

Te
C25
1985

UNIVERSIDAD DEL VALLE DE GUATEMALA

FACULTAD DE CIENCIAS Y HUMANIDADES

BIBLIOTECA
DE LA
UNIVERSIDAD DEL VALLE DE GUATEMALA

*Algoritmos Matemáticos
de Programación Administrativa Aplicados a
Proyectos de Construcción Civil
y su Relación con la Computadora.*

Minor Alberto Cardona Escobar

BIBLIOTECA
DE LA
UNIVERSIDAD DEL VALLE DE GUATEMALA

GUATEMALA, 1985

Algoritmos Matemáticos
de Programación Administrativa Aplicados a
Proyectos de Construcción Civil
y su Relación con la Computadora

BIBLIOTECA
DE LA
UNIVERSIDAD DEL VALLE DE GUATEMALA

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

ALGORITMOS MATEMATICOS
DE PROGRAMACION ADMINISTRATIVA APLICADOS A
PROYECTOS DE CONSTRUCCION CIVIL
Y SU RELACION CON LA COMPUTADORA

MINOR ALBERTO CARDONA ESCOBAR

Trabajo de investigación presentado para optar al
grado académico de Licenciado en Ingeniería Civil

Guatemala

1985

BIBLIOTECA
DE LA
UNIVERSIDAD DEL VALLE DE GUATEMALA

Vº Bº:

(f)



Ingeniero Franklin Matzdorf
Asesor

Tribunals:

(f)


Ingeniero Richard Köhler Gebhardt

(f)


Ingeniero Roberto Solís Hegel

(f)


Ingeniero Franklin Matzdorf

Fecha de aprobación: 8 de junio de 1985

**AL PADRE, AL HIJO Y AL ESPIRITU SANTO,
TODA ALABANZA, HONRA, GLORIA, Y PODER,
POR LOS SIGLOS DE LOS SIGLOS.**

A LA MEMORIA DE MI PADRE.

A MI MADRE Y HERMANOS.

A MIS CUÑADOS Y SOBRINOS.

A MIS FAMILIARES.

A LA UNIVERSIDAD DEL VALLE.

A MIS CATEDRATICOS.

A MIS COMPAÑEROS Y AMIGOS.

AGRADECIMIENTO

QUIERO DEJAR CONSTANCIA DE MI AGRADECIMIENTO AL ING. FRANKLIN MATZDORF, POR SU ESPONTÁNEA, DECIDIDA, Y CONSTANTE ASESORÍA A LO LARGO DE LA ELABORACIÓN DE ESTE TRABAJO.

A LOS HERMANOS CANJURA GUZMÁN: GUILLERMO, ROGER, ROXANA Y FÁTIMA, Y A MI HERMANO SERGIO, POR SU COOPERACIÓN EN LA RECOPIACIÓN DEL MATERIAL BIBLIOGRÁFICO.

QUIERO AGRADECER MUY ESPECIALMENTE A MI AMIGO GERARDO MARTÍNEZ Y A SU MAMÁ, DOÑA LILIAN DE MARTÍNEZ, POR LA EXCELENTE IMPRESIÓN Y EL NÍTIDO LEVANTADO DE TEXTO.

FINALMENTE, EXPRESO MI AGRADECIMIENTO A MICHAEL GRÜNNER QUE, DE UNA FORMA U OTRA, CONTRIBUYÓ A LA CULMINACIÓN DE ESTE TRABAJO, Y A MI HERMANO MEME POR EL ENCUADERNADO Y LA IMPRESIÓN DE LAS CARÁTULAS.

PREFACIO

La búsqueda por lo mejor, lo máximo, lo mínimo o, en general, las soluciones óptimas para una variedad de problemas ha entretenido e intrigado al hombre a través de las edades.

Euclides en su libro III, estaba interesado en hallar la línea más larga y la más corta que pudieran dibujarse desde un punto de la circunferencia de un círculo. En su libro IV describe cómo hallar el paralelogramo de máxima área, con un perímetro dado. Sin embargo, el riguroso enfoque a éstos y más sofisticados problemas tuvo que esperar hasta que los grandes matemáticos de los siglos XVII y XVIII, desarrollaron los poderosos métodos del Cálculo Infinitesimal y del Cálculo de Variaciones. Con estas técnicas podemos hallar soluciones máximas y mínimas, a una amplia variedad de problemas de optimización.

Pero estos y otros procedimientos matemáticos para optimización, estaban relacionados a solucionar problemas de naturaleza física, dinámica o geométrica.

Desde hace muchos años, una nueva clase de problemas para optimización, ha producido la "Infraestructura Organizativa" que hace que nuestra moderna sociedad evolucione aceleradamente.

Y es aquí donde estamos hablando de asuntos tales como: la forma más eficiente en que se debe administrar una economía o el despliegue óptimo de aviones que maximizan la oportunidad de un país para ganar una guerra o, simplemente, con tareas tan triviales como la de mezclar los ingredientes de un fertilizante que reúnan las especificaciones agrícolas a un costo mínimo.

La búsqueda de "cómo formular y resolver tales problemas" ha conducido al desarrollo de una importante técnica de optimización: "La Programación Lineal", cuyo modelo es simple en su estructura matemática, pero poderoso en su adaptabilidad a un amplio rango de aplicaciones.

Históricamente el modelo general de la programación lineal fue desarrollado y aplicado en 1947 por un grupo de investigadores dirigidos por George Dantzig y Marshall Wood, del U. S. Department of the Air Force. En ese tiempo, el grupo fue llamado a investigar la factibilidad de aplicar técnicas matemáticas y afines a problemas de logística. Esta interrogante condujo a Dantzig a proponer que: "Las relaciones entre actividades de una gran organización, deben ser vistas como el tipo de modelo de una programación lineal, cuyo programa de optimización debe determinarse al minimizarse una función lineal primal*."

* La función objetivo.

Con el fin de extender y desarrollar estas ideas para el futuro, la fuerza aérea organizó a un grupo de investigadores bajo el título de PROJECT SCOOP (Scientific Computation of Optimum Program). Además de formular la programación de la fuerza aérea, y su colección de problemas, en una base científica, la mayor contribución del "SCOOP" fue el desarrollo formal y la aplicación del modelo de la programación lineal, cuyas primeras aplicaciones estaban orientadas a tres grandes categorías: Aplicaciones militares, economía inter-industrial y la teoría de juegos. En los últimos años, las áreas de aplicación se han extendido y desarrollado, pero el principal énfasis se ha visto en el área industrial, como por ejemplo el de la industria de la construcción, que es el tema que nos interesa. El enunciado inicial para el problema general de la programación lineal fue desarrollado durante el año 1947 por George Dantzig a través del Método Simplex, un procedimiento sistemático para resolver el problema. Antes de esto, un sin número de problemas se detectaron como el de la "Teoría de Desigualdades", que trata de optimizar una función lineal sujeta a restricciones de desigualdades lineales. Los primeros ejemplos incluían el problema del transporte, planteado por Hitchcock en 1941 y Koopman en 1907, así como el de la dieta, planteado por Stigler en 1945.

Posteriormente surgió la Teoría de Redes, como una extensión de la programación lineal, que permite representar problemas del mundo real en forma gráfica, así como la ayuda conceptual que otorga para la comprensión de relaciones entre eventos y objetos, usados en casi todos los campos científicos, económicos y sociales. Una de las principales razones para la difusión de la teoría de Redes es el avance de los métodos computacionales que permiten resolver problemas de enorme tamaño.

Las redes tienen innumerables aplicaciones como la de representar: sistemas de ríos, sistemas de distribución, cartas de flujo, ordenamiento de los eventos de precedencia, la administración y planificación de todo tipo de proyectos, etc.

Esencialmente una red es un diagrama esquemático que ilustra la secuencia e interrelación de las partes componentes de un proyecto.

En la industria de la construcción son muy utilizados los diagramas de redes para determinar el mejor programa de operación. Entre los métodos utilizados para determinar la ruta más larga de una red de un problema de ruta mínima, está el CPM, Pert, el algoritmo de Dijkstra, el algoritmo de Floyd, el método del doble barrido y, finalmente, el algoritmo primal simplex, que es el método de la programación lineal que vamos a utilizar para resolver redes

de problemas de ruta mínima, en su caso específico de la industria de la construcción. Para la aplicación del algoritmo Simplex, utilizamos el enfoque de Egerváry, para hallar una matriz de incidencia, y el trabajo de Ford-Fulkerson para redes de flujo.

CONTENIDO

Página N°

CAPITULO 1

1.0	PROGRAMACION LINEAL Y SU RELACION CON LAS REDES DE FLUJOS	1
1.1	Introducción.	1
1.2	Programación lineal.	6
1.3	Notación.	9
1.4	Enunciado clásico de la programación lineal.	12
1.5	Definiciones.	13
1.6	Convexidad.	14
1.7	Funciones lineales.	25
1.8	Teorema del punto extremo.	27

CAPITULO 2

2.0	HISTORIA DEL METODO SIMPLEX	32
2.1	Técnicas previas a la aplicación del algoritmo simplex.	34
2.2	La tabla extendida.	35
2.3	La tabla condensada.	44
2.4	El método simplex.	46
2.5	El algoritmo simplex.	48
2.6	Análisis gráfico del método simplex.	53
2.7	Interpretación tridimensional del método simplex.	57
2.8	Salución no acotada.	62
2.9	Degeneración.	64

CAPITULO 3

3.0	TEOREMA FUNDAMENTAL DE LA DUALIDAD	79
3.1	Definición del problema dual.	79
3.2	Teorema fundamental de la dualidad.	83
3.3	Teorema dual-simétrico.	84
3.4	Teorema dual-asimétrico.	85
3.5	Interpretación geométrica del problema dual.	88

CAPITULO 4

4.0	REDES	90
4.1	Introducción al análisis de redes.	90
4.2	Definiciones, notación y simbolismo.	94
4.3	Representación matricial de las redes.	100
4.4	Teoría de la conservación de flujo.	105

CAPITULO 5

5.0	REDES DE PROBLEMAS DE RUTA MINIMA	109
-----	-----------------------------------	-----

CAPITULO 6

6.0	EL METODO SIMPLEX APLICADO A LA RESOLUCION DE REDES DE PROBLEMAS DE RUTA MINIMA	117
6.1	Introducción.	117
6.2	Algoritmo primal simplex.	118
6.3	Redes de ruta mínima con actividades redundantes.	121
6.4	Redes de ruta mínima con "varias rutas mínimas".	126
6.5	La presencia del ciclo en las redes de ruta mínima.	129
6.6	Las "actividades ficticias" en las redes de ruta mínima.	132
6.7	Problema ejemplo.	136

CAPITULO 7

7.0	HISTORIA SOBRE LOS CODIGOS DE COMPUTACION	140
7.1	Primeros logros hacia la programación lineal.	143
7.2	Formato de datos para programas de computadora.	145

CAPITULO 8

8.0	CONCLUSIONES	147
	BIBLIOGRAFIA	151
	APENDICE	153

CAPITULO 1

1.0 PROGRAMACION LINEAL Y SU RELACION CON LAS REDES DE FLUJOS

1.1 Introducción

La relación que hay entre la programación lineal y la programación de las redes de flujo, se descubrió al principio del estudio de ambas materias.

Es muy natural, quizás, esperar que la investigación en la teoría y métodos de la programación lineal sean enfocados hacia problemas de ingeniería, ciencias naturales y campos afines, en donde la matemática ha jugado un papel muy importante durante muchos años, no tanto como un lenguaje común sino como un instrumento para la investigación y la síntesis.

Un hecho fundamental y que será demostrado en esta tesis, es que la mayoría de problemas de redes de flujo, pueden formularse como modelos de programación lineal.

Una interrogante muy natural a este hecho es el de ¿por qué deseamos proceder a resolver las redes con una formulación de programación lineal?

Primero: Los modelos de redes de flujo de problemas del mundo real, poseen una estructura muy especial que, a menudo, puede explotarse para obtener soluciones eficientes a través de métodos, en vez de buscar una solución a través de la red misma.

Segundo: Estos "métodos", comunmente llamados algoritmos, nos permiten resolver programas lineales extremadamente grandes, en una fracción del tiempo normal de cálculo.

Tercero: El simple hecho de que podamos enunciar formulaciones matemáticas a través de las representaciones, nodos-arcos, nos ayudará no sólo a la presentación sino a la solución de problemas del mundo real.

Debido a que las representaciones de las estructuras especiales de las redes del flujo, como problemas de programación lineal, juegan un papel central para el desarrollo de un algoritmo eficiente, es un ahorro de tiempo explorar estas estructuras. En particular, es muy ventajoso predecir cuándo la maquinaria de la programación lineal, producirá una solución óptima que es toda entera.

CLASIFICACION DE LOS PROBLEMAS DE PROGRAMACION LINEAL

Discretos o continuos
Etapas múltiples o etapas no múltiples
Dinámicos o no dinámicos

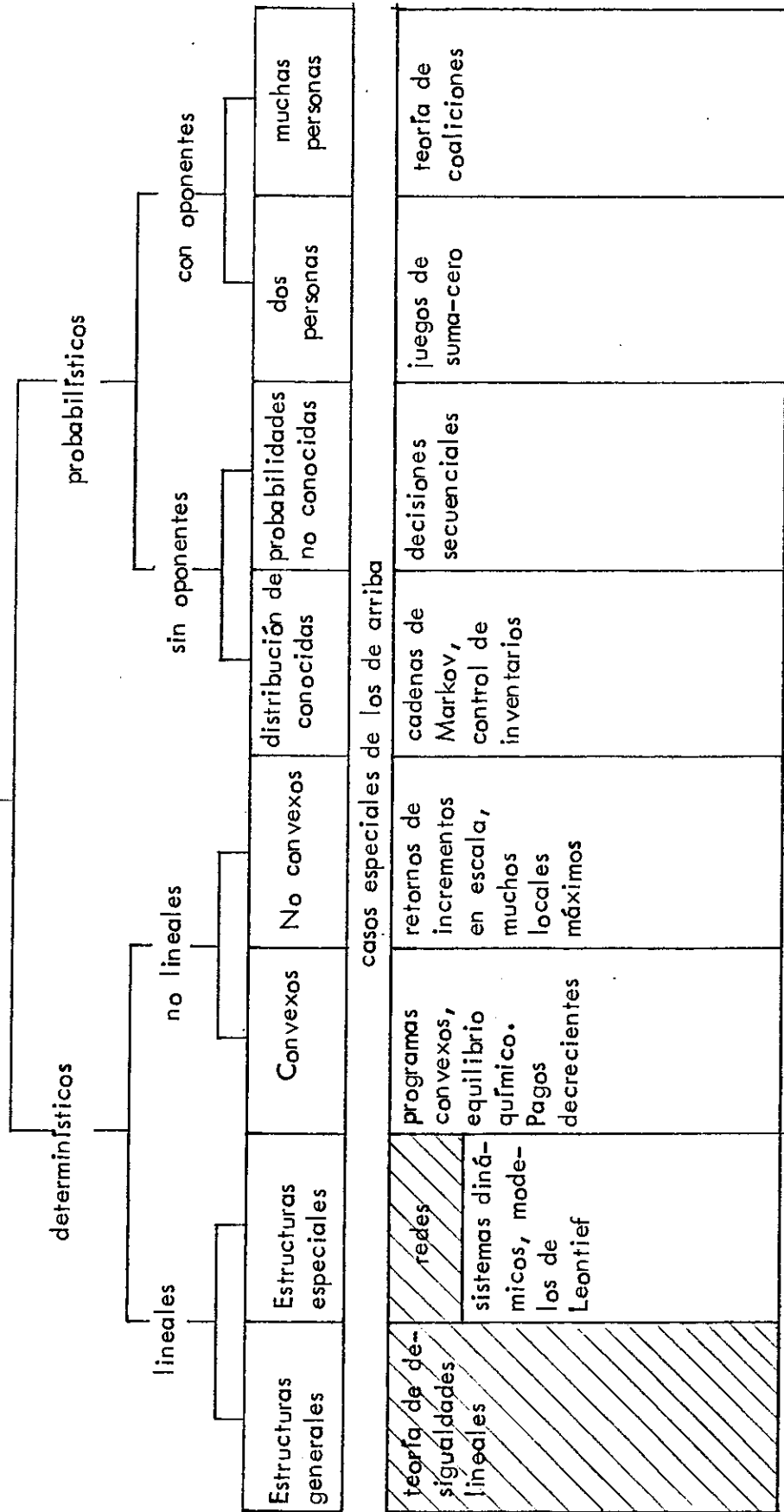


Figura 1.1

RELACION ENTRE LA PROGRAMACION LINEAL Y LOS PROBLEMAS DE PROGRAMACION DE REDES DE FLUJO

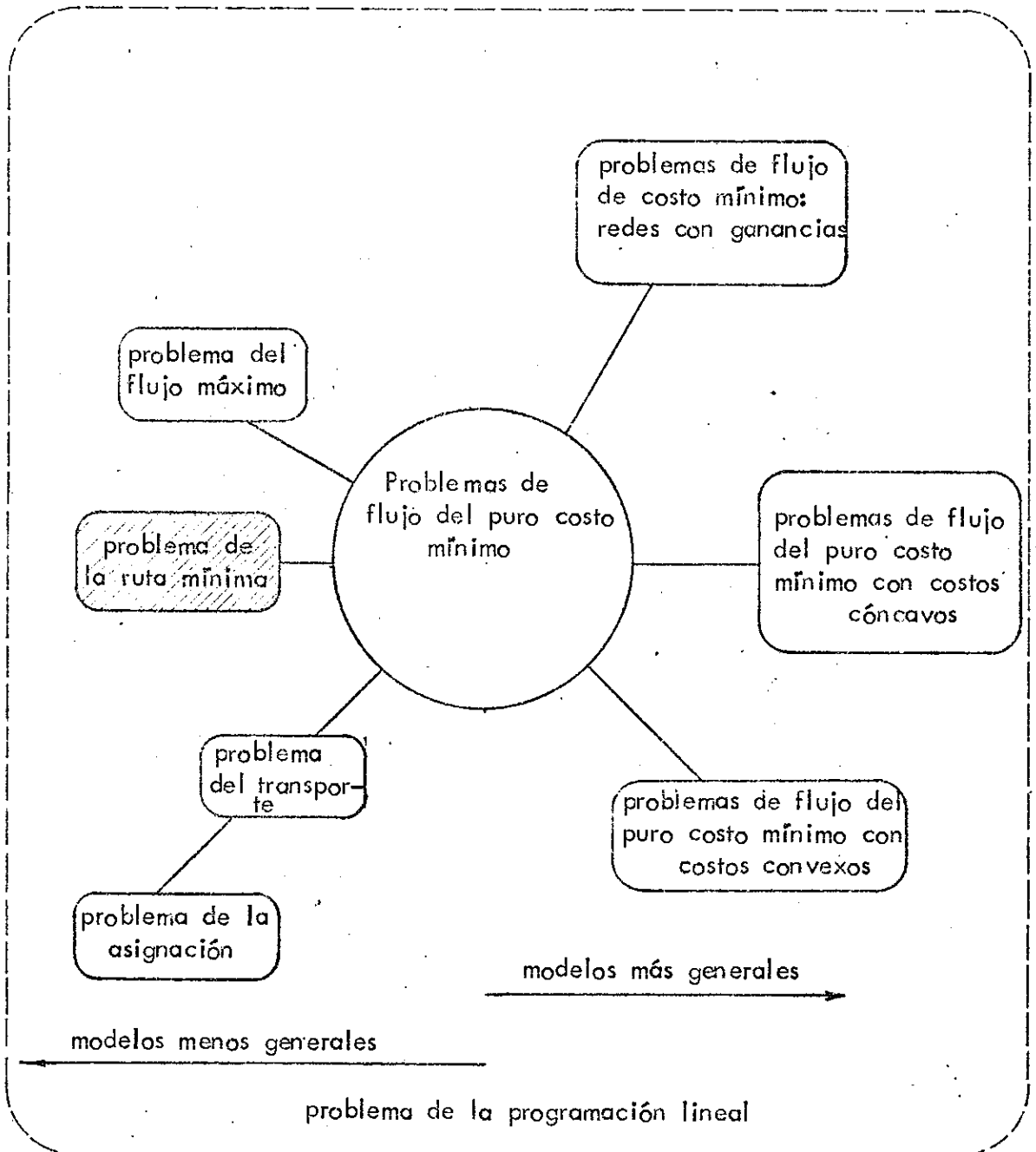


Figura 1.2

En la figura 1.2 pueden observarse los problemas de programación de redes que pueden formularse como problemas de optimización a través de la programación lineal. Los problemas ubicados a la izquierda son menos generales en el sentido de que son especializaciones del problema básico (al centro) y los ubicados a la derecha, son más generales en el sentido de que el problema básico (al centro) es, en alguna forma, una especialización para cada uno de estos problemas.

Los algoritmos desarrollados para los problemas menos generales son más eficientes tanto en tiempo de computadora como en necesidad de memoria, en comparación con los de los problemas más generales. Estos, a su vez, pueden resolver problemas de forma menos generales, mientras lo contrario es raramente verdadero. El problema que nos ocupará, en esta tesis, es el de aplicar la maquinaria de la programación lineal al problema de la ruta mínima, tan identificado con proyectos de construcción civil.

1.2 Programación lineal

No es necesario justificar el uso de la técnica conocida como programación lineal, ya que anualmente se gastan millones de dólares por su aplicación a la optimización de problemas de ingeniería industrial, administración de proyectos, problemas de transporte, licitaciones, etc. Las técnicas de la programación lineal son una forma para que las grandes empresas funcionen más eficiente y rápidamente, a un mínimo costo.

Se le puede considerar como parte de una disciplina más general, conocida como programación matemática, aplicada a problemas de planeamiento, logística y ubicación, que incluyen no-linealidades.

Sin embargo, los métodos básicos continúan resolviéndose con álgebra de desigualdades lineales.

Desde que la programación matemática tiene otras connotaciones en computación, preferimos denominarla programación lineal o PL, el cual se entiende que incluye muchas especializaciones y extensiones.

De paso, el término programación tiene estrecha relación con la computación y el campo del procesamiento de datos.

La PL tiene estrecha asociación con las computadoras, habiendo crecido con ellas, a causa de su enorme necesidad para la manipulación

aritmética y de datos. Es importante notar que aunque la PL incluye una aplicación de computadoras, el término en sí mismo no se refiere a ningún modo de programar las computadoras PER SE.

La aplicación práctica de la PL a problemas industriales y otros, incluye el uso de sistemas de programas de computadoras preparados con gran esfuerzo, los que mencionaremos de vez en cuando.

Para evitar confusión entre las diferentes acepciones con que se utiliza el término programador, diremos que: "Un analista es el que utiliza un sistema de programas de computadora para la PL, como un instrumento".

"El analista de sistemas o programador de sistemas, es el que construye el sistema de programas de computadora".

De 1953 a la fecha, se han publicado innumerables artículos y capítulos sobre PL; sin embargo, el lector puede muy bien preguntar ¿qué justificación hay aún para otro enfoque?

La respuesta es que la PL tiene muchas facetas, algunas de las cuales nunca han sido adecuadamente tratadas; lo cual merece una respuesta aún más ampliada.

La PL ha crecido a la altura de una disciplina aplicada que muy bien puede desglosarse en siete partes:

1.2.1 Experiencia matemática: Por la teoría de desigualdades lineales y conjuntos convexos.

1.2.2 Aplicación teórica: Por el modelaje de situaciones del mundo real y la formulación de problemas.

1.2.3 Fundamentos algebraicos y teóricos: Por la solución de algoritmos y procedimientos analíticos.

1.2.4 Construcción de sistemas de programas de computadora que manejan la PL y problemas afines.

1.2.5 Recolección de datos, conversión y transcripción.

1.2.6 Administración de las facilidades que ofrece la computación, como: Procedimientos, biblioteca de datos, operaciones periféricas variadas.

1.2.7 Interpretación y presentación de resultados: Base para la toma de decisiones a nivel gerencial.

El anterior listado refleja el amplio rango de especialistas que están involucrados en alguna fase de la programación lineal.

Así: La parte 1.2.1 es del dominio de los matemáticos

La parte 1.2.2 es del dominio de ingenieros y economistas

La parte 1.2.6 es del dominio de técnicos y mediana gerencia

La parte 1.2.7 es el punto de vista de mediana y alta gerencia

1.3 Notación

Es indudable que la notación utilizada al discutir el álgebra de matrices, desigualdades lineales y la programación lineal, a menudo es un obstáculo para el lector. Desde un punto de vista positivo, una buena notación es el mejor vehículo por medio del cual las ideas pueden desarrollarse y comunicarse.

Usaremos tres tipos de notaciones:

1.3.1 Notación indicial

1.3.2 Notación matricial

1.3.3 Diagramas

Trataremos con ecuaciones lineales simultáneas de muchas incógnitas.

Así, sea $a = (a_1, a_2, \dots, a_n)$ una n -tupla de coeficientes considerados como un vector fila y $X = (X_1, X_2, \dots, X_n)$ una n -tupla de variables consideradas como un vector columna y "b" una constante. Tal ecuación puede escribirse así:

$$a_1 X_1 + a_2 X_2 + \dots + a_n X_n = b$$

o más brevemente:

$$\sum_{j=1}^n a_j X_j = b$$

o como un producto punto:

$$a \cdot X = b \underline{\hspace{2cm}}$$

Si consideramos sólo cierto número de ecuaciones, digamos "m", usaremos la siguiente notación para distinguirlas:

$$\sum_{i=1}^n a_{ij} X_i = b_j \quad \text{con } i = 1, 2, \dots, m$$

podemos considerar los coeficientes a_{ij} como "n" columnas de "m" elementos a_{ij} : $a_{1j}, a_{2j}, \dots, a_{mj}$, y las constantes como otra columna b_j : b_1, b_2, \dots, b_m . Esta simbología se utiliza tanto para las ecuaciones como para los lenguajes de computadora o códigos.

Si ahora consideramos a los a_{ij} como una matriz A de dimensión $m \times n$, los representaremos así:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & a_{ij} \\ \cdot & \cdot & \dots & \cdot \\ a_m & a_{m2} & \dots & a_{mn} \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \\ \cdot \\ \cdot \\ \cdot \\ X_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_m \end{bmatrix}$$

que representa a un conjunto de "m" ecuaciones y "n" incógnitas. En notación matricial los representamos así:

$$AX = b \underline{\hspace{2cm}}$$

Otro tipo de restricciones que hallaremos serán:

$$\sum_{j=1}^n a_{ij} X_j = b_i \quad \text{para algún "i"}$$

$$\sum_{j=1}^n a_{ij} X_j \leq b_i \quad \text{para otro "i"}$$

$$\sum_{j=1}^n a_{ij} X_j \geq b_i \quad \text{para un otro "i"}$$

$$\sum_{j=1}^n a_{ij} X_j \geq b_i \quad \text{para un restante "i"}$$

Donde la forma actual de c_i se especifica en una lista. Una notación general para indicar cualquiera de los de arriba será:

$$\sum_{j=1}^n a_{ij} X_j \leq b_i \quad \text{con } i = 1, 2, \dots, m.$$

1.4 Enunciado clásico de la programación lineal

A través de la presente exposición, trataremos con problemas lineales llamados programas lineales. De forma que, como la palabra lineal lo sugiere, todas las variables de un programa lineal son de primer grado.

Un programa lineal se compone de una función lineal de "n" variables, que debe ser maximizada o minimizada, sujeta a un conjunto de "m" restricciones. Una restricción es una igualdad o desigualdad débil*, que representa un límite para las variables de un problema.

El problema central de la programación lineal, generalmente se enuncia así:

" Dados "m" objetos de coeficientes C_i y con coeficientes tecnológicos a_{ij} de dimensión $m \times n$; con "n" constantes " b_i " = ($i = 1, \dots, m$; $j = 1, 2, \dots, n$). Hallar valores para los X_j (si existen), de forma que:

$$\sum_{i=1}^m C_i X_i = \text{mínimo} \quad 1.4.1$$

sujeto a:

$$\sum_{i=1}^m a_{ij} X_i \leq b_i \quad \text{con } i = 1, 2, \dots, m \quad 1.4.2$$

$$\text{y } X_i \geq 0 \quad 1.4.3$$

* Una "desigualdad débil", puede ser satisfecha, también, por una igualdad.

De tal forma que, cualquier conjunto de X_i que satisfaga las ecuaciones 1.4.1 y 1.4.2, se llamará el Conjunto Solución o Solución Factible. Y una solución factible que satisfaga la ecuación 1.4.3, se llamará Solución Optima Factible.

Geoméricamente las X_i son las coordenadas de un punto (vector) en el espacio "m-dimensional". Y las C_i no son más que los coeficientes de la función objetivo que deseamos optimizar.

1.5 Definiciones

1.5.1 "Las coordenadas de un punto que satisfacen todas las "m" restricciones lineales en 1.4.2, constituyen un conjunto solución".

1.5.2 "Si las coordenadas de un "punto" solución, son todas positivas, la solución se llama Solución Factible".

1.5.3 "Si una o más de las coordenadas del conjunto solución, son negativas, la solución se llama Solución No Factible".

1.5.4 "Al conjunto de todas las coordenadas factibles se le conoce como la región factible del problema".

1.5.5 "Si entre las soluciones factibles hay una que minimiza el valor de la función 1.4.1, entonces la denominaremos Solución Factible Optima".

1.5.6 "A la función lineal de 1.4.1, se le denomina Función Objetivo y a su valor en un punto de la región factible, se le llama valor primal.

1.6 Convexidad

1.6.1 Conjuntos convexos

Definición 1 "Una región o conjunto "R" es convexa, si y sólo si, para c/dos puntos de R, el segmento de línea que une esos dos puntos, descansa completamente dentro de R".

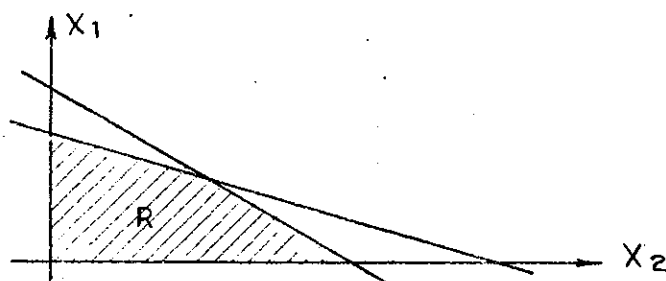


Figura 1.3

Si observamos la figura anterior, vemos que cada vértice es un "punto factible" y en conjunto producen lo que llamaremos Soluciones Básicas Factibles, ya que ellos forman una base para todas las posibles soluciones factibles. De modo que cualquier combinación lineal convexa de esos puntos, puede expresar acualquier punto factible, así:

$$P = \alpha_1 P_1 + \alpha_2 P_2 + \dots + \alpha_n P_n$$

Donde los escalares $\alpha_i \geq 0$ y $\sum \alpha_i = 1$

Ejemplos de conjuntos convexos son: El espacio euclidiano E_n , un círculo, un cubo, etc. No así, el conjunto de puntos que forman la periferia de un círculo.

Ahora demostraremos que para un conjunto convexo R , "cualquier combinación lineal convexa de cualquier conjunto de puntos en R está también, en R ".

$$\text{Sea } P = \alpha_1 P_1 + \alpha_2 P_2 + \alpha_3 P_3$$

$$\text{Para } \alpha_i \geq 0, \quad \sum_{i=1}^3 \alpha_i = 1 \quad \stackrel{?}{\implies} \in R$$

Hipótesis:

$$\text{Sea } \alpha'_i = \frac{\alpha_i}{\alpha_1 + \alpha_2} \quad \text{Para } i = 1, 2$$

Vemos que $\alpha'_i = 1$ y $\alpha'_i \geq 0$, luego:

$$\begin{aligned} P &= \alpha_1 P_1 + \alpha_2 P_2 + \alpha_3 P_3 = (\alpha_1 + \alpha_2) \alpha'_1 P_1 + (\alpha_1 + \alpha_2) \alpha'_2 P_2 + \alpha_3 P_3 \\ &= (\alpha_1 + \alpha_2) (\alpha'_1 P_1 + \alpha'_2 P_2) + \alpha_3 P_3 \end{aligned}$$

Por hipótesis sabemos que $\alpha'_1 P_1 + \alpha'_2 P_2$ está en R , luego:

$P_4 = (\alpha'_1 P_1 + \alpha'_2 P_2)$ Finalmente: $P = (\alpha_1 + \alpha_2) P_4 + \alpha_3 P_3 \in R$, el cual expresamos como una combinación lineal convexa de dos puntos en R

Ejemplo: El punto $P = (2, 1)$, podemos escribirlo así:

$$P_0 = (0, 0) \quad P_1 = (5, 0) \quad P_2 = (0, 4) \quad P_3 = (2, 3)$$

$$P = \alpha_0 P_0 + \alpha_1 P_1 + \alpha_2 P_2 + \alpha_3 P_3$$

$$(2, 1) = 0.4 P_0 + 0.2667 P_1 + 0.333 P_2 = 0.4 (0, 0) + 0.2667 (5, 0) +$$

$$(2, 1) = (2, 1) + 0.333 (2, 3) = (2, 1)$$

Los P_i son un espacio solución, que para el presente ejemplo es un espacio "cuatridimensional".

1.6.1.1

Teorema: "Cualquier punto sobre el segmento de línea que une dos puntos en el espacio euclidiano E_n , puede expresarse como una combinación lineal convexo de los dos puntos".

Demostración: Sean los puntos U y V , diremos que el punto W descansa sobre el segmento de línea que une a U con V , que es lo que necesitamos demostrar; de la gráfica, podemos ver que el segmento de línea sobre el que hallamos a W , está definida por la dirección del vector $U-V$.

Por las reglas de adición de vectores, sabemos que para cualquier:

$$0 \leq \lambda \leq 1$$

$$V + \lambda(U-V) = W \Rightarrow (1-\lambda)V + \lambda U = W$$

Que es la expresión del punto W como una combinación lineal convexa de los puntos U y V .

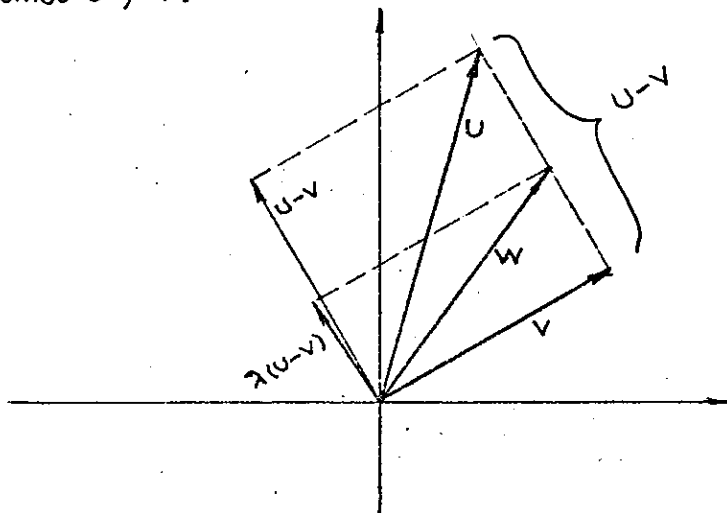


Figura 1.4

Corolario: "Cualquier punto que pueda expresarse como una combinación lineal convexa de una pareja de puntos E_n descansa exactamente sobre el segmento de línea que une a esa pareja de bipuntos".

Demostración:

$$\text{Sobemos que } W = (1-\lambda) V + \lambda U$$

$$W - V = \lambda(U - V) \quad \text{Donde } 0 \leq \lambda \leq 1$$

Podemos observar que el vector $W-V$ es algún múltiplo positivo del vector $U-V$, ya que el vector $W-V$ coincide en la misma dirección del vector $U-V$. En otras palabras, el segmento de línea que une a U con V y el segmento de línea que une a W con V , son paralelas entre sí. Por lo tanto, el punto W debe descansar en el segmento de línea que une a U con V .

Del teorema anterior y su corolario, deducimos que "geoméricamente un conjunto convexo es aquél que contiene todos los segmentos de línea que unen cualquier pareja de puntos del conjunto".

De esta forma, cualquiera de los puntos W es llamado una combinación lineal convexa de los puntos U y V . El segmento de línea UV , no es más que todo el conjunto de puntos W , cuyas coordenadas satisfacen las ecuaciones paramétricas:

$$W_1 = (1-\lambda) V_1 + \lambda U_1$$

$$W_2 = (1-\lambda) V_2 + \lambda U_2 \quad \text{Para } 0 \leq \lambda \leq 1$$

En una forma más resumida: $w_i = (1 - \lambda) V_i + \lambda U_i \quad i = 1, \dots, m$

"Al conjunto de todas las combinaciones lineales convexas de U y V , para $0 \leq \lambda \leq 1$, es precisamente el segmento de línea UV , que es convexo".

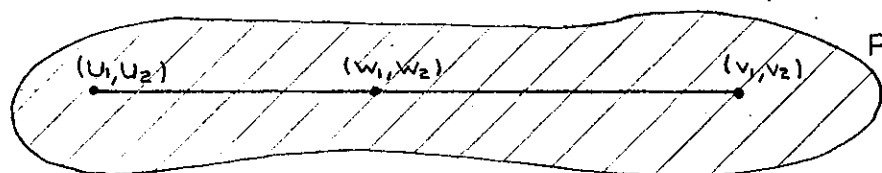


Figura 1.5

Sea R una región convexa del plano E . La definición de convexidad afirma que el "conjunto de puntos que satisfacen las ecuaciones paramétricas pertenecen a R , siempre y cuando (U_1, U_2) , pertenezcan también a R . Obsérvese que los valores 0 y 1, del parámetro λ , produce los puntos extremos U y V , respectivamente. Ya que para $\lambda = 1$, el punto W va a estar colocado en U y para $\lambda = 0$, el punto W estará en V .

El segmento de línea definido por las ecuaciones paramétricas en se le llama un segmento cerrado, ya que incluye los bipuntos extremos U y V .

Un segmento abierto omitiría los puntos extremos. Si λ es un real no restringido, las ecuaciones en representarían a la línea entera definida por los bipuntos U y V .

1.6.1.2

Teorema: "El segmento de plano cerrado $aX_1 + bX_2 \leq C$ es una región convexa".

Demostración: Sea $U (U_1, U_2)$ y $V (v_1, v_2)$, pertenecientes al segmento de plano cerrado $aX_1 + bX_2 \leq C$, cuyas coordenadas satisfacen:

$$a \cdot U_1 + b U_2 \leq C \quad f(u_1, u_2) = M$$

$$a \cdot V_1 + b V_2 \leq C \quad f(V_1, V_2) = V$$

Y sea $W (w_1, w_2)$, cualquier punto sobre el segmento de línea UV. De las ecuaciones paramétricas, tenemos:

$$\begin{aligned} a \cdot w_1 + b \cdot w_2 &= a [(1-\lambda) V_1 + \lambda U_1] + b [(1-\lambda) V_2 + \lambda U_2] \\ &= (1-\lambda) (a V_1 + b V_2) + \lambda (a \cdot U_1 + b \cdot U_2) \\ &\leq (1-\lambda) C + \lambda C \end{aligned}$$

pero como $1-\lambda$ y λ , son positivos, nos queda finalmente:

$$a \cdot w_1 + b \cdot w_2 \leq C$$

Esto demuestra que todos los puntos que están sobre el segmento de línea UV, están en $aX_1 + bX_2 \leq C$. Y, por lo tanto, el segmento de plano cerrado es convexo.

La combinación del teorema anterior y el ante-anterior, da el siguiente teorema:

1.6.1.3

Teorema: "La intersección de cualquier número de segmentos de planos, es una región convexa".

Definición 2: "Una región poligonal convexa, es la intersección de un número positivo finito de segmentos de líneas, cerrados".

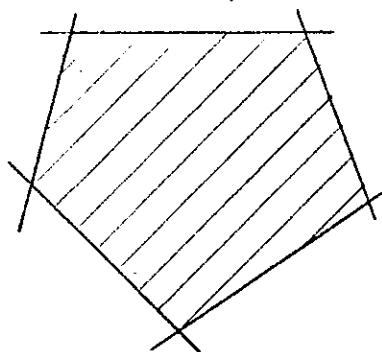


Figura 1.6

La región factible de un conjunto finito de desigualdades lineales, con dos variables, es la intersección de un finito número de segmentos de planos cerrados y es una región poligonal convexa. Las regiones factibles de la interpretación gráfica, son regiones poligonales convexas.

Definición 3: "Un polígono convexo, es una región poligonal convexa R , tal que, cualquier línea que atraviese un punto de R , intersecta a R en un segmento cerrado."

Ya que un segmento cerrado tiene longitud finita, un polígono convexo es una región limitada del plano. Los límites de una región poligonal convexa están formados de parte o todos los segmentos de líneas de los planos intersectantes. Estos límites pueden ser líneas, rayos o segmentos cerrados. Así, en la figura (a), los límites de R son líneas paralelas. En la figura (b) los bordes A y B son rayos o mitades de líneas. En la figura (c) todos los bordes son segmentos de líneas. Nótese que las figuras (a) y (b) representan regiones poligonales convexas que son ilimitadas o de extensión infinita y la figura (c) ilustra un polígono convexo que es una región finita del plano.

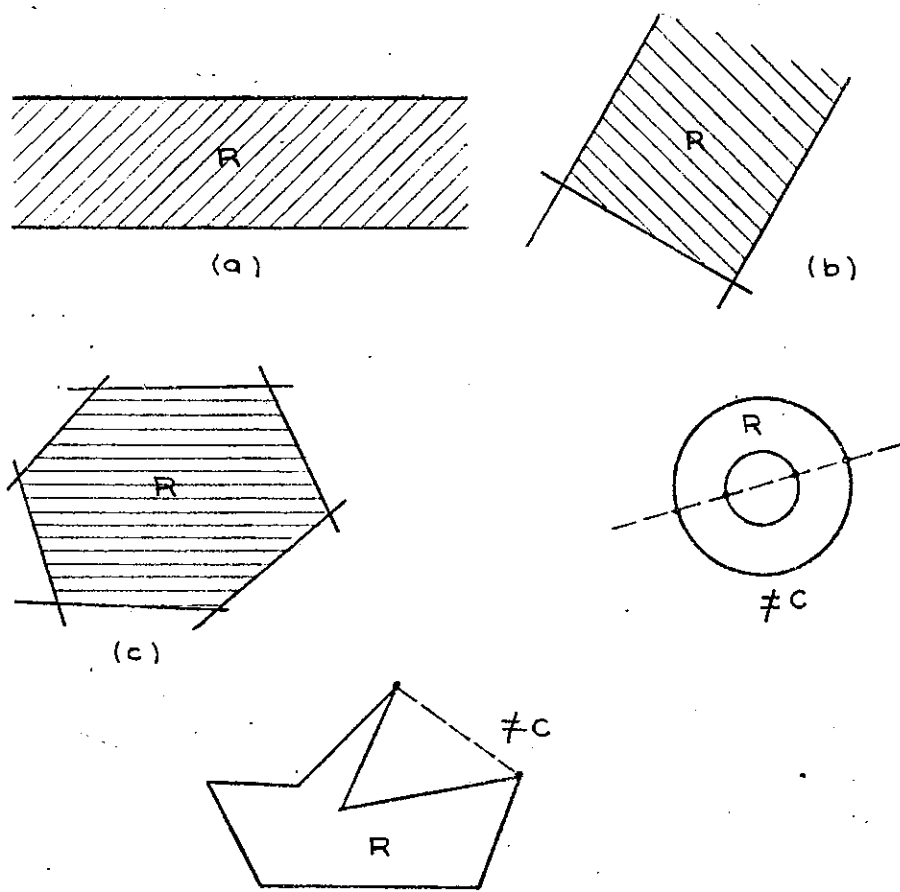


Figura 1.7

Al análogo de un polígono convexo en tres dimensiones, se le llama un polihedro convexo y cada una de sus caras forma un polígono convexo. Todos los polihedros regulares tales como el tetrahedro y el cubo, son convexos. Una característica de estas figuras, es que tienen esquinas llamadas vértices o puntos extremos.

Definición 4: "Un vértice o punto extremo de un conjunto convexo R , es un punto de R que no puede ser expresado como una combinación lineal convexa de otros dos puntos distintos de R ".

Bidimensionalmente, un vértice de una región poligonal convexa, es un punto en la región donde se unen dos líneas limítrofes.

Cada punto del perímetro de una circunferencia es un punto extremo del conjunto convexo, que incluye el perímetro y el círculo. Los puntos extremos de un triángulo son sus vértices. Una región polihédrica muy interesante es la esfera, ya que cada punto de su superficie es un punto extremo. En un polihedro convexo, los vértices estarán donde se unen las orillas.

Según la definición, los vértices se mantienen por sí solos. Ellos no pueden ser alcanzados por ningún segmento de línea que conecte dos puntos de R , que son diferentes del mismo vértice.

Un vértice es un punto de la región, exactamente sobre dos líneas limítrofes.

En dos dimensiones, más de dos líneas limítrofes pueden pasar a través de un vértice, pero una de ellas será la redundante al momento de definir la región convexa.

Un punto "U" de un conjunto convexo "C", se llama un punto extremo, si "U" no puede expresarse como una combinación lineal convexa de cualesquiera otros dos puntos distintos en "C". Cada punto situado sobre el límite de un círculo es un punto extremo del conjunto convexo, que incluye el límite y el interior del círculo.

Si el conjunto convexo lo incluyera el límite, este conjunto NO contendrá puntos extremos.

1.6.2

Ejemplo de conjuntos convexos:

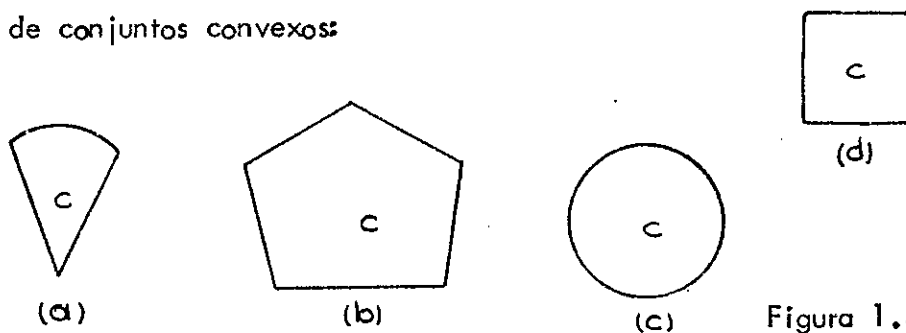


Figura 1.8

El cascarón convexo $C(S)$ de cualquier conjunto, dado de puntos S , es el conjunto de todas las combinaciones lineales convexas, de conjuntos de puntos desde " S ". $C(S)$ es el más pequeño conjunto convexo que contiene a " S ".

Si " S " son los ocho vértices de un cubo, luego $C(S)$ es el cubo completo.

Si " S " es el perímetro de una circunferencia, luego $C(S)$ es el círculo completo.

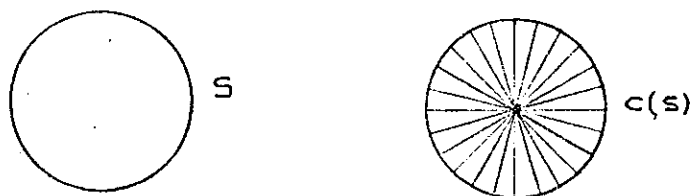


Figura 1.9

Si el conjunto " S " consiste de un número finito de puntos, el cascarón convexo de " S " es un polihedro convexo. Así, el $C(S)$ de los ocho vértices de un cubo es un polihedro convexo.

A un conjunto de vectores "R" se le llama un cono, si para cada vector "U" en "S", λU está en "R". Donde λ es un escalar positivo.

Ejemplos de un cono son: El espacio completo, el origen y el conjunto "R" de la siguiente figura. Obsérvese que este cono contiene el origen, ya que $\lambda = 0$. Y, además, no es convexo. Ya que un cono convexo es un cono que es convexo.

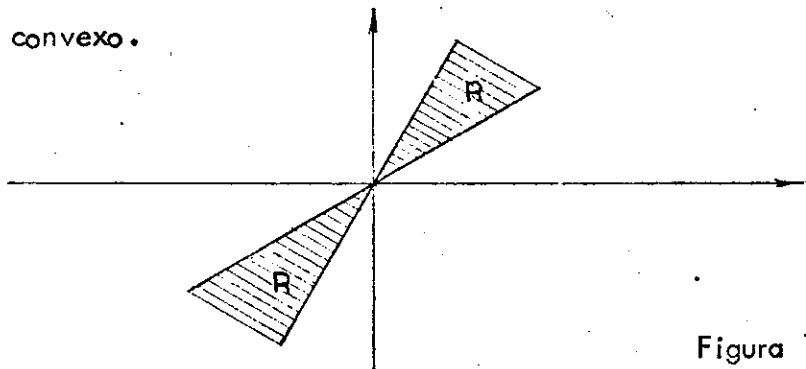


Figura 1.10

Un simplex es un polihedro convexo, n -dimensional, que tiene exactamente $n + 1$ vértices. Cada límite del simplex contiene simples de más baja dimensión, que se llaman facetas simpliciales. El número de tales facetas o caras, de dimensión m es $\binom{n}{m} = \frac{n!}{m!(n-m)!} = \binom{n+1}{m+1}$

En la dimensión cero, un simplex es un punto. En una dimensión, es una línea. En dos dimensiones es un triángulo (ver figura anterior) y en tres dimensiones es un tetrahedro.

La ecuación de un simplex con intercepciones unitarias es:

$$x_i \geq 0 \quad \sum_{i=1}^m x_i = 1$$

Para $i = 3$ tenemos el tetrahedro con vértices

$$(0, 0, 0), (0, 1, 0), (0, 0, 1), (1, 0, 0)$$

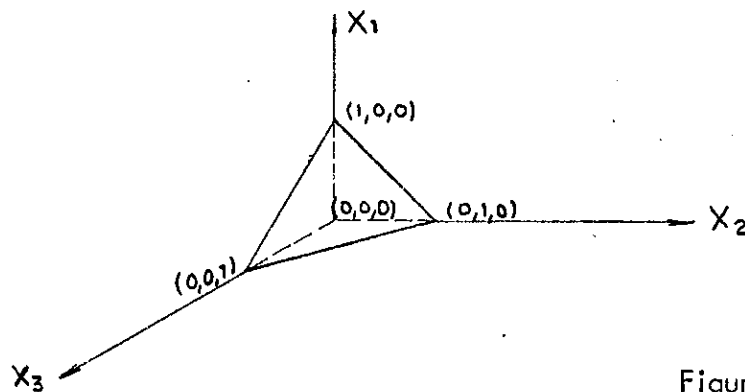


Figura 1.11

1.7 Funciones lineales

En el plano de coordenadas (X_1, X_2) examinaremos funciones lineales de la forma $f(X_1, X_2) = aX_1 + bX_2$, donde a, b son constantes reales distintas de cero. Si $U(U_1, U_2)$ y $V(V_1, V_2)$, son bipuntos distintos en el plano, las ecuaciones paramétricas de la línea que va de U a V son:

$$X_1 = (1-\lambda) V_1 + \lambda U_1$$

$$X_2 = (1-\lambda) V_2 + \lambda U_2 \quad \text{Para todos los números reales } \lambda \\ \forall \lambda \in \mathbb{R}.$$

Sea $f(U_1, U_2) = M$ y $f(V_1, V_2) = m$

Para cualquier punto $X(X_1, X_2)$ sobre la línea \overline{UV} tenemos:

$$\begin{aligned} f(X_1, X_2) &= a \cdot [(1-\lambda) V_1 + \lambda U_1] + [(1-\lambda) V_2 + \lambda U_2] \cdot b \\ &= (1-\lambda) (a V_1 + b V_2) + \lambda (a U_1 + b U_2) \\ &= (1-\lambda) f(V_1, V_2) + \lambda f(U_1, U_2) \\ &= (1-\lambda) m + \lambda M \end{aligned}$$

$$f(X_1, X_2) = m + \lambda(M-m) \quad 1.7.1$$

Con este resultado, pasamos a demostrar el siguiente:

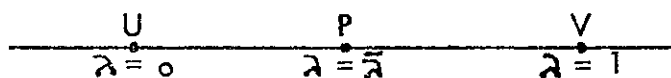
1.7.2

Teorema: "Si f es lineal y tiene el mismo valor en dos puntos distintos U y V , luego f permanece constante a lo largo de la línea que une \overline{UV} . De lo contrario, si f tiene diferentes valores en U y V , en cada punto del segmento abierto \overline{UV} , f tendrá estrictamente un valor entre los valores de U y V ".

Demostración: De la ecuación 1.7.1, sea $m=M$ y luego $f(X_1, X_2) = m$ es constante. Si $m \neq M$, digamos que $m < M$, entonces $M-m > 0$, La ecuación (3) puede considerarse una función de parámetro λ y la llamaremos $f(\lambda)$.

$$f(\lambda) = m + \lambda(M - m)$$

$$f(0) = m \quad \text{y} \quad f(1) = M$$



Para cualesquiera dos valores del parámetro $\lambda_1 < \lambda_2$:

$$(M - m)\lambda_1 < (M - m)\lambda_2$$

$$m + (M - m)\lambda_1 < m + (M - m)\lambda_2$$

$$f(\lambda_1) < f(\lambda_2)$$

Esto significa que f es una función estrictamente creciente de parámetro λ . De modo que cuando λ incrementa los valores funcionales, $f(\lambda)$ incrementa también de continuo.

Ahora, sea $P(X_1, X_2)$, con un parámetro de valor $\bar{\lambda}$, un punto cualquiera del segmento abierto \overline{UV} de la figura 2.3). En las ecuaciones paramétricas del segmento \overline{UV} , el parámetro λ también incrementa estrictamente tanto como $0 < \bar{\lambda} < 1$. Luego,

$$f(0) < f(\bar{\lambda}) < f(1) \quad \text{o} \quad m < f(X_1, X_2) < M$$

De lo cual podemos concluir del teorema "que una función lineal definida sobre un segmento cerrado, asumirá su máximo y mínimo, en los puntos extremos del segmento".

1.8 Teorema del punto extremo

Nuestro interés se centra especialmente en los vértices o puntos extremos de una región poligonal convexa. Lo fundamental de este desarrollo, es la idea observada en el análisis gráfico, en el que la función objetivo obtiene su valor óptimo sobre un vértice de una región convexa.

1.8.1 "Los valores máximos y mínimos de una función lineal f , definida sobre un polígono convexo R , existen y se hallan en los vértices de R ".

Demostración: Supongamos que $P(\bar{X}, \bar{Y})$ es un punto interior de R no en el límite de R . Sea L un segmento de recta arbitrario que pasa por el punto P , el cual por definición 3, intersecta a R en un segmento cerrado. Sean $P_1(X_1, Y_1)$ y $P_2(X_2, Y_2)$ los puntos extremos del segmento. Ya que el resto de la línea L es exterior a R , los puntos extremos P_1 y P_2 deben estar sobre el límite de R . Si la restricción lineal que define el límite en P_2 es $cX+dY \leq e$, luego $cX_2 + dY_2 = e$, en tanto $c\bar{X} + d\bar{Y} = K < e$, ya que P es un punto anterior. La función lineal $cX+dY$ asumirá valores entre K y e para puntos sobre L entre P y P_2 , por el teorema 1.7.2. De igual forma, estrictas desigualdades serán satisfechas por todos los puntos entre P y cualquier otra línea limítrofe. De forma que todos los puntos entre un punto interior y un punto limítrofe, son puntos interiores. Esto significa que el segmento abierto entre P_1 y P_2 , consiste de puntos interiores y L puede intersectar el límite en sólo dos puntos. Sea la función lineal f . Si $f(X_1, Y_1) \leq f(X_2, Y_2)$, luego por 1.7.2.

$$f(X_1, Y_1) \leq f(\bar{X}, \bar{Y}) \leq f(X_2, Y_2) \quad 1.8.1.1$$

Los límites de R son segmentos cerrados, por definición 3. Los puntos extremos de los segmentos limítrofes, descansan exactamente sobre dos líneas limítrofes, que forman vértices. Digamos P_3 y P_4 . Aplicando de nuevo el teorema 1.7.2 $f(X_2, Y_2)$ está limitado sobre ese segmento limítrofe, digamos:

$$f(X_3, Y_3) \leq f(X_2, Y_2) \leq f(X_4, Y_4) \quad 1.8.1.2$$

Combinando las desigualdades 1.8.1.1 y 1.8.1.2, nos queda

$$f(\bar{X}, \bar{Y}) \leq f(X_4, Y_4) \quad 1.8.1.3$$

Utilizando el mismo argumento, $f(X_1, Y_1)$ está limitado por los valores de f en dos vértices, es decir; P_6 y P_5 .

$$f(X_5, Y_5) \leq f(\bar{X}_1, \bar{Y}_1) \leq f(X_6, Y_6) \quad 1.8.1.4$$

Combinando las desigualdades 1.8.1.1, 1.8.1.3 y 1.8.1.4, obtenemos

$$f(X_5, Y_5) \leq f(\bar{X}, \bar{Y}) \leq f(X_4, Y_4) \quad 1.8.1.5$$

De esta forma hemos demostrado que f , para cualquier punto interior P , está limitado por los valores de f sobre los límites y éstos, a su vez, están limitados por los valores de f en los vértices.

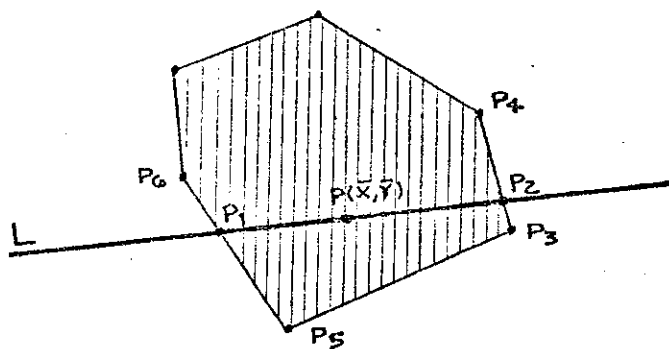


Figura 1.12

Debido a que hay sólo un número finito de vértices en un polígono convexo, podemos examinar f en cada vértice y escoger un valor de f tan grande como sea posible y otro con un valor tan pequeño como sea posible.

Esta selección de vértices puede no ser única, pero ello representa los puntos máximos y mínimos de f en R .

En el caso de una región poligonal convexa que puede ser ilimitada, una función lineal sobre esa región puede aumentar o disminuir sin límite. La intersección de una línea a través de un punto de R con R , puede ser un rayo o una línea entera. A lo largo de un rayo, al menos una coordenada debe continuar aumentando o disminuyendo, de forma que cualquier f lineal que involucra a esa coordenada, puede aumentar o disminuir sin el límite. Para enunciar un teorema, en este caso, es necesario hipotetizar la existencia de un óptimo. Enunciaremos el teorema sin demostrarlo, ya que el razonamiento sigue al mismo argumento dado en teorema 1.8.1.

1.8.2

Teorema: "Si el valor máximo o mínimo de una función lineal, definida sobre una región poligonal convexa, existe, entonces se halla sobre el límite o vértice de la región".

Si un límite existiera sobre una línea entera, luego no habría ningún vértice sobre ese límite. Sin embargo, para una región factible situada en el primer cuadrante, esto no sucedería, ya que cada línea limítrofe contiene al menos un vértice. Con la condición de que un vértice se produce en el límite óptimo, el valor óptimo ocurre en un vértice. Así, el resultado es el mismo como en el caso de un polígono convexo, con tal que el óptimo exista. Las pruebas en este capítulo se han llevado a cabo en dos dimensiones, por

simplicidad y facilidad de comprensión. Las ideas generalizadas a n dimensiones, se dejan al problema general de la programación lineal.

CAPITULO 2

2.0 HISTORIA DEL METODO SIMPLEX

El método simplex fue desarrollado por George Dantzig durante el año 1947, siendo su primera aplicación al problema de la nutrición o dieta. Y posteriormente se le aplicó al problema del transporte.

Entre las mejores presentaciones de la teoría de Dantzig, además de la suya, se encuentran las de Saul Gass, Charnes y Cooper, y la de George Hadley. Posteriormente surgieron variantes del algoritmo simplex, como el método dual simplex, el algoritmo primal-dual, el método simplex revisado, el algoritmo paramétrico auto-dual y el de las bases dual y primal complementarias.

La palabra simplex se deriva del concepto de la figura convexa posible, más simple, que tiene exactamente un vértice más que la dimensión del espacio. Así, un simplex en el espacio de dimensión cero es un punto. En el espacio de una dimensión, es un segmento de línea con dos puntos extremos. En el espacio bidimensional, es un triángulo; y en el de tres dimensiones es un tetrahedro o pirámide de cuatro vértices.

Cada una de estas figuras es el conjunto convexo más pequeño que contiene sus vértices y se le denomina el cascarón convexo de los vértices. Siempre que el número de vértices o puntos extremos sea finito, al cascarón

convexo formado por esos puntos se le llama poliedro convexo. Los simples son el caso especial de los poliedros convexos que tienen un vértice más que el espacio dimensión y con la propiedad particular de que cada una de sus caras o vértices, son simples de la siguiente dimensión más pequeña.

La programación lineal ha tomado la palabra simplex de la geometría y la ha aplicado al proceso de localizar un vértice óptimo entre los vértices de una región factible, de dimensión "n+1".

El método simplex es un proceso iterativo que permite realizar una exploración dirigida del conjunto de puntos extremos de la región factible, lo cual conlleva un notable ahorro en tiempo y cálculos.

George Dantzig utilizó el método de eliminación de Gauss-Jordan, para la formulación del método simplex.

Por medio del método de Gauss-Jordan, resolvemos todos los sistemas que dan una solución, lo cual conduce a un considerable número de cálculos, incluso para los problemas de menor talla.

El número total de sistemas que habría de resolverse para un sistema de "m" ecuaciones y "n" incógnitas, si todas las submatrices de orden "m" fueran regulares, sería igual a

$$C_n^m = \frac{n!}{m!(n-m)!}$$

De tal forma que un problema con veintiuna ecuaciones y veinticuatro incógnitas, exigiría la resolución de alrededor de 2 024 sistemas de veintiún ecuaciones con veinticuatro incógnitas!! La mayoría de estos cálculos serían inútiles, pues hay sistemas de orden "m" que no tienen solución.

Para el caso de la determinación de la ruta crítica o mínima de una red, utilizando una matriz de incidencia unimodular para obtener una solución inicial básica factible —previo a la aplicación del método simplex— hasta llegar a la solución óptima factible, sólo necesitamos resolver 38 sistemas de veintiuna ecuaciones y veinticuatro incógnitas!!

2.1 Técnicas previas a la aplicación del algoritmo simplex

Antes de aplicar el método simplex a cualquier problema de programación lineal, debemos considerar que tenemos que reducir el sistema de desigualdades lineales a una forma estándar (variables de holgura), pero si existen variables de holgura negativas, producto de una desigualdad del tipo II, tendremos que insertar las variables artificiales para obtener una forma canónica que nos permita una solución inicial básica factible, como paso previo a la aplicación del algoritmo simplex.

Para más claridad, diremos que el método simplex se inicia con un programa lineal que está en forma estándar y que, a su vez, el algoritmo simplex se inicia cuando el programa está en la forma canónica, el cual consiste de una serie de operaciones pivote y forma la subrutina principal del método simplex.

Como veremos más adelante, si un problema contiene variables de holgura del tipo II, el método simplex observará, en su aplicación, la ejecución de la fase I y fase II, para llegar a la solución óptima factible.

2.2 La tabla extendida

2.2.1 Variables de holgura

Cuando vamos a transformar un sistema de desigualdades lineales a un sistema de igualdades lineales, para así tenerlo en la forma estándar, y luego le insertamos ciertas variables de "holgura", para poder aplicar el método simplex, debemos de considerar lo siguiente:

2.2.1.1 Variables de holgura del tipo I

Se presentan cuando tenemos una desigualdad, así:

$$a_{11} X_1 + a_{12} X_2 + a_{13} X_3 + \dots + a_{1n} X_n \leq b_1$$

Para convertirla a una igualdad de "forma estándar", le agregamos una variable de holgura positiva y la igualdad queda así:

$$a_{11} X_1 + a_{12} X_2 + a_{13} X_3 + \dots + a_{1n} X_n + X_{n+1} = b_1$$

con $X_{n+1} \geq 0$

Si todas las desigualdades son del tipo I, solamente necesitaremos aplicar la fase I del método simplex, ya que la forma estándar coincide con la forma canónica.

2.2.1.2 Variables de holgura del tipo II

Se presentan cuando tenemos una desigualdad así:

$$a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n \geq b_1$$

Para convertirla a una igualdad de "forma estándar", le agregamos una variable de holgura negativa y la igualdad queda así:

$$a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n - X_{n+1} = b_1$$

con $X_{n+1} \geq 0$

Para poder aplicar el algoritmo simplex, necesitamos una solución inicial básica factible con variables de holgura positivas. Como este caso es la excepción, tendremos que insertar seguidamente las llamadas variables artificiales y, entonces, habrá necesidad de ejecutar la fase I y II del método simplex, para obtener la solución óptima factible.

2.2.2 Variables artificiales

Como vimos en los ejemplos anteriores, las desigualdades lineales las pusimos en forma estándar con poco o ningún esfuerzo.

Por ejemplo, la adición de variables de holgura con coeficientes positivos suministra las variables básicas necesarias para la solución básica factible inicial. Por otra parte, el sistema de restricciones con desigualdades de la forma \geq no contiene soluciones básicas tan obvias.

Lo que debe desarrollarse es una técnica para determinar una solución básica factible inicial para cualquier sistema arbitrario de ecuaciones. Esta técnica, también, debe ser capaz de manejar un problema que no tiene solución factible. Tal problema podría surgir, por ejemplo, en un modelo que contiene un simple error en la formulación o en un modelo de producción compleja, donde no son obvias las necesidades con un limitado número de recursos disponibles.

La idea básica del método utilizado para hallar una solución básica factible inicial es simple.

Introducimos dentro del problema un número suficiente de variables llamadas variables artificiales, para convertir el sistema de restricciones en la forma canónica, con estas variables como las variables básicas. Luego aplicamos el método simplex, no a la función objetivo del problema original, sino a una nueva función definida en tal forma que su valor mínimo es obtenido en una solución factible para el problema original.

Considérese le sistema de desigualdades lineales

$$a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n \leq b_1$$

$$a_{21} X_1 + a_{22} X_2 + \dots + a_{2n} X_n \geq b_2$$

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$a_{m1} X_1 + a_{m2} X_2 + \dots + a_{mn} X_n = b_m$$

2.2.2.1

Que minimiza la función:

$$C_1 X_1 + C_2 X_2 + \dots + C_n X_n = Z \quad 2.2.2.2$$

Asumimos que todas las constantes b_i , $i = 1, \dots, m$, son positivas.

Ahora introducimos dentro del sistema de restricciones, "m" nuevas variables:

$X_{n+1}, X_{n+2}, \dots, X_{n+m}$ llamadas "variables artificiales", una por cada

ecuación. El sistema resultante es:

$$\begin{array}{rcl}
 & a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n + X_{n+1} & = b_1 \\
 2.2.2.3 & a_{21} X_1 + a_{22} X_2 + \dots + a_{2n} X_n + X_{n+2} & = b_2 \\
 & \vdots & \\
 & \vdots & \\
 & \vdots & \\
 & a_{m1} X_1 + a_{m2} X_2 + \dots + a_{mn} X_n + X_{n+m} & = b_m
 \end{array}$$

Nótese que el sistema está en la forma estándar en las variables X_{n+1}, \dots, X_{n+m} . Ahora, considérese el problema de determinar el valor máximo de la función

$$W = X_{n+1} + \dots + X_{n+m} \quad 2.2.2.4$$

Al aplicar el simplex, si se llegara a una solución factible óptima, la función W tomaría el valor de cero para cualquier solución factible de 2.2.2.2 momento en el que todas las variables artificiales estarán a nivel cero.

De otro modo, cuando aplicamos el método simplex a la función W y si llegamos a un paso en el que no podemos sino pivotar valores de W que son

mayores que cero, podemos concluir que el problema original no tiene soluciones factibles

Antes de hacer el problema ejemplo, tenemos que tomar en cuenta las siguientes observaciones de naturaleza técnica.

Primero, antes de aplicar el algoritmo simplex a la función

$$2.2.2.5 \quad W = X_{n+1} + \dots + X_{n+m}$$

El problema debe estar en forma canónica y eso se logra colocando las variables artificiales como variables básicas, consiguiendo con ello una solución básica factible, con el único problema de que la función W no está expresada sólo en términos de las variables no básicas. Para rectificar esto, a cada ecuación del sistema de restricciones se le debe sustraer la ecuación que define la función W en 2.2.2.5.

Segundo, si las operaciones pivote conducidas por el problema de minimizar W , se ejecutan simultáneamente en la ecuación:

$$2.2.2.2 \quad Z = C_1 X_1 + C_2 X_2 + \dots + C_n X_n$$

que define la función objetivo del problema original, entonces tendremos que expresar esta función en términos de variables no básicas a cada paso. De tal forma que si una solución inicial factible básica está a la vista en el problema original, el método simplex puede reiniciarse inmediatamente en la función Z ,

incorporándola, así, dentro de la notación y las operaciones del problema de minimizar W .

En resumen, el primer paso del problema general en (1), reside en considerar el problema de minimizar $W_0 + W$, con:

$$\begin{array}{rcl}
 a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n + X_{n+1} & = & b_1 \\
 a_{21} X_1 + a_{22} X_2 + \dots + a_{2n} X_n - X_{n+2} & = & b_2 \\
 \vdots & & \vdots \\
 \vdots & & \vdots \\
 a_{m1} X_1 + a_{m2} X_2 + \dots + a_{mn} X_n + X_{n+m} & = & b_m \\
 C_1 X_1 + C_2 X_2 + \dots + C_n X_n & = & Z \\
 d_1 X_1 + d_2 X_2 + \dots + d_n X_n + X_{n+2} & = & W_0 + W \\
 \text{Y } d_j = -(a_{1j} + a_{2j} + \dots + a_{mj}) \text{ y } W_0 = -(b_1 + b_2 + \dots + b_m)
 \end{array}$$

2.2.2.6

Al proceso de minimizar la función $W_0 + W$, se le llama la fase I del método simplex, y al de minimizar la función Z , se le llama la fase II del método simplex.

Cuando en un sistema de desigualdades lineales, no existen desigualdades de la forma (\leq), únicamente aplicamos la fase II, ya que a la hora de insertar las variables de holgura del tipo I, nuestra forma estándar coincidirá con la forma canónica del tipo II.

Ejemplo de variables de holgura tipo I

$$\text{Minimizar } 2X_1 - 3X_2 + X_3 + X_4 = Z$$

$$\text{Sujeto a } X_1 - 2X_2 - 3X_3 - 2X_4 \leq 3$$

$$X_1 - X_2 + 2X_3 + X_4 \leq 11$$

$$\text{con } X_1, X_2, X_3, X_4 \geq 0$$

Introduciendo variables de holgura positivas, X_5, X_6 , obtenemos:

$$\text{Minimizar } 2X_1 - 3X_2 + X_3 + X_4 = Z$$

$$\text{Sujeto a } X_1 - 2X_2 - 3X_3 - 2X_4 + X_5 = 3$$

$$X_1 - X_2 + 2X_3 + X_4 + X_6 = 11$$

El cual está en forma estándar, ya que la matriz de coeficientes tendría un vector columna identidad, en las columnas de X_5 y X_6 .

$$\left[\begin{array}{cccccc|c} 1 & -2 & -3 & -2 & 1 & 0 & 3 \\ 1 & -1 & 2 & 1 & 0 & 1 & 11 \\ \hline 2 & -3 & 1 & 1 & 0 & 0 & 0 \end{array} \right] \begin{array}{l} \text{Ecuación uno} \\ \text{Ecuación dos} \\ Z \end{array}$$

Esta forma estándar nos da una solución inicial básica factible previo a la aplicación del algoritmo simplex, que es lo que llamamos la tabla extendida.

Ejemplo de variables de holgura tipo II

$$\text{Minimizar } X_1 + X_2 + X_3 = Z$$

$$\text{Sujeto a } -X_1 + 2X_2 + X_3 \leq 1$$

$$-X_1 + 2X_3 \geq 4$$

$$X_1 - X_2 + 2X_3 = 4$$

$$\text{con } X_j \geq 0$$

Agregando variables de holgura tipo I y II, obtenemos:

$$\begin{array}{rcl}
 \text{Minimizar} & X_1 + X_2 + X_3 & = Z \\
 \text{Sujeto} & -X_1 + 2X_2 + X_3 + X_4 & = 1 \\
 & -X_1 + 2X_3 - X_5 & = 4 \\
 & X_1 - X_2 + 2X_3 & = -4
 \end{array}$$

lo cual es un problema, porque no podemos aplicar el algoritmo simplex.

Ejemplo de variables artificiales

Si siguiendo con el ejemplo de las variables de holgura tipo II, vemos que en ese estado de forma estándar, no podemos iniciar el proceso de optimización. Agregándole variables artificiales X_6 y X_7 , nos queda:

$$\begin{array}{rcl}
 \text{Minimizar} & X_1 + X_2 + X_3 & = Z \\
 \text{Sujeto a} & -X_1 + 2X_2 + X_3 + X_4 & = 1 \\
 & -X_1 + 2X_3 - X_5 + X_6 & = 4 \\
 & X_1 - X_2 + 2X_3 + X_7 & = 4 \\
 & X_6 + X_7 & = W
 \end{array}$$

Debiendo agregarle una función W , definida por las variables artificiales. El siguiente paso es obtener una solución inicial básica factible, o sea reducir el sistema anterior a la forma canónica.

Sustituyendo la segunda y tercera ecuaciones de W , nos queda la ecuación

$$X_2 - 4X_3 + X_5 = W - 8$$

Ahora, la expresión no contiene las variables básicas X_4 , X_6 y X_7 .

El sistema queda así:

$$\begin{array}{rcccccccl} -X_1 & +2X_2 & +X_3 & +X_4 & & & & = & 1 \\ -X_1 & & +2X_3 & & -X_5 & +X_6 & & = & 4 \\ X_1 & -X_2 & +2X_3 & & & & +X_7 & = & 4 \\ X_1 & +X_2 & +X_3 & & & & & = & Z \\ & X_2 & -4X_3 & & +X_5 & & & = & W - 8 \end{array}$$

que está en la forma canónica. Matricialmente:

X_1	X_2	X_3	X_4	X_5	X_6	X_7	b_i	
-1	2	1	1	0	0	0	1	
-1	0	2	0	-1	1	0	4	
1	-1	2	0	0	0	1	4	
1	1	1	0	0	0	0	0	$C_j - Z_j$
0	1	-4	0	1	0	0	-8	$W + W_0$

Ahora sí podemos aplicar el algoritmo simplex, que explicaremos más adelante.

2.3 La tabla condensada

2.3.1 Método matricial

Antes de aplicar la maquinaria del simplex a cualquier problema formulado en términos de programación lineal, necesitaremos una solución inicial básica factible, que puede obtenerse mediante la inserción de variables artificiales y de holgura, según sea el caso, o mediante la reducción de nuestra matriz de coeficientes a la forma canónica.

Al desarrollar el método simplex, George Dantzig utilizó el proceso clásico de eliminación de Gauss-Jordan, muy familiar para todos aquellos que han resuelto un sistema de ecuaciones lineales.

De tal forma que, un sistema de ecuaciones estará en forma canónica si: Posee un conjunto de vectores columna unitarios.

Ejemplo 1:

P_1	P_2	P_3	P_4	P_5	P_6	b_i
1	5	-2	0	4	0	-21
0	4	1	1	0	0	8
0	-2	7	0	3	1	4
0	0	0	0	0	0	0
0	0	0	0	0	0	0

Los vectores P_1 , P_4 y P_6 son vectores columna unitarios, cuyos "unos" aparecerían exactamente en la diagonal si los reagrupamos.

Esta matriz, tal y como se presenta, tiene una solución inicial básica factible que permite la aplicación inmediata del algoritmo simplex. Este método matricial es una alternativa al de insertar variables artificiales.

Ejemplo 2: Resolver el sistema siguiente:

$$\begin{array}{rcccccc}
 2X_1 & +X_2 & -3X_3 & & X_5 & +X_6 & = & 4 \\
 X_1 & -X_2 & & +2X_4 & -X_5 & -X_6 & = & 0 \\
 3X_1 & +2X_2 & +2X_3 & +X_4 & +X_5 & & = & 1 \\
 & X_2 & -X_3 & -X_4 & +2X_5 & -3X_6 & = & 2
 \end{array}$$

La forma matricial es

Luego de operar obtenemos:

$$\left[\begin{array}{cccccc|c}
 2 & 1 & -3 & 0 & -1 & 1 & 4 \\
 1 & -1 & 0 & 2 & -1 & -1 & 0 \\
 3 & 2 & 2 & 1 & 1 & 0 & 1 \\
 0 & 1 & -1 & -1 & 2 & -3 & 2
 \end{array} \right]
 \quad
 \left[\begin{array}{cccccc|c}
 1/2 & 0 & -5/3 & 0 & 0 & 1 & 3/2 \\
 3/2 & 0 & -3/2 & 1 & 0 & 0 & 2 \\
 3/2 & 0 & -37/6 & 0 & 1 & 0 & 6 \\
 0 & 1 & 29/6 & 0 & 0 & 0 & -7/2
 \end{array} \right]$$

Esta última matriz está en forma canónica, ya que posee cuatro vectores unitarios que al agruparlos forman una matriz identidad cuadrada.

2.4 El método simplex

El primer paso en el método simplex para resolver el problema general, consiste en convertir el sistema de "m" desigualdades lineales:

$$\begin{array}{rcl}
 a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n & \leq & b_1 \\
 a_{21} X_1 + a_{22} X_2 + \dots + a_{2n} X_n & \leq & b_2 \\
 \vdots & & \vdots \\
 \vdots & & \vdots \\
 a_{m1} X_1 + a_{m2} X_2 + \dots + a_{mn} X_n & \leq & b_m
 \end{array} \tag{2.2.2.1}$$

A un sistema de "m" ecuaciones lineales. Esto se hace mediante la inserción de las llamadas "variables de holgura". $X_{n+1}, X_{n+2} \dots X_{n+m}$, una en cada desigualdad, de manera que 2.2.2.1 se convierte en:

$$\begin{array}{rcl}
 a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n + X_{n+1} & = & b_1 \\
 a_{21} X_1 + a_{22} X_2 + \dots + a_{2n} X_n + X_{n+2} & = & b_2 \\
 \vdots & & \vdots \\
 \vdots & & \vdots \\
 a_{m1} X_1 + a_{m2} X_2 + \dots + a_{mn} X_n + X_{n+m} & = & b_m
 \end{array} \tag{2.2.2.3}$$

El problema de la programación lineal se puede presentar ahora como sigue: De entre todos los conjuntos de puntos $(X_1, X_2, \dots, X_{n+m})$ que satisfacen 2.2.2.3, con cada X_j no negativa, se trata de hallar aquel punto $(\bar{X}_1, \bar{X}_2, \dots, \bar{X}_{n+m})$ que minimiza la función lineal:

$$Z = C_1 X_1 + C_2 X_2 + \dots + C_n X_n \tag{2.2.2.2}$$

Una vez que el punto de máximo $(\bar{X}_1, \bar{X}_2, \dots, \bar{X}_{n+m})$ se ha encontrado, la respuesta al problema original es simplemente el subconjunto $(\bar{X}_1, \bar{X}_2, \dots, \bar{X}_n)$. Las variables de holgura son únicamente un artificio para llegar a la solución óptima, y pueden o no tener un significado físico, en problemas individuales.

El teorema básico del método simplex dice, en efecto, que el conjunto $(X_1, X_2, \dots, X_{m+n})$ que maximiza 2.2.2.2 sujeta a 2.2.2.3, debe necesariamente poseer "m" elementos que son ceros. Si conociéramos cuales coordenadas son cero en el óptimo, podríamos insertar sus valores en 2.2.2.3, este último sistema sería entonces de "m" ecuaciones con "m" incógnitas y podría resolverse algebraicamente para hallar el óptimo. Sin ese conocimiento, debemos contentarnos con un procedimiento secuencial que nos lleve a la solución.

El procedimiento se describe así: Escogemos "K" de las "n+m" coordenadas y les asignamos el valor cero. Usando estos valores, resolvemos las ecuaciones de restricción para las "m" coordenadas que quedan. Si cualquiera de ellas resulta negativa, se desecha esta solución porque viola la restricción de positividad y se busca otro conjunto de "K" coordenadas a las que se da el valor cero. Si alguna de las coordenadas resulta negativa, tenemos lo que se llama una solución factible, que es el punto de partida del proceso iterativo. La técnica del simplex nos permite probar la optimalidad de la solución factible y, en caso de que no sea óptima, proceder directamente a una solución factible diferente, con una Z mejor. Finalmente, la técnica debe llevarnos a una solución óptima.

2.5 El algoritmo simplex

Los siguientes pasos se aplican cuando el problema está en forma canónica.

2.5.1 Fase I

2.5.1.1 Optimización de la función $-W$

2.5.1.1.1 Si todos los $d_i = 0$. Pasamos de inmediato a optimizar la función "Z".

2.5.1.1.2 Si algunos $d_i > 0$; los eliminamos del proceso iterativo.

2.5.1.1.3 Si existen $d_i < 0$, procedemos al proceso de seleccionar la columna de entrada, así:

$$d_s = \text{Mínimo } d_i \text{ (el más negativo)}$$

2.5.1.2

2.5.1.2.1 Si todos los $a_{is} \leq 0$, no hay solución factible. La función objetivo $-W$ no tiene cota inferior.

2.5.1.2.2 Si existen algunos $a_{is} > 0$, escogemos de entre ellos la fila de entrada que, al intersecarla con el " d_i " mínimo, nos indicará el vector pivote. Esa fila la escogemos de la siguiente relación:

$$\frac{b_r}{a_{rs}} = \text{Mínimo} \left\{ \frac{b_i}{a_{is}} \mid a_{is} > 0 \right\}$$

Obsérvese que la condición $a_{is} > 0$, evita que suceda 2.5.1.2.1

2.5.1.3

- 2.5.1.3.1 Si después de pivotar, observamos que todos los $\bar{d}_i > 0$, implica que no hay soluciones factibles.
- 2.5.1.3.2 Si luego de pivotar, algunos $\bar{d}_i < 0$ regresamos al paso 2.5.1.1.3.
- 2.5.1.3.3 Si todos los $\bar{d}_i = 0$, pasamos de inmediato a optimizar la función "Z".

2.5.2 Fase II

Optimización de la función "Z".

2.5.2.1

- 2.5.2.1.1 Si todos los $\bar{c}_i > 0$, hemos alcanzado el valor mínimo de la función.
- 2.5.2.1.2 Si algunos $\bar{c}_i < 0$, escogemos de entre ellos la columna de entrada, así:

$$C^*s = \text{Mínimo } \bar{C}_i \text{ (el más negativo)}$$

- 2.5.2.1.3 Si algunos $\bar{C}_i > 0$, los eliminamos del proceso iterativo

2.5.2.2

- 2.5.2.2.1 Si todos los $\bar{a}_{is} \leq 0$, no hay solución factible. La función "Z" no tiene cota inferior.

2.5.2.2.2 Si existen algunos $\bar{a}_{is} > 0$, escogemos de entre ellos la fila de entrada.

$$\frac{br^*}{a^*rs} = \text{Mínimo} \left\{ \frac{\bar{b}_i}{\bar{a}_{is}} \mid \bar{a}_{is} > 0 \right\}$$

2.5.2.3 Si después de pivotar, algunos $\bar{C}_i < 0$. Regresamos al paso 2.5.2.1.2 de la fase II.

2.5.2.3.1 Si después de pivotar, todos los $\bar{C}_i \geq 0$. Hemos alcanzado el valor mínimo de la función primal "Z", es decir, la solución óptima factible.

2.5.3 Tabla del método simplex

Vectores básicos	Variables admisibles			Variables artificiales				Constantes b_i
	X_1	X_s	X_n	X_{n+1}	X_{n+2}	...	X_{n+m}	
X_{n+1}	a_{11}	a_{1s}	a_{1n}	1	0	...	0	b_1
X_{n+2}	a_{21}	a_{2s}	a_{2n}	0	1	...	0	b_2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
X_{n+m}	a_{m1}	a_{ms}	a_{mn}	0	0	0	1	b_m
-Z	C_1	C_s	C_m	0	0	0	0	0
-W	d_1	d_s	d_n	0	0	0	0	$-\sum b_i$

□ : Indica el vector columna de entrada.

Tabla 2.1

○ : Indica la fila de entrada.

/// : Indica el vector "no útil" y que debe eliminarse para los futuros cálculos.

Problema ejemplo de la fase I y II

$$\text{Minimizar } X_1 + 6X_2 - 7X_3 + X_4 + 5X_5 = Z$$

$$\text{Sujeto a } 5X_1 - 4X_2 + 13X_3 - 2X_4 + X_5 \leq 20$$

$$X_1 - X_2 + 5X_3 - X_4 + X_5 \leq 8$$

ya que los términos constantes son positivos, iniciamos la fase II del método simplex con el siguiente sistema aumentado:

$$5X_1 - 4X_2 + 13X_3 - 2X_4 + X_5 + X_6 = 20$$

$$X_1 - X_2 + 5X_3 - X_4 + X_5 + X_7 = 8$$

$$X_1 + 6X_2 - 7X_3 + X_4 + 5X_5 - Z = 0$$

que está en forma estándar.

Reduciéndolo a la forma canónica y sustrayéndole a la última ecuación la suma de las primeras dos, nos queda:

$$5X_1 - 4X_2 + 13X_3 - 2X_4 + X_5 + X_6 = 20$$

$$X_1 - X_2 + 5X_3 - X_4 + X_5 + X_7 = 8$$

$$X_1 + 6X_2 - 7X_3 + X_4 + 5X_5 - Z = 0$$

$$-6X_1 + 5X_2 - 18X_3 + 3X_4 - 2X_5 - W = -28$$

Aplicando ahora el algoritmo simplex, la tabla del método simplex queda:

CICLO 1 DE LA FASE 1

BASE	X1	X2	X3	X4	X5	X6	X7		
X6	5	-4	13	2	1	1	0	20	20/13
X7	1	-1	5	-1	1	0	1	8	8/5
-Z	1	6	-7	1	5	0	0	0	—
-W	-6	5	-18	3	-2	0	0	-28	—

VECTOR PIVOTE: X3; VECTOR SALIENTE: X6

CICLO 2 DE LA FASE 1

BASE	X1	X2	X3	X4	X5	X7		
X3	5/13	-4/13	1	-2/13	1/13	0	20/13	20
X7	-12/13	7/13	0	-3/13	8/13	1	4/13	1/2
-Z	48/13	50/13	0	-1/13	72/13	0	140/13	—
-W	12/13	-7/13	0	3/13	-8/13	0	-4/3	—

VECTOR PIVOTE: X5; VECTOR SALIENTE: X7

CICLO 3; FIN DE LA FASE 1 E INICIO DE LA FASE 2

BASE	X2	X3	X5		
X3	-3/8	1	0	3/2	4/7
X5	7/8	0	1	4/8	4/7
-Z	-1	0	0	8	—
-W	0	0	0	0	—

VECTOR PIVOTE: X2; VECTOR SALIENTE: X5

CICLO 4 Y FIN DE LA FASE 2

BASE	X2	X3		
X3	0	1	12/7	—
X2	1	0	4/7	—
-Z	0	0	60/7	—

LA SOLUCIÓN ÓPTIMA SE OBTIENE PARA $Z = -60/7$, EN LAS COORDENADAS $(0, 4/7, 12/7, 0, 0)$.

2.6 Análisis gráfico del método simplex

2.6.1 Maximización

Nuestra incapacidad para visualizar más de tres dimensiones, limita severamente nuestra comprensión para la interpretación geométrica; por lo que nos será muy útil proceder por analogía con ejemplos relativamente simples. Para empezar, consideremos el siguiente problema:

Sea: $X_1 + X_2 = \text{Máximo}$ la función objetivo

Sujeto a: $X_1 + 2X_2 \leq 8$ $X_1 \geq 0$

$X_1 + X_2 \leq 5$ $X_2 \geq 0$

Esto está típicamente planteado en el llamado "espacio solución"*, que utiliza las variables estructurales como coordenadas.

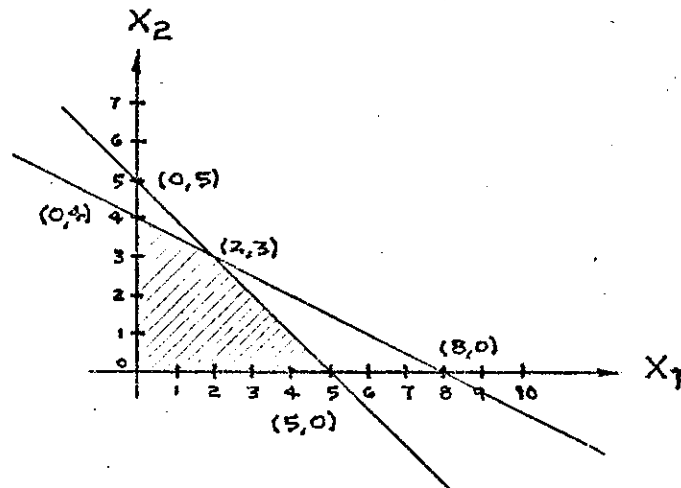


Figura 2.1

* Algunos lo llaman "espacio restringido".

La "región factible" es el "polígono convexo" que se indica por un achurado, de forma que cada vértice corresponde a una solución básica. Y "el número total de soluciones básicas no debe ser mayor que el número total de variables".

En nuestro problema, la función objetivo es $f(X_1, X_2) = X_1 + X_2$. Si consideramos "Z" como un parámetro y graficamos $Z = X_1 + X_2$, para varios valores de Z, tendremos una familia de líneas paralelas con pendiente -1.

Algunas de estas líneas es probable que intersecten la región factible y entonces habrá muchas soluciones factibles, en tanto que otras no acertarán y, por lo tanto, no habrá solución factible. Nuestro interés, es hallar la línea de esta familia, que intersecta la región factible y que es la más alejada del origen, así como tiene el valor más grande para Z.

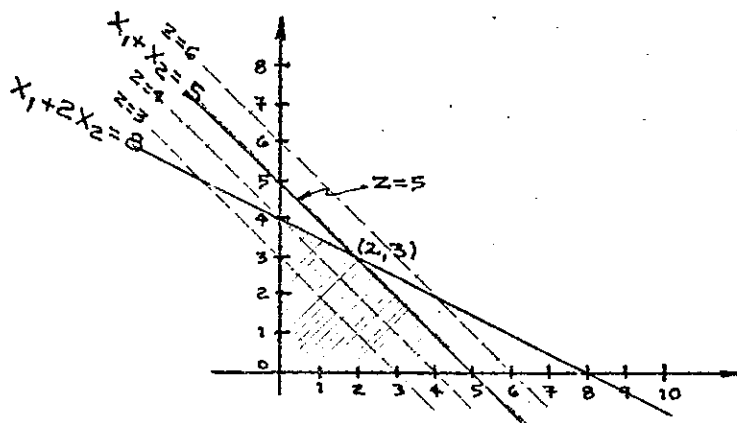


Figura 2.2

De la gráfica, en la figura 2.2, vemos que para $M = 5$, se efectúa el trabajo. Intersecta a la región factible en el vértice $(2, 3)$, pero esta solución no es única.

Obviamente, cualquier punto factible sobre la línea $X_1 + X_2 = 5$, da una solución óptima; "siempre y cuando esté localizada entre los vértices $(2, 3)$ y $(5, 0)$ ".

2.6.2 Minimización

Vamos a considerar otro ejemplo de un problema lineal, un caso en el cual la función objetivo debe minimizarse.

Sea: $13X_1 + 5X_2 = \text{Mínimo} = f(X_1, X_2)$

Sujeto a: $2X_1 + X_2 \geq 11$

$$X_1 + 4X_2 \geq 12$$

Como en el primer ejemplo, graficamos la región cerrada correspondiente a cada restricción y hallamos la intersección de estos planos. Nótese que ahora la región factible, indicada por el achurado, es infinita en extensión, ya que incluye todas las líneas que están arriba de las funciones restrictivas. Un máximo, en este caso, sería imposible de hallar. Lo que ahora buscamos es un mínimo. Consideremos el parámetro $Z = 13X_1 + 5X_2$ y dibujemos la línea para varios valores de Z . De esta forma, estamos buscando una línea de esta familia de líneas, que interseccione a la región factible y que esté tan cerca como sea posible del origen.

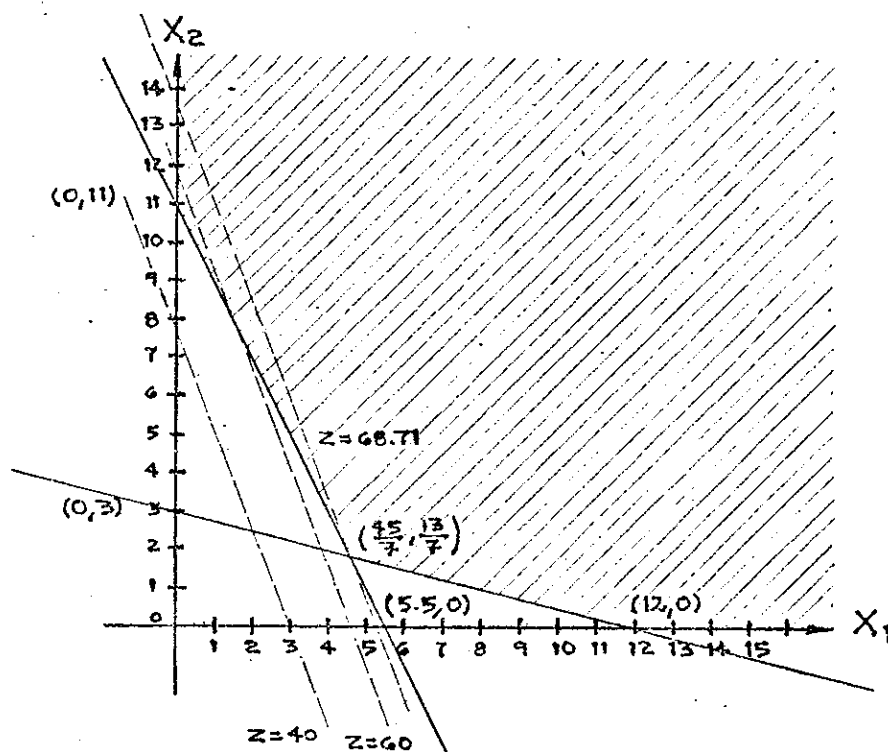


Figura 2.3

Note en la figura 2.3 que la línea apropiada es $Z = 68.71$ y que la solución factible óptima, ocurre en el vértice $(\frac{32}{7}, \frac{13}{7})$. Aquí observamos que la solución es única, a diferencia del ejemplo anterior.

Tanto el análisis gráfico de maximizar o minimizar una función objetivo, sujeta a ciertas restricciones lineales, nos muestra claramente la forma en que el método simplex trabaja.

2.7 Interpretación tridimensional del método simplex

La siguiente interpretación tridimensional del procedimiento simplex, fue desarrollada por Hoffman, Mannos, Sokolowsky y Wiegmann.

Sean los P'_1, P'_2, \dots, P'_n vectores, del espacio dimensional $(m + 1)$; que se obtienen de los coeficientes C_1, C_2, \dots, C_n respectivamente. Al adjuntarlos a los vectores columna p_1, p_2, \dots, p_n del espacio dimensional " m ". Es decir, para cada P'_j el correspondiente c_j , llega a ser la $(m + 1)$ ava. coordenada. Sea " C " el cono convexo del espacio $(m + 1)$, determinado por P'_1, P'_2, \dots, P'_n . Sea " B ", una línea en el espacio $(m + 1)$, que consiste de todos los puntos, cuyas primeras m coordenadas son b_1, b_2, \dots, b_m . Supondremos que esta línea es generada por un vector P'_0 del espacio dimensional $(m + 1)$, cuyas primeras " m " coordenadas son b_1, b_2, \dots, b_m . Y en cuyas $(m + 1)$ avas. coordenadas toma todos los valores. El objetivo del cálculo, es hallar el punto más bajo de " B ", que está también en " C "; eso es el punto de " B " en " C " cuyas $(m + 1)$ avas coordenadas, es un mínimo.

La técnica del cálculo es como sigue:

Suponemos que los primeros " m " vectores P'_1, P'_2, \dots, P'_m son linealmente independientes y tienen la propiedad de que el cono " m " dimensional " D ", que ellos forman contiene un punto de " B ".

Esta propiedad, es equivalente a decir que los "m" vectores son la base admisible, o sea, corresponden a una solución factible.

El hiperplano* que contiene a "D", divide a los restantes vectores P^j en dos grupos. Un grupo está sobre el lado del hiperplano que contiene las $(m+1)$ avas coordenadas positivas del eje.

El segundo grupo contiene a todos los vectores que están debajo del hiperplano y que están disponibles para ser seleccionados como una vieja base de vectores. Cada vector de este segundo grupo, puede unirse al hiperplano que contiene "D", por un segmento de línea paralelo a las $(m+1)$ avas coordenadas del eje.

* HIPERPLANO: "Una condición lineal en E_2 , tal como $X_1 M_1 + X_2 M_2 = a$, donde X_1 , X_2 y "a" son constantes, definen una línea. Una condición lineal en E_3 define un plano. El correspondiente objeto definido por una condición lineal en E_n es llamado un hiperplano. Un hiperplano $H(X, a)$ en E_n , es el conjunto de todos los vectores "U", tales que $X \cdot U = a$, para una dada $X \neq 0$ y un número real "a". Un hiperplano divide E_n en dos medios, que son $H^+(X, a) = (U | X \cdot U \geq a)$; $H^-(X, a) = (U | X \cdot U \leq a)$.

Sea P^k , el vector con la propiedad de que su "segmento de línea" es el más largo. Esto es equivalente a seleccionar el vector con el $\text{MAX } (Z_j - C_j) > 0$. Luego P^k y cierto conjunto de los $m - 1$ vectores P^1, P^2, \dots, P^m tienen la propiedad de que el cono m -dimensional que ellos forman, contienen un punto de B y este punto será más bajo que la intersección de D con B .

Reemplazamos el vector descartado, por P^k y continuamos el proceso hasta que hallamos una solución mínima.

Ejemplo: Consideremos el problema de minimizar:

$$5X_1 + 3X_2 + 4X_3 + 6X_4$$

Sujeto a las condiciones:

$$X_1 + 3X_2 + 5X_3 + 6X_4 = 3$$

$$6X_1 + 4X_2 + 2X_3 + X_4 = 2$$

$$\text{con } X_j \geq 0$$

De tal forma que obtenemos los vectores:

$$P^1 = \begin{pmatrix} 1 \\ 6 \\ 5 \end{pmatrix} \quad P^2 = \begin{pmatrix} 3 \\ 4 \\ 3 \end{pmatrix} \quad P^3 = \begin{pmatrix} 5 \\ 2 \\ 4 \end{pmatrix} \quad P^4 = \begin{pmatrix} 6 \\ 1 \\ 6 \end{pmatrix}$$

Y los ploteamos según la figura 2.4

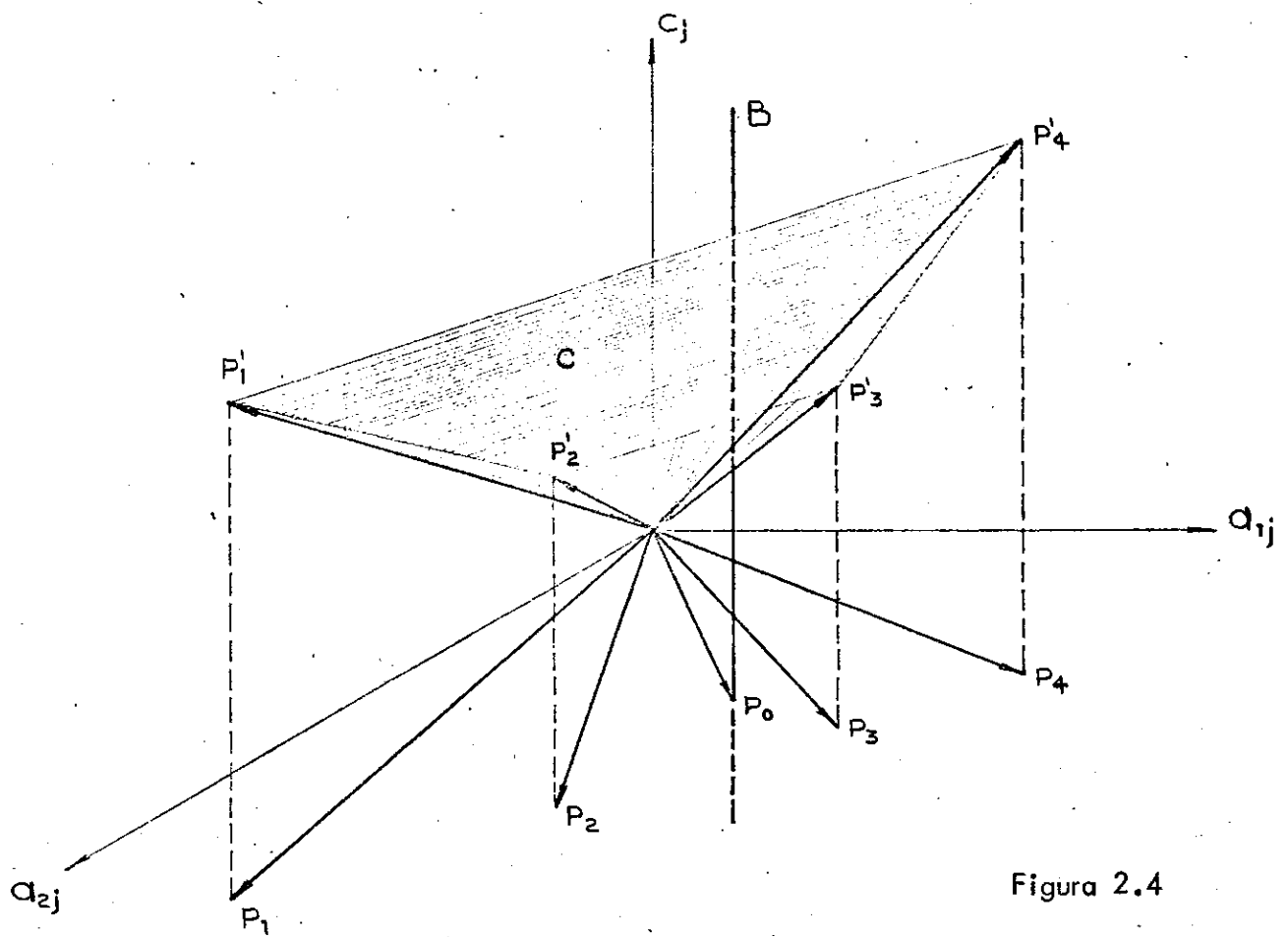


Figura 2.4

Sabemos que existen soluciones factibles, desde el momento en que la línea B intersecta al cono C. Fácilmente podemos demostrar que P'_1 y P'_4 son linealmente independientes y que P_0 puede expresarse como una combinación positiva de P_1 y P_4 ; determinamos el cono bi-dimensional formado por los vectores P'_1 y P'_4 (figura 2.5). El punto P_2 está debajo del hiperplano que contiene a D, una distancia Z_2 .

$(X_{1j} C_1 + X_{2j} C_2 + \dots + X_{mj} C_m = Z_j$ con $j = 1, \dots, n)$ y P'_2 está debajo del hiperplano que contiene a D, una distancia $Z_2 - C_2$. El valor correspondiente para la función objetivo, está dado por Z.

Debido a que $Z_2 - C_2 > Z_3 - C_3$, introducimos P_2 a la base.

De la figura (2.5), vemos que P_0 puede expresarse como una combinación positiva de los vectores P_2 y P_4 , pero no puede expresarse como una combinación positiva de P_2 y P_1 ; es decir, P_0 no está contenido en el cono generado, por los vectores P_1 y P_2 . De esta forma, introducimos P_2 a la base y eliminamos P_1 , obteniendo una nueva base con P_2 y P_4 . Para esta nueva base, podemos realizar un análisis similar, luego del cual introduciremos a P_3 dentro de la nueva base, eliminando a P_4 .

Esta nueva base, formada por los vectores P_2 y P_3 , corresponde a la solución mínima.

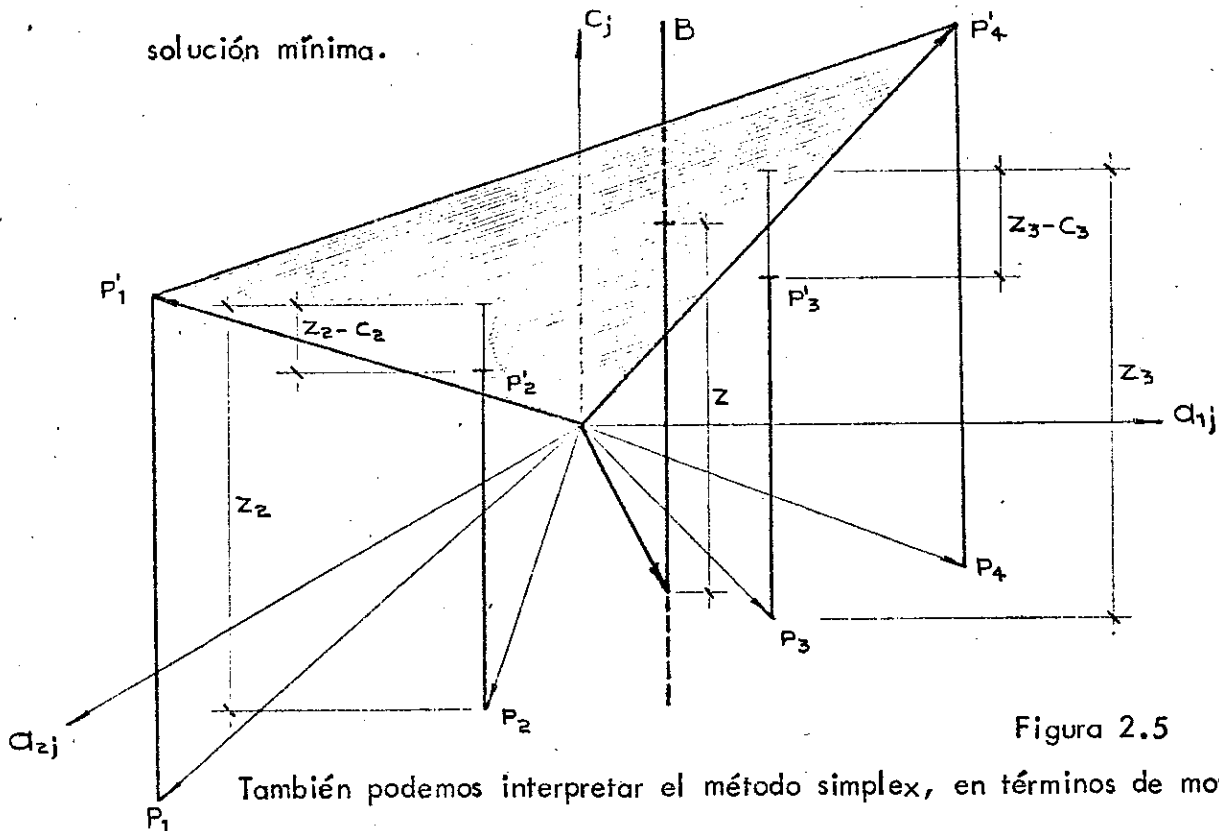


Figura 2.5

También podemos interpretar el método simplex, en términos de movernos de un punto extremo a otro punto extremo adyacente.

2.8 Solución no acotada

Una solución no acotada ocurre cuando una función objetivo, que está siendo maximizada, puede igualar al $+\infty$ o cuando una función objetivo que está siendo minimizada puede igualar al $-\infty$. El siguiente ejemplo ilustra esta situación.

$$\begin{aligned} \text{Maximizar} \quad & X_1 + 2X_2 = Z \\ \text{Sujeta a} \quad & X_1 - X_2 \leq 5 \\ & X_1 \leq 10, \quad \text{con } X_i \geq 0 \end{aligned}$$

Un análisis gráfico de este problema, muestra que la región factible se extiende a lo largo del eje X_2 hasta el infinito. Debido a que la función objetivo contiene a la variable X_2 con un coeficiente positivo, aumentará indefinidamente así como X_2 incrementa.

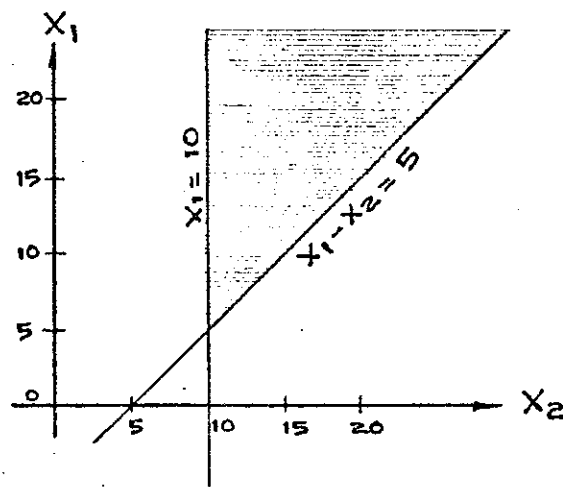


Figura 2.6

El problema escrito en forma canónica, queda así:

$$\text{Maximizar } X_1 + 2X_2 = Z$$

$$\text{Sujeto a } X_1 - X_2 + X_3 = 5$$

$$X_1 + X_4 = 10$$

Y la tabla para aplicar el simplex queda de esta forma:

	X_1	X_2	X_3	X_4	b_i	b_i/a_{ij}
X_3	1	-1	1	0	5	-
X_4	1	0	0	1	10	$10/0 = +\infty$
Z	1	2	0	0	0	

Tabla 2.2

Como se observa en la tabla, la variable columna-entrada es el "2".

Pero el único coeficiente positivo disponible para localizar al vector de salida es cero, pero $10/0 = +\infty$ el problema no tiene cota superior!!

2.9 Degeneración

Dos formas de degeneración, pueden ocurrir en los modelos de la aplicación del algoritmo simplex de programación lineal, los cuales trataremos seguidamente: La redundancia y el ciclo.

2.9.1 Redundancia

Esta resulta de restricciones redundantes —pero consistentes— y no causa dificultad, excepto por el cálculo aritmético extra.

El problema de la redundancia, reside en que en el sistema original de restricciones, una ecuación o algunas ecuaciones son "combinaciones lineales" del resto de ecuaciones del sistema. Esto ocurre muy a menudo y sucede cuando se introducen dentro del problema, un número mayor de variables que las necesarias.

Claramente, sin embargo, si un sistema de ecuaciones está en forma canónica, no pueden haber ecuaciones redundantes a causa de la naturaleza aislada de las variables básicas. Parecería —a primera vista— necesario eliminar las ecuaciones redundantes del sistema de restricciones, antes que la maquinaria del simplex se aplique al problema de la programación lineal. Afortunadamente, este no es el caso. En el ejemplo que desarrollaremos, demostraremos que las redundancias existentes en el sistema original de restricciones pueden descubrirse y que las deficiencias causadas por la pérdida de un conjunto

completo de variables básicas del conjunto original de variables, puede compensarse por la presencia de las variables artificiales.

Así, supóngase que después de varias operaciones pivote, el valor de la función "W" llega a ser cero. Si a esta altura ninguna variable artificial permanece en la "base", el sistema original de restricciones debe estar en forma canónica y no contiene ecuaciones redundantes. La fase II del método simplex de minimizar o maximizar la función "Z" puede empezar seguidamente.

Nótese que "W" es la suma de las variables artificiales y que hemos alcanzado un punto en el proceso simplex en el cual el valor de "W" asociado a la solución básica factible es cero, por lo tanto, esas variables artificiales deben estar a nivel cero o, en otras palabras, los términos constantes " b_i^* " de las ecuaciones de restricciones que contienen las variables artificiales debenser cero. Es en este momento cuando intentamos reemplazar estas variables artificiales de la base con variables del conjunto original. Pero si seguimos pivoteando y llegamos a un punto en que los coeficientes correspondientes a la "ecuación redundante" son cero, es decir, los $a_{ij}^* = 0$, podemos concluir que a causa de las redundancias, es imposible hallar un conjunto completo de "n" variables básicas, del conjunto original.

En efecto, el número de ecuaciones redundantes igualaría al número de variables artificiales que permanecen en la fila con coeficientes cero. Sin

embargo, la fase II del método simplex —de optimizar la función "Z"— puede iniciarse en la tabla y las filas de ceros correspondientes al resto de variables artificiales, ignorarse (ver tercera iteración del ejemplo, con X_7).

En esencia, esta tabla que ignora las filas ceros, corresponde a un sistema de ecuaciones independientes en forma canónica y equivalente al sistema original de restricciones.

Seguidamente, el ejemplo ilustrativo.

Ejemplo de redundancia:

$$\begin{array}{rcl}
 \text{Minimizar} & X_1 + 4X_2 + 3X_3 + 2X_4 & = Z \\
 \text{Sujeto a} & X_1 + 2X_2 + \quad \quad + X_4 & = 20 \\
 & 2X_1 + X_2 + X_3 & = 10 \\
 & -X_1 + 4X_2 - 2X_3 + 3X_4 & = 40 \\
 & & \text{con } X_i \geq 0
 \end{array}$$

Agregando las variables artificiales X_5 , X_6 y X_7 ; y expresando $W = X_5 + X_6 + X_7$, en términos de X_1 , X_2 , X_3 , X_4 ; el sistema queda en la siguiente forma canónica.

$$\begin{array}{rcl}
 X_1 + 2X_2 + \quad \quad + X_4 + X_5 & = & 20 \\
 2X_1 + X_2 + X_3 + \quad \quad + X_6 & = & 10 \\
 -X_1 + 4X_2 - 2X_3 + 3X_4 + X_7 & = & 40 \\
 X_1 + 4X_2 + 3X_3 + 2X_4 & = & Z \\
 -2X_1 - 7X_2 + X_3 - 4X_4 & = & -70 + W
 \end{array}$$

APLICANDO AHORA EL ALGORITMO SIMPLEX:

CICLO 1 DE LA FASE 1

BASE	X1	X2	X3	X4	X5	X6	X7	bi	br/dir
X5	1	2	0	1	1	0	0	20	10
X6	2	①	1	0	0	1	0	10	10
X7	-1	4	-2	3	0	0	1	40	10
-Z	1	4	3	2	0	0	0	0	
-W	-2	⑦	1	-4	0	0	0	-70	

VECTOR DE SALIDA: X6 (NÓTESE LA PARIDAD EN LAS RELACIONES br/dir.
ESCOGEMOS EL dir MÁS PEQUEÑO O SEA 1.)

VECTOR DE ENTRADA: X2.

TABLA .2.3

CICLO 2 DE LA FASE 1

X5	-3	0	-2	①	1	0	0	0
X2	2	1	1	0	0	0	10	—
X7	-9	0	-6	3	0	1	0	0
-Z	-7	0	-1	2	0	0	-40	
-W	12	0	8	④	0	0	0	

VECTOR DE ENTRADA: X4; VECTOR DE SALIDA: X5 (LA PARIDAD EN LAS RELACIONES br/dir ES QUEBRADA POR EL dir > 0, QUE ES EL MÍNIMO.)

TABLA 2.4

CICLO 3; FIN DE LA FASE 1 E INICIO DE LA FASE 2

X4	-3	0	②	1	0	0	—
X2	②	1	1	0	0	10	5
X7	0	0	0	0	1	0	—
-Z	①	0	3	0	0	-40	
-W	0	0	0	0	0	0	

VECTOR PIVOTE: X1; VECTOR SALIENTE: X3.

TABLA 2.5

CICLO 4 Y FIN DE LA FASE 2

X4	0	3/2	1	0	15	—
X1	1	1/2	0	0	5	—
X7*	0	0	0	1	0	—
-Z	0	1/2	0	0	-35	—

*LA VARIABLE X7 ES UNA VARIABLE ARTIFICIAL, LO CUAL IMPLICA LA EXISTENCIA DE LA REDUNDANCIA.

TABLA 2.6

2.9.1.1 Interpretación geométrica de la redundancia

Geoméricamente, un problema "degenerado por redundancia", es aquél en el cual el vector P_0 descansa sobre un plano limitado o una orilla del cono convexo, definido por sus vectores básicos.

Así, en la siguiente figura, el vector P_0 puede expresarse como una combinación lineal positiva de P_1 y P_2 ; asimismo, puede expresarse como una combinación lineal positiva de P_1 , P_2 y P_3 , teniendo a $X_3 = 0$.

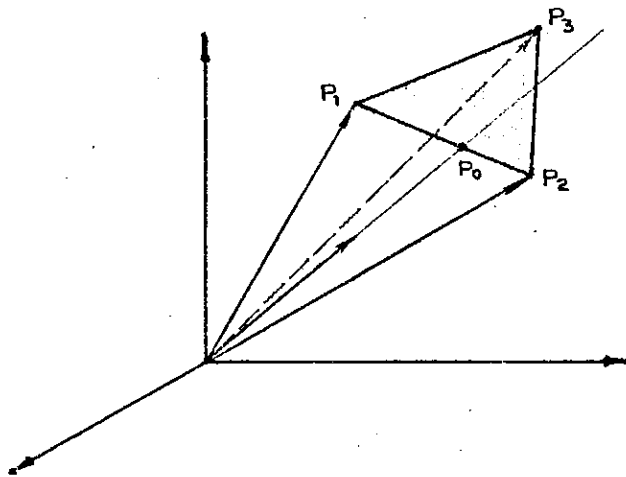


Figura 2.7

2.9.2 El ciclo

A pesar de que no existan restricciones "redundantes", algunas variables "básicas iniciales", pueden permanecer a nivel "cero" a través de varias iteraciones. Pero cuando se escoge una de estas variables como variable de salida, entonces ocurre "el ciclo", en el cual el procedimiento iterativo retorna a la misma "solución básica inicial" y el ciclo continúa indefinidamente.

A estas alturas sabemos que la cantidad de soluciones básicas factibles, en un problema de programación lineal, es finito.

Existe degeneración por ciclo, cuando una o varias variables básicas tienen el valor "cero".

La condición necesaria y suficiente para que, al menos, una variable básica tome el valor cero, "es que entre en la base en lugar de una variable que ya es nula" y "que el mínimo calculado con el criterio del simplex, no sea único" (estas dos condiciones no se excluyen mutuamente).

Si existe la "degeneración" y no se ha alcanzado una solución básica factible óptima, ocurrirá el ciclo y retornaremos a la solución inicial básica factible, obteniendo el mismo valor para la función objetivo "Z".

Otra respuesta al origen del ciclo y cómo se le determina en la práctica, es la siguiente: "Las soluciones degeneradas son posibles cuando los

términos constantes " b_i ", del vector solución, tienen una relación muy especial con los coeficientes de las variables básicas. Esto es muy claro, ya que el proceso de reducción de uno de los conjuntos finitos de formas canónicas, depende únicamente de los coeficientes y NO del vector columna solución " b_i ".

A pesar de todo lo anterior, es una práctica muy común basada en la solución de millares de problemas de programación lineal por el método simplex, el que cada problema está muy cerca de la degeneración (semidegeneración) en alguna etapa del proceso y ninguno ha formado jamás el ciclo.

Podría pensarse que, debido a que la semidegeneración ocurre muy frecuentemente, habrían muchos casos de ciclo.

Sin embargo, hasta la fecha nadie ha informado de la aparición de un caso de ciclo, excepto en los problemas teóricos construidos por A. J. Hoffmann en 1951 y E. M. L. Beale en 1955.

El ejemplo de Hoffman, consistía de tres ecuaciones y once variables y la solución inicial básica factible se repetía cada 10 ciclos indefinidamente.

Por su parte, el de Beale tenía tres ecuaciones y siete variables y la solución inicial básica factible se repetía cada 7 iteraciones.

Seguidamente, el problema ejemplo debido a Beale:

$$\begin{aligned}
 \text{Minimizar} \quad & -3/4 X_1 + 150X_2 - 1/50 X_3 + 6X_4 & = Z \\
 \text{Sujeto a} \quad & 1/4 X_1 - 60X_2 - 1/25 X_3 + 9X_4 + X_5 & = 0 \\
 & 1/2 X_1 - 90X_2 - 1/50 X_3 + 3X_4 + X_6 & = 0 \\
 & X_3 + X_7 & = 1 \\
 & & \text{con } X_i \geq 0
 \end{aligned}$$

2.9.2.1 Problema-ejemplo de ciclo de E. M. L. Beale

Basis		B _i	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇
	Z		-3/4	150	-3/50	6	0	0	0
X ₅	0	0	3/4	-60	-3/25	9	1	0	0
X ₆	0	0	1/2	-90	-3/50	3	0	1	0
X ₇	0	1	0	0	1	0	0	0	1
	0		3/4	-150	3/50	-6	0	0	0

CICLO 1

TABLA 2.7

X ₁	-3/4	0	1	-240	-3/25	36	4	0	0
X ₆	0	0	0	30	3/50	-15	-2	1	0
X ₇	0	1	0	0	1	0	0	0	1
	0	0	0	30	3/50	-33	-3	0	0

CICLO 2

TABLA 2.8

X ₁	-3/4	0	1	0	8/25	-84	-12	8	0
X ₂	150	0	0	1	3/500	-3/2	-3/15	3/50	0
X ₇	0	1	0	0	1	0	0	0	1
	0	0	0	0	3/25	-18	-1	-1	0

CICLO 3

TABLA 2.9

X3	$-\frac{3}{50}$	0	$\frac{25}{8}$	0	1	$-\frac{525}{8}$	$-\frac{75}{2}$	25	0
X2	150	0	$-\frac{1}{160}$	1	0	$\frac{3}{40}$	$\frac{3}{120}$	$-\frac{3}{60}$	0
X7	0	1	$-\frac{25}{8}$	0	0	$\frac{525}{8}$	$\frac{75}{2}$	-25	1
		0	$-\frac{1}{4}$	0	0	-3	2	-3	0

CICLO 4

TABLA 2.10

X3	$-\frac{3}{50}$	0	$-\frac{125}{2}$	10,500	1	0	50	-150	0
X4	6	0	$-\frac{1}{4}$	40	0	1	$\frac{3}{8}$	$-\frac{3}{2}$	0
X7	0	1	$\frac{125}{2}$	-10,500	0	0	-50	150	1
		0	$\frac{3}{2}$	-120	0	0	1	-1	0

CICLO 5

TABLA 2.11

X6	0	0	$-\frac{3}{4}$	210	$\frac{3}{50}$	0	1	-3	0
X4	-6	0	$\frac{3}{8}$	-30	$-\frac{1}{50}$	1	0	$\frac{3}{8}$	0
X7	0	1	0	0	1	0	0	0	1
		0	$\frac{3}{4}$	-330	$-\frac{3}{50}$	0	0	2	0

CICLO 6

TABLA 2.12

X5	0	0	$\frac{3}{4}$	-60	$-\frac{1}{50}$	9	1	0	0
X6	0	0	$\frac{3}{2}$	-90	$-\frac{1}{50}$	3	0	1	0
X7	0	1	0	0	1	0	0	0	1
		0	$\frac{3}{4}$	-150	$\frac{3}{50}$	-6	0	0	0

CICLO 7

TABLA 2.13

EL CICLO 7 ES EXACTAMENTE IGUAL AL CICLO 1. EL PROCESO CONTINÚA INDEFINIDAMENTE PER SECLA SECLORUM!!

Desde el punto de vista matemático, era necesario garantizar la validez del algoritmo simplex para cualquier caso, utilizando un procedimiento que evitara la repetición de una base.

El problema principal era hallar la forma de evitar el ciclo, que incluyera poco trabajo extra, hasta donde fuera posible.

Fue así como se pensó en "reemplazar el problema original por un problema generalizado, en el que no fuera posible la degeneración por ciclo y que, por consiguiente, garantizara una variación NO NULA de la función a optimizar".

Los dos procedimientos más conocidos, para evitar la degeneración por ciclo, son:

- 1.0 La técnica de las perturbaciones, desarrollada por A. Charnes; y
- 2.0 El método del ordenamiento léxicográfico de los vectores, desarrollado paralelamente por G. Dantzig, A. Orden y P. Wolfe.

El método lexicográfico posee la ventaja de "no utilizar más que la noción algebraica de Orden", en tanto que el de Charnes, se apoya en la noción de límite.

2.9.2.2 La técnica de las perturbaciones

De la interpretación geométrica de la redundancia, vamos a utilizar los mismos vectores para explicar la técnica de las perturbaciones y cómo evitar la degeneración por ciclo.

Si perturbamos el vector P_0 , de forma que descansa dentro del cono convexo, determinado por los vectores P_1 , P_2 y P_3 , tendremos una solución básica factible no degenerada. Y esto podemos hacerlo, al tomar una combinación lineal positiva de esos vectores y agregarlos a P_0 , sin que ello altere o destruya el problema original.

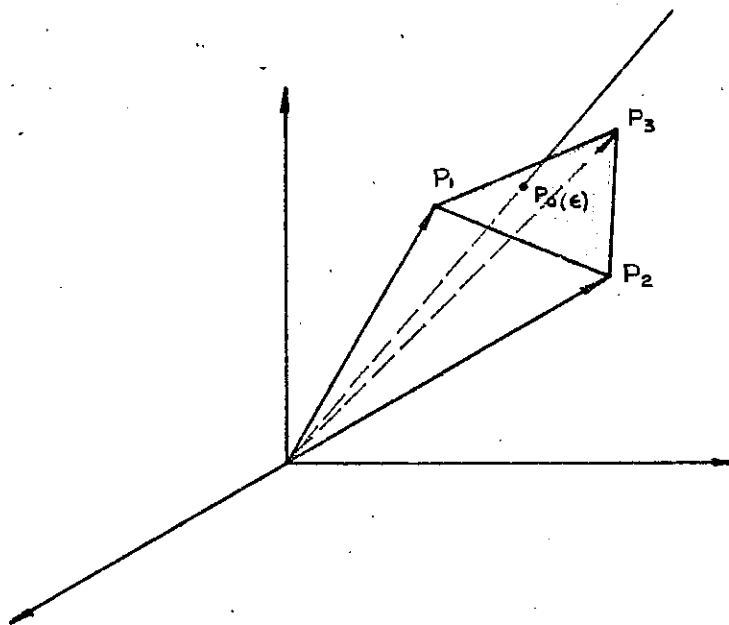


Figura 2.8

Tomamos una combinación lineal positiva, que sea pequeña así:

$$\epsilon P_1 + \epsilon^2 P_2 + \epsilon^3 P_3$$

Donde el exponente de ϵ depende de la columna y ϵ es cualquier número positivo pequeño.

Las restricciones para el nuevo problema serían:

$$P_0(\epsilon) = X_1 P_1 + X_2 P_2 + X_3 P_3 = P_0 + \epsilon P_1 + \epsilon^2 P_2 + \epsilon^3 P_3$$

Ejemplo:

BASE	X1	X2	X3	X4	X5	X6	b _i	b _r /a _{rs}
X1	1	0	-1	5	0	4	2	1/2
X5	0	0	4	3	1	4	2	1/2
X2	0	1	3	1	0	-1	1	-
-Z	0	0	2	-1	0	-3	5	

TABLA 2.14

En este problema vemos que hay paridad en el mínimo $\left\{ \frac{b_i}{a_{ij}} \mid a_{ij} > 0 \right\}$

O sea, la columna que hay que perturbar es la "b_i".

De hecho, la perturbación previene la "degeneración por ciclo" en todas las iteraciones sucesivas, lo cual a su vez garantiza un cambio en el valor del programa de cada iteración.

Este cambio positivo en el programa asegura que ninguna solución puede generarse por segunda vez y, por lo tanto, no puede producirse el ciclo.

Si hacemos $\epsilon = 0.01$, que resulta muy grande, pero adecuado para nuestra perturbación, calcularemos la nueva fila del vector "bi".

$$b'1 = 2 + 1 (0.01) - 1(0.01)^3 + 5 (0.01)^4 + 4 (0.01)^6 = 2.0100011$$

$$b'2 = 2 + 4 (0.01)^3 + 3(0.01)^4 + 1 (0.01)^5 + 4 (0.01)^6 = 2.000004$$

$$b'3 = 1 + 1 (0.01)^2 + 3(0.01)^3 + 1 (0.01)^4 - 1 (0.01)^6 = 1.000103$$

BASE	X1	X2	X3	X4	X5	X6	b _i	b _r /a _{ij}
X1	1	0	-1	5	0	4	2.0100011	0.5025003
X5	0	0	4	3	1	④	2.000004	0.5000001
X2	0	1	3	1	0	-1	1.000103	—
-Z	0	0	2	-1	0	③	5	

TABLA 2.15

Como vemos en la tabla, la paridad se rompió entre los $\left(\frac{b_r}{a_{rj}}\right)_i \leq$ y la próxima solución será "no degenerada".

Los cálculos pueden continuarse hasta obtener la solución óptima. Se elimina la perturbación y se obtiene la solución del problema original.

Seguidamente, el ejemplo de Beale resuelto por la técnica de las perturbaciones.

2.9.2.3 Problema-ejemplo de Beale, resuelto por la técnica de las perturbaciones de A. Charnes.

La primera consideración teórica del problema se debe a Saul I. Gass, que lo resolvió utilizando la técnica de las perturbaciones.

BASE		B_1	X1	X2	X3	X4	X5	X6	X7
	Z		$-\frac{3}{4}$	150	$-\frac{1}{50}$	6	0	0	0
X5	0	0	$\frac{3}{4}$	-60	$-\frac{1}{25}$	9	1	0	0
X6	0	0	$\frac{1}{2}$	-90	$-\frac{1}{50}$	3	0	1	0
X7	0	1	0	0	1	0	0	0	1
		0	$\frac{3}{4}$	-150	$\frac{1}{50}$	-6	0	0	0

CICLO 1

TABLA 2.16

X1	$-\frac{3}{4}$	0	1	-240	$-\frac{1}{25}$	36	4	0	0
X6	0	0	0	30	$\frac{3}{50}$	-15	-2	1	0
X7	0	1	0	0	1	0	0	0	1
		0	0	30	$\frac{1}{50}$	-33	-3	0	0

CICLO 2

TABLA 2.17

X1	$-\frac{3}{4}$	0	1	0	$\frac{8}{25}$	-84	-12	8	0
X2	150	0	0	1	$\frac{1}{500}$	$-\frac{1}{2}$	$-\frac{1}{15}$	$\frac{1}{50}$	0
X7	0	1	0	0	1	0	0	0	1
		0	0	0	$\frac{3}{25}$	-18	-1	-1	0

CICLO 3

TABLA 2.18

X1	$-\frac{3}{4}$	0	1	-160	0	-4	$-\frac{4}{3}$	$\frac{8}{3}$	0
X3	$-\frac{1}{50}$	0	0	500	1	-250	$-\frac{100}{3}$	$\frac{50}{3}$	0
X7	0	1	0	-500	0	250	$\frac{100}{3}$	$-\frac{50}{3}$	1
		0	0	-40	0	2	$\frac{2}{3}$	$-\frac{2}{3}$	0

CICLO 4

TABLA 2.19

X1	$-\frac{3}{4}$	$\frac{3}{25}$	1	-168	0	0	$-\frac{4}{5}$	$\frac{12}{5}$	$\frac{3}{25}$
X3	$-\frac{1}{50}$	1	0	0	1	0	0	0	1
X4	6	$\frac{1}{250}$	0	-2	0	1	$\frac{3}{15}$	$-\frac{1}{15}$	$\frac{1}{250}$
		$-\frac{1}{25}$	0	-36	0	0	$\frac{2}{5}$	$-\frac{11}{5}$	$-\frac{1}{25}$

CICLO 5

TABLA 2.20

X1	$-\frac{3}{4}$	$\frac{3}{25}$	1	-180	0	6	0	2	$\frac{3}{25}$
X3	$-\frac{1}{50}$	1	0	0	1	0	0	0	1
X5	0	$\frac{3}{100}$	0	-15	0	$\frac{15}{2}$	1	$-\frac{1}{2}$	$\frac{3}{100}$
		$-\frac{1}{50}$	0	-15	0	$-\frac{21}{2}$	0	$-\frac{3}{2}$	$-\frac{1}{50}$

CICLO 6

TABLA 2.21

La paridad ocurre en el ciclo I y III. Las primeras tres soluciones son las mismas del problema "degenerado" (tabla anterior).

La principal diferencia se da del ciclo III al IV, donde hemos eliminado X_2 en vez de X_3 . El valor mínima es: $Z = -\frac{1}{20}$ y se logra en las coordenadas $(\frac{1}{25}, 0, 1, 0)$.

La tabla final tiene una ecuación redundante.

CAPITULO 3

3.0 TEOREMA FUNDAMENTAL DE LA DUALIDAD

3.1 Definición del problema dual

Comunmente en matemática existe relación entre conceptos, sistemas o problemas que aparentemente no la tienen pero que, una vez comprendidos, producen muchos dividendos. Por ejemplo, considere la relación que hay en el cálculo, entre la integral y la derivada expresada a través del teorema fundamental del cálculo o en álgebra lineal, la relación que hay entre transformaciones lineales y matrices. Relaciones como éstas no sólo pueden usarse para fines prácticos o computacionales, sino también pueden suministrar una teoría coherente y unificada, de tal forma que la comprensión y las técnicas de un sistema, pueden contribuir a comprender y utilizar otro sistema.

Antes de definir el dual de un problema de programación lineal, necesitaremos que el problema sea expresado en una forma especial llamada la forma primal del problema:

$$\text{Minimizar } Z = C_1 X_1 + C_2 X_2 + \dots + C_n X_n \quad 2.2.2.2$$

$$\text{Sujeto a } a_{11} X_1 + a_{12} X_2 + \dots + a_{1n} X_n \leq b_1$$

$$a_{21} X_1 + a_{22} X_2 + \dots + a_{2n} X_n \leq b_2 \quad 2.2.2.1$$

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$a_{m1} X_1 + a_{m2} X_2 + \dots + a_{mn} X_n \leq b_m$$

$$\text{con } X_1, X_2, \dots, X_n \geq 0$$

De modo que la forma primal de un problema de programación lineal, llamado simplemente el problema primal, es un problema de minimización con un sistema de restricciones compuestas de desigualdades (\leq). Para este ejemplo particular obsérvese que aquí no hay restricciones en los signos de los coeficientes a_{ij} , los términos constantes b_i y los coeficientes C_j . Es de esta forma como a este problema primal asociamos su problema dual:

$$\text{Minimizar } V = b_1 Y_1 + b_2 Y_2 + \dots + b_m Y_m \quad 3.1.1$$

$$\begin{aligned} \text{Sujeto a } & a_{11} Y_1 + a_{21} Y_2 + \dots + a_{m1} Y_m \geq C_1 \\ & a_{12} Y_1 + a_{22} Y_2 + \dots + a_{m2} Y_m \geq C_2 \\ & \vdots \\ & a_{1n} Y_1 + a_{2n} Y_2 + \dots + a_{mn} Y_m \geq C_n \end{aligned} \quad 3.1.2$$

con $Y_1, Y_2, \dots, Y_m \geq 0$

Los componentes del problema dual se relacionan con el problema primal, de la siguiente forma:

3.2.1 Variables: El número de variables positivas del problema dual, iguala al número de desigualdades (\geq) del primal.

3.2.2 Restricciones

3.2.2.1 El número de desigualdades del problema dual iguala al número de variables positivas del primal.

3.2.2.2 Los coeficientes para las desigualdades del problema dual, son los coeficientes de las desigualdades del problema primal, con las filas y columnas transpuestas.

3.2.2.3 Los términos constantes " C_j " para las desigualdades del problema dual, son los coeficientes de la función objetivo " Z " del problema primal.

3.2.3 Función objetivo

3.2.3.1 Los coeficientes para la función objetivo " V ", del problema dual, son los términos constantes " b_j " de las restricciones del problema primal.

3.2.3.2 La función objetivo del problema dual debe maximizarse, en tanto la función objetivo del problema primal debe minimizarse.

Ejemplo:

$$\text{Minimizar} \quad 6X_1 + X_2 + 4X_3 = Z$$

$$\text{Sujeto a} \quad 3X_1 + 7X_2 + X_3 \leq 15$$

$$X_1 - 2X_2 + 3X_3 \leq 20$$

$$\text{con } X_j \geq 0$$

Su correspondiente problema dual, es:

$$\text{Maximizar} \quad 15Y_1 + 20Y_2 = V$$

$$\text{Sujeto a} \quad 3Y_1 + Y_2 \geq 6$$

$$7Y_1 - 2Y_2 \geq 1$$

$$Y_1 + 3Y_2 \geq 4$$

Podemos usar, también, notación matricial para expresar cualquier problema de programación lineal y, en particular, el problema primal y dual más brevemente.

El problema anterior, lo expresamos así:

$$\begin{bmatrix} 3 & 7 & 1 \\ 1 & -2 & 3 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} \leq \begin{bmatrix} 15 \\ 20 \end{bmatrix} \quad C = \begin{bmatrix} 6 \\ 1 \\ 4 \end{bmatrix} \text{ Los coeficientes de la función objetivo "Z"}$$

Siendo $A^T = \begin{bmatrix} 3 & 1 \\ 7 & -2 \\ 1 & 3 \end{bmatrix}$ La transpuesta de la matriz de coeficientes

En forma más abreviada:

$$\begin{array}{l} \text{Minimizar} \\ \text{Maximizar} \end{array} \left. \begin{array}{l} A \cdot X \leq b \\ C \cdot X = Z \\ A^T \cdot Y \geq C \\ b \cdot Y = V \end{array} \right\} \begin{array}{l} \text{Problema primal} \\ X \geq 0 \\ \text{Problema dual} \\ Y \geq 0 \end{array}$$

Cualquier problema primal de programación lineal, puede expresarse como un problema dual. Ya vimos cómo un problema de minimización puede convertirse en uno equivalente de maximización.

De esta forma, cada problema de programación lineal tiene su correspondiente problema dual, en que las variables no restringidas en signo, se pueden reemplazar por variables restringidas en signo.

3.2 Teorema fundamental de la dualidad

John von Neumann fue el primer matemático que reconoció la importancia del principio de dualidad y el que condujo al desarrollo de la prueba del teorema de dualidad, que dice:

"Si, ya sea que, el problema primal o el problema dual, tienen una solución óptima finita, diremos que el otro problema tiene, también, una solución óptima finita y que los extremos de las funciones lineales son iguales, es decir,

$$\text{Mínimo } Z = \text{Máximo } V$$

Además, si un problema tiene una solución óptima, no acotada, diremos que el otro problema, también, no tiene soluciones factibles".

Del teorema anterior, podemos inferir que existen exactamente cuatro diferentes categorías, dentro de las cuales las soluciones a los problemas primal y dual pueden caer. Estas son:

3.2.1 Ambos problemas tienen soluciones factibles

Los conjuntos de valores posibles para las funciones objetivo Z y V , se relacionan en la línea real de la siguiente forma:

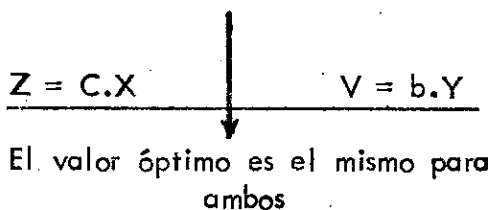


Figura 3.1

3.2.2 La función objetivo "Z" no tiene cota superior y el problema dual no tiene soluciones factibles.

3.2.3 La función objetivo "V" no tiene cota inferior y el problema primal no tiene soluciones factibles.

3.2.4 Ambos problemas no tienen soluciones factibles.

3.3 Teorema dual-simétrico *

Si el problema primal está definido por:

$$A.X \leq b$$

Minimizando

$$C.X = Z$$

$$\text{con } X \geq 0$$

Su problema dual estará definido por:

$$A^T.Y \geq C$$

Maximizando

$$b.Y = V$$

La ventaja del dual simétrico, reside en que no hay necesidad de aumentar el problema dual, reemplazando cada variable dual no condicionada por dos variables positivas.

* La demostración puede verse en R. W. Llewelyn.

3.4 Teorema dual-asimétrico *

Si el problema primal está definido por:

$$\begin{array}{l} \text{Minimizando} \\ \text{con} \end{array} \quad \begin{array}{l} A \cdot X = b \\ C \cdot X = Z \\ X \geq 0 \end{array}$$

Su problema dual estará definido por:

$$\begin{array}{l} \text{Maximizando} \\ \text{con} \end{array} \quad \begin{array}{l} A^T \cdot Y \geq C \\ b \cdot Y = V \\ Y \leq 0 \end{array} \quad \begin{array}{l} \text{No hay condición de positividad para} \\ \text{las componentes de "Y"} \end{array}$$

Algo muy importante es lo siguiente: No es ventajoso resolver un dual asimétrico en vez de su primal, ya que el número de cálculos aumenta considerablemente.

Problema ejemplo:

$$\begin{array}{l} \text{Minimizar} \\ \text{Sujeto a} \end{array} \quad \begin{array}{l} 10X_1 + 5X_2 + 4X_3 = Z \\ 3X_1 + 2X_2 - 3X_3 \geq 3 \\ 4X_1 \quad + 2X_3 \geq 10 \end{array} \quad \text{con } X_j \geq 0$$

* La demostración puede leerse en R. W. Llewelyn.

El problema dual correspondiente es:

$$\begin{array}{rcl}
 \text{Maximizar} & 3Y_1 + 10Y_2 & = V \\
 \text{Sujeto a} & 3Y_1 + 4Y_2 & \leq 10 \\
 & 2Y_1 & \leq 5 \\
 & -3Y_1 + 2Y_2 & \leq 4 \quad \text{con } Y_j \geq 0
 \end{array}$$

Reduciendo el sistema a la forma canónica, obtenemos:

$$\begin{array}{rcl}
 \text{Maximizar} & 3Y_1 + 10Y_2 & = V \\
 \text{Sujeto a} & 3Y_1 + 4Y_2 + Y_3 & = 10 \\
 & 2Y_1 + Y_4 & = 5 \\
 & -3Y_1 + 2Y_2 + Y_5 & = 4 \\
 & & \text{con } Y_j \geq 0
 \end{array}$$

De lo cual podemos decir, que las variables artificiales, únicamente deben insertarse cuando el problema está en su forma dual.

Si en el ejemplo anterior las hubiéramos introducido en el problema primal, tendríamos ahora variables negativas, lo cual nos hubiera obligado a introducir variables artificiales o de lo contrario, multiplicar las restricciones por -1 , de donde tendríamos " C_i " negativos y, entonces, no podríamos aplicar el algoritmo simplex.

La tabla dual simplex es:

Base	Y_1	Y_2	Y_3	Y_4	Y_5	C_i	C_r / A_{rj}
Y_3	3	4	1	0	0	10	$5/2$
Y_4	2	0	0	1	0	5	-
Y_5	-3	②	0	0	1	4	2
-V	-3	-10	0	0	0	0	

Ciclo I
Fase II

Tabla 3.1

Y_3	③	0	1	0	-2	2	$2/9$
Y_4	2	0	0	1	0	5	$5/2$
Y_2	$-3/2$	1	0	0	$1/2$	2	-
-V	-18	0	0	0	5	20	

Ciclo II
Fase II

Tabla 3.2

Y_1	1	0	$1/9$	0	$-2/9$	$2/9$	-
Y_4	0	0	$-2/9$	1	$4/9$	$41/9$	-
Y_2	0	1	$1/6$	0	$1/6$	$7/3$	-
-V	0	0	2	0	1	24	

Ciclo III
Fase II

Tabla 3.3

Desde que todos los " b_j " son ≤ 0 , llegamos a la solución óptima. El valor máximo de " V " o el mínimo de " Z ", es -24 y se obtiene en los puntos $(\frac{2}{9}, \frac{7}{3})$ para la maximización y en los puntos $(2, 0, 1)$ para la minimización, que corresponden a las filas de las variables artificiales.

3.5 Interpretación geométrica del problema dual

Analizaremos el siguiente ejemplo bidimensional:

$$\text{Maximizar } X_1 + X_2 = Z$$

$$\text{Sujeto a } X_1 + 2X_2 \leq 8$$

$$X_1 + X_2 \leq 5 \quad \text{con } X_j \geq 0$$

Su problema dual es:

$$\text{Minimizar } 8Y_1 + 5Y_2 = V$$

$$\text{Sujeto a } Y_1 + Y_2 \geq 1$$

$$2Y_1 + Y_2 \geq 1$$

Habíamos visto que al graficar el problema primal, obteníamos el espacio solución o región factible. De manera análoga, al graficar el problema dual, obtendremos una región factible o espacio solución; nótese que las líneas limitantes o de restricción tienen cota inferior, que es lo inverso del problema primal, que tiene cota superior.

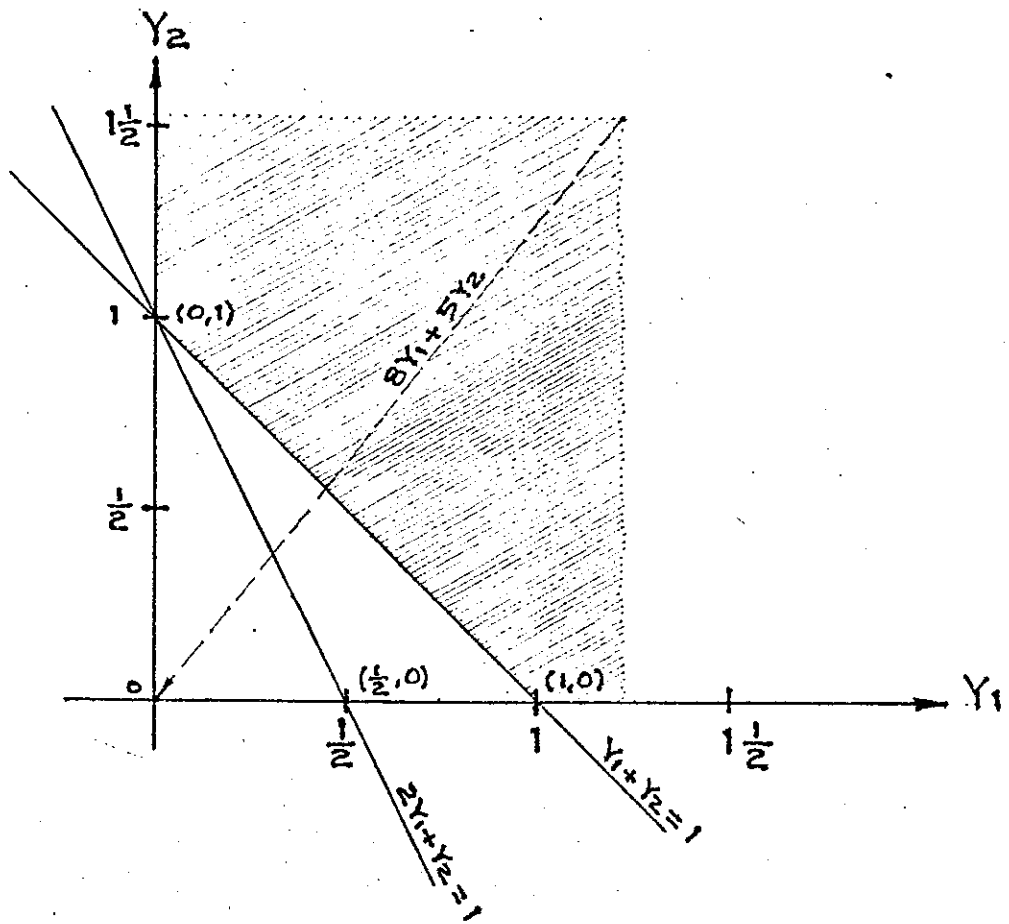


Figura 3.2

PARTE II

R E D E S

Hace más de un siglo, James Maxwell y Gustav Kirchoff descubrieron ciertos principios básicos del análisis de redes durante sus estudios sobre circuitos eléctricos. Desde luego, el análisis de redes se ha convertido en una herramienta muy importante para la investigación de sistemas eléctricos.

A principios del siglo XX, los ingenieros de teléfonos de Europa y Estados Unidos, utilizaban los métodos de redes para determinar la mejor capacidad de las líneas troncales de teléfonos y centros de interrupción, a efecto de garantizar los niveles específicos de serviciabilidad al consumidor.

En 1940, durante el período de la Segunda Guerra Mundial, el desarrollo de la investigación de operaciones, produjo un número de técnicas para el estudio matemático de sistemas a gran escala. El trabajo pionero en el moderno análisis de redes, fue dirigido por Hitchcock en 1941 y por Koopmans en 1947. Desde entonces, el análisis de redes ha sido un área de investigación muy activa y productiva, con alrededor de más de mil artículos publicados. El énfasis de la investigación en los años 50 y principios del 60, fue sobre la formulación de nuevos modelos y desarrollo de nuevos algoritmos.

Posteriormente el énfasis en la investigación, se elevó en extensión, implementación de computadoras y en el análisis de algoritmos y modelos previamente desarrollados.

Se han escrito artículos de investigación por Fulkerson, Bradley, Elmaghraby, Magnanti y Golden, entre otros.

Debido a que el campo del análisis de redes se ha desarrollado al paso de los años, la necesidad de literatura ha aumentado hasta la fecha, para las diferentes orientaciones a varios niveles de discusión.

Se han escrito libros con un tratamiento extensivo de las redes, por Ford y Fulkerson, Charnes y Cooper, Dantzig, Busacker, Jarvis y Minieka.

Modelos de redes y análisis, son ampliamente usados en investigación de operaciones para diversas aplicaciones, tales como el análisis y diseño de sistemas de irrigación a gran escala, redes de computadora, redes de televisión por cable, sistemas de transporte y redes de comunicación vía satélite y tierra.

Eficientes metodologías para redes, se han implementado para resolver problemas industriales tales como el almacenamiento y distribución de bienes, calendarización de actividades, reemplazo de equipo, control de costos, estudios de tráfico, análisis de colas, balance de líneas de ensamble, control de inventarios, etc. para nombrar tan sólo unos pocos.

El análisis de redes no es una disciplina confinada a una sola rama de la academia o industria. La efectividad del enfoque de las redes, descansa en el hecho de que puede aplicarse satisfactoriamente a casi cualquier problema, siempre y cuando el modelador tenga suficiente conocimiento y logre construir la red adecuada.

La ventaja de utilizar modelos de redes, se resume así:

4.1.1 Los modelos de redes representan exactamente muchos sistemas del mundo real.

4.1.2 Los modelos de redes son aceptados más fácilmente por los no analistas, que cualquier otro modelo utilizado en investigación de operaciones.

Este fenómeno parte del hecho de que "una foto es mejor que mil palabras".

Los gerentes parecen aceptar un diagrama de redes más fácilmente que lo que ellos harían por abstraer símbolos.

Adicionalmente, ya que los modelos de redes están a menudo muy relacionados a los problemas físicos, ellos pueden explicársele fácilmente a personas con poco conocimiento.

4.1.3 Los algoritmos de redes permiten soluciones eficientes a algunos modelos de gran escala.

4.1.4 Los algoritmos de redes pueden, a menudo, resolver problemas con muchas más variables y restricciones, que los resueltos por otras técnicas de optimización. Este fenómeno se debe al hecho que el enfoque de una red, a menudo, permite la explicación de estructuras particulares en un modelo.

4.2 Definiciones, notación y simbolismo

4.2.1 Red

Una red consiste de nodos y un conjunto de arcos que conectan a los nodos.

Los nodos no son más que los vértices y los arcos son llamados las orillas, líneas, ramas o eslabones. Una red puede representarse por la notación $G = (N, A)$ donde N es el conjunto de nodos y A es el conjunto de arcos de la red G .

4.2.2 Los nodos

Los nodos de una red pueden representar intersecciones de carreteras, estaciones eléctricas, intercambiadoras de teléfonos, estaciones de ferrocarril, terminales aéreas, tanques de agua, computadoras, etc.

En general, un nodo puede representar un punto donde alguna clase de flujo se origina, se retransmite o termina. Por esta razón, un nodo puede ser visto como un punto de ramificación en una red típica.

4.2.3 Los arcos

Los arcos de una red pueden representar calles, cables de electricidad, líneas telefónicas, rutas aéreas, tuberías de agua, etc.

En algunos casos los arcos no tienen significado físico, pero sirven para dirigir flujo en una secuencia lógica o para mantener una relación de precedencia específica.

Un arco entre el nodo i y el nodo j , se dice que tiene una orientación de i hacia j , si 1 unidad de flujo es enviada desde el nodo i hasta el nodo j a lo largo del arco que incrementaría el valor de la corriente de flujo en el arco, pero lo disminuiría si se enviara en dirección opuesta.

En una red típica, los arcos pueden clasificarse como no dirigidos, dirigidos y bidirigidos.

4.2.3.1 Arcos no dirigidos

Cuando no tienen orientación específica



Figura 4.1

4.2.3.2 Arcos dirigidos

Si tienen exactamente una orientación



Figura 4.2

4.2.3.3 Arcos bidirigidos

Si tienen dos orientaciones

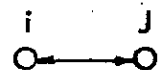


Figura 4.3

Gráficamente utilizamos círculos para representar los nodos y líneas para indicar los arcos y cabezas de flecha para indicar la orientación. También usaremos la notación (i, j) para indicar la dirección del arco que va del nodo i al nodo j .

La cantidad neta de flujo en cualquier nodo es igual a la diferencia entre la cantidad de flujo que sale del nodo y la cantidad de flujo que llega al nodo.

Si la cantidad neta es positiva, el nodo es llamado el nodo fuente.

Si la cantidad neta es negativa, el nodo es llamado el nodo terminal.

Una red puede contener más de un nodo fuente o nodo terminal.

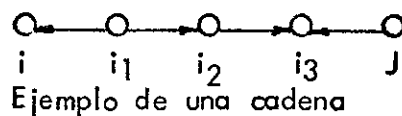
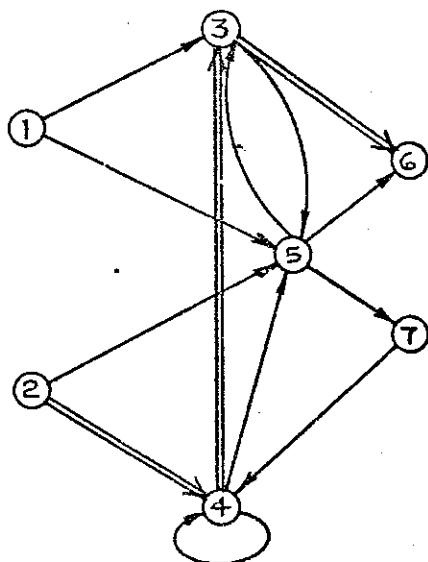


Figura 4.4

En la figura 4.4 los nodos 1 y 2 son nodos fuentes y los nodos 6 y 7 son nodos terminales.

Para el desarrollo de algunos algoritmos, se asume que la red contiene un nodo fuente simple y un nodo terminal simple. Tal red puede construirse, creando un super nodo fuente y un super nodo terminal, conectando las fuentes múltiples al super nodo fuente y las terminales múltiples al super nodo terminal por medio de arcos ficticios.

Un concepto muy importante en el análisis de redes es el de conectar dos nodos i y j , con una secuencia de arcos $E = \{a_1, a_2, \dots, a_k\}$

4.2.4 Una cadena

que va del nodo i al j , es una secuencia de arcos y nodos, de tal forma que el nodo terminal de cada arco, coincide con el nodo inicial del siguiente arco (excluyendo el primero y último nodo en la secuencia). Cada arco de la secuencia está orientado, alejándose del nodo i y hacia el nodo j .

En la figura 4.4 la secuencia de arcos $(1, 3)$, $(3, 5)$, $(5, 3)$ y $(3, 6)$ forman una cadena.

4.2.5 Una ruta

es similar a una cadena, con la excepción de que uno o más arcos, están orientados hacia el nodo j en vez de hacia el nodo i . Esto significa que en una ruta, a diferencia de la cadena, los arcos no están orientados

todos en la misma dirección. La secuencia de arcos $(2, 4)$, $(4, 3)$ y $(3, 6)$ forman una ruta, en la figura 4.4

4.2.6 Red no dirigida

Es la que sus arcos no tienen orientación específica. En este caso, los términos ruta y cadena, se usan indistintamente.

4.2.7 Un ciclo

Es una ruta finita que empieza y termina en el mismo nodo. En la figura 4.4 la secuencia de arcos $(4, 7)$, $(4, 5)$ y $(5, 7)$ forman un ciclo.

4.2.8 Un circuito

Es una cadena finita en el que el primero y el último nodo, son coincidentes.

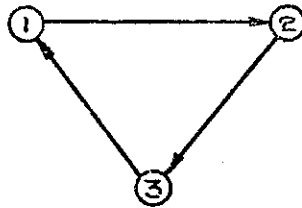


Figura 4.5

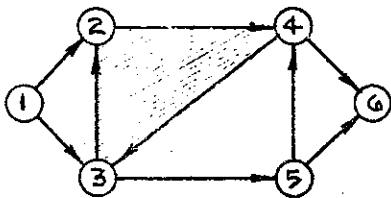
4.2.9 Un loop

Es simplemente un nodo y un arco que forman un circuito y un ciclo. El arco $(4, 4)$ de la figura 4.4 es un loop.

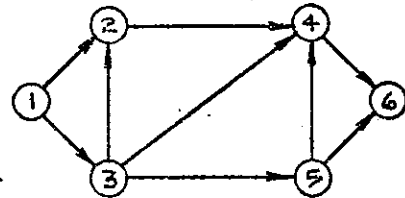
4.2.10 Red sin circuito

Es la que no contiene circuitos.

Este tipo de redes posee características deseables que son explotadas en algunas aplicaciones de rutas más cortas o más largas y tales características forman la lógica central, en la derivación de los algoritmos CPM/PERT.

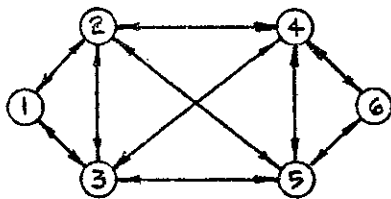


Red con uncircuito

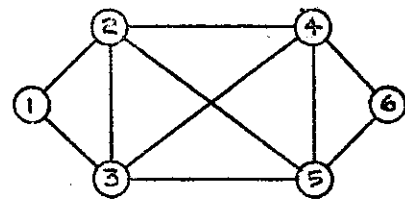


Red sin circuito

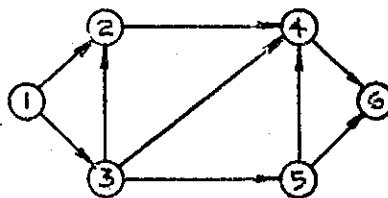
Figura 4.6



Red bidirigida



Red no dirigida



Red dirigida

Figura 4.7

4.3 Representación matricial de las redes

Es evidente que existen diferentes configuraciones de redes, que podran representar a los mismos elementos y relaciones de un sistema dado. Para redes más grandes, la equivalencia de dos formas de redes puede ser difícil de reconocer. En la teoría de gráficos, dos redes que tienen correspondencia uno a uno, en relación a sus arcos y nodos, se dice que son isomórficas. Si dos gráficos son isomorfos, cualquier ecuación definida por uno, también es definida por el otro.

Algunas formas comunes de representación de redes hacen que los cálculos sean más fáciles a revelan características de problemas que facilitan la eliminación de ciertos cálculos o un análisis posterior. Las representaciones gráficas o matriciales son muy útiles desde el punto de vista pedagógico y computacional.

Las características cuantitativas de los arcos, así como las interconexiones entre nodos de una red, pueden representarse por medio de una matriz de distancia o matriz de costo. Las interconexiones de los nodos pueden representarse separadamente por medio de una matriz de adyacencia o matriz de incidencia nodo-arco.

Considérese la red $G = (N, A)$, donde los nodos del conjunto N están numerados de $1, 2, \dots, N$ y donde a cada arco (i, j) del conjunto A se le

asigna un parámetro cuantitativo C_{ij} , comunmente llamado longitud del arco o costo generalizado.

La matriz de costos de la red se define como el arreglo $C = [C_{ij}]$
 $\forall (i, j) \in A$.

Los elementos de C representan las limitaciones naturales o capacidades del sistema, tales como: Distancias, costos de transporte, capacidades de conductos de agua, cargas pico para líneas de transmisión, factibilidades de sistemas de componentes, etc.

Para el caso de una red G no dirigida, $C_{ij} = C_{ji} \forall (i, j) \in A$ en tal caso, C es una matriz simétrica.

4.3.1 Representación matricial de una red dirigida

4.3.1.1 Matriz de Adyacencia

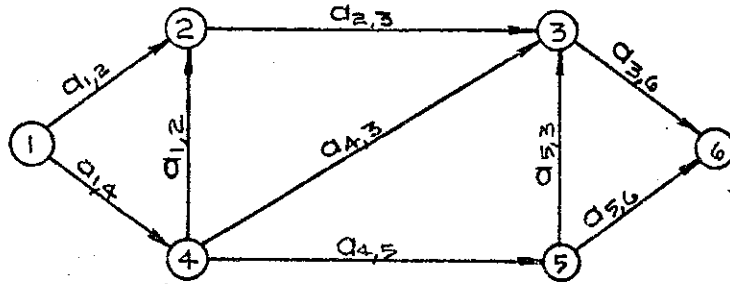
La matriz de adyacencia de una red dirigida G , se define como un arreglo $X = [X_{ij}]$ donde:

$$X_{ij} = \begin{cases} 1 & \text{Si el arco } (i, j) \in A, \text{ se dirige del nodo } i \in N \\ & \text{hacia el nodo } j \in N. \\ 0 & \text{Lo contrario} \end{cases}$$

Considérese la siguiente red dirigida, para la cual el conjunto de nodos N y el conjunto de arcos A , se define así:

$$N = \{i\} = \{1, 2, 3, 4, 5, 6\}$$

$$A = \{a_k\} = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9\}$$



Red dirigida

Figura 4.8

La matriz de adyacencia X es la siguiente:

$X = [X_{ij}]$

	1	2	3	4	5	6
1	0	1	0	1	0	0
2	0	0	1	0	0	0
3	0	0	0	0	0	1
4	0	1	1	0	1	0
5	0	0	1	0	0	1
6	0	0	0	0	0	0

↑ Nodo fuente
 Nodo terminal →
~~Loops~~

Figura 4.9

De esta matriz deducimos lo siguiente:

4.3.1.1.1 Cualquier columna cuyos elementos sean todos cero, representa un nodo fuente.

4.3.1.1.2 Cualquier fila cuyos elementos sean todos cero, representa un nodo terminal.

4.3.1.1.3 Si los elementos de la diagonal son todos ceros, no hay loops en la red. A la inversa, si hay un elemento distinto de cero, entonces hay un loop.

4.3.1.1.4 La matriz no es simétrica respecto a la diagonal.

4.3.1.1.5 La matriz es cuadrada.

4.3.2 Matriz de incidencia nodo-arco

La matriz de incidencia nodo-arco de una red dirigida G , se define como un arreglo $U = [U_{ij}]$, donde:

$$U_{ij} \begin{cases} +1 & \text{Si el nodo } i \in N, \text{ es el nodo de partida del arco } A_k \in A \\ -1 & \text{Si el nodo } j \in N \text{ es el nodo de llega del arco } A_k \in A \\ 0 & \text{Todo lo contrario de lo anterior} \end{cases}$$

Para la red de la figura 4.8, considérese la matriz de incidencia U :

$$U = [U_{ij}] = \begin{matrix} & \begin{matrix} a_{1,2} & a_{1,3} & a_{2,4} & a_{3,2} & a_{4,2} & a_{4,3} & a_{4,5} & a_{5,3} & a_{5,6} \end{matrix} \\ \begin{matrix} \textcircled{1} \\ \textcircled{2} \\ \textcircled{3} \\ \textcircled{4} \\ \textcircled{5} \\ \textcircled{6} \end{matrix} & \begin{bmatrix} +1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & +1 & 0 & -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & +1 & +1 & +1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & +1 & +1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \end{matrix} \begin{matrix} \text{Nodo} \\ \text{fuente} \\ \\ \\ \\ \text{Nodo} \\ \text{terminal} \end{matrix}$$

Figura 4.10

De esta matriz deducimos lo siguiente:

4.3.2.1 Cualquier entrada en la matriz es: $+1$, -1 ó 0 .

4.3.2.2 Cualquier filo cuyos elementos distintos de cero sean todos positivos, representa un nodo fuente.

4.3.2.3 Cualquier filo cuyos elementos distintos de cero, sean todos negativos, representa un nodo terminal.

4.3.2.4 Todas las columnas tienen, como máximo, dos elementos distintos de cero.

4.3.2.5 Los elementos distintos de cero de la diagonal, no constituyen loops.

4.3.2.6 La transpuesta de una matriz de incidencia es otra matriz de incidencia.

4.3.2.7 Si la combinamos con una matriz identidad, la matriz resultante es, también, de incidencia.

4.4 Teoría de la conservación de flujo (Teoría de Ford y Fulkerson)

Al considerar la ejecución de una red, es necesario calcular el valor óptimo de una función, del flujo entre el nodo fuente "S" y el nodo terminal "t". A este cálculo generalmente se le llama un problema de flujo de comercio, debido a que los flujos en los arcos de la red corresponden a una entidad individual tal como energía eléctrica, tráfico aéreo, información, agua, etc.

Sea β_i el conjunto de nodos conectados al nodo i por arcos orientados hacia el nodo i (los arcos que llegan).

Sea α_i el conjunto de nodos conectados al nodo i por arcos orientados que se alejan del nodo i (los arcos que salen).

Sea f_{ij} , i, j una función de valor entero, definida sobre A . Se dice que f_{ij} , i, j es un flujo para una red dirigida $G = (N, A)$ si:

$$f_{i,j} \geq 0 \quad \forall (i,j) \in A \quad 4.4.1$$

$$\sum_{j \in \alpha_i} f_{i,j} - \sum_{i \in \beta_j} f_{j,i} = 0 \quad \forall i \in N; i \neq s, i \neq t \quad 4.4.2$$

$$f_{i,j} \leq C_{i,j} \quad \forall (i,j) \in A \quad 4.4.3$$

$$f_{j,i} \leq C_{j,i} \quad \forall (i,j) \in A \quad 4.4.4$$

En esta discusión, f_{ij} puede interpretarse como un volumen de comercio que fluye a lo largo del arco (i, j) desde el nodo i al nodo j , no excediendo la capacidad del arco C_{ij} .

Si J no es el nodo fuente ni el nodo terminal, la cantidad que fluye hacia el nodo J debe ser igual a la cantidad que fluye hacia afuera de ese nodo. Esta condición se conoce como la propiedad de conservación de flujo.

Las primeras tres restricciones 4.4.1-3 corresponden a la condición de conservación de flujo para la red G .

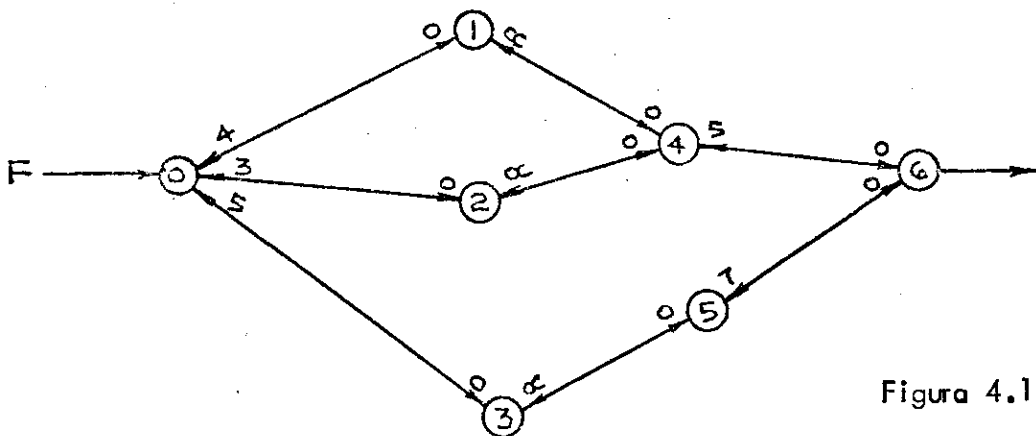


Figura 4.11

Para ejemplificar la propiedad de conservación de flujo, desarrollada por Ford y Fulkerson (1954, 1960), consideraremos la red de la figura 4.11, donde cada arco tiene asignada dos números que representan la capacidad de flujo a lo largo del arco en ambas direcciones.

La condición 4.4.1 nos dice que el flujo en cada arco a lo mínimo es igual a cero. O sea que la condición mínima será cuando no haya flujo.

La condición 4.4.2 nos dice que la suma total de flujos que entran y salen de un nodo debe ser igual a cero.

La condición 4.4.3 nos dice que el flujo a través de las arcos no debe exceder la capacidad de los mismos, tanto de ida como de regreso.

$$\begin{aligned} \text{Ejemplo:} \quad f_{01} &\leq 4 & f_{14} &\leq \infty \\ f_{10} &\leq 0 & f_{41} &\leq 0 \end{aligned}$$

Las últimas dos propiedades serán:

$$F + \sum_i f_{i,0} - \sum_i f_{0,i} = 0 \quad 4.4.5$$

$$\sum_i X_{i,n} - \sum_i X_{n,i} - F = 0 \quad 4.4.6$$

donde F es el flujo que entra al nodo fuente, afuera de la red.

La condición 4.4.5 nos dice que el flujo afuera del sistema iguala a F en el destino del nodo fuente.

La condición 4.4.6 nos dice que el flujo afuera de la red, en el nodo terminal, iguala a F .

CAPITULO 5

5.0 REDES DE PROBLEMAS DE RUTA MINIMA

Si cada arco tiene un parámetro costo que define la longitud del arco, la longitud de una ruta puede definirse como la suma de las longitudes de los arcos de la ruta.

El problema que nos ocupará a partir de este momento, es hallar la ruta mínima (shortest path) desde un nodo fuente específico "S" a un nodo terminal específico "t" o a otros de la red. En el problema de la ruta mínima existen tres casos:

5.1 Todas las longitudes de los arcos son positivas.

5.2 Existen algunos arcos con longitud negativa, pero no existen ciclos de longitud negativa.

5.3 Se presentan uno o más ciclos con longitud negativa.

En el último caso, no será posible resolver el problema de la ruta mínima, ya que los algoritmos conducirán al descubrimiento e identificación de ciclos con longitudes negativas.

El caso que nos ocupará en la presente tesis, será el de resolver problemas de ruta mínima con arcos de longitud positiva.

Seguidamente, asumiremos que el parámetro $C_{i,j}$ de cada arco representa el costo o tiempo requerido para transmitir 1 unidad de flujo desde el nodo i al nodo j .

El problema es determinar el mínimo costo (tiempo) para trasladar 1 unidad de flujo del nodo fuente al nodo terminal, a través de una ruta que lo haga mínima.

Si todos los arcos son dirigidos y hacemos circular exactamente un flujo unitario, los límites para las arcos serán:

$$0 \leq f_{i,j} \leq 1, \text{ con } a_1 = 1 \text{ y } b_n = 1 \quad 4.4.7$$

ya no necesariamente deben tomarse en cuenta, ya que su violación es imposible. Finalmente, asumimos que no hay ciclos negativos en la red.

Matemáticamente, el problema de la ruta mínima, puede formularse como un problema primal de programación lineal, de la siguiente manera:

$$\text{Minimizar } \sum_{i=1}^n \sum_{j=1}^n C_{i,j} f_{i,j} \quad 4.4.8$$

$$\text{Sujeto a } \sum_{i \in N} f_{1,i} - 1 = 0 \quad 4.4.9$$

$$\sum_{j \in N} f_{j,N} - 1 = 0$$

$$\sum_{j=1}^n f_{i,j} - \sum_{j=1}^n f_{j,i} = 0 \quad i \neq 1; i \neq n \quad 4.4.10$$

$$\text{con } f_{ij} \geq 0; (i,j) \in A \quad 4.4.11$$

Es de notar que las restricciones de desigualdad se sostendrán como igualdades estrictas en la solución óptima. Además, la total unimodularidad de las ecuaciones de conservación de flujo, asegura que todos los flujos óptimos serán cantidades enteras, ya que los parámetros: a_i (flujo unitario que entra por el nodo fuente), b_j (flujo unitario que sale por el nodo terminal) y U_{ij} (matriz unimodular) son cantidades enteras.

De las condiciones (9) y (10), vemos que éstas son aplicables al nodo fuente y terminal, respectivamente. En tanto que la condición (11), es aplicable a todos los nodos distintos de los anteriores.

Consideremos ahora el siguiente problema ejemplo:

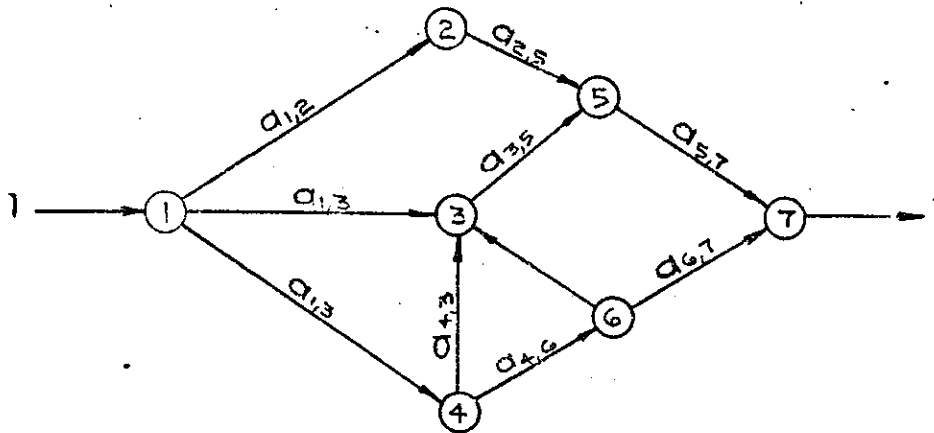


Figura 5.1

El cual puede formularse matemáticamente así:

$$\text{Minimizar } C_{1,2} f_{1,2} + C_{1,3} f_{1,3} + C_{1,4} f_{1,4} + C_{2,5} f_{2,5} + C_{3,5} f_{3,5} + C_{4,3} f_{4,3} + C_{4,6} f_{4,6} + C_{5,7} f_{5,7} + C_{6,3} f_{6,3} + C_{6,7} f_{6,7}$$

Sujeto a

$$\begin{aligned} f_{1,2} + f_{1,3} + f_{1,4} & & & & & & & & & & & -1 & = & 0 \\ -f_{1,2} & & & & + f_{2,5} & & & & & & & & & = & 0 \\ & & -f_{1,3} & & & + f_{3,5} - f_{4,3} & & & & & -f_{6,3} & & & = & 0 \\ & & & -f_{1,4} & & + f_{4,3} + f_{4,6} & & & & & & & & = & 0 \quad 44. \\ & & & & -f_{2,5} - f_{3,5} & & & & + f_{5,7} & & & & & = & 0 \\ & & & & & & -f_{4,6} & & + f_{6,3} + f_{6,7} & & & & & = & 0 \\ & & & & & & & & & -f_{5,7} & & -f_{6,7} + 1 & = & 0 \end{aligned}$$

Si la primera ecuación la igualamos a +1 y la última a -1, obtenemos

la siguiente matriz:

	na- dos	$C_{1,2}$	$C_{1,3}$	$C_{1,4}$	$C_{2,5}$	$C_{3,5}$	$C_{4,3}$	$C_{4,6}$	$C_{5,7}$	$C_{6,3}$	$C_{6,7}$	7
	1	+1	+1	+1	0	0	0	0	0	0	0	+1
	2	-1	0	0	+1	0	0	0	0	0	0	0
$f_{i,i}$	3	0	-1	0	0	+1	-1	0	0	-1	0	0
	4	0	0	-1	0	0	+1	+1	0	0	0	0
	5	0	0	0	-1	-1	0	0	+1	0	0	0
	6	0	0	0	0	0	0	-1	0	+1	+1	0
	7	0	0	0	0	0	0	0	-1	0	-1	-1

Figura 5.2

Maximizar $\pi_1 + 0\pi_2 + 0\pi_3 + 0\pi_4 + 0\pi_5 + 0\pi_6 - \pi_7$

4.4.12

Sujeto a

$$\begin{array}{rccccccc}
 \pi_1 & & & & & & & \leq \delta_{1,2} \\
 \pi_1 & & & -\pi_3 & & & & \leq \delta_{1,3} \\
 \pi_1 & & & & -\pi_4 & & & \leq \delta_{1,4} \\
 & \pi_2 & & & & -\pi_5 & & \leq \delta_{2,5} \\
 & & \pi_3 & & & -\pi_5 & & \leq \delta_{3,5} \\
 & & -\pi_3 & \pi_4 & & & & \leq \delta_{3,4} \\
 & & & \pi_4 & & & -\pi_6 & \leq \delta_{4,6} \\
 & & & & \pi_5 & & -\pi_7 & \leq \delta_{5,7} \\
 & & -\pi_3 & & & \pi_6 & & \leq \delta_{3,6} \\
 & & & & & \pi_6 & -\pi_7 & \leq \delta_{6,7}
 \end{array}$$

4.4.13

En forma resumida, obtenemos: La expresión matemática del problema dual del problema de la ruta mínima

Maximizar $\sum_{i=1}^n \pi_i$ 4.4.14

Sujeto a $\pi_i - \pi_j \leq \delta_{\kappa}, \text{ con } \kappa(i,j) \in N$ 4.4.15

con π_i no restringido para todo i 4.4.16

Problema ejemplo:

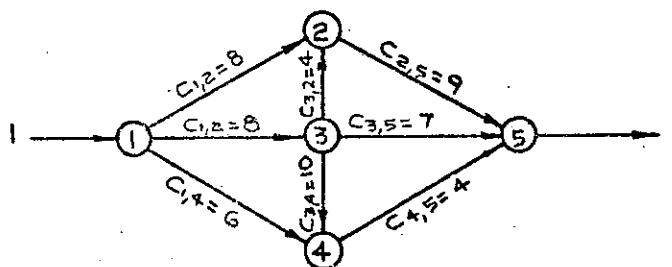


Figura 5.4

El problema primal de la red anterior, es:

$$\text{Minimizar } 8f_{1,2} + 8f_{1,3} + 6f_{1,4} + 9f_{2,5} + 4f_{3,2} + 10f_{3,4} + 7f_{3,5} + 4f_{4,5}$$

$$\begin{array}{rcccccccc} \text{Sujeto a} & f_{1,2} & +f_{1,3} & +f_{1,4} & & & & & & = & 1 \\ & -f_{1,2} & & & +f_{2,5} & -f_{3,2} & & & & = & 0 \\ & & -f_{1,3} & & & +f_{3,2} & +f_{3,4} & +f_{3,5} & & = & 0 \\ & & & -f_{1,4} & & & -f_{3,4} & & +f_{4,5} & = & 0 \\ & & & & -f_{2,5} & & & -f_{3,5} & -f_{4,5} & = & -1 \end{array}$$

Y su correspondiente matriz de incidencia es:

	$f_{1,2}$	$f_{1,3}$	$f_{1,4}$	$f_{2,5}$	$f_{3,2}$	$f_{3,4}$	$f_{3,5}$	$f_{4,5}$	
①	+1	+1	+1	0	0	0	0	0	+1
②	-1	0	0	+1	-1	0	0	0	0
③	0	-1	0	0	+1	+1	+1	0	0
④	0	0	-1	0	0	-1	0	+1	0
⑤	0	0	0	-1	0	0	-1	-1	-1

esta es la
columna co-
rrespondiente
a b_j

Figura 5.5

El problema dual de la red anterior es:

Maximizar

$$\pi_1 + 0\pi_2 + 0\pi_3 + 0\pi_4 - \pi_5$$

Sujeto a

$$\begin{array}{rcll}
 \pi_1 - \pi_2 & & & \leq 8 \\
 \pi_1 & & -\pi_3 & \leq 8 \\
 \pi_1 & & & -\pi_4 & \leq 6 \\
 & \pi_2 & & & -\pi_5 & \leq 9 \\
 & -\pi_2 & +\pi_3 & & & \leq 4 \\
 & & \pi_3 & -\pi_4 & & \leq 10 \\
 & & \pi_3 & & -\pi_5 & \leq 7 \\
 & & & \pi_4 & -\pi_5 & \leq 4
 \end{array}$$

Y su respectiva matriz de incidencia es:

	①	②	③	④	⑤
$f_{1,2}$	+1	-1	0	0	0
$f_{1,3}$	+1	0	-1	0	0
$f_{1,4}$	+1	0	0	-1	0
$f_{2,5}$	0	+1	0	0	-1
$f_{3,2}$	0	-1	+1	0	0
$f_{3,4}$	0	0	+1	-1	0
$f_{3,5}$	0	0	+1	0	-1
$f_{4,5}$	0	0	0	+1	-1
⑤	+1	0	0	0	-1

esta es la
fila corres-
pondiente a
 b_j

Figura 5.6

CAPITULO 6

6.0 EL METODO SIMPLEX APLICADO A LA RESOLUCION DE REDES DE PROBLEMAS DE RUTA MINIMA

6.1 Introducción

Una de las áreas activas de mayor interés, en la actualidad, es la solución básica del método simplex aplicado a la resolución de redes.

Los primeros precursores de este trabajo fueron Dantzig (1951), y Charnes y Cooper (1954), que aplicaron la técnica del método simplex al problema del transporte.

Orden (1956) extendió estos resultados al problema del transbordo. Recientemente, varios autores han utilizado la representación especial de bases de árboles, para incrementar notablemente la eficiencia computacional de las técnicas del método simplex.

A la par del método simplex han surgido otras técnicas, así Srinivasan y Thompson (1972, 1973); Glover, Karney y Klingman (1972) utilizan la representación de la triple etiqueta (triple label representation) para problemas de transporte capacitado.

Glover, Karney y Klingman (1974 a) aplican la técnica de la triple etiqueta a problemas de redes puras.

Glover, Klingman y Stutz (1974) introdujeron el método del índice de hilos aumentado para el problema de redes puras.

Bradley, Brown y Graves (1977), suministran los detalles para un algoritmo primal usando como base esta representación.

Finalmente, podemos decir que estudios computacionales de investigadores tales como Srinivasan y Thompson (1973); Glover, Karney y Klingman (1974); Glover, Karney, Klingman y Napier (1974) dan testimonio de la superioridad del método primal simplex para problemas de programas de redes de flujo.

6.2 Algoritmo primal simplex

Este algoritmo usa una base de árbol para implementar un procedimiento análogo al del algoritmo primal del simplex, del problema general de la programación lineal.

Recientes estudios indican que esta es, quizás, la forma más eficiente para resolver problemas de ruta mínima.

Los pasos generales del método simplex, para la resolución del problema de la ruta mínima, son como siguen:

6.2.1 Hallar una solución inicial básica factible. Ya sea a través de una matriz en forma canónica (tabla condensada) o a través de la inclusión de

variables artificiales (tabla extendida); pero esto último no se puede hacer en este problema específico.

Esta matriz en forma canónica la obtenemos a través de la matriz de incidencia:

6.2.2 Calcular $Z_{i,j} - C_{i,j}$ para cada arco no básico $a_{r,s}$.

$$\text{Siendo } Z_{i,j} = \sum_{\substack{(i,i) \\ (r,s)} \in A} a_{r,s} C_{i,j}$$

6.2.3 La optimalidad se alcanza cuando todos los valores $Z_{i,j} - C_{i,j}$ son todos positivos. De otra forma, seleccionamos el siguiente vector entrante o pivote.

En los campos del análisis y diseño de ingeniería, existen muchos problemas que pueden formularse como modelos de ruta mínima; entre ellos tenemos el problema del reemplazo de equipo, el problema de la planeación de proyectos, que involucra la investigación y desarrollo de proyectos, lanzamiento de misiles, y proyectos de administración de construcciones.

Para el caso de la administración de construcciones, a la que aplicaremos la maquinaria del simplex, cada arco recibe el nombre de actividad y cada nodo representa un evento.

En una red de actividades dirigidas, la longitud del arco está representada por la duración o tiempo que cada actividad necesita para completar un evento.

Así, la cadena más larga desde el nodo fuente hasta el nodo terminal, representa la duración mínima del proyecto, y también se le llama la ruta crítica, ya que el retraso de cualquier tarea sobre esa ruta, provocará un atraso en la completación del proyecto.

Esto forma parte de la base de los métodos PERT y CPM, tan utilizados y difundidos en nuestro medio.

Para la aplicación del método simplex, específicamente el problema primal, debemos de hacer algunas avariaciones en la tabla de cálculos, ya que ahora lo vamos a aplicar a redes dirigidas acíclicas. Para hacerlo, necesitamos una solución inicial básica factible que obtenemos a partir de nuestra matriz de incidencia. Teniendo nuestra matriz de incidencia en forma canónica, la tabla simplex queda así:

		Duración de cada actividad				$C_{i,j}$	Base
		$C_{i,j}$	r	s	t		
	$a_{r,s}$	$a_{1,1}$	$a_{1,2}$	$a_{i,j}$	$a_{u,v}$	$C_{i,j}$	
nodos	①	1	0	0	d_1	r	$a_{1,1}$
	②	0	1	0	d_2	s	$a_{1,2}$
	③	0	0	1	d_m	t	$C_{i,j}$
Z_{ij}	$-C_{i,j}$	0	0	0	d_q		

Tabla 6.1

Como podrá observar el lector, en nuestra tabla simplex no aparece la columna b_i correspondiente al vector solución, ya que, en vez de ella, utilizamos la columna de los $C_{i,j}$ correspondiente, que nos determinará la ruta mínima.

Como nuestro problema es de minimización, el cálculo de la columna de entrada es exactamente igual al simplex corriente, o sea el $Z_{i,j} - C_{i,j} < 0$, (el más negativo).

Para la fila de entrada, utilizamos el $C_{i,j}/a_{r,s}$; $a_{r,s} > 0$; que sea el mínimo positivo.

A continuación, aplicaciones del método simplex a redes de ruta mínima.

6.3 Redes de ruta mínima con actividades redundantes

Las "actividades redundantes" son aquellas que no forman parte de la ruta mínima, pero que aparecen en la solución óptima factible.

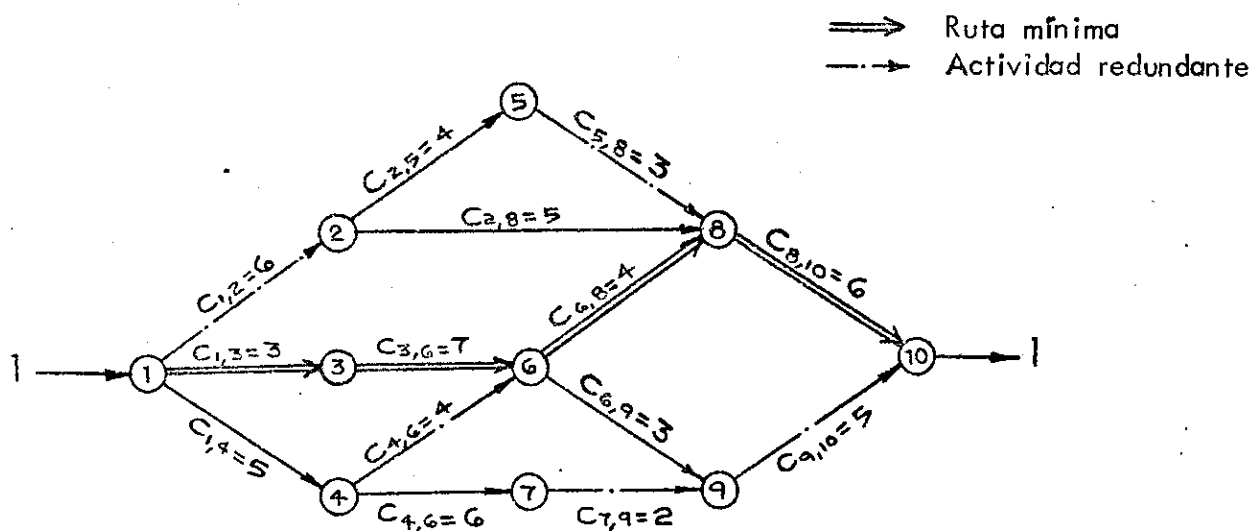


Figura 6.1

	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{2,8}$	$a_{3,6}$	$a_{4,6}$	$a_{4,7}$	$a_{5,8}$	$a_{6,8}$	$a_{6,9}$	$a_{7,9}$	$a_{8,10}$	$a_{9,10}$
①	1	1	1	0	0	0	0	0	0	0	0	0	0	0
②	-1	0	0	1	1	0	0	0	0	0	0	0	0	0
③	0	-1	0	0	0	1	0	0	0	0	0	0	0	0
④	0	0	-1	0	0	0	1	1	0	0	0	0	0	0
⑤	0	0	0	-1	0	0	0	0	1	0	0	0	0	0
⑥	0	0	0	0	0	-1	-1	0	0	1	1	0	0	0
⑦	0	0	0	0	0	0	0	-1	0	0	0	1	0	0
⑧	0	0	0	0	-1	0	0	0	-1	-1	0	0	1	0
⑨	0	0	0	0	0	0	0	0	0	0	-1	-1	0	1
⑩	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1

Tabla 6.2

Tabla inicial: Matriz de incidencia

Columna "no útil" ya que $Z_{ij} - C_{ij} \geq 0$

Matriz de incidencia en forma canónica y aplicación del método simplex

$C_{i,j}$	6	3	5	4	5	7	4	6	3	4	3	2	6	5	Base	
$a_{r,s}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{2,8}$	$a_{3,6}$	$a_{4,6}$	$a_{4,7}$	$a_{5,8}$	$a_{6,8}$	$a_{6,9}$	$a_{7,9}$	$a_{8,10}$	$a_{9,10}$	$C_{i,j}$	
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	6	$a_{1,2}$
2	0	1	1	1	1	0	0	0	0	0	0	0	0	0	5	$a_{2,8}$
3	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	7	$a_{3,6}$
4	0	0	-1	0	0	0	1	1	0	0	0	0	0	0	4	$a_{4,6}$ II
5	0	0	0	-1	0	0	0	0	1	0	0	0	0	0	3	$a_{5,8}$
6	0	-1	-1	0	0	0	0	1	0	1	0	0	0	0	3	Fila de entrada $a_{6,9}$
7	0	0	0	0	0	0	0	-1	0	0	0	1	0	0	2	$a_{7,9}$ $a_{6,9}/a_{6,8}$
8	0	1	1	1	0	0	0	0	0	-1	0	0	1	0	6	$a_{8,10} = 3$ Mínimo
9	0	-1	-1	0	0	0	0	0	0	1	0	0	0	1	5	$a_{9,10}$ positivo
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-

Tabla 6.3

$Z_{r,s} - C_{r,s} = \sum a_{r,s} * C_{ij} - C_{r,s}$ columna entrante $\Rightarrow Z_{ij} - C_{ij} < 0$ (más negativo)

$Z_{1,4} - C_{1,4} = [1 \times 6 + 1 \times 5 + 0 \times 7 - 1 \times 4 + 0 \times 3 - 1 \times 3 + 0 \times 2 + 1 \times 6 - 1 \times 5] - 5 = 0$

$Z_{1,3} - C_{1,3} = [1 \times 6 + 1 \times 5 - 1 \times 7 + 0 \times 4 + 0 \times 3 - 1 \times 3 + 0 \times 2 + 1 \times 6 - 1 \times 5] - 3 = -1$

$Z_{2,5} - C_{2,5} = [0 \times 6 + 1 \times 5 + 0 \times 7 + 0 \times 4 - 1 \times 3 + 0 \times 3 + 0 \times 2 + 1 \times 6 + 0 \times 5 + 0 \times 0] - 4 = 4$

$Z_{6,8} - C_{6,8} = [0 \times 6 + 0 \times 5 + 0 \times 7 + 0 \times 4 + 0 \times 3 + 1 \times 3 + 0 \times 2 - 1 \times 6 + 1 \times 5] - 4 = -2$

Tabla 6.4

$C_{i,j}$	6	3	4	5	7	4	6	3	4	3	2	6	5	Base	
$a_{1,s}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{2,8}$	$a_{3,6}$	$a_{4,6}$	$a_{4,7}$	$a_{5,8}$	$a_{6,8}$	$a_{7,9}$	$a_{8,10}$	$a_{9,10}$	$C_{i,j}$	
①	1	1	0	0	0	0	0	0	0	0	0	0	0	6	$a_{1,2}$
②	0	①	1	1	0	0	0	0	0	0	0	0	0	5	$a_{2,8}$ Fila de entrada
③	0	-1	0	0	1	0	0	0	0	0	0	0	0	7	$a_{3,6}$ $a_{2,8}/a_{1,3}$
④	0	0	0	0	0	1	1	1	0	0	0	0	0	4	$a_{4,6}$ = 5 (Mínimo
⑤	0	0	-1	0	0	0	0	0	1	0	0	0	0	3	positivo)
⑥	0	-1	0	0	0	0	1	1	0	1	0	0	0	4	$a_{6,8}$ III
⑦	0	0	0	0	0	0	-1	-1	0	0	1	0	0	2	$a_{7,9}$
⑧	0	0	1	0	0	0	1	1	0	0	0	1	0	6	$a_{8,10}$
⑨	0	0	0	0	0	0	-1	-1	0	0	0	0	1	5	$a_{9,10}$
⑩	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
$Z_{i,j} - C_{i,j}$	0	③	4	0	0	0	1	1	0	0	0	0	0	2	

Columna entrante $Z_{i,j} - C_{i,j} \geq 0$

$C_{i,j}$	6	3	5	4	3	4	2	5	Base		
$a_{r,s}$	$a_{1,2}$	$a_{1,3}$	$a_{2,8}$	$a_{3,6}$	$a_{4,6}$	$a_{5,8}$	$a_{6,8}$	$a_{7,9}$	$a_{8,10}$	$a_{9,10}$	
①	1	0	0	0	0	0	0	0	0	6	$a_{1,2}$
②	0	①	1	0	0	0	0	0	0	3	$a_{1,3}$
③	0	0	1	①	1	0	0	0	0	7	$a_{3,6}$
④	0	0	0	0	1	0	0	0	0	4	$a_{4,6}$
⑤	0	0	0	0	0	1	0	0	0	3	$a_{5,8}$
⑥	0	0	1	0	0	0	①	0	0	4	$a_{6,8}$
⑦	0	0	0	0	0	0	0	0	①	2	$a_{7,9}$
⑧	0	0	0	0	0	0	0	0	0	5	$a_{9,10}$

$$Z_{i,j} - C_{i,j} \quad 9 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

Desde que todos los $Z_{i,j} - C_{i,j}$ son ≥ 0 , llegamos a una solución óptima factible. Siendo los vectores encefalogramados dentro de un cuadro los que determinan la ruta mínima y los demás constituyen actividades redundantes. El valor de la duración de la ruta es $3C_{1,3} + 7C_{3,6} + 4C_{6,8} + 6C_{8,10} = 20$.

Tabla 6.5

6.4 Redes de ruta mínima con "varias rutas mínimas"

	$a_{1,2}$	$a_{1,3}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{3,4}$	$a_{3,5}$	$a_{4,6}$	$a_{5,4}$	$a_{5,6}$
①	1	1	0	0	0	0	0	0	0	0
②	-1	0	1	1	1	0	0	0	0	0
③	0	-1	-1	0	0	1	1	0	0	0
④	0	0	0	-1	0	-1	0	1	-1	0
⑤	0	0	0	0	-1	0	-1	0	1	1
⑥	0	0	0	0	0	0	0	-1	0	-1

Tabla inicial: Matriz de incidencia

Tabla 6.6

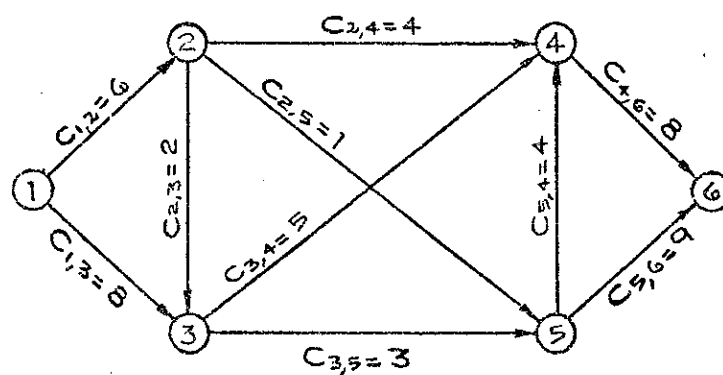


Figura 6.2

Columnas "no útiles": $Z_{ij} - C_{ij} \geq 0$

$C_{i,j}$	6	8	2	4	1	5	3	8	4	9	$C_{i,j}$	Base
$a_{r,s}$	$a_{1,2}$	$a_{1,3}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{3,4}$	$a_{3,5}$	$a_{4,6}$	$a_{5,4}$	$a_{5,6}$		
1	1	1	0	0	0	0	0	0	0	0	6	$a_{1,2}$
2	0	1	1	1	1	0	0	0	0	0	2	$a_{2,3}$
3	0	0	0	1	1	1	0	0	0	0	5	$a_{3,4}$
4	0	0	0	0	1	0	0	1	-1	0	8	$a_{4,6}$
5	0	0	0	0	-1	0	0	0	①	1	9	$a_{5,6}$ <small>Fila entrante</small>
6	0	0	0	0	0	0	1	0	0	0	3	$a_{3,5}$

$Z_{ij} - C_{ij}$ 0 0 3 5 0 0 0 -3 0

Indica existencia de otra ruta mínima

Columna de entrada

$$Z_{5,4} - C_{5,4} = -1 \times 8 + 1 \times 9 - 4 = -3$$

$$\text{Min } a_{5,6} / a_{5,4} = 9/1 = 9$$

Vector entrante

Nuevo vector pivote: $C_{5,4}$

Vector saliente: $C_{5,6}$

Matriz en forma canónica y aplicación del método simplex

Tabla 6.7

C_{ij}	6	2	5	3	8	4	9		
$a_{r,s}$	$a_{1,2}$	$a_{2,3}$	$a_{3,4}$	$a_{3,5}$	$a_{4,6}$	$a_{5,4}$	$a_{5,6}$	C_{ij}	Base
①	1	0	0	0	0	0	0	6	1,2
②	0	1	0	0	0	0	0	2	2,3
③	0	0	1	0	0	0	0	5	3,4
④	0	0	0	0	1	0	1	8	4,6
⑤	0	0	0	0	0	1	1	4	5,4
⑥	0	0	0	1	0	0	0	3	3,5
$Z_{ij}-C_{ij}$	0	0	0	0	0	0	3		

DESDE QUE TODOS LOS $(Z_{ij}-C_{ij}) \neq 0$, SON TODOS POSITIVOS, LLEGAMOS A UNA SOLUCIÓN ÓPTIMA FACTIBLE. LA DURACIÓN ES

$$6C_{1,2} + 2C_{2,3} + 3C_{3,5} + 8C_{4,6} + 4C_{5,4} = 23$$

TABLA 6.8

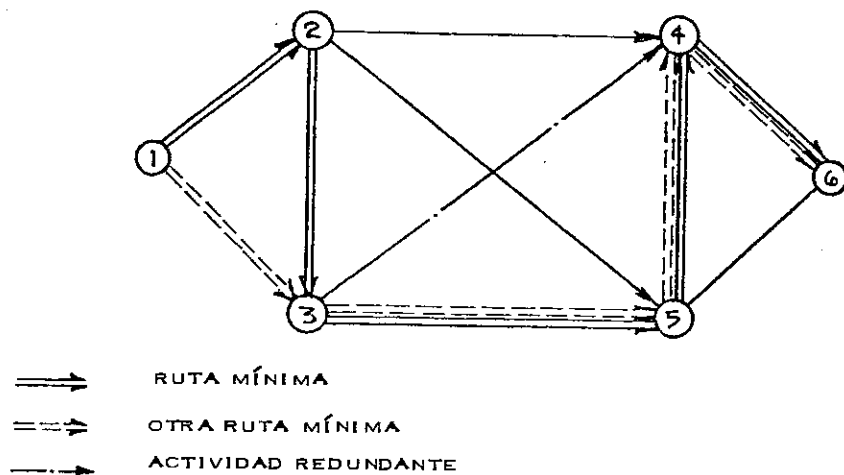


Figura 6.3

6.5 La presencia del ciclo en las redes de ruta mínima

Como ya habíamos visto con anterioridad, una de las complicaciones que ofrece la aplicación del método simplex es la "degeneración", que puede presentarse ya sea con actividades redundantes (en el caso de redes de ruta mínima acíclicas y sin circuitos y dirigidas) y/o a través del ciclo.

Como ya habíamos visto, el ciclo tiene la peculiaridad de que la tabla óptima es exactamente igual a la matriz de incidencia inicial en forma canónica.

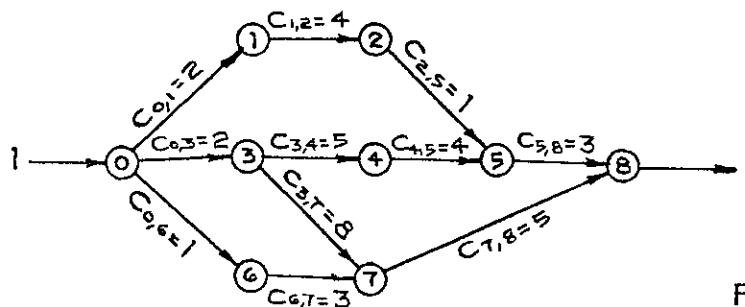


Figura 6.4

	$a_{0,1}$	$a_{0,3}$	$a_{0,6}$	$a_{1,2}$	$a_{2,5}$	$a_{3,4}$	$a_{3,7}$	$a_{4,5}$	$a_{5,8}$	$a_{6,7}$	$a_{7,8}$
①	1	1	1	1	0	0	0	0	0	0	0
②	1	0	0	1	0	0	0	0	0	0	0
③	0	0	0	1	1	0	0	0	0	0	0
④	0	1	0	0	0	1	1	0	0	0	0
⑤	0	0	0	0	0	1	0	1	0	0	0
⑥	0	0	0	0	1	0	0	1	1	0	0
⑦	0	0	1	0	0	0	0	0	0	1	0
⑧	0	0	0	0	0	0	1	0	0	1	1
⑨	0	0	0	0	0	0	0	0	1	0	1

TABLA INICIAL: MATRIZ DE INCIDENCIA ARCO-NODO.

TABLA 6.9

MATRIZ DE INCIDENCIA EN FORMA CANONICA Y APLICACION DEL SIMPLEX

COLUMNA NO UTIL, YA QUE $Z_{1j} - C_{1j} \geq 0$

C_{1j}	2	2	1	4	1	5	8	4	3	3	5	C_{1j}	BASE
$C_{r,s}$	$a_{0,1}$	$a_{0,3}$	$a_{0,4}$	$a_{1,2}$	$a_{2,5}$	$a_{3,4}$	$a_{3,7}$	$a_{4,5}$	$a_{5,8}$	$a_{6,7}$	$a_{7,8}$	C_{1j}	BASE
①	1	1	1	0	0	0	0	0	0	0	0	2	$a_{0,1}$
②	0	1	1	1	0	0	0	0	0	0	0	4	$a_{1,2}$
③	0	①	1	0	1	0	0	0	0	0	0	1	$a_{2,5}$
④	0	-1	0	0	0	1	1	0	0	0	0	8	$a_{3,7}$
⑤	0	0	0	0	0	-1	0	1	0	0	0	4	$a_{4,5}$
⑥	0	1	1	0	0	-1	0	0	1	0	0	3	$a_{5,8}$
⑦	0	0	-1	0	0	0	0	0	0	1	0	3	$a_{6,7}$
⑧	0	-1	-1	0	0	1	0	0	0	0	1	5	$a_{7,8}$
⑨	0	0	0	0	0	0	0	0	0	0	0	-	—
$Z_{1j} - C_{1j}$	0	⑤	1	0	0	1	0	0	0	0	0		

COLUMNA DE ENTRADA

TABLA 6.10

$$Z_{0,3} - C_{0,3} = \left[\sum C_{1j} \cdot a_{0,3} \right] - C_{0,3} = [1 \times 2 + 1 \times 4 + 1 \times 1 - 1 \times 8 + 0 \times 4 + 1 \times 3 + 0 \times 3 - 1 \times 5] - 2 = -5$$

$$Z_{0,6} - C_{0,6} = \left[\sum C_{1j} \cdot a_{0,6} \right] - C_{0,6} = [1 \times 2 + 1 \times 4 + 1 \times 1 + 0 \times 8 + 0 \times 4 + 1 \times 3 - 1 \times 3 - 1 \times 5] - 1 = 1$$

$$Z_{3,4} - C_{3,4} = \left[\sum C_{1j} \cdot a_{3,4} \right] - C_{3,4} = [0 \times 2 + 0 \times 4 + 0 \times 1 + 1 \times 8 - 1 \times 4 - 1 \times 3 + 0 \times 3 + 1 \times 5] - 5 = 1$$

VECTOR DE ENTRADA: $C_{0,3}$; VECTOR DE SALIDA: $C_{2,5}$

FILA DE ENTRADA: $a_{2,5}/a_{0,3} = 1$ (MÍNIMO POSITIVO.)

C_{1j}	2	2	4	1	8	4	3	3	5	C_{1j}	BASE
$C_{r,s}$	$a_{0,1}$	$a_{0,3}$	$a_{1,2}$	$a_{2,5}$	$a_{3,7}$	$a_{4,5}$	$a_{5,8}$	$a_{6,7}$	$a_{7,8}$	C_{1j}	BASE
①	1	0	0	-1	0	0	0	0	0	2	$a_{0,1}$
②	0	0	1	-1	0	0	0	0	0	4	$a_{1,2}$
③	0	1	0	①	0	0	0	0	0	2	$a_{0,3}$
④	0	0	0	0	1	0	0	0	0	8	$a_{3,7}$
⑤	0	0	0	0	0	1	0	0	0	4	$a_{4,5}$
⑥	0	0	0	-1	0	0	1	0	0	3	$a_{5,8}$
⑦	0	0	0	0	0	0	0	1	0	3	$a_{6,7}$
⑧	0	0	0	1	0	0	0	0	1	5	$a_{7,8}$
$Z_{1j} - C_{1j}$	0	0	0	③	0	0	0	0	0		

COLUMNA DE ENTRADA

TABLA 6.11

VECTOR DE SALIDA: $C_{0,3}$; VECTOR ENTRANTE: $C_{2,5}$

$C_{i,j}$	2	2	4	1	8	4	3	3	5		
$a_{r,s}$	$a_{0,1}$	$a_{0,3}$	$a_{1,2}$	$a_{2,5}$	$a_{3,7}$	$a_{4,5}$	$a_{5,8}$	$a_{6,7}$	$a_{7,8}$	$C_{i,j}$	BASE
①	1	1	0	0	0	0	0	0	0	2	$a_{0,1}$
②	0	1	1	0	0	0	0	0	0	4	$a_{1,2}$
③	0	1	0	1	0	0	0	0	0	1	$a_{2,5}$
④	0	0	0	0	1	0	0	0	0	8	$a_{3,7}$
⑤	0	0	0	0	0	1	0	0	0	4	$a_{4,5}$
⑥	0	1	0	0	0	0	1	0	0	3	$a_{5,8}$
⑦	0	0	0	0	0	0	0	1	0	3	$a_{6,7}$
⑧	0	-1	0	0	0	0	0	0	1	5	$a_{7,8}$
$Z_{i,j} - C_{i,j}$	0	3	0	0	0	0	0	0	0		

TABLA 6.12

¿Y llegamos a una solución óptima factible?; a simple vista diríamos que sí; ya que todos los $Z_{i,j} - C_{i,j} \neq 0$, son positivos. Pero si leemos, nuestra solución básica es exactamente igual a la de partida. He aquí la presencia del ciclo.

En la resolución de redes de ruta mínima, a diferencia de otros problemas de programación lineal en donde la solución inicial básica factible se repite cada 7 ciclos (como en el ejemplo de Beale), el ciclo es finito; es decir, al no más tener todos los $Z_{i,j} - C_{i,j} > 0$, llegamos al final del proceso iterativo.

6.6 Las "actividades ficticias" en las redes de ruta mínima

Antes de hablar de las actividades ficticias, hablaremos un poco de su razón de ser y diremos que éstas se utilizan para:

6.6.1 Mantener la lógica de la red y

6.6.2 para asegurarse de una identificación única de las actividades.

Además, es axiomático que las actividades ficticias no requieren tiempo para su completación.

En seguida, un ejemplo con actividad ficticia.

Algo muy curioso sobre las actividades ficticias, es que si les asignamos un $C_{i,j} = 1$, llegaremos al mismo resultado. Lo que es una evidencia de la eficiencia del poderoso método simplex.

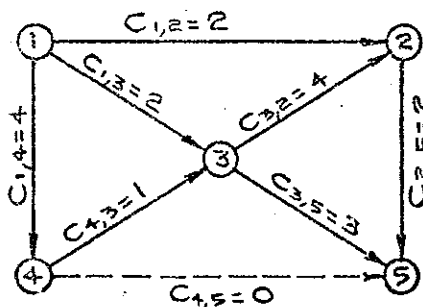


Figura 6.5

	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{3,2}$	$a_{3,5}$	$a_{4,3}$	$a_{4,5}$
①	1	1	1	0	0	0	0	0
②	-1	0	0	1	-1	0	0	0
③	0	-1	0	0	1	1	-1	0
④	0	0	-1	0	0	0	1	1
⑤	0	0	0	-1	0	-1	0	-1

TABLA INICIAL: MATRIZ DE INCIDENCIA

TABLA 6.13

$C_{1,j}$	2	2	4	2	4	3	1	0		
$a_{r,5}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{3,2}$	$a_{3,5}$	$a_{4,3}$	$a_{4,5}$	$C_{1,j}$	BASE
①	1	1	1	0	0	0	0	0	2	$a_{1,2}$
②	0	1	1	1	-1	0	0	0	2	$a_{2,5}$
③	0	-1	0	0	1	1	-1	0	3	$a_{3,5}$
④	0	0	-1	0	0	0	1	1	0	$a_{4,5}$
⑤	0	0	0	0	0	0	0	0	—	—
$Z_{1,j} - C_{1,j}$	0	-1	0	0	-3	0	-4	0		

ESTA NO ES UNA RUTA CRÍTICA SINO UNA ACTIVIDAD FICTICIA.

$$Z_{1,3} - C_{1,3} = [1x_2 + 1x_2 - 1x_3 + 0x_0] - 2 = -1$$

$$Z_{1,4} - C_{1,4} = [1x_2 + 1x_2 + 0x_3 - 1x_0] - 4 = 0$$

$$Z_{3,2} - C_{3,2} = [0x_2 - 1x_2 + 1x_3 + 0x_0] - 4 = -3$$

$$Z_{4,3} - C_{4,3} = [0x_2 + 0x_2 - 1x_3 + 1x_0] - 1 = -4$$

TABLA 6.14

Como podemos observar, la columna $C_{1,4}$ "no" la podemos anular en esta oportunidad, ni tampoco representa una ruta mínima diferente, sino que es producto de la presencia de una actividad ficticia. El reemplazo de la actividad ficticia no es ningún problema, ya que aunque hayan otros "unos positivos"; siempre la podremos reemplazar, al ser $C_{r,s}/a_{r,s} = 0$, si es que sea necesario.

$C_{1,j}$	2	2	4	2	4	3	1	0		
$a_{r,s}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{3,2}$	$a_{3,5}$	$a_{4,3}$	$a_{4,5}$	$C_{1,j}$	BASE
①	1	1	①	0	0	0	0	0	2	$a_{1,2}$
②	0	1	1	1	-1	0	0	0	2	$a_{2,5}$
③	0	-1	-1	0	1	1	0	1	3	$a_{3,5}$
④	0	0	-1	0	0	0	1	1	1	$a_{4,3}$
$Z_{1,j}-C_{1,j}$	0	-1	④	0	-3	0	0	4		

Tabla 6.15

Nótese que ahora la actividad ficticia está fuera de la solución básica factible, si podemos anular la columna de $C_{4,5}$ FINALMENTE.

$C_{1,j}$	2	2	4	2	4	3	1		
$a_{r,s}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{2,5}$	$a_{3,2}$	$a_{3,5}$	$a_{4,3}$	$C_{1,j}$	BASE
①	1	1	1	0	0	0	0	4	$a_{1,4}$
②	1	0	0	1	1	0	0	2	$a_{2,5}$
③	1	0	0	0	1	1	0	3	$a_{3,5}$
④	1	1	0	0	0	0	1	1	$a_{4,3}$
$Z_{1,j}-C_{1,j}$	4	3	0	0	4	0	0		

TABLA 6.16

$C_{1,j}$	4	2	4	3	1		
$a_{r,s}$	$a_{1,4}$	$a_{2,5}$	$a_{3,2}$	$a_{3,5}$	$a_{4,3}$	$C_{1,j}$	BASE
①	1	0	0	0	0	4	$a_{1,4}$
②	0	1	0	1	0	2	$a_{2,5}$
③	0	0	1	1	0	4	$a_{3,2}$
④	0	0	0	1	1	1	$a_{4,3}$
$Z_{1,j}-C_{1,j}$	0	0	0	4	0		

TABLA 6.17

TABLA FINAL OPTIMA: $4C_{1,4} + 2C_{2,5} + 4C_{3,2} + C_{4,3} = 11$

6.7 Problema ejemplo

Un contratista constructor debe construir un muro, pero ante la fluctuación de la moneda y el "dumping" imperante en el mercado, necesita saber el mínimo tiempo que puede tardarse para optimizar sus costos. La duración y secuencia de las actividades y eventos se describe a continuación.

Evento	Duración	Actividad	Precedencia
1	-	Inicio	-
2	1	Compra hierro #3 y #2; alambre amarre.	1
3	1	Compra madera; clavo, aceite quemado.	1
4	1	Compra blocks, materiales mortero.	1
5	1	Compra materiales concreto.	1
6	1	Hechura andamio.	3
7	3	Hechura estribos.	2
8	2	Armadura columnas y soleras.	7
9	1	Centrado y colocación columnas.	8
10	1	Preparar la mezcla.	4
11	3	1a. fase del levantado.	9, 10
12	1	Colocación armadura solera central.	11
13	2	Colocado formaleta 1a. fase.	6, 12
14	1	Fundición 1a. fase.	5, 13
15	1	Desencofrado 1a. fase.	14
16	1	Colocación andamio.	6, 15
17	4	Levantado 2a. fase.	16
18	1	Armadura solera final.	17
19	2	Formaleta 2a. fase.	18
20	1	Fundición 2a. fase.	19
21	1	Desencofrado 2a. fase y quitado del andamio. Finalización.	20

Tabla 6.18

Red del problema de ruta mínima, teniendo indicadas las actividades redundantes y la "ruta mínima".

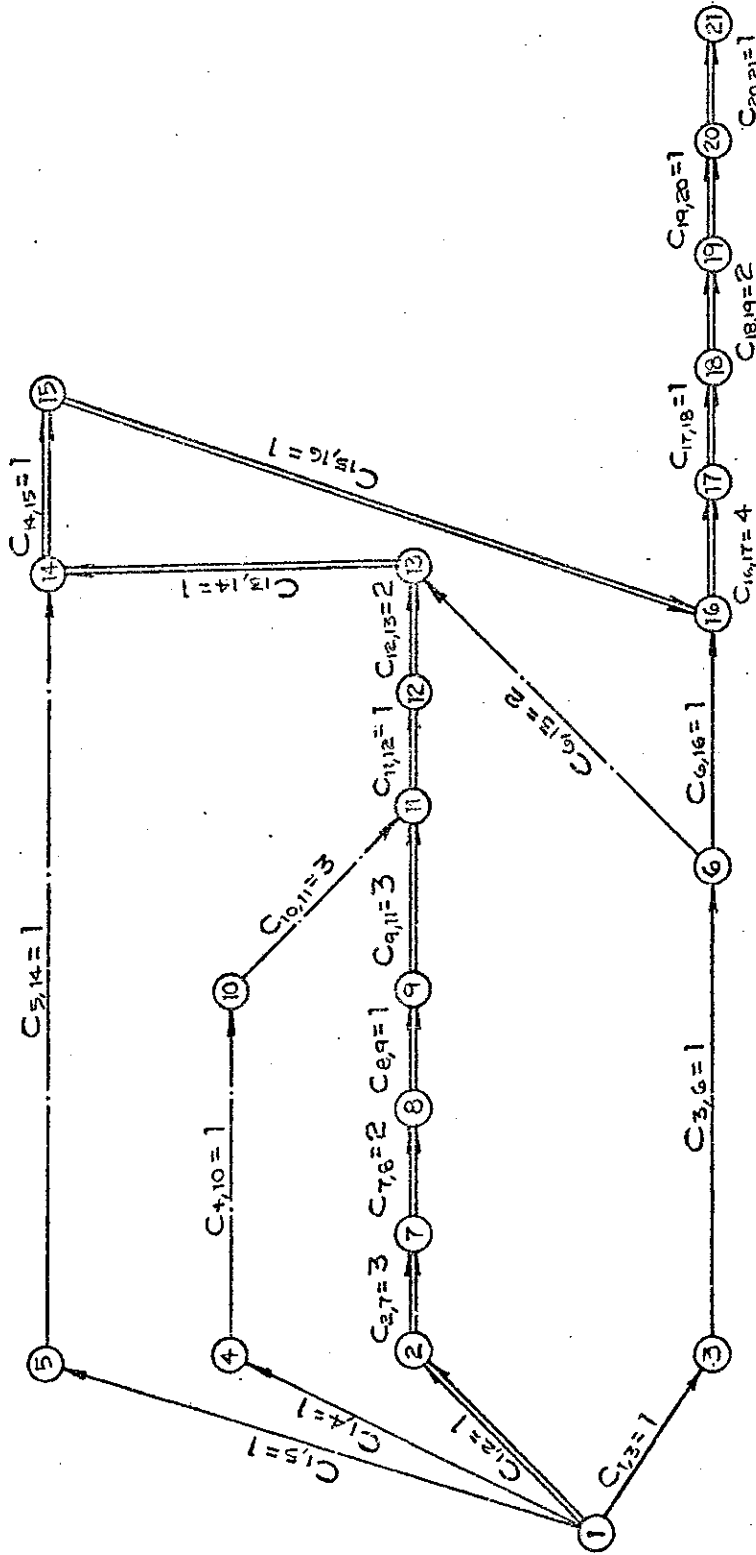


Figura 6.6

	$C_{1,2}$	$C_{1,3}$	$C_{1,4}$	$C_{1,5}$	$C_{1,7}$	$C_{2,5}$	$C_{2,7}$	$C_{3,5}$	$C_{3,7}$	$C_{4,5}$	$C_{4,7}$	$C_{5,10}$	$C_{5,14}$	$C_{5,13}$	$C_{6,3}$	$C_{6,11}$	$C_{7,8}$	$C_{8,3}$	$C_{8,11}$	$C_{9,12}$	$C_{10,11}$	$C_{11,12}$	$C_{12,13}$	$C_{13,14}$	$C_{14,15}$	$C_{15,16}$	$C_{16,17}$	$C_{17,18}$	$C_{18,17}$	$C_{19,10}$	$C_{19,21}$	$C_{20,16}$			
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
2	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
3	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
4	0	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
5	0	0	0	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	0	0	0	0	-1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLA INICIAL: MATRIZ DE INCIDENCIA.

TABLA 6.19

CAPITULO 7

7.0 HISTORIA SOBRE LOS CODIGOS DE COMPUTACION

Los métodos para resolver el problema de la programación lineal, se desarrollaron entre 1948 y 1952, fecha en la que se hicieron intentos de utilizar el equipo de computación, los cuales se vieron restringidas en alcance y éxito. De los años 1952 a 1956, se vieron grandes progresos y a partir de 1957, todas las técnicas de la programación lineal se desarrollaron a "pasos agigantados".

El crecimiento de las técnicas de la programación lineal y de la computación durante los últimos 36 años, han sido tan sorprendentes que es muy difícil recordar la situación prevaleciente en 1952.

Es dudoso saber si una matriz de orden 100, haya sido invertida alguna vez. Actualmente, un problema de 100 ecuaciones se considera pequeño. De tan drásticos cambios, resultan diferentes puntos de vista. Sin embargo, es útil recordar los crecimientos pasados.

La primera solución exitosa a un problema de programación lineal, utilizando una computadora electrónica de alta velocidad, ocurrió en enero de 1952, en la Oficina Nacional de Patrones para Máquinas —SEAC—, cuando Orchard-Hays, de la Corporación.RAND, adaptó el método simplex para el código de la IBM-CPC.

Posteriormente lo hizo para la IBM-701, en 1954 y para la IBM-704 en 1956, siendo este último código el más flexible para resolver problemas hasta de 200 ecuaciones y mil o más variables, en cerca de cinco horas, con bastante exactitud y eficiencia.

Décadas más tarde, se resolvía un sistema de 3,000 ecuaciones y 35,000 variables en tan sólo 51 segundos, utilizando una computadora CDC-6600.

Durante los años posteriores a la implementación del método simplex, se encararon cuatro principales obstáculos para poder resolver problemas de programación lineal:

- 7.0.1 Hallar una solución inicial básica factible.
- 7.0.2 Cuidarse de situaciones conocidas como "degeneración", en las que el proceso iterativo circulaba infinitamente sin converger.
- 7.03 Reducir tanto la aritmética como el almacenaje requerido para mantener el tiempo de solución, dentro de límites razonables.
- 7.04 Mantener suficiente precisión numérica, para obtener resultados significativos y, aún, para mantener el proceso iterativo libre de situaciones ambiguas.

Todos estos obstáculos se resolvieron, en gran medida, durante los siguientes tres años.

Es obvio que durante los primeros años, hallar una solución inicial básica factible empezando por "meros tanteos", era por sí mismo un problema de programación lineal, al igual que el hallar una solución óptima, a partir de una solución inicial básica factible.

En lo que respecta a la "degeneración", hubo mucha polémica a nivel profesional en los primeros días, utilizándose una variedad de complejos instrumentos para evitarla.

En la práctica no fue un problema tan serio de temer. Aunque la semidegeneración permanece como uno de los problemas de la programación lineal que conduce a largos tiempos de cálculo inesperadamente.

En lo referente al tiempo de solución y por razones difíciles de reconstruir, todos los matemáticos que estaban fuera del problema real, "lo hicieron más difícil de lo que era" o, dicho de otro modo, distorsionaron la explicación del método empleado y, de esta forma, las malas interpretaciones giraban en torno de los usuarios de la programación lineal.

El problema de la precisión se trató por la década del cincuenta, llegando a utilizarse la aritmética de doble precisión, que permite manejar números más grandes.

El costo-tiempo de computadora se alivió años más tarde con la incorporación de multiplicadoras y sumadoras rápidas, que hacían los cálculos en aritmética de punto flotante, una versión de computadora de notación científica.

7.1 Primeros logros hacia la programación lineal

Los primeros logros hacia la programación lineal, fueron:

7.1.1 La parametrización de los coeficientes de la función objetivo y

7.1.2 La parametrización del vector solución o columna de constantes b_j .

Esto se logró durante la segunda mitad de la década de los cincuenta.

Durante el año 1957, se hicieron tres grandes desarrollos:

7.1.3 La técnica de la inversión de matrices

7.1.4 La introducción del algoritmo dual

7.1.5 El manejo de variables implícitas (con ecuaciones explícitas)

Sin embargo, ninguno se usó durante años. Pero posteriormente aparecen como características estándar en sistemas de programación de computadora para programación lineal.

Mucho de los logros alcanzados durante 1963, no fueron más que técnicas algebraicas y algoritmos, así como consideraciones sobre los sistemas de computación.

Grandes avances se lograron en la formulación de problemas como modelos de programación lineal; sin embargo, la primera aplicación de la programación lineal fue al problema del transporte que, bajo suposiciones muy severas, suministraba valores enteros a las variables.

Desde el principio de la década del 60, aumentó el interés por las técnicas de "descomposición", pero muchos años atrás Orchard-Hays y Dantzig trabajaron sobre este aspecto. Por su parte, otros investigadores aplicaron el principio de la "partición de matrices", a problemas específicos, con gran éxito.

A pesar de todo, el algoritmo formal de Dantzig y Wolfe, estimuló el interés y la experimentación.

Los códigos de computación, por lo general, modificaciones de sistemas existentes, se han venido utilizando desde 1963, quizás, en forma limitada.

Además del método simplex, surgieron posteriormente otros métodos como el método de Dijkstra, el algoritmo de Floyd y el método del Doble Barrido, etc., todos orientados a resolver redes de problemas de ruta mínima.

Las diferencias entre algoritmos se miden en forma práctica como la necesidad o espacio para memoria y/o tiempo de ejecución y en forma teórica, con los parámetros que permiten la comparación de dos o más algoritmos, diseñados para resolver el mismo problema.

7.2 Formato de datos para programas de computadora

En los primeros códigos de computación para programación, los elementos a_{ij} de una matriz, se indicaban así: $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$. En las aplicaciones reales resultó muy difícil su manejo y a finales de los años cincuenta, los nombres simbólicos se utilizaron, finalmente, para indicar filas y columnas.

Los datos para un problema de programación lineal, con variables definidas, consisten en cinco clasificaciones:

7.2.1 Nombres lógicos o identificaciones

Identifican a las ecuaciones (filas).

7.2.2 Nombres estructurales

Se le denomina a las actividades (arcos); (columnas).

7.2.3 Los elementos de la matriz

A los valores numéricos a_{ij}

7.2.4 Conjuntos definidos para variables lógicas y estructurales.

7.2.5 El vector columna solución de las cantidades constantes.

Se identifica con los valores numéricos b_j .

La clasificación 7.2.2 generalmente no es explícita, pues impone una carga necesaria al usuario, con preparar una larga y separada lista de nombres de columnas.

En la clasificación 7.2.3 el orden de aparición de los nombres de columnas, junto con los tipos de filas, define la clasificación 7.2.4.

CAPITULO 8

8.0 CONCLUSIONES

8.1 En ejemplos como los anteriores no se puede apreciar la eficiencia y rapidez del método simplex que, en su aplicación a la resolución de redes de problemas de ruta mínima, utiliza las excelentes propiedades de la matriz de incidencia arco-nodo, para así obtener una solución inicial básica factible a través de la matriz incidente en forma canónica o tabla condensada, como paso previo a la aplicación de la maquinaria del simplex.

El hecho de que el método simplex de la programación lineal haya sido creado a finales de la década del 40, no implica que sea obsoleto o pasado de moda, como algunos suponen en la actualidad. Recientes experiencias computacionales de investigadores tales como Srinivasan y Thompson (1973), Glover, Karney y Klingman (1974); Glover, Karney, Klingman y Napier (1974), son testimonio de la superioridad del método simplex para la resolución de problemas primales de redes de flujo, sobre otros algoritmos conocidos.

Buffa-Elwood, presentan en su libro *Management/Science/Operations Research* una sinopsis de la eficiencia y economía, de resolver por computadora las redes, basados en cargos de computadora de \$.0.04/segundo.

Tiempo de resolución y costos				
Problema	Número de nodos	Número de arcos	Tiempo de resolución	Costo por la resolución
1	8 000	15 000	80.5 Seg.	\$.3.22
2	5 000	23 000	80 "	\$.3.20
3	3 000	35 000	63 "	\$.2.52
4	5 000	15 000	51 "	\$.2.04
5	3 000	23 000	37.5 "	\$.1.50

Tabla 8.1

Imagínese el lector que, mientras usted está resolviendo una matriz de 8×16 , solamente le tomaría un segundo el ver los tiempos de resolución y costos, para convencerse de la eficiencia computacional del algoritmo del simplex para la resolución de redes. Como cosa curiosa, Orchard-Hays de la Corporation Rand, desarrolló un programa para la computadora IBM 704 en 1956, que resolvía una matriz de $200 \times 1,000$, en poco más de cinco horas. Con mucha exactitud y eficiencia para su época.

8.2 En la aplicación del método simplex de la programación lineal, a la resolución de redes del problema de la ruta mínima, es inevitable la aparición de actividades redundantes en la tabla óptima, pero de todas maneras su presencia no afecta en la determinación de la ruta más larga o "ruta mínima", por ser actividades sueltas que no forman una ruta.

8.3 Las actividades ficticias con tiempo cero o actividades artificiales con tiempo unitario, no alteran en ningún sentido la aplicación del método simplex para determinar la ruta mínima; cosa que no sucede con el método de la ruta crítica (C.P.M.), donde una actividad artificial lo único que hace es demorar la completación total del proyecto.

8.4 Para la resolución de redes de problemas de ruta mínima, por medio del método simplex, es más ventajoso utilizar el problema primal del modelo de la programación lineal, debido a que la matriz de incidencia resultante no es más que una matriz de unimodularidad, cuyas columnas siempre tienen dos entradas $+1$ ó -1 . Lo cual es muy ventajoso para poder convertirla en forma canónica, cosa que no sucedería con el problema dual donde hay más de dos entradas por columna. Además, significa un gran ahorro de espacio en la memoria de la computadora.

8.5 En lo que respecta al problema de la degeneración por ciclo, no es un problema serio de temer en la resolución de redes de problemas de ruta mínima, como se aprecia en el problema ejemplo, debido a que las iteraciones son finitas y la solución inicial básica factible sólo se repite una vez.

Es decir, lo que apreciamos es un caso de semidegeneración.

8.6 En la determinación de una "ruta mínima", si fuera única, lo que estamos averiguando es el camino más largo o "ruta crítica", de tal manera que el retraso de cualquiera de las actividades que conforman la "ruta crítica", provocará una demora en la ejecución total del proyecto, traduciéndose en un mayor número de días para su completación.

Por lo expuesto anteriormente, bien podríamos llamarle a este tipo de redes de problemas de ruta mínima, "redes de problemas de ruta crítica", pero no lo hemos hecho así por la sencilla razón de que no queremos que se confunda con el método de la ruta crítica (CPM), tan utilizado y difundido en nuestro medio para la resolución de redes de ruta mínima.

8.7 Finalmente, en esta tesis hemos expuesto en forma meticulosa los fundamentos matemáticos tanto del método simplex de la programación lineal, como del análisis del flujo de redes y, asimismo, sus probables complicaciones al momento de su ejecución, sin llegar a pretender, siquiera, que los ingenieros o responsables de la administración de proyectos, traten de conocerlos; sino más bien, que sepan distinguir que una red de ruta mínima NO es un método para resolver redes y que la programación lineal NO es ningún lenguaje de computadora.

BIBLIOGRAFIA

- Antill, J. M. y Woodhead, R. W. Critical path methods in construction practice. Segunda edición. John Wiley and Sons. 425 Págs.
1982
- Bazaraa, Mokhtar S. Linear programming and network flows. John Wiley and Sons Inc.
1978
- Buffa, Elwood and Dyer, James S. Management science/operations research.
1981 John Wiley and Sons.
- Charnes, A. y Cooper, W. Management models and industrial applications of linear programming. Volúmenes I y II. John Wiley and Sons, Inc.
1961
- Charnes, A. y Cooper, W. A network interpretation and a directed subdual algorithm for critical path scheduling. Journal of industrial Engineering, Vol. 13, # 14, Págs. 213-218.
1962
- Dantzig, George B. Linear programming and extensions. Princeton University Press. Princeton, N. J. Quinta impresión. 627 Págs.
1972
- Ford, L. R. and Fulkerson, D. R. Flows in networks. Princeton University Press. Princeton, J. J.
1962
- Gass, Saul I. Linear programming: Methods and applications. Mc Graw-Hill Company, Inc. 2a. edición. 280 Págs.
1958
- Hadley, G. Linear programming. Addison-Wesley. Publishing Company, Inc.
1963
- Hohn, Franz E. Matrix algebra. The Mac Millan Company. New York. U.S.A. 2a. edición. 453 Págs.
1964
- Jensen, Paul and Barnes, J. Wesley. Network flow programming. John Wiley and Sons. 1a. edición. 408 Págs.
1980
- Kaufmann, A. Introduction to operations research. Mathematics in science and engineering. Vol. 47.
1968
- Kelly Jr., James E. Critical path planning and scheduling: Mathematical basis. Operations research. Vol. 9, # 3. May-june. U.S.A.
1961

- Lang, Serge. Linear algebra. 3a. edición. Addison-Wesley, Publishing Co.
1972
- Llewelyn, Robert W. Linear programming. Holt, Rinehart and Winston Inc.
1964
- Moder, Joseph and Phillips, Cecil. Project management with CPM and Pert.
1964 Reinhold Publishing Corporation.
- Orchard, William and Hays, . Advanced linear programming
and computing techniques. Mc Graw-Hill Co. Inc.
1968
- Phillips, Don T. and García-Díaz, Alberto. 1a. edición. Fundamentals of
network analysis. Prentice-Hall, Inc. 474 Págs.
1981
- Sasieny, M.; Yaspan, A.; Friedman, L. Operations research. John Wiley
1959 and Sons. Inc., N. Y. 316 Págs.
- Simmonard, M. Linear programming. Paraninfo, Madrid. 1a. edición.
1972 527 Págs.
- Smithe Jr., William and Johnson, Lindwood. Introduction to linear programming
with applications. Prentice-Hall Inc.
1966
- Thie, Paul R. An introduction to linear programming and game theory. John
1979 Wiley and Sons. 1a. edición. 335 Págs.
- Wolfe, Carvel S. Linear programming with fortran. Scott, Foresman and
1970 Cowaly. Glenview, Illinois.

APENDICE

La mayoría de problemas prácticos que pueden formularse como programas de programación lineal, involucran muchas variables que conducen a grandes tablas, de modo que el cálculo a mano llega a ser largo y muy complicado, caro y tedioso. Los algoritmos del método simplex son iterativos y conducen, naturalmente, a un programa de computadora.

Uno de los lenguajes ampliamente utilizados, dentro de los disponibles, es el FORTRAN. A pesar de que ha sufrido varias revisiones, el control principal de las instrucciones (statements) es el mismo en cada versión.

En este apéndice únicamente presentaremos subrutinas que llevan a cabo el pivoteaje. Estas subrutinas pueden utilizarse en Fortran II, Fortran IV o cualquier versión posterior de Fortran.

Las diferencias del lenguaje se manejan en el programa principal, que controla los datos de entrada, los llamados a las subrutinas necesitadas y la forma de la salida.

1.0 Subrutina para pivotar la Tabla Extendida

Continuando con la notación matricial utilizada previamente, indicaremos el tamaño de la tabla con variables enteras: M el número de fila, N el número de columnas. Nuestro pivote se halla ahora en la p -ésima fila y Q -ésima columna. Para que estas variables sean enteras, las designaremos IP e

IQ respectivamente. A la variable a_{ij} la denominaremos $Y(I, J)$ para la tabla de entrada en la I -ésima fila y J -ésima columna.

Si IP, IQ, M, N y la tabla Y son conocidas, la siguiente subrutina efectuará una iteración de pivoteaje.

```

SUBROUTINE PIVEX (IP, IQ, M, N, Y)
  DIMENSION Y (25, 25)
  A = Y (IP, IQ)
  DO 10 J = 1, N
    Y(IP,J) = Y (IP, J)/A
  DO 50 I = 1, M
    IF (I-IP) 30, 50, 30
  30  B=Y (I, IQ).
    DO 40 J = 1, N
  40  Y(I,J) = Y (I, J) -B * Y (IP, J)
  50  CONTINUE
  RETURN
END

```

El propósito de esta columna es producir un vector columna unitario en la columna que se está pivoteando. Los corchetes indican los "do loops", de forma que la secuencia lógica puede seguirse fácilmente. La declaración 10, deja un uno en la posición pivote cuando $J = IQ$.

La declaración 40 produce ceros en la columna pivotal, excepto en la fila pivote. La declaración IF se utilizó para saltarse a la fila pivotal del loop exterior.

El "Loop" interior ejecuta el proceso de eliminación Gauss-Jordan a través de la i -ésima fila, en tanto que el "Loop" exterior avanza el cálculo de fila en fila. Así, obtenemos un vector columna unitario en la columna pivotal.

PIVEX tiene varias aplicaciones como:

1.1 Diagonalizar una matriz, para resolver un conjunto de ecuaciones lineales; consistentes e independientes a fin de que no aparezca un cero en la diagonal. Hay que tener el cuidado de incluir siempre la columna de los términos constantes, ya que a la hora de obtener la matriz identidad, la solución para cada variable aparecerá en la columna de términos constantes.

1.2 Invertir una matriz no singular

Si la matriz es:
$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

La tabla inicial es así:
$$\begin{bmatrix} a_{11} & a_{12} & 1 & 0 \\ a_{21} & a_{22} & 0 & 1 \end{bmatrix}$$

Al diagonalizar la matriz principal, la matriz identidad se convierte en la inversa de la principal y la matriz principal se reduce a la identidad.

De paso, PIVEX puede utilizarse para pivotar la tabla extendida de un programa lineal para producir una nueva solución básica factible. Sin embargo, la mayoría de problemas de programación lineal se resuelven por pivotar la tabla condensada.

2.0 Subrutina para pivotar la Tabla Condensada

Para poder pivotar la tabla condensada, introducimos dos nuevas variables, $K(J)$, $J = 1 \text{ a } N-1$ y $L(I)$, $I = 1 \text{ a } M-1$.

$K(J)$ identifica a los vectores que están fuera de la base y que se indican arriba de nuestra tabla.

$L(I)$ identifica a los vectores que pertenecen a la base, y que aparecen al lado izquierdo de nuestra tabla.

Los valores iniciales de $K(J)$ y $L(I)$ son leídos por el programa principal juntamente con la tabla inicial.

Luego que se determinan las posiciones de pivote P e IQ , la siguiente subrutina efectuará los cálculos.

```

SUBROUTINE PIVCO (Y, K, L, M, N, IP, IQ)
  DIMENSION Y (30, 30), K (30), L (30)

  R = 1./Y (IP, IQ)

  DO 10 J = 1, N
2a. etapa 10 Y (IP, J) = R * Y (IP, J)

1a. etapa Y (IP, IQ) = R

  DO 50 I = 1, M
    IF (I -IP) 20, 50, 20
      DO 40 J = 1, N
        IF (J -IQ) 30, 40, 30
4a. etapa 30 Y (I, J) = Y (I, J) -Y (I, IQ) * Y (IP, J)
        40 CONTINUE
3a. etapa Y (I, IQ) = R * Y (I, IQ)
      50 CONTINUE

    X = K (IQ)

5a. etapa K (IQ) = L (IP)

    L (IP) = X

    RETURN

  END

```

La primera operación después de la declaración "DIMENSION" almacena en el registro R, el recíproco del pivote, para su futuro uso.

El paso 10 del primer "Loop" corresponde a la segunda etapa de nuestro algoritmo y el paso después de 10 identifica el pivote y corresponde a la 1a. etapa.

La declaración 30 efectúa la cuarta etapa, después que las dos declaraciones "IF" saltean la fila y la columna pivotal. El "Loop" interior calcula las entradas en la i -ésima fila. El "Loop" exterior moviliza el cálculo de fila en fila a través de la tabla. La declaración entre 40 y 50, corresponde a la tercera etapa. Pertenece al "Loop" externo y calcula la entrada en la columna pivotal, para cada fila, excepto la columna pivotal. Finalmente, la quinta etapa la ejecutan las tres últimas declaraciones antes de retornar al programa principal.

Nótese que la cifra entera K (IQ) se almacena en el registro X temporalmente, para no ser borrada cuando se reemplace por el valor entero L (IP).

Las etapas del algoritmo para la Tabla Condensada son:

- 2.01 Reemplace el pivote por su recíproco.
- 2.02 Divida (Multiplique) el resto de entradas de la fila pivotal por el pivote.
- 2.03 Divida (multiplique) el resto de entradas de la columna pivotal, por el negativo del pivote.

2.04 Para las entradas de las posiciones restantes, sustraiga de la posición correspondiente en la vieja tabla, el producto del valor antiguo, tanto en la fila como en la columna pivotal, las veces que la nueva entrada aparecerá en esa columna y fila pivotal.

2.05 Intercambie el vector no básico arriba de la columna pivotal, por el vector básico a la izquierda de la fila pivotal.

3.0 Subrutina de pivoteado automático

El objetivo de esta subrutina es que la computadora halle la posición pivote y que, automáticamente, continúe pivoteando hasta que la tabla final sea alcanzada.

El programa principal necesita solamente imprimir la tabla final al finalizar un problema. Tres subrutinas adicionales se utilizan, para lograr el proceso automático.

La primera investiga las columnas y determina cual de ellas está disponible. Si no existe columna pivotal posible, la subrutina regresa al programa principal con la declaración $IQ = -1$.

```

SUBROUTINE COL (Y, K, M, N, IQ)
DIMENSION Y (30, 30), K (30)

IQ = -1

B = 0.

NN = N -1

DO 30 J = 1, NN
  IF (K (J)) 30, 10, 10
10 IF (Y (M, J)) 20, 30, 30
20 IF (Y (M, J) -B) 25, 30, 30
25 B = Y (M, J)

  IQ = J

30 CONTINUE

RETURN

END

```

Nótese que el primer "IF" previene que cualquier variable artificial regrese a la base (cuando es tabla extendida). Si $K(J)$ es negativa, se saltea esa columna y nunca se usa de nuevo al escoger un pivote.

La declaración 10 escoge los números negativos de la última fila. Las declaraciones 20 y 25, nos aseguran que hemos escogido el número negativo, de valor absoluto, más grande, antes de igualar a IQ con el número de esa columna.

Así, el valor de IQ retorna al programa principal ya sea para representar a la columna pivotal o al hecho de que no hay disponibilidad. Si IQ permanece negativo, el programa principal imprimirá los valores corrientes de $Y(I,J)$ como la tabla final.

La segunda subrutina investiga las filas y escoge una fila pivotal con la declaración $IP = -1$.

```
SUBROUTINE ROW (Y, M, N, IP, IQ)
```

```
  DIMENSION Y (30, 30)
```

```
  IP = -1
```

```
  S = .1 E 19
```

```
  MM = M - 1
```

```
  DO 80 I = 1, MM
```

```
    IF (Y (I, IQ)) 80, 80, 10
```

```
    10 R = Y (I, N)/Y (I, Q)
```

```
    IF (S -R) 80, 80, 20
```

```
    20 S = R
```

```
    IP = I
```

```
  80 CONTINUE
```

```
  RETURN
```

```
  END
```

El primer "IF" se saltea cualquier entrada negativa o de valor cero de la columna pivotal, para considerar sólo relaciones de Θ 'S positivos, en la declaración IO.

El segundo "IF" nos asegura que hemos escogido el más pequeño de los Θ 's, antes de igualar a IP con esa fila. Si existe paridad por el menor, nuestro programa escoge a la primera de las filas halladas.

Si ROW y COL retoman valores positivos al programa principal, el siguiente comando en el programa principal será:

CALL PIVCO (Y, K, L, M, N, IP, IQ)

que ejecutará el pivotaje.

Si ROW o COL retoman valores negativos, la tabla final se ha alcanzado y se imprimirá.

La tercera subrutina es la que distingue entre una fase y otra de pivoteo. El programa principal debe saber si alguna variable artificial permanece en la base. Si es así, llamará al investigador de columnas en la fila M. Esta es la fase uno, donde los números negativos son buscados en la última fila. Tan pronto como la última variable artificial deja la base, el programa principal debe llamar al investigador de columnas en la fila M -1.

Esta es la fase dos donde los números negativos son buscados en la primera de las dos filas objetivos.

Si el problema original no tiene variables artificiales, entonces se usa sólo una fila objetivo.

Para permitir estas tres posibles opciones, la subrutina "Look" es llamada, antes de cada investigador de columnas, hasta que las variables artificiales son removidas.

```

SOUBROUTINE LOOK (L, M, V)
  DIMENSION L (30)
  DO 10 I = 1, M
    IF (L (I)) 30, 10, 10
  10 CONTINUE
  V = 1.
  RETURN
30 V = -1.
  RETURN
END

```

Por comodidad, todas las filas originales de la tabla, se numeran a la izquierda, de modo que $L(I)$ se define de 1 a M . A las filas objetivos, se les puede asignar cualquier valor positivo. La subrutina "Look" vuelve al programa principal con la declaración $V = -1$. Si algún $L(I)$ es negativo y

regresa con $V = 1$. Si todos los $L(I)$ son positivos. En cualquier etapa, el programa principal puede saber si hay variables artificiales en la base con sólo llamar a "Look" y examinar el valor de V .

Esto completa las subrutinas necesarias para ejecutar un programa de pivoteado automático.

Debe tenerse mucho cuidado en el programa principal al utilizar el investigador de columnas, solamente para las primeras $M-2$ filas en problemas con una doble fila objetivo ya que nunca pivoteamos sobre las filas objetivo. Antes de alcanzar la tabla final, el programa imprimirá los $K(J)$ y $L(I)$ en sus respectivas posiciones, junto con los $Y(I, J)$ para una completa interpretación de los resultados. Finalmente, es bueno colocar un contador que nos indique las veces que la tabla original se pivoteó.