zurmühl, Praktische Mathematik für Ingenieure und Physiker. Springer, Berlin/Göttingen/Heidelberg, 1963, pp. 214-221.

Subroutine QSF

Purpose:

To compute the vector of integral values for a given equidistant table of function values.

Usage:

CALL QSF(H,Y,Z,NDIM)

Description of parameters:

H - The increment of argument values.
Y - The input vector of function values.
Z - The resulting vector of integral values. Z may be identical to Y.
NDIM - The dimension of vectors Y and Z.

Remarks:

No action in case NDIM less then 3.

Subroutines and function subprograms required:

None

Method:

Beginning with Z(1) = 0, evaluation of vector Z is done by means of Simpson's rule together with Newton's 3/8 rule or a combination of these two rules. Truncation error is of order H**5 (that is, fourth-order method). Only in case NDIM=3 truncation error of Z(2) is of order H**4.

This subroutine uses the Runge-Kutta method for the solution of initial-value problems.

The purpose of the Runge-Kutta method is to obtain an approximate solution of a system of first-order ordinary differential equations with given initial values. It is a fourth-order integration procedure which is stable and self-starting, that is, only the functional values at a single previous point are required to obtain the functional values ahead. For this reason it is easy to change the step size h at any step in the calculations. On the other hand, each Runge-Kutta step requires the evaluation of the right-hand side of the system four times, which is a great disadvantage compared with other methods of the same order of accuracy, especially predictor-corrector methods. Another disadvantage of the method is that neither the truncation errors nor estimates of them are obtained in the calculation procedure. Therefore, control of accuracy and adjustment of the step size h is done by comparison of the results due to double and single step size 2h and h.

Given the system of first-order ordinary differential equations:

$$y_1' = \frac{dy_1}{dx} = f_1(x, y_1, y_2, \ldots, y_n)$$

$$y_2' = \frac{dy_2}{dx} = f_2(x, y_1, y_2, \ldots, y_n)$$

$$\ldots\ldots\ldots\ldots$$

$$y_n' = \frac{dy_n}{dx} = f_n(x, y_1, y_2, \ldots, y_n)$$

and the initial values:

$$y_1(x_0) = y_{1,0},\ y_2(x_0) = y_{2,0},\ \ldots,\ y_n(x_0) = y_{n,0}$$

and using the following vector notations:

$$Y(x) = \begin{pmatrix} y_1(x) \\ y_2(x) \\ \cdot \\ \cdot \\ \cdot \\ y_n(x) \end{pmatrix},\quad F(x, Y) = \begin{pmatrix} f_1(x, Y) \\ f_2(x, Y) \\ \cdot \\ \cdot \\ \cdot \\ f_n(x, Y) \end{pmatrix},\quad Y_0 = \begin{pmatrix} y_{1,0} \\ y_{2,0} \\ \cdot \\ \cdot \\ \cdot \\ y_{n,0} \end{pmatrix}$$

where Y, F and Y_0 are column vectors, the given problem appears as follows:

$$Y' = \frac{dY}{dx} = F(x, Y) \text{ with } Y(x_0) = Y_0$$

With respect to storage requirements and compensation of accumulated roundoff errors, Gill's modification of the classical Runge-Kutta formulas is preferred. Thus, starting at x_0 with $Y(x_0) = Y_0$ and vector $Q_0 = 0$, the resulting vector $Y_1 = Y(x_0 + h)$ is computed by the following formulas:

$$K_1 = hF(x_0, Y_0) \quad ; Y_1 = Y_0 + \frac{1}{2}(K_1 - 2Q_0)$$

$$Q_1 = Q_0 + 3\left[\frac{1}{2}(K_1 - 2Q_0)\right] - \frac{1}{2}K_1$$

$$K_2 = hF(x_0 + \frac{h}{2}, Y_1) \; ; Y_2 = Y_1 + (1 - \sqrt{\tfrac{1}{2}})(K_2 - Q_1)$$

$$Q_2 = Q_1 + 3\left[(1 - \sqrt{\tfrac{1}{2}})(K_2 - Q_1)\right] - (1 - \sqrt{\tfrac{1}{2}})K_2 \qquad (1)$$

$$K_3 = hF(x_0 + \frac{h}{2}, Y_2) \; ; Y_3 = Y_2 + (1 + \sqrt{\tfrac{1}{2}})(K_3 - Q_2)$$

$$Q_3 = Q_2 + 3\left[(1 + \sqrt{\tfrac{1}{2}})(K_3 - Q_2)\right] - (1 + \sqrt{\tfrac{1}{2}})K_3$$

$$K_4 = hF(x_0 + h, Y_3) \; ; Y_4 = Y_3 + \frac{1}{6}(K_4 - 2Q_3)$$

$$Q_4 = Q_3 + 3\left[\frac{1}{6}(K_4 - 2Q_3)\right] - \frac{1}{2}K_4$$

where K_1, K_2, K_3, K_4, Y_1, Y_2, Y_3, Y_4, Q_1, Q_2, Q_3, Q_4 are all column vectors with n components. If the procedure were carried out with infinite precision (that is, no rounding errors), vector Q_4 defined above would be zero. In practice this is not true, and Q_4 represents approximately three times the roundoff error in Y_4 accumulated during one step. To compensate for this accumulated roundoff, Q_4 is used as Q_0 for the next step. Also $(x_0 + h)$ and Y_4 serve as x_0 and Y_0 respectively at the next step.

For initial control of accuracy, an approximation for $Y(x_0 + 2h)$ called $Y^{(2)}(x_0 + 2h)$ is computed using the step size 2h, and then an approximation called $Y^{(1)}(x_0 + 2h)$, using two times the step size h. From these two approximations, a test value δ for accuracy is generated in the following way:

$$\delta = \frac{1}{15}\sum_{i=1}^{n} a_i \cdot \left| y_i^{(1)} - y_i^{(2)} \right| \qquad (2)$$

where the coefficients a_i are error-weights specified in the input of the procedure.

Test value δ is an approximate measure for the local truncation error at point x_0+2h. If δ is greater than a given tolerance ε_2, increment h is halved and the procedure starts again at the point x_0. If δ is less than ε_2, the results $Y^{(1)}(x_0+h)$ and $Y^{(1)}(x_0+2h)$

Matrix Solution of Linear Equations

Size

Purpose:
Matrix solution of a set of simultaneous linear equations AX = B.

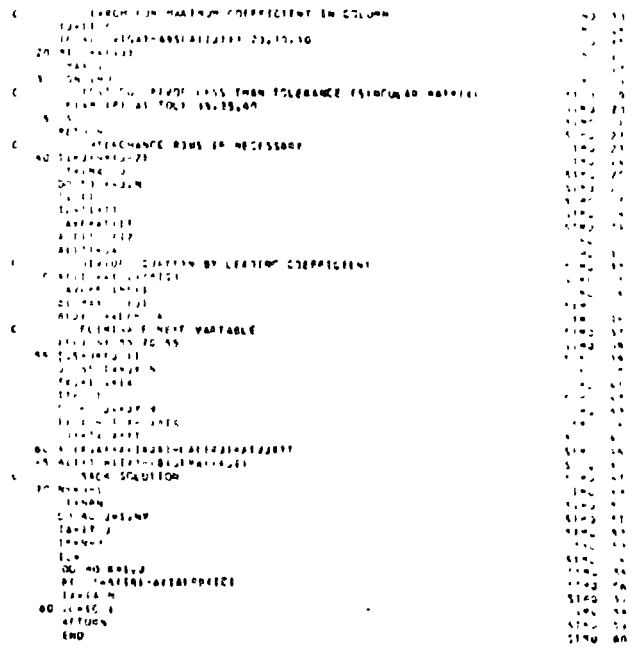
Usage:
CALL SIMQ(A, B, N, KS)

Description of Parameters:
A - Matrix of coefficients stored columnwise. These are destroyed in the computation. The size of matrix A is N by N.
B - Vector of original constants (length N). These are replaced by final solution values, vector X.
N - Number of equations and variables. N must be greater than 1.
KS - Output digit
0 For a normal solution.
1 For a singular set of equations.

Remarks:
Matrix A must be general.
If matrix is singular, solution values are meaningless.
An alternative solution may be obtained by using matrix inversion (MINV) and matrix product (GMPRD).

Subroutines and function subprograms required:
None.

Method
Method of solution is by elimination using largest pivotal divisor. Each stage of elimination consists of interchanging rows when necessary to avoid division by zero or small elements. The forward solution to obtain variable N is done in N stages. The back solution for the other variables is calculated by successive substitutions. Final solution values are developed in vector B, with variable 1 in B(1), variable 2 in B(2), ... variable N in B(N).
If the pivot can be found exceeding a tolerance of 0.0 the matrix is considered singular and KS is set to 1. This tolerance can be modified by replacing the first statement.



IEND - Maximum number of iteration steps specified.

IER - Resultant error parameter coded as follows:

IER=0 - no error

IER=1 - no convergence after IEND iteration steps

IER=2 - at some iteration step derivative DERF was equal to zero

Remarks:

The procedure is bypassed and gives the error message IER=2 if at any iteration step the derivative of F(X) is equal to 0. Possibly the procedure would be successful if it were started again with another initial guess XST.

Subroutines and function subprograms required:

The external subroutine FCT(X, F, DERF) must be furnished by the user.

Method:

Solution of the equation F(X)=0 is obtained by means of Newton's iteration method, which starts at the initial guess XST of a root X. Convergence is quadratic if the derivative of F(X) at root X is not equal to zero. One iteration step requires one evaluation of F(X) and one evaluation of the derivative of F(X). For tests on satisfactory accuracy see formula (2) of the mathematical description.

```
      SUBROUTINE SYNTEST(F,DERF,FCT,XST,EPS,IEND,IER)                   SYN1    1
C     SYNTEST 1 FEB 68                                                  SYN1    2
      IERR=0                                                            SYN1    3
      KEND=0                                                            SYN1    4
      IER=0                                                             SYN1    5
      CALL FCT(XST,F,DERF)                                              SYN1    6
      IF(DERF) 2,1,2                                                    SYN1    7
C     START ITERATION LOOP                                              SYN1    8
    3 DO 8 K=1,IEND                                                     SYN1    9
      KEND=K                                                            SYN1   10
C     F(X) IS NOT SATISFIED BY X                                        SYN1   11
    4 IF(ABS(F)-EPS) 9,9,5                                              SYN1   12
C     DERF=0 IS POSSIBLE                                                SYN1   13
    5 IF(DERF) 6,1,6                                                    SYN1   14
    6 DX=F/DERF                                                         SYN1   15
      XST=XST-DX                                                        SYN1   16
      CALL FCT(XST,F,DERF)                                              SYN1   17
C     TEST ON SATISFACTORY ACCURACY                                     SYN1   18
      IF(ABS(DX)-EPS) 7,7,8                                             SYN1   19
    7 IF(ABS(F)-EPS) 9,9,8                                              SYN1   20
    8 CONTINUE                                                          SYN1   21
      IER=1                                                             SYN1   22
      IF(IEND-1) 9,9,10                                                 SYN1   23
   10 IF(IEND-KEND) 9,9,11                                              SYN1   24
   11 IER=KEND-1                                                        SYN1   25
C     END OF ITERATION LOOP                                             SYN1   26
C     NO CONVERGENCE AFTER IEND ITERATION STEPS, ERROR RETURN           SYN1   27
      GOTO 9                                                            SYN1   28
    1 IER=2                                                             SYN1   29
C     ERROR RETURN IN CASE OF ZERO DIVISOR                              SYN1   30
    9 RETURN                                                            SYN1   31
      END                                                               SYN1   32
```

Mathematics - Roots of Polynomial

POLRT

This subroutine computes the real and complex roots of a real polynomial.

Given a polynomial

$$f(z) = \sum_{n=0}^{N} a_n z^n \tag{1}$$

Let

$Z = X + iY$ be a starting value for a root of f(z).

Then

$$Z^n = (X + iY)^n. \tag{2}$$

Define X_n as real terms of expanded equation (2).

Define Y_n as imaginary terms of expanded equation (2).

Then for:

$n = 0$

$X_o = 1.0$

$Y_o = 0.0$

$n > 0$

$$X_n = X \cdot X_{n-1} - Y \cdot Y_{n-1} \tag{3}$$

$$Y_n = X \cdot Y_{n-1} + Y \cdot X_{n-1} \tag{4}$$

Let U be the real terms of (1).

V be the imaginary terms of (1).

Then:

$$U = \sum_{n=0}^{N} a_n X_n \tag{5}$$

$$V = \sum_{n=0}^{N} a_n Y_n \tag{6}$$

or

$$U = a_o + \sum_{n=1}^{N} a_n X_n \tag{7}$$

$$V = \sum_{n=1}^{N} a_n Y_n \tag{8}$$

$$\frac{\partial U}{\partial X} = \sum_{n=1}^{N} n \cdot X_{n-1} \cdot a_n \tag{9}$$

$$\frac{\partial U}{\partial Y} = -\sum_{n=1}^{N} n\, Y_{n-1}\, a_n \tag{10}$$

Note that equations (3), (4), (7), (8), (9), and (10) can be performed iteratively for n = 1 to N by saving X_{n-1} and Y_{n-1}.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

METODOS NUMERICOS Y APLICACIONES EN LA COMPUTADORA DIGITAL

Marzo 29 de 1976.

Palacio de Minería
Tacuba 5, primer piso. México 1, D. F.
Tels.: 521-40-23 521-73-35 5123-123

* De acuerdo con las definiciones introducidas en la sección 4.5, las cantidades que aparecen en la última relación, son productos marginales. Puede por lo tanto establecerse para este caso el siguiente criterio de optimalidad.

En el capítulo 7, al estudiar el tema de estimación del valor de los insumos, se emplearán con frecuencia estos conceptos de costo e ingreso marginal.

En la siguiente sección se introduce la técnica de optimización matemática conocida con el nombre de programación lineal. Entre las técnicas de optimización, es esta probablemente la más empleada.

Existen muchos problemas de optimización cuyo modelo matemático es de tal naturaleza que se pueden resolver con la técnica de optimización conocida con el nombre de programación lineal. Se han desarrollado algoritmos y basados en ellos, programas de computadora digital para la solución de estos problemas.

* La estructura de los problemas que pueden resolverse con esta técnica es siempre la misma, de manera que contando con un buen programa

* Para maximizar el producto de insumos fijos.

Ingreso marginal = Costo marginal.

6.5 Programación lineal

6.5.1 Ejemplos

* Todos los problemas de programación

para la solución de estos pueden resolverse sin necesidad de tener que escribir programas especiales para la solución de problemas particulares.. Los problemas de optimización que se pueden resolver con la técnica de programación dinámica por otra parte no tiene esta característica y con frecuencia es necesario desarrollar programas particulares para obtener la solución de un problema específico.

En esta sección se empezará a ilustrar con ejemplos la formulación de modelos matemáticos que permiten aplicar al programación lineal para optimizarlos. Posteriormente se estudia la forma normal de modelos de programación Li- [illegible] A continuación, la ilustración geométrica de la solución del problema de programación lineal, sirve para introducir el método simplex de solución de problemas.

El primer ejemplo ilustra un problema de transporte. Supóngase que una embotelladora tiene dos plantas , una en Tlaxcala y otra en Tehuacán, con capacidad de 7000 y 13000 cajas de refrescos al día, además tiene dos centros de consumo que son Puebla y Orizaba, que pueden consumir hasta 12000 y 8000 cajas diarias respectivamente. El costo de envio de una caja de refresco de los diferentes lugares de producción a los diferentes destinos está dado en la tabla 6.5.1.

Ejemplo 6.5.1

7000 Tlaxcala

12 000 Puebla

13000

8000

a \ de	Tlaxcala 1	Tehuacán 2
Puebla 1	0.8	1.00
Orizaba 2	1.30	0.90

Tabla 6.5.1 Costos de Transporte en el ejemplo 6.4.1

El administrador de la empresa debe determinar cuantas cajas deben enviarse de cada embotelladora a cada centro de consumo, de manera que se satisfagan las siguientes condiciones :

1). Cada embotelladora no puede enviar más cajas que el máximo que puede producir.

2). Cada centro de consumo puede obtener tantas cajas - como puede consumir.

3). Deben minimizarse los gastos de transporte.

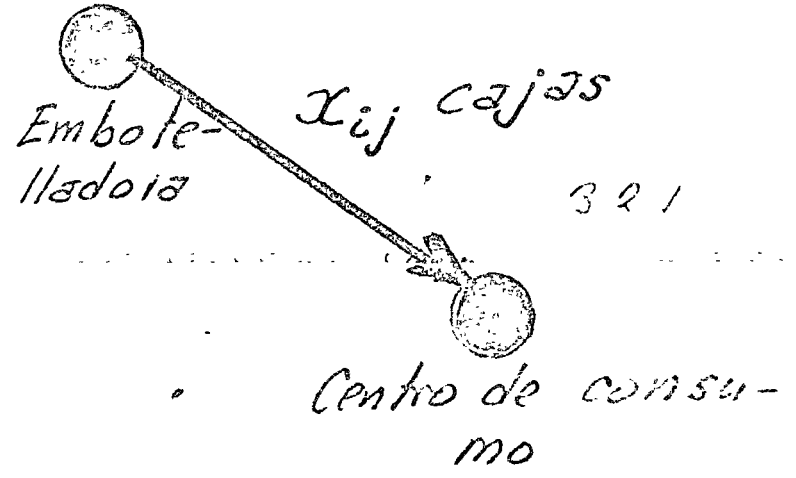
Para plantear este problema en el marco de las ecuaciones (6.1.4) y --

(6.1.2)

$$M = M(x_1, x_2, \ldots x_n) \quad (6.1.1)$$

$$C_i = C_i(x_1, x_2, \ldots x_n) \geq 0 \text{ para } i = 1, 2, \ldots p$$
$$C_i = C_i(x_1, x_2, \ldots x_n) \leq 0 \text{ para } i = p+1, \ldots r$$
$$C_i = C_i(x_1, x_2, \ldots x_n) = 0 \text{ para } i = r+1, \ldots n \qquad (6.1.2)$$

es necesario definir la siguiente variable : x_{ij} es el número de cajas - enviadas de la embotelladora situada en la localidad i'sima -------- (i = 1 corresponde a Tlaxcala e i = 2 a Tehuacán) al centro consumidor j'simo (1 es el índice de Puebla y 2 el de Orizaba). Con la introducción de esta variable el problema puede plantearse de la siguiente forma :



Las cajas enviadas de la localidad 1 (Tlaxcala) al centro de consumo 1 (Puebla), que se ha acordado representar con x_{11}, más las cajas enviadas de la localidad 1 al centro de consumo 2 (Orizaba), x_{12}, no deben exceder la capacidad de la embotelladora de la localidad 1 que es de 7000 cajas, es decir

$$x_{11} + x_{12} \leq 7\,000 \qquad (6.5.1)$$

La figura 6.5.1 ilustra el planteamiento de esta ecuación :

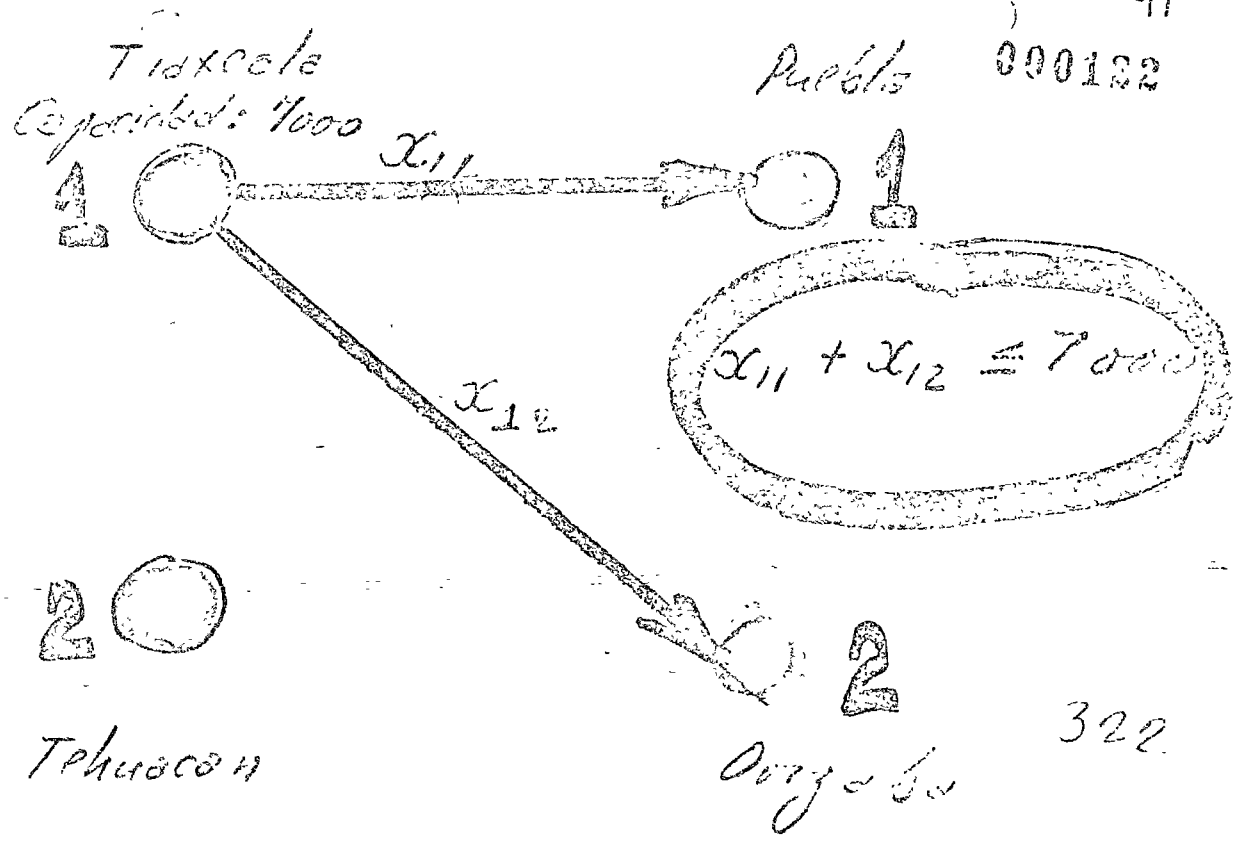


Fig. 6.5.1 Cajas enviadas desde la embotelladora en Tlaxcala.

En forma similar puede establecerse la siguiente ecuación que limite la producción total de la embotelladora de la 2da. localidad a 13000 cajas, a - saber :

$$x_{21} + x_{22} \leq 13\,000 \quad (6.5.2)$$

La figura 6.5.2. ilustra el planteamiento de otras ecuaciones.

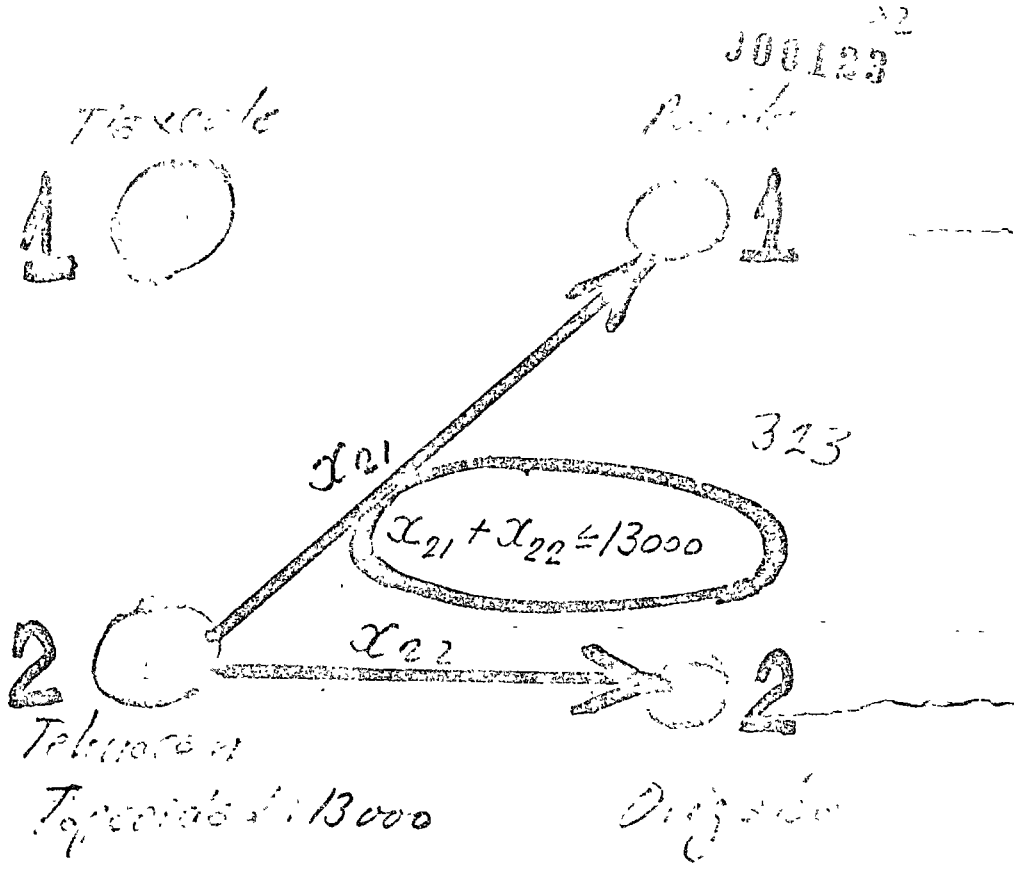


Fig. 6.5.2 Cajas enviadas desde la embotelladora en Tehuacán

Por otra parte, se ha señalado que cada centro de consumo puede obtener antas cajas como desea.

Al centro consumidor 1, Puebla, le llegan x_{11} cajas de Tlaxcala y x_{21} cajas de Tehuacán tal como ilustra la fig. 6.5.3 . Por lo tanto, como el consumo de Puebla es de 12,000 cajas :

$$x_{11} + x_{21} \geq 12000 \quad (6.53)$$

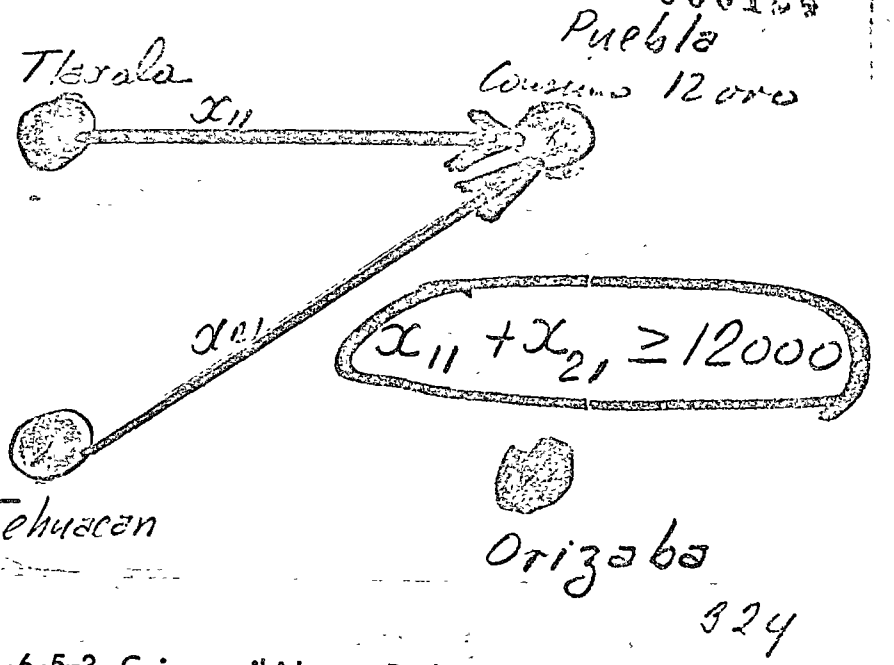


Fig. 6.5.3 Cajas recibidas en Puebla.

Finalmente como última restricción se tiene que las cajas que recibe Orizaba del centro consumidor 2, deben ser iguales o mayor a 8000 cajas. Se tiene por lo tanto;

$$x_{12} + x_{22} \geq 8000 \quad (6.5.4)$$

La figura 6.5.4 ilustra el significado de esta ecuación

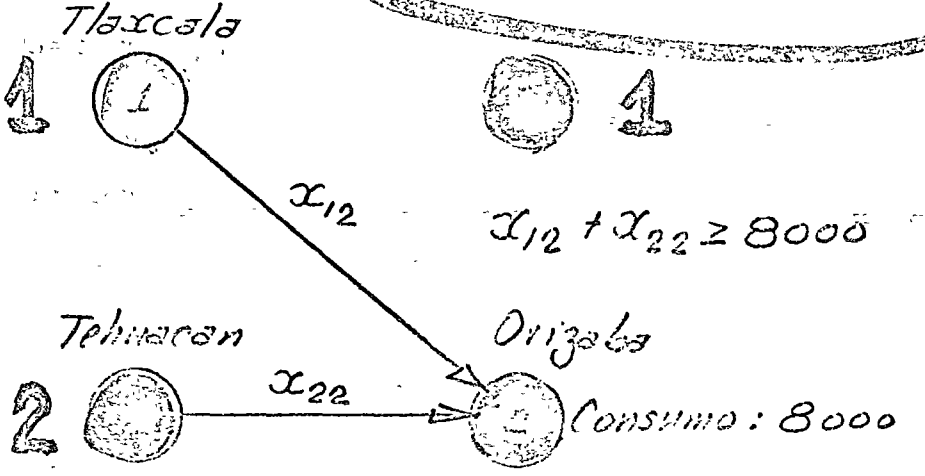


Fig. 6.5.4 Cajas recibidas por Orizaba.

Para terminar con el establecimiento del modelo matemático de este problema es necesario establecer la función objetivo.

El objetivo de análisis es minimizar los costos de transporte que están dados por:

$$M = 0.8 X_{11} + 1 X_{21} + 1.3 X_{12} + 0.9 X_{22} \qquad (6.5.5)$$

326

Debe además imponerse la siguiente condición:

$$X_{ij} \geq 0 \quad \forall i, \forall i \qquad (6.5.6)$$

327

ya que no tendrá significado valores negativos de envios de cajas.

En resumen puede decirse que el problema consiste en minimizar la función objetivo.

$$M = 0.8x_{11} + 1.x_{21} + 1.3x_{12} + 0.9x_{22} \quad (6.5.5)$$

Sujeto a las restricciones

$$x_{11} + x_{12} \leq 7000 \quad (6.5.1)$$

$$x_{21} + x_{22} \leq 13,000 \quad (6.5.2)$$

$$x_{11} + x_{21} \geq 12,000 \quad (6.5.3)$$

$$x_{12} + x_{22} \geq 8,000 \quad (6.5.4)$$

$$x_{ij} \geq 0 \quad \forall i, \forall j \quad (6.5.6)$$

Todos los modelos matemáticos de problemas de programación lineal tienen precisamente esta forma.

Antes de continuar conviene recordar algunas definiciones introducidas en la sección 6.1.2

* Un conjunto de valores de las variables que satisface todas las restricciones del problema se llama una solución factible del problema de programación lineal. Empleando la difinición anterior, puede decirse que la solución del problem consiste en encontrar una so-

* La solución factible satisface todas las restricciones

lución factible que sea óptima. En este caso del problema del transporte (solución factible) que minimice la función objetivo (6.5.5).

* Este problema tiene cuatro variables que hay que determinar, x_{11}, x_{12}, x_{21} y x_{22}, con objeto de visualizar geométricamente la solución de los problemas de programación lineal e introducir otro tipo de problemas de optimización de este tipo, se incluye un segundo ejemplo:

* Variables del problema
x_{11}, x_{12}, x_{21} y x_{22}

330

Ejemplo 6.5.2

331

* Supóngase que una compañía de transporte tiene x_1 camionetas de 2 toneladas y x_2 camionetas de 4 toneladas y desea maximizar su capacidad de transporte. La función objetivo es (medida) consiste en maximizar dicha expresión.

* x_1 camionetas de 2 ton 332
x_2 camionetas de 4 ton

$$m = 2x_1 + 4x_2 \quad (6.5.7)$$

* Además la compañía tiene las siguientes restricciones:

* La primera es la siguiente: Las camionetas chicas requieren 1 día de mantenimiento al mes, y las grandes 4 días y la compañía solo tiene disponibles 24 días de mecánico al mes. Matemáticamente esta restricción se expresa de la siguiente forma:

* Restricciones 333
* 1era: Mantenimiento:
24 días mecánico/mes

$$x_1 + 4x_2 \leq 24 \quad (6.5.8)$$

* La segunda restricción en este problema se refiere a la disponibilidad de andenes de carga. Ambos tipos de vehículo, requieren de igual número de andenes de carga, y que la compañía solo cuenta con 9 andenes. Empleando las variables X_1 y X_2 está restricción establece :

* 2da Andenes de carga: 9 andenes

$$x_1 + x_2 \leq 9 \quad (6.5.9)$$

* La última restricción se refiere al personal que se requiere para cargarlas. Este -- personal que está restringido a 21 personas . Las camionetas chicas requieren tres personas para cargarlas y las grandes solamente una persona. Se tiene por lo tanto

* 3ra Cargado: 21 personas

$$3x_1 + x_2 \leq 21 \quad (6.5.10)$$

* Desde luego que las variables X_1 y X_2 , número de camionetas de 2 - toneladas y de 4 toneladas con que cuenta la compañía respectivamente, no pueden ser negativas, por lo tanto las últimas restricciones en este problema son :

* Ultima: no negatividad

$$x_1 \geq 0 ; x_2 \geq 0 \quad (6.5.11)$$

Desde luego existen otros muchos problemas donde puede aplicarse la programación lineal. Entre ellos pueden citarse problemas de mezclado y planeación de la -- producción como el ejemplo 6.5.4 de la sección 6.5.5.

Después de estos ejemplos se procederá a plan- - tear en forma formal el problema de programación lineal y se estudiarán las condiciones que debe satisfacer tanto la función objetivo como las restricciones.

6.5.2 Planteamiento Formal

* Si se analiza la formulación de los problemas de los dos ejemplos introducidos en la sección anterior, _ pueden detectarse ciertas variables que se llaman en forma genérica actividades.

* En el ejemplo 6.5.1 las actividades consisten en envíar cajas de refrescos de la embotelladora al centro consumidor y se han representado con los símbolos:

x_{ij} , $i, j = 1, 2$

* En el ejemplo 6.5.2 estas actividades consisten - en operar camiones de carga y se han empleado los símbolos x_1 y x_2 para representarlas.

* Cada actividad queda caracterizada por una variable que se designa como nivel de actividad.

Además se observa que los problemas de los ejemplos anteriores satisfacen las siguientes condiciones:

393

1.- No negatividad de los niveles
es decir $x_i \geq 0, \forall_i$

2.- Linealidad.

*Tanto las restricciones como la función objetivo son funciones lineales de los niveles de actividad. Al ser lineales estas funciones son <u>homogéneas y aditivas</u>.

*Funciones objetivo y restricciones
son lineales ⟹ homogéneas y aditivas

394

Una función

$f(x_1, x_2, \ldots x_n)$

es lineal si dados dos conjuntos

395

*Conjuntos de variables

396

x_i, $i = 1, 2, \ldots n$ y x_i' $i = 1, 2, \ldots n$

*y dos constantes cualquiera K y K' se tiene:

*Constantes K y K'

397

$$f(Kx_1 + K'x_1', \ldots Kx_n + K'x_n') = Kf(x_1, x_2, \ldots, x_n) + K'f(x_1', x_2', \ldots x_n') \quad (6.5.12)$$

*La condición de linealidad (6.5.12) es equivalente a dos condiciones. En primer lugar una función lineal tiene un factor constante de escala es decir.

*Condición de linealidad → factor constante de escala

$$f(Kx_1, Kx_2, \ldots Kx_n) = Kf(x_1, x_2, \ldots, x_n) \quad (6.5.13)$$

*y es en segundo lugar es aditiva:

*Condición de linealidad → aditividad

$$f(x_1+x_1', x_2+x_2', \ldots x_n+x_n') = f(x_1, x_2, \ldots x_n) + f(x_1', x_2', \ldots x_n') \quad (6.5.14)$$

Un ejemplo servirá para ilustrar es importante concepto y señalar que funciones del tipo

$$f(x) = a + bx \quad (6.5.15)$$

* no son lineales. Es decir, si en las funciones hay cargos fijos (el término a) no es posible aplicar directamente el concepto de programación lineal.

Ejemplo 6.5.3

Determine si las siguientes funciones son lineales y justifique la respuesta.

a) $y = 3x_1 + 2x_2$

b) $y = 3x + 5$

Solución:

a) Como $a3X_1 + b3X_1 + a2X_2 + bX_2$

$= a(3X_1 + 2X_2) + b(3X_1 + 2X_2)$

se cumple la condición (6.5.12) y la función es lineal.

b) Como $a3X + 5 + b3X + 5 \neq a\,(3X+5) + b(3X+5)$

la función no es lineal.

El problema de programación lineal por lo tanto - puede plantearse de la siguiente forma.

* Hay que determinar el valor de los niveles de actividad $x_1, x_2, \ldots x_n$, que maximicen a la función objetivo:

* Encontrar x que maximice:

$$m = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n \quad (6.5.16a)$$

y satisfaga:

* sujeto a las siguietes resticciones:

* restricciones

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n = b_i \quad i = 1, 2, \ldots p$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n \leq b_i \quad i = p+1, \ldots r \quad (6.5.16b)$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n \geq b_i \quad i = r+1, \ldots m$$

357

$$x_j \geq 0 \quad j = 1, 2, \ldots n$$

* Los coeficientes C_i de la función objetivo se nococen con el nombre de coeficientes de costo, y los coeficientes a_{ij} de las ecuaciones de restricción se llaman coeficientes estructurales.

* $C_i \equiv$ coeficientes de costo

$a_{ij} \equiv$ coeficientes estructurales

353

Como se ilustra en el ejemplo 6.5.3 un problema de maximización puede convertirse en un problema de minimización. Como muestra el sistema de ecuaciones (6.5.16) las restricciones pueden ser del tipo de desigualdad o igualdad. * Para la solución del problema de programación lineal conviene convertir todas las desigualdades en igualdades introduciendo variables -

354

* Variables de holgura ≥ 0 para convertir desigualdades en

000134

de holgura, que de preferencia deben de ser positivas. La siguiente desigualdad:

puede convertirse en una igualdad introduciendo una variable positiva x_{n+q} llamada de holgura, En efecto :

* Si por otra parte se tiene en la ecuación de restricción la ~~holgura~~ desigualdad en sentido contrario.

la introducción de la variable de holgura positiva - x_{n+q}, convierte la desigualdad [en una igualdad], ya que:

Además los métodos de solución del problema de programación lineal exigen que los niveles de actividad sean positivos, es decir $x_i \geq 0, \forall i$. * Si un nivel de actividad no está sujeto a esta restricción se le puede sustituir por la diferencia de dos niveles de actividad positivos. Supongamos que el nivel x_i - no está restringido. Si se introducen las variables

desigualdades

desigualdad

$a_{q1}x_1 + a_{q2}x_2 + \dots + a_{qn}x_n \leq b_q$

+

variable de holgura 360

$x_{n+q} > 0$

↓

igualdad

$a_{q1}x_1 + a_{q2}x_2 + \dots + a_{qn}x_n + x_{n+q} = b_q$

* Desigualdad 361

$a_{q1}x_1 + a_{q2}x_2 + \dots + a_{qn}x_n \geq b_q$

+

variable de holgura

$x_{n+q} > 0$

↓

igualdad

$a_{q1}x_1 + a_{q2}x_2 + \dots + a_{qn}x_n - x_{n+q} = b_q$

362

* Si nivel de actividad

$x_i \leq 0 \geq 0$

000135

x_i^+ y x_i^- relacionadas con la variable x_i mediante la siguiente diferencia.

$$x_i = x_i^+ - x_i^- \qquad (6.5.17)$$
$$x_i^+ \geq 0,\ x_i^- \geq 0$$

la variable ó nivel de actividad original puede ser mayor, igual o menor que cero, sin que las variables x_i^+ y x_i^- tomen valores negativos. El siguiente ejemplo ilustra tanto la introducción de variable de holgura como el empleo de la relación (6.5.17) y la transformación de un problema de minimización en uno de maximización.

Ejemplo 6.5.3

Convierta el siguiente problema de minimización en un problema de maximización, transforme todas las ecuaciones de restricción en igualdades mediante la introducción de variables de holgura y transforme en las variables no negativas.

$$\min: m = 2x_1 + 5x_2$$
$$3x_1 + 2x_2 \geq 6$$
$$x_1 - 6x_2 \leq 4$$
$$x_1 \geq 0;\ x_2 \text{ sin restricción}$$

367 000136

Solución:

se sabe que:

Min. $m=3x_1 + 5x_2$ es equivalente a :

Max. $-m=-3x_1 - 5x_2$.

368

Definiendo una nueva función de objetivo.

* Nueva función objetivo n: 369

$n \equiv -m$ 369

la función objetivo se convierte en:

$\max: n = -3x_1 - 5x_2$ 370

Para convertir las dos desigualdades de restricción en igualdad es necesario introducir dos nuevas variables x_3 y x_4 para realizar los siguientes cambios en las restricciones.

$3x_1 + 2x_2 \geq 6 \longrightarrow 3x_1 + 2x_2 - x_3 = 6$

$x_1 - 6x_2 \leq 4 \longrightarrow x_1 - 6x_2 + x_4 = 4$ 371

* Finalmente la variable x_2, no restringida debe sustituirse por la diferencia de dos variables no negativas $x_2 = x_2^+ - x_2^-$

* x_2 variable sin restricción 372

Realizando esta sustitución, las ecuaciones ó condiciones de restricción tienen la siguiente forma:

$3x_1 + 2x_2^+ - 2x_2^- - x_3 = 6$

$x_1 - 6x_2^+ + 6x_2^- + x_4 = 4$

$x_1, x_2^+, x_2^-, x_3, x_4 \geq 0$

Y la función objetivo es: $\max: n = -3x_1 - 5x_2$

También es posible resolver un problema de minimización recurriendo a su formulación dual que se estudia en la sección 6.5.5.

* La estructura del problema de programación lineal se presta para el empleo de la notación matricial. Si se definen* la matriz de coeficientes estructurales

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} \ldots a_{1n} \\ a_{21} \ldots & \\ \vdots & \\ a_{m1} & a_{m2} \ldots a_{mn} \end{bmatrix} \quad (6.5.17)$$

y los* vectores de actividades:

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (6.5.18)$$

de costos*

$$\underline{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \quad (6.5.19)$$

* Formulación matricial

* Coeficientes estructurales

* Actividades

* costos

* Restricciones

000138

x

y de restricciones

$$\underline{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \quad (6.5.20)$$

379

El problema de programación lineal queda planteado de la siguiente forma:

$$\max: m = \underline{c}^T \underline{x} \quad (6.5.21)$$

380

Sujeto a las restricciones

$$\underline{A}\,\underline{x} \leq \underline{b} \quad (6.5.22)$$

$$\underline{x} \geq \underline{0} \quad (6.5.23)$$

381

En la siguiente sección se ilustra gráficamente la forma de obtener la solución del problema de programación lineal.

6.5.3 Solución gráfica.

383

En esta sección ilustraremos gráficamente la solución del problema de programación lineal. Como es difícil representar gráficamente funciones de más de dos variables, se empleará el ejemplo 6.5.2 para realizar esta representación.

El modelo matemático de este problema es el si--

./.

guiente:

$$\max: \; m = 2x_1 + 4x_2 \quad (6.5.7)$$

Sujeto a las restricciones

$$x_1 + 4x_2 \leq 24 \quad (6.5.8)$$

$$x_1 + x_2 \leq 9 \quad (6.5.9)$$

$$3x_1 + x_2 \leq 21 \quad (6.5.10)$$

$$x_1, x_2 \geq 0$$

Las restricciones de este problema establecen una zona del plano (x_1, x_2) donde deben encontrarse las soluciones factibles, tal como se señaló en la sección 6.1.2. Observe que la ecuación $x_1 + 4x_2 = 24$ corresponde a una recta, que divide al plano en dos regiones. En la inferior se cumple $x_1 + 4x_2 \leq 24$, por lo tanto la solución factible debe estar "abajo" de dicha recta. La figura 6.5.5 ilustra la zona definida por esta restricción.

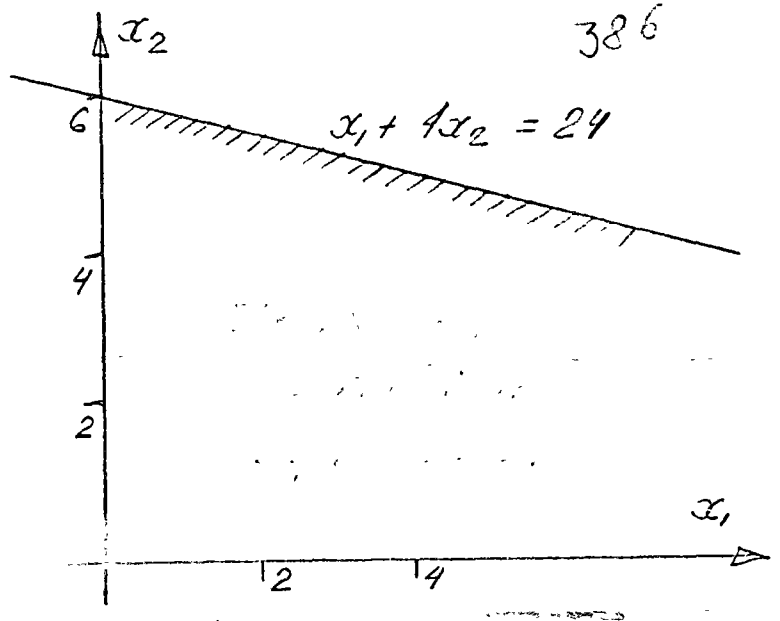


Fig. 6.5.5.- Zona con restricción $x_1 + 4x_2 \leq 24$

Un razonamiento similar lleva a concluir que la solución factible también debe estar a la "izquierda" de las rectas $x_1 + x_2 = 9$ y $3x_1 + x_2 = 21$ (fig 6.5.6)

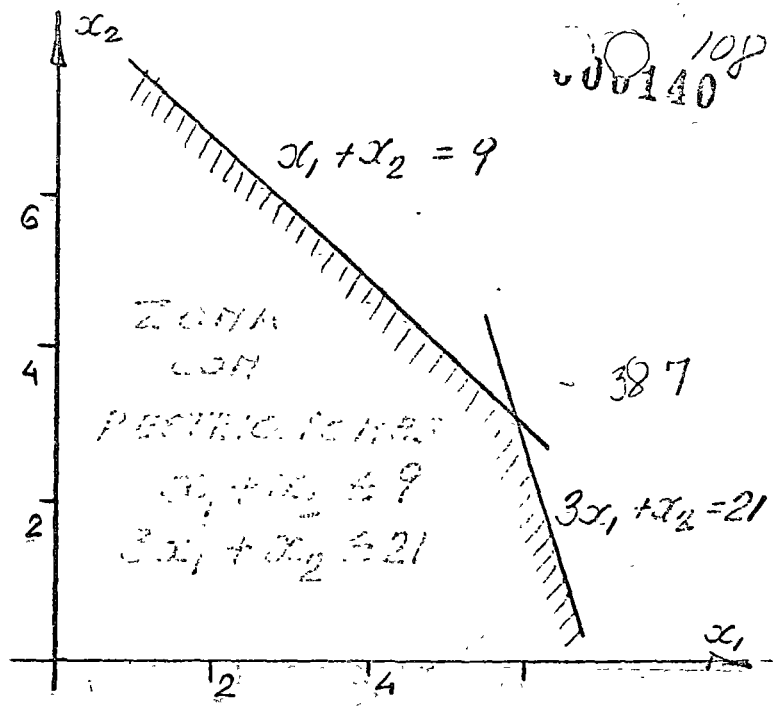


Fig.6.5.6 Zona con restricciones $x_1 + x_2 \leq 9$ y $3x_1 + x_2 \leq 21$

Además la condicion $x_1 \geq 0$ y $x_2 \geq 0$ impone que debe estar en el primer cuadrante. La región del plano donde se cumplen todas las restricciones es por lo tanto polígono convexo OA BC DO que aparece en la figura 6.5.7.

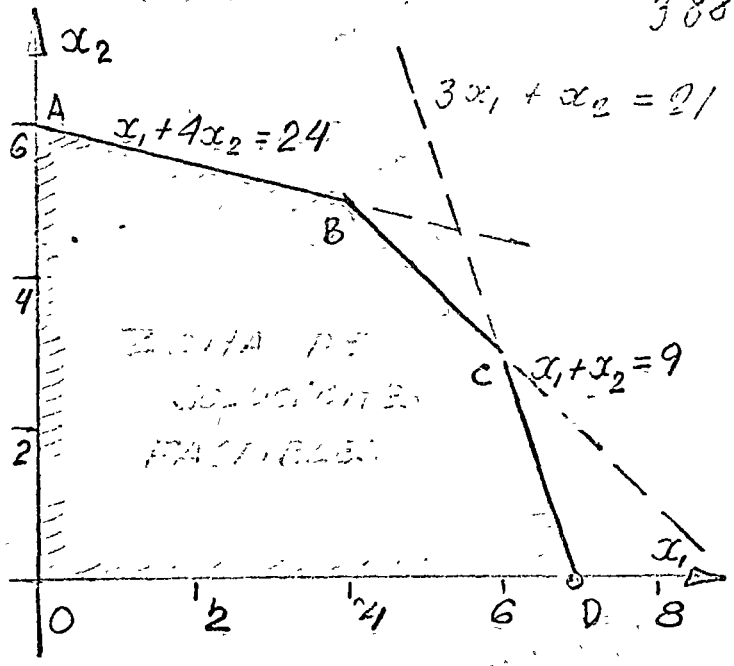


Fig.6.5.7 Zona de soluciones factibles del ejemplo 6.5.2

./.

El siguiente paso en la solución consiste en encontrar dentro de los puntos de dicho polígono, -- que son soluciones factibles todos ellos, aquel punto para el cua l la función objetivo 6.5.7 $2X_1+4X_2$ - es máxima. Nótese primero que cualquier recta dependiente $-\frac{2}{4}$ cumple con la condición $2X_1 + 4X_2$. Además entre mayor sea la distancia al origen de una recta dependiente $-\frac{1}{2}$, tanto mayor es $2X_1 + 4X_2$ tal como se ilustra en la figura 6.5.8.

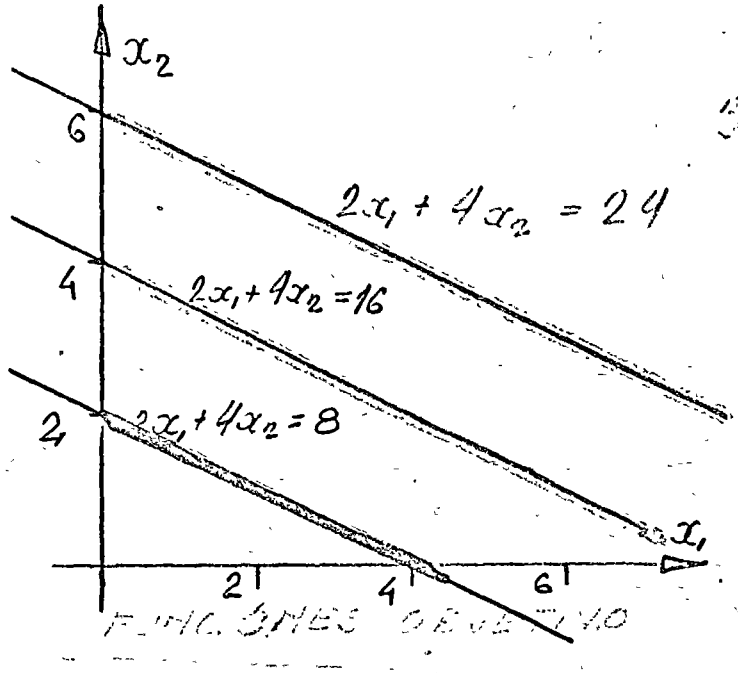


Fig. 6.5.8 Funciones objetivo del ejemplo 6.5.2

Para obtener el valor máximo de la función objetivo $2X_1 + 4X_2$ es necesario desplazar una recta dependiente $-\frac{1}{2}$ de manera que su distancia al - origen sea máxima, pero tenga por lo menos un punto - dentro de la región OABCDO. En la figura 6.5.9 se - - ilustra este procedimeitno de búsqueda del máximo. En el punto B de coordenadas (4,5) el valor de la fun- -

390

ción objetivo $2x_1 + 4x_2$ es de 28 y se cumplen todas las restricciones. Por lo tanto $x_1 = 4$, $x_2 = 5$ es - la solución del problema de programación lineal. - Haciendo referencia a la fig. 6.5.9 obsérvese además que para dicho punto, tiene las características resúmidas en el cuadro (de la tabla) 6.5.1

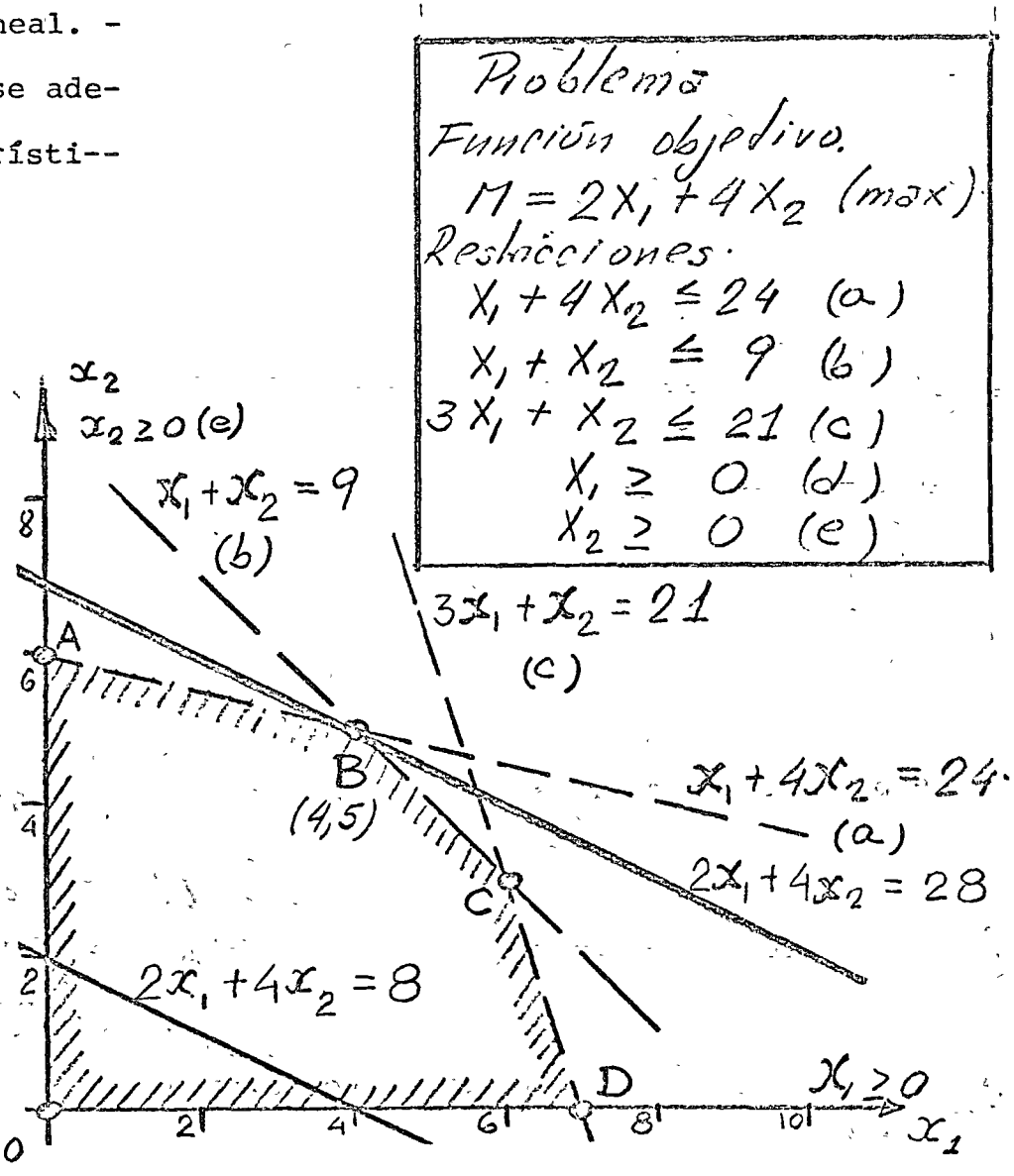


Fig. 6.5.9 Ilustración de la solución gráfica del problema de programación lineal

Restricción		Holgura
$x_1 + 4x_2 =$	24	0
$x_1 + x_2 =$	9	0
$3x_1 + x_2 =$	$17 \leq 21$	4

Tabla 6.5.1 Propiedades de punto óptimo B del ejemplo 6.5.2

Es decir, el recurso mecánico "del que se cuenta con 24 días más, el de "andenes de carga" con el --

que se cuenta con 9, se emplea plenamente si se usan 4 camionetas de dos toneladas y 5 de 4 toneladas. -- Mientras que de tercer recurso, del que se cuenta -- con 21 unidades, solo se usan 17. Sin embargo ninguna otra combinación de x_1 y x_2 permite obtener mayor volumen de carga sin violar las restricciones (6.5.8)-(6.5.10). Antes de continuar, nótese que la región -- definida por las restricciones (6.5.8)-6.5.10) es convexa, como muestra la figura 6.5.10, ya que cualquier recta que une dos puntos cualquiera de la periferia = de la zona se encuentra en la frontera o dentro de la región.

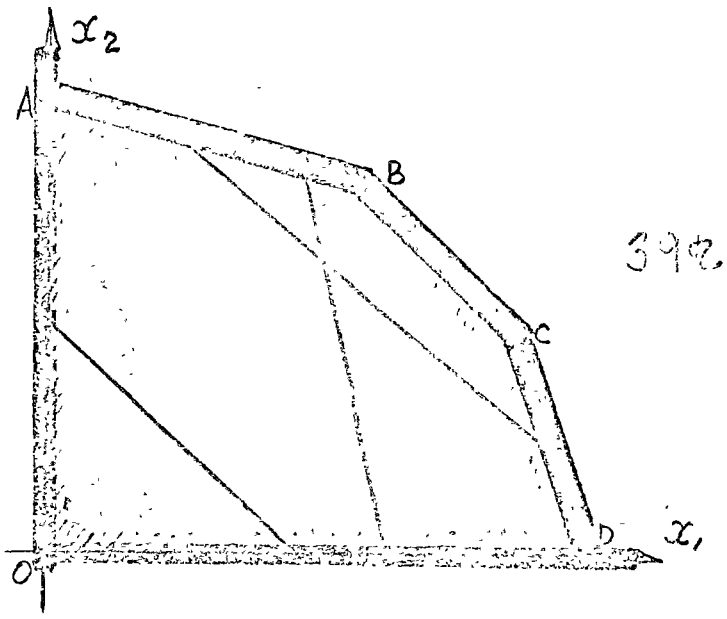


Fig. 6.5.10 Zona convexa de soluciones factibles

En la sección 6.5.4 se empleará la representación gráfica de la solución del programación lineal - para visualizar fácilmente diversos casos especiales de problemas de este tipo.

./.

* El método gráfico de solución del problema de programación lineal está restringido a modelos con dos variables. Prácticamente todos los problemas de interés para el analista tienen más de dos variables, por lo cual el método gráfico no se puede emplear en estos casos.* Es necesario contar con métodos algebraicos que se puedan programar en una computadora digital, con objeto de resolver problemas con un gran número de variables, como son la mayoría de los que se encuentran en la práctica. El método simplex que se introduce en la siguiente sección tiene esta propiedad. Sin embargo es importante familiarizarse con la solución gráfica estudiada en esta sección, ya que ayuda a entender la naturaleza de la solución del problema.

Al ir desarrollando el método simplex de solución analítica, continuamente se hará referencia a la solución gráfica. Los autores consideran que de esta forma el lector (lo) comprenderá con mayor facilidad.

* Método gráfico para problemas con dos variables 393

* Métodos algebraicos para resolver sistemas con muchos variables 394

Foto 395

396

5.4 Solución analítica

El método analítico más importante para

solución de este tipo de problemas es el método Simplex, que introduciremos resolviendo el ejemplo 6.5.2.

* Método Simplex

397 000146

* La función objetivo de este ejemplo es:

max: $m=2x_1 + 4x_2$

* Función objetivo

max: $m=2x_1+4x_2$ (6.5.7) 398

* Sujeto a las restricciones

* Restricciones

$x_1 + 4x_2 \leq 24$ (6.5.8) 399

$x_1 + x_2 \leq 9$ (6.5.9)

$3x_1 + x_2 \leq 21$ (6.5.10)

$x_1, x_2 \geq 0$

* El primer paso en este método consiste en introducir variables de holgura x_3, x_4, x_5 para convertir las desigualdades de las ecuaciones de restricción en igualdades, tal como se señaló en la sección 6.5.2

* Introduzca variables de holgura x_3, x_4 y x_5 400

$x_1 + 4x_2 + x_3 = 24$

$x_1 + x_2 + x_4 = 9$ (6.5.24) 401

$3x_1 + x_2 + x_5 = 21$

* Debido al signo de las desigualdades, las variables de holgura deben ser positivas, es decir:

* Variables de holgura positivas 402

$x_3, x_4, x_5 \geq 0$ (4.5.3) 403

* El problema consiste en encontrar los valores de las variables x_j que maximicen a la función objetivo (6.5.7).

* Como el sistema (6.5.24) tiene 3 ecuaciones - con 5 incognitas pueden expresarse 3 de ellas cualesquiera en función de las dos restantes.

* Como la variable x_3 solo aparece en la 1er. - ecuación, la x_4 en la 2da. y x_5 en la 3er. ecuación - lo más conveniente es tomar $x_1=0$ y $x_2=0$, obteniéndose de inmediato del sistema (6.5.24) que $x_3=24$, $x_4=9$ y - $x_5=21$. Esta solución se conoce con el nombre de una - solución básica, y las variables cuyo valor se ha fijado reciben el nombre de variables base. Teniendo -- presente la definición de solución factible, se nota que el conjunto $x_1=0$, $x_2=0$, $x_3=24$, $x_4=9$ y $x_5=21$ es -- una solución factible aunque no óptima, ya que en este caso la función objetivo vale $m=0$

* Encontrar x_j para maximizar $2x_1+4x_2$

* Sistema de 3 ecuaciones con 5 incógnitas

* $x_1 + 4x_2 + x_3 = 24$
$x_1 + x_2 + x_4 = 9$
$3x_1 + x_2 + x_5 = 21$

Si $x_1 = x_2 = 0$
$x_3 = 24$, $x_4 = 9$, $x_5 = 21$

* Variables cuyo valor se fija ($x_1, x_2 = 0$) se llaman variables base.

* $x_1 = x_2 = 0$, $x_3 = 24$
$x_4 = 9$, $x_5 = 21$ son una solución factible no óptima

Haciendo referencia a la figura 6.5.11 que -- muestra gráficamente la región donde se cumplen las - restricciones (6.5.8) a (6.5.10), se observa que la - solución básica $x_1=x_2=0$ y $x_3=24$, $x_4=9$ y $x_5=21$ corresponde al origen del sistema. Nótese además que el valor de las variables de holgura indica que no se está empleando ningún recurso en este punto.

404

Restricción	Valor en O	Holgura
$x_1+4x_2\leq 24$	$x_1+4x_2=0$	24
$x_1+x_2\leq 9$	$x_1+x_2=0$	9
$3x_1+x_2\leq 21$	$3x_1+x_2=0$	21

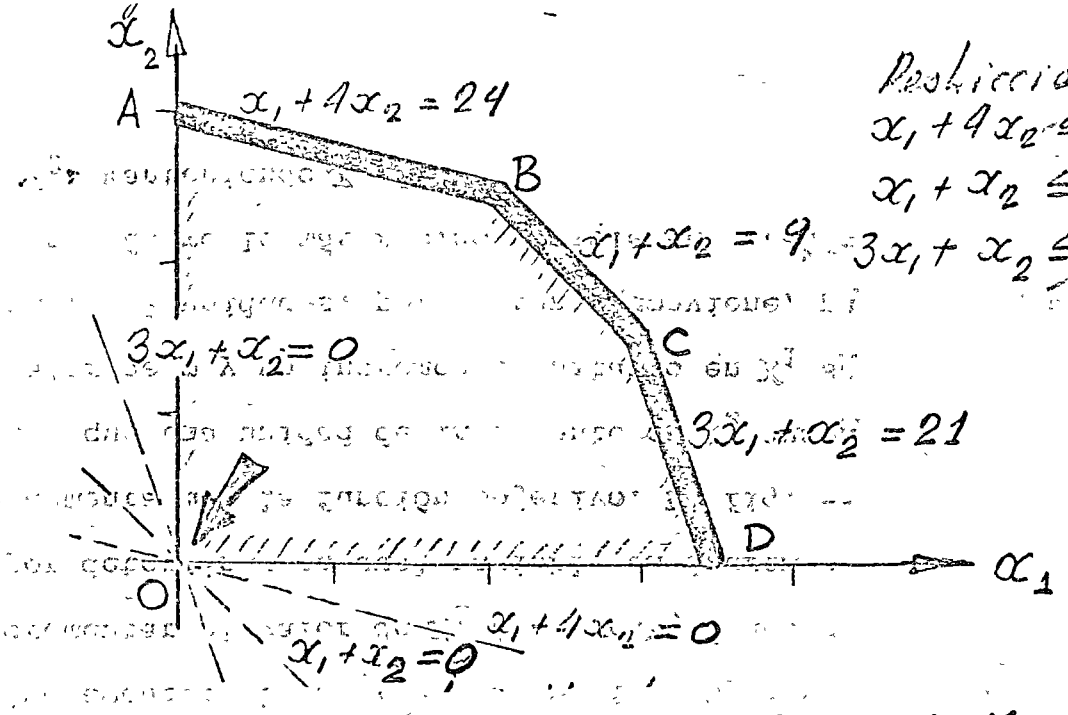


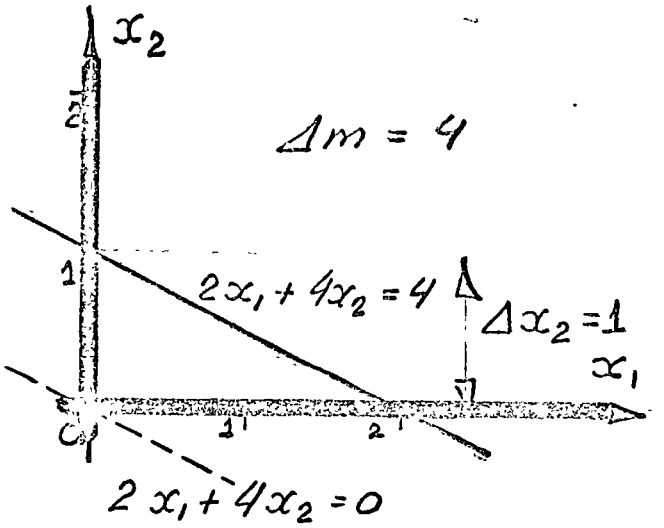
Fig. 6.5.11 Valor de las funciones de restricción en el punto de solución básica.

* Para incrementar el valor de la función objetivo se puede incrementar el valor de x_1 ó el de x_2 ó ambas. Se empieza por determinar en cual variable un incremento unitario aumenta más la función objetivo. La fig. 6.5.12 ilustra que una unidad de incremento en x_2 aumenta en 4 el valor de m y un incremento unitario en x_1 solo aumenta a m en 2 unidades, por lo tanto conviene, para encontrar el máximo lo más rápido posible aumentando el valor de x_2, manteniendo $x_1=0$.

* m ↑ si x_1 ↑ y/o x_2 ↑

410

000140



411

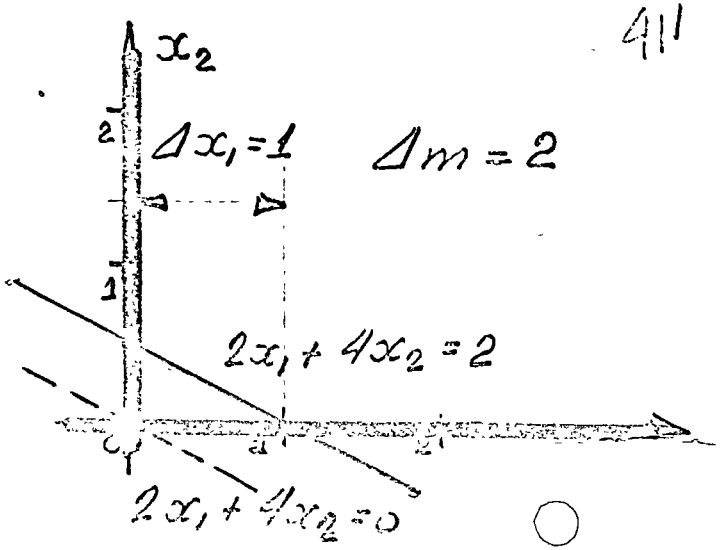


FIG. 6.5.12 Incremento de la función objetivo

Solución factible

$x_2 = 6,\ x_4 = 3,\ x_5 = 15$

Incremento inicial en x_1 → Incremento en $M = 1$

" " en x_3 → Decremento en M

$x_3 = 0$	x_1 max	
$x_2 = 6 - \frac{1}{4}x_1$	24	485
$x_4 = 3 - \frac{3}{4}x_1$	4	
$x_5 = 15 - \frac{11}{4}x_1$	$\frac{60}{11}$	

Nuevas Variables base

$x_3 = x_4 = 0$

3er etapa

$$\begin{aligned} x_2 + \tfrac{1}{3}x_3 + \tfrac{1}{3}x_4 &= 5 \\ x_1 - \tfrac{1}{3}x_3 + \tfrac{4}{3}x_4 &= 4 \\ \tfrac{2}{3}x_3 - \tfrac{11}{3}x_4 + x_5 &= 4 \\ -\tfrac{2}{3}x_3 - \tfrac{4}{3}x_4 + 28 &= M \end{aligned}$$

$$\left[\begin{array}{ccccc|c} 0 & 1 & \frac{1}{3} & -\frac{1}{3} & 0 & 5 \\ 1 & 0 & -\frac{1}{3} & \frac{4}{3} & 0 & 4 \\ 0 & 0 & \frac{2}{3} & -\frac{11}{3} & 1 & 4 \\ \hline 0 & 0 & +\frac{2}{3} & +\frac{4}{3} & 0 & +28 \end{array}\right]$$

Solución factible

$x_1 = 4,\ x_2 = 5$ y $x_5 = 4$

Tabla 6.5.2 Solución del ejemplo 6.5.2 por el método Simplex (Continuación)

El máximo valor de x_2 puede ser 6, ya que si es mayor de 6, $x_3 \leq 0$ y se violaría la condición $x_i \geq 0$, - i=1,2, ...5. Gráficamente, al ir moviendo la recta - - $2x_1 + 4x_2$ que representa la función objetivo paralelamente así misma, a lo largo de la recta $x_1=0$, se llega al - punto A, otra esquina del polígono ABCD que define a la región de soluciones factibles.

La figura 6.5.13 muestra este traslado de la función objetivo $m=2x_1+4x_2$

Para $x_1 = 0$, de (6.5.24) se obtiene:

$$x_3 = 24 - 4x_2$$
$$x_4 = 9 - x_2 \qquad (6.5.25)$$
$$x_5 = 21 - x_2$$

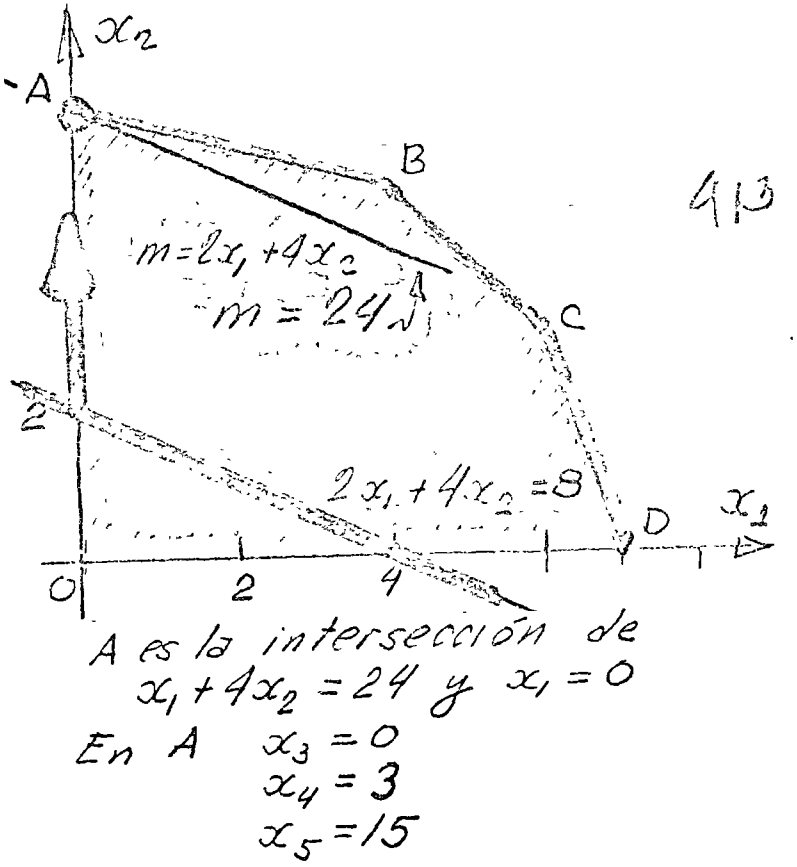


Fig. 6.5.13 Búsqueda del máximo de la función objetivo a lo largo de la recta $x_1 = 0$

000151

Del sistema de ecuaciones (6.5.25) para -

$x_1 = 0$ y $x_2 = 6$ se tiene:

$$x_3 = 0$$

$$x_4 = 3$$

$$x_5 = 15$$

Como x_3 era la holgura de la ecuación de -- restricción (6.5.8)

$$x_1 + 4x_2 + x_3 = 24 \qquad (6.5.8)$$

Se deduce que en el punto A el recurso limitado correspondiente a esa ecuación de restricción se ha empleado en su totalidad. En efecto el punto A se encuentra sobre la recta de ecuación

$$x_1 + 4x_2 = 24$$

y la ecuación

$$x_1 = 0.$$

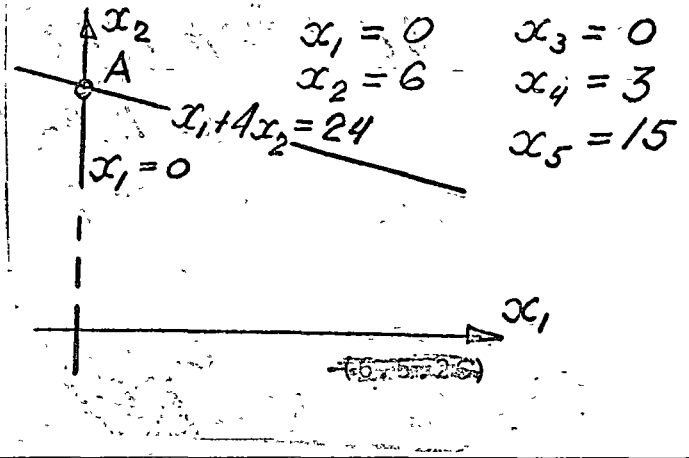
* En resumen en el punto A el valor de todas - las variables del problema son:

414

415

416

* Valor de las variables en el punto A:



000152

* Las nuevas variables no básicas, es decir -- las que son diferentes de cero son:

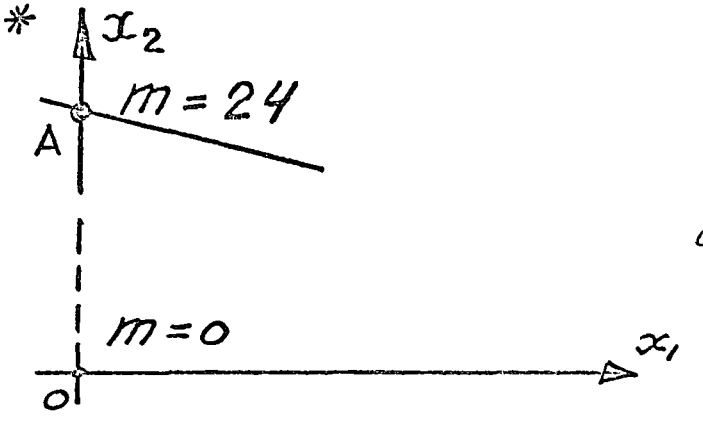
* Variables no básicas $\neq 0$ 417

x_2, x_4 y x_5

El valor de la función objetivo es:

$m = 2x_1 + 4x_2 = 24$ (6.5.26) 418

* que resulta mayor que el valor de esta función en el 1er. punto explorado, el origen, donde valía cero.

* 

419

* Cuando las variables básicas eran x_1 y x_2, o sea (en) el 1er. paso de la solución del problema, también llamada lera. iteración, el sistema de ecuaciones algebraicas que hubo que resolver eran:

* Con $x_1 = x_2 = 0$ (variables básicas el sistema de ecuaciones era: 420

$x_1 + 4x_2 + x_3 = 24$

$x_1 + x_2 + x_4 = 9$ (6.5.24)

$3x_1 + x_2 + x_5 = 21$

* En este sistema las variables no básicas -

* Las variables $x_3, x_4, x_5 \neq 0$ (no básicas) aparecen una en cada ecuación 421

x_3, x_4 y x_5 aparecían en una ecuación cada una, y esto facilitó su evaluación.

*Para proseguir con igual facilidad, se debe manipular algebráicamente a las ecuaciones (6.5.24) para que en cada una de ellas aparezca solamente una de las nuevas variables no básicas x_2, x_4 y x_5 de preferencia con coeficiente unitario. * De la 1er. ecuación del sistema (6.5.24).

*Manipule las ecuaciones para que las variables no básicas (x_2, x_4, x_5) aparezcan en una sola ecuación.

422

*1er ecuación

(1)-(3)

$$x_1 + x_3 + 4x_2 = 24$$

423

se tiene al dividir entre 4

$\div 4$

$$\frac{1}{4}x_1 + \frac{1}{4}x_3 + x_2 = 6 \qquad (6.5.26)$$

*Esta ecuación ya tiene una sola variable no básica x_2 con coeficiente unitario. *En la 2da. ecuación del sistema (6.5.24)

*Unica v.n.b. x_2

424

*2da. ecuación

$$x_1 + x_2 + x_4 = 9$$

425

*aparecen dos variables no básicas, x_2 y x_4. Como x_2 ya quedó en la ecuación anterior se debe dejar

*v.n.b x_2 y x_4

elimine x_2

426

en esta x_4. Restando de la ecuación

$$(1)\ x_1 + x_2 + x_4 = 9$$

la ecuación anterior

menos

427

$$(2)\ \frac{1}{4} x_1 + \frac{1}{4} x_3 + x_2 = 24$$

*se elimina la variable x_2. En efecto se tiene:

*se elimina x_2

$$(1)-(2)\ \frac{3}{4} x_1 - \frac{1}{4} x_3 + x_4 = 3$$

*Finalmente la última ecuación del sistema (6.5.24)

*Ultima ecuación

428

$$3 x_1 + x_2 + x_5 = 21$$

*Contiene las variables no básicas x_2 y x_5. Hay que eliminar x_2 para que solo quede una. Restando a esta ecuación la ecuación *(6.5.26) se elimina en efecto x_2

*V.n.b. x_2 y x_5

429

elimine x_2

$$3x_1 + x_2 + x_5 = 21$$

$$-\left(\frac{1}{4} x_1 + x_2 + \frac{1}{4} x_3 = 6\right)$$

430

Realizando esta operación se obtiene:

$$\frac{11}{4} x_1 + x_5 - \frac{1}{4} x_3 = 15$$

*El sistema de ecuaciones de restricción ha quedado de la forma deseada:

$$\frac{1}{4}x_1 + \frac{1}{4}x_3 + x_2 = 6 \qquad (6.5.26)$$

$$\frac{3}{4}x_1 - \frac{1}{4}x_3 + x_4 = 3 \qquad (6.5.27)$$

$$\frac{11}{4}x_1 - \frac{1}{4}x_3 + x_5 = 15 \qquad (6.5.28)$$

*En este sistema de ecuaciones en cada una de ellas aparece solamente una de las variables no básicas.

*La función objetivo hay que expresarla en función de las variables básicas. x_1 y x_3, es decir hay que eliminar x_2. En la etapa anterior esta función estaba expresada en función de x_1 y x_2:

despejando de la ecuación (6.5.26)

*a x_2 se tiene:

*Nuevas ecuaciones de restricción

431

*En cada ecuación aparece una sola v.n.b. (x_2, x_4, x_5)

432

*Expresar m en función de las v.b (x_1, x_3)

433

$m = f(x_1, x_2)$

$m = 2x_1 + 4x_2$

434

$$\frac{1}{4}x_1 + \frac{1}{4}x_3 + x_2 = 6 \qquad (6.5.26)$$

435

*despejando a x_2

$$x_2 = 6 - \frac{1}{4}x_1 - \frac{1}{4}x_3 - 6$$

000150

* Sustituyendo a x_2 en la función objetivo por este valor se tiene:

$$m = 2x_1 + 4\left(6 - \frac{1}{4}x_1 - \frac{1}{4}x_3\right) \qquad (6.5.29)$$

$$m = x_1 - x_3 + 24$$

La función objetivo ha quedado expresada en función de las variables básicas x_1 y x_3 y en valor para $x_1 = 0$ y $x_3 = 0$ es m=24

A continuación hay que determinar que pasa con la función objetivo si se aumenta x_1 ó x_3. Para incrementar m, y seguir satisfaciendo la condición de no negatividad de las variables debe -- mantenerse $x_3 = 0$ ya que dado el coeficiente negativo de x_3, si x_3 aumenta, m disminuye. (Debe) incremen-se ta x_1. * Del sistema de ecuaciones de restricción - para $x_3 = 0$, las variables no básicas en función - de la variable base x_2, quedan expresadas en la - siguiente forma:

$$x_2 = 6 - \frac{x_1}{4}$$

$$x_4 = 3 - \frac{3}{4}x_1 \qquad (6.5.30)$$

$$x_5 = 15 - \frac{11}{4}x_1$$

* Sustituyendo en m 436

* $m = f(v.b., x_1 \text{ y } x_3)$ 437
para $x_1 = x_3 = 0 \Rightarrow m = 24$

* $m = x_1 - x_3 + 24$ 438
si $x_3 \uparrow$ $m \downarrow$

439

* Para $x_3 = 0$ las ecuaciones de restricción son:

440

* Deben analizarse las ecuaciones (6.5.30) para determinar cual es el máximo valor de x_1, para el cual todas las variables no básicas x_2, x_4 y x_5 sean mayores o iguales a cero. Se tiene

* Máximo valor de x_1 sin violar condiciones de no negatividad 441

$$x_2 = 6 - \frac{x_1}{4} \qquad x_{1_{max}} = 24 \qquad x_2 = 0$$

$$x_4 = 3 - \frac{3}{4}x_1 \qquad x_{1_{max}} = 4 \qquad x_4 = 0$$

$$x_5 = 15 - \frac{11}{4}x_1 \qquad x_{1_{max}} = \frac{60}{11} \qquad x_5 = 0$$

* Se Vemos que el máximo valor posible de la variable x_1, sin que ninguna de las variables x_2, x_4 y x_5 se vuelvan negativas es 4, para lo cual $x_4 = 0$.

* Máximo valor de posible de $x_1 = 4 \Rightarrow x_4 = 0$ 442

* Se finaliza este paso se tiene que $x_3 = x_4 = 0$. Estas variables se toman como base para el siguiente paso.

* Nuevas V.b. $x_3 = x_4 = 0$

* Para $x_3 = x_4 = 0$, y $x_1 = 4$ el valor del resto de las variables es $x_2 = 5$ y $x_5 = 4$. Estos valores (se) obtienen del sistema de restricciones.

* Del sistema de restricciones: Si $x_3 = x_4 = 0$ y $x_1 = 4 \Rightarrow x_2 = 5$, $x_5 = 4$ 443

* El siguiente conjunto de valores de las variables

* Nueva solución factible 445

$$x_1 = 4, x_2 = 5, x_3 = 0, x_4 = 0 \text{ y } x_5 = 4$$

constituye una nueva solución factible.

* Antes de continuar resulta ilustrativo interpretar gráficamente este segundo paso de solución. En este paso de solución se incrementa el valor de la variable --

* Interpretación gráfica de la 2da iteración. x_1 de 0 a 4 446

.1.

básica x_1 de 0 a 4 manteniendo a la otra variable básica $x_3 = 0$. Como x_3 es la variable de holgura de la 1er. ecuación de restricción $x_1 + 4x_2 \leq 24$ esta búsqueda de un mayor valor en la función objetivo se realiza a lo largo de la frontera AB de la zona de soluciones factibles tal como se ilustra en la figura 6.5.14.

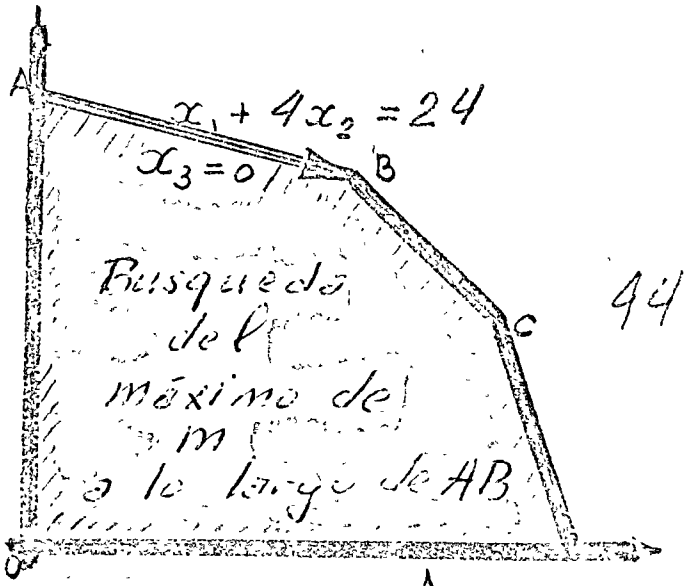


Fig. 6.5.14 Búsqueda del máximo de la función objetivo a lo largo de la recta AB.

Para continuar debe volverse a manipular el sistema de ecuaciones (6.5.25) - (6.5.28), para dejar en cada ecuación una sola de las variables no básicas x_1, x_2

y x_5. Realizando operaciones algebraicas elementales sobre ese sistema similares a las descritas previamente se obtiene:

$$\begin{aligned} x_2 + \tfrac{1}{3}x_3 - \tfrac{1}{3}x_4 &= 5 \\ x_1 - \tfrac{1}{3}x_3 + \tfrac{4}{3}x_4 &= 4 \\ + \tfrac{2}{3}x_3 - \tfrac{11}{3}x_4 + x_5 &= 4 \end{aligned} \qquad (6.5.30)$$

y volviendo a expresar la función objetivo en relación a las nuevas variables base x_3 y x_4 se tiene:

$$m = 28 - \tfrac{2}{3}x_3 - \tfrac{4}{3}x_4 \qquad (6.5.32)$$

* Como en el último paso x_3 y x_4 eran nulas, la única forma de alterarlas, y satisfaciendo simultáneamente la condición de no negatividad de las variables es incrementándolas, pero esto disminuiría el valor de m. Por lo tanto la solución factible que a su vez es óptima es precisamente la solución obtenida en el paso anterior a saber:

$$x_1 = 4,\ x_2 = 5,\ x_3 = 0,\ x_4 = 0 \text{ y } x_5 = 0$$

* Recuérdese el plantamiento del problema:

Maximizar la función objetivo.

$$m = 2x_1 + 4x_2 \qquad (6.5.7)$$

* sujeto a las restricciones

(mecánicas) $x_1 + 4x_2 + x_3 = 24$ (6.5.8)

* No puede x_3 y x_4 porque $m \neq$ solución factible del paso anterior es óptima

Foto

* Problema

./.

(andenes) $x_1 + x_2 + x_4 = 29$ (6.5.9)

(cargadores) $3x_1 + x_2 + x_5 = 21$ (6.5.10)

donde se recordará que la primer restricción la imponía la disponibilidad de mecánicos, la segunda estaba relacionada con la existencia de andenes y la tercera con la disponibilidad de cargadores.

* Al operar 4 camionetas chicas y 5 grandes, como indica la solución del problema, ($x_1 = 4, x_2 = 5$), la primer variable de holgura es nula ($x_3 = 0$), la segunda también es nula ($x_4 = 0$), y la tercera vale 4 ($x_5 = 4$). Este conjunto de valores de la variable de holgura significa que el primer recurso (mecánicos) se aproveche en su totalidad al igual que el segundo (andenes). - - Mientras que del tercer recurso se le emplea la cantidad disponible menos la holgura, es decir

$$21 - x_5 = 21 - 4 = 17$$

* Para resolver un problema de programación lineal empleando el método simplex es necesario realizar repe-- titivamente diversas operaciones, como se acaba de ilustrar. Es posible sistematizar el método solución expuesto empleando la notación matricial.

Se empieza por formar una tabla o matriz cuyas -

* $x_1 = 4 \equiv$ operar 4 camionetas chicas
$x_2 = 5 \equiv$ operar 5 camionetas grandes
$x_3 = 0 \equiv$ se emplean todos los mecánicos
$x_4 = 0 \equiv$ se emplean todos los andenes
$x_5 = 4 \equiv$ se emplean 21 - 4 mecánicos

* Sistematización del método empleando matrices

columnas menos ~~viene~~ la última tienen por valor los coeficientes de las variables en las ecuaciones de -- restricción y en la función objetivo. En esta última ecuación debe cambiarse el signo de los coefi-- cientes. La última columna tiene por valor los recursos disponibles y un cero en la última posición. Los elementos del último renglón de esta tabla, -- exceptuando el último , se llaman -- los indicadores del problema. Como se señala a continuación, si después de realizar las operaciones que se indican posteriormente, todos los indicadores son positivos, la búsqueda del óptimo a terminado. Con objeto de familiarizar al lector con el método se presentan las ecuaciones en forma explícita y en notación matricial, tal como aparecen a continuación:

1er Etapa

$$
\begin{aligned}
x_1 + 4x_2 + x_3 &= 24 \\
x_1 + x_2 + x_4 &= 9 \\
3x_1 + x_2 + x_5 &= 21 \\
2x_1 + 4x_2 &= m
\end{aligned}
$$

x_1	x_2	x_3	x_4	x_5	b	
1	4	1	0	0	24	6 ⟸
1	1	0	1	0	9	9
3	1	0	0	1	21	21
-2	-4	0	0	0	0	
*	*					

(6.5.32)

indicadores

* variables base

1ª solución factible: $x_1 = x_2 = 0$ (v. 6)

$x_3 = 24$, $x_4 = 9$, $x_5 = 21$ (v. n. 6)

El problema se inicia buscando una solución factible. En este caso puede ser $x_1 = x_2 = 0$, $x_3=24$ $x_4 = 9$ y $x_5=21$.

Posteriormente se señala como puede sistematizarse la búsqueda de la 1er. solución factible.

* Posteriormente debe seleccionarse la columna con el término más negativo en el último renglón ó sea el correspondiente a la función objetivo. Con objeto de determinar el incremento de cual variable hace crecer más rápidamente a la función objetivo. -- En este primer paso la segunda columna, correspon-- diente a x_2 tiene esta propiedad, ó sea más negativo el último renglón.* A continuación se dividen los elementos correspondientes a la disponibilidad de recursos, es decir los elementos de la última columan, -- exceptuando el último, entre los correspondientes -- elementos de la columna seleccionada anteriormente - en este caso la segunda. El valor de estos cocientes se anota en una última columna y se selecciona el --- renglón con el valor menor de esta columna, en este caso el primero. En este ejemplo estos valores fueron 6,9 y 21. Este problema se inicializó con $x_1 = x_2 = 0$.

* Busque columna con último elemento más negativo

$$\left[\begin{array}{ccccc|c} 1 & (4) & 1 & 0 & 0 & 24 \\ 1 & 1 & 0 & 1 & 0 & 9 \\ 3 & 1 & 0 & 0 & 1 & 21 \\ \hline -2 & -4 & 0 & 0 & 0 & 0 \end{array}\right]$$

461

* Divida última columna entre columna seleccionada anteriormente

(2)	(6)	(6)/(2)
4	24	6
1	9	9
1	21	21
-4		

462

Los incrementos en x_2 resultan aumentar más la función objetivo que los de x_1. La columan adicional indica que el máximo valor que puede darse a x_2 es de 6, sin hacer negativa alguna de las variables -- x_3, x_4 ó x_5.

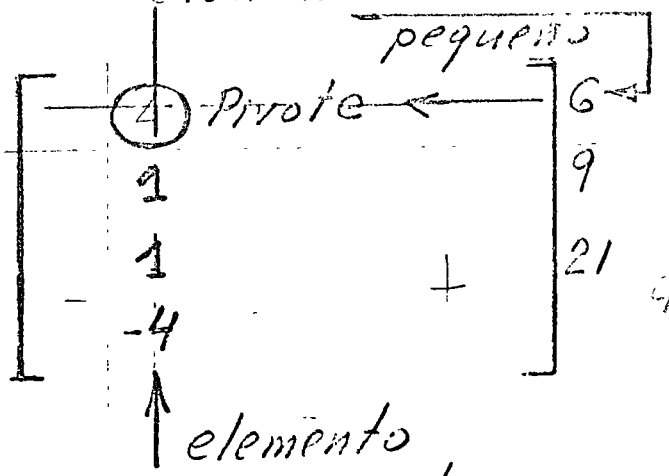
* Se ha encontrado hasta el momento que el primer renglón tiene el elemento más pequeño en la última columna y la segunda columna el más negativo en el último renglón. La intersección de este renglón (el primero) y esta columna, la segunda, definen el elemento llamado pivote, en este caso 4.

* Posteriormente se divide la el renglón del pivote, en este caso el primero entre el pivote como se ilustra a continuación:

* x_2 incrementa más rápidamente a un que x_1 463

[6] ⇐ 9 máximo valor de x_2 que no viola $x_i \geq 0 \forall i$ 464
21

* elemento más pequeño



elemento más negativo

465

* División del renglón del pivote entre el pivote 466

Formulación explícita

Formulación matricial

	x_1	x_2	x_3	x_4	x_5	
$x_1 + 4x_2 + x_3 = 24$	1	(4)	1	0	0	24
$\frac{1}{4}x_1 + x_2 + \frac{1}{4}x_3 = 6$	$\frac{1}{4}$	(1)	$\frac{1}{4}$	0	0	6

Después se emplea esta última ecuación para-- eliminar la variable x_2 de las ecuaciones restantes - del sistema. Como en la 2da. y 3er.ecuación x_2 tiene uno por coeficiente basta restar el renglón ó sea la ecuación del pivote de cada una de esas ecuaciones.

	Formulación explícita	Formulación matricial					
		x_1	x_2	x_3	x_4	x_5	
1da ecuación	$x_1 + x_2 + x_4 = 9$	1	1	0	1	0	9
ación del ote normalizada	$\frac{1}{4}x_1 + x_2 + \frac{1}{4}x_3 = 6$	$\frac{1}{4}$	1	$\frac{1}{4}$	0	0	6
Resta	$\frac{3}{4}x_1 - \frac{1}{4}x_3 + x_4 = 3$	$\frac{3}{4}$	0	$-\frac{1}{4}$	1	0	3
ecuación	$3x_1 + x_2 + x_5 = 21$	3	1	0	0	1	21
ación del ote normalizada	$\frac{1}{4}x_1 + x_2 + \frac{1}{4}x_3 = 6$	$\frac{1}{4}$	1	$\frac{1}{4}$	0	0	6
Resta	$\frac{11}{4}x_1 - \frac{1}{4}x_3 + x_5 = 15$	$\frac{11}{4}$	0	$-\frac{1}{4}$	0	1	15

Para eliminar a x_2 de la función objetivo, es necesario multiplicar la ecuación del pivote -- por - 4 y restarla de la función objetivo, ya que - 4 es el coeficiente de x_2 en la función objetivo, tal como se ilustra:

	Formulación explícita			Formulación matricial					
				x_1	x_2	x_3	x_4	x_5	
Función objetivo	$-2x_1 - 4x_2$	$= 0$		-2	-4	0	0	0	0
Renglón del v. norm. x -4	$-x_1 - 4x_2 - x_3$	$= -24$		-1	-4	-1	0	0	-24
Resta	$-x_1 \quad + x_3$	$= 24$		-1	0	1	0	0	24

Después de realizadas las funciones anteriores x_2, x_4 y x_5, las variables no básicas, quedan multiplicadas por 1, tal como muestra el siguiente cuadro:

./.

Formulación explícita

$$\frac{1}{4}x_1 + x_2 + \frac{1}{4}x_3 = 6$$

$$\frac{3}{4}x_1 - \frac{1}{4}x_3 + x_4 = 3$$

$$\frac{11}{4}x_1 - \frac{1}{4}x_3 + x_5 = 15$$

$$-x_1 + x_3 = 24$$

Formulación matricial

	x_1	x_2	x_3	x_4	x_5		
	$\frac{1}{4}$	1	$\frac{1}{4}$	0	0	6	24
	$\frac{3}{4}$	0	$-\frac{1}{4}$	1	0	3	4 ⇐
	$\frac{11}{4}$	0	$-\frac{1}{4}$	0	1	15	$\frac{60}{11}$
	-1	0	1	0	0	24	
	*		*				

470

En este cuadro la columna con el elemento más negativo es la 1era. y el cociente de la primer columna entre la de las restricciones es 24, 4, $\frac{60}{11}$ ó sea - los elementos de la columna auxiliar situada fuera de las llaves de la matriz. Como el elemento más pequeño es 4, el del segundo renglón, el pivote es el aumento de la 1er. columna y del segundo renglón y se va a -- emplear para eliminar x_1 de las ecuaciones restantes. Se empieza dividiendo el renglón del pivote entre el pivote, tal como se ilustra, para obtener la ecuación del pivote normalizada. (e.p.n)

471

* Ecuación del pivote normalizada: e.p.n.

Formulación explícita

Ecs. del pivote: $\frac{3}{4}x_1 - \frac{1}{4}x_3 + x_4 = 3$

Ec. norm.: $x_1 - \frac{1}{3}x_3 + \frac{4}{3}x_4 = 4$

Formulación matricial

x_1	x_2	x_3	x_4	x_5	
$\frac{3}{4}$	0	$-\frac{1}{4}$	1	0	3
1	0	$-\frac{1}{3}$	$\frac{4}{3}$	0	4

Para eliminar x_1 de la primer ecuación es necesario restarle la ecuación del pivote multiplicada por $\frac{1}{4}$ -- que es el coeficiente de x_1 en la 1er. ecs.

Formulación explícita

1ª Ecs. $\frac{1}{4}x_1 + x_2 + \frac{1}{4}x_3 = 6$

p.n × $\frac{1}{4}$: $\frac{1}{4}x_1 - \frac{1}{12}x_3 + \frac{1}{3}x_4 = 1$

Resta: $0 \quad x_2 + \frac{1}{3}x_3 - \frac{1}{3}x_4 = 5$

Formulación matricial

x_1	x_2	x_3	x_4	x_5	
$\frac{1}{4}$	1	$\frac{1}{4}$	0	0	6
$\frac{1}{4}$	0	$-\frac{1}{12}$	$\frac{1}{3}$	0	1
0	1	$\frac{1}{3}$	$-\frac{1}{3}$		5

Para eliminar x_1 de la 3er. ecuación hay que restarle la del pivote normalizada multiplicada por $\frac{11}{4}$.

Formulación explícita

3er Ecs.: $\frac{11}{4}x_1 - \frac{1}{4}x_3 + x_5 = 15$

p.n × $\frac{11}{4}$: $\frac{11}{4}x_1 - \frac{11}{12}x_3 + \frac{11}{3}x_4 = 11$

Resta: $\frac{2}{3}x_3 - \frac{11}{3}x_4 + x_5 = 4$

Formulación matricial

x_1	x_2	x_3	x_4	x_5	
$\frac{11}{4}$	0	$-\frac{1}{4}$	0	1	15
$\frac{11}{4}$	0	$-\frac{11}{12}$	$\frac{11}{3}$	0	11
0	0	$\frac{2}{3}$	$-\frac{11}{3}$	1	4

Y finalmente para eliminar x_1 de la función objetivo debe restársele la normalizada del pivote multiplicada por - 1, coeficiente de x_1 en la función objetivo, tal como se muestra:

Formulación explícita

Func. obj $-x_1 + x_3 = 24$

$-x_1 + \frac{1}{3}x_3 - \frac{4}{3}x_4 = -4$

Fonx-1

Resta $\frac{2}{3}x_3 + \frac{4}{3}x_4 = 28$

Formulación matricial

x_1	x_2	x_3	x_4	x_5	
-1	0	1	0	0	+24
-1	0	$\frac{1}{3}$	$-\frac{4}{3}$	0	-4
0	0	$\frac{2}{3}$	$\frac{4}{3}$	0	28

475

Una vez realizadas estas operaciones el cuadro queda como se muestra:

3er. Etapa

Formulación explícita

$x_2 + \frac{1}{3}x_3 - \frac{1}{3}x_4 = 5$

$x_1 - \frac{1}{3}x_3 + \frac{4}{3}x_4 = 4$

$\frac{2}{3}x_3 - \frac{11}{3}x_4 + x_5 = 4$

$\frac{2}{3}x_3 + \frac{4}{3}x_4 = 28$

Formulación matricial

x_1	x_2	x_3	x_4	x_5	
0	1	$\frac{1}{3}$	$-\frac{1}{3}$	0	5
1	0	$-\frac{1}{3}$	$\frac{4}{3}$	0	4
0	0	$\frac{2}{3}$	$-\frac{11}{3}$	1	4
0	0	$\frac{2}{3}$	$\frac{4}{3}$	0	28

indicadores

476

óptimo

* Antes de continuar es necesario revisar el signo de los elementos del último renglón, exceptuando el último ó sea de los indicadores. Cuando todos son positivos se ha encontrado el óptimo.

* Revise indicadores, si todos ≥ 0 se ha encontrado el óptimo ≡ último elemento de la tabla

477

El valor del óptimo está dado por el último del último renglón. 28

último elemento

Como en esta etapa ya todos los indicadores son positivos, la búsqueda del óptimo ha terminado. El valor óptimo es 28.* Las variables con coeficiente diferente de cero en el último renglón valen cero, en este caso $x_3 = x_4 = 0$.

* último renglón

0 0 $\frac{2}{3}$ $\frac{4}{3}$ 0 28

↑ ↑

$x_3 = x_4 = 0$

472

Del último sistema de ecuaciones:

$$x_2 + \frac{1}{3}x_3 - \frac{1}{3}x_4 = 5$$
$$x_1 - \frac{1}{3}x_3 + \frac{4}{3}x_4 = 4$$
$$\frac{2}{3}x_3 - \frac{11}{3}x_4 + x_5 = 4$$

para

se obtiene:

$x_3 = x_4 = 0$

$x_2 = 5,\ x_1 = 4,\ x_5 = 4$

479

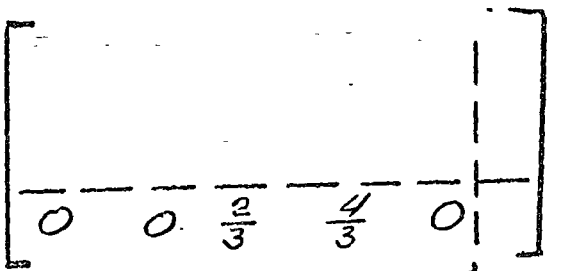
480

* Es posible obtener el valor de los niveles de actividad en el punto óptimo a partir de la última tabla del método simplex,

* Obtención de los niveles óptimos de actividad a partir de la última tabla simplex.

* La llamada "última tabla" del método simplex dice indicadores únicamente positivos.

* última tabla: 637 000171



481

Para obtener la nivel óptimo de actividad de la última tabla basta numerar los renglones de acuerdo con la posición donde se encuentra una columna unitaria, es decir una columna con un solo uno y el resto ceros.

Para aclarar este paso nos referimos a la fig. 6.5.15 donde aparece la tabla terminal del ejemplo. El 1er. renglón tiene su 1er.columna unitaria en la segunda posición, por eso se le designa con 2, y el 2do. renglón tiene la 1er. columna unitaria en la primer posición. Se le designa con 1. Se continúa hasta terminar con todos los renglones menos el último. En el punto óptimo las variables diferentes de cero tienen por índice el número -

con el que se han designado los renglones y su valor está dado en la última columna. En este caso $x_2 = 5$, $x_1 = 4$ y $x_5 = 4$.

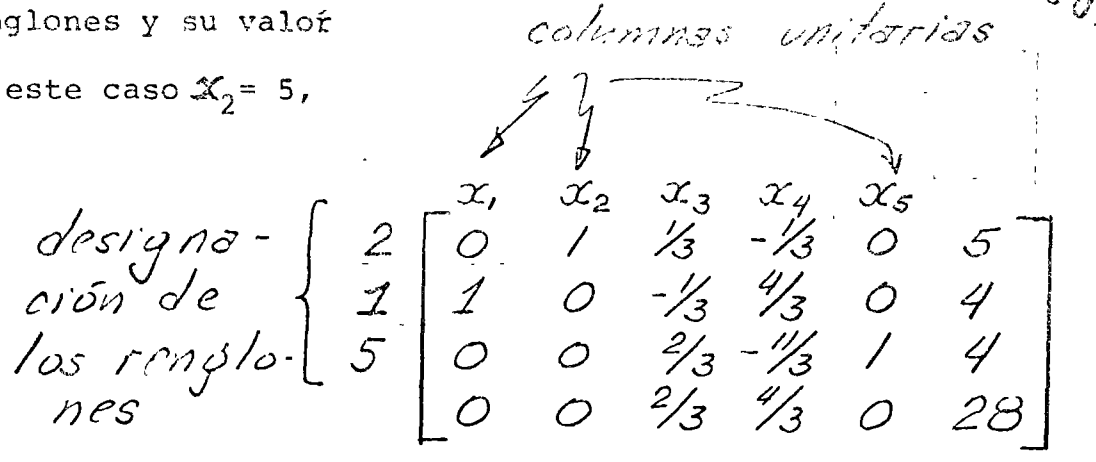


Fig. 6.5.15 Tabla terminal del problema

En la tabla 6.5.2 se resumen los diferentes pasos que se siguen en la solución de un problema de programación lineal mediante el método simplex. En las matrices de esta tabla la columna y el renglón marcados con una flecha definen la posición del pivote y las columnas marcadas con un asterisco (*) corresponden a las variables base, es decir que se han tomado como nulas.

Solución analítica del problema de programación lineal

Problema

Función objetivo:

$$m = 2x_1 + 4x_2 \quad (max)$$

Restricciones

$$x_1 + 4x_2 \leq 24 \quad (a)$$
$$x_1 + x_2 \leq 9 \quad (b)$$
$$3x_1 + x_2 \leq 21 \quad (c)$$
$$x_1 \geq 0 \quad (d)$$
$$x_2 \geq 0 \quad (e)$$

483

Formulación explícito

1ª Etapa

$$x_1 + 4x_2 + x_3 = 24$$
$$x_1 + x_2 + x_4 = 9$$
$$3x_1 + x_2 + x_5 = 21$$
$$+2x_1 + 4x_2 + 0 = m$$

Formulación matricial

	x_1	x_2	x_3	x_4	x_5		
	1	(4)	1	0	0	24	6 ←
	1	1	0	1	0	9	9
	3	1	0	0	1	21	21
	-2	-4				0	
	*	*					

⇑

Variables base

$x_1 = x_2 = 0$ *

Solución factible

$x_3 = 24 \quad x_4 = 9 \quad x_5 = 21$

Incremento unitario en $x_1 \rightarrow$ Incremento en $m = 2$

Incremento unitario en $x_2 \rightarrow$ Incremento en $m = 4$ ↑↑

	$x_1 = 0$ x_2 max
$x_3 = 24 - 4x_2 = 0$	6
$x_4 = 9 - x_2 = 0$	9
$x_5 = 21 - x_2 = 0$	21

con $x_2 = 6 \Rightarrow x_3 = 0$

Nuevas Variables base

$x_1 = 0 \quad x_3 = 0$ *

2da etapa.

$$\frac{1}{4}x_1 + x_2 + \frac{1}{4}x_3 = 6$$
$$\frac{3}{4}x_1 - \frac{1}{4}x_3 + x_4 = 3$$
$$\frac{11}{4}x_4 - \frac{1}{4}x_3 + x_5 = 15$$
$$x_1 - x_3 + 24 = M$$

$$\left[\begin{array}{ccccc|c} \frac{1}{4} & 1 & \frac{1}{4} & 0 & 0 & 6 \\ \frac{3}{4} & 0 & -\frac{1}{4} & 1 & 0 & 3 \\ \frac{11}{4} & 0 & -\frac{1}{4} & 0 & 1 & 15 \\ -1 & 0 & 1 & 0 & 0 & +24 \end{array}\right] \begin{array}{l} 24 \\ 4 \Leftarrow \\ \frac{60}{11} \end{array}$$

* *

↑↑

* Antes de continuar es necesario indicar como se obtiene la 1er. solución factible en el problema de programación lineal.

* Obtención de la 1er solución factible.

486

Recuérdese que el problema de programación - lineal tiene n incognitas, los niveles de actividad y existen m ecuaciones de restricción. Si todas las ecuaciones de restricción son desigualdades se introducen m variables de holgura, y en el primer paso de solución, igualando a cero las variables de holgura toman determinados valores, que forman una 1er. solución factible. En este caso se encuentra el problema del ejemplo anterior, cuyas restricciones eran:

(del problema (niveles de actividad) las variables en)

487

$$x_1 + 4x_2 \leq 24 \quad \Rightarrow \quad x_1 + 4x_2 + x_3 = 24$$

$$x_1 + x_2 \leq 9 \quad \Rightarrow \quad x_1 + x_2 + x_4 = 9$$

$$3x_1 + x_2 \leq 21 \quad \Rightarrow \quad 3x_1 + x_2 + x_5 = 21$$

Niveles de actividad x_1, x_2

Variables de holgura x_3, x_4, x_5

1er. Solución Factible

Niveles de actividad $x_1 = x_2 = 0$

Variables de holgura $x_3 = 24$; $x_4 = 9$ y $x_5 = 21$

488

En algunos casos algunas restricciones son mayores que cero o igualdades. En este caso habrá menos

* Con restricciones de

de m variables de holgura y no se puede formar la - igualdad
1er. solución factible igualando los niveles de actividad a cero, como se ilustra a continuación.

Max: $m = 2x_1 + 4x_2 + x_3$

Sujeto a las siguientes restricciones:

$$x_1 + 2x_2 + x_3 \leq 4$$
$$2x_1 + 4x_2 + x_3 = 8$$
$$4x_1 + 2x_2 - x_3 \geq 6$$

* La introducción de dos variables de holgura, ya que solo hay dos desigualdades convierte a las -- ecuaciones de restricción en:

* Con dos variables de holgura

$$x_1 + 2x_2 + x_3 + x_4 = 4$$
$$2x_1 + 4x_2 + x_3 = 8$$
$$4x_1 + 2x_2 - x_3 - x_5 = 6$$

Si se da a los niveles de actividad x_1, x_2 - x_3 el valor cero, como en el caso anterior, para inicializar el problema, se viola la segunda restricción ya que

$$2x_1 + 4x_2 + x_3 \Big| \neq 8 \quad \Big| x_1 = x_2 = x_3 = 0$$

De manera que no es posible obtener en esta forma la 1er. solución factible para iniciar la solución - del problema de programación lineal si se presenta un - problema de este tipo es necesario incluir * variables artificiales

* Variables artificiales

tificiales en el problema. Una por cada ecuación de restricción que sea una igualdad y una desigualdad del tipo "mayor o igual que cero". En el ejemplo es necesario introducir las variables artificiales - - x_6 y x_7 ya que hay una igualdad y una desigualdad - del tipo "mayor o igual que cero" entre las restric ciones. El sistema de ecuaciones de restricción, -- después de introducir estas variables queda:

$$x_1 + 2x_2 + x_3 + x_4 = 4$$
$$2x_1 + 4x_2 + x_3 + x_6 = 8$$
$$4x_1 + 2x_2 - x_3 - x_5 + x_7 = 6$$

En este caso asignando a las variables estruc turales y a una de las de holgura el valor cero se -- puede obtener la primer solución factible.

* En efecto con $x_1 = x_2 = x_3 = x_5 = 0$, el resto de las variables asume el valor de $x_4 = 4$, $x_6 = 8$ y $x_7 = 12$.

* Las variables artificiales no deben aparecer en la solución final en la función objetivo. Para asegurarse de que esto no suceda, se deben incluir en la función objetivo con grandes coeficientes negativos - en problemas de maximización. Estos grandes coeficientes negativos aseguran que las variables artificiales

* Con $x_1 = x_2 = x_3 = x_5 = 0$ $x_4 = 4$, $x_6 = 8$ y $x_7 = 12$

* Introducir en problemas de maximización las variables artificiales con grandes coeficientes negativos

deben ser nulas para maximizar la función objetivo.

Antes de terminar con esta sección para estudiar el problema dual en la siguiente, es necesario enunciar un importante teorema de programación lineal y explicar porque este método es un método de gradiente.

* Teorema 497

498

Puede demostrarse * que en un problema de programación lineal con las restricciones definiendo una zona convexa, el punto óptimo (ya sea máximo ó mínimo de la función objetivo) se encuentra siempre en la frontera de la zona convexa definida por las restricciones.

* Busqueda a lo largo de la frontera 499

Debido a este teorema la búsqueda del óptimo se realiza a lo largo de la frontera de la zona definida por las restricciones, como se ilustró en la solución gráfica y analítica del problema del ejemplo 6.5.2, ~~haciendo referencia~~ En la figura 6.5.9 referente a este problema, se recuerda que el método simlex (P) se empezó por calcular el valor de la función objetivo en 0, después en A y finalmente en B. No fué sin embargo necesario evaluarla en todos los vértices del polígono OABCD. Faltaron los puntos C y D. El método permite ir buscando valores siempre

* Noble, B: Applied Linear Algebra, Cap. 6, Prentice Hall. Englewood Cliffs N.J. 1969. * ref. 10 cap. 6

crecientes (en un problema de maximización) de la función objetivo en los vértices del polígono.* Esta búsqueda se realiza siempre a lo largo de aquella arista donde el valor de la función objetivo crece (o decrece) con mayor rapidez. Por esta razón se trata de un método de gradiente. El método permite descubrir cuando se ha encontrado el valor optimo, sin necesidad de tener que evaluar en general la función objetivo en todos los vértices del polígono y tener finalmente que buscar el valor optimo de la función objetivo entre estos valores.

Un método de fuerza ~~bruta~~ bruta para encontrar el optimo consistiría en evaluar la función objetivo en todos los vértices y después ~~hacer~~ buscar el máximo ó mínimo de esta entre todos estos valores. El método simplex no solamente reduce el número de vértices donde hay que calcular la función objetivo, sino al ir de paso en paso incrementando (ó decrementando) el valor de la función objetivo hace inecesaria la búsqueda final del optimo. En problemas con gran número de variables, el no tener que explorar todos los vértices y de no tener que alma

* Busqueda en dirección de máxima variación de la función objetivo

500

./.

cenar para una búsqueda final del óptimo el valor de las coordenadas de los vertices y de la función objetivo, ahorra mucho tiempo y requerimiento de memoria al procesarse digitalmente estos problemas. Desde luego que esta ventaja computacional tiene como precio las restricciones que impone al modelo matemático el problema, las de linealidad en sus ecuaciones y de convexidad y de la zona de soluciones factibles. Afortunadamente existen múltiples problemas, de gran interés para el analisista de sistemas, en los que puede plantearse un modelo matemático con las restricciones anteriores.

Foto 501

6.5.5 Problema dual. 502

* Se indicó en la sección 6.5.2 que el problema de programación lineal puede plantearse en forma matricial de la siguiente manera:

* Formulación matricial 503

max: $m = \underline{c}^T \underline{x}$ (6.5.21)

* Sujeto a las restricciones

* restricciones

$$\underline{A}\,\underline{x} \leq \underline{b} \quad (6.5.22)$$

$$\underline{x} \geq \underline{0} \quad (6.5.23)$$

donde $\underline{x}$ el vector de niveles de actividad, $\underline{b}$ el de restricciones y $\underline{c}$ el de costos. La matriz $\underline{A}$ tiene por elementos los coeficientes estructurales del problema.

La ilustración del problema dual puede realizarse con el ejemplo 6.5.2, sin embargo no se le empleará, con objeto de introducir otro tipo de problema.

Ejemplo 6.5.4

* En un taller se cuenta con tres máquinas A, B, y C. Se emplean para fabricar dos productos 1 y 2. La tabla 6.5.3 muestra las horas de maquinado que

* Las máquinas A, B y C fabrican los productos 1 y 2

requiere cada producto, las horas disponibles en cada tipo de máquina y la ganancia que se obtiene en la venta de cada producto.

Tipo de máquina	Producto 1	Producto 2	Horas disponibles
A	1	2	200
B	1	1	125
C	1	0	100
Beneficio	2	3	

Tabla 6.5.3 Datos para el ejemplo 6.5.4

* Se trata de planear la producción de manera que se obtenga la máxima ganancia posible. Plantee el modelo matemático para este problema.

Solución

Sean x_1 y x_2 las cantidades del producto 1 y 2 fabricados.

507

508

* Planeación de la producción para maximización de la ganancia

509

* cantidad fabricada x_1 y x_2

510

* El objetivo será por lo tanto maximizar la ganancia es decir

* maximizar la ganancia

$$\max: m = 2x_1 + 3x_2$$

* Para producir x_1 unidades del producto 1 y x_2 del 2 se requieren las siguientes horas de la máquina A que están restringidas a 200:

* Carga de la máquina A

$$x_1 + 2x_2 \leq 200$$

* En forma similar las restricciones que provienen de la máquina B y C son:

* Carga de las máquinas B y C

$$x_1 + x_2 \leq 125$$
$$x_1 \leq 100$$

Desde luego que no tiene significado físico producir unidades negativas por lo tanto :

$$x_1, x_2 \geq 0$$

* El planteamiento matricial del problema es:

* Maximizar 514

$$\max: \quad m = \begin{bmatrix} 2 \\ 3 \end{bmatrix}^T \longleftrightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

* Sujeto a las restricciones: * Restricciones 515

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 0 \end{bmatrix} \longleftrightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 200 \\ 125 \\ 100 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq 0$$

* A continuación se introduce el llamado problema dual o dual simplemente, del problema de programación lineal. * Problema dual 516

Si m son el número de restricciones y n el número de variables del problema para definir el dual es necesario introducir un vector w de m componentes

* m restricciones 517
n variables
Introduzca vector $\underline{w}$

$$\underline{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

cuya interpretación se dará posteciormente. El dual del problema de programación lineal es otro problema cuya formulación matricial, comparada con la del último aparece en el siguiente cuadro:

Problema original o primo	Problema dual
max: $m = \underline{c}^T \underline{x}$	min: $n = \underline{b}^T \underline{w}$
sujeto a las restricciones:	
$\underline{A}\ \underline{x} \leq \underline{b}$	$\underline{A}^T\ \underline{w} \geq c$
$\underline{x} \geq 0$	$\underline{w} \geq 0$

A continuación se ilustra el planteamiento del problema dual.

Ejemplo 6.5.5

Plantee el problema dual del ejemplo 6.5.4

Solución:

La solución aparece en el siguiente cuadro:

Problema original ó primo		Problema dual
$\max: m = \begin{bmatrix} 2 \\ 3 \end{bmatrix}^T \Longleftrightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$		$\min: n = \begin{bmatrix} 200 \\ 125 \\ 100 \end{bmatrix}^T \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$

Sujeto a las restricciones:

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \Longleftrightarrow \begin{bmatrix} 200 \\ 125 \\ 00 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 1 \\ & & \\ 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \geq \begin{bmatrix} 2 \\ \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq \underline{0} \qquad \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \geq \underline{0}$$

El teorema más importante de la programación lineal establece la siguiente relación entre el problema original ó primo y el problema dual:

Teorema: La función objetivo m de un problema de maximización de programación lineal asume su valor máximo si y solamente si la función objetivo n del problema dual correspondiente alcanza un mínimo y en este caso.

$$\max \; m = \min \; n.$$

Además si P y Q son soluciones factibles tales que $m(P) = n(Q)$, entonces las soluciones P y Q son las óptimas del problema primo y del problema dual respectivamente.

La demostración de este teorema aparece en la mayoria de los textos de programación lineal (ref.4).

* Una primer aplicación de este teorema se encuentra en la solución de problemas de minimización.

Antes de una interpretación económica al problema dual se resolverá el problema de producción del ejemplo 6.5.4. Este ejemplo no solo sirve como repaso del método simplex sino muestra como la tabla terminal de este problema permite resolver tanto el problema original como el dual.

Empleando el método simplex resuelva el problema de producción del ejemplo 6.5.4

A continuación aparecen las diferentes tablas que se establecen hasta encontrar la solución con el pivote en cada ocasión encerrado en un circulo y las columnas de las variables base marcadas con un asterisco (*).

* Aplicación del teorema dual a problemas de minimización.

Ejemplo 6.5.6

Solución:

	x_1	x_2	x_3	x_4	x_5		
	1	(2)	1	0	0	200	100 ⟸
	1	1	0	1	0	125	125
	1	0	0	0	1	100	∞
	-2	-3	0	0	0	0	

Punto O

528

½	1	½	0	0	100	200
(½)	0	-½	1	0	25	50 ⟸
1	0	0	0	1	100	100
-½	0	¾	0	0	300	

Punto A

529

x_2	0	1	1	-1	0	75
x_1	1	0	-1	2	0	50
x_5	0	0	1	-2	1	50
	0	0	1	1	0	325

Punto B

530

Tabla 6.5.3 Tablas simplex del ejemplo 6.5.6

Siguiendo las reglas expuestas previamente se puede obtener de inmediato la solución del problema de la última tabla. A saber:

* El valor máximo de la función objetivo es precisamente 325, el valor de las variables es:

$x_1 = 50, x_2 = 75, x_3 = x_4 = 0$ y $x_5 = 50$

Con objeto de aclarar el método también se incluye la solución gráfica en la figura 6.5.16.

A continuación se señala *como se obtiene la solución del problema dual de la tabla final del método simplex del problema original.

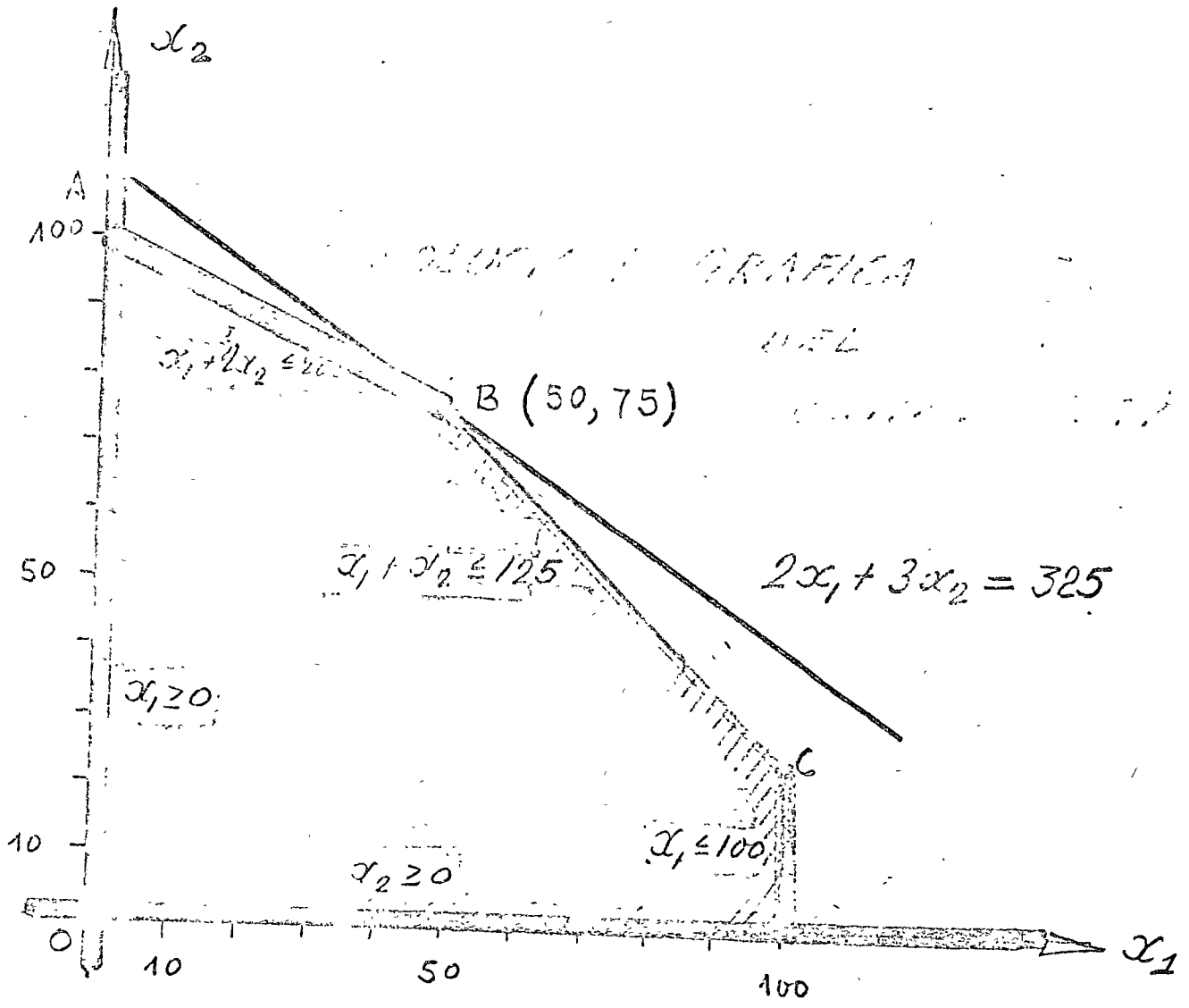


Fig. 6.5.16 Solucion grafica del ejemplo 6.5.7

La tabla 6.5.4 muestra la taba final del método simplex del problema original y como se obtienen de ella resultados del problema original y del dual.

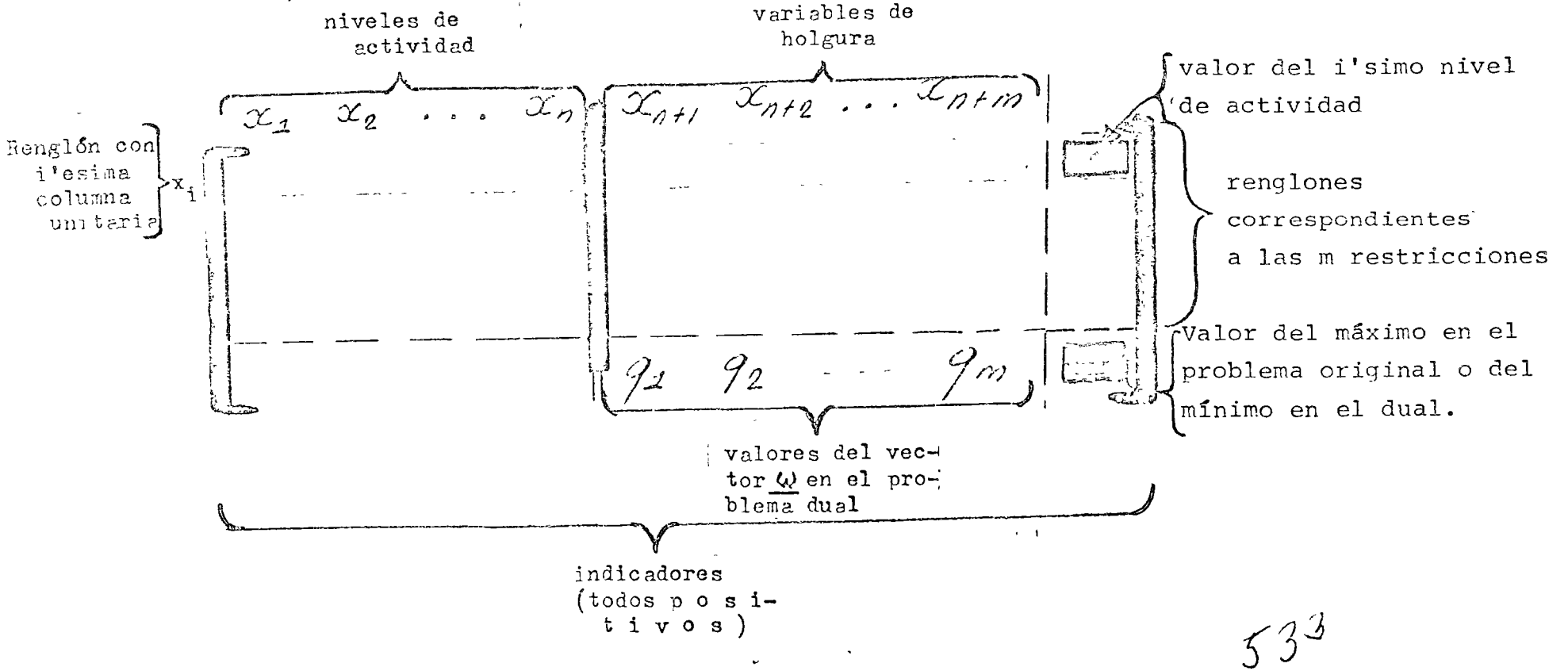


Tabla 6.5.4 Tabla final del método simplex del problema original.

El siguiente ejemplo ilustra el empleo de la tabla 6.5.4 para resolver el problema original y su dual.

Ejemplo 6.5.7

Obtenga de la tabla final del método simplex la solución del problema original y del dual del ejemplo 6.5.4

Solución:

A continuación aparece el planteamiento original del problema y el del dual.

Problema original

$$\max: \; m = \begin{bmatrix} 4 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Problema dual

$$\min: \; n = \begin{bmatrix} 200 & 125 & 100 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

Sujeto a las restricciones

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 200 \\ 125 \\ 100 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 1 \\ 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \geq \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

539

De la tabla 6.5.3 se tiene

$$\begin{array}{c} \\ x_2 \\ x_1 \\ \\ \end{array} \left[\begin{array}{ccccc|c} 0 & 1 & 1 & -1 & 0 & 75 \\ 1 & 0 & -1 & 2 & 0 & 50 \\ 0 & 0 & 1 & -2 & 1 & 50 \\ \hline 0 & 0 & 1 & 1 & 0 & 325 \end{array}\right]$$

$$m\big|_{max} = 325 \qquad x\big]_{m_{max}} = \begin{bmatrix} 50 \\ 75 \end{bmatrix}$$

$$n\big|_{min} = 325 \qquad w\big]_{n_{min}} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

* En efecto el valor mínimo de la función objetivo en el problema dual es:

* Valor mínimo de la función objetivo en el dual: 538

$$n = \begin{bmatrix} 200 \\ 125 \\ 100 \end{bmatrix}^T \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = 325$$

* A continuación se dará una interpretación económica a la solución del problema dual. Recuérdese el planteamiento del problema original y del problema dual:

* Interpretación económica del problema dual 539

Problema original

$$\max: \quad m = \underline{c}^T \underline{x}$$

sujeto a las restricciones

$$\underline{A}\ \underline{x} \leq \underline{b}$$

$$\underline{x} \geq 0$$

Problema dual 540

$$\min: \quad n = \underline{b}^T \underline{w}$$

$$\underline{A}^T \underline{w} \geq c$$

$$\underline{w} \geq 0$$

Si w_1, w_2 y w_3 son los costos de operación por hora de las máquinas A, B, y C respectivamente, la suma anterior en efecto representa el costo total de operación. Estos costos de operación, o sea las componentes del vector $\underline{w}$ reciben el nombre de precios sombra.

* $\underline{w}$ ≡ precios sombra 548

Es necesario ahora interpretar la otra ecuación del problema dual:

$$\underline{A}^T \underline{w} \geq \underline{c}$$ 549

Recuérdese que la componente a_{ij} de la matriz $\underline{A}$ representa el número de horas de la máquina i que se requiere para producir una unidad del producto j. En el producto $\underline{A}^T \underline{w}$ el primer renglón es:

* $\underline{A} = a_{ij}$, a_{ij} ≡ horas de máquina i para producir una unidad del producto j 550

* 1er renglón de $\underline{A}^T \underline{w}$

$$\left[\underline{A}^T \underline{w}\right]_1 = a_{11} w_1 + a_{21} w_2 + \ldots$$ 551

El término $a_{11} w_1$ representa el costo en la máquina 1 para producir una unidad del producto 1, y $a_{21} w_2$ el costo de la máquina 2, para producir una unidad del producto 1. Por lo tanto el producto

552

* $\underline{A}^T \underline{w}$ ≡ costo total de produc-

representa el costo total de producción de una unidad de cada uno de los productos. Como $*\underline{c}$ es el costo de venta de una unidad de cada uno de los productos, la desigualdad $\underline{A}^{T} \underline{w} \geq \underline{c}^{*}$ puede interpretarse de la siguiente manera: El costo de producción unitario $\underline{A}^{T} \underline{w}$ es por lo menos tan grande como el beneficio $\underline{c}$. Es posible extender esta interpretación del problema dual, para lo cual es necesario introducir teoremas que están fuera del alcance de esta obra **

Como con frecuencia es más facil resolver el problema dual que el primo o viceversa, resulta conveniente conocer ambos métodos.

Con el siguiente comentario finalizará esta sección sobre programación lineal.

ción por unidad de cada artículo

$*\underline{c} \equiv$ costo de los artículos

$*\underline{A}^{T} \underline{w} \geq \underline{c} \Longrightarrow$ Costo de producción unitario no debe exceder beneficio

---

**Ver capítulo 6, [illegible] (ref. 10) y capítulo 3 y 4 [illegible] (ref. 9).

Si en el ejemplo 6.5.2 los datos fuesen diferentes de manera que el sistema de ecuaciones del problema de programación lineal hubiese sido:

$$M = x_1 + 2\,x_2$$
$$x_1 + 4\,x_2 \leq 12$$
$$x_1 + x_2 \leq 4.5$$
$$3x_1 + x_2 \leq 10.5$$

554

* La solución óptima hubiese sido $x_1 = 2$, y $x_2 = 2.5$ como el lector podrá checar facilmente (ver problema 6.8.11). La solución de este problema para ser relevante debe ser entera. * Cuando como en este caso, la solución debe ser entera, puede recurrirse si las variables son suficientemente grandes y el resultado no es sensible a errores de aproximación a redondear el resultado a la cifra más próxima, ó puede recurrirse a la programación entera. (ref. 3)

*Solución optima: 555
$x_1 = 2$, $x_2 = 2.5$

* Redondear o programación entera 556

En la sección 6.8 el lector puede encontrar diversos problemas de programación lineal (problemas 6.8.10-6.8.15) y en el apéndice IA-171 encuentra un progra

ma de computadora para resolver este tipo de modelos.

El problema del ejemplo 6.5.2 ha sido resuelto empleando el programa A.17

Los datos de este problema aparecen en la tabla 6.5.5 y los resultados en la 6.5.6. En esta tabla las variables que no aparecen tienen un valor nulo.

bargo, existe un método que permite determinar la naturaleza de la línea; esta línea se llama *línea de regresión* y se calcula por medio del método de mínimos cuadrados[1].

Supongamos que queremos representar los datos correspondientes a n puntos (x_i, y_i) por medio de una línea recta. Los coeficientes a y b se pueden encontrar por medio del método de mínimos cuadrados, el cual nos da las fórmulas siguientes:

$$a = \frac{\Sigma y_i \Sigma x_i^2 - \Sigma x_i \Sigma x_i y_i}{\Delta} \qquad (10.17.1)$$

$$b = \frac{n\Sigma x_i y_i - \Sigma x_i \Sigma y_i}{\Delta} \qquad (10.17.2)$$

donde $\Delta = n\Sigma x_i^2 - (\Sigma x_i)^2$ (10.17.3)

La figura 10.17.2 muestra un programa para llevar a cabo estos cálculos; la figura 10.17.1 muestra el diagrama de flujo correspondiente. El programa consta de dos partes: un programa principal y

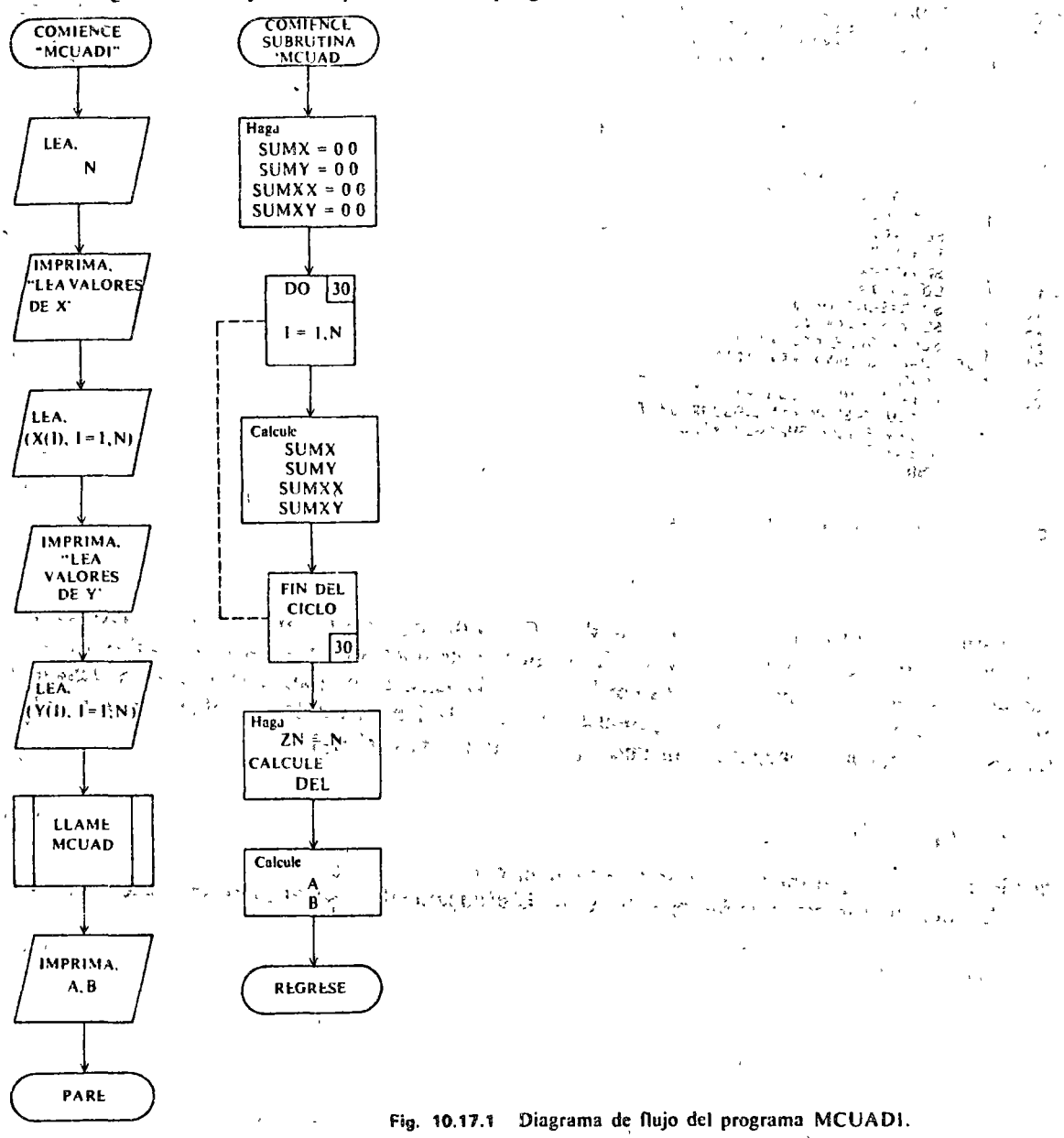


Fig. 10.17.1 Diagrama de flujo del programa MCUADI.

[1] W. Volk, *Applied Statistics for Engineers*, 2d. ed., McGraw-Hill, Nueva York, 1969, página 261.

una subrutina *llamada* MNCUAD El programa principal y la subrutina comparten los datos por medio de las proposiciones COMMON que aparecen en las líneas 1 y 12. Usted observará que no se hizo necesario escribir proposiciones DIMENSION para la definición de las variables X(I) y Y(I) ya que éstas son dimensionadas en las proposiciones COMMON. El programa principal es bastante corto. Esencialmente consta de tres proposiciones READ, una proposición CALL y tres proposiciones PRINT. Todos los cálculos se llevan a cabo en la subrutina. El ciclo DO que comienza en la línea 17, calcula las cuatro sumas requeridas por las ecuaciones de (10.17.1) a (10.17.3). Estas sumas se almacenan en las localizaciones SUMX, SUMY, SUMXY y SUMXX, las cuales han recibido nombres que concuerdan con los valores que contienen; luego se calcula el valor de DEL y, finalmente, se calculan los valores de A y B. La figura 10.17.3 muestra un ejemplo de la corrida del programa.

```
    C     MCUAD1
 1        COMMON X(20),Y(20),N,A,B
 2        READ,N
 3        PRINT,' LEA VALORES DE X'
 4        READ,(X(I),I=1,N)
 5        PRINT,' LEA VALORES DE Y'
 6        READ,(Y(I),I=1,N)
 7        CALL MNCUAD
 8        PRINT,A,B
 9        STOP
10        END
11        SUBROUTINE MNCUAD
12        COMMON X(20),Y(20),N,A,B
13        SUMX=0.
14        SUMY=0.
15        SUMXX=0.
16        SUMXY=0.
17        DO 30 I=1,N
18        SUMX=SUMX+X(I)
19        SUMY=SUMY+Y(I)
20        SUMXX=SUMXX+X(I)**2
21     30 SUMXY=SUMXY+X(I)*Y(I)
22        ZN=N
23        DEL=ZN*SUMXX-SUMX**2
24        A=(SUMY*SUMXX-SUMX*SUMXY)/DEL
25        B=(ZN*SUMXY-SUMX*SUMY)/DEL
26        RETURN
27        END
```

Fig 10.17.2 Programa MCUADI

Estudiemos brevemente la aplicación del método de mínimos cuadrados a datos recolectados en el estudio de flujo de tráfico[1]. Las variables comúnmente usadas en este estudio son dos: u, la velocidad de los vehículos medida en millas por hora, y d, la densidad de vehículos medida en vehículos por milla. Los ingenieros de tráfico han propuesto varias relaciones entre las variables u y d; nosotros estudiamos la siguiente relación exponencial (propuesta por Greenberg)[2]:

$$d = d_m \, e^{-u/u_m} \tag{10.17.4}$$

donde d_m y u_m son parámetros que se desean determinar.

Al hacer una trasformación logarítmica de la ecuación (10.17.4), encontramos que

$$\ln d = \ln d_m - \frac{1}{u_m} u \tag{10.17.5}$$

[1] D. R. Drew, *Traffic Flow Theory and Control*, McGraw-Hill, Nueva York, 1968, página 310
[2] M. Wohl y B. V. Martin, *Traffic System Analysis*, McGraw-Hill, Nueva York, 1967, página 332

DIRECTORIO DE ASISTENTES AL CURSO DE METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL (DEL 29 DE MARZO AL 10 DE ABRIL DE 1976)

	NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
1.	FRANCISCO ALMADA VALENZUELA Calle Ocho "A" 19-3 Col. Vértiz Narvarte México 13, D. F.	SECRETARIA DE RECURSOS HIDRAULICOS Viena 20-302 Col. Juárez México 6, D. F. Tel: 5-92-39-82
2.	ARMANDO AYALA FONTES Calzada Arteaga 1209-C Tlatelolco México 3, D. F. Tel: 5-83-95-18	SECRETARIA DE RECURSOS HIDRAULICOS Viena 20-302 Col. Juárez México 6. D. F. Tel: 5-92-35-68
3.	HUMBERTO BASTIDAS ORTIZ R.G. Robles 73 Sur Culiacán Rosales, Sin. Tel: 2-97-29	UNIVERSIDAD AUTONOMA DE SINALOA Angel Flores s/n Pte. Culiacán Rosales, Sin. Tel: 2-35-50
4.	GABRIEL CARMONA WALKUP Av. Mazatlán Edificio Condesa T-8 Col. Condesa México 11, D. F. Tel: 5-53-42-13	SECRETARIA DE RECURSOS HIDRAULICOS Teotihuacán No. 19 Col. Roma México 7, D. F. Tel: 5-74-56-09
5.	ING. VICTOR CASTILLO México, D. F.	COMISION FEDERAL DE ELECTRICIDAD Ródano No. 14 México, D. F.
6.	ING. JOSUE CORNEJO Electricistas No. 103 Col. 20 de Noviembre México 2, D. F. Tel: 5-29-41-92	SECRETARIA DE MARINA Lerdo de Tejada No. 6 México, D. F. Tel: 5-69-37-69
7.	JESUS DIAZ BARRIGA CHAVEZ Unidad H. Juan de Dios Bátiz Edificio 24-2-A México, D. F.	SECRETARIA DE OBRAS PUBLICAS Xola y Av. Universidad Col. Narvarte México, D. F.

DIRECTORIO DE ASISTENTES AL CURSO DE METODOS NUMERICOS Y APLICACIONES CON LA COMPUTADORA DIGITAL (DEL 29 DE MARZO AL 10 DE ABRIL DE 1976)

	NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
8.	JOSE LUIS FIGUEROA CORREA Av. 8 No. 249 Col. Ignacio Zaragoza México 9, D. F. Tel: 5-71-46-74	ALTOS HORNOS DE MEXICO, S.A. Av. Juárez No. 90 México 1, D. F. Tel: 5-85-57-00 Ext. 293
9.	ING. ARTURO GARCIA GALINDO Ramón Prida No. 30 Col. Jardín Balbuena México 9, D. F. Tel: 5-52-11-93	COMISION FEDERAL DE ELECTRICIDAD Paseo de la Reforma No. 107-4o. Piso Mexico, D. F.
10.	GONZALO GUTIERREZ TORRES Isabel La Católica No. 1002 Niños Héroes México 13, D. F. Tel: 5-90-72-08	SISTEMA DE TRANSPORTE COLECTIVO "METRO" Delicias No. 67 México 4. D. F. Tel: 5-21-86-20 Ext. 554
11.	ING. XAVIER HARO SOLORZANO Aniceto Ortega No. 955 Col. del Valle México 12, D. F. Tel: 5-75-04-28	COMISION DE AGUAS DEL VALLE DE MEXICO Balderas No. 55 México 1, D. F. Tel: 5-10-02-94
12.	ROSENDA LUZ LARA ALVAREZ Tuxpango 127 Col. Industrial México 14, D. F. Tel: 5-17-09-11	INSTITUTO DE ASTRONOMIA Torre de Ciencias México 20, D. F. Tel: 5-48-53-05
13.	ZOILO MENDOZA NUÑEZ Calle 623 No. 109 San Juan de Aragón México 14, D. F.	UNIVERSIDAD AUTONOMA METROPOLITANA UNIDAD AZCAPOTZALCO Av. San Pablo s/n Azcapotzalco México, D. F. Tel: 5-61-37-33 Ext. 183
14.	ROGELIO ORTEGA ARENAS Bosque del Corregidor No. 22 Fracc. La Herradura México 10, D. F. Tel: 5-89-16-45	PETROLEOS MEXICANOS Av. Marina Nacional No. 329 Col. Anáhuac México, D. F. Tel: 5-31-62-63

...AL CURSO DE METODOS NUMERICOS Y APLICACIONE...
...CON LA COMPUTADORA DIGIT... (DEL 29 DE MARZO AL 10 DE ABRIL DE 1976)

	NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
15.	HECTOR E. MORALES CARREÑO Juan de Dios Peza No. 152 Col. Obrera México 8, D. F. Tel: 5-78-58-56	PETROLEOS MEXICANOS Av. Marina Nacional No. 329 Col. Anzures México 17, D. F. Tel: 5-81-96-65
16.	VICTOR MANUEL RIOS NORIEGA Playa Gaviotas No. 22 Col. Reforma Iztaccihuatl México 13, D. F. Tel: 5-39-90-58	COMISION DE AGUAS DEL VALLE DE MEXICO Balderas No. 55 México 1, D. F. Tel: 5-10-02-94
17.	ING. MANUEL ARNOLDO RODRIGUEZ Lag. de Tamiahua No. 1359 Las Quintas Culiacán Rosales, Sin.	UNIVERSIDAD AUTONOMA DE SINALOA Angel Flores y Riva Palacio Culiacán Rosales, Sin.
18.	ING. IGNACIO ROSAS IBARRA 17 Poniente 1719 Puebla, Pue. Tel: 42-17-32	UNIVERSIDAD AUTONOMA DE PUEBLA 4 Sur No. 104 Puebla, Pue.
19.	ING. MIGUEL A. RUIZ VELASCO Y R. Mier y Pesado No. 136 Col. del Valle México 12 D. F. Tel: 5-23-08-23	FACULTAD DE INGENIERIA, UNAM Ciudad Universitaria México 20, D. F. Tel: 5-50-00-40
0.	ARQ. ROBERTO SAEZ ZUBIETA Ave. Minerva No. 286 Col. Florida México 20, D. F. Tel: 5-34-53-25	DESPACHO PARTICULAR Av. Minerva No. 286 Col. Florida México 20, D. F. Tel: 5-34-53-25
	CARLOS A. TELLA SAIZ Av. Pacífico No. 277 Deplo. 101-D Coyoacán México 21, D. F. Tel: 5-44-44-39	BUFETE INDUSTRIAL DISEÑOS Y PROYECTOS Tolstoi No. 22 Col. Anzures México 5, D. F. Tel: 5-33-15-00 Ext. 106

DIRECTORIO DE ASISTENTES AL CURSO [illegible]
CON LA COMPUTADORA DIGITAL (DEL 29 DE MARZO AL [illegible]

	NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
22.	ING. ALFONSO TOVAR SANTANA Nonoalco 205 Edif. V. Gro. "B"-704 Unidad Nonoalco México 3, D. F. Tel: 5-83-29-37	ESCUELA SUPERIOR DE INGENIERIA Y ARQUITECTURA I.P.N. Zacatenco México, D. F.