UNIVERSIDAD NACIONAL AUTONOMA

DE MEXICO

EL SIMPLEX DESDE UN PUNTO DE VISTA NUMERICO

T E S I S

Que Para Obtener el Título de

M A T E M A T I C O

P r e s e n t a

MARTIN TORRES VALDERRAMA

México, D.F. 1982

Prólogo

En este trabajo intentamos dar una versión simplificada de algunos aspectos numéricos que tienen los problemas de la Programación Lineal y el método que normalmente se emplea para resolverlos: el Método Simplex.

La presentación que nosotros hacemos muestra un tanto la evolución de las diferentes implementaciones que se han hecho del Método Simplex, para darnos cuenta cómo se van incorporando las nuevas ideas, los recursos tanto matemáticos como computacionales. Es nuestro particular punto de vista que en nuestro medio no hay material reunido donde se muestre este tema a nivel de divulgación, lo cual nos ha motivado a presentar fundamentalmente las ideas, simplificar algunas partes, omitir ciertos detalles y remitir al lector interesado a la Literatura para profundizar en cualesquiera de los tópicos que sean de su interés. No puede encontrarse por consiguiente un estudio muy detallado de cada una de las partes que integran

tanto la Programación Lineal como el Método Simplex.

Desde la aparición del trabajo de Bartels-Golub, al final de la década de los 60's, se ha iniciado un estilo de trabajo que hacía falta en esta rama de las Matemáticas Aplicadas. Después de B-G se suceden modificaciones a la idea principal, hay nuevos puntos de vista, y se combinan esquemas; Forrest-Tomlin, Golfarb, Duff-Reid, Saunders, Gill-Murray, Gay, son algunos de los principales autores en esta materia. Algunos de estos matemáticos logran implementaciones notables por su estabilidad numérica y por su competitividad con las versiones clásicas.

El material que presentamos lo hemos reunido en cinco capítulos. En el capítulo 1 mostramos básicamente los elementos de la Programación Lineal y el Método Simplex. Un enfoque geométrico motiva la estrecha relación que hay entre el problema y la metodología. En el capítulo 2 presentamos una discusión de los métodos de eliminación para resolver el proble-

ma $AX=b$, haciendo algunos comentarios sobre la estabilidad numérica, la cantidad de aritmética involucrada y el aspecto del "llenado" de las matrices durante el proceso de eliminación. Como una parte importante del ciclo que emplea el Método Simplex, en el capítulo 3 se muestran algunas formas de actualizar la inversa $\hat{B}^{-1}$ de una matriz básica $\hat{B}$, a partir de una matriz básica B disponible. En el entendido de que en el caso: $B-G$, se actualizan los factores triangulares $L\,U$. Brevemente en el capítulo 4 presentamos las técnicas para resolver el vector de costos, lo hacemos por separado del "ciclo del Simplex" por ser una parte que consume significativamente tiempo de máquina. Finalmente en el capítulo 5 bosquejamos el mismo Método Simplex pero abordado desde el punto de vista de las transformaciones ortogonales.

Advertencia

Durante el desarrollo de este trabajo hablaremos indistintamente del problema de maximización y el de minimización. El objeto de hacerlo de esta manera obedece a que en ocasiones resulta más claro hablar en términos de maximizar, en tanto que en otras, hablar en estos términos oscurece el concepto. Además adoptaremos una notación que sea la más común.

Una matriz con coeficientes reales la denotaremos de varias formas según sea el contexto en que se emplee

A : matriz de m renglones y n columnas

B : matriz formada por columnas básicas

D : matriz formada por columnas no básicas

S : matriz de orden m.

Frecuentemente emplearemos la siguiente notación

$$A = a_{ij}$$

$$= [\, a_1 \mid a_2 \mid \cdots \mid a_n \,]$$

$$= [\, B \mid D \,]$$

$$= \left[\begin{array}{c|c} X & Y \\ \hline Z & W \end{array}\right]$$

La notación para los vectores será la usual aunque también emplearemos letras griegas para representarlos.

b : vector columna de m componentes

x : vector columna de n componentes

c^T : vector renglón de n componentes

e_j : vector j-ésimo de I_m $e_j = (0, \ldots, 0, 1, 0, \ldots, 0)^T$

η : vector columna que "define" las matrices elementales

Por $x \geq 0$ debemos entender que las componentes x_i, $i = 1, \ldots, n$ del vector x son no negativas.

Indice General

1 Introducción al Simplex 1

1.1 Geometría del Problema 2
1.2 Soluciones Básicas Admisibles 8
1.3 Existencia de Soluciones Básicas Admisibles 17
1.4 Cambio de Soluciones Básicas 22
1.5 Simplex Revisado 30

2 Sistemas Sparse 34

2.1 Eliminación de Gauss-Jordan 36
2.1.A Descripción del Método 36
2.1.B Forma Matricial 38
2.1.C Consideraciones Numéricas 43
2.2 Eliminación de Gauss 52
2.2.A Descripción del Método 52
2.2.B Forma Matricial 55
2.2.C Consideraciones Numéricas 62
2.2.D Consideraciones Sparse 69
2.3 Descomposición LU 78
2.4 Descomposición LU para Matrices Ralas 83

2.4.A Esquema de Saunders 89
2.4.B Esquema de Reid 91

3 Actualización de Sistemas Sparse 95

3.1 Actualización en F.P.I. 97
3.2 Actualización en F.E.I. 101
3.3 Actualización en F.E.I. según Tomlin 105
3.4 Actualización de la Descomposición LU según Bartels-Golub 113
3.5 Actualización en LU según Reid 118
3.6 Actualización en LU según Saunders 127
3.7 Actualización en LU según Gay 132

4 Vector de Costos 135

4.1 Solución del Sistema $B^T\lambda = C_B$ 137
4.2 Actualización 139
4.3 Actualización según Tomlin 140
4.4 Actualización según Golfarb 143

5. El Simplex vía la Descomposición QR 146

5.1 Transformaciones de Householder 147
5.2 Descomposición QR y LQ 152

5.2.A Descripción del Método 152
5.3 Versión de Saunders del Simplex 156
5.4 Consideraciones Numéricas 159
5.5 Actualización de la Factorización LQ 164
5.5A Actualización de LDL^T según Saunders por el Método Directo 166
5.5.B Actualización de LDL^T vía Transformaciones Ortogonales según Gill y Murray 169
5.6 Consideraciones Sparse 173
5.6.A Triangular por Bloques y Espigas 174
5.6.B Estructura por Bloques 175
5.6.C Estructura de Escalera 177
5.6.D Angular por Bloques 178

Bibliografía 179

1. Introducción al Simplex

Introducción

Como un problema matemático, el problema clásico general de la Programación Lineal, podemos describirlo de la manera siguiente: dado un conjunto de restricciones lineales, hallar una solución que minimice o maximice (optimice) una funcional lineal dada. En términos más breves podemos decir

$$\text{Minimice } z = c^T x$$
$$Ax \leq b \quad , x \geq 0$$

donde A es una matriz de $m \times n$, los vectores c y x son de n componentes, y el vector b es de m componentes.

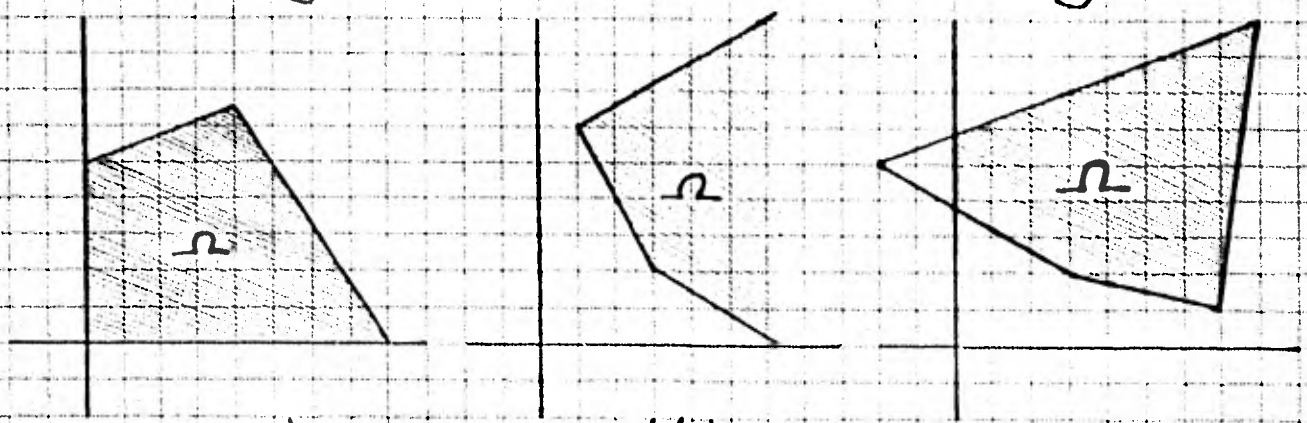
Problemas de este tipo los hay en número considerable, baste citar unos cuantos: asignación de recursos, distribución de personal, elaboración de dietas, transporte, optimización en el uso de maquinaria, problemas relacionados con la Industria del Petróleo, teoría de redes, etc. La técnica comúnmente empleada en la solución de ésos y muchos otros problemas es conocida como el Método Simplex. Los elementos mínimos que nos permitirán hablar de él se exponen a continuación.

1.1 Geometría del Problema

Denotemos por Ω al conjunto de los $X \in \mathbb{R}^n$ tales que

$$AX \leq b \,, \quad X \geq 0$$

A Ω le llamaremos conjunto de soluciones admisibles. Algunos ejemplos en $\mathbb{R}^2$ de este tipo de conjuntos se muestran en la figura 1.



(a) (b) (c)

fig. 1

Estos politopos pueden ser acotados (fig.1 (a) y (c)), o no (fig. 1 (b)), sin embargo todos ellos son conjuntos convexos. Esto es, dados $X, Y \in \Omega$, $\alpha X + \beta Y \in \Omega$, para todos los $\alpha, \beta \geq 0$ tales que $\alpha + \beta = 1$

Puesto que el objetivo es encontrar una solución $X^* \in \Omega$ donde el valor de la funcional

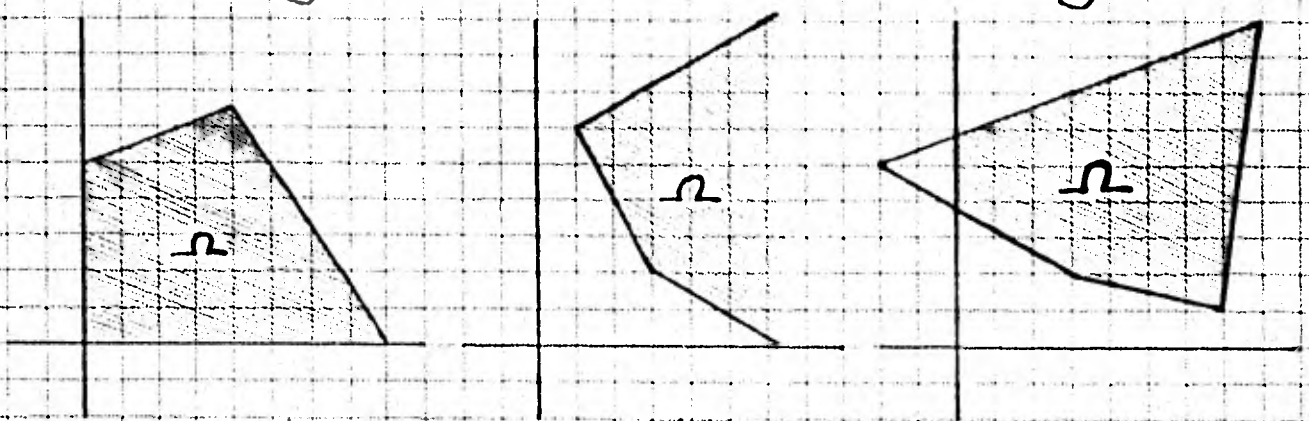
$$Z = C^T X$$

1.1 Geometría del Problema

Denotemos por Ω al conjunto de los $X \in \mathbb{R}^n$ tales que

$$AX \leq b \quad , \quad X \geq 0$$

A Ω le llamaremos conjunto de soluciones admisibles. Algunos ejemplos en $\mathbb{R}^2$ de este tipo de conjuntos se muestran en la figura 1.



(a) (b) (c)

fig. 1

Estos politopos pueden ser acotados (fig.1 (a) y (c)), o no (fig. 1 (b)), sin embargo todos ellos son conjuntos convexos. Esto es, dados $X, Y \in \Omega$, $\alpha X + \beta Y \in \Omega$, para todos los $\alpha, \beta \geq 0$ tales que $\alpha + \beta = 1$

Puesto que el objetivo es encontrar una solución $X^* \in \Omega$ donde el valor de la funcional

$$Z = C^T X$$

sea óptimo, es natural plantearse el problema de caracterizar, bajo el supuesto de existencia, el subconjunto de Ω de los puntos óptimos.

Para ello, por un hiperplano H, entenderemos un conjunto de la forma

$$H = \{x \in \mathbb{R}^n \mid c^T x = cte.\}$$

Es fácil ver que H es un conjunto convexo.

En lo que sigue, H_z denotará al hiperplano

$$H_z = \{x \in \mathbb{R}^n \mid c^T x = z\}, \ (z \in \mathbb{R}, \text{dado}).$$

Claramente H_z pasará por el origen si y sólo si $z = 0$

Tomando dos vectores $x_1, x_2 \in H_z$ se tiene

$$c^T(x_1 - x_2) = c^T x_1 - c^T x_2 = z - z = 0$$

Por consiguiente c es ortogonal al segmento dirigido $\overline{x_1 x_2}$. De modo que c es ortogonal a todo segmento dirigido contenido en el hiperplano H_z

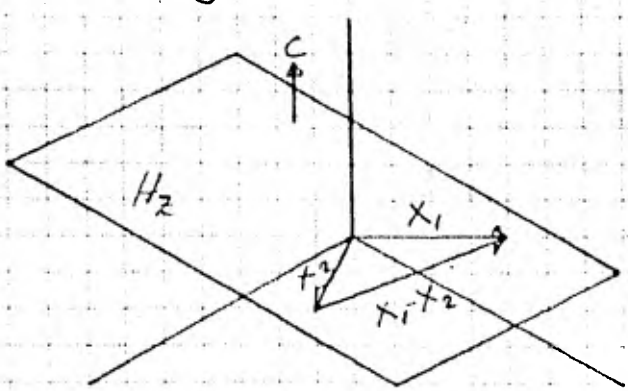


fig 2

El hiperplano H_z define dos semiespacios cerrados, positivo y negativo

$$H^+ = \{ x \in \mathbb{R}^n / c^T x \geq z \}$$

$$H^- = \{ x \in \mathbb{R}^n / c^T x \leq z \}$$

así como también dos semiespacios abiertos igualmente positivo y negativo, respectivamente.

$$\mathring{H}^+ = \{ x \in \mathbb{R}^n / c^T x > z \}$$

$$\mathring{H}^- = \{ x \in \mathbb{R}^n / c^T x < z \}$$

Es directo demostrar que estos semiespacios (cerrados como abiertos), son conjuntos convexos. Asimismo, H_z es un conjunto convexo.

Ahora si movemos H_z en dirección de su vector normal, tendremos que z se incrementa, mientras que si movemos H_z en dirección de menos su vector normal, z se decrementa. Para el caso $n=2$, la figura 3 ilustra esta situación.

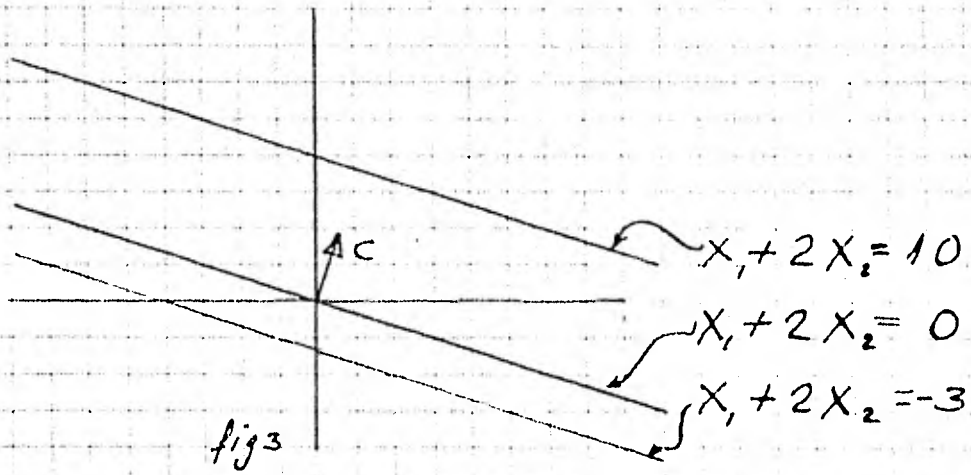


fig 3

Conjuntando lo anterior para el problema

$$(\mathcal{P}) \quad \begin{cases} \text{Máx } z = c^T x \\ \text{s.a. } Ax \leq b \quad , x \geq 0 \end{cases}$$

ilustrado para el caso $n=2$, en la siguiente figura.

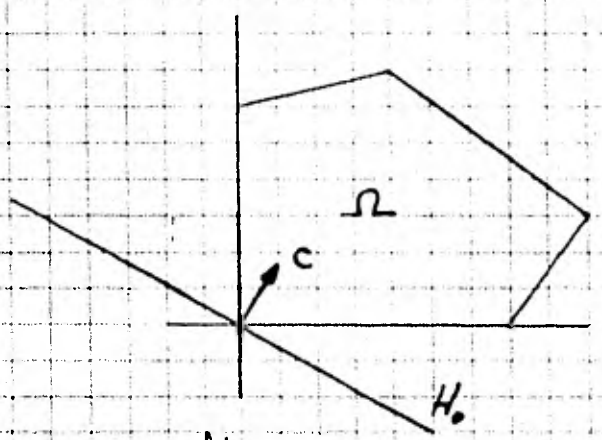


fig. 4

Tenemos que andamos buscando el elemento H_μ de la familia H_z $(z \in \mathbb{R})$, más alejado de H_0 en dirección de c, que corte a Ω (Véase fig. 5). Resultando $H_\mu \cap \Omega$ el conjunto de soluciones del problema.

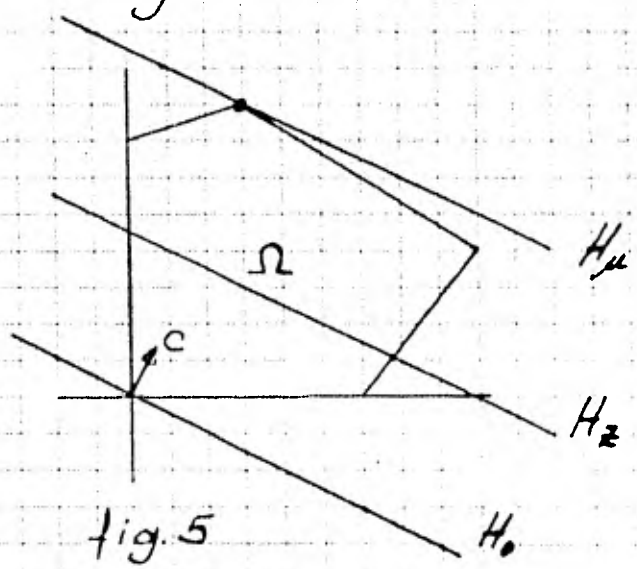


fig. 5

En la figura 6 se ilustran otras situaciones geométricas para $H_\mu \cap \Omega$. Es inmediato ver que el conjunto de soluciones $H_\mu \cap \Omega$ es un conjunto convexo.

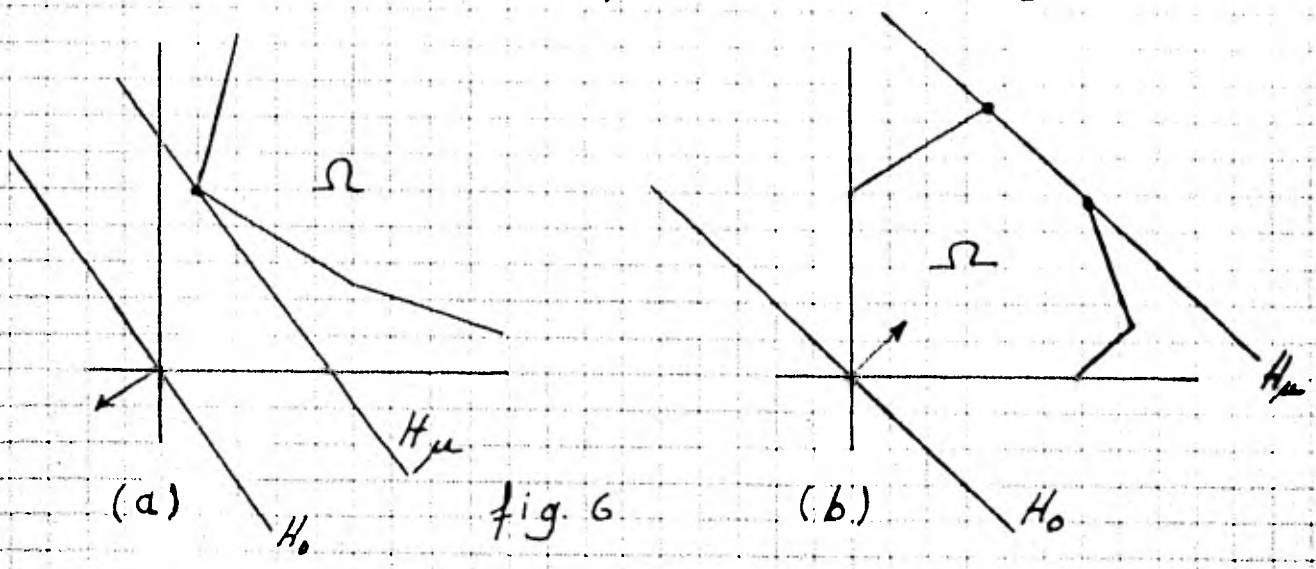


fig. 6

Definición. Sea $K \neq \emptyset$ un conjunto convexo. Supóngase que K no es todo $\mathbb{R}^n$. Un hiperplano H es llamado soporte para K si $H \cap \bar{K} \neq \emptyset$,(1) y K está contenido en uno de los semiespacios cerrados definidos por H.

En general se tiene el siguiente

Teorema. El problema $\mathcal{P}$ tiene solución si y sólo si existe $H_\mu \in \{H_z\}$ tal que H_μ es un hiperplano soporte para Ω y $H_{\mu+\varepsilon} \cap \Omega = \emptyset$ para cualquier $\varepsilon > 0$ dado. Más aún, $H_\mu \cap \Omega$ el conjunto de soluciones de $\mathcal{P}$ es un conjunto convexo [39].

(1) Por $\bar{K}$ se entiende la cerradura de K.

El conjunto convexo Ω tiene ciertos elementos muy especiales. Estos en lenguaje geométrico son llamados vértices, en tanto nosotros los definiremos como sigue.

Definición. Un punto $x \in \Omega$ es un punto extremal de Ω si no se puede escribir como combinación convexa de otros dos puntos $x_1, x_2 \in \Omega$.

El siguiente resultado juega un papel central en la Programación Lineal:

Teorema. Si el problema P tiene solución, entonces el conjunto solución $H_\mu \cap \Omega$ contiene al menos un punto extremal de Ω [39].

En la siguiente sección veremos que el conjunto de todos los puntos extremales de Ω es siempre finito. Este hecho y el resultado anterior sugieren un procedimiento para hallar un punto de Ω extremal y óptimo, el cual consiste: en una primera etapa, en hallar un punto extremal de Ω; y en una segunda etapa, en moverse sobre puntos extremales de Ω, siempre buscando el incremento de la función objetivo hasta alcanzar un punto extremal óptimo. Este procedimiento es la filosofía que hay en el fondo del Método Simplex.

1.2 Soluciones Básicas Admisibles.

Con objeto de caracterizar algebraicamente las soluciones extremales de un problema de Programación Lineal, se replanteará la formulación inicial. Esto es, daremos una formulación que resulte equivalente a la original pero que sea de más fácil manejo en la práctica.

Dado el problema

$$\text{Min } \tilde{z} = \tilde{c}^T \tilde{x} \tag{1.2.1}$$
$$\text{s.a. } \tilde{A}\tilde{x} \leq \tilde{b}\,,\ \tilde{x} \geq 0$$

se transforma a uno de la forma

$$\text{Min } z = c^T x \tag{1.2.2}$$
$$\text{s.a. } Ax = b\,,\ x \geq 0$$

donde $c^T = [\tilde{c}, 0]$, $x = [\tilde{x}, 0]^T$, $A = [\tilde{A}, I]$, añadiendo a cada una de las restricciones una cantidad x_{n+i} no negativa, i.e.,

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n + x_{n+i} = b_i$$
$$x_{n+i} \geq 0\,,\ i = 1, \ldots, m\,.$$

De tal modo que si se tienen m-restricciones y n-variables en (1.2.1), se tendrán las mismas restricciones pero con (n+m)-variables en (1.2.2). Estas m-variables adicionales se llaman variables de holgura o simplemente holguras.

Esta nueva formulación la denominaremos formulación estandar del problema de Programación Lineal.

Primeramente conviene señalar el carácter homogéneo de todas las variables en esta nueva formulación.

Sean la recta

$$L: a_1 X_1 + a_2 X_2 + b = 0$$

y el punto $P_0 = (X_1^\circ, X_2^\circ)$

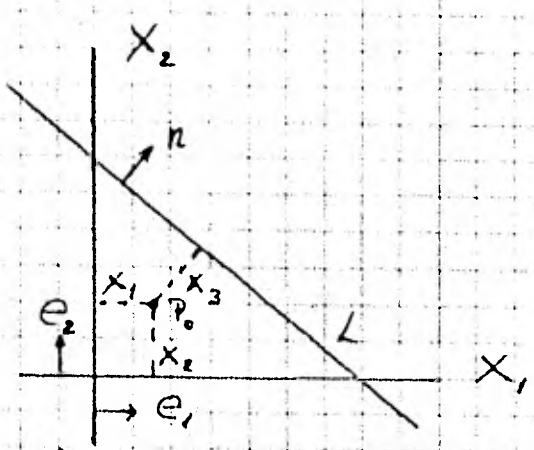


fig. 7

El punto P_0 puede ser referido no solamente al sistema (X_1, X_2), sino también al sistema formado por las rectas X_1, X_2 y L, el cual denotaremos como (X_1, X_2, L). Por consiguiente las componentes de P_0 en este sistema serán $(X_1^\circ, X_2^\circ, X_3)$, en donde cada X_i, $i = 1, 2, 3$, mide la distancia dirigida de P_0 a cada recta. Obviamente X_1°, X_2° son las componentes

usuales, mientras que X_3 estará dada por

$$-X_3 = d(P_0, L)\sqrt{a_1^2 + a_2^2}$$

donde

$$d(P_0, L) = \frac{a_1 X_1^0 + a_2 X_2^0 + b}{\sqrt{a_1^2 + a_2^2}}$$

De la misma manera, si tenemos las rectas L_i, $i=1,\ldots,m$, un punto P_0 podemos referirlo igualmente al sistema formado por dichas rectas. Es decir, P_0 tendrá tantas componentes como rectas tenga el sistema.

Consideremos el siguiente ejemplo

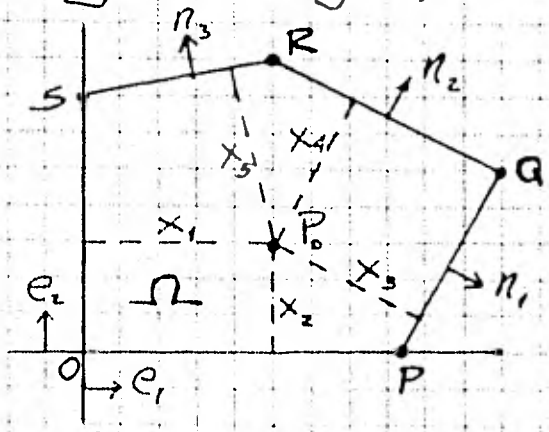


fig. 8

Un punto como indica la figura 8 tiene "distancias" $X_i \neq 0$, $i=1,2,\ldots,5$. Considerando las "distancias de la siguiente manera

En este ejemplo cada L_i, $i=1,2,3$, es de la forma

$$L_i : a_{i1} X_1^0 + a_{i2} X_2^0 - b_i = 0$$

entonces la distancia de un punto P_0 a cualesquiera de las rectas L_i viene dada por

$$d(P_0, L_i) = \frac{a_{i1}x_1^0 + a_{i2}x_2^0 - b_i}{\sqrt{a_{i1}^2 + a_{i2}^2}}, \quad i = 1, 2, 3$$

Escribiendo $\alpha_i = \sqrt{a_{i1}^2 + a_{i2}^2}$, y haciendo $a_{i1}x_1 + a_{i2}x_2 - b_i = -x_{2+i}$, $i = 1, 2, 3$, se tiene que

$$\alpha_i d(P_0, L_i) = -x_{2+i}, \quad i = 1, 2, 3$$

Esto es, cada variable x_{2+i} es igual a menos la distancia dirigida de P_0 a cada restricción multiplicada por un escalar positivo α_i. Las otras dos distancias x_1, x_2 son las componentes usuales.

Pasemos a los puntos extremales de Ω.

La idea de distancia dirigida en este contexto sigue siendo válida si suponemos que P_0 es alguno de los puntos extremales de Ω. Es decir, si se está por ejemplo en el punto O, entonces $x_1 = x_2 = 0$ y x_3, x_4, x_5 positivos, situación análoga se tiene al considerar algún otro punto, digamos P_5 para el cual $x_2 = x_3 = 0$, y x_1, x_4, x_5 positivos. Ver figura 9

entonces la distancia de un punto P_0 a cualesquiera de las rectas L_i viene dada por

$$d(P_0, L_i) = \frac{a_{i1}X_1^0 + a_{i2}X_2^0 - b_i}{\sqrt{a_{i1}^2 + a_{i2}^2}}, \quad i = 1, 2, 3$$

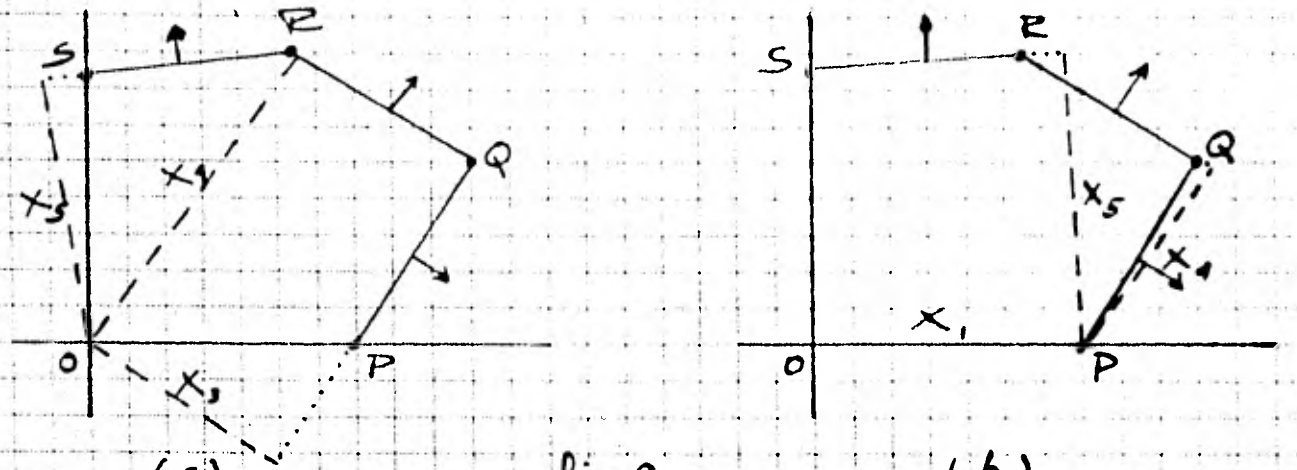
Escribiendo $\alpha_i = \sqrt{a_{i1}^2 + a_{i2}^2}$, y haciendo $a_{i1}X_1 + a_{i2}X_2 - b_i = -X_{2+i}$, $i = 1, 2, 3$, se tiene que

$$\alpha_i d(P_0, L_i) = -X_{2+i}, \quad i = 1, 2, 3$$

Esto es, cada variable X_{2+i} es igual a menos la distancia dirigida de P_0 a cada restricción multiplicada por un escalar positivo α_i. Las otras dos distancias X_1, X_2 son las componentes usuales.

Pasemos a los puntos extremales de Ω.

La idea de distancia dirigida en este contexto sigue siendo válida si suponemos que P_0 es alguno de los puntos extremales de Ω. Es decir, si se está por ejemplo en el punto O, entonces $X_1 = X_2 = 0$ y X_3, X_4, X_5 positivos, situación análoga se tiene al considerar algún otro punto, digamos P_1, para el cual $X_2 = X_3 = 0$, y X_1, X_4, X_5 positivos. Ver figura 9



(a) fig. 9 (b)

Con el propósito de formalizar las anotaciones del párrafo anterior, se introduce la siguiente

Definición. Sea A una $(m \times n)$-matriz real con $m < n$. Una solución del problema

1.2.3 $$Ax = b \ , \ x \geq 0$$

se dirá que es básica si (i) ésta no tiene más de m-componentes positivas y (ii) las columnas de A correspondientes a las componentes positivas de $\hat{x}$ constituyen un conjunto linealmente independiente (l.i.). Y se dirá que $\hat{x}$ es una solución básica no-degenerada si en vez de (i) se pide (i') $\hat{x}$ tiene m-componentes positivas..

En base a la cual se tiene el siguiente resultado fundamental

Teorema ([39], pág. 21). Los puntos extremales de

$$\Omega = \{ x \in \mathbb{R}^n \mid Ax = b, x \geq 0 \}$$

están en correspondencia 1–1 con las soluciones básicas admisibles de

$$Ax = b, x \geq 0$$

Ahora bien, se sigue de manera natural, que toda solución básica admisible $\hat{x}$ tiene asociada una única submatriz de A:

$$B = [a^{s_1} | a^{s_2} | \ldots | a^{s_{m'}}],$$

siendo s_i el subíndice correspondiente a la s_i-ésima componente positiva de $\hat{x}$, con las propiedades siguientes:

(1.2.a) B tiene columnas l.i. (lo que implica que $m' \leq m$), y

(1.2.b) $b \in \dot{C}ono(B)$

en donde

$$\dot{C}ono(B) = \{ y \in \mathbb{R}^m \mid y = Bz, z > 0 \}$$

En lo que sigue se dirá que B es una submatriz básica de A, si B cumple con (1.2.a) y (1.2.b).

Para ilustrar lo anterior, volvamos a nuestro ejemplo, que en notación matricial toma la forma

(1.2.4)
$$\begin{bmatrix} a_{11} & a_{12} & 1 & 0 & 0 \\ a_{21} & a_{22} & 0 & 1 & 0 \\ a_{31} & a_{32} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

con $x_i \geq 0$, $i = 1, \ldots, 5$

Tenemos que la solución básica con $x_1 = x_2 = 0$, correspondiente al punto extremal O, dá lugar al subsistema

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

con submatriz básica

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

Mientras que la solución básica con $x_2 = x_3 = 0$, correspondiente al punto extremal P, tiene asociado el subsistema

$$\begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & 1 & 0 \\ a_{31} & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

con submatriz básica

$$\begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & 1 & 0 \\ a_{31} & 0 & 1 \end{bmatrix}$$

Hemos visto que las soluciones básicas del problema

$$AX = b \quad , \quad X \geq 0$$

están en correspondencia 1-1 con las submatrices básicas B de A. Así, la demostración de nuestro teorema corresponde a probar la correspondencia 1-1 entre los puntos extremales de

$$\Omega = \{ X \in \mathbb{R}^n \mid AX = b, X \geq 0 \}$$

y las submatrices básicas B de A ([35], [39]).

Como un corolario del resultado central de esta sección tenemos que el número de puntos extremales y sus correspondientes submatrices básicas, considerando solamente puntos extremales que correspondan a soluciones básicas no degeneradas, está acotado por $\binom{n}{m}$. De aquí en adelante y mientras no se aclare lo contrario, hablaremos de soluciones básicas, entendiendo sólo soluciones básicas no degeneradas.

Como decíamos arriba, debido a que hay solamente un número finito de soluciones básicas, un procedimiento inicial sería calcular todas ellas y ver cuál es la óptima para la funcional Z. Sin embargo, el Método Simplex una vez calculada una solución básica inicial, busca mediante cierto criterio, otras soluciones básicas que incrementen el valor de Z hasta alcanzar su valor óptimo.

1.3 Existencia de Soluciones Básicas Admisibles

En la sección anterior se hizo notar la estrecha relación entre puntos extremales de Ω y los conjuntos de m columnas linealmente independientes del sistema

(1.3.1) $$AX = b \ , \ X \geq 0$$

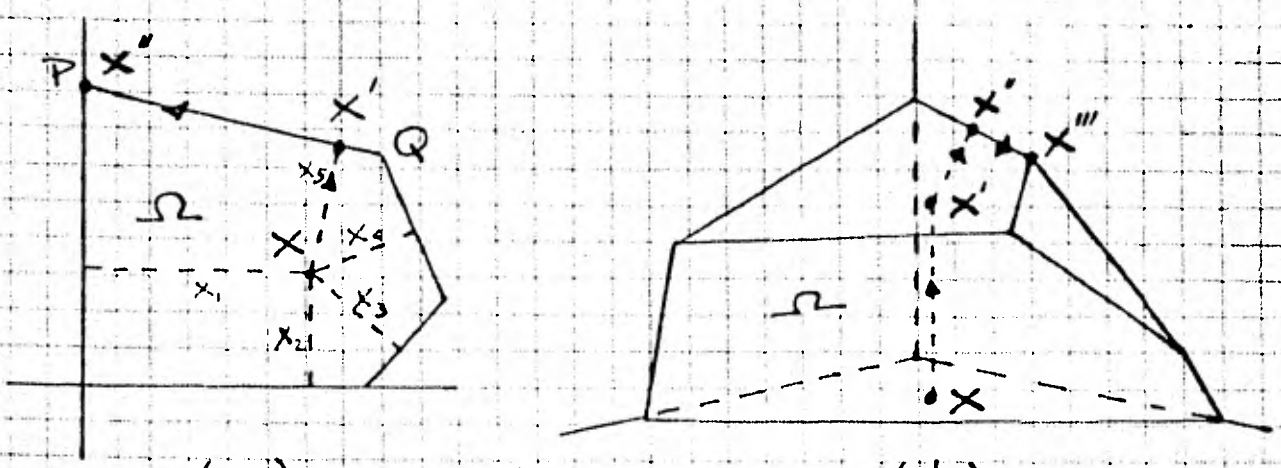
Entonces puede adoptarse cualquier punto de vista para probar que si hay una solución admisible para (1.3.1), esto es si $\Omega \neq \emptyset$, entonces hay una solución básica admisible.

Intuitivamente es inmediato advertir este hecho. Sea $X \in \Omega \neq \emptyset$, si X no es un punto extremal, entonces tiene más de m componentes positivas. Teniendo esta situación se puede partir y analizar cómo esta solución admisible puede ser "transformada" en básica admisible.

Supóngase que $X \in \Omega \subset \mathbb{R}^n$, este punto siempre puede ser proyectado a una cara de Ω; entendiendo por cara la intersección de Ω con una variedad

lineal de dimensión $(n-1)$ de Ω. Cuando se tiene esta situación, la componente correspondiente a esta restricción se anula. Denotemos por X' al punto así proyectado. Si se puede repetir esta operación, proyéctese X' en una variedad lineal de dim $(n-2)$ de Ω. De esta forma, si es posible, puede seguirse proyectando X hasta una variedad de dim (0). Cuando se dé esta situación, se estará en un punto extremal de Ω.

Se ilustran en la figura 10 los casos para $n=2$, y $n=3$.



(a) fig. 10 (b)

Dado $X \in \Omega \in \mathbb{R}^2$ (fig 10(a)), se proyecta sobre uno de sus lados, sea este PQ; de ahí se proyecta a P. Se ha llegado así

a un punto extremal de Ω (solución admisible básica) Es decir, el punto $X=(x_1,x_2,x_3,x_4,x_5)^T$ al ser proyectado en el lado PQ, su quinta componente se anula, i.e $X'=(x_1,x_2,x_3,x_4,0)^T$. En el siguiente paso, cuando se proyecta al punto P, otra componente se anula, de modo que $X''=(0,x_2,x_3,x_4,0)^T$ y éste es un punto extremal de Ω. El caso $n=3$ está ilustrado en la fig. 10(b).

La discusión anterior queda formalizada con el siguiente teorema.

Teorema. Dado el sistema de ecuaciones (1.3.1) con $A_{m\times n}$ y $Ran(A)=m$. Si hay una solución admisible, existe una solución admisible básica.

Demostración

Denotando por $a_1, a_2, \ldots, a_n$ las columnas de A y por $X=(x_1,x_2,\ldots,x_m)^T$ una solución admisible para (1.3.1). En estos términos

(1.3.2) $$\sum_{i=1}^{m} x_i a_i = b$$

Sin perder generalidad pueden suponerse que las primeras p-variables son positivas. Entonces

$$\sum_{i=1}^{p} x_i a_i = b \qquad (1.3.3)$$

Puesto que las columnas a_i, $i=1,\ldots,p$ pueden ser linealmente independientes (l.i.) o linealmente dependientes (l.d.) deben analizarse tales casos.

Caso 1. a_i, $i=1,\ldots,p$, son l.i. En este caso $p \leq m$. Si $p=m$, la solución es admisible básica y el teorema es cierto. Si $p<m$, entonces como $Ran(A)=m$, pueden seleccionarse de las restantes $(n-p)$ columnas, de modo que el conjunto resultante sea de m-columnas l.i., asignando a las restantes $(m-p)$ el valor cero. Se obtiene de esta manera una solución admisible básica.

Caso 2. a_i, $i=1,\ldots,p$ son l.d. Puesto que las columnas a_i para $i=1,\ldots,p$, son l.d., pueden encontrarse y_i, $i=1,\ldots,p$, donde al menos uno de ellos se supone positivo

Entonces

(1.3.4) $$\sum_{i=1}^{p} y_i a_i = 0$$

Multiplicando (1.4.4) por un escalar ε y sustrayéndolo a (1.3.4) se tiene

$$\sum_{i=1}^{p} (x_i - \varepsilon y_i) a_i = b$$

Denotando por $y = (y_1, \ldots, y_p, 0, \ldots, 0)$ para cualquier escalar ε

(1.3.5) $$x - \varepsilon y$$

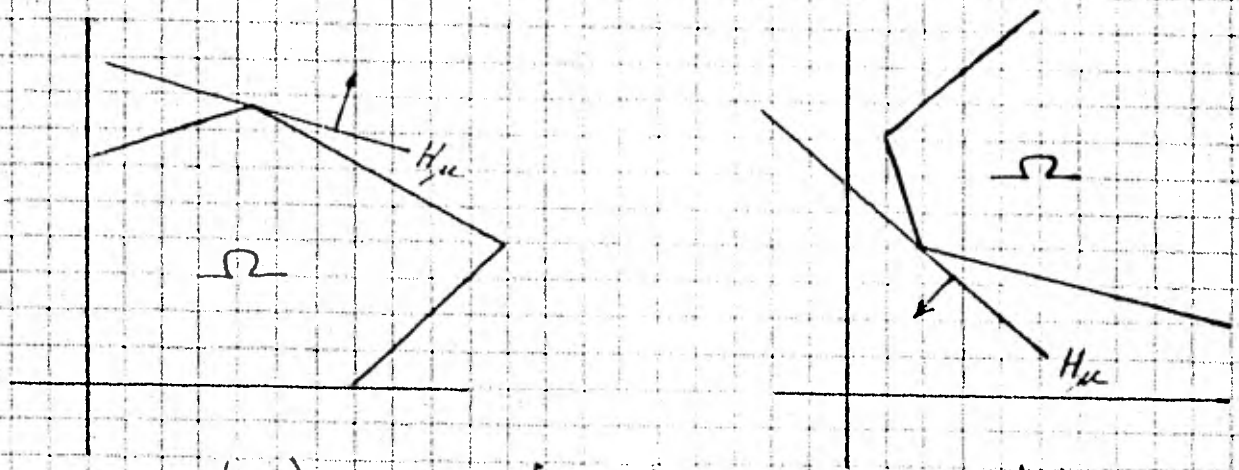
se tiene una solución. Para $\varepsilon = 0$, se tiene la solución original. Además como se supone la existencia de al menos un $y_i > 0$, $1 \leq i \leq p$, entonces al menos una componente decrecerá cuando ε se incremente. Incrementado ε hasta que una componente o más se anulen. Específicamente poniendo

$$\varepsilon = \min_i \{ x_i / y_i \mid y_i > 0 \}$$

Para este valor de ε, la solución dada por (1.3.5) es factible y con a lo más $(p-1)$-variables positivas. Repitiendo este proceso pueden seguirse eliminando variables positivas hasta tener un conjunto de columnas l.i. cayéndose así en el caso anterior ∎

1.4 Cambio de Soluciones Básicas

Bajo la hipótesis de existencia de solución, la funcional z alcanza su valor óptimo cuando de la familia H_z de hiperplanos, existe H_μ tal que H_μ es hiperplano soporte para Ω (fig. 11)



(a) fig. 11 (b)

Y como se hizo notar en la sección 1.1, si $\Omega \neq \phi$, entonces $T = H_\mu \cap \Omega \neq \phi$, y los puntos extremales de T son también de Ω. Así, si se tiene una solución admisible básica (ponto extremal de Ω), que no es óptima, ¿qué hacer para obtener una solución básica óptima?. El objetivo de esta sección es contestar a esta pregunta.

Antes de presentar un criterio que nos permita movernos sobre soluciones básicas admisibles y siempre acercarnos hacia una solución básica óptima, veamos cómo pasar de una solución básica a otra solución básica, en términos de submatrices básicas.

Supongamos que el problema

$$AX = b, \quad X \geq 0$$

se puede escribir como

$$[B|D]\begin{bmatrix} X_B \\ X_D \end{bmatrix} = b, \quad X_B \geq 0, X_D \geq 0,$$

siendo B una submatriz básica de A, correspondiente a la solución básica $\begin{bmatrix} \bar{X}_B \\ 0 \end{bmatrix}$ en donde $\bar{X}_B$ es tal que $B\bar{X}_B = b$. Pasar de esta solución básica a otra, corresponde a construir una nueva submatriz $\tilde{B}$, a partir de B.

Como $B = [b_1 | b_2 | \ldots | b_m]$ es una submatriz básica, el conjunto $\beta = \{b_1, b_2, \ldots, b_m\}$ es una base para $\mathbb{R}^m$. Ahora, sea a_j una columna de D que al expresarse en términos de la base β tenga al menos una

componente positiva. Esto es, si α es tal que

$$B\alpha = a_j$$

entonces $\alpha_i > 0$, para alguna i.

Multiplicando este último sistema por un escalar $\theta \geq 0$ y sustrayendo a $BX_B = b$, miembro a miembro se tiene

$$B(X_B - \theta\alpha) + \theta a_j = b$$

Así se habrá obtenido la solución admisible

$$\tilde{X} = (X_1 - \theta\alpha_1, X_2 - \theta\alpha_2, \ldots, X_m - \theta\alpha_m, 0, \ldots, \overset{j}{\theta}, \ldots, 0)^T$$

Como se requiere que esta solución sea admisible básica, entonces no debe tener más de m-componentes positivas, de modo que necesitamos hacer una igual a cero. Para ello basta tomar

$$\theta_0 = \min_i \{ X_i / \alpha_i \mid \alpha_i > 0 \}$$

Nótese que bajo la hipótesis de no existencia de soluciones básicas degeneradas solo hay una i para la que se tiene el mínimo. Sea ésta $i = k$. De esta forma

se tiene la solución

$$\tilde{X}_B = (X'_1, \ldots, X'_m)$$

donde

$$X'_i = X_i - \theta_0 \alpha_i \quad , i \neq k$$

$$X'_k = \theta_0$$

la cual satisface

$$\tilde{B}\, \tilde{X}_B = b$$

siendo

$$\tilde{B} = [b_1 \mid b_2 \mid \cdots \mid b_{k-1} \mid a_j \mid b_{k+1} \mid \cdots \mid b_m]$$

Es claro que $b \in \text{Cono}(\tilde{B})$. Así $\tilde{X}_B$ es una nueva solución admisible básica, si el conjunto de vectores columna de $\tilde{B}$ es linealmente independiente

Proposición. El conjunto $\{b_1, \ldots, b_{k-1}, a_j, b_{k+1}, \ldots, b_m\}$ es linealmente independiente.

Demostración

Demostremos esta proposición suponiendo lo contrario. Podemos entonces escribir

$$\beta_1 b_1 + \cdots + \beta_{k-1} b_{k-1} + \beta_k a_j + \beta_{k+1} b_{k+1} + \cdots + \beta_m b_m = 0$$

donde no todos los β_l son cero $(l = 1, \ldots, m)$.

Puesto que que el conjunto $\{b_1, \ldots, b_m\}$ es linealmente independiente (l.i.), entonces cualquier subconjunto de él es también l.i. Esto implica que $\beta_k \neq 0$. Así puede escribirse a_j como combinación lineal de los restantes b_j. Esto es

$$\gamma_1 b + \gamma_2 b_2 + \cdots + \gamma_m b_m = a_j$$

con $\gamma_i = \beta_i / \beta_k$, $i = 1, \ldots, m$.

Sustrayendo esta expresión de a_j al sistema $\mathcal{B}\alpha = a_j$ resulta

$$(\alpha_1 - \gamma_1) b_1 + \cdots + (\alpha_{k-1} - \gamma_{k-1}) b_{k-1} + (\alpha_{k+1} - \gamma_{k+1}) b_{k+1} + \cdots + (\alpha_m - \gamma_m) b_m + \alpha_k b_k = 0$$

Pero el conjunto $\{b_1, \ldots, b_m\}$ es l.i., lo cual implica que todos los coeficientes son cero y puesto que $\alpha_k > 0$, entonces la suposición de dependencia lineal de los $b_1, \ldots, b_{k-1}, a_j, b_{k+1}, \ldots, b_m$ nos lleva a una contradicción. Por tanto deben ser l.i. ∎

A continuación se presenta un criterio que caracteriza a las soluciones básicas óptimas. Para ello permitámonos escribir el problema

$$\text{Min } z = c^T x \tag{1.4.1}$$

$$\text{s.a. } Ax = b \;,\; x \geq 0$$

en la forma

$$\text{Min } z = c_B^T x_B + c_D^T x_D \tag{1.4.2}$$

$$\text{s.a } Bx_B + Dx_D = b \;;\; x_B \geq 0 \;,\; x_D \geq 0$$

la cual se obtiene al tomar

$$c = \begin{bmatrix} c_B \\ c_D \end{bmatrix} \;,\; x = \begin{bmatrix} x_B \\ x_D \end{bmatrix}, \text{ y } A = [B \mid D]$$

siendo B una submatriz básica de A.

De (1.4.2) se sigue que

$$x_B = B^{-1}b - B^{-1}Dx_D \;,$$

lo que permite escribir la funcional z en términos de x_D, quedando

$$z = c_B^T B^{-1} b + (c_D^T - c_B^T B^{-1} D) x_D$$

De aquí se sigue que si el así llamado vector de costos relativos

$$r^T = C_D^T - C_B^T B^{-1} D \geq 0 \ ,$$

entonces Z alcanza su valor mínimo en la solución básica

$$x = [\bar{x}_B \mid 0]^T .$$

donde $\bar{x}_B$ es tal que $B\bar{x}_B = b$. Es directo verificar que el inverso también es cierto. Luego se tiene establecido el criterio de optimalidad siguiente

Proposición. Para el problema (1.4.1) se tiene que $r^T = C_D^T - C_B^T B^{-1} D \geq 0$ si y sólo si la solución básica $\bar{x} = [\bar{x}_B \mid 0]$ es óptima.

Se sigue que si

$$r^T = C_D^T - C_B^T B^{-1} D$$

tiene alguna componente negativa, digamos r_j, entonces el valor de Z decrece al incrementar desde cero la correspondiente componente de x_D. Así, la correspondiente columna a_j de D es candidato a formar parte, reemplazando a b_k,

de la nueva submatriz básica $\tilde{B}$ con un correspondiente valor de z menor que $c_B^T B^{-1} b$, el valor de z correspondiente a la submatriz básica B.

El párrafo anterior sugiere de hecho una estrategia para movernos sobre soluciones básicas logrando siempre un decremento de la funcional z por cada cambio de submatriz básica. Por haber un número finito de submatrices (soluciones) básicas, tenemos que después de un número finito de ensayos alcanzamos una solución básica óptima.

1.5 Simplex Revisado

Todo el proceso del Método Simplex para calcular el mínimo de una funcional lineal sujeta también a restricciones lineales, puede ser resumido en los pasos siguientes

i) Calcular la solución básica x_B. Esto es, con la base B calcular

$$Bx_B = b$$

ii) Calcular los costos relativos $r^T = C_D^T - C_B^T B^{-1} D$ lo cual puede hacerse calculando primero

$$\lambda^T B = C_B^T$$

así que $r^T = C_D^T - \lambda^T D$ dá el criterio de optimalidad.

iii) Determinar el vector a_j que entra en la base, mediante la elección, por ejemplo del r_j más negativo y calcular

$$B\alpha = a_j$$

iv) Seleccionar θ_0 para determinar el vector b_k que deja la base

v) Actualizar la nueva base $\tilde{B}$ la cual está formada de los mismos vectores columna

de B, excepto por la k-ésima, pues en $\tilde{B}$ aparece a_j por b_k.

vi) Regresar al paso (i).

Como puede advertirse, las dificultades del método, desde un punto de vista numérico, se centran en la solución de los sistemas

$$Bx_B = b$$

(1.5.1)
$$\lambda^T B = C_B^T$$

$$B\alpha = a_j$$

por cada iteración del Método Simplex.

En el presente trabajo nos proponemos presentar una panorámica, desde un punto de vista numérico, de los aspectos relevantes que tienen que ver con la resolución numérica de los sistemas lineales anteriores conjuntamente con las dificultades técnicas que ello conlleva. Dificultades técnicas porque una característica muy notable de los problemas de Programación Lineal es que las restricciones dan lugar a matrices A

con dimensiones muy grandes y con muy pocos componentes diferentes de cero, es decir, dan lugar a matrices "ralas" (sparse en Inglés).

Este hecho es muy importante por la limitación física de la memoria de una computadora digital, y plantea problemas técnicos de almacenamiento y manejo de datos, pues solamente es necesario almacenar los componentes diferentes de cero. Los archivos donde se almacena la información dada y que se genera tienen problemas aparejados al dar y (o) mantener alguna estructura(*) de las matrices, y con ello aminorar los requerimientos de memoria al máximo posible conforme evoluciona el Simplex.

En el capítulo 2 se discuten los métodos numéricos de eliminación más populares en la Programación Lineal para resolver sistemas de ecuaciones lineales. En el capítulo 3, se discuten los métodos llamados de actua

(*) Más adelante, en las secciones 2.4.A, 2.4.B, 5.6.A y siguientes se mencionan algunas estructuras que poseen las matrices de ciertos problemas, y que conviene aprovechar, y también las estructuras deseables si no hay estructura alguna.

lización con el objetivo de aprovechar el trabajo realizado al resolver los sistemas (1.5.1), cuando se cambia de la submatriz básica B a $\tilde{B}$, pues éstas difieren sólo por una columna.

De los sistemas (1.5.1), el segundo merece atención especial pues cuando se soluciona consume alrededor del 40% del tiempo total por cada iteración del Método Simplex (ver Tomlin [61]). Por el momento sólo diremos que en las implementaciones comúnmente no se calcula, en vez de ello, se "actualiza". A la discusión relativa de estos aspectos está dedicado el capítulo 4.

En el capítulo 5, se presenta la versión de Saunders ([54], [55]), a la proposición del Simplex de Gill y Murray [27], la cual se basa en la resolución de los sistemas (1.5.1) mediante la aplicación de la descomposición QR de la matriz básica B.

2. Sistemas Sparse

Introducción

Como se apuntaba en el capítulo anterior, el Método Simplex para el problema

$$\text{Minimice } z = c^T x$$

$$\text{sujeto a } Ax = b,\ x \geq 0,$$

consta principalmente de los pasos siguientes:

PASO 1 Elegida la base B, calcúlese la solución básica correspondiente

$$B x_B = b$$

PASO 2 Se calculan los costos relativos

$$r^T = c_D^T - \lambda^T D,$$

donde $\lambda^T B = c_B^T$ (o bien $B^T \lambda = c_B$).
Si $r^T \geq 0$ entonces x_B es óptima. Si r^T tiene componentes negativas, entonces mediante algún criterio se elige la columna entrante a_j de entre las que tienen costo relativo $r_j < 0$.

PASO 3 Para determinar la columna de B saliente, se resuelve el sistema

$$B\alpha = a_j.$$

PASO 4 Actualizar la nueva base $\tilde{B}$ resultante del intercambio de una de sus columnas.

Luego, el Método Simplex, requiere por iteración, de la solución de tres sistemas, dos directos

$$B x_B = b \qquad y \qquad B\alpha = a_j$$

y uno transpuesto

$$B^T \lambda = c_B$$

En lo que sigue se hablará de cómo solucionar sistemas directos

$$Sx = b ,$$

donde S es una $m \times m$ matriz real y b un m vector real, de acuerdo a dos filosofías. La primera que opta por el cálculo de la inversa de S; y la segunda que calcula directamente la solución del sistema. Así que el objeto principal de este capítulo será exhibir algunos métodos computacionales para conseguir tanto S^{-1} como algunas de las factorizaciones de S para el cálculo de la solución del sistema.

2.1 Eliminación de Gauss-Jordan

La versión clásica del Simplex Revisado en su implementación práctica usa el método conocido como Eliminación de Gauss-Jordan. Este método calcula la inversa de la matriz S mediante operaciones elementales. A continuación se hace la descripción

2.1.A Descripción del Método

El sistema $SX = b$, se puede escribir como

$$
\begin{aligned}
& s_{11}^{(1)} X_1 + s_{12}^{(1)} X_2 + \cdots + s_{1m}^{(1)} X_m = b_1^{(1)} \\
& s_{21}^{(1)} X_1 + s_{22}^{(1)} X_2 + \cdots + s_{2m}^{(1)} X_m = b_2^{(1)} \\
& \vdots \\
& s_{m1}^{(1)} X_1 + s_{m2}^{(1)} X_2 + \cdots + s_{mm}^{(1)} X_m = b_m^{(1)}
\end{aligned}
\qquad (2.1.1)
$$

Para evitar entrar en detalles innecesarios se supondrá que la matriz S es no singular. Suponiendo $s_{11}^{(1)} \neq 0$, multiplíquese la primera ecuación por $1/s_{11}^{(1)}$. Si se multiplica esta ecuación por $-s_{i1}^{(1)}$ y se suma al i-ésimo renglón ($i = 2, \ldots, m$), se obtiene el sistema

$$
\begin{aligned}
&x_1 + s_{12}^{(2)} x_2 + \cdots + s_{1m}^{(2)} x_m = b_1^{(2)} \\
&\qquad s_{22}^{(2)} x_2 + \cdots + s_{2m}^{(2)} x_m = b_2^{(2)} \\
&\qquad\qquad \vdots \qquad\qquad\qquad \vdots \\
&\qquad s_{m2}^{(2)} x_2 + \cdots + s_{mm}^{(2)} x_m = b_m^{(2)}
\end{aligned}
\tag{2.1.2}
$$

De manera similar si $s_{22}^{(2)} \neq 0$, multiplíquese la segonda ecuación por $1/s_{22}^{(1)}$. Multiplicando esta ecuación por $-s_{i2}^{(2)}$ y sumando al i-ésimo renglón, para $i = 1, \ldots, m$ $(i \neq 2)$, se obtiene

$$
\begin{aligned}
x_1 \qquad & s_{13}^{(3)} x_3 + \cdots + s_{1m}^{(3)} x_m = b_1^{(3)} \\
x_2 + & s_{23}^{(3)} x_3 + \cdots + s_{2m}^{(3)} x_m = b_2^{(3)} \\
& s_{33}^{(3)} x_3 + \cdots + s_{3m}^{(3)} x_m = b_3^{(3)} \\
& \vdots \qquad\qquad\qquad \vdots \\
& s_{m3}^{(3)} x_3 + \cdots + s_{mm}^{(3)} x_m = b_m^{(3)}
\end{aligned}
$$

Continuando de esta manera, en el j-ésimo paso si $s_{jj}^{(j)} \neq 0$, se multiplica la j-ésima ecuación por $1/s_{jj}^{(j)}$. De nuevo, multiplicando esta Ec. por $-s_{ij}^{(j)}$ y sumando al i-ésimo renglón $(i \neq j)$ se obtiene

$$
\begin{aligned}
x_1 \qquad & s_{1,j+1}^{(j+1)} x_{j+1} + \cdots + s_{1m}^{(j+1)} x_m = b_1^{(j+1)} \\
& \vdots \qquad\qquad\qquad \vdots \\
x_j + & s_{j,j+1}^{(j+1)} x_{j+1} + \cdots + s_{jm}^{(j+1)} x_m = b_j^{(j+1)} \\
& \vdots \qquad\qquad\qquad \vdots \\
& s_{m,j+1}^{(j+1)} x_{j+1} + \cdots + s_{mm}^{(j+1)} x_m = b_m^{(j+1)}
\end{aligned}
$$

En el m-ésimo paso se habrá transformado el sistema (2.1.1) de tal modo que su solución se obtiene directamente de las relaciones

$$x_i = b_i^{(m+1)}, \quad i = 1, 2, \ldots, m.$$

2.1.B Forma Matricial

Antes de dar una descripción matricial del método de eliminación de Gauss-Jordan, tenemos la siguiente

Definición. Una matriz elemental E, es una matriz que difiere de la idéntica, en una de sus columnas

$$E = \begin{bmatrix} 1 & & \eta_1 & & \\ & \ddots & \vdots & & \\ & & \eta_j & & \\ & & \vdots & 1 & \\ & & \eta_m & & 1 \end{bmatrix}$$

donde los η_i $(i = 1, \ldots, m)$ no son todos cero. Esto es, las matrices elementales corresponden a aplicar a lo más m de las siguientes dos operaciones elementales de eliminación

a) Multiplicar un renglón (o columna) por

una constante, y

b) Multiplicar un renglón (o columna) por una constante y luego sumárselo a otro renglón (o columna), dependiendo si la matriz elemental E se aplica a una matriz S por la izquierda o por la derecha.

El método de Gauss-Jordan para resolver el sistema $Sx=b$ permite obtener la matriz inversa S^{-1} de S en forma de un producto de matrices elementales conocida como Forma Producto de la Inversa (F.P.I.). Estas matrices elementales cancelan las componentes s_{ij} fuera de la diagonal principal para llevarla a la matriz idéntica I_m de orden m. El algoritmo puede describirse en la siguiente forma

Denótese por $S^{(1)} = S$ $S^{(j)}$ a la matriz en el j-ésimo paso ($j=1,\ldots,m$) y $S^{(m+1)} = I_m$.

Para hacer ceros debajo de $s_{11}^{(1)}$ se multiplica a la matriz $S^{(1)}$ por la izquierda por una matriz elemental E_1

$$E_1 = \begin{bmatrix} \frac{1}{S_{11}^{(1)}} & 0 & \cdots & 0 \\ -\frac{S_{21}^{(1)}}{S_{11}^{(1)}} & 1 & & \vdots \\ \vdots & 0 & \ddots & \vdots \\ \vdots & \vdots & & 0 \\ -\frac{S_{n1}^{(1)}}{S_{11}^{(1)}} & 0 & \cdots & 1 \end{bmatrix}$$

De esta manera se obtiene $S^{(2)} = E_1 S^{(1)}$, resultando $S^{(2)}$ de la forma

$$\begin{bmatrix} 1 & S_{12}^{(2)} & \cdots & S_{1m}^{(2)} \\ 0 & S_{22}^{(2)} & \cdots & S_{2m}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & S_{n2}^{(2)} & \cdots & S_{nm}^{(2)} \end{bmatrix}$$

Igualmente para cancelar los $S_{ij}^{(2)}$ $(j \neq 2)$, se multiplica a la matriz $S^{(2)}$ por la izquierda por una matriz elemental E_2

$$E_2 = \begin{bmatrix} 1 & -\frac{S_{12}^{(2)}}{S_{22}^{(2)}} & 0 & \cdots & 0 \\ 0 & \frac{1}{S_{22}^{(2)}} & 0 & & \vdots \\ \vdots & \vdots & 1 & & \vdots \\ \vdots & \vdots & 0 & \ddots & \vdots \\ \vdots & \vdots & \vdots & & 0 \\ 0 & -\frac{S_{n2}^{(2)}}{S_{22}^{(2)}} & 0 & \cdots & 1 \end{bmatrix}$$

Se obtiene entonces $S^{(3)} = E_2 S^{(2)}$ de la forma

$$\begin{bmatrix} 1 & 0 & s_{13}^{(3)} & \cdots & s_{1m}^{(3)} \\ 0 & 1 & s_{23}^{(3)} & \cdots & s_{2m}^{(3)} \\ \vdots & 0 & s_{33}^{(3)} & \cdots & s_{3m}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & s_{m3}^{(3)} & \cdots & s_{mm}^{(3)} \end{bmatrix}$$

En el j-ésimo paso, $S^{(j)}$ es idéntica a I_m en sus primeras $(j-1)$ columnas. Su j-ésima columna en esta etapa es transformada en e_j con la aplicación de una matriz elemental E_j

$$E_j = \begin{bmatrix} 1 & 0 & \cdots & 0 & -\frac{s_{1j}^{(j)}}{s_{jj}^{(j)}} & 0 & \cdots & 0 \\ 0 & \ddots & & 0 & \vdots & & & \vdots \\ \vdots & & \ddots & 1 & & & & \\ & & 0 & \frac{1}{s_{jj}^{(j)}} & 0 & & & \vdots \\ & & & \vdots & 1 & \ddots & & \\ & & & & 0 & & \ddots & 0 \\ 0 & \cdots & 0 & -\frac{s_{mj}^{(j)}}{s_{jj}^{(j)}} & 0 & \cdots & 0 & 1 \end{bmatrix}$$

Así se obtiene $S^{(j+1)} = E_j S^{(j)}$ de la forma

$$\begin{bmatrix} 1 & 0 & \cdots & 0 & s_{1,j+1}^{(j+1)} & \cdots & s_{1m}^{(j+1)} \\ 0 & \ddots & & 0 & & & \\ \vdots & & \ddots & 0 & & & \\ & & & 1 & s_{j,j+1}^{(j+1)} & \cdots & s_{jm}^{(j+1)} \\ & & & 0 & s_{j+1,j+1}^{(j+1)} & & \\ & & & \vdots & & & \\ 0 & \cdots & 0 & & s_{m,j+1}^{(j+1)} & \cdots & s_{mm}^{(j+1)} \end{bmatrix}$$

Al ejecutar la m-ésima etapa del algoritmo se obtiene I_m. Por consiguiente si

$$S^{(1)} = S \quad ; \quad S^{(m+1)} = I_m \text{, entonces}$$

$$I_m = S^{(m+1)} = E_m S^{(m)}$$
$$= E_m E_{m-1} \cdots E_1 S^{(1)}$$

lo cual nos proporciona la Forma Producto de la Inversa. Esto es

$$S^{-1} = E_m E_{m-1} \cdots E_1$$

También se obtiene que $X = b^{(m+1)}$ es la solución del sistema $Sx = b$, donde

$$b^{(m+1)} = E_m E_{m-1} \cdots E_1 b^{(1)}$$

con $b^{(1)} = b$.

2.1.C Consideraciones Numéricas

Obsérvese que al eliminar los componentes $s_{ij}^{(j)}$ encima y bajo la diagonal principal de S en cada paso del método, si se aplica tal cual, no siempre llegará a su término, además si ésto no ocurre se tiene el peligro siempre latente de acarrear grandes errores de redondeo. Veamos por qué.

Consideremos la matriz

$$S = \begin{bmatrix} 1 & -1 & 4 \\ -1 & 1 & 0 \\ 0 & 1 & 2 \end{bmatrix}$$

Como $s_{11} \neq 0$ podemos empezar el proceso de eliminación, i.e.,

$$S_1 = \begin{bmatrix} 1 & -1 & 4 \\ 0 & 0 & 4 \\ 0 & 1 & 2 \end{bmatrix}$$

Sin embargo no puede continuarse porque $s_{22}=0$. Esta dificultad desaparece si se permutan el tercer renglón con el segundo. En este ejemplo era evidente que $s_{22}=0$, pero en otros

muchos casos debemos decidir que un pivote es "casi cero", o puede considerársele como tal. Para mostrar lo desastroso que puede resultar el método si no se toma ésto en consideración tomemos el siguiente ejemplo (Ver Forsythe - Moler [22])

$$0.000100X_1 + 1.00X_2 = 1.00$$
$$1.00\quad X_1 + 1.00X_2 = 2.00$$

Su solución es

$$X_1 = 1.00010 \approx 1.0$$
$$X_2 = \emptyset.99990 \approx 1.0$$

Tomemos la matriz aumentada de este sistema

$$S = \begin{bmatrix} 0.000100 & 1.00 & 1.00 \\ 1.00 & 1.00 & 2.00 \end{bmatrix}$$

Entonces

$$S_1 = \begin{bmatrix} 1.00 & 100000.00 & 100000.00 \\ 0.00 & (1.00-100000.00) & (2.00-100000.00 \end{bmatrix}$$

Si se tiene en cuenta que una máquina digital trabaja con precisión finita, entonces

$$S_1 \approx \begin{bmatrix} 1.00 & 100000.00 & 100000.00 \\ 0.00 & -100000.00 & -100000.00 \end{bmatrix}$$

$$S_2 \approx \begin{bmatrix} 1.00 & 0.00 & 0.00 \\ 0.00 & 1.00 & 1.00 \end{bmatrix}$$

Por consiguiente la solución es

$$x_1 = 0.00 \quad ; \quad x_2 = 1.00$$

En cambio si consideramos que s_{11} es muy pequeño, entonces permutamos renglones y efectuamos la eliminación, i.e.,

$$S = \begin{bmatrix} 1.00 & 1.00 & 2.00 \\ 0.000100 & 1.00 & 1.00 \end{bmatrix}$$

$$S_1 = \begin{bmatrix} 1.00 & 1.00 & 2.00 \\ 0.00 & (1.00-0.000100) & (1.00-0.000200) \end{bmatrix} =$$

$$= \begin{bmatrix} 1.00 & 1.00 & 2.00 \\ 0.00 & 0.999900 & 0.999800 \end{bmatrix} \approx$$

$$\approx \begin{bmatrix} 1.00 & 1.00 & 2.00 \\ 0.00 & 1.00 & 1.00 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} 1.00 & 0.00 & 1.00 \\ 0.00 & 1.00 & 1.00 \end{bmatrix}$$

Por tanto

$$x_1 = 1.00 \quad ; \quad x_2 = 1.00$$

lo cual concuerda con la solución dada.

Los anteriores ejemplos sugieren que lo ideal para amortiguar los efectos del redondeo sería tener la opción de elegir el máximo $S_{ij}^{(j)}$ como pivote en cada paso de la eliminación, pero desafortunadamente esto no es posible. Esta consideración es fácil de explicar.

Por hipótesis el sistema (2.1.1) es no degenerado de modo que debe existir algún $S_{ij}^{(j)} \neq 0$, para alguna i entre j y m, y así permutar los renglones correspondientes para tomarlo como pivote. Como puede apreciarse, las oportunidades de elegir los mejores pivotes a medida que progresa la eliminación, se reducen hasta hacerse nulas en el último paso.

El número de operaciones aritméticas que este método de eliminación requiere son

m	recíprocos
$\frac{1}{2}m^3 + m^2 - \frac{1}{2}m$	multiplicaciones
$\frac{1}{2}m^3 - \frac{1}{2}m$	adiciones

Para el cálculo de estas fórmulas véase por ejemplo Fox [24]

Un aspecto notable de los problemas de Programación Lineal es el que se refiere a la gran cantidad de componentes iguales a cero es decir, en su formulación sólo una parte mínima es diferente de cero, y resulta que en el proceso de solución se "crean" nuevos componentes diferentes de cero. Este fenómeno debe tenerse en cuenta pues los problemas en su mayoría son de grandes dimensiones y un crecimiento excesivo del número de sus componentes puede rebasar la capacidad de memoria de una computadora digital. Para visualizar esta situación tomemos el siguiente ejemplo Orchard-Hays [43]

El número de operaciones aritméticas que este método de eliminación requiere son

m	recíprocos
$\frac{1}{2}m^3 + m^2 - \frac{1}{2}m$	multiplicaciones
$\frac{1}{2}m^3 - \frac{1}{2}m$	adiciones

Para el cálculo de estas fórmulas véase por ejemplo Fox [24]

Un aspecto notable de los problemas de Programación Lineal es el que se refiere a la gran cantidad de componentes iguales a cero es decir, en su formulación sólo una parte mínima es diferente de cero, y resulta que en el proceso de solución se "crean" nuevos componentes diferentes de cero. Este fenómeno debe tenerse en cuenta pues los problemas en su mayoría son de grandes dimensiones y un crecimiento excesivo del número de sus componentes puede rebasar la capacidad de memoria de una computadora digital. Para visualizar esta situación tomemos el siguiente ejemplo Orchard-Hays [43]

$$S = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Calculando la inversa de S directamente obtenemos

$$S^{-1} = \begin{bmatrix} 1/2 & 1/2 & -1/2 & 1/2 & -1/2 \\ -1/2 & 1/2 & 1/2 & -1/2 & 1/2 \\ 1/2 & -1/2 & 1/2 & 1/2 & -1/2 \\ -1/2 & 1/2 & -1/2 & 1/2 & 1/2 \\ 1/2 & -1/2 & 1/2 & -1/2 & 1/2 \end{bmatrix}$$

En cambio aplicando F. P. I. para el mismo cálculo de S^{-1}, esto es, la inversa de S obtenida en forma producto, observaremos cómo la creación de nuevos componentes diferentes de cero es menor.

Recordemos que la inversa de una matriz S en forma producto es

$$S^{-1} = E_m E_{m-1} \cdots E_1$$

donde

$$E_j = I_m + (\eta_j - e_j) e_j^T \quad , \quad j = 1, \ldots, m$$

es una matriz elemental.

Obsérvese que cada E_j queda completamente definida por el vector η_j. Ahora bien, si queremos aplicar F. P. I. usando una computadora podemos entonces operar y almacenar solamente los vectores η_j, pues I_m no necesita ser almacenada. Además las transformaciones E_j necesarias para obtener en forma producto S^{-1} pueden ser almacenadas en lugar de las columnas de la matriz S. Es decir, la transformación E_1 (el vector η_1 que la define), toma el lugar de la columna s_1; E_2 el lugar de s_2, y así sucesivamente Los vectores columna η_j del ejemplo de arriba serán entonces

$$\eta_1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}; \quad \eta_2 = \begin{bmatrix} 0 \\ 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}; \quad \eta_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \\ 0 \end{bmatrix}; \quad \eta_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}; \quad \eta_5 = \begin{bmatrix} -1/2 \\ 1/2 \\ -1/2 \\ 1/2 \\ 1/2 \end{bmatrix}$$

Nótese que en forma producto sólo necesitamos de 13 localidades, en contra de las 25 que requiere la inversa S^{-1} en forma explícita.

Es inmediato advertir las ventajas prácticas de este esquema de eliminación pues la inversa de S, S^{-1} calculada mediante F.P.I. es, en general, menos densa y más fácil de manejar que la S^{-1} calculada explícitamente. Es por ello que se plantea como una alternativa para usarse en la solución de problemas de Programación Lineal.

Esquemáticamente F.P.I. es como sigue

$j = 1, \ldots, m$

$\eta_{jj} \leftarrow 1 / s_{jj}$

$i = 1, \ldots, m$

$\eta_{ij} \leftarrow -s_{ij} / s_{jj} \quad (i \neq j)$

(i)

Esta parte corresponde al cálculo de E_j para ser aplicada a la izquierda de la matriz S

$i = 1, \ldots, m$

$\eta_{ij} \leftarrow 0$

$k = 1, \ldots, m$

$\eta_{ij} \leftarrow \eta_{ik} s_{kj} + \eta_{ij}$

(k)

(i)

Se ha calculado el producto $E_j \cdot S$

$\mu \leftarrow 0$

$i = 1, \ldots, m$

$k = 1, \ldots, m$

$\mu \leftarrow \hat{\eta}_{ik} b_k + \mu$ ①

$b_i \leftarrow \mu$ ②

La misma transformación E_j se aplica al lado derecho del sistema para obtener directamente la solución

$x_j \leftarrow b_j$ ③

2.2 Eliminación de Gauss.

Uno de los métodos más populares para el cálculo de soluciones para sistemas de ecuaciones lineales es el método de Eliminación Gaussiana, el cual se ha adaptado al Método Simplex como una alternativa a la solución de los sistemas (2.1.1). Básicamente el método esta dividido en dos partes: Eliminación hacia adelante y Sustitución hacia atrás. En la primera parte, se transforma la matriz de coeficientes mediante operaciones elementales en una matriz triangular superior U; en la segunda, dado que el último renglon de U consta de ceros excepto su última componente, puede calcularse inmediatamente X_m, sostituir este valor en las ecuaciones anteriores y obtener fácilmente X_{m-1}, y así sucesivamente. La descripción de este método puede hacerse del modo siguiente.

2.2.A Descripción del Método

Como se hizo con el método de Gauss-Jordan dado el sistema

$$
\begin{aligned}
s_{11}^{(1)}x_1 + s_{12}^{(1)}x_2 + \cdots + s_{1m}^{(1)}x_m &= b_1^{(1)} \\
s_{21}^{(1)}x_1 + s_{22}^{(1)}x_2 + \cdots + s_{2m}^{(1)}x_m &= b_2^{(1)} \\
&\vdots \\
s_{m1}^{(1)}x_1 + s_{m2}^{(1)}x_2 + \cdots + s_{mm}^{(1)}x_m &= b_m^{(1)}
\end{aligned}
$$

Se multiplica la primera ecuación por $1/s_{11}^{(1)}$, si $s_{11}^{(1)} \neq 0$, multiplicando esta Ec. por $-s_{i1}^{(1)}$ y sumando al i-ésimo renglón ($i = 2, \ldots, m$) se obtiene el mismo sistema que en (2.1.2). En el siguiente paso, la segunda ecuación se multiplica por $1/s_{22}^{(2)}$, si $s_{22}^{(2)} \neq 0$. Si se multiplica esta Ec. por $-s_{i2}^{(2)}$ y se suma al i-ésimo renglón ($i = 3, \ldots, m$), se obtiene

$$
\begin{aligned}
x_1 + s_{12}^{(3)}x_2 + s_{13}^{(3)}x_3 + \cdots + s_{1m}^{(3)}x_m &= b_1^{(3)} \\
x_2 + s_{23}^{(3)}x_3 + \cdots + s_{2m}^{(3)}x_m &= b_2^{(3)} \\
s_{33}^{(3)}x_3 + \cdots + s_{3m}^{(3)}x_m &= b_3^{(3)} \\
&\vdots \\
s_{m3}^{(3)}x_3 + \cdots + s_{mm}^{(3)}x_m &= b_m^{(3)}
\end{aligned}
$$

En el j-ésimo paso, la j-ésima ecuación se multiplica por $1/s_{jj}^{(j)}$. Se multiplica esta Ec. por $-s_{ij}^{(j)}$ y se suma al i-ésimo renglón $i = j+1, \ldots, m$, para obtener el sistema

$$
\begin{aligned}
x_1 + s_{12}^{(j+1)} x_2 + \cdots + s_{1j}^{(j+1)} x_j + s_{1,j+1}^{(j+1)} x_{j+1} + \cdots + s_{1m}^{(j+1)} x_m &= b_1^{(j+1)} \\
x_2 + \cdots + s_{2j}^{(j+1)} x_j + s_{2,j+1}^{(j+1)} x_{j+1} + \cdots + s_{2m}^{(j+1)} x_m &= b_2^{(j+1)} \\
\vdots \qquad\qquad \vdots \qquad\qquad \vdots \qquad\qquad & \vdots \\
x_j + s_{j,j+1}^{(j+1)} x_{j+1} + \cdots + s_{jm}^{(j+1)} x_m &= b_j^{(j+1)} \\
\vdots \qquad\qquad \vdots \qquad\qquad & \vdots \\
s_{m,j+1}^{(j+1)} x_{j+1} + \cdots + s_{mm}^{(j+1)} x_m &= b_m^{(j+1)}
\end{aligned}
$$

Continuando el mismo proceso, en la m-ésima etapa se habrá conseguido un sistema triangular superior que es fácilmente soluble. Así la última ecuación resulta ser

$$x_m = b_m^{(m+1)}$$

Este valor de x_m es sustituído en todas las ecuaciones anteriores y se obtiene nuevamente un sistema triangular, ahora, de orden $(m-1)$

$$
\begin{aligned}
x_1 + s_{12}^{(m+1)} x_2 + \cdots + s_{1,m-1}^{(m+1)} x_{m-1} &= b_1^{(m+1)} \\
x_2 + \cdots + s_{2,m-1}^{(m+1)} x_{m-1} &= b_2^{(m+1)} \\
\vdots \qquad\qquad & \vdots \\
x_{m-1} &= b_{m-1}^{(m+1)}
\end{aligned}
$$

De manera análoga se pueden ir obteniendo las x_i $(i = m-1, \ldots, 1)$. Esta parte es la que se denomina Sustitución hacia atrás.

2.2-B Forma Matricial

Este método de eliminación nos permite obtener la inversa de S como un producto de matrices elementales conocido por: Forma Eliminación de la Inversa F.E.I.

El sistema $Sx=b$ como hemos visto, es transformado en uno equivalente pero más fácil de solucionar. Las dos etapas principales de este método son ejecutadas mediante la aplicación de operaciones elementales ejecutadas como multiplicaciones con matrices elementales. Estas matrices, en la primera etapa, permiten llevar la matriz S a una triangular superior U; y en la segunda, la obtención de la solución.

Sean $S^{(1)}=S$; $S^{(j)}$ en el j-ésimo paso y $S^{n+1}=U$. La forma de cancelar los $S_{ij}^{(1)}$ debajo de $S_{11}^{(1)}$ es mediante la aplicación a S de la matriz elemental L_1

$$L_1=\begin{bmatrix} \frac{1}{S_{11}^{(1)}} & 0 & \cdots & \cdots & 0 \\ -\frac{S_{21}^{(1)}}{S_{11}^{(1)}} & 1 & \cdots & \cdots & \vdots \\ \vdots & 0 & \ddots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \ddots & 0 \\ -\frac{S_{n1}^{(1)}}{S_{11}^{(1)}} & 0 & \cdots & 0 & 1 \end{bmatrix}$$

Así se obtiene $S^{(2)} = L_1 S^{(1)}$ que es de la forma

$$\begin{bmatrix} 1 & s_{12}^{(2)} & \cdots & s_{1m}^{(2)} \\ 0 & s_{22}^{(2)} & \cdots & s_{2m}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & s_{m2}^{(2)} & \cdots & s_{mm}^{(2)} \end{bmatrix}$$

Análogamente pueden eliminarse los $s_{2j}^{(2)}$, $j=3,\ldots,m$ de $S^{(2)}$ con la aplicación por la izquierda de la matriz elemental L_2

$$L_2 = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \frac{1}{s_{22}^{(2)}} & & & \\ \vdots & -\frac{s_{32}^{(2)}}{s_{22}^{(2)}} & 1 & & \vdots \\ \vdots & 0 & & & 0 \\ 0 & -\frac{s_{m2}^{(2)}}{s_{22}^{(2)}} & 0 & \cdots & 1 \end{bmatrix}$$

Con lo que se obtiene $S^{(3)} = L_2 S^{(2)}$ de la forma

$$\begin{bmatrix} 1 & s_{12}^{(3)} & s_{13}^{(3)} & \cdots & s_{1m}^{(3)} \\ 0 & 1 & s_{23}^{(3)} & \cdots & s_{2m}^{(3)} \\ \vdots & 0 & s_{33}^{(3)} & \cdots & s_{3m}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & s_{m3}^{(3)} & \cdots & s_{mm}^{(3)} \end{bmatrix}$$

En el j-ésimo paso $S^{(j)}$ será una matriz triangular en sus primeras $(j-1)$ columnas. En esta etapa, la eliminación de las componentes $s_{jk}^{(j)}$ para $k=j+1,\dots,m$, se efectúa con la matriz elemental L_j

$$L_j = \begin{bmatrix} 1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \ddots & & & & \vdots \\ \vdots & & 1 & & & \\ & & 0 & \frac{1}{s_{jj}^{(j)}} & & \\ & & & -\frac{s_{j+1,j}^{(j)}}{s_{jj}^{(j)}} & 1 & \\ & & & \vdots & & \ddots \\ 0 & \cdots & 0 & -\frac{s_{mj}^{(j)}}{s_{jj}^{(j)}} & 0 \cdots 0 & 1 \end{bmatrix}$$

Obteniéndose $S^{(j+1)} = L_j \cdot S^{(j)}$ de la forma

$$\begin{bmatrix} 1 & s_{12}^{(j+1)} & \cdots & s_{1,j+1}^{(j+1)} & \cdots & s_{1m}^{(j+1)} \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & & & \\ & & 1 & s_{j,j+1}^{(j+1)} & \cdots & s_{jm}^{(j+1)} \\ & & 0 & s_{j+1,j+1}^{(j+1)} & \cdots & s_{j+1,m}^{(j+1)} \\ & & & \vdots & & \vdots \\ 0 & \cdots & 0 & s_{m,j+1}^{(j+1)} & \cdots & s_{mm}^{(j+1)} \end{bmatrix}$$

En la m-ésima etapa se obtiene la matriz triangular superior U. Por tanto

$$S^{(m+1)} = L_m S^{(m)} = L_m L_{m-1} \cdots L_1 S_1^{(1)}$$

Escribiendo $\hat{L} = L_m L_{m-1} \cdots L_1$, se tiene

$$S^{(m+1)} = \hat{L} S^{(1)}$$

y puesto que $S^{(1)} = S$; $S^{(m+1)} = U$, entonces

$$U = \hat{L} S$$

En vista de que las transformaciones L_j $j = 1, \ldots, m$ son aplicadas en ambos miembros del sistema $SX = b$, se sigue que éste se transforma en el sistema equivalente

$$UX = \hat{L} b^{(m)} \qquad (LSX = Lb)$$

Resolver la última ecuación de este sistema es lo mismo que transformar la última columna de U en el vector e_m. Esto es, eliminar las componentes sobre la diagonal principal. Esto se efectúa con la aplicación, por la izquierda, a U de la matriz elemental U_m

$$U_m = \begin{bmatrix} 1 & 0 & \cdots & 0 & -u_{1,m} \\ 0 & \ddots & & & \vdots \\ \vdots & & \ddots & & \\ & & & 0 & -u_{m-2,m} \\ & & & 1 & -u_{m-1,m} \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

De esta manera se obtiene $U_m U$

$$\begin{bmatrix} 1 & u_{12} & \cdots & u_{1,m-1} & 0 \\ 0 & \ddots & & \vdots & \vdots \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & u_{m-2,m-1} & 0 \\ \vdots & & & 1 & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

(Por comodidad de notación, en lugar de escribir s_{ij} $(j>i)$ los elementos de U, éstos serán denotados por u_{ij})

En el siguiente paso, se eliminan las componentes sobre la diagonal de la penúltima columna, obteniéndose el vector e_{m-1}. Esto con la aplicación de U_{m-1} a la izquierda de $U_m U$.

$$U_{m-1} = \begin{bmatrix} 1 & 0 & \cdots & 0 & -u_{1,m-1} & 0 \\ 0 & \ddots & & & \vdots & \vdots \\ \vdots & \ddots & \ddots & & \vdots & \vdots \\ \vdots & & 0 & & \vdots & \vdots \\ \vdots & & & \ddots & -u_{m-2,m-1} & 0 \\ \vdots & & & & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

Así $U_{m-1} U_m U$ es de la forma

$$\begin{bmatrix} 1 & u_{12} & \cdots & u_{1,m-1} & 0 & 0 \\ 0 & \ddots & & & \vdots & \vdots \\ \vdots & \ddots & \ddots & & \vdots & \vdots \\ & & & u_{m-1,m-1} & \vdots & \vdots \\ & & & 0 & \vdots & \vdots \\ & & & 1 & 0 & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & \end{bmatrix}$$

Antes de ejecutar el j-ésimo paso, la matriz $\tilde{U} = U_{j-1} \cdots U_{m-1} U_m U$ es idéntica a I_m en sus últimas $(m-j+1)$ columnas, en esta etapa se transforma su j-ésima columna en el vector e_j con la matriz elemental U_j, i.e.

$$U_j = \begin{bmatrix} 1 & 0 & \cdots & 0 & -u_{1j} & 0 & \cdots & 0 \\ 0 & \ddots & & 0 & \vdots & & & \vdots \\ \vdots & & \ddots & & \vdots & & & \vdots \\ & & & & -u_{j-1,j} & & & \vdots \\ & & & & 0 & & & \vdots \\ & & & & 1 & \ddots & & \vdots \\ & & & & & & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & 1 & \end{bmatrix}$$

Se obtiene entonces $U_j \cdots U_m U$ de la forma

$$\begin{bmatrix} 1 & u_{12} & \cdots & u_{1,j-1} & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & & \vdots & \vdots & & & \vdots \\ \vdots & \ddots & \ddots & u_{j-2,j-1} & 0 & & & \vdots \\ & & & 1 & \ddots & & & \vdots \\ & & & & \ddots & \ddots & & \vdots \\ & & & & & \ddots & & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 & 1 \end{bmatrix}$$

Por tanto, al aplicar la transformación U_2 a $U_3 \cdots U_m U$, obtendremos I_m. Esto es

$$I_m = U_2 U_3 \cdots U_m U$$

De modo que

$$U^{-1} = U_2 U_3 \cdots U_m$$

Resumiendo, las dos etapas principales de la Eliminación Gaussiana: Eliminación hacia adelante y Sustitución hacia atrás, nos permitieron llevar el sistema $Sx = b$, a uno triangular superior

$$Ux = \hat{L} b^{(m)}$$

en donde $\hat{L} = L_m \cdot L_{m-1} \cdots L_1$

Por consiguiente

$$x = U^{-1} \hat{L} b^{(m)}$$
$$= U_2 \cdots U_{m-1} U_m L_m L_{m-1} \cdots L_1 b^{(m)}$$

En conclusión, hemos obtenido S^{-1} de la forma

$$S^{-1} = U_2 \cdots U_{m-1} U_m L_m L_{m-1} \cdots L_1$$

expresión conocida como: Forma Eliminación de la Inversa F.E.I., donde S^{-1} es expresada como un producto de $(m-1)$ matrices triangulares superiores U_k y m triangulares inferiores L_k.

2.2.C Consideraciones Numéricas

Respecto al proceso de eliminación hacia adelante, las observaciones hechas en F.P.I. siguen siendo válidas en F.E.I. Es decir, si no se pivotea en cada paso de la eliminación, los errores de redondeo podrán ser tales que los resultados no serán confiables...

A continuación haremos algunos comentarios sobre la estabilidad numérica de los métodos de Eliminación Gaussiana y de Gauss-Jordan para resolver el problema

$$Sx = b \tag{2.2.1}$$

La estabilidad numérica del método de Eliminación Gaussiana con pivoteo parcial ha sido demostrada (ver por ejemplo Wilkinson [63], Stewart [56]). En tal análisis se prueba que la solución obtenida x_c con tal método es solución exacta de un sistema vecino

$$(S+E)x = b$$

"vecino" en el sentido de que para E se tiene que

$$\frac{\|E\|}{\|S\|} \leq f(m)\, g\, \beta^{-t} \tag{2.2.2}$$

donde f es una función polinomial de grado chico; g es conocido como el factor de crecimiento el cual está dado por

$$g = \frac{\max\{g_k\}}{g_1}, \quad k = 1, \ldots, m$$

donde

$$g_k = \max\{|s_{ij}^{(k)}|\}, \; k = 1, \ldots, m \; ; \; g_1 = \max\{|s_{ij}|\} \; \forall i,j$$

Esto es g es el cociente de la componente más grande de $S^{(j)}$, $j = 1, \ldots, m$, y la componente más grande de S; y el último factor β es relativo a la precisión del sistema numérico de punto flotante con base β y t-dígitos en la mantisa.

Aunque teóricamente el número g puede llegar a alcanzar el valor de 2^{m-1}, ver Wilkinson [63], en la práctica sin embargo se comporta como un número del orden de la unidad; de manera similar $f(m)$ difícilmente supera a m. De modo que (2.2.2) puede quedar como

$$\frac{\|E\|}{\|S\|} \leq m\beta^{-t}$$

Si calculamos la norma del error relativo, ver Forsythe, Moler [22], Stewart [56] por ejem., obtenemos

$$\frac{\|x_c - x\|_\infty}{\|x\|_\infty} \leq \frac{m\beta^{-t}k(S)}{(1 - m\beta^{-t}k(S))} \leq \frac{10}{9} m\beta^{-t}k(S)$$

donde $K(S) = \|S\|_\infty \|S^{-1}\|_\infty$, siempre que $K(S)$ sea tal que $m\beta^{-t}K(S) \approx 1/10$.

Vemos que la calidad en la exactitud de la solución calculada por el método de Eliminación Gaussiana, depende fundamentalmente del número de condición $K(S)$ de S. Para más detalles véase a Forsythe, Moler [22] y o Stewart [56].

Por otro lado, la norma del residual satisface

$$\|r\|_\infty = \|b - Sx_c\|_\infty = \|Ex_c\|_\infty \leq \|E\|_\infty \|x_c\|_\infty \leq$$

$$\leq m\beta^{-t}\|S\|_\infty \|x_c\|_\infty$$

Es decir hemos encontrado una cota para $\|r\|_\infty$, la cual depende de la norma de la solución calculada y no del número de condición $K(S)$ de S; y por lo tanto no de la exactitud de x_c.

Como veremos más adelante cuando hablemos del llenado que ocurre en la matriz S al tratar de resolver (2.2.1), la diferencia que hay al calcularla mediante F.P.I. o F.E.I.,

está en la forma de calcular S^{-1}. Es decir, en F.P.I. la inversa S^{-1} de S se obtiene como un producto de matrices elementales E_j, $j=1,\ldots,m$. Esto es

$$S^{-1} = E_m E_{m-1} \cdots E_1$$

en tanto que en F.E.I., se obtiene en dos etapas; en la primera, se obtiene una matriz triangular superior U con la aplicación de m-matrices elementales L_i a S (eliminación hacia adelante), y posteriormente se aplican $(m-1)$ matrices elementales U_j (sustitución hacia atrás). Esto es

$$S^{-1} = U_2 \cdots U_{m-1} U_m L_m L_{m-1} \cdots L_1$$

De modo que la diferencia entre los dos métodos para la obtención de S^{-1} está en la cancelación de los componentes de la parte triangular superior de S. Es decir, la diferencia está en calcular la solución de un sistema de la forma

$$Ux = d$$

Analicemos las dos formas para resolver

está en la forma de calcular S^{-1}. Es decir, en F.P.I. la inversa S^{-1} de S se obtiene como un producto de matrices elementales E_j, $j=1,\ldots,m$. Esto es

$$S^{-1} = E_m E_{m-1} \cdots E_1$$

en tanto que en F.E.I., se obtiene en dos etapas; en la primera, se obtiene una matriz triangular superior U con la aplicación de m-matrices elementales L_i a S (eliminación hacia adelante), y posteriormente se aplican $(m-1)$ matrices elementales U_j (sustitución hacia atrás). Esto es

$$S^{-1} = U_2 \cdots U_{m-1} U_m L_m L_{m-1} \cdots L_1$$

De modo que la diferencia entre los dos métodos para la obtención de S^{-1} está en la cancelación de los componentes de la parte triangular superior de S. Es decir, la diferencia está en calcular la solución de un sistema de la forma

$$Ux = d$$

Analicemos las dos formas para resolver

este sistema triangular. Para ello, antes veamos que, como ya se ha anotado antes, (Forsythe, Moler [22]), empleando Eliminación Gaussiana, la solución calculada es solución exacta de un sistema vecino

$$(U + \delta U)x_c = d + \delta d$$

en el sentido de que $\frac{\|\delta U\|}{\|U\|} \leq m\beta^{-t}$

en cambio, con el método de Gauss-Jordan la solución calculada no resulta ser la solución exacta de un sistema vecino, sino que cada componente $\hat{x}^{(j)}$ de x_c resulta ser la j-ésima componente de la solución exacta de un sistema vecino

$$(U + \delta U^{(j)})x_c^{(j)} = d + \delta d^{(j)}$$

para cada j, entendiendo por vecino que

$$|\delta U^{(j)}| \leq m\beta^{-t}|U| \; ; \; |\delta d^{(j)}| \leq m\beta^{-t}|d|$$

Con lo anterior ya podemos decir que mientras con el método de Eliminación Gaussiana se tienen cotas a posteriori para el residual, en cambio con el método de Gauss-Jordan no hay tales.

Veamos un ejemplo (Peters y Wilkinson [46])

$$U = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ & \varepsilon_2 & 1 & 1 & 1 \\ & & \varepsilon_3 & 1 & 1 \\ & & & \varepsilon_4 & 1 \\ & & & & 1 \end{bmatrix}$$

En el segundo paso de la eliminación del método de Gauss-Jordan, la matriz U tiene la forma

$$\begin{bmatrix} 1 & 0 & 0 & (\varepsilon_2\varepsilon_3)^{-1} & (\varepsilon_2\varepsilon_3)^{-1} \\ & \varepsilon_2 & 0 & \varepsilon_3^{-1} & \varepsilon_3^{-1} \\ & & \varepsilon_3 & 1 & 1 \\ & & & \varepsilon_4 & 1 \\ & & & & 1 \end{bmatrix}$$

Como puede apreciarse, en el siguiente paso al eliminar los componentes de U sobre ε_4 la última columna de U sería de la forma

$$u_5 = [(\varepsilon_2\varepsilon_3\varepsilon_4)^{-1}, (\varepsilon_3\varepsilon_4)^{-1}, \varepsilon_4^{-1}, 1, 1]^T$$

en la cual al efectuar la eliminación sobre u_{55} pudieran causar problemas algunos de sus componentes.

Podemos decir por consiguiente que la solución de un sistema $Sx=b$, por Eliminación Gaussiana es solución exacta de un sistema vecino $(S+E)x=b$, lo cual no es válido para el método de Gauss-Jordan. Sin embargo podemos ver que la cota del error relativo a priori para la solución calculada $\tilde{x}_c$ por Gauss-Jordan es

$$\frac{\|\tilde{x}_c - x\|_\infty}{\|x\|_\infty} \leq \frac{2m\beta^{-t}k(S)}{1-m\beta^{-t}k(S)}$$

que esencialmente es la obtenida por Eliminación Gaussiana. Podemos decir que para el método de Gauss-Jordan no es posible dar una cota a posteriori del residual $r=b-S\tilde{x}_c$, pues $\tilde{x}_c$ como ya se dijo, no resulta ser solución de un cierto sistema vecino.

La situación es la misma cuando se requiere el cálculo de la inversa S^{-1} de S con los dos métodos

La aritmética involucrada en F.E.I. es

m recíprocos

$1/3\,m^3 + m^2 - 1/3\,m$ multiplicaciones

$1/3\,m^3 + 1/2\,m^2 - 5/6\,m$ adiciones

Esto muestra que F.E.I requiere menos aritmética que F.P.I. (Compárese con la tabla corresp.).

2.2.D Consideraciones Sparse

Un aspècto de gran importancia que debe considerarse al resolver el sistema

$$Sx=b$$

es el "llenado" de las matrices S.

En la sección anterior vimos que respecto al proceso de eliminación hacia adelante (Es decir en la eliminación de la parte triangular inferior de S), tanto F.P.I. como F.E.I. eran idénticos; en cambio en la eliminación de la parte triangular superior eran diferentes. Veremos aquí que se tiene la misma situación respecto al llenado.

Puesto que los vectores ζ_k que definan a las matrices elementales U_k están formados con los componentes diferentes de cero de U, se tiene entonces que en la sustitución hacia atrás de F.E.I no se crean nuevos componentes diferentes de cero. Esto se verá más adelante. De modo que los únicos componentes creados serán en el

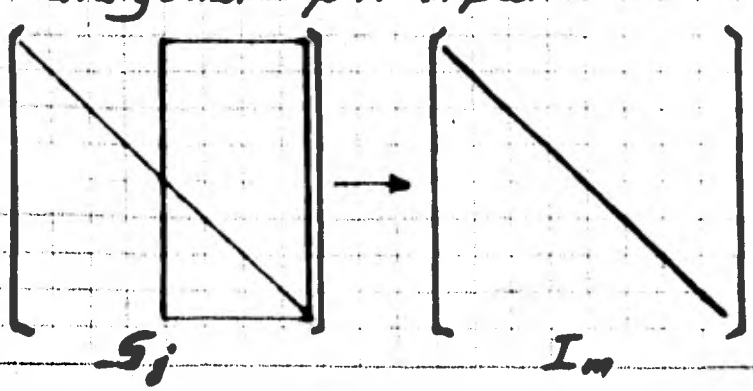
proceso de eliminación hacia adelante. El llenado, consecuentemente, también resulta ser menor que en F.P.I.

Como puede advertirse, este método de eliminación supera en forma significativa a F.P.I., en cuanto a que: requiere de menos aritmética, y tiende a ser menor el llenado de la matriz S^{-1}!

En los esquemas de eliminación previamente descritos para la solución del sistema $Sx=b$, pueden hacerse las observaciones siguientes.

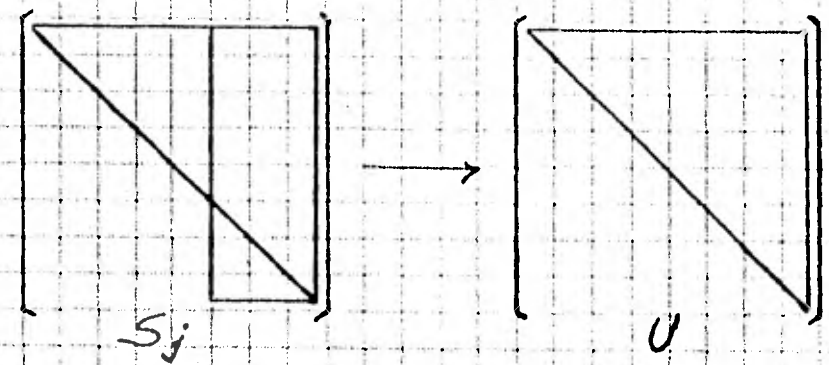
OBSERVACION 1

Tanto en F.P.I. como en F.E.I., la consecución de I_m se realiza mediante operaciones elementales aplicadas a la matriz S. En el primero, la obtención de I_m se realiza de una vez, es decir, haciendo ceros bajo y sobre la diagonal principal

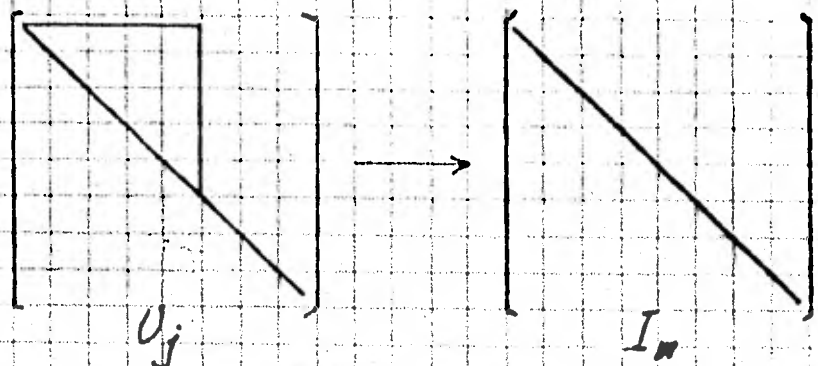


Mientras que en el esquema de F.E.I. es como sigue

Eliminación hacia adelante



Eliminación hacia atrás



De modo que si nos restringimos a la eliminación de la parte triangular inferior, ambos esquemas resultan equivalentes. Por tanto la segonda observación.

OBSERVACION 2

La eliminación de los S_{ij} $(i<j)$ para ambos esquemas es diferente, pues en F.P.I se realiza de la segunda columna a la mésima; en cambio, en F.E.I se realiza en orden inverso: de la mésima columna a la segunda (la primera por construcción es e_1).

En lo que sigue nos avocamos a discutir la obtención de U^{-1} para cada uno de los esquemas F. P. I. y F. E. I.

Sea U la parte triangular superior de S

$$U = \begin{bmatrix} 1 & u_{12} & u_{13} & \cdots & u_{1m} \\ 0 & 1 & u_{23} & \cdots & u_{2m} \\ \vdots & \ddots & \ddots & & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & u_{m-1,m} \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

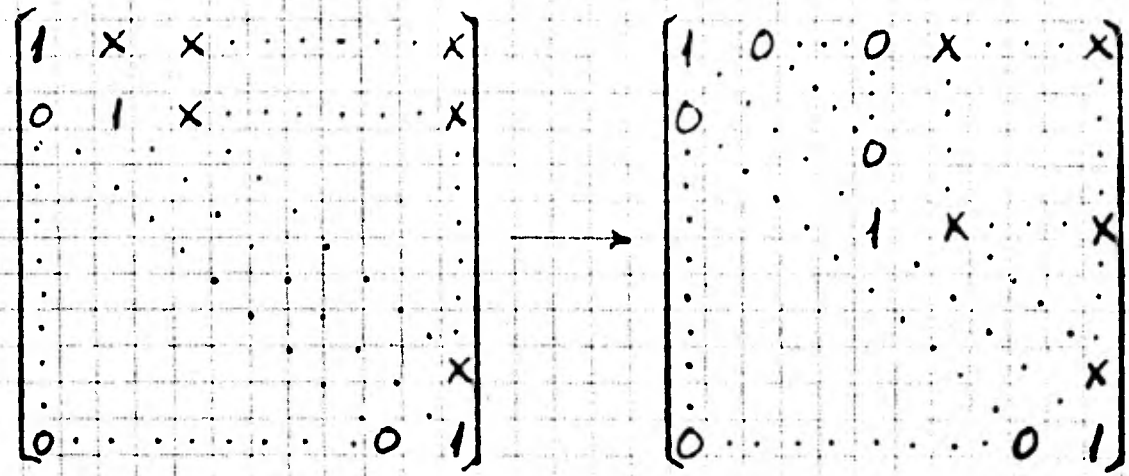
Como decíamos, para invertir U tenemos dos opciones: hacer la eliminación de los u_{ij}, a partir de la segunda columna, i.e.

$$(A) \qquad \tilde{U}_m^{-1} \tilde{U}_{m-1}^{-1} \cdots \tilde{U}_2^{-1} U$$

donde cada $\tilde{U}_j^{-1}$, $j = 2, \ldots, m$ está dada por

$$U_j^{-1} = I_m - \tilde{\zeta}^j e_j^T$$

con $\tilde{\zeta}_i^j = -u_{ij}$, $i < j$; $\tilde{\zeta}_i^j = 0$, $i \geq j$



F.P.I.

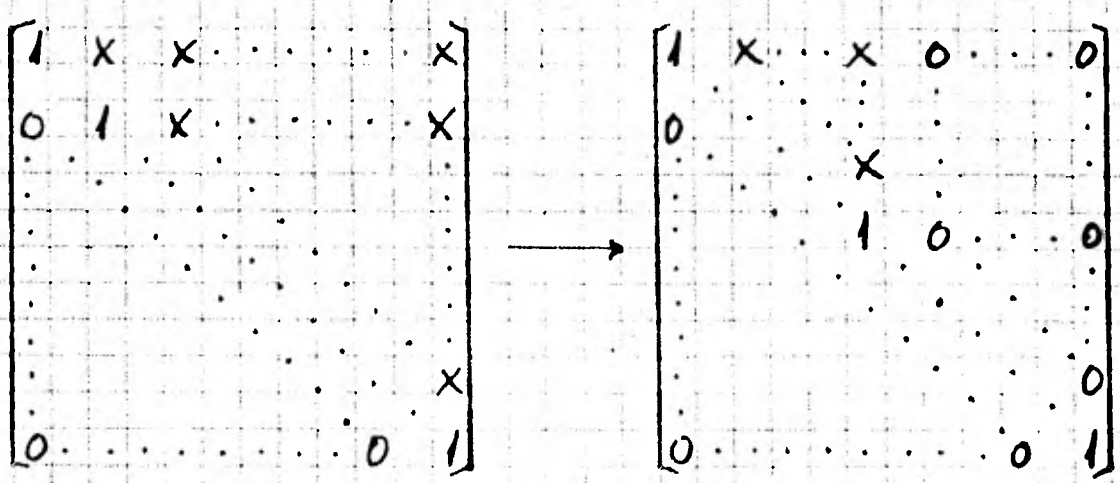
O bien de la última columna a la segunda

$$U_2^{-1} \cdots U_{m-1}^{-1} U_m^{-1} U$$

donde las U_j^{-1}, $j = m, m-1, \ldots, 2$ dadas por

$$U_j^{-1} = I_m - \zeta^j e_j^T$$

con $\zeta_i^j = -u_{ij}$, $i < j$; $\zeta_i^j = 0$, $i > j$



F.E.I.

Analicemos estos dos esquemas. Tomemos primero el siguiente ejemplo. Sea U la matriz de orden 6.

$$U = \begin{bmatrix} 1 & u_{12} & u_{13} & 0 & 0 & u_{16} \\ 0 & 1 & 0 & 0 & u_{25} & 0 \\ \cdot & \cdot & 1 & u_{34} & 0 & u_{36} \\ \cdot & \cdot & \cdot & 1 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & 1 & u_{56} \\ 0 & \cdot & \cdot & \cdot & \cdot & 1 \end{bmatrix}$$

Como cada U_j queda completamente definida por el vector ζ^j, entonces podemos resumir la serie de transformaciones (A) aplicadas a U en la siguiente tabla

ζ_2	ζ_3	ζ_4	ζ_5	ζ_6
$-u_{12}$	$-u_{13}$	$-\tilde{u}_{14}$	$-\tilde{u}_{15}$	$-\tilde{u}_{16}$
1	0	0	$-u_{25}$	$-\tilde{u}_{26}$
	1	$-u_{34}$	0	$-u_{36}$
		1	0	0
			1	$-u_{56}$
				1

Nota: los componentes $\tilde{u}_{ij}$ de ζ_j de esta tabla son en general no-ceros.

Para las transformaciones (B) su tabla es

ζ_2	ζ_3	ζ_4	ζ_5	ζ_6
$-u_{12}$	$-u_{13}$	0	0	$-u_{16}$
1	0	0	$-u_{25}$	0
	1	$-u_{34}$	0	$-u_{36}$
		1	0	0
			1	$-u_{56}$
				1

Es inmediato observar de las dos tablas que en la segunda, los vectores ζ_j tienen tantos elementos diferentes de cero como hay en U, en cambio, en la primera pueden aparecer otros más. Es decir no hay llenado en la matriz U^{-1} al efectuar la eliminación de la última columna a la segunda (F.E.I.), en contra de la primera opción. Este fenómeno del llenado en U^{-1} es típico para el esquema F.P.I.

Para ilustrar con más detalle escogemos los renglones k y $(j+1)$-ésimos ($k \leq j$) de alguna matriz triangular superior U.

u_k : 0 ··· 0 1 0 ··· 0 X_α 0 X 0 0 X 0 X

u_{j+1} : 0 ··· ··· ··· 0 1 x' 0 x' 0 x x' 0

Para eliminar X_α $(1 \le \alpha \le j)$ de u_k debemos sumar $-X_\alpha(u_{j+1})$-veces a u_k. Esto es

u_k : 0 ··· 0 1 0 ··· 0 $\boxed{0}$ x' x x' 0 $\hat{x}$ x' x

u_{j+1} : 0 ··· ··· ··· 0 1 x' 0 x' 0 x' x' 0

Entonces de cuatro elementos diferentes de cero que había en u_k resultan después de la operación seis elementos diferentes de cero.

En el esquema F.E.I., igualmente tomemos dos renglones u_k y u_j $(k > j)$

u_k : 0 ··· 0 1 X ··· X X x_α ··· 0

u_j : 0 ··· ··· ··· 0 1 0 ··· 0

Al eliminar el elemento X_α del renglón u_k sumamos $-X_\alpha(u_j)$-veces a u_k, i.e.

u_k : 0 ··· 0 1 X ··· X $\boxed{0}$ 0 ··· 0

u_j : 0 ··· ··· ··· 0 1 0 ··· 0

Por tanto podemos enunciar el siguiente

Teorema. La inversa de U en F.E.I. es menos densa que en F.P.I.

Para más detalles réase a Brayton [8]

En resumen, desde el punto de vista del número de operaciones aritméticas, de la estabilidad numérica, tenemos que el método de F.E.I. es ligeramente superior que el método de F.P.I., sin embargo Dantzing en un trabajo realizado con Harvey, R.P., McKnight, R.D. y Smith, S.S. (ver Tewarson [59], pág. 140), comonica que en la práctica F.E.I es claramente superior a F.P.I. Por otro lado tenemos que para F.E.I. hag una estimación del error relativo a posteriori de la solución calculada en términos del residual, pues la solución calculada resulta ser la solución de un sistema recino, cosa que en general no es posible para F.P.I.

2.3 Descomposición LU

Consideremos un método para solucionar sistemas de ecuaciones lineales equivalente al bien conocido de Eliminación Gaussiana.

Este método consiste escencialmente en descomponer la matriz de coeficientes en dos matrices triangulares, una inferior L y una superior U.

Antes de exhibir este método demos la siguiente definición.

Definición. Una matriz triangular inferior $L = l_{ij}$ es una matriz cuadrada cuyos $l_{ij} = 0$ para $i < j$; análogamente se define una triangular superior $U = u_{ij}$, sólo que en ésta sus $u_{ij} = 0$ para $i > j$.

Proposición. i) El producto de matrices triangulares superiores (inferiores) es también una matriz triangular superior (inferior)

ii) La inversa, si existe, de una matriz triangular superior (inferior) es igualmente triangular superior (inferior).

El esquema que a continuación se describe transforma, mediante operaciones elementales, a la matriz S en una triangular superior U, tal como en la sección anterior. Esto es, con matrices elementales L_j, $j=1,\ldots,m$, se obtiene

$$L_m L_{m-1} \cdots L_1 S = U$$

Las matrices elementales, en este caso, son triangulares inferiores. De la proposición anterior se sigue que $L_m L_{m-1} \cdots L_1 = \hat{L}$ es triangular inferior. Por consiguiente

$$\hat{L} S = U$$

Como cada L_j $(j=1,\ldots,m)$ es no singular, $\hat{L}$ por tanto es no singular así que

$$S = LU \qquad (\text{con } L = \hat{L}^{-1})$$

expresión conocida como la descomposición L-U de S.

De este modo el sistema $Sx=b$ se puede escribir como

$$LUx = b$$

Haciendo $y = Ux$ tenemos entonces que resolver $Sx = b$ es lo mismo que resolver

$$Ly = b$$
$$Ux = y$$

En estos dos sistemas la solución es inmediata, pues ambas matrices L y U son triangulares. En $Ly = b$, la primera ecuación sólo involucra la incógnita y_1, en la segunda y_1, y_2, etc. Análogamente en el sistema $Ux = y$, pero en orden inverso.

En resumen, si y es solución del sistema triangular

$$Ly = b$$

entonces la solución buscada se obtendrá resolviendo

$$Ux = y$$

Es importante hacer notar que no se requiere efectuar aritmética para calcular L a partir de $L_1 L_2 \cdots L_m$.

<u>Proposición</u>. Sean L_i, L_j dos factores de L. Entonces

$$L_i L_j = I_m + (\xi^{(i)} - e_i) e_i^T + (\xi^{(j)} - e_j) e_j^T, \quad i < j$$

Demostración

$$L_i L_j = [I_n + (\xi^{(i)} - e_i) e_i^T][I_n + (\xi^{(j)} - e_j) e_j^T]$$
$$= I_n + (\xi^{(i)} - e_i) e_i^T + (\xi^{(j)} - e_j) e_j^T + (\xi^{(i)} - e_i) e_i^T (\xi^{(j)} - e_j) e_j^T$$

Del segundo miembro de esta igualdad, consideremos el último término de la manera siguiente

$$(\xi^{(i)} - e_i)[e_i^T (\xi^{(j)} - e_j)] e_j^T$$

La expresión encerrada en los paréntesis [...] es cero ya que $i < j$, i.e.

$$[0 \cdots 0 \underset{i}{1} 0 \cdots 0 \underset{j}{\cdots} 0] \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \xi_j - 1 \\ \xi_{j+1} \\ \vdots \\ \xi_n \end{bmatrix} \begin{matrix} \\ \\ \leftarrow i \\ \\ \\ \leftarrow j \\ \\ \\ \end{matrix} = 0$$

Por tanto

$$L_i L_j = I_n + (\xi^{(i)} - e_i) e_i^T + (\xi^{(j)} - e_j) e_j^T \quad \blacksquare$$

Corolario. La matriz $L = L_1 \cdots L_{n-1} L_n$ está dada por

$$L = I_n + \sum_{j=1}^{n} (\xi^{(j)} - e_j) e_j^T$$

Este resultado nos dice que L está formada por la yuxtaposición de los vectores

columna que definen a cada L_k, $k=1,\dots,m$. Visualcemos ésto con dos matrices L_i, L_j, $1 \le i < j \le m$.

$$L_i = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & \xi_i^{(i)} & & & \\ & & \xi_{i+1}^{(i)} & 1 & & \\ & & \vdots & & \ddots & \\ & & \xi_m^{(i)} & & & 1 \end{bmatrix}, \quad L_j = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & \ddots & & & \\ & & & 1 & & \\ & & & \xi_j^{(j)} & & \\ & & & \xi_{j+1}^{(j)} & 1 & \\ & & & \vdots & & \ddots \\ & & & \xi_m^{(j)} & & 1 \end{bmatrix}$$

$$L_i L_j = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & \xi_i^{(i)} & & & \\ & & \xi_{i+1}^{(i)} & \cdots & \xi_j^{(j)} & \\ & & \vdots & & \xi_{j+1}^{(j)} & 1 \\ & & \xi_m^{(i)} & & \vdots & \ddots \\ & & & & \xi_m^{(j)} & 1 \end{bmatrix}$$

2.4 Descomposición LU para Matrices Ralas.

La necesidad de emplear permutaciones, como decíamos anteriormente, viene de la posibilidad de que algún $S_{jj}=0$ y debido a ello no poder continuar con el proceso de eliminación; otra razón de su empleo era la búsqueda de los mejores pivotes para amortiguar los errores por redondeo. Aquí veremos un tercer aspecto: evitar el llenado de la matriz S

Este aspecto es de gran importancia si recordamos que los sistemas de ecuaciones en Programación Lineal, la matriz posee muy pocos componentes diferentes de cero. Por eso, si se requiere efectuar la descomposición triangular de S, nos interesa que los factores L, U también tengan pocos componentes diferentes de cero. Veamos el siguiente ejemplo.

Supongamos un sistema de ecuaciones lineales cuya matriz S es la siguiente.

$$S=\begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} & s_{15} & s_{16} \\ s_{21} & s_{22} & 0 & \cdots & \cdots & 0 \\ s_{31} & 0 & s_{33} & \ddots & & \vdots \\ s_{41} & \vdots & \ddots & s_{44} & \ddots & \vdots \\ s_{51} & \vdots & & \ddots & s_{55} & 0 \\ s_{61} & 0 & \cdots & \cdots & 0 & s_{66} \end{bmatrix}$$

Al eliminar los s_{i1}, $i=2,\ldots,6$ nos resulta

$$S'=\begin{bmatrix} 1 & s'_{12} & s'_{13} & \cdots & \cdots & s'_{16} \\ 0 & s'_{22} & s'_{23} & \cdots & \cdots & s'_{26} \\ \vdots & \vdots & \vdots & & & \vdots \\ \vdots & \vdots & \vdots & & & \vdots \\ \vdots & \vdots & \vdots & & & \vdots \\ 0 & s'_{62} & s'_{63} & \cdots & \cdots & s_{66} \end{bmatrix}$$

y de aquí en adelante la eliminación de los s_{ij}, $i>j$ se tendrá que hacer con una matriz, en general densa. Por lo que los factores L y U serán asimismo densos.

$$L=\begin{bmatrix} \xi_{11} & 0 & \cdots & \cdots & \cdots & 0 \\ \xi_{21} & \xi_{22} & \ddots & & & \vdots \\ \vdots & \vdots & \ddots & & & \vdots \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & \vdots & \vdots & 0 & \\ \xi_{61} & \xi_{62} & \cdots & \cdots & \xi_{66} & \end{bmatrix}, \quad U=\begin{bmatrix} 1 & u_{12} & u_{13} & \cdots & \cdots & u_{16} \\ 0 & 1 & u_{23} & \cdots & \cdots & u_{26} \\ \vdots & \vdots & \ddots & & & \vdots \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & u_{65} \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

En cambio si permutamos la primera columna a la última posición y el primer renglón hasta el fondo antes de iniciar el proceso. Esto es

$$S' = \begin{bmatrix} s'_{11} & 0 & \cdots & 0 & s'_{16} \\ 0 & s'_{22} & & & s'_{26} \\ \vdots & & s'_{33} & & s'_{36} \\ & & & s'_{44} & 0 \quad s'_{46} \\ 0 & \cdots & 0 & s'_{55} & s'_{56} \\ s'_{61} & s'_{62} & s'_{63} & s'_{64} & s'_{65} \quad s'_{66} \end{bmatrix}$$

Los correspondientes factores L y U serán

$$L = \begin{bmatrix} \xi_{11} & 0 & \cdots & 0 \\ 0 & \xi_{22} & & \\ \vdots & & \ddots & \\ 0 & \cdots & 0 & 0 \\ \xi_{61} & \xi_{62} & \cdots & \xi_{66} \end{bmatrix}, \quad U = \begin{bmatrix} 1 & 0 & \cdots & 0 & u_{16} \\ 0 & & & & u_{26} \\ \vdots & & \ddots & & \vdots \\ & & & 0 & \\ & & & 1 & u_{56} \\ 0 & \cdots & & 0 & 1 \end{bmatrix}$$

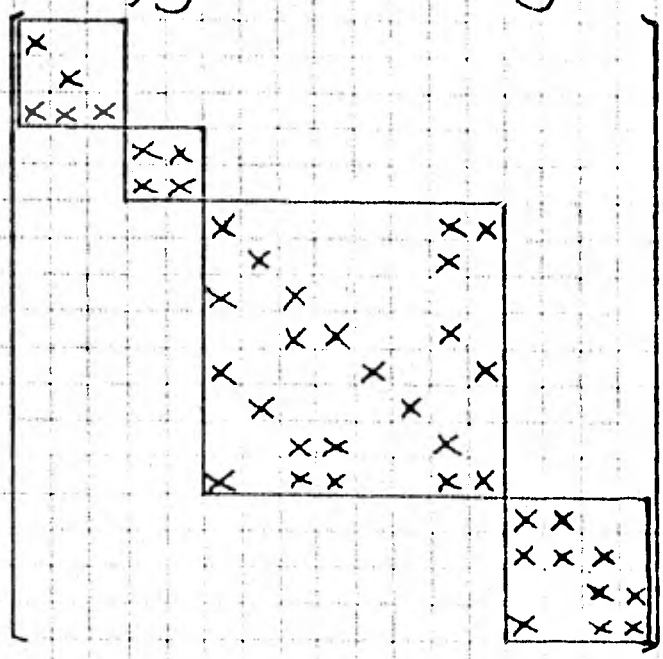
Resulta obvia la ventaja si se emplean permutaciones para conservar la baja densidad de la matriz.

Con el mismo propósito analicemos una descomposición de la matriz de carácter muy elemental

pero básica en la cual descansan muchas otras.

Cuando se tiene un sistema de ecuaciones muy grande, los requerimientos de memoria son un serio obstáculo para solucionarlo, sin embargo si es posible permutar los elementos de la matriz de modo que se le pueda llevar a una forma de diagonal por bloques, tal como aparece en la figura de abajo,

$S =$



entonces la solución de todo el sistema se reduce ahora a solucionar pequeños bloques. En otros términos, en lugar de manejar toda la matriz se toma uno de los bloques y una vez resuelto se almacena (en disco o en cinta),

se accesa otro bloque de la matriz y así sucesivamente las partes restantes. Con ésto además de superar las limitaciones de memoria, se logra disminuir el llenado de la matriz, pues es fácil de advertir que éste se restringe solamente a los bloques.

Considerando algono de los bloques de S y efectuando la eliminación de los s_{ij} $(i>j)$, puede observarse cómo se dá este llenado.

Supóngase que se está en el bloque S_j de S, el cual tiene dos espigas

$$S_j = \begin{bmatrix} s_{11} & & & & & s_{16} \\ s_{21} & s_{22} & & & s_{25} & \\ & s_{32} & s_{33} & & & \\ s_{41} & & & s_{44} & & \\ s_{51} & s_{52} & & & s_{55} & \\ s_{61} & & & & & s_{66} \end{bmatrix}$$

↑ Espigas

Los vectores $\xi^{(j)}$ de las matrices L_j, $j=1,\ldots,6$ para hacer ceros bajo la diagonal principal de S_j quedan como sigue

$\xi^{(1)}$	$\xi^{(2)}$	$\xi^{(3)}$	$\xi^{(4)}$	$\xi^{(5)}$	$\xi^{(6)}$
$\frac{1}{s_{11}}$					
$-\frac{s_{21}}{s_{11}}$	$\frac{1}{s_{22}}$				
0	$-\frac{s_{32}}{s_{22}}$	$\frac{1}{s_{33}}$			
$-\frac{s_{41}}{s_{11}}$	0	0	$\frac{1}{s_{44}}$		
$-\frac{s_{51}}{s_{11}}$	$-\frac{s_{52}}{s_{22}}$	$-\frac{s_{53}}{s_{33}}$	$-\frac{s_{54}}{s_{44}}$	$\frac{1}{s_{55}}$	
$-\frac{s_{61}}{s_{11}}$	0	0	0	0	$\frac{1}{s_{66}}$

La matriz triangular superior U_j resulta ser entonces

$$\begin{bmatrix} 1 & & & & & u_{16} \\ & 1 & & & & u_{25}u_{26} \\ & & 1 & u_{34} & u_{35} & u_{36} \\ & & & 1 & & u_{46} \\ & & & & 1 & u_{56} \\ & & & & & 1 \end{bmatrix}$$

Puede verse que el llenado en U_j tiene lugar principalmente en las dos últimas dos columnas (espigas) de S_j.

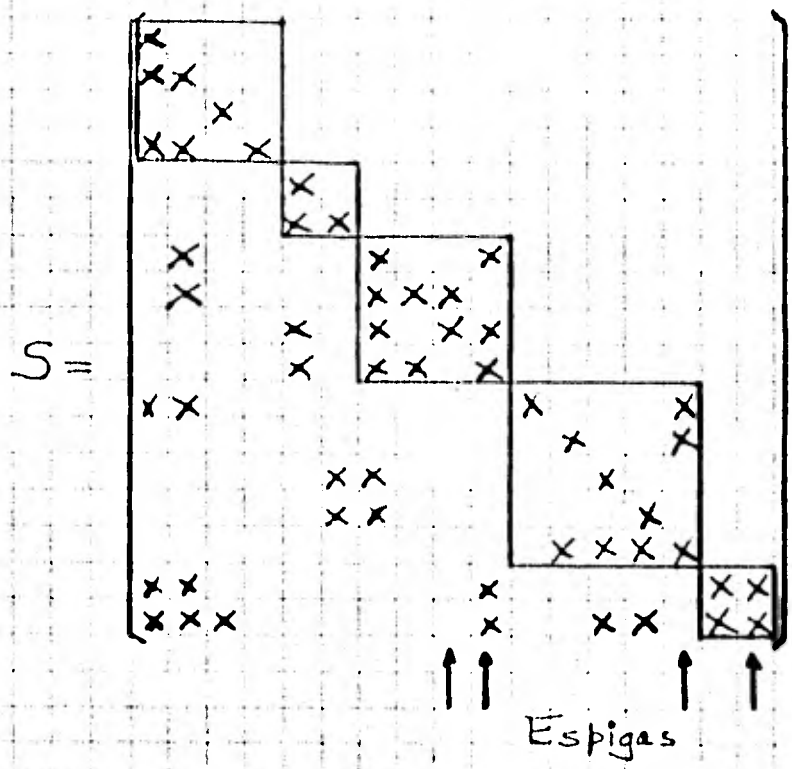
Por supuesto no siempre es posible descomponer, mediante permutaciones una matriz dada en una diagonal por bloques, así que deben buscarse otras alternativas.

2.4.A Esquema de Saunders

Un esquema de propósito general dado por M. A. Saunders, consiste en llevar la matriz dada a una triangular inferior por bloques. Cada bloque es asimismo triangular inferior, aunque algunas columnas pueden tener componentes sobre la diagonal principal; es decir, bloques con "espigas". Se prefiere que las espigas estén lo más a la derecha de cada bloque de la matriz.

Esta forma de acomodar la información antes de ser procesada tiene su importancia cuando el tamaño de los problemas, es decir de las matrices asociadas a ellos, hace necesario el uso de dispositivos de acceso secundario como las cintas y discos.

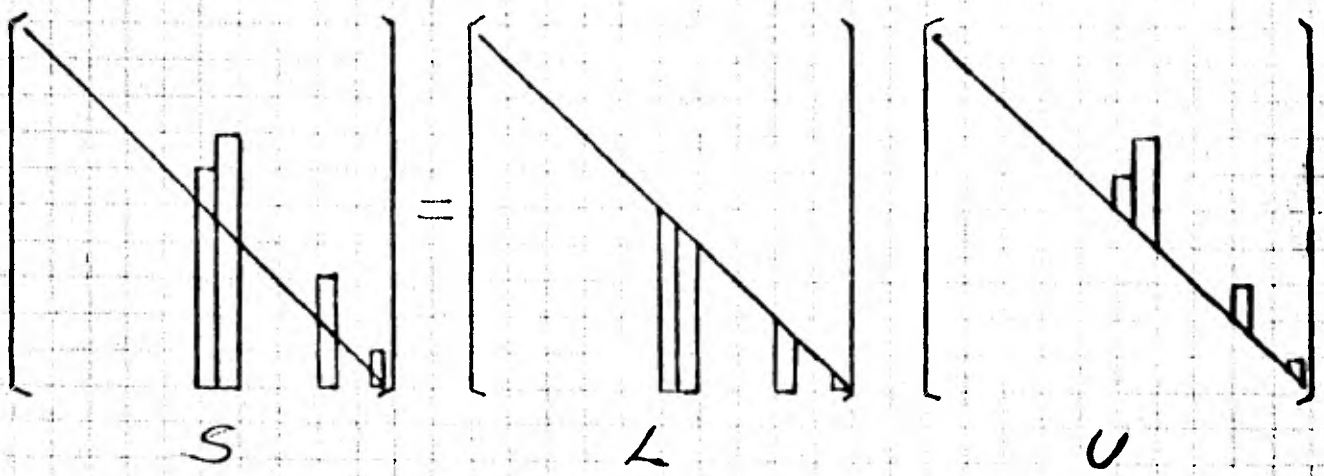
Por ejemplo, mediante permutaciones de renglones y columnas pudo haberse conseguido la siguiente matriz



En ésta se puede ir accesando columna por columna e ir guardando las transformaciones L_j, $j=1,\ldots,m$, para formar la matriz triangular inferior L. De este modo al efectuar la eliminación de los S_{ij} $(i \geq j)$ el llenado se da fundamentalmente en la parte triangular inferior y en las espigas de la matriz S.

Lo importante, que puede observase, es la preservación de la estructura durante el proceso de solución, y que el llenado se tiene principalmente en las espigas. De esta

manera las columnas 9, 10, 15 y 17 que son las espigas de la matriz S, al efectuar la descomposición triangular, el llenado se dá fundamentalmente en esas columnas.



2.4.B Esquema de Reid

Si se sigue la idea de efectuar permutaciones a la matriz original, es natural preguntarse si en vez de intentar llevarla a una triangular inferior por bloques, se efectúan permutaciones para intentar conseguir una triangular superior por bloques. En ésto consiste básicamente el esquema de J. K. Reid.

Para problemas de mediana escala, este

esquema ha probado ser eficiente, puesto que la información se encuentra concentrada en la parte triangular superior y para obtener la descomposición LU de la matriz habrá que efectuarse relativamente poca aritmética con el consecuente ahorro de tiempo y "costo".

Algunas ventajas que pueden apreciarse son:

a) La matriz L de la descomposición LU será de muy baja densidad, ya que se han permutado la mayor parte de los componentes de la matriz S a la parte triangular superior.

b) Dado que solamente se almacenan los factores L_j, $j=1,\ldots,m$ (en realidad los vectores ξ_j que los definen), de L y éstos son "ralos", entonces al calcular la columna que se hará básica, también se verá favorecida si L es rala.

c) Durante cada cambio de base, como se verá en el siguiente capítulo, como la matriz U es más densa, en las actuali-

zaciones, el número de componentes diferentes de cero permanecerá casi constante en número.

Por otro lado, para efectuar la descomposición triangular de una manera estable, hay varios criterios para la elección de los pivotes. Algunos métodos basan la elección de los mejores pivotes en función de que llenen lo menos posible la matriz, y a menos que sea demasiado pequeño y pueda causar resultados numéricamente desastrosos se tendrá que desechar; otros sin embargo, basan la elección de los pivotes en función de la estabilidad numérica. También estos últimos tienen criterios para desechar algún pivote si éste causara mucho llenado en la matriz.

De los criterios más importantes podemos mencionar los dados por Hellerman-Rarick[37] y Markowitz[40]. Esta elección conviene subrayar, generalmente tiene lugar dentro de los bloques, aún a costa de formar más espigas. En esta situación nos enfrentamos al siguiente problema: hacer un balance entre

el llenado de la matriz y la estabilidad numérica de la descomposición.

Para un estudio más detallado de las técnicas para almacenar información, de pivoteo y formas alternativas en que conviene descomponer la matriz, pueden consultarse por ejemplo Brameller [7] y Tewarson [59].

3. Actualización de Sistemas Sparse

Introducción

El método Simplex como se hizo ver en el capítulo 1, en cada cambio de base soluciona los sistemas

$$B X_B = b$$
$$\lambda^T B = C_B^T$$
$$B\alpha = a_j$$

y como el cambio de base consiste en cambiar sólo una columna a la matriz básica por una columna no básica, se tienen entonces dos matrices asociadas a tales bases: B a la base actual y $\hat{B}$ a la modificada. De modo que la matriz básica B y la modificada $\hat{B}$ difieren por una columna.

Ahora bien como se tiene B^{-1} y se requiere calcular $\hat{B}^{-1}$, lo natural es plantearse si es posible calcular esta última sin tener que repetir todo el proceso para obtenerla, esto es, si de alguna manera se puede aprovechar la B^{-1} disponible para el cálculo de $\hat{B}^{-1}$. En efecto tal posibilidad existe y el objeto de este capítulo es mostrar como

hacerlo. Esta operación de obtener $\hat{B}^{-1}$ a partir de B^{-1} es lo que se denomina en la literatura, actualización de la inversa

Ahora bien, para el algoritmo de Bartels-Golub del Método Simplex, por actualización entenderemos el cálculo de la factorización $\hat{L}\hat{U}$ de $\hat{B}$ a partir de la factorización LU de B. Así que siguiendo el procedimiento del capítulo anterior, describiremos las formas de actualizar en F.P.I., F.E.I., LU, etc.

3.1 Actualización en F.P.I.

Supongamos que tenemos una matriz básica B del sistema

$$AX=b$$

Supongamos además que se ha calculado B^{-1}; al cambiar una columna de B, digamos la b_k con $1\leq k\leq m$, por una a_j, $m<j\leq n$. El problema consiste en obtener $\hat{B}^{-1}$ a partir de B^{-1}.

Podemos expresar el lado izquierdo del sistema lineal asociado a B como

$$BX=\sum_{i=1}^{m} X_i b_i$$

Al intercambiar una columna de B se obtiene el siguiente lado izquierdo

$$\begin{aligned}\hat{B}X &= \sum_{\substack{i=1\\ i\neq j}}^{m} X_i b_i + X_j a_j \\ &= \sum_{i=1}^{m} X_i b_i + X_j(a_j-b_j) \\ &= [B+(a_j-b_j)e_j^T]X\end{aligned}$$

De modo que la matriz $\hat{B}$ correspondiente es

$$\hat{B}=B+(a_j-b_j)e_j^T$$

<u>Lema</u> Si $\hat{B} = B + (a_j - b_j) e_j^T$ y B son invertibles, entonces

$$\hat{B}^{-1} = \hat{E}_j \cdot B^{-1}$$

en donde

$$\hat{E}_j = I_m + (\eta^{(j)} - e_j) e_j^T$$

siendo

$$\eta^{(j)} = B^{-1} a_j$$

Demostración

Multiplicando por B^{-1} a la izquierda de $\hat{B}$ se tiene

$$B^{-1}\hat{B} = I_m + B^{-1}(a_j - b_j) e_j^T$$
$$= I_m + (B^{-1}a_j - e_j) e_j^T$$

Tomando inversas en ambos miembros y desarrollando obtenemos

$$\hat{B}^{-1} = \left[I_m + (B^{-1}a_j - e_j) e_j^T \right]^{-1} B^{-1}$$

$$= \hat{E}_j B^{-1}$$

Donde

$$\hat{E}_j = I_m + \mu^{(j)} e_j^T \quad \text{con}$$

$$\mu_j^{(j)} = \hat{a}_{jj} - 1 \quad ; \quad \mu_i^{(j)} = \hat{a}_{ij}$$

en donde cada $\hat{a}_{ij}$ denota la componente i-ésima de $B^{-1}a_j - e_j$.

Ahora bien como hemos calculado B^{-1} en forma producto, esto es

$$B^{-1} = E_m E_{m-1} \cdots E_1$$

en donde cada E_i, $i = 1, \ldots, m$ es de la forma

$$E_i = \begin{bmatrix} 1 & & -\frac{b_{i1}}{b_{ii}} & & \\ & \ddots & \vdots & & \\ & & 1 & & \\ & & \frac{1}{b_{ii}} & & \\ & & \vdots & 1 & \\ & & -\frac{b_{mi}}{b_{ii}} & & \ddots \\ & & & & & 1 \end{bmatrix}$$

entonces $\hat{B}^{-1}$ tiene un factor adicional $\hat{E}_j$ respecto a B^{-1}, i.e.

$$\hat{B}^{-1} = \hat{E}_j B^{-1} = \hat{E}_j E_m E_{m-1} \cdots E_1$$

De la misma manera otras columnas de B pueden ser cambiadas y actualizar su inversa. Por supuesto a cada cambio de columna se añade un factor $\hat{E}_k$ a la inversa de B.

Algunas desventajas de este método son

a) A medida que se vayan cambiando

columnas a la matriz básica B, el resultado se traduce en una cadena de matrices elementales cada vez más grande, lo cual tiende a agotar los dispositivos de almacenamiento de una computadora.

b) La nula posibilidad de pivoteo. Por consiguiente la latente inestabilidad numérica.

En la siguiente sección se verá una alternativa a tales inconvenientes.

3.2 Actualización en F.E.I.

Como anteriormente se hizo, sea b_j con $1 \leq j \leq m$ la columna cambiada de la base $\mathcal{B}$, por una a_k, $m < k \leq n$, entonces la matriz $\hat{\mathcal{B}}$ es de la forma

$$\hat{\mathcal{B}} = \mathcal{B} + (a_j - b_j) e_j^T$$

Calculemos $\hat{\mathcal{B}}^{-1}$, la actualización de $\mathcal{B}^{-1}$ teniendo esta última en forma eliminación.

Sea $\mathcal{L} = \mathcal{L}_m \mathcal{L}_{m-1} \cdots \mathcal{L}_1$, entonces

$$\begin{aligned} \mathcal{L}\hat{\mathcal{B}} &= \mathcal{L}\mathcal{B} + \mathcal{L}(a_j - b_j) e_j^T \\ &= U + \mathcal{L}(a_j - b_j) e_j^T \\ &= U + z_j e_j^T \end{aligned}$$

donde $z_j = \mathcal{L}(a_j - b_j)$

Entonces $\mathcal{L}\hat{\mathcal{B}}$ es de la forma

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & z_{1j} & \cdots & u_{1m} \\ 0 & u_{22} & & z_{2j} & & u_{2m} \\ \vdots & & \ddots & \vdots & & \vdots \\ & & & z_{jj} & & u_{jm} \\ & & & \vdots & \ddots & \vdots \\ & & & 0 & & \\ & & & \vdots & & \\ 0 & \cdots & 0 & z_{mj} & 0 \cdots 0 & u_{mm} \end{bmatrix}$$

Ahora veamos como es $\mathcal{L}\hat{B}$ hasta antes del j-ésimo paso de la sustitución hacia atrás.

$$U_{j+1}\cdots U_m \mathcal{L}\hat{B} = \begin{bmatrix} u_{11} & u_{12} & \cdots & z_{1j} & 0 & \cdots & 0 \\ 0 & u_{22} & \cdots & z_{2j} & & & \\ \vdots & & \ddots & \vdots & & & \\ & & & 0 & z_{jj} & 0 & \\ & & & & 0 & 1 & \\ & & & & & & 0 \\ 0 & \cdots & 0 & z_{mj} & 0 & \cdots & 1 \end{bmatrix}$$

De manera que en el j-ésimo paso sus últimas $(m-j)$ columnas son idénticas a las $(m-j)$ de I_m. Como consecuencia al aplicar la transformación de Jordan E_j^* (Ver Cap. 2) tenemos

$$E_j^* U_{j+1}\cdots U_m \mathcal{L}\hat{B} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1,j-1} & 0 & \cdots & 0 \\ 0 & u_{22} & \cdots & u_{2,j-1} & & & \\ \vdots & & \ddots & \vdots & & & \\ & & & u_{j-1,j-1} & 0 & & \\ & & & & 1 & & \\ & & & & 0 & & \\ & & & & & & 0 \\ 0 & \cdots & \cdots & 0 & \cdots & 0 & 1 \end{bmatrix}$$

Por consiguiente, al continuar el proceso de sustitución obtenemos I_m. Esto es

$$I_m = U_1 \cdots U_{j-1} E_j^* U_{j+1} \cdots U_m L \hat{B}$$

Por consiguiente

$$\hat{B}^{-1} = U_1 \cdots U_{j-1} E_j^* U_{j+1} \cdots U_m L$$

Entonces la inversa $\hat{B}^{-1}$ de $\hat{B}$ se ha obtenido con un reemplazo de U_j por una transformación de Jordan E_j^*.

Al efectuar otro cambio de base, puede ocurrir que

a) La nueva columna a_k esté a la izquierda de E_j^* ($k < j$), en cuyo caso, E_k^* reemplaza a U_k

b) La columna a_k cae a la derecha de E_j^*. En este caso E_k^* se anexa inmediatamente después de E_j^*.

Algunas observaciones a este método son las siguientes

a) En este método de eliminación F.E.I., las posibilidades de pivoteo siguen siendo las mismas que en F.P.I. durante cada actualización, debido a que de otro modo

se rompe la estructura, de ahí su potencial inestabilidad numérica

b) Hay sin embargo la ventaja respecto a F. P. I. de que en cada cambio de base no siempre se anexa una transformación de Jordan para conseguir la actualización. En otras palabras, la representación de $\hat{B}^{-1}$ en F. E. I. no siempre incrementará el número de sus factores durante cada cambio de base como ocurre en F. P. I., lo cual representa un importante ahorro de memoria para su almacenamiento.

3.3 Actualización en F.E.I. según Tomlin

Una variante de actualización de la inversa en forma eliminación la ha dado J. A. Tomlin [21]. Es decir, cuando se ha calculado la inversa de la matriz básica B mediante operaciones elementales en la forma

$$B^{-1} = U_1^{-1} \cdots U_{m-1}^{-1} U_m^{-1} L_m^{-1} L_{m-1}^{-1} \cdots L_1^{-1}$$
$$= U^{-1} L^{-1}$$

y a la matriz B se ha cambiado una columna, esto es, se ha hecho un cambio de base, la forma de actualizar su inversa es como sigue.

Sean $U_0 = U$ y $\hat{B}_0$ la matriz básica B cuya j-ésima columna ha sido cambiada. Así que en

$$L^{-1} \hat{B}_0 = \hat{U}_0$$

$\hat{U}_0$ es de la forma

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & z_{1j} & \cdots & u_{1m} \\ 0 & u_{22} & \cdots & z_{2j} & \cdots & u_{2m} \\ \vdots & \ddots & & \vdots & & \vdots \\ & & 0 & z_{jj} & \cdots & u_{jm} \\ & & & \vdots & & \vdots \\ & & & 0 & & \vdots \\ 0 & \cdots & 0 & z_{nj} & 0 \cdots 0 & u_{nn} \end{bmatrix}$$

Mediante permutaciones la columna insertada es llevada a la m-ésima posición, obteniéndose una matriz de Hessemberg superior H_1, i.e.

$$\hat{U}_0 P_{j,j+1} P_{j+1,j+2} \cdots P_{m-1,m} = \hat{U}_0 P_0 = H_1$$

donde $P_{i,k}$ denota la matriz de permutación que intercambia la columna i por la columna k.

$$H_1 = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1,j+1} & \cdots & u_{1m} & z_{1j} \\ 0 & u_{22} & \cdots & u_{2,j+1} & \cdots & u_{2m} & z_{2j} \\ \vdots & \ddots & & \vdots & & \vdots & \vdots \\ & & 0 & \boxed{u_{j,j+1} \cdots u_{jm}} & & & z_{jj} \\ & & & u_{j+1,j+1} & \cdots & u_{j+1,m} & z_{j+1,j} \\ & & & 0 & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & u_{mm} & z_{mj} \end{bmatrix}$$

Para llevar esta Hessemberg a una triangular superior, obsérvese que eliminar sus componentes del j-ésimo renglón, a partir de la columna j a la $(m-1)$, es equivalente a eliminar los componentes del j-ésimo renglón de U_0, desde la columna $(j+1)$ a la columna m.

$$U_0 = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1j} & u_{1,j+1} & \cdots & u_{1m} \\ 0 & u_{22} & \cdots & u_{2j} & u_{2,j+1} & \cdots & u_{2m} \\ \vdots & & \ddots & \vdots & \vdots & & \vdots \\ & & & u_{jj} & \boxed{u_{j,j+1} \cdots u_{jm}} & & \\ & & & & u_{j+1,j+1} & \cdots & u_{j+1,m} \\ & & & & \vdots & & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & & u_{mm} \end{bmatrix}$$

Supongamos que R_1^{-1} es una transformación que aplicada por la izquierda a U_0 hace cero a sus componentes $u_{j,k}$, $k = j+1, \ldots, m$. Esto es

$$R_1^{-1} U_0 = U_0 - e_j u^T$$

donde

$$u^T = (0, \ldots, 0, u_{j,j+1}, \ldots, u_{j,m})$$

Se sigue que R_1^{-1} es de la forma

$$I_m - e_j r^T = \begin{bmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & & & & \vdots \\ \vdots & & \ddots & & & \vdots \\ & & & 0 & \cdots & 0 \\ & & & 1 - r_{j+1} & \cdots & -r_m \\ & & & & 0 \cdots 0 & \\ & & & & & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

donde

$$r^T U_0 = u^T$$

Por tanto aplicando R_1^{-1} a H_1, sus componentes $u_{j,j+1}, \ldots, u_{j,m}$ son puestos igual a cero. Por consiguiente

$$\tilde{U}_1 = R_1^{-1} H_1 = R_1^{-1} \hat{U}_0 P_0$$

que es de la forma

$$\begin{bmatrix}
u_{11} & u_{12} & \cdots & u_{1j} & \cdots & u_{1m} & z_{1j} \\
0 & u_{22} & \cdots & u_{2j} & \cdots & u_{2m} & z_{2j} \\
\vdots & \ddots & & \vdots & & \vdots & \vdots \\
 & & u_{j-1,j-1} & u_{j-1,j} & \cdots & u_{j-1,m} & z_{j-1,j} \\
 & & 0 & \boxed{0 \cdots 0} & & & z_{jj} \\
 & & & u_{j+1,j+1} & \cdots & u_{j+1,m} & z_{j+1,j} \\
 & & & 0 & & & \vdots \\
0 & \cdots & 0 & 0 & \cdots & u_{mm} & z_{mj}
\end{bmatrix}$$

Nótese que $\tilde{U}_1$ difiere de H_1 en el j-ésimo renglón pues sus $u_{jk} = 0$, $k = j+1, \ldots, m$. Permutando el elemento z_{jj} a la posición (m,m) y recorriendo los renglones a partir del k-ésimo ($k > j$) un lugar hacia arriba se obtiene una matriz triangular superior. Es decir $P_0^{-1} \tilde{U}_1$, es triangular superior. Así que

$$\hat{U}_0 = R_1 \tilde{U}_1 P_0^{-1}$$

y por consiguiente

$$\bar{B}_1^{-1} = P_0 \tilde{U}_1^{-1} R_1^{-1} L^{-1}$$

En el siguiente cambio de base se dispone entonces de

$$\bar{B}_1 = L R_1 \tilde{U}_1 P_0^{-1}$$

Al cambiar una columna a esta matriz $\bar{B}_1$, se tendrá que

$$\hat{U}_1 = R_1^{-1} L^{-1} \hat{B}_1 P_0$$

Siguiendo el proceso arriba descrito podemos efectuar la actualización de su inversa.

De nuevo, la columna insertada es permutada a la m-ésima posición obteniéndose una matriz de Hessemberg superior H_2, i.e.

$$\hat{U}_1 P_1 = H_2$$

Al aplicar la transformación R_2^{-1} a H_2 se obtiene

$$\tilde{U}_2 = R_2^{-1} H_2 = R_2^{-1} \hat{U}_1 P_1$$

luego

$$\hat{U}_1 = R_2 \tilde{U}_2 P_1^{-1}$$

De modo que

$$R_1^{-1} L^{-1} \hat{B}_1 P_0 = R_2 \tilde{U}_2 P_1^{-1}$$

Por consiguiente

$$\bar{B}_2^{-1} = P_0 P_1 \tilde{U}_2^{-1} R_2^{-1} R_1^{-1} L^{-1}$$

Se sigue por inducción que en la k-ésima actualización se obtiene la fórmula.

$$\bar{B}_k^{-1} = P_0 \cdots P_k \tilde{U}_k^{-1} R_k^{-1} \cdots R_1^{-1} L^{-1}$$

Así, por cada actualización se hace necesaria la inserción de un nuevo factor R_{k+1}^{-1}, a la derecha de $\tilde{U}_k^{-1}$ y a la izquierda de $R_k^{-1} R_{k-1}^{-1} \cdots R_1^{-1} L^{-1}$. Lo que resulta muy complicado de manejar con un solo archivo. Esta dificultad es superada si se trabaja con dos archivos: uno para $\tilde{U}_k^{-1}$, y otro para $R_k^{-1} R_{k-1}^{-1} \cdots R^{-1} L^{-1}$.

En la implementación de esta forma de actualizar la inversa dada como

$$B^{-1} = U_1^{-1} \cdots U_m^{-1} L_m^{-1} \cdots L^{-1}$$
$$= U^{-1} L^{-1}$$

hay que hacer algunas observaciones.

Nótese que en cada cambio de base, si bien se inserta una transformación R^{-1} para

actualizar $\bar{B}^{-1}$; también un factor U_j^{-1} de U^{-1} desaparece. Además la actualización $\bar{U}_k^{-1}$ de U_k $(k>j)$ es muy simple pues en cada una de ellas su componente $u_{jk}=0$. Por tanto

$$U^{-1}=U_1^{-1}\cdots U_{j-1}^{-1}U_j^{-1}U_{j+1}^{-1}\cdots U_m^{-1}$$

$$\tilde{U}^{-1}=U_1^{-1}\cdots U_{j-1}^{-1}\bar{U}_{j+1}^{-1}\cdots\bar{U}_m^{-1}\bar{U}_{m+1}^{-1}$$

Esto es, U^{-1} y $\tilde{U}^{-1}$ son idénticas en sus primeras $(j-1)$ columnas; en $\tilde{U}^{-1}$ no aparece U_j^{-1}, y como última columna tiene a $\bar{U}_{m+1}^{-1}$. Veamos cómo es esta última columna.

Si denotamos por $z_j=L^{-1}a_k$, $1\le k\le n$, a la columna insertada en B, entonces

$$\bar{U}_{m+1}^{-1}=R^{-1}L^{-1}a_k=R^{-1}z_j$$

El cálculo del vector r^T que define la transformación R^{-1}, puede verse que es una transformación hacia atrás parcial usando el renglón $u_j^T=(0,\ldots,0,u_{j,j+1},\ldots,u_{jm})$ de U.

A continuación se bosqueja la actualización de los factores U_k^{-1} $(k>j)$, con el cálculo

simultáneo del vector r^T.

En U^{-1} no se consideran las $U_1^{-1}, \ldots, U_{j-1}^{-1}$; se elimina la U_j^{-1}. Si la componente $u_{jk} \neq 0$ para $k > j$, entonces se coloca en la k-ésima posición de r^T, se marca u_{jk} como cero en U_k^{-1} y se postmultiplica r^T por U_k^{-1}. Esto se hace para $k = j+1, \ldots, m$. El resultado final en r^T es el r^T buscado.

Finalmente se calcula $R^{-1}z$ y se anexa a la derecha del archivo $\tilde{U}^{-1}$ y R^{-1} a la izquierda del archivo L^{-1}.

3.4 Actualización de la descomposición LU según Bartels - Golub

Supongamos que se dispone de la descomposición triangular de la matriz básica B

$$PB = LU$$

Una vez que se ha realizado el cambio de base, el proceso de actualización propuesto por Bartels - Golub([1],[2],[5]), es como sigue.

Sea $\hat{B}$ la matriz resultante del cambio de la j-ésima columna a B. Entonces

$$P\hat{B} = PB + P(a_j - b_j)e_j^T$$

Así que

$$\begin{aligned} L^{-1}P\hat{B} &= L^{-1}PB + L^{-1}P(a_j - b_j)e_j^T \\ &= U + z_j e_j^T \\ &= \hat{U} \end{aligned}$$

donde la matriz $\hat{U}$ es de la forma

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & z_{1j} & \cdots & u_{1m} \\ 0 & u_{22} & \cdots & z_{2j} & \cdots & u_{2m} \\ \vdots & \ddots & \ddots & \vdots & & \vdots \\ \vdots & & 0 & z_{jj} & \cdots & u_{jm} \\ \vdots & & & \vdots & & \vdots \\ \vdots & & & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & z_{mj} & 0 \cdots 0 & u_{mm} \end{bmatrix}$$

Para obtener la actualización de B, primero se permuta la columna insertada a la posición m-ésima obteniéndose de esta manera una matriz de Hessemberg superior H.

$$H = \hat{U} Q_1$$
$$= L^{-1} P \hat{B} Q_1$$

donde esta matriz de Hessemberg es de la forma

$$\begin{bmatrix} u_{11} u_{12} & \cdots & u_{1,j+1} & \cdots & u_{1m} z_{1j} \\ 0 \; u_{22} & \cdots & u_{2,j+1} & \cdots & u_{2m} z_{2j} \\ \ddots & \ddots & \vdots & & \vdots \\ & \ddots & \vdots & & \vdots \\ & & 0 \; u_{j,j+1} & \cdots & u_{jm} z_{jj} \\ & & \vdots \; u_{j+1,j+1} & \cdots & u_{j+1,m} z_{j+1,j} \\ & & \vdots \; 0 & \ddots & \vdots \\ 0 & \cdots \; 0 & 0 \cdots 0 & & u_{mm} z_{mj} \end{bmatrix}$$

Para llevarla a una triangular superior U, se le aplica eliminación gaussiana para hacer ceros bajo su diagonal principal. Se puede en esta parte hacer intercambio de renglones adyacentes para elegir pivotes. Obtendremos de esta manera U a partir de H por medio de la aplicación a H de

transformaciones T_i del modo siguiente

$$U = T_{m-1} P_{m-1} \cdots T_j P_j H$$

donde cada T_i, $i = m-1, \ldots, j$ es de la forma

$$\begin{bmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & & \vdots \\ \vdots & & & 0 & \frac{1}{u_{i,i+1}} & \vdots \\ \vdots & & & & -\frac{u_{i+1,i+1}}{u_{i,i+1}} & 1 \\ \vdots & & & & 0 & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$$

Y cada P_i, $i = j, \ldots, m-1$ es la matriz idéntica I_m o bien I_m con sus renglones i e $(i+1)$ permutados. Entonces de

$$H = L^{-1} P \hat{B} Q_1$$

y

$$U = T_{m-1} P_{m-1} \cdots T_j P_j H$$

se tiene que

$$R \hat{B} Q_1 = L P_j T_j^{-1} \cdots P_{m-1} T_{m-1}^{-1} U$$

Al efectuar otros cambios de base, el proceso de actualización descrito arriba

puede repetirse.

Tomando

$$\hat{B}^{(1)}=\hat{B}\ ;\ U^{(1)}=U\ ;\ G^{(1)}=P_j^{(1)}T_j^{(1)-1}\cdots P_{n-1}^{(1)}T_{n-1}^{(1)-1}$$

para el primer cambio de base y

$$G^{(i)}=P_j^{(i)}T_j^{(i)-1}\cdots P_{n-1}^{(i)}T_{n-1}^{(i)-1}\ ,\ i=1,\ldots,k$$

$$\tilde{G}=G^{(1)}G^{(2)}\cdots G^{(k)}\ ,$$

para las siguientes. Entonces tendremos

$$P\hat{B}^{(2)}=LG^{(1)}\hat{U}$$

después de cambiar una columna a la nueva base. De nuevo para reducir la Hessemberg

$$H=(LG^{(1)})^{-1}P\hat{B}^{(2)}Q^{(2)}$$

a una triangular superior, aplicamos la serie de transformaciones $G^{(2)}$, i.e.

$$U^{(2)}=G^{(2)}H$$

Por consiguiente

$$PB^{(2)}Q^{(2)}=LG^{(1)}G^{(2)}U^{(2)}$$

Se sigue por inducción que en la k-ésima actualización obtenemos la fórmula

$$PB^{(k)}Q^{(k)}=LG^{(1)}\cdots G^{(k)}U^{(k)}$$
$$=L\tilde{G}U^{(k)}$$

A este método de actualización podemos hacer las siguientes observaciones.

a) Empecemos por anotar que este esquema de Bartels-Golub de actualización da lugar a un proceso "en la práctica" numéricamente estable, debido a que siempre se tiene la opción a efectuar pivoteo.

b) Debido a la poca densidad de las matrices básicas, las transformaciones T_k para la actualización, en muchos casos es la matriz idéntica I_m, lo cual represente un significativo ahorro de aritmética.

c) Debido a que en la elección de pivotes se tienen que hacer permutaciones por renglones, esto representa la posibilidad de que la matriz sea cada vez más densa.

d) Otra dificultad en la implementación de este método se encuentra al tratar de solucionar problemas muy grandes, pues la serie de transformaciones $G^{(k)}$ requeridas en cada actualización hacen difícil su manejo, si se entiende que para almacenar la información se hace uso de dispositivos de almacenamiento secundario como los discos.

3.5 Actualización en L U según Reid

Como se recordará, cuando los problemas que se van a resolver no exceden los límites de capacidad de la máquina (Secc. 2.4.B), conviene emplear permutaciones que lleven la matriz de coeficientes a una triangular superior por bloques

$$B = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1n} \\ & B_{22} & \cdots & B_{2n} \\ & & \ddots & \vdots \\ & & & B_{nn} \end{bmatrix}$$

Una vez hecho lo anterior, supóngase que mediante operaciones elementales a B se ha conseguido obtener

$$L_m^{-1} L_{m-1}^{-1} \cdots L_1^{-1} B = P U Q$$

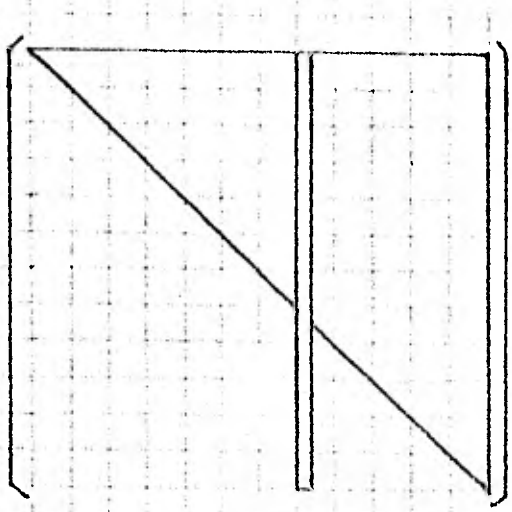
$$L^{-1} B = P U Q$$

con P y Q matrices de permutación; U una matriz triangular superior.

Al efectuar un cambio de base, con la nueva matriz $\bar{B}$ se tendrá

$$L^{-1}\bar{B} = P\hat{U}Q$$

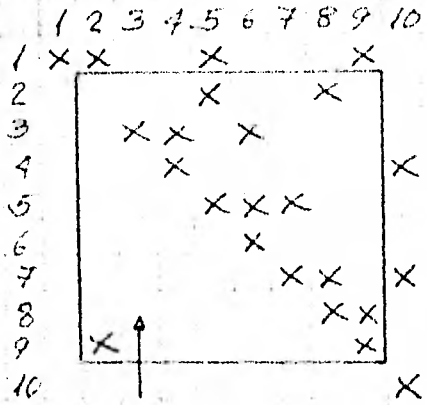
donde $\hat{U}$ difiere de U en la misma columna en que difieren $\bar{B}$ y B. La matriz $\hat{U}$ tiene la forma



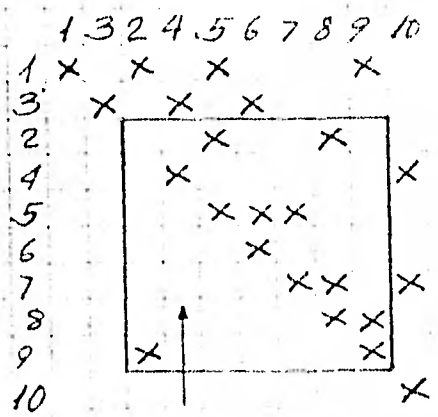
Para la reducción de $\hat{U}$ a una triangular superior, según el esquema de J. K. Reid [49], antes de efectuar alguna operación aritmética se emplean permutaciones adicionales con el fin de ahorrar aritmética y llenado en la matriz U.

Sea b_i la columna cambiada (espiga) con cuyo último componente diferente de cero se encuentra en el renglón r_k $(k \leq n)$, se restrin

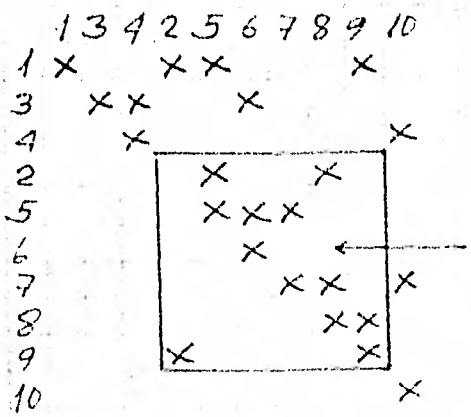
ge entonces la atención al bloque comprendido por los renglones Γ_i, Γ_k y las columnas C_i, C_k. Con el siguiente ejemplo se ilustra como hacer tales permutaciones



En la submatriz comprendida por las columnas y renglones i y k-ésimos, se busca de izquierda a derecha alguna columna con un sólo elemento diferente de cero; si la hay, ésta debe tener tal elemento en la diagonal principal (pues B es no singular y por consiguiente U), dicho elemento se permuta a la posición (i,i), moviendo las restantes columnas un lugar a la derecha, reduciéndose el orden del bloque (y la espiga) en uno.

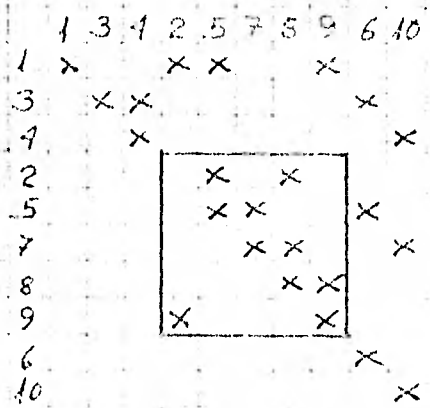


La búsqueda se continúa pero a partir de la columna ya encontrada. Es decir, tal como aparece en el ejemplo, la columna con un sólo elemento diferente de cero (aquí la tercera), sirve de nuevo punto de partida en la búsqueda de otras tales columnas.

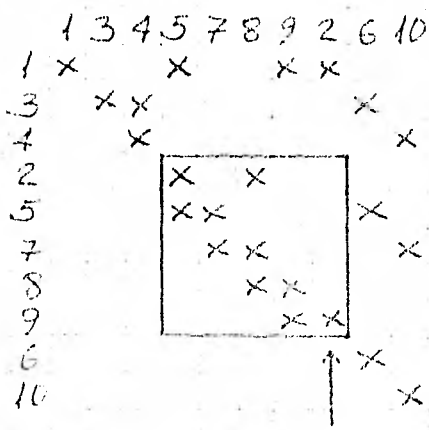


Cuando ya no haya columnas con un sólo elemento diferente de cero, entonces se buscan los renglones también con un sólo elemento diferente cero y el primero que se encuentre se permuta a la posición (k, k), moviendo

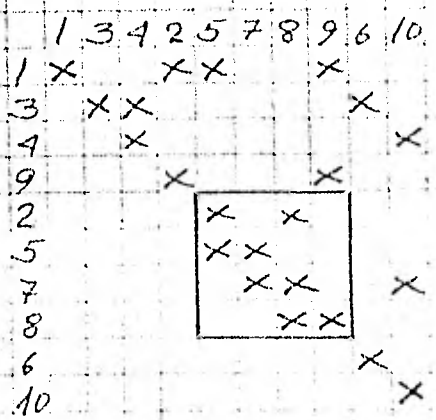
los restantes renglones un lugar a la izquierda. De nuevo se reduce el orden del bloque en uno.

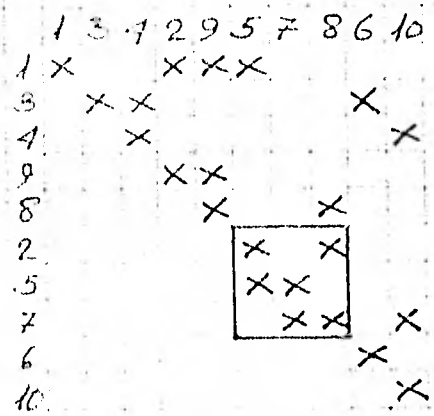


También en este caso partimos en las búsquedas subsecuentes a partir del renglón anterior. Una vez encontrados todos los renglones y columnas con un sólo elemento diferente de cero, se permuta la espiga lo más a la derecha del bloque obteniéndose una Hessemberg superior



De nuevo en esta matriz se repite el procedimiento previamente descrito, i.e.





En el momento de no encontrar columnas y renglones con solamente un elemento diferente de cero, entonces, tal como se hace en el esquema de Bartels-Golub se procede a la reducción de la Hessemberg a una triangular superior mediante operaciones elementales $L_{m+1}^{-1}, \ldots, L_{\bar{m}}^{-1}$. Esto da por resultado la matriz

$$\bar{B} = \bar{L}\bar{P}\bar{U}\bar{Q}$$

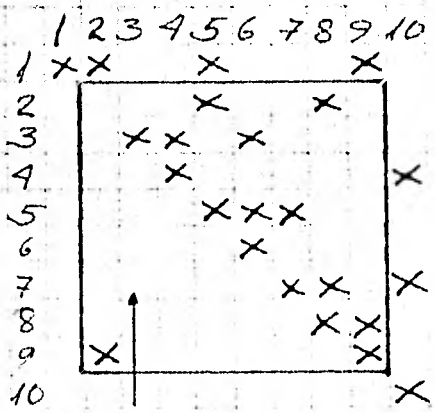
donde $\bar{L}$ difiere de L solamente en que puede tener adicionales L_j, $j = m+1, \ldots, \bar{m}$, en tanto $\bar{U}$ difiere de U en la columna reemplazada y por las operaciones elementales aplicadas.

Puede resumirse el proceso arriba descrito en los pasos siguientes

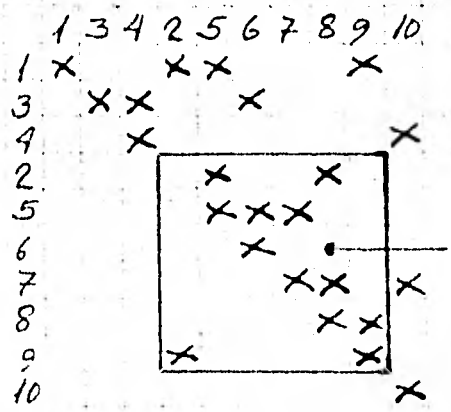
Sea a_i la columna intercambiada con último componente diferente de cero en el renglón r_k $(k \leq m)$. Se forma el bloque con los renglones r_i, r_k y las columnas C_i, C_k.

Paso 1

En el bloque se buscan las columnas con un sólo elemento y se permutan sucesivamente a las posiciones (i,i), $(i+1, i+1)$, ...



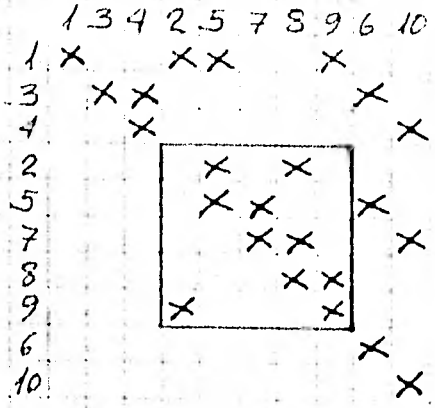
Matriz Original



Matriz con las columnas de un sólo elemento ya permutadas

Paso 2

De la misma manera en el bloque tal vez más reducido, los renglones con un sólo elemento se permutan a las posiciones (k,k), $(k-1, k-1)$, ...



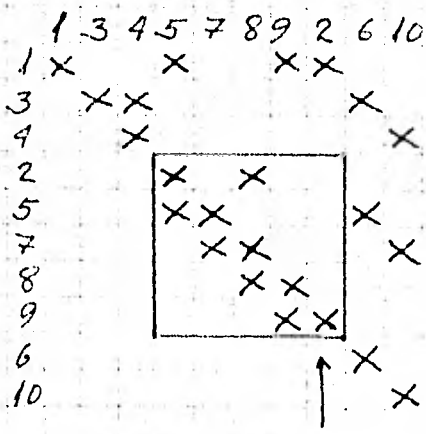
Matriz con los renglones de un sólo elemento ya permutados

Paso 3

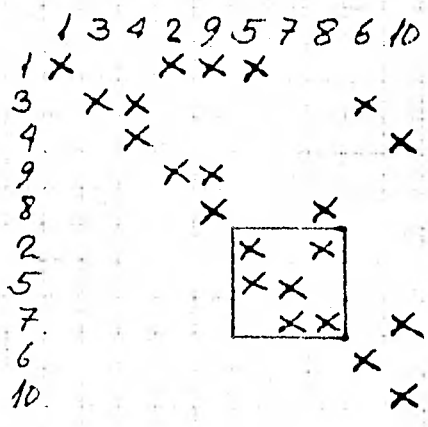
Se permuta la espiga a la última posición dentro del bloque, obteniéndose una Hessemberg superior.

Paso 4

Los pasos (1) y (2) se repiten en la matriz de Hessemberg



Matriz de Hessemberg



Hessemberg Reducida

Paso 5.

Aplicar operaciones elementales para obtener la matriz triangular superior $\bar{U}$.

Algunas ventajas que se consiguen al permutar la mayor cantidad de información a la parte triangular superior son

a) La matriz L de la descomposición triangular LU será de muy baja densidad.

b) Como los factores L_j, $j=1,\ldots,m$, en realidad los vectores $\eta^{(i)}$ que los definen, son "ralos", entonces al calcular la columna que se hará básica, también se verá favorecida.

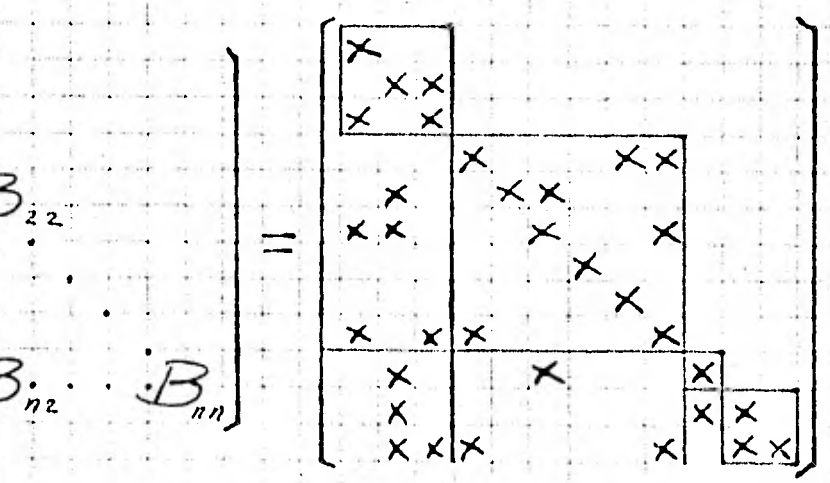
c) Durante cada cambio de base, debido a que U es densa, el número de sus componentes tenderá a permanecer en equilibrio.

Sin embargo esta forma de actualización tiene una limitación muy fuerte, ya que para hacer un uso eficiente se requiere tener toda la información en la memoria principal de la computadora, y por consiguiente es sólo aplicable a problemas de mediana escala.

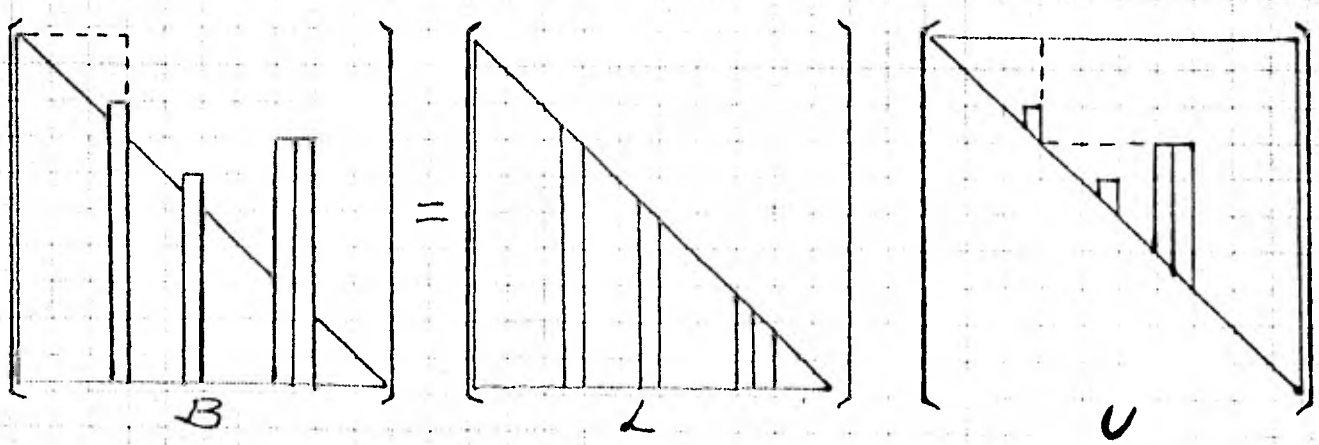
3.6 Actualización en LU según Saunders

Songamos que se dispone de una matriz básica B la cual ha sido llevada, mediante permutaciones, a una triangular inferior por bloques y espigas (Ver Secc. 4A Cap. 2).

$$B = \begin{bmatrix} B_{11} & & & \\ B_{12} & B_{22} & & \\ \vdots & \vdots & \ddots & \\ B_{n1} & B_{n2} & \cdots & B_{nn} \end{bmatrix} =$$



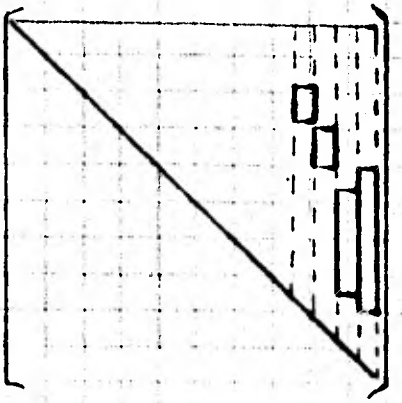
Como se recordará, al efectuar la descomposición LU de B, su estructura de bloques y espigas se transmite también a sus factores triangulares



Cuando se ha cambiado de base, la actualización de B puede efectuarse según M. A. Saunders [53] de la manera siguiente

Una vez conseguida la descomposición LU, a la matriz triangular superior U se permutan sus espigas al final (conservando su orden). Aplicando las mismas permutaciones a sus renglones obtenemos una matriz $\hat{U}$ de la forma

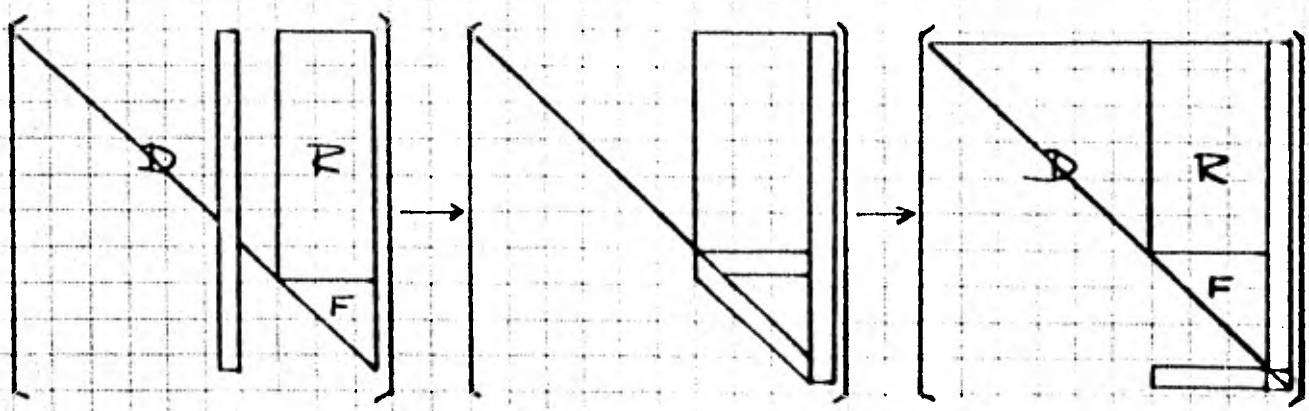
$$\hat{U} = P^T U P = \left[\begin{array}{c|c} D & R \\ \hline & F \end{array}\right] = \left[\begin{array}{c} \\ \\ \end{array}\right.$$



En lo que sigue de esta sección consideramos $U = \hat{U}$.

Al efectuar un cambio de base, siendo b_j la columna que deja la base por una z_j (nueva espiga). La actualización del método de Saunders puede resumirse del modo siguiente

Primeramente se permuta la espiga al final obteniéndose una matriz de Hessemberg superior, al igual que en el método de Bartels - Golub. El j-ésimo renglón de U, r_j^T se permuta hasta el fondo (Ver figura)



Finalmente, el renglón r_j^T ya permutado es reducido a un múltiplo de e_m^T mediante eliminación gaussiana. En esta parte pueden hacerse intercambio de renglones para elegir pivotes.

Las dimensiones de K y F, es importante hacer notar, dependen del número de espigas. Si por ejemplo B es una matriz de orden m y tiene k-espigas, el bloque F será una matriz de orden k y triangular superior, en tanto que el bloque K será de di-

mensión $(m-k)\times k$. Por tanto, si la columna intercambiada cae a la izquierda de R (y F) entonces ambos aumentan su dimensión en uno; en caso contrario retienen sus dimensiones.

Obsérvese también que en la reducción de la Hessemberg superior a una triangular superior, el único bloque que toma parte es F, por consiguiente es importante conseguir una descomposición LU con un número reducido de espigas.

De este método de actualización podemos hacer las siguientes observaciones.

a) En cada actualización una columna de la matriz U desaparece para dar lugar a una z_j. Si esta columna cae en los bloques R y F, entonces se amortiguará el llenado de U.

b) Por las características de actualización de este método, se puede hacer uso de dispositivos de almacenamiento secundario. Esto es, ya que excepto F, la parte restante de U puede tenerse en disco, lo cual abre

la vía para ser aplicado a problemas de gran tamaño.

c) Por las posibilidades de elegir pivotes durante la eliminación del vector r^T, este método es una alternativa numéricamente estable que hace practicable las ideas centrales de Bartels - Golub.

3.7 Actualización en LU según Gay

Un esquema de actualización de los factores triangulares de B, que parece ser prometedor, incorpora las ideas de Saunders y Reid. Tal esquema ha sido propuesto por D. M. Gay [25].

Se permuta primeramente la matriz básica B para conseguir una triangular inferior por bloques y espigas.

$$B = \begin{bmatrix} B_{11} & & & \\ B_{21} & B_{22} & & \\ \vdots & \vdots & \ddots & \\ B_{n1} & B_{n2} & \cdots & B_{nn} \end{bmatrix}$$

Después de haber conseguido la descomposición triangular

$$B = LU$$

donde la matriz U tiene muy pocos elementos sobre su diagonal (espigas). Estas espigas son permutadas al final para conseguir

$$\hat{U} = P^T U P = \left[\begin{array}{c|c} D & R \\ \hline & F \end{array}\right]$$

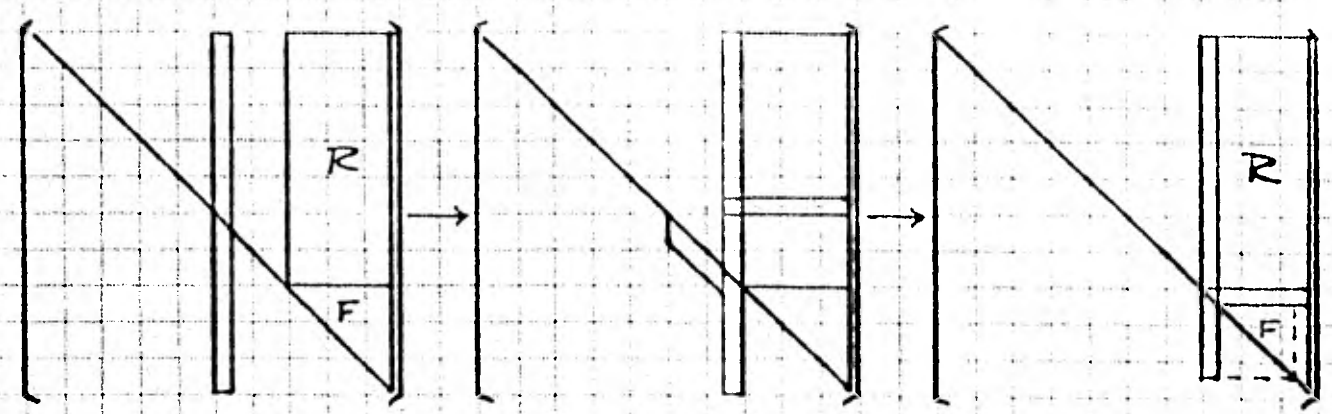
Cuando se efectúa un cambio de base puede ocurrir:

i) La columna intercambiada cae a la izquierda de los bloques R y F, entonces R aumenta una columna y pierde un renglón, en tanto F aumenta su orden en uno. O bien

ii) Si cae la columna intercambiada dentro de los bloques R y F, entonces ambos retienen sus dimensiones.

En ambos casos sin embargo, sólo F juega un papel activo en la actualización.

El esquema de actualización propuesto por Gay sugiere, en el primer paso, que la columna intercambiada se permute hasta ser la primera del bloque R, y el renglón perdido por R sea ahora el primero de F.



Entonces en el bloque F aplicar el tipo de permutaciones dadas en el esquema de Reid. Es decir, al esquema de actualización de Saunders aplicar permutaciones a F tal como se hace en el esquema de actualización de Reid.

4. Vector de Costos

Introducción

Recordemos que la formulación del problema de Programación Lineal en su forma estandar dice

Minimizar la funcional

$$z = c^T x$$

sujeta a las restricciones

$$Ax = b, \quad x \geq 0$$

Para obtener el óptimo de la funcional z, el Método Simplex lo consigue iterativamente y en cada iteración simplex se resuelven tres sistemas. A saber, dos directos

$$Bx_B = b$$

$$By = a_j$$

y uno transpuesto

$$B^T \lambda = c_B$$

Hemos visto en los capítulos anteriores cómo resolver y actualizar los sistemas directos. Ahora veamos qué sucede con el sistema transpuesto en cada iteración del Método Simplex.

Es importante tener en cuenta que calcular el vector λ^T en cada iteración, es una de las operaciones que consume mayor tiempo. De Bu-chet[9] menciona que el cálculo de $\lambda^T = C_B^T B^{-1}$ consume casi tres veces el tiempo requerido para calcular el vector columna que se hace básico; y el cálculo de los costos relativos $r^T = C_D^T - \lambda^T D$, alrededor del 40% de tiempo por iteración.

Básicamente hay dos políticas respecto a qué hacer con el cálculo de r^T:

a) Resolver el sistema $B^T \lambda = C_B$; o bien

b) Actualizar r^T, siendo la actualización de λ una manera de hacerlo.

4.1 Solución del sistema $B^T\lambda = C_B$

Es común que en la solución del sistema $B^T\lambda = C_B$ se tenga la inversa (o alguna de sus factorizaciones), de la matriz básica B en forma producto. Esto es

$$B^{-1} = E_m E_{m-1} \cdots E_1$$

de modo que para elegir la nueva variable básica se necesita

i) Calcular el vector λ^T

$$\lambda^T = C_B^T B^{-1} = C_B^T E_m E_{m-1} \cdots E_1 \qquad (4.1.1)$$

ii) Seleccionar, de las columnas no básicas la columna entrante calculando los costos relativos.

$$r^T = C_D^T - \lambda^T D \qquad (4.1.2)$$

donde $\lambda^T B = C_B^T$ (o bien $B^T\lambda = C_B$)

Como decíamos, resolver $B^T\lambda = C_B$ requiere aproximadamente tres veces lo que el cálculo del vector entrante $\alpha = B^{-1}a_p$, debido a que el acceso a los factores η_k que definen las columnas de la inversa B^{-1} tiene que

hacerse en orden inverso. Esta limitación es propia de las diversas versiones existentes en donde se calcula λ, debido a que tienen a B^{-1}, en cierto sentido, en forma producto y el acceso a sus factores η_k se hace en orden inverso a como fueron creados.

4.2 Actualización

En las diferentes versiones implementadas del Método Simplex, la resolución del sistema transpuesto $B^T\lambda = c_B$ se efectúa en cada iteración, en tanto que el sistema directo se actualiza. Aún en el esquema propuesto por Reid [50] básicamente lo que se hace es diseñar el código de manera que se simplifique la resolución del sistema transpuesto a costa de complicar la resolución del directo.

Con el enfoque seguido por Saunders [55] a las ideas dadas por Gill y Murray, los papeles se invierten, es decir, se resuelve el sistema transpuesto y se actualiza el directo. Ideas en este sentido serán bosquejadas a continuación.

4.3 Actualización según Tomlin

Para la actualización de los costos relativos

(4.3.1) $$r^T = C_D^T - \lambda^T D$$

donde

(4.3.2) $$\lambda^T B = C_B^T \quad (\text{o bien } B^T\lambda = C_B)$$

el esquema propuesto por J. A. Tomlin [61], aprovecha la información obtenida al momento de actualizar la matriz básica.

Recordemos que la actualización de la matriz básica $\hat{B} = B + (b_j - a_k)e_j^T$, se efectúa con la aplicación a $\hat{B}$ de una transformación $E^{-1} = I_m - e_k r^T$ ($r = u^T(U^{-1})$)

donde E es de la forma

$$\begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & & & \vdots \\ \vdots & & 1 & 0 & \cdots & 0 \\ & & & 1 & r_{k+1} & r_m \\ & & & & 1 & 0 \cdots 0 \\ & & & & & 0 \\ 0 & \cdots & \cdots & 0 & & 1 \end{bmatrix}$$

Aquí se ha supuesto que se dispone de B^{-1} en forma producto. Esto es

$$B^{-1} = U_1^{-1} \cdots U_m^{-1} L_m^{-1} \cdots L_1^{-1}$$

donde las transformaciones elementales U_j y L_j, $j = 1, \ldots, m$ se almacenan en archivos por separado.

Brevemente el esquema es como sigue. Denotemos por

(4.3.3) $$r_j = \lambda^T a_j \quad , j = m+1, \ldots, n$$

y sea X_q la nueva variable básica. Si denotamos por a_{pq} la componente de la matriz que consideramos como pivote tenemos para los nuevos costos relativos r_j'

$$r_j' = r_j - (a_{pj}/a_{pq}) r_q$$

Si formamos

(4.3.4) $$\lambda_p^T = e_p^T B^{-1}$$

entonces para el cálculo de los costos relativos

(4.3.5) $$r_j' = r_j - (\lambda_p)^T a_j \, r_q$$

almacenamos r_q, formamos λ'_p de (4.3.4) y empleamos (4.3.5) para el cálculo de cada r_j.

Debemos notar que en general λ'_p tiene menos componentes que λ. Esto como consecuencia tiene ventajas al requerir menos aritmética en el cálculo de λ_p en lugar de λ; y menos aritmética al calcular $\lambda_p^T a_j$, en lugar de λa_j, $j = k+1, \ldots, n$.

Observando las relaciones (4.3.3) y (4.3.5) puede verse cómo los costos relativos están dados, i.e.,

$$r'_j = \lambda^T a_j - (\lambda'_p)^T a_j \, r_q$$

$$= (\lambda - r_q \lambda'_p)^T a_j$$

Por consiguiente en lugar de calcular λ' directamente de $\lambda^T = C^T B^{-1}$ con la nueva inversa, podemos guardar λ, calcular λ_p^T y tomar λ' como

$$\lambda' = \lambda - r_q \lambda'_p$$

4.4 Actualización según Golfarb

Una forma de actualizar tanto el vector λ como los costos relativos propuesta por Golfarb[33] consiste en lo siguiente.

Sea X_k con $1 \leq k \leq m$ la variable básica reemplazada por la variable X_p, $m < p \leq n$, no básica.

Para el vector $\hat{\lambda}^T$ se requiere

$$\hat{\lambda}^T \leftarrow \lambda^T - (z_p/\alpha)\, v \tag{4.4.1}$$

donde $\alpha = v^T a_p$, es el elemento pivote.

Y para los costos relativos

$$\hat{z}_j \leftarrow z_j - (z_p/\alpha)\, v^T a_j \tag{4.4.2}$$

donde $B^T v = e_k$

La ventaja del método de actualización propuesto por Golfarb, está en el cálculo de $B^T v = e_k$, pero con la matriz básica ya actualizada $\hat{B}$. Esto como consecuencia del hecho que v es un múltiplo de $\hat{B}^{-T} e_l$, $l \geq k$.

Respecto a la versión de Bartels-Golub clásica se tiene el siguiente

<u>Lema.</u> Si $B^T v = e_k$, entonces $V = \alpha \hat{V}$ donde $\hat{B}\hat{V} = e_m$

Demostración.

Sean b_k la columna correspondiente a la variable básica x_k, y a_p la columna de la variable que se hace básica. Entonces

$$\hat{B} = (B + (a_p - b_k) e_k^T) P$$

donde P es una matriz de permutación

$$\begin{aligned} v^T\hat{B} &= (v^T B + v^T a_p e_k^T - v^T b_k e_k^T) P \\ &= (v^T B + v^T a_p e_k^T - v^T B e_k e_k^T) P \\ &= (e_k^T + v^T a_p e_k^T - (e_k^T e_k) e_k^T) P \\ &= (v^T a_p e_k^T) P \end{aligned}$$

Y como $\alpha = v^T a_p$, tenemos que

$$v^T\hat{B} = \alpha e_k^T P$$

Ahora como $e_k^T P = (P^T e_k)^T = e_m^T$

$$v^T\hat{B} = \alpha e_m^T$$

entonces $V = \alpha e_m^T \hat{B}^{-T} = \alpha \hat{V}$

donde $\hat{B}^T\hat{V} = e_m$ ∎

Además en la resolución de

$$\hat{B}^T \hat{v} = e_m \quad , \quad v = \alpha \hat{v}$$

sólo se requiere de un "recorrido" en $\hat{L}^T$ y ninguno en $\hat{U}$, de la factorización de $\hat{B}$

$$\hat{B}^T \hat{v} = \hat{U}^T \hat{L}^T \hat{v} = e_m$$

$$\hat{L}^T v = \hat{U}^{-T} e_m$$

$$= \frac{1}{\hat{\mu}_{mm}} e_m$$

Más aún la superioridad de emplear (4.4.1) y (4.4.2) se nota si consideramos el ahorro de aritmética, pues en (4.4.2) no se requiere calcular $\hat{\lambda}^T D$.

5. El Simplex vía la Descomposición QR.

Introducción

Para el problema

$$\text{Mín } z = c^T x$$

$$\text{sujeto a } Ax = b \;,\; x \geq 0$$

el Método Simplex requiere de la solución de tres sistemas de ecuaciones lineales

$$Bx = b$$

$$B^T \lambda = c_B$$

$$B\alpha = a_j$$

en cada iteración

Se han presentado en los capítulos anteriores las implementaciones más conocidas las cuales solucionan tales sistemas mediante transformaciones elementales de eliminación.

En este capítulo se presenta una versión numéricamente estable trabajada por M.A. Saunders [54], [55], la cual fue sugerida por P.E. Gill y W. Murray [27], por primera vez. Esta descomposición se basa en la descomposición de la matriz básica B, mediante transformaciones ortogonales en el producto

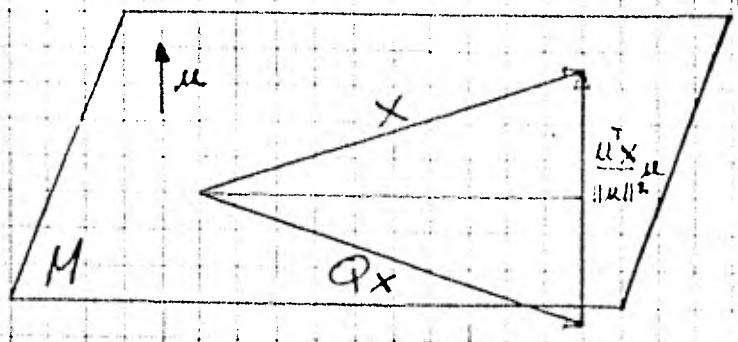
$$B = LQ$$

donde L es una matriz triangular inferior y Q ortogonal.

La descomposición ortogonal de cualquier matriz S se obtiene con la aplicación a S de transformaciones ortogonales Q_j también llamadas transformaciones de Householder. Esta descomposición es numéricamente estable, pues las transformaciones rígidas conservan la norma y es precisamente este hecho donde radica su importancia.

5.1 Transformaciones de Householder

Con el propósito de exhibir como son las transformaciones de Householder (reflexiones), sean M un subespacio de $\mathbb{R}^m$, $x \in \mathbb{R}^m$. La proyección de X en un vector u ortogonal a M es



$$\text{Proy}_u X = \frac{u^T X}{\|u\|^2} u$$

entonces la reflexión de Householder Q x queda dada por

$$Qx = x - 2\frac{u^T x}{\|u\|^2}u$$

$$\therefore Q = I_m - 2\frac{uu^T}{\|u\|^2}$$

Si $\|u\|^2 = 1$, entonces

$$Q = I_m - 2uu^T$$

Puede darse entonces la siguiente definición

Definición. Una reflexión de Householder es una matriz de la forma

$$Q = I - 2uu^T \qquad (u^T u = 1)$$

Estas matrices tienen una serie de propiedades que las hacen interesantes

Lema. Sea Q una reflexión de Householder, entonces

i) Q es simétrica $(Q = Q^T)$

ii) Q es ortogonal $(Q^T Q = I)$

La propiedad fundamental de estas transformaciones es que pueden usarse para hacer ceros. Es decir, dado $x \in \mathbb{R}^m$, se puede encon-

trar una matriz Q tal que

$$Qx = -\sigma e_1$$

donde σ es determinada de manera que

(5.1.1) $$|\sigma| = \| -\sigma e_1 \|_2 = \| Qx \|_2 = \| x \|_2$$

Cómo calcular u de Q lo dice el teorema siguiente

Teorema 1. Sea $x \in \mathbb{R}^n$, $\sigma = \text{signo}(x_1) \| x \|_2$ y supongamos $x \neq \sigma e_1$. Tomemos

$$u = x + \sigma e_1$$
$$\pi = \tfrac{1}{2} \| u \|_2^2$$

entonces $Q = I - \pi^{-1} u u^T$ es una reflexión y

$$Qx = -\sigma e_1$$

Demostración

$$\begin{aligned} \pi &= \tfrac{1}{2} \| u \|_2^2 = \tfrac{1}{2} u^T u \\ &= \tfrac{1}{2} (x + \sigma e_1)^T (x + \sigma e_1) \\ &= \tfrac{1}{2} (x^T x + \sigma x^T e_1 + \sigma e_1^T x + \sigma^2 e_1^T e_1) \\ &= \tfrac{1}{2} (x^T x + 2\sigma x_1 + \sigma^2) \end{aligned}$$

Por (5.1.1) se sigue que

$$\begin{aligned} \pi &= \tfrac{1}{2} (2\sigma^2 + 2\sigma x_1) \\ &= \sigma^2 + \sigma x_1 \end{aligned}$$

Por consiguiente

$$Qx = x - \frac{uu^T}{\pi}x$$

$$= x - \frac{(x+\sigma e_1)(x+\sigma e_1)^T}{\pi}x$$

$$= x - \frac{(x+\sigma e_1)(x^Tx+\sigma e_1^Tx)}{\pi}$$

$$= x - \frac{(x+\sigma e_1)(\sigma^2+\sigma x_1)}{\sigma^2+\sigma x_1}$$

$$= x-(x+\sigma e_1)$$

$$= -\sigma e_1 \blacksquare$$

Una vez determinada una reflexión Q, conviene apuntar que calcular el producto QS, con S denotada por columnas $S=[s_1|s_2|,\ldots,|s_m]$

$$QS=[Qs_1|Qs_2|,\ldots,|Qs_m]$$

se reduce a calcular m-productos de la forma

$$QS=(I-\pi^{-1}uu^T)S=S-(\pi u^Ts)u.$$

Por medio de una reflexión de Householder podemos transformar un vector dado a otro de igual norma con ceros a partir de uno de

sus componentes. En efecto, tenemos el siguiente

Corolario. Sea s un vector dado. Definamos

$$\alpha = sign(s_k)(s_k^2 + s_{k+1}^2 + \cdots + s_m^2)^{1/2}$$

$$u = (0, \ldots, 0, s_k + \alpha s_{k+1}, \ldots, s_m)^T$$

$$\beta = \alpha u_k$$

Entonces $Q = I_m - \frac{1}{\beta} u u^T$ es una reflexión de Householder que deja fijos los primeros $(k-1)$ componentes del vector s, cambia la componente s_k a $-\alpha$ y anula los restantes s_i. $i = k+1, \ldots, m$ Esto es.

$$Qs = (s_1, \ldots, s_{k-1}, -\alpha, 0, \ldots, 0)^T$$

Con estos elementos se puede describir la forma de encontrar la descomposición de la matriz S^T mediante reflexiones de Householder en el producto

$$S^T = LQ$$

donde L es una matriz triangular inferior y Q una matriz ortogonal.

5.2 Descomposición QR y LQ

La descomposición ortogonal de una matriz S ha sido usada ampliamente en la solución de sistemas sobredeterminados

$$Sx = b$$

donde S es una matriz de $m \times n$, con $m > n$.

Este método de descomposición es análogo al de eliminación gaussiana, en el sentido de ir aplicando transformaciones (en este caso, ortogonales), a la izquierda de la matriz S (Ver Forsythe, G.E and Moler, C.B. [22], Stewart, G.W. [56], Romero R.[51]).

5.2A Descripción del Método

Antes de hacer la descripción de la factorización ortogonal de una matriz S, demos el siguiente

Teorema (Descomposición QR). Sea S una matriz de $m \times n$, con $m > n$. Entonces existe una matriz ortogonal Q^T, la cual es el producto de reflexiones de Householder

$$Q^T = Q_m Q_{m-1} \cdots Q_1$$

de modo que

$$Q^T S = \begin{bmatrix} U \\ \hline 0 \end{bmatrix}$$

donde U es una matriz triangular superior

Demostración

Sean $S^{(1)} = S = (s_{ij})$; $S^{(j)} = (s_{ik}^{(j)})$ en el jésimo paso y $S^{(n+1)} = U$.

Por el teorema (1) de la sección anterior, hay una reflexión de Householder Q_1 tal que $Q_1 S^{(1)}$ es un múltiplo del vector e_1. Sea $S^{(2)} = Q_1 S^{(1)}$. Por el corolario (1), también de la Secc. anterior, hay una reflexión de Householder que anula las componentes de la segunda columna de $S^{(2)}$, excepto los dos primeros. Esquemáticamente

$$S^{(3)} = Q_2 S^{(2)} = \begin{bmatrix} s_{11}^{(3)} & s_{12}^{(3)} & s_{13}^{(3)} & \cdots & s_{1n}^{(3)} \\ 0 & s_{22}^{(3)} & s_{23}^{(3)} & \cdots & s_{2n}^{(3)} \\ \vdots & 0 & s_{33}^{(3)} & \cdots & s_{3n}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & s_{m3}^{(3)} & \cdots & s_{mn}^{(3)} \end{bmatrix}$$

Continuando en esta forma, en el j-ésimo paso, una transformación de Householder es aplicada a la izquierda de $S^{(j)}$ para eliminar sus componentes $S^{(j)}_{k,j}$, $k=j+1,\ldots,m$, sin alterar las primeras. Esto es

$$S^{(j+1)} = Q_j S^{(j)} = \begin{bmatrix} s^{(j+1)}_{11} & s^{(j+1)}_{12} & \cdots & s^{(j+1)}_{1j} & \cdots & s^{(j+1)}_{1n} \\ 0 & s^{(j+1)}_{22} & \cdots & s^{(j+1)}_{2j} & \cdots & s^{(j+1)}_{2n} \\ \vdots & \ddots & \ddots & \vdots & & \vdots \\ & & & s^{(j+1)}_{jj} & \cdots & s^{(j+1)}_{jn} \\ & & & 0 & s^{(j+1)}_{j+1,j+1} & s^{(j+1)}_{j+1,n} \\ & & & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & s^{(j+1)}_{m,j+1} & s^{(j+1)}_{m,n} \end{bmatrix}$$

En el último paso, una reflexión Q_n cancela los componentes s_{kn}, $k=n+1,\ldots,m$ de $S^{(n+1)}$ sin alterar los anteriores, i.e

$$S^{(n+1)} = Q_n S^{(n)} = \begin{bmatrix} s^{(n+1)}_{11} & s^{(n+1)}_{12} & \cdots & s^{(n+1)}_{1n} \\ 0 & s^{(n+1)}_{22} & \cdots & s^{(n+1)}_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ & & & s^{(n+1)}_{nn} \\ & & & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}$$

Por lo que se tiene

$$S^{(n+1)} = Q_n Q_{n-1} \cdots Q_1 S^{(1)}$$

Tomando $Q^T = Q_n Q_{n-1} \cdots Q_1$ se tiene que

$$Q^T S = \begin{bmatrix} U \\ \hline 0 \end{bmatrix}$$

y puesto que Q es ortogonal, entonces

$$S = Q \begin{bmatrix} U \\ \hline 0 \end{bmatrix}$$

<u>Corolario</u> (Descomposición LQ). Sea S una matriz de dimensiones $n \times m$ ($m > n$). Existen matrices L triangular inferior y Q ortogonal tales que

$$S^T = [L|0]\, Q$$

En efecto, basta aplicar el teorema anterior a la matriz S^T.

En todo lo que sigue trataremos con matrices cuadradas B de orden m. En este caso, las descomposiciones QR y LQ toman la forma

$$B = QR \qquad (\text{o } B = LQ)$$

$$B^T = LQ \qquad (\text{o } B^T = QR)$$

con R triangular superior, L triangular inferior y Q ortogonal

5.3 Versión de Saunders del Simplex

El algoritmo propuesto por M.A. Saunders [55], difiere de las versiones estandar y está basado en la factorización de la matriz básica B en el producto

$$B = LQ$$

o equivalentemente en la factorización de Cholesky de BB^T

$$BB^T = LL^T.$$

Siguiendo las ideas de Gill-Murray [27], Saunders propone un algoritmo que en brere es como sigue

Resuelve el sistema transpuesto

(5.3.1) $$B^T\lambda = c_B$$

mediante la descomposición QR de B^T, vía la descomposición LQ de B.

Resuelve el sistema directo

(5.3.2) $$By = a_j$$

aprovechando la descomposición LQ de B, resolviendo el sistema

$$B^TBw = a_j$$

o equivalentemente

$$L L^T w = a_j$$

pues claramente

$$y = B^T w$$

es la solución de (5.3.2)

Actualiza la solución básica x de

$$Bx = b$$

En lo que sigue de la presente sección, se discute desde la perspectiva del análisis numérico los aspectos centrales de cada uno de los pasos anteriores.

La importancia de emplear la descomposición ortogonal de la matriz básica B, radica en su estabilidad numérica, Stewart [56], siendo por ello superior a los métodos de eliminación los cuales son sólo en la práctica numéricamente estables (ver Sección 2.2.C).

La resolución del sistema $B^T \lambda = c_B$, como se decía más arriba, se lleva a cabo mediante la descomposición QR de B^T, lo que corresponde a un proceso numéricamente

estable. Por ello en lo que sigue solamente se discutirá el sistema directo $By = a_j$.

Debido a que los problemas en Programación Lineal son de grandes dimensiones, la solución de (5.3.2) por el método QR requiere de almacenar la matriz ortogonal Q. La siguiente alternativa supera esta dificultad.

Considerando la descomposición $B = LQ$ o bien $QB^T = R$ (R matriz triangular superior), se tiene

$$BB^T = LL^T$$

Entonces puede verse que sólo necesitamos un factor L y no necesitamos almacenar Q. Debe hacerse notar que el producto BB^T no se calcula. Por lo anterior se plantea su uso en la solución de problemas de Programación Lineal.

La solución del sistema $Bx = b$, por el "método Q" así llamado por M.A. Saunders, está dada como

$$x = Q^T L^{-1} b$$

o bien podemos obtenerla resolviendo

$$BB^T w = LL^T w = b$$

resolviendo los sistemas triangulares

$$Ly = b$$
$$L^T w = y$$

y luego tomando

$$x = B^T w$$

Estos dos enfoques aunque en aritmética real son equivalentes, cuando se implementan en una computadora digital, los resultados pueden ser diferentes a causa de los errores por redondeo.

5.4 Consideraciones Numéricas

En ocasiones se ha pensado que la segunda variante da resultados poco confiables y por consiguiente dudar de su aplicación. Esto se presenta por ejemplo con las ecuaciones normales.

A continuación veremos que la mencionada anterior desconfianza esta sólo parcialmente justificada. Para aclarar lo anterior discutiremos dos maneras de enfocar la resolución del sistema

(5.4.1) $$BB^T w = b$$

para hallar la solución del sistema $Bx = b$,

mediante $x = B^T w$, al considerar que los errores por redondeo en $B^T w$ son despreciables.

Sea B no singular y el sistema $BB^T w = b$ es resuelto en aritmética de punto flotante con precisión ε con la cual obtenemos $\bar{w}$. La pregunta ¿ qué tanto $\bar{x} = B^T \bar{w}$ se aproxima a $x = B^T w$?. Esta pregunta en el entendido de que los errores por redondeo en el cálculo de $B^T \bar{w}$ los podemos despreciar. El sistema (5.4.1) podemos resolverlo en dos formas.

La primera consiste en calcular directamente la descomposición de Cholesky LL^T de la matriz positiva definida BB^T y resolver con dicha descomposición. Ahora del análisis de error por redondeo se tiene que $\bar{w}$ satisface

$$(BB^T + E)\bar{w} = b$$

donde $\|E\|_2 = \varepsilon \leq f(n)\varepsilon$

aquí $f(n)$ es una función de n.

La anterior expresión la podemos poner como

$$B(I + B^{-1}EB^{-T})B^T\bar{w} = b = BB^T w$$

Multiplicando por B^{-1} tenemos

$$B^T\bar{\omega} + B^{-1}EB^{-T}B^T\bar{\omega} = B^T\omega \Rightarrow$$
$$B^T\omega - B^T\bar{\omega} = B^{-1}EB^{-T}B^T\bar{\omega}$$

y

$$\|x-\bar{x}\|_2 = \|B^T\omega - B^T\bar{\omega}\|_2 = \|B^{-1}EB^{-T}B^T\bar{\omega}\|_2 \leq$$
$$\leq \|B^{-1}\|_2 \|E\|_2 \|\bar{\omega}\|_2 = \chi\varepsilon\|\bar{\omega}\|_2$$

donde χ es el número de condición de B, y bajo el supuesto de que $\|B\|_2 = 1$.

Como $\bar{\omega} = B^{-T}\bar{x}$, entonces

$$\|x-\bar{x}\|_2 \leq \chi\varepsilon\|\bar{\omega}\|_2 \leq \chi\varepsilon\|B^T\|_2\|\bar{x}\|_2 = \chi^2\varepsilon\|\bar{x}\|_2$$

La solución de $BB^T\omega = b$, podemos también obtenerla resolviendo primero

$$By = b$$

y a continuación

$$B^T\omega = y$$

empleando para ello por ejemplo la descomposición triangular con pivoteo. En este caso $\bar{\omega}$ satisface

$$(B+E_2)(B^T+E_3)\bar{\omega} = b$$

donde $\|E_i\|_2 = \varepsilon_i \leq f(n)\Sigma\varepsilon$

en donde Σ depende del $\max\{\hat{b}_{ij}\}$ (componente máximo que aparece en la descomposición de

la matriz $\mathcal{B}$). Por consiguiente

$$\mathcal{B}x = \mathcal{B}(I + \mathcal{B}^{-1}E_2)(I + E_3\mathcal{B}^{-T})\mathcal{B}^T\bar{\omega} = b$$

así que

(5.4.2) $$\|x - \bar{x}\|_2 \leq (\chi\varepsilon_2 + \chi\varepsilon_3 + \chi^2\varepsilon_2\varepsilon_3)\|\bar{x}\|_2$$

y si $\chi\varepsilon_i$ es suficientemente pequeño para que $\chi^2\varepsilon_2\varepsilon_3$ sea despreciable con respecto a $\chi\varepsilon_2$ y $\chi\varepsilon_3$ de la precisión de punto flotante en cuestión, entonces estas cotas son del orden de magnitud como si hubieramos obtenido la solución directamente.

Cuando se calcula la descomposición LQ de la matriz $\mathcal{B}$ ya sea mediante rotaciones de Givens o reflexiones de Householder, se puede mostrar (Véase a Wilkinson [63], Paige [44]), que satisface

$$\mathcal{B} + E = \hat{L}\hat{Q}$$

donde $\|E\|_2 \approx \varepsilon$

siendo ε la precisión de la máquina.

Entonces la solución $\bar{\omega}$ de $\hat{L}y = b$, y $\hat{L}^T\omega = y$, satisface la relación

$$(\hat{L} + E_1)(\hat{L}^T + E_2)\bar{\omega} = b$$

donde $\| E_i \|_2 = \varepsilon_i \approx \varepsilon$

en tanto que para $\bar{x}$ se tiene

$$\bar{x} = (B^T + E_3)\,\bar{w}$$

donde $\| E_3 \| \approx \varepsilon$

Por tanto la relación (5.4.2) se sostiene para el "procedimiento L".

En resumen, el segundo procedimiento es razonablemente aceptable con respecto a la calidad de la solución del sistema $Bx = b$ cuando éste es resuelto directamente.

5.5 Actualización de la Factorización LQ

Al igual que como hicimos con los métodos de eliminación, en este caso calculamos inicialmente una factorización ortogonal de la matriz básica B, y en la siguiente iteración al efectuar el cambio de base, no necesitamos repetir todo el proceso para obtener la descomposición ortogonal de la nueva base $\hat{B}$, sino que aprovechamos la ya disponible y solamente la actualizamos. Este proceso de actualizar la descomposición ortogonal será discutido en esta sección.

Para minimizar el cálculo de raíces cuadradas y divisiones durante cada iteración, optaremos por la descomposición ortogonal de BB^T en la forma

(5.5.1) $$BB^T = LDL^T$$

con D una matriz diagonal ($D = \text{diag}(d_i)$, $d_i > 0$), y L triangular inferior.

Teniendo la descomposición (5.5.1), se quiere saber cómo pasar de B a una nueva $\bar{B}$, a

través de su factorización

$$\bar{L}\bar{D}\bar{L}^T$$

cuando LDL^T sufre dos modificaciones

a) $LDL^T - a_s a_s^T = \hat{L}\hat{D}\hat{L}^T$

debida a quitarle a $\mathcal{B}$ su a_s columna, y

b) $\hat{L}\hat{D}\hat{L}^T + a_k a_k^T = \bar{L}\bar{D}\bar{L}^T$

debida a agregarle a $\mathcal{B}$ la a_k columna de A.

Los pasos (a) y (b), se pueden ver como uno solo, pues a través de

$$LDL^T + \alpha VV^T$$

podemos obtener la factorización

$$\bar{L}\bar{D}\bar{L}^T$$

de la nueva matriz básica $\bar{\mathcal{B}}$, tomando $\alpha = -1$ para el primer caso, y $\alpha = 1$ para el último.

A continuación se bosquejan dos formas de actualizar la factorización

$$LDL^T + \alpha VV^T$$

5.5.A Actualización de LDL^T según Saunders por el Método Directo.

Sea p un vector dado por

$$Lp = v$$

entonces

$$\bar{L}\bar{D}\bar{L}^T = LDL^T + \alpha (Lp)(Lp)^T$$
$$= LDL^T + \alpha\, Lpp^T L^T$$
$$= L(D + \alpha pp^T)L^T$$

Poniendo $M\Delta M^T = D + \alpha pp^T$, se tiene

$$\bar{L}\bar{D}\bar{L}^T = L(M\Delta M^T)L^T$$
$$= (LM)\Delta(LM)^T$$

Entonces los factores modificados son

$$\bar{L} = LM \quad ; \quad \bar{D} = \Delta$$

donde M es una matriz de la forma

$$M = \begin{bmatrix} 1 & & & & \\ p_2\beta_1 & 1 & & & \\ p_3\beta_1 & p_3\beta_2 & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ p_m\beta_1 & p_m\beta_2 & \cdots & p_m\beta_{m-1} & 1 \end{bmatrix}$$

A continuación veremos que M es una matriz triangular inferior "especial", definida por dos vectores p y β como sigue

(5.5.2) $\hat{l}_{rs} = p_r \beta_s$; $r = j, j+1, \ldots, m$; $s = 1, \ldots, j-1$

cuyos componentes se determinan por el método directo. Esto es

Explícitamente, la j-ésima columna de la factorización $\hat{L}\hat{D}\hat{L}^T$ da lugar a las ecuaciones para $\hat{d}_j$ y $\hat{l}_{rj}$, $r = j+1, \ldots, m$

$$\sum_{i=1}^{j-1} \hat{d}_i \hat{l}_{ji}^2 + \hat{d}_j = d_j + \alpha p_j^2$$

$$\sum_{i=1}^{j-1} \hat{d}_i \hat{l}_{ji} \hat{l}_{ri} + \hat{d}_j \hat{l}_{rj} = \alpha p_j p_r \quad , r = j+1, \ldots, m$$

Usando la ecuación (5.5.2) en estas dos últimas ecuaciones se tiene

$$\sum_{i=1}^{j-1} \hat{d}_i p_j^2 \beta_i^2 + \hat{d}_j = d_j + \alpha p_j^2$$

$$p_j^2 \sum_{i=1}^{j-1} \hat{d}_i \beta_i^2 + \hat{d}_j = d_j + \alpha p_j^2$$

y

$$\sum_{i=1}^{j-1} \hat{d}_i p_j \beta_i p_r \beta_i + \hat{d}_j \hat{l}_{rj} = \alpha p_j p_r$$

$$p_j p_r \sum_{i=1}^{j-1} \hat{d}_i \beta_i^2 + \hat{d}_j \hat{l}_{rj} = \alpha p_j p_r$$

Despejando $\hat{l}_{rj}$ de esta última ecuación

$$\hat{l}_{rj} = \frac{p_j}{\hat{d}_j}\left(\alpha - \sum_{i=1}^{j-1} \hat{d}_i \beta_i^2\right) p_r \quad , \quad r = j+1, \ldots, m.$$

definiendo $\beta_j = \dfrac{p_j}{\hat{d}_j}\left(\alpha - \sum_{i=1}^{j-1} \hat{d}_i \beta_i^2\right)$ obtenemos que

$$\hat{l}_{rj} = p_r \beta_j \quad , \quad r = j+1, \ldots, m.$$

Esto lo que nos dice es que los componentes de la subdiagonal de la j-ésima columna de M son múltiplos de los correspondientes componentes del vector p.

En la práctica primero se calcula la cantidad

$$\alpha_j - \sum_{i=1}^{j-1} \hat{d}_i \beta_i^2$$

y el cálculo de los α_j, $\hat{d}_j$ y β_j se obtienen con el algoritmo

$\alpha_1 \leftarrow \alpha$

Para $j = 1, \ldots, m$

$\hat{d}_j \leftarrow d_j + \alpha_j p_j^2$

$\beta_j \leftarrow \alpha_j p_j / \hat{d}_j$

$\alpha_{j+1} \leftarrow \alpha_j d_j / \hat{d}_j$

5.5.B Actualización de LDL^T vía Transformaciones Ortogonales según Gill-Murray

Otro método para construir M y Δ ha sido propuesto por Gill-Murray [27] usando matrices Hermitianas. Este método tiene ventajas numéricas cuando $LDL^T + \alpha vv^T$ es casi singular y cuando $\alpha < 0$ (compárese con el anterior algoritmo).

Los detalles del método pueden verse en Saunders [55], Gill, Golub, Murray, Saunders [30], aquí solamente resumiremos como podría quedar el algoritmo para contruir M y Δ.

Algoritmo

$\alpha_1 \leftarrow \alpha$

$s_1 \leftarrow p^T D^{-1} p = \sum_{j=1}^{m} p_j^2 / d_j$

$\sigma_1 \leftarrow \alpha / [1 + (1 + \alpha s_1)^{1/2}]$

Para $j = 1, \ldots, m$

$q_j \leftarrow p_j^2 / d_j$

$\theta_j \leftarrow 1 + \sigma_j q_j$

$$
\begin{aligned}
s_{j+1} &\leftarrow s_j - q_j \\
\gamma_j^2 &\leftarrow \theta_j^2 + \sigma_j q_j s_{j+1} \\
\hat{d}_j &\leftarrow \gamma_j^2 d_j \\
\beta_j &\leftarrow \alpha_j p_j / \hat{d}_j \\
\alpha_{j+1} &\leftarrow \alpha_j / \gamma_j^2 \\
\sigma_{j+1} &\leftarrow \sigma_j (1 + \gamma_j) / [\gamma_j (\theta_j + \gamma_j)]
\end{aligned}
$$

Debemos mencionar que en ambos algoritmos α, p_j y d_j son datos de entrada para generar los β_j y $\hat{d}_j$ que definen a las matrices M y Δ.

El siguiente algoritmo es una transcripción del propuesto por M.A. Saunders [55].

En el algoritmo hemos supuesto que se va a efectuar la $(k+1)$-ésima iteración.

Algoritmo

1. BTRAN1 (Backward Transformation 1)
   Se resuelve el sistema $L_k^T \pi = \gamma_k$
   i.e. $M_{2k}^T \cdots M_2^T M_1^T L_0 \pi = \gamma_k$
2. PRICE Se lee la matriz A y se calculan los costos reducidos para las variables no básicas. Seleccione la columna a_s

para la cual $c_s - \pi^T a_s < 0$

3. FTRAN1 (Forward Transformation 1)
Resuelva $L_k p_1 = a$, i.e. $L_0 M_1 M_2 \cdots M_{2k} p_1 = a_s$

4. UPDATE1 Calcule $w = D_k^{-1} p_1$
Use p_1 para calcular β_1 y modificar D_k.
Almacene los componentes diferentes de cero de la pareja de vectores (p_1, β_1) y añádalos al final del archivo donde se almacena L.

5. BTRAN2 Resuelva el sistema $L_k^T u = w$

6. READB Lea la matriz B para calcular $y = B^T u$.
Añada la columna a_s al archivo donde se almacena B

7. CHUZR Use $\hat{x}$ y y para determinar la columna a_r que será removida de B. Encuentre $\theta = \hat{x}_r / y_r$

8. SEEKR Actualización de la solución $\hat{x}$ de acuerdo a: $\hat{x} \leftarrow \hat{x} - \theta y$

9. FTRAN2 Resuelva $L_k M_{2k+1} p_2 = a_r$

10. UPDATE2 Use p_2 para calcular β_2, D_{k+1} y añada la pareja de vectores (p_2, β_2) al archivo donde se almacena L.

Antes de efectuar la descomposición ortogonal de alguna matriz básica B, conviene hacer la siguiente observación.

Sean P_1 y P_2 dos matrices para permutar renglones y columnas de B, entonces en la descomposición

$$P_1 B P_2 = L D^{1/2} Q$$

las matrices L y D son independientes de la matriz de permutación P_2. En efecto

$$(P_1 B P_2)(P_1 B P_2)^T = (L D^{1/2} Q)(L D^{1/2} Q)^T$$

o bien

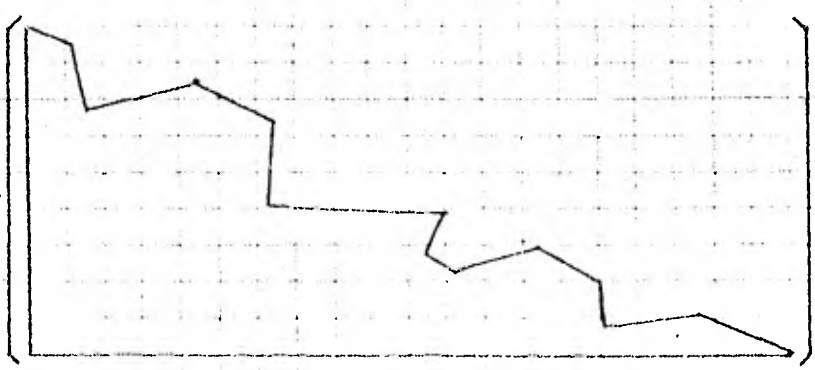
$$P_1 B B^T P_1^T = L D L^T$$

De ésto se desprende que si queremos una matriz triangular inferior L, lo más rala posible, debemos concentrarnos en ordenar únicamente los renglones de la matriz básica B. Por tanto lo recomendable es encontrar un ordenamiento óptimo de renglones dentro de cada bloque que resulte de aplicar alguna de las técnicas de ordenamiento.

5.6 Consideraciones Sparse

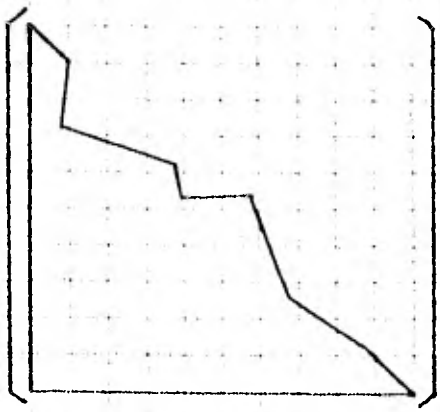
Como hemos visto, por limitaciones físicas de memoria en las máquinas digitales, y evitar cálculos numéricos innecesarios, conviene que la escasa cantidad de componentes diferentes de cero en las matrices de los problemas de Programación Lineal, sea agrupada de modo tal que ésta se localice por regiones. De este modo cuando se efectúen los cálculos numéricos con las matrices básicas B_j, $j=1,\ldots,k$, el llenado de las B_j estará localizado también por regiones.

Una forma general de agrupar la información de un problema consiste en llevarla a la parte triangular inferior

$PAQ=\hat{A}=$ 

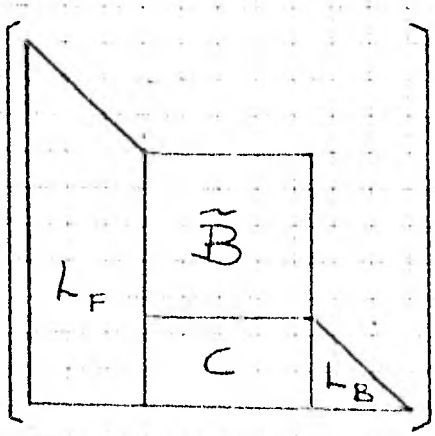
Con esta disposición de los elementos, en

las matrices básicas B_j, el llenado que se dé estará localizado en la parte triangular inferior.

$B_j =$ 

5.6.A Triangular por Bloques y Espigas

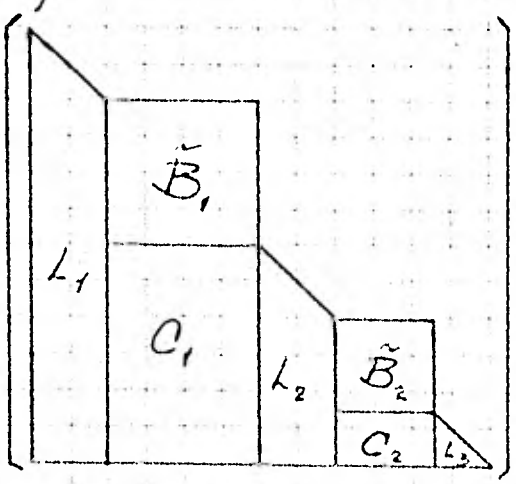
Otra forma de agrupar la información para amortiguar el llenado de las matrices básicas, consiste en llevar la información a una triangular inferior por bloques y espigas (Ver 3.6)

$B =$ 

donde L_F y L_B son columnas de B sin componentes sobre la diagonal principal,

$\tilde{B}$ es un bloque en general ralo y donde debemos aplicar la descomposición ortogonal. Para evitar aún más el llenado en B, podemos permutar los renglones y columnas de $\tilde{B}$, mediante la técnica de Hellerman-Rarick, y así conseguir bloques $\tilde{B}_i$ y espigas dentro de $\tilde{B}$ de la forma

$\tilde{B} =$



5.6.B Estructura por bloques

Debido a características típicas de los problemas de Programación Lineal, éstos poseen ciertas estructuras que conviene aprovechar. Con frecuencia, la disposición de los componentes en las matrices dan lugar a una estructura de la forma

$$A = \begin{bmatrix} B_1 & & & \\ & B_2 & & \\ & & \ddots & \\ & & & B_k \\ C & & & \end{bmatrix}$$

donde cada bloque B_i es aún ralo.

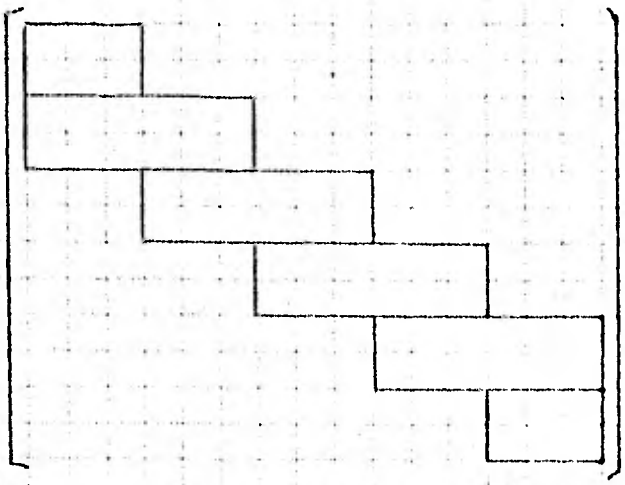
Teniendo esta estructura de datos, al efectuar la descomposición ortogonal, cada base tendrá una estructura similar. En la figura de abajo se muestra una base B y su correspondiente factor de Cholesky.

$$B = \begin{bmatrix} B_1 & & & \\ & B_2 & & \\ & & B_3 & \\ C & & & \end{bmatrix} \quad ; \quad L = \begin{bmatrix} L_1 & & & \\ & L_2 & & \\ & & L_3 & \\ M & & & \end{bmatrix}$$

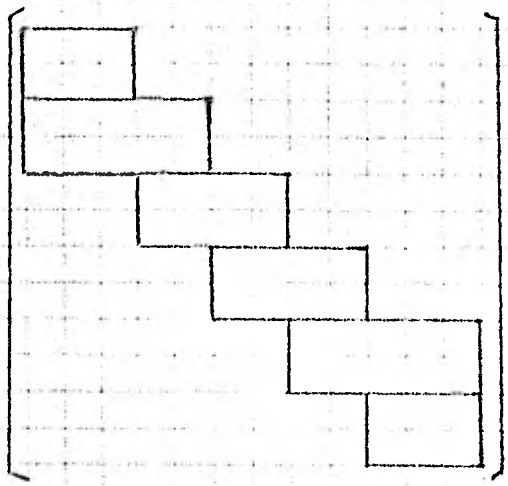
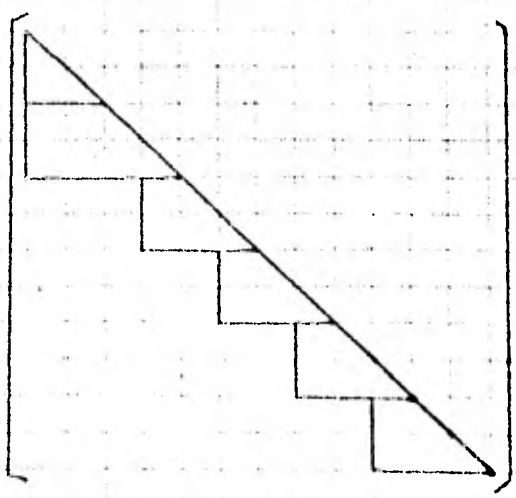
Con esta estructura de problemas, el llenado se encuentra localizado en las regiones L_i y M del factor de Cholesky L.

5.6.C Estructura de Escalera

Es común que las matrices de los problemas adopten la forma de una escalera

$A =$ 

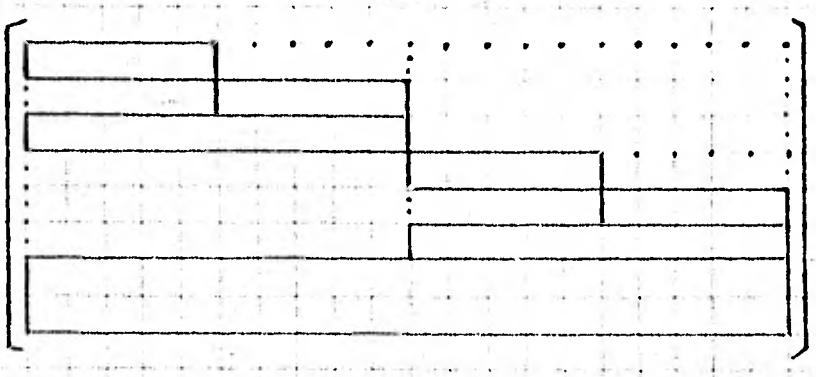
En esta estructura, los factores de Cholesky de las matrices básicas preservan la forma. Es decir, el llenado, al efectuar la factorización, está localizado en las regiones que forman los bloques de la matriz básica

$B =$  ; $L =$ 

5.6.D Angular por Bloques

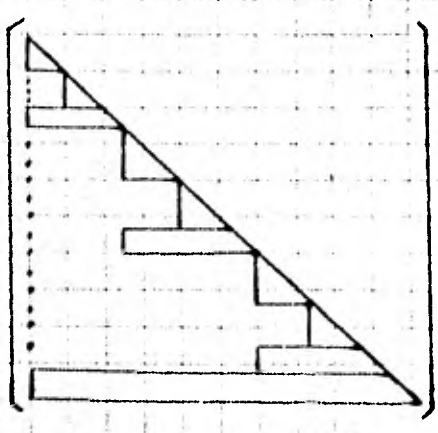
En otros problemas, sin embargo, no es aparente una disposición definida de los componentes de la matriz, así que una alternativa podría consistir en emplear el método de Weil-Kettler [ ], para arreglar la matriz a una forma angular por bloques.

$A =$



Es posible que aún dentro de cada bloque pueda aplicarse el mismo método, resultando que el factor de Cholesky de alguna matriz básica sea de la forma.

$L =$



Bibliografía

1. BARTELS, R.H., A Numerical Investigation of the Simplex Method, Tech. Rep. CS 104, Univ. Stanford (1968)

2. BARTELS, R.H., A Stabilization of the Simplex Method, Numer. Math., 16 (1971) 414-434

3. BARTELS, R.H., Large, Sparse Linear Programming (Notes for the Applied Matrix Methods Workshop)

4. BARTELS, R.H., & GOLUB, G.H., Algorithm: 350: Simplex Method Procedure Employing LU Decomposition, Comm. ACM, 12, 5 (1969) 275-278

5. BARTELS, R.H. & GOLUB, G.H., The Simplex Method of Linear Programming Using LU Decomposition, Comm. ACM, 12, 5, (1969) 266-268

6. BEALE, E.M.L., Sparseness in Linear Programming, en REID, J.K. (Ed.) Sparseness in Linear Programming, Academic Press (1971) 1-15

7. BRAMELER, A. & ALLAN, R.N., The Role of Sparsity in the Analysis of Large Systems, Comp. Aided Des, 6 (1974)

8. BEAYTON, R.K., GUSTAVSON, F.G. & WILLOUGHBY, R.A., Some Results on Sparse Matrices, Math. of Comp., 24, 112 (1970) 937-954

9. BUCHET, JACQUES DE, How to Take Into Account the Low Density of Matrices to Design Mathematical Programming Package, Relevant Effects on Optimization and Inversion Algorithms, en REID, J.K., (Ed.) Sparseness in Linear Programming, Academic Press (1971) 211-217

10. BUNCH, J.R., Block Methods for Solving Sparse Linear Systems, en BUNCH, J.R. & ROSE, D.J., (Ed.) Sparse Matrix Computations, Academic Press (1976) 39-58

11. BUNCH, J.R., Complexity of Sparse Elimination, en TRAUB, J.F. (Ed.) Complexity of Sequential and Parallel Numerical Algorithms, Academic Press (1973) 197-220

12. CLASEN, R.J., Techniques for Automatic Tolerance Control in Linear Programming, Comm. ACM, 9, 11, (1966) 802-803

13. CURTIS, A.R. & REID, J.K., On the Automatic Scaling of Matrices for Gaussian Elimination, Theoretical Physics Div., UKAEA, Research Group, AERE Harwell (1971)

14. DANTZING, G.B., Compact Basis Triangularization for the Simplex Method, en GRAVES, R.L. & WOLFE, P. (Eds.) Recent Advances in Mathematical Programming, Mc Graw-Hill (1963) 125-132

15. DANTZING, G.B., Linear Programming and Extensions, Princeton Univ., Princeton, N.J. (1963)

16. DUFF, I.S., On Permutations to Block Triangular Form, J. Inst. Math. Applics., 19 (1977) 339-342

17. DUFF, I.S. & REID, J.K., A Comparision of Sparsity Orderings for Obtaining a Pivotal Sequence in Gaussian Elimination, J. Inst. Math. Applics., 14 (1974) 281-291

18. DUFF, I.S. & REID, J.K., An Implementation of Tarjan's Algorithm for the Block Triangularization of a Matrix, ACM Transact. Math. Soft., 4, 2 (1978) 137-147

19. DUFF, I.S. & REID, J.K., Some Design Features of a Sparse Matrix Code, Tech. Rep. CSS 48, Computer Sci. & Systems Div., AERE Harwell (1977)

20. ERISMAN, A.M. & REID, J.K., Monitoring the Stability of the Triangular Factorization of a Sparse Matrix, Tech. Rep. TP 525, AERE Harwell (1973)

21. FORREST, J. J. H. & TOMLIN, J.A., Updated Triangular Factors of the Basis to Maintain Sparsity in the Product Form Simplex Method, Math. Prog. 2 (1972) 263 - 278

22. FORSYTHE, G. E. & MOLER, C.B., Compoter Solotion of Algebraic Systems, Prentice-Hall, Englewood Cliffs N. J. (1967)

23. FORSYTHE, G. E., MALCOLM, M. A. & MOLER, C. B., Computer Methods for Mathematical Computations, Prentice-Hall, Englewood Cliffs., N. J. (1977)

24. FOX, L., An Introduction to Numerical Linear Algebra, Oxford Univ. Press, London (1964)

25. GAY, D. M., On combining the Schemeds of Reid and Saunders for Sparse LP Bases, Sloan School Working Paper 1037-79 (1979)

26. GILL, P. E. A., Recent Developments in Numerically Stable Methods of Linear Programming, Inst. Math. and Applics. (1974) 180 - 186

27. GILL, P. E. & MURRAY, W., A Numerically Stable Form of the Simplex Algorithm, Lin. Alg. and its Applics., 7, (1973) 99 - 138

28. GILL, P.E. & MURRAY, W., The Ortogonal Factorization of a Large Sparse Matrix, en BUNCH, J.R. & ROSE, D.J. (Eds.) Sparse Matrix Computations, Academic Press (1976) 201-212

29. GILL, P.E., MURRAY, W. & SAUNDERS, M.A., Methods for Computing and Modifying the LDV Factors of a Matrix, Math. of Comp., 29, 132 (1975) 1051-1077

30. GILL, P.E., GOLUB, G.H., MURRAY, W. & SAUNDERS, M.A., Methods for Modifying Matrix Factorizations, Math. of Comp. 28, 126 (1974) 505-535

31. GOLFARB, D., On the Bartels-Golub Decomposition for Linear Programming Bases, Math. Prog., 13 (1977) 272-279

32. GOLFARB, D., Using the Steepest Edge Simplex Algorithm to Solve Sparse Linear Programs, en BUNCH, J.R. & ROSE, D.J. (Eds.) Sparse Matrix Computations, Academic Press (1976) 227-240

33. GOLFARB, D. & REID, J.K., A practicable Steepest-Edge Simplex Algorithm, Tech. Rep. CSS 19, Sci. & Systems Div. AERE Harwell (1975)

34. GUSTAVSON, F., Finding the Block Lower Triangular Form of a Sparse Matrix, en BUNCH, J.R. & ROSE, D.J. (Eds) Sparse Matrix Computations, Academic Press (1976) 235-289

35. HADLEY, G., Linear Programming, Addison-Wesley Publishing Co., Reading Mass. (1963)

36. HARRIS, P.M.J., Pivot Selection Methods of the Devex LP Code, Math. Prog. 5 (1973) 1-28

37. HELLERMAN, E. & RARICK, D., Reinversion with the Preasigned Pivot Procedure, Math. Prog. 1 (1971) 195-216

38. HOWELL, T.D., Partitioning using PAQ, en BUNCH, J.R. & ROSE, D.J. (Eds) Sparse Matrix Computations, Academic Press (1976) 23-37

39. LUENBERGER, D.G., Introduction to Linear and non Linear Programming, Addison-Wesley Publishing Co, Reading Mass (1973)

40. MARKOWITZ, H.M., The Elimination Form of the Inverse and its Applications to Linear Programming Manag. Sci., 3 (1957) 255-269

41. Mc BRIDE, R.D., A Bomp Triangolar Dynamic Factorization for the Simplex Method, Math. Prog., 18 (1980) 49-61.

42. ORCHARD-HAYS, W., Advanced Linear Programming Computing Techniques, Mc Graw-Hill, N.Y. (1968)

43. ORCHARD-HAYS, W., Evolution of Linear Programming Techniques, Manag. Sci., 4 (1958) 183-198

44. PAIGE, C.C., An Error Analysis of a Method for Solving Matrix Equations, Math. of Comp. 27, 122. (1973) 335-339

45. PARTER, S., The Use of Linear Graphs in Gauss Elimination, SIAM Rev., 3, 2 (1961) 119-129

46. PETERS, G. & WILKINSON, J.H., On the Stability of Gauss-Jordan Elimination with Pivoting, Comm. ACM, 18, 1 (1975) 20-24

47. POWELL, S., A Development of the Product Form Algorithm for the Simplex Method Using Reduced Transformation Vectors, Math. Prog. 4 (1975) 93-107

48. REID, J.K., A Note on the Stability of Gaussian Elimination, Tech. Rep. TP441 AERE, Harwell (1971)

49. REID, J.K., A Sparsity-Exploiting Variant of the Bartels-Golub Decomposition for Linear Programming Bases, Tech. Rep. CSS 20, AERE, Harwell, Didcot, Oxfordshire (1975)

50. REID, J.K., Fortran Subroutines for Handling Sparse Linear Programming Bases, Computer Sci. & Systems Division, AERE Harwell, Didcot, Oxfordshire, (1976)

51. ROMERO, E.A., Aspectos Numéricos del Método de Mínimos Cuadrados (por aparecer)

52. ROSE, D.J., Triangulated Graphs and the Elimination Process, J. of Math. Anal. and Applics., 32 (1970) 597-609

53. SAUNDERS, M.A., A Fast, Stable Implementation of the Simplex Method Using Bartels-Golub Updating, en BUNCH, J.R. & ROSE, D.J. (Eds.) Sparse Matrix Computations, Academic Press (1976) 213-226

54. SAUNDERS, M.A., Large-Scale Linear Programming Using the Cholesky Factorization, Tech. Rep. STAN-CS-72-252, Univ. Stanford (1972)

55. SAUNDERS, M.A., Product Form of the Cholesky Factorization for Large-Scale Linear Programming, Tech. Rep., STAN-CS-72-301, Univ. Stanford (1972)

56. STEWART, G.W., Introduction to Matrix Computations, Academic Press, N.Y., (1973)

57. SVERRE, S., Error Control in the Simplex Technique, BIT, 7 (1967) 216-225

58. TARJAN, R.E., Graph Theory and Gaussian Elimination, en. BUNCH, J.R. & ROSE, D.J. (Eds.) Sparse Matrix Computations, Academic Press (1976) 3-22

59. TEWARSON, R.P., Sparse Matrices, Academic Press, N.Y. (1973)

60. TOMLIN, J.A., Modifying Triangular Factors of the Basis in the Simplex Method, en ROSE, D.J. & WILLOUGHBY, R.A. (Eds.) Sparse Matrices and their Applications, Plenum Press, (1972) 77-85

61. TOMLIN, J.A., On Pricing and Backward Transformation in Linear Programming, Math. Prog., 6 (1974) 42-47

62. WEIL, R.L. & KETTLER, P.C., Rearranging Matrices to Block-Angular Form for the Decomposition (and other) Algorithms, Manag. Sci, 18, 1 (1971) 98-108

63. WILKINSON, J.H., The Algebraic Eigenvalue Problem, Oxford University Press, London (1965)

64. WOLFE, P., Error in the Solution of Linear Programming Problems, en RALL, B.L. (Ed.) Error in Digital Compotation, 2 (1965) 271-284