

UNIVERSIDAD AUTONOMA DE QUERÉTARO
FACULTAD DE INGENIERÍA

**Estudio del Problema de Flujo de Bienes
Múltiples a Costo Mínimo**

TESIS

Que para obtener el grado de:

LICENCIADO EN MATEMÁTICAS APLICADAS

Presenta:

CARLA ALEJANDRA RIVERA RAMÍREZ

Santiago de Querétaro, Qro., Abril 1997

No Adq. 456261

No. Título _____

Clas. 519.72

R 62/1e

LIBRARY OF THE
CONGRESS

LIBRARY OF THE
CONGRESS

LIBRARY OF THE
CONGRESS

LIBRARY OF THE
CONGRESS

LIBRARY OF THE
CONGRESS

LIBRARY OF THE
CONGRESS

LIBRARY OF THE
CONGRESS



ACUERDO/36460/97
ENERO/23/97

C. CARLA ALEJANDRA RIVERA RAMIREZ,
P R E S E N T E .

En relación a su atenta solicitud relativa a la selección de opción de Tesis consistente en la elaboración de Tesis con el Título -- "ESTUDIO DEL PROBLEMA DE FLUJO DE BIENES MÚLTIPLES A COSTO MÍNIMO", - me permito informarle que fue aceptado el Tema y la DRA. REBECA DEL - ROCIO PENICHE VERA tuvo a bien ser Director del Comité de Titulación.

El contenido aceptado por el H. Consejo Académico es el siguiente:

CAPITULO I PROBLEMA DE FLUJO EN REDES

- 1.1 Flujo Máximo
 - 1.1.1 Algoritmo de Ford y Fulkerson
 - 1.1.1.1 Ejemplo
- 1.2 Flujo a costo mínimo
 - 1.2.1 Algoritmo de Klein
 - 1.2.1.1 Ejemplo
 - 1.2.2 Algoritmo de Busaker y Gowen
 - 1.2.2.1 Ejemplo
 - 1.2.3 Simplex especializado en redes
 - 1.2.3.1 Ejemplo

CAPITULO II Planteamiento del problema de flujo de bienes múltiples a costo mínimo.

- II.1 Formulación nodo-arco
- II.2 Formulación arco-cadena

CAPITULO III

Un método de resolución basado en el método de Dantzing Wolfe.

- III.1 Formulación nodo-arco
 - III.1.1 Ejemplo
- III.2 Formulación nodo-arco
 - III.2.1 Ejemplo

CAPITULO IV Planteamiento y resolución de dos casos particulares

- IV.1 Problema de transporte de bienes múltiples
 - IV.1.1 Ejemplo
- IV.2 Redes con determinada propiedad topológica y su transformación al problema de flujo de un solo bien
 - IV.2.1 Ejemplo
 - IV.2.2 Un caso particular
 - IV.2.2.1 Ejemplo



Universidad Autónoma de Querétaro
Facultad de Ingeniería



CAPITULO V CARACTERISTIZACION DE UNA CONDICION BAJO LA CUAL EXISTEN SOLUCIONES ENTERAS PARA EL PROBLEMA EN GENERAL.

- V.1. Preliminares
- V.2. Desarrollo de Commoner
- V.3. Presentación de la operación binaria
- V.4. Caracterización algebraica

CONCLUSIONES

Apéndice A CONCEPTOS BASICOS

- A.1 Grupos
- A.2 Teoría de gráficas
- A.3 Ruta más corta

Apéndice B EL METODO DE DANZING-WOLFE

Apéndice C EL PROBLEMA DE TRANSPORTE

- C.1 Presentación del Problema
- C.2 Resolución
- C.3 Ejemplo.



Universidad Autónoma de Querétaro
Facultad de Ingeniería



BIBLIOGRAFIA:

1. M.S. Bazaraa y J.J. Jarvis. "Programación Lineal y Flujo en Redes" Limusa, 1981.
2. J.R. Evans. A Combinatorial Equivalence Between a Class of Multicommodity Flow Problems and the Capacitated Transportation Problem. Math. Programming 10. (1976) 401, 404.
3. J.R. Evans. A single-Commodity Transformation for Certain Multicommodity Network Flow Problem. Opns. Res. 26. (1978) 673-680
4. J.R. Evans & J.J. Jarvis & Network Topology and Integral Multicommodity Flow Problem. network 8 (1978) 107-119.
5. J.R. Evans J.J. Jarvis & R. Duke. Graphic Matroids and the Multicommodity Transportation Problem. Math Programming 13 (1977) 323-328.
6. L. Ford & D. Fulkerson "Flows in Networks" Princeton University -- Press, 1962.
7. L. Ford & D. Fulkerson. A Suggested Computation for maximal Multicommodity Network Flows Management Sci. 5. (1958) 97-101.
8. M.D. Grigoriadis & W. White. A Partitioning Algorithm for Multicommodity Network Flows Problem. Math. Programming 3. (1972) 157-177
9. R. Grinold. A Multicommodity MAX_Flow Algorithm. Opns. Res. 16 (1968) 1234-1238.
10. G. Hadley. "Linear Programming" Addison Wesley Publishing Co. INC., 1963.
11. J. Hartman & L. Lasdon. A Generalized Upper Bounding Algorithm -- for Multicommodity Network Flow Problems. Network Programming -- Wiley-Inter Science Publication., 1980.
12. J. Kennington. A Survey of Linear Cost Multicommodity Network -- Flows Opns. Res. 2. (1978) 209-236.
13. J. Kennington & R. Helgason. "Algorithms for Network Programming". Wiley-Inter Science Publication., 1980.
14. D. Kleitman. An Algorithm for Certain Multi-Commodity Flow Problems. Network 1 (1971) 75-90.
15. J. Prawda. "Métodos y Modelos de Investigación de Operaciones". Limusa I., 1979.
16. W. Rothfarb & I.T. Frisch. Common Terminal Multicommodity Flow. Opns. Res. 16. (1968) 202-205
17. W Rothfarb & I.T. Frisch. On the 3-Commodity Flow Problem. SIAM J. Appl. Math. 17 (1968) 46-58.
18. B. Rothschild & A. Whinston. On two Commodity Network Flows. Opns. Res. 14. (1966) 377-387.
19. B. Rothschild, A. Whinston. Feasibility on Two Commodity Network Flows. Opns Res. 14. (1966) 1125-1129
20. B. Rothschild, A. Whinston & J. Kent. Computing two-Commodity Flows Opns. Res. 16. (1968) 446-450
21. M. Sakarovitch. Two Commodity Network Flows. And Linear Programming Math. Programming 4. (1973) 1-20



Universidad Autónoma de Querétaro
Facultad de Ingeniería



22. H. Soroush & P.B. Mirchadani. The Stochastic Multicommodity Flow Problem. Networks 20. (1990) 121-155
23. D. Tang Comments on Feasibility Conditions of Simultaneous Flows in a Network. Opns. Res. 13 (1965) 143-146.
24. J.A. Tomlin Minimum-cost Multicommodity Networks Flows, Opns. Res. 14. (1966) 45-51

También hago de su conocimiento las disposiciones de nuestra Facultad, en el sentido de que antes de su Examen Profesional deberá cumplir los requisitos de nuestra Legislación y que el presente oficio se imprima en todos los ejemplares de su Tesis.

ATENTAMENTE,


~~M.I. JOSE JESUS HERNANDEZ ESPINO,~~
DIRECTOR.

c.c.p. Archivo

Estudio del Problema de Flujo de Bienes Múltiples a Costo Mínimo

Carla Alejandra Rivera Ramírez

Directora de Tesis: Dra. Rebeca del Rocío Peniche Vera

AGRADECIMIENTOS

A Dios, por permitirme terminar esta etapa de mi vida.

A mis profesores Alejandro Díaz Barriga y Jesús López Estrada, por transmitirme ese gusto de hacer matemáticas.

A Rocío Peniche, por la paciencia y la confianza que tuvo siempre para conmigo.

A todos los profesores que contribuyeron a mi formación: Alejandro Padilla, Angel Balderas, Agustín Pacheco, Antonio Ordaz, Herminio Blancarte, Lydia López, Jorge Martínez, J. Guadalupe Reyes, Martín Larios, Javier Gómez Mont, Roberto Torres, Marcia Canónico y José Guerrero.

A todos mis amigos: Martha, Félix, Manuel, Humberto, Verónica, Sergio, Ernesto, Yesica y Adela, por todos los momentos agradables que compartimos.

A Roberto Gómez, por darle una presentación digna a este trabajo.

A la Universidad Autónoma de Querétaro y a la Facultad de Ingeniería, por el apoyo ofrecido a esta carrera.

A mis padres Sara y Francisco

A Joaquín

Contenido

Introducción	3
1 El Problema de Flujo en Redes	5
1.1 Flujo Máximo	5
1.1.1 Algoritmo de Ford y Fulkerson	8
Ejemplo	9
1.2 Flujo a Costo Mínimo	11
1.2.1 Algoritmo de Klein	14
Ejemplo	16
1.2.2 Algoritmo de Busacker y Gowen	17
Ejemplo	19
1.2.3 Simplex Especializado en Redes	20
Ejemplo	23
2 Planteamiento del Problema	27
2.1 Formulación Nodo-Arco	27
2.2 Formulación Arco-Cadena	29
3 Un Método de Resolución Basado en el Método de Dantzig-Wolfe	35
3.1 Formulación Nodo-Arco	35
3.1.1 Ejemplo	39
3.2 Formulación Arco-Cadena	45
3.2.1 Ejemplo	48
4 Planteamiento y Resolución de Dos Casos Particulares	55
4.1 Problema de Transporte de Bienes Múltiples (Propiedad Topológica)	55
4.1.1 Ejemplo	62
4.2 Una Propiedad Topológica de la Red	64
4.2.1 Ejemplo	78
4.2.2 Un Caso Particular	79
Ejemplo	81
5 Caracterización de una Condición Bajo la Cual Existen Soluciones Enteras para el Problema en General	83
5.1 Preliminares	83
5.2 Desarrollo de Commoner	89
5.3 Presentación de la Operación Binaria “*”	90
5.4 Caracterización Algebraica	94
Conclusiones	101

A Conceptos Básicos	
A.1 Grupos	103
A.2 Teoría de Gráficas	103
A.3 Ruta más Corta	104
	109
B Principio de Descomposición de Dantzig-Wolfe	115
C El Problema de Transporte	
C.1 Presentación del Problema	119
C.2 Resolución	119
C.3 Ejemplo	120
	123
Bibliografía	125

Introducción

El problema de enviar un flujo de un artículo dado en una red, conocido en la literatura como el Problema de Flujo, es un tema muy estudiado y de hecho forma parte del temario de la materia de Análisis de Redes. Existen varios algoritmos, que además de ser eficaces para resolver el problema, estudian las propiedades del problema y cómo tomar ventaja de ellas. Ejemplo de estos algoritmos son el algoritmo de etiquetaje de Ford y Fulkerson, si el objetivo es maximizar la cantidad de flujo del artículo; o si se trata de minimizar costos, existen los algoritmos de Klein, de Busacker y Gowen, el simplex especializado en redes y el algoritmo de las desviaciones, entre otros.

Sin embargo, cuando en una red queremos enviar al mismo tiempo más de un artículo, el problema se complica un poco más. Esta complicación incluye el incremento en el número de variables y de restricciones, así como la pérdida de una propiedad algebraica muy importante que el problema de flujo de un solo artículo siempre presenta en su matriz de restricciones: la total unimodularidad. Por lo tanto, los métodos de resolución de marcaje para el problema de flujo de un solo bien ya no pueden aplicarse o generalizarse cuando tenemos dos o más artículos que deben circular en la red.

El propósito del presente trabajo es, pues, el estudio y análisis de esta generalización de problema de flujo a costo mínimo. El trabajo está dividido en cinco capítulos; se presenta en el primero el problema de flujo de un solo bien, tanto para el caso de maximizar la cantidad de flujo, así como la minimización de los costos, exponiendo algunos métodos de resolución. El segundo capítulo trata el planteamiento del problema que nos ocupa, con dos diferentes formulaciones, la formulación nodo-arco y la de arco-cadena, ya que ambas presentan una característica especial en su estructura, que permitirán, en el tercer capítulo, proponer un método de resolución para el problema en general; en [11], [14], [16] y [30] se exponen otros métodos de resolución al problema, sin embargo, el presentado aquí se eligió debido a que la estructura del problema satisface las condiciones del principio de descomposición de Dantzig-Wolfe, lo que lo hace interesante, desde el punto de vista de Análisis de Redes, puesto que los subproblemas son ampliamente conocidos y fáciles de resolver. En el cuarto capítulo, se analizan dos casos particulares en los que se presentan condiciones que debe cumplir la red asociada al problema, que permiten demostrar una equivalencia con problemas de flujo de un solo bien. Esto facilita la búsqueda de la solución del problema. En el quinto capítulo se establece una condición algebraica que garantiza la existencia de soluciones enteras para el problema en general, esto favorece el estudio de un algoritmo de marcaje eficiente que tome ventaja de esta propiedad.

Por último se presentan las conclusiones que se derivan de este estudio, y se anexan también tres apéndices; el primero presenta los conceptos y resultados necesarios para la total comprensión del contenido del trabajo. En el apéndice B se resume el principio de descomposición que se utiliza en el capítulo 3. En el apéndice C se presenta el problema de transporte, para ubicar al lector en el capítulo 4 sección 1.

Capítulo 1

El Problema de Flujo en Redes

Con objeto de hacer una introducción al lector en el tema, se presenta, basado en [18], el problema de flujo y sus diferentes métodos de resolución, tema al que nos vamos a referir durante el desarrollo de este trabajo. Supongamos que tenemos un lugar (al que llamaremos origen) donde se produce cierto artículo, y un lugar (llamado destino) donde es requerido. Tenemos varias opciones para transportar la cantidad del artículo en cuestión y estas rutas (o conductores) tienen cierta capacidad, es decir, la cantidad del artículo debe ser a lo más lo que cada conductor puede tolerar por unidad de tiempo. Imaginemos el caso, por ejemplo, de transportar costales de harina de un molino a un expendio, en este caso el camión (que es el conductor) tiene un volumen que no puede ser sobrepasado por viaje. Otra condición que existe en este problema, es considerar que entre los puntos intermedios de las rutas de transporte, no debe haber pérdida en la cantidad de flujo que está circulando, en otras palabras, si hay un lugar donde ni se produce, ni se requiere del artículo, la cantidad de flujo que entra a este punto debe ser la misma que la que sale de él; matemáticamente se dice que el flujo debe ser conservativo. Bajo estas condiciones consideraremos dos opciones para el objetivo del problema, a saber, maximizar la cantidad de flujo total en la red o minimizar los costos que se derivan de hacer circular una cierta cantidad de flujo.

En este capítulo se presentará a grandes rasgos, un método de resolución para cada uno de los dos problemas.

Se remite al lector al apéndice A, sección A.2 para la presentación de las definiciones de Teoría de Gráficas que se utilizarán en este trabajo.

1.1 Flujo Máximo

El modelo matemático para el problema de flujo sería el siguiente:

Definición 1. Sea una gráfica $G = (N, A)$ con capacidades en los arcos b_{ij} , un vértice origen o y un vértice destino d . Un vector de números ξ_{ij} definidos sobre los arcos (i, j) se llama *flujo* si satisface:

$$\sum_{j \in \Gamma^+(i)} \xi_{ij} - \sum_{k \in \Gamma^-(i)} \xi_{ki} = \begin{cases} v & \text{si } i = o \\ -v & \text{si } i = d \\ 0 & \text{si } i \neq o, d \end{cases} \quad (1.1)$$

El problema de flujo máximo de o a d en la red $R = (N, A, b)$ consiste en buscar un vector $\xi \in \mathbf{R}^m$ tal que

(1) ξ sea un flujo sobre la gráfica G .

(2) $0 \leq \xi_{ij} \leq b_{ij}$.

(3) v sea máximo bajo las condiciones precedentes.

Es decir,

$$\left\{ \begin{array}{l} \max z = \sum_{j \in \Gamma^+(o)} \xi_{oj} - \sum_{k \in \Gamma^-(o)} \xi_{ko} \\ \text{s.a.} \quad \sum_{j \in \Gamma^+(i)} \xi_{ij} - \sum_{k \in \Gamma^-(i)} \xi_{ki} = 0 \quad i \in N - \{o, d\} \\ \xi_{ij} \leq b_{ij} \quad (i, j) \in A \\ \xi_{ij} \geq 0 \quad (i, j) \in A \end{array} \right.$$

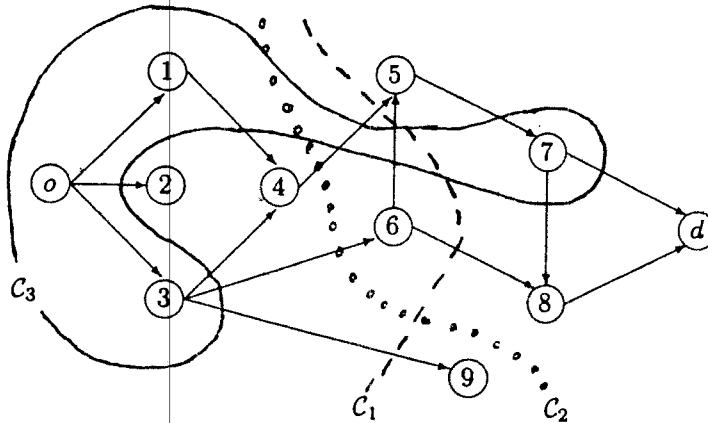
donde $\sum_{j \in \Gamma^+(o)} \xi_{oj} - \sum_{k \in \Gamma^-(o)} \xi_{ko} = v$ por (1.1).

Definición 2. Un vector $\xi \in \mathbb{R}^m$ que satisface (1) y (2) se llamará *flujo factible* sobre la red R .

Un método de solución fue dado por Ford y Fulkerson y su técnica forma la base de otros algoritmos que resuelven ciertos problemas que son generalizaciones simples o extensiones del problema.

Definición 3. En la red $R = (N, A, b)$ un conjunto de arcos C se llama *cortadura que separa* o de d si podemos encontrar $Y \subset N$ con $o \in Y, d \in \bar{Y}$, tal que $C = \{(i, j) \in A : x \in Y, y \in \bar{Y}\} = (Y, \bar{Y})$.

Ejemplo 1. En la siguiente gráfica hemos descrito tres cortaduras, a saber,



$$C_1 : Y_1 = \{o, 1, 2, 3, 4, 6\}, \quad \bar{Y}_1 = \{5, 7, 8, 9, d\}$$

$$C_2 : Y_2 = \{o, 1, 2, 3, 4, 9\}, \quad \bar{Y}_2 = \{5, 6, 8, d\}$$

$$C_3 : Y_3 = \{o, 1, 3, 7\}, \quad \bar{Y}_3 = \{2, 4, 5, 6, 8, 9, d\}$$

El arco $(5, 7)$ no es de la cortadura, pues $5 \in \bar{Y}_3$ y $7 \in Y_3$.

Observación 1. Sea c un camino de o a d en G y sea C una cortadura, tenemos:

$$c \cap C \neq \emptyset.$$

Definición 4. Llamaremos *valor del flujo* a

$$v(\xi) = \sum_{j \in \Gamma^+(o)} \xi_{oj} - \sum_{k \in \Gamma^-(o)} \xi_{ko}$$

Definición 5. Llamaremos *capacidad* de una cortadura C a la suma de las capacidades de los arcos que la constituyen.

$$b(C) = \sum_{(i,j) \in C} b_{ij}$$

Teorema 1. Para todo flujo factible ξ sobre $R = (N, A, b)$ y para toda cortadura que separa o de d tenemos

$$v(\xi) \leq b(C)$$

Demostración: Por definición de flujo tenemos

$$v(\xi) = \sum_{i \in Y} \sum_{j \in \Gamma^+(i)} \xi_{ij} - \sum_{i \in Y} \sum_{k \in \Gamma^-(i)} \xi_{ki} = \sum_{\substack{i \in Y \\ j \in Y}} \xi_{ij} + \underbrace{\sum_{\substack{i \in Y \\ j \in \bar{Y}}} \xi_{ij}}_{\text{sale de } C} - \sum_{\substack{i \in Y \\ k \in Y}} \xi_{ki} - \underbrace{\sum_{\substack{i \in Y \\ k \in \bar{Y}}} \xi_{ki}}_{\text{entra a } C}$$

Por lo tanto, y por ser flujo factible,

$$v(\xi) \leq \sum_{\substack{i \in Y \\ j \in \bar{Y}}} \xi_{ij} \leq \sum_{\substack{i \in Y \\ j \in \bar{Y}}} b_{ij}$$

Es decir,

$$v(\xi) \leq b(C).$$

El algoritmo de Ford y Fulkerson está basado en el teorema siguiente: ■

Teorema 2. El valor de un flujo máximo de o a d es igual al valor de la cortadura mínima que separa o de d .

Demostración: Es obvio que el flujo máximo de o a d no puede ser mayor que $b(C_{\min})$ porque todos los caminos de o a d tienen un arco de esta cortadura. Por el teorema 1, sabemos que $v(\xi) \leq b(C)$. Se demostrará que existe un flujo que alcanza el valor $b(C_{\min})$.

Supongamos dado un flujo por el vector ξ m -dimensional y definiremos una cortadura (Y_0, \bar{Y}_0) aplicando recursivamente el paso (b):

(a) Sea $Y_0 = \{o\}$

(b) Si $i \in Y_0$ y $\xi_{ij} < b_{ij}$ o $\xi_{ji} > 0$, colocar j en Y_0 y repetir hasta que Y_0 no pueda incrementarse.

Entonces suceden dos casos: $d \in Y_0$ o $d \notin Y_0$.

Supongamos $d \in Y_0$. De acuerdo con el paso (b), $d \in Y_0$ implica que existe una cadena de arcos de o a d tal que para que todo arco (i, j) en la cadena, usado en la dirección "directa" (de o a d) $\xi_{ij} < b_{ij}$ y para todo arco (k, l) usado en la cadena en dirección "inversa" (de l a k) $\xi_{kl} > 0$. Esta cadena se llamará *augmentante*. Sean

$$\begin{aligned} \delta_D &= \min \{b_{ij} - \xi_{ij} : (i, j) \text{ es directo}\} \\ \delta_I &= \min \{\xi_{kl} : (k, l) \text{ es inverso}\} \end{aligned}$$

tomemos

$$\delta = \min \{\delta_D, \delta_I\}$$

si δ se agrega al flujo en todos los arcos directos y se sustrae de todos los arcos inversos, el resultado es un nuevo flujo factible con δ unidades mayor que el anterior. Usando el nuevo flujo pueden aplicarse los pasos (a) y (b) anteriores para definir una nueva cortadura y repetir el argumento.

Supongamos $d \notin Y_0$. De acuerdo con el paso (b)

$$\xi_{ij} = b_{ij} \quad \forall (i, j) \in (Y_0, \bar{Y}_0)$$

y

$$\xi_{kl} = 0 \quad \forall (k, l) \in (\bar{Y}_0, Y_0)$$

de donde

$$\sum_{(i,j) \in (Y_0, \bar{Y}_0)} \xi_{ij} = \sum_{(i,j) \in (Y_0, \bar{Y}_0)} b_{ij}$$

y

$$\sum_{(k,l) \in (\bar{Y}_0, Y_0)} \xi_{kl} = 0$$

Entonces el valor del flujo

$$\begin{aligned} v(\xi) &= \sum_{(i,j) \in (Y_0, \bar{Y}_0)} \xi_{ij} + \sum_{(k,l) \in (\bar{Y}_0, Y_0)} \xi_{kl} \\ &= \sum_{(i,j) \in (Y_0, \bar{Y}_0)} b_{ij} + 0 \\ &= b(Y_0, \bar{Y}_0). \end{aligned}$$

Tenemos pues, un flujo cuyo valor es igual al valor de una cortadura. El flujo debe ser máximo y la cortadura mínima. ■

1.1.1 Algoritmo de Ford y Fulkerson

A) Proceso de etiquetado.

- (1) Etiquetar al vértice o con $\{+o, \infty\}$
- (2) Elegir un vértice etiquetado, sea i , sea su etiqueta: $\{\pm i, \delta(i)\}$
 - (a) Para todos los vértices $j \in \Gamma^+(i)$ tales que no estén etiquetados y para los cuales $\xi_{ij} < b_{ij}$, asignarles la etiqueta: $\{+i, \delta(i)\}$, donde $\delta(j) = \min \{\delta(i), b_{ij} - \xi_{ij}\}$
 - (b) Para todos los vértices $j \in \Gamma^-(i)$ tales que no estén etiquetados y para los cuales $\xi_{ji} > 0$, asignarles la etiqueta: $\{-i, \delta(i)\}$, donde $\delta(j) = \min \{\delta(i), \xi_{ji}\}$

Se dice ahora que el vértice i ha sido etiquetado.

(3) Repetir el paso (2) hasta que suceda:

- (a) El vértice d no ha sido etiquetado y ya todos los vértices han sido examinados (excepto d , claro). ALTO, el flujo ξ es el flujo máximo.
- (b) El vértice d recibe etiqueta, ir a (4).

B) Proceso de aumentación de flujo.

- (4) Sea $j = d$.
- (5) (a) Si la etiqueta de j es de la forma $(+i, \delta(j))$, cambiar el flujo a lo largo del arco (i, j) de ξ_{ij} a $\xi_{ij} + \delta(j)$.
- (b) Si la etiqueta de j es de la forma $(-i, \delta(j))$, cambiar el flujo a lo largo del arco (j, i) de ξ_{ji} a $\xi_{ji} - \delta(j)$.

(6) Si $i = o$ borrar todas las etiquetas y regresar al paso (1), repetir el proceso de etiquetado empezando con el nuevo flujo calculado en (5).

Si $i \neq o$ poner $j = i$ y regresar al paso (5).

C) Cortadura mínima.

Si d ya no pudo ser etiquetado.

La cortadura mínima está formada por el conjunto de arcos (Y_0, \bar{Y}_0) donde $Y_0 = \{i : i \in N, i \text{ etiquetado}\}$.

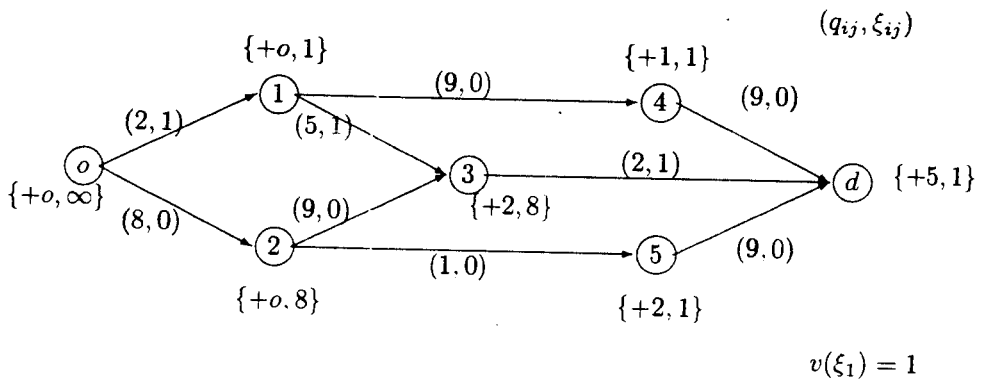
Teorema 3 (finitud del algoritmo). Si las capacidades de los arcos son enteros y si existe una cortadura de capacidad finita, el algoritmo de Ford y Fulkerson proporciona, en un número finito de iteraciones, una solución óptima del problema de flujo máximo de o a d .

Demostración: Partiendo de un flujo ξ de valor 0, en cada iteración, éste se incrementa en una cantidad δ que es un entero, el valor del flujo está acotado superiormente, ya que $v(\xi) \leq b(C)$ y existe una cortadura C de capacidad finita. Por lo tanto, el número de iteraciones es finito. ■

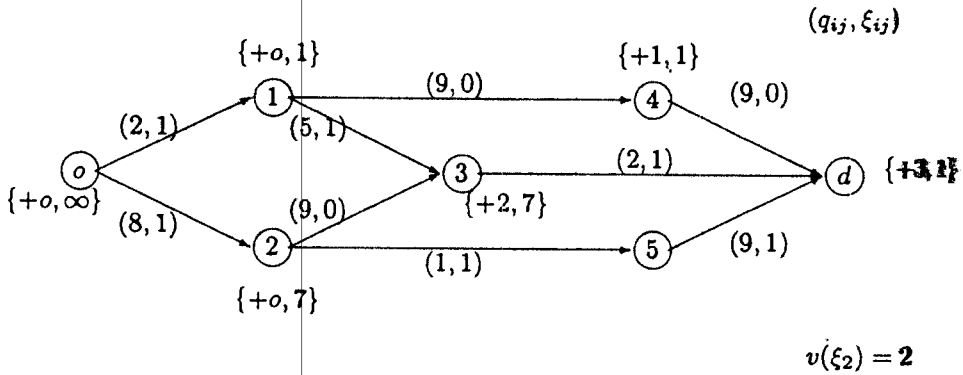
Ejemplo

Obtener el flujo máximo en la red.

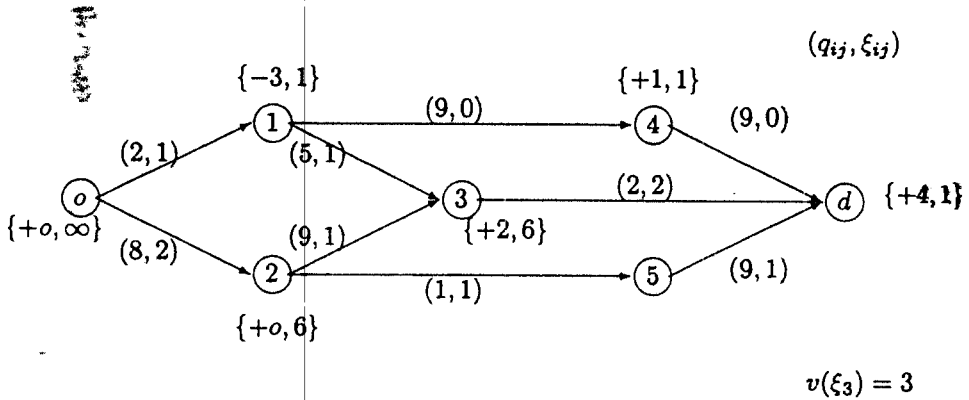
Se tiene una unidad de flujo circulando en la red.



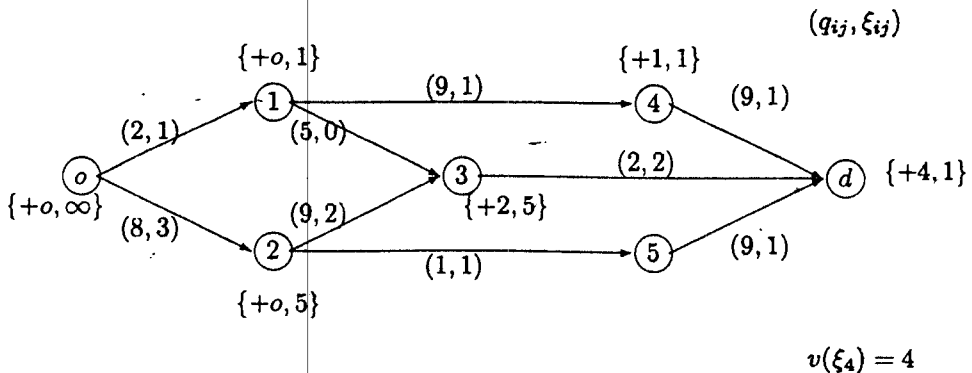
Se ha encontrado una cadena por donde puede circular una unidad de flujo $\{o, 2, 5, d\}$. Se actualiza la red, se borran todas las etiquetas y se busca otra cadena aumentante.



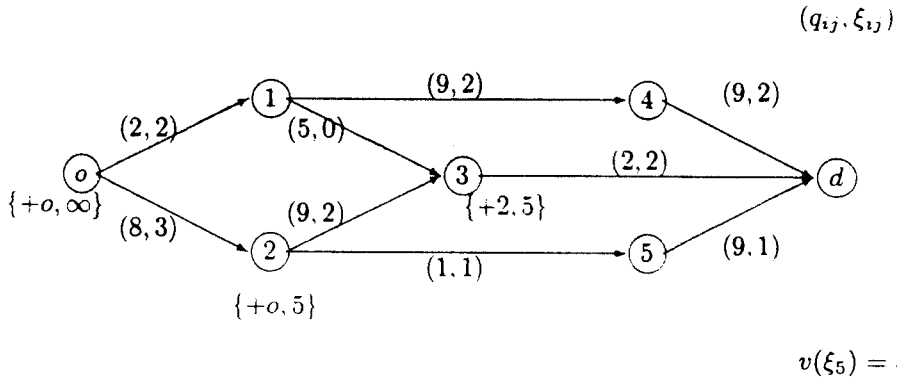
Encontramos la cadena aumentante con vértices $\{o, 2, 3, d\}$, actualizamos la red, borramos las etiquetas y buscamos otra cadena aumentante.



Se tiene ahora la cadena $\{o, 2, 3, 1, 4, d\}$, por donde puede circular una unidad de flujo, se actualiza la red, se borran las etiquetas y se busca otra cadena aumentante.



Tenemos ahora la cadena aumentante $\{o, 1, 4, d\}$. Actualizamos la red, borramos etiquetas y buscamos otra cadena aumentante.



Como d ya no puede recibir etiqueta, obtenemos una cortadura $C = (Y_0, \bar{Y}_0)$, donde $Y_0 = \{o, 2, 3\}$ y con capacidad, $q(C) = 2 + 2 + 1 = 5$. Por lo tanto, $v(\xi_{\max}) = 5$.

1.2 Flujo a Costo Mínimo

Ahora vamos a considerar como el objetivo del problema la minimización de los costos. En este caso la cantidad de flujo es conocida y cada ruta (cada arco) tiene asociado un costo por unidad de flujo. Por ejemplo, el costo de viajar en autobús de Querétaro a México está determinado por un precio en el boleto de viaje, y es un costo por persona que viaja.

Las dos primeras secciones están destinadas a presentar métodos de resolución basados en el algoritmo de Ford y Fulkerson, para ello es necesario introducir algunos conceptos. En la tercera sección se expone una variante del algoritmo simplex para redes con capacidades en los nodos.

El planteamiento matemático del problema de flujo a costo mínimo está dado por:

$$\left\{ \begin{array}{l} \min w = \sum_{(i,j) \in A} c_{ij} \xi_{ij} \\ \text{s.a.} \quad \sum_{j \in \Gamma^+(o)} \xi_{oj} - \sum_{k \in \Gamma^-(o)} \xi_{ko} = v \\ \sum_{j \in \Gamma^+(i)} \xi_{ij} - \sum_{k \in \Gamma^-(i)} \xi_{ki} = 0 \quad i \in N - \{o, d\} \\ \sum_{j \in \Gamma^+(d)} \xi_{dj} - \sum_{k \in \Gamma^-(d)} \xi_{kd} = -v \\ \xi_{ij} \leq b_{ij} \quad (i, j) \in A \\ \xi_{ij} \geq 0 \quad (i, j) \in A \end{array} \right.$$

donde c_{ij} es el costo por unidad de flujo del arco (i, j) , y v , el valor del flujo, es conocido.

Definición 6. Un circuito C tal que $\sum_{u \in C} c(u) < 0$, donde $c(u)$ es el costo asociado al arco u , se llama *circuito absorbente*.

Observación 2. Este tipo de circuitos pueden ser encontrados con algunos algoritmos de ruta más corta tomando como distancias en los arcos los costos. En el apéndice A, en la sección A.3 se presenta un algoritmo de ruta más corta que permite detectar circuitos absorbentes en una red, a parte, claro, de proporcionar la ruta más corta del origen al destino.

Definición 7. El proceso de encontrar una cadena aumentante en una gráfica $G = (N, A)$ donde los flujos en los arcos están dados por el vector ξ , puede ser considerado como una búsqueda de caminos en una *gráfica incremental* $G^\mu(\xi) = (N, A^\mu)$ definida como sigue:

$$A^\mu = A_1^\mu \cup A_2^\mu$$

donde

$$A_1^\mu = \{(i^\mu, j^\mu) : \xi_{ij} < b_{ij}\}.$$

Un arco $(i^\mu, j^\mu) \in A_1^\mu$ tendrá capacidad

$$b_{ij}^\mu = b_{ij} - \xi_{ij}$$

y costo

$$c_{ij}^\mu = c_{ij}$$

y

$$A_2^\mu = \{(j^\mu, i^\mu) : \xi_{ij} > 0\}$$

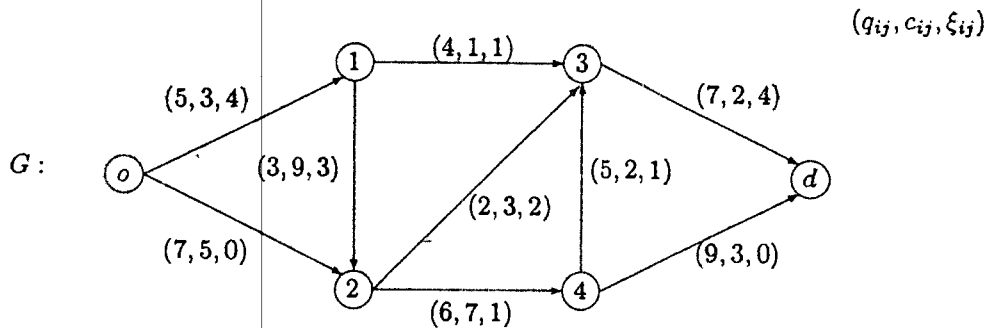
Un arco $(j^\mu, i^\mu) \in A_2^\mu$ tendrá capacidad

$$b_{ji}^\mu = \xi_{ij}$$

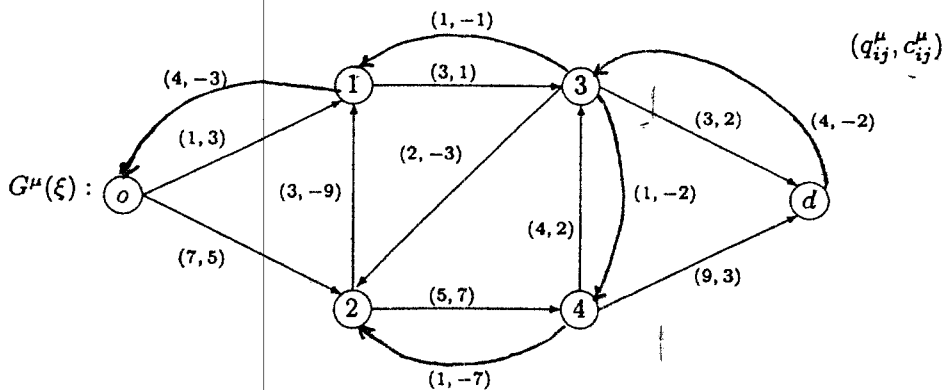
y costo

$$c_{ji}^\mu = -c_{ij}$$

Ejemplo 2. Considérese la red siguiente



Su gráfica incremental está dada por:



Algunas veces es deseable representar un flujo como la suma de flujos elementales. Esto es útil porque contribuye a un mejor entendimiento de la naturaleza de los flujos en redes y sirve para justificar algunos de los algoritmos de flujo en redes.

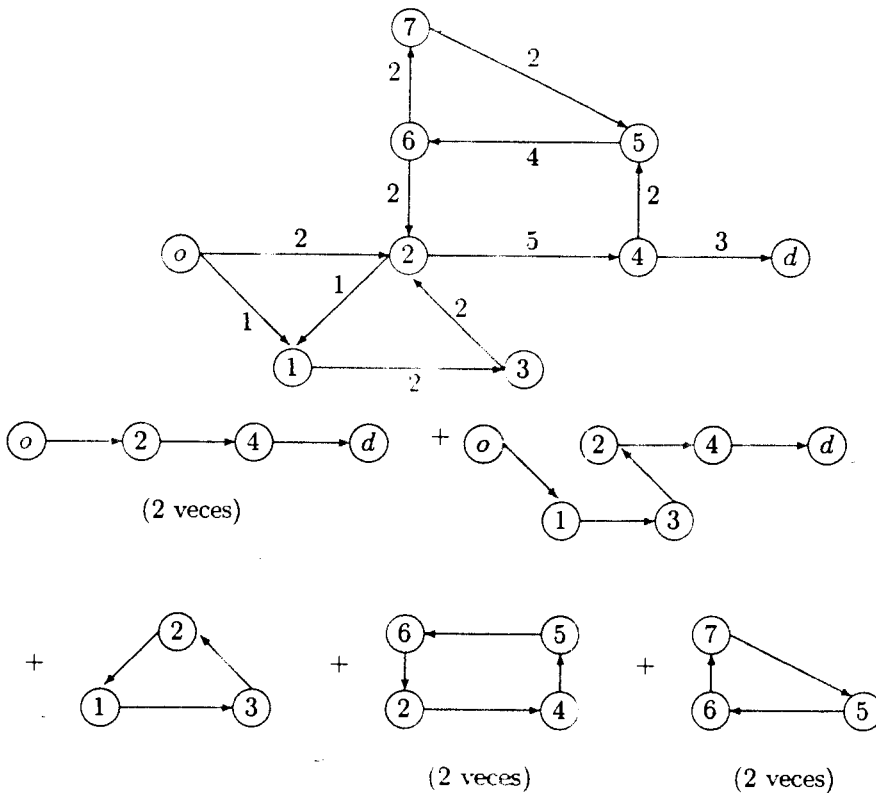
Denotemos por $h \circ (S)$ un flujo en G en el cual los arcos $(i, j) \in S$ tienen $\xi_{ij} = h$. Obviamente $h \circ (S)$ no es un flujo para cualquier conjunto de arcos S puesto que por ser flujo debe satisfacer las ecuaciones de conservación de flujo. Es claro que para que $h \circ (S)$ represente un flujo, el conjunto de arcos de S debe formar un camino en G (o varios) de o a d o un circuito (o varios) en G .

Teorema 4. Si ξ es cualquier flujo (de o a d) de valor entero v en una gráfica G , entonces ξ puede descomponerse de la siguiente manera:

$$\xi = 1 \circ P_1 + 1 \circ P_2 + \dots + 1 \circ P_v + 1 \circ (\phi_1) + 1 \circ (\phi_2) + \dots + 1 \circ (\phi_k)$$

donde P_1, P_2, \dots, P_v son caminos elementales (de o a d) de G y $\phi_1, \phi_2, \dots, \phi_k$ son circuitos elementales de G .

Ejemplo 3. Considérese la siguiente red con un flujo ξ de valor $v(\xi) = 3$.



Demostración del teorema 4: A partir de la gráfica $G = (N, A)$ con flujo ξ construir la gráfica unitaria $G^e = (N, A^e)$ como sigue: Si ξ_{ij} es el flujo en el arco $(i, j) \in A$ entonces poner ξ_{ij} arcos en paralelo entre los correspondientes vértices i^e y j^e de G^e . Si $\xi_{ij} = 0$, entonces no ponemos arco entre i^e y j^e . Puesto que cada arco en G^e corresponde a un arco de flujo unitario en G , G^e representa el flujo ξ en G .

En G^e los grados de los vértices deben satisfacer (por las condiciones de conservación de flujo) que:

$$\begin{aligned}g^+(i^e) &= g^-(i^e) \quad \forall i^e \neq o^e, d^e \\g^+(o^e) &= g^-(d^e) = v\end{aligned}$$

Se tienen v caminos de o^e a d^e que en su totalidad atraviesan cada arco de G^e exactamente una vez. Sean esos caminos P'_1, P'_2, \dots, P'_v . Los caminos P'_i no son necesariamente elementales aunque sí deben ser simples. Como cualquier camino no elemental puede ser considerado como la suma de un camino elemental (de o^e a d^e) y un número de circuitos elementales arco-ajenos, obtenemos:

$$\xi = 1 \circ (P_1) + 1 \circ (P_2) + \dots + 1 \circ (P_v) + 1 \circ (\phi_1) + 1 \circ (\phi_2) + \dots + 1 \circ (\phi_k)$$

donde P_i son caminos de o^e a d^e elementales y ϕ_i son circuitos elementales. ■

Observación 3. Dada cualquier red R , el valor del flujo que circula a través de ella es

$$v(\xi) = \sum_{i \in \Gamma^+(o)} \xi_{oi} = \sum_{j \in \Gamma^-(d)} \xi_{jd}$$

Proposición 1. Dado un flujo factible ξ , si

$$\xi = 1 \circ P_1 + 1 \circ P_2 + \dots + 1 \circ P_v + 1 \circ (\phi_1) + 1 \circ (\phi_2) + \dots + 1 \circ (\phi_k)$$

entonces $v(\xi) = v$.

Demostración: Sabemos que $v(\xi) = \sum_{i \in \Gamma^+(o)} \xi_{oi}$

Como cada P_i es un camino elemental que va de o a d con una unidad de flujo se tiene que

$$v(\xi) = \underbrace{1 + \dots + 1}_{v \text{ veces}} = v$$

El valor del flujo en los circuitos ϕ_i no cuenta en el cálculo de $v(\xi)$, pues no son caminos de o a d , aún cuando en alguno de ellos intervenga el vértice o o el vértice d (no ambos, pues si estuvieran los dos, del circuito puede obtenerse un camino de o a d y éste entraría como un P_i). ■

1.2.1 Algoritmo de Klein

El procedimiento mejor conocido para resolver el problema de flujo a costo mínimo es el algoritmo *out-of-kilter* (desviaciones) de Ford y Fulkerson, pero aquí se verán dos algoritmos, el de Klein y el de Busacker y Gowen, que son conceptualmente más sencillos que el *out-of-kilter*, y desde el punto de vista teórico tienen un atractivo particular, ya que constituyen un par de algoritmos duales entre sí.

Computacionalmente hablando, los métodos son comparables.

Algoritmo de Klein.

Se basa en la determinación de circuitos negativos (absorbentes).

- (1) Usar un algoritmo para encontrar un flujo factible ξ de valor v en G .
- (2) Construir la gráfica $G^\mu(\xi)$.
- (3) Verificar si existe un circuito absorbente sobre los costos ϕ en $G^\mu(\xi)$. Si existe, identificarlo e ir a (4). Si no existe, ALTO, el flujo es de costo mínimo.

(4) Calcular $\delta = \min \{b_{ij}^\mu : (i^\mu, j^\mu) \in \phi\}$

(a) $\forall (i^\mu, j^\mu) \in \phi$ con $(i^\mu, j^\mu) \in A_2^\mu$, cambiar el flujo ξ_{ji} en el arco correspondiente (j, i) de G de ξ_{ij} a $\xi_{ij} - \delta$.

(b) $\forall (i^\mu, j^\mu) \in \phi$ con $(i^\mu, j^\mu) \in A_1^\mu$, cambiar el flujo ξ_{ij} en el arco correspondiente (i, j) de G de ξ_{ij} a $\xi_{ij} + \delta$.

(5) Con este nuevo flujo ξ , ir a (2).

Teorema 5. ξ es un flujo de costo mínimo de valor v si y sólo si, no existe circuito ϕ en $G^\mu(\xi)$ tal que la suma de los costos de los arcos en ϕ sea negativa.

Demostración: Sea $c[\xi]$ el costo del flujo ξ en G y $c[\phi : G^\mu(\xi)]$ la suma de los costos de los arcos en el circuito ϕ con respecto a $G^\mu(\xi)$.

Necesidad. Supongamos que ξ es un flujo de costo mínimo de valor v . Supongamos que existe un circuito ϕ en $G^\mu(\xi)$ tal que $c[\phi : G^\mu(\xi)] < 0$. La circulación de una unidad de flujo adicional a través del circuito ϕ produce el nuevo flujo $\xi + 1 \circ (\phi)$ y del valor v del flujo de G es invariable. El costo del flujo $\xi + 1 \circ (\phi)$ es

$$c[\xi] + c[\phi : G^\mu(\xi)] < c[\xi]$$

pero esto contradice el hecho de que ξ es un flujo de costo mínimo.

Suficiencia. Supongamos que $c[\phi : G^\mu(\xi)] \geq 0 \forall$ circuito ϕ en $G^\mu(\xi)$. Por demostrar que ξ es un flujo de costo mínimo de valor v . Supongamos que $\xi^* (\neq \xi)$ es el flujo de costo mínimo de valor v . Sea $\xi^* - \xi$ el flujo para el cual el flujo en el arco (i, j) es $\xi_{ij}^* - \xi_{ij}$. $G^\mu(\xi^* - \xi)$ consiste de una colección de circuitos de flujo unitario. Sean, por ejemplo, $\phi_1, \phi_2, \dots, \phi_k$ y podemos escribir:

$$\xi^* - \xi = 1 \circ (\phi_1) + 1 \circ (\phi_2) + \dots + 1 \circ (\phi_k)$$

Sabemos que el flujo ξ es factible, cualquier suma

$$\xi + 1 \circ (\phi_1) + 1 \circ (\phi_2) + \dots + 1 \circ (\phi_k)$$

es factible para cualquier $1 \leq l \leq k$. Así, considerando el flujo $\xi + 1 \circ (\phi_1)$ tenemos:

$$c[\xi + 1 \circ (\phi_1)] = c[\xi] + c[\phi_1 : G^\mu(\xi)] \geq c[\xi]$$

El costo del flujo $\xi + 1 \circ (\phi_1) + 1 \circ (\phi_2)$ es

$$c[\xi + 1 \circ (\phi_1) + 1 \circ (\phi_2)] = c[\xi + 1 \circ (\phi_1)] + c[\phi_2 : G^\mu(\xi)] \geq c[\xi + 1 \circ (\phi_1)] \geq c[\xi]$$

Continuando de esta manera, finalmente obtenemos

$$c[\xi^*] \geq c[\xi],$$

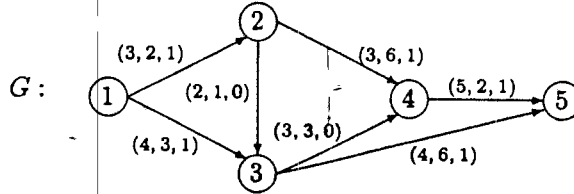
de donde ξ es un flujo de costo mínimo de valor v . ■

Justificación del algoritmo de Klein: En el paso (3) del algoritmo, si no existe un circuito absorbente, i.e. un circuito tal que la suma de los arcos (sobre los costos) del ciclo sea negativa, entonces por el teorema 5 el flujo ξ es el flujo de costo mínimo de valor v .

Ejemplo

Encontrar un flujo de valor 2 a costo mínimo.

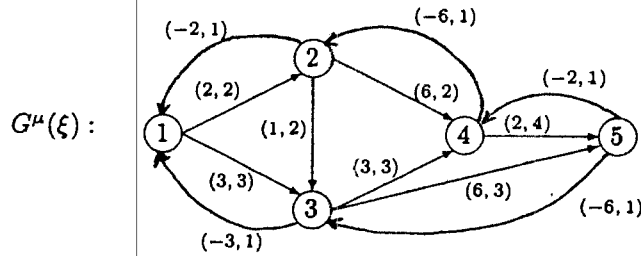
$(q_{ij}, c_{ij}, \xi_{ij})$



$v(\xi) = 2$ Con costo $c_1 = 19$

Construimos la gráfica incremental:

(c_{ij}^μ, q_{ij}^μ)

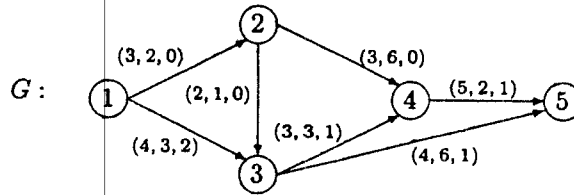


$\phi_1 = \{(1, 3), (3, 4), (4, 2), (2, 1)\}$

$v(\phi_1) = -2$

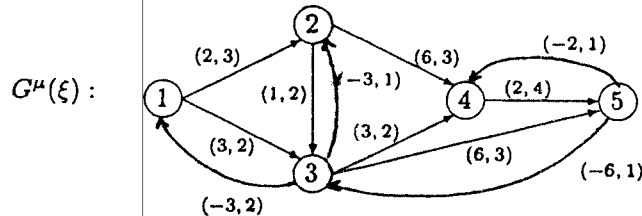
$\delta = \min \{1, 1, 3, 3\} = 1$

Actualizandó el flujo en la gráfica original:



Por lo tanto, $v(\xi) = 2$ Con costo $c_2 = 17$

Construimos nuevamente la gráfica incremental:

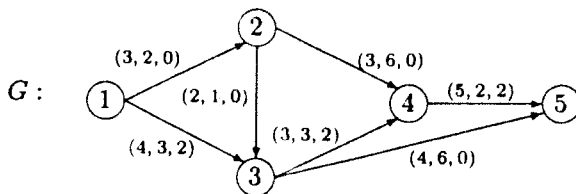


$\phi_2 = \{(3, 4), (4, 5), (5, 3)\}$

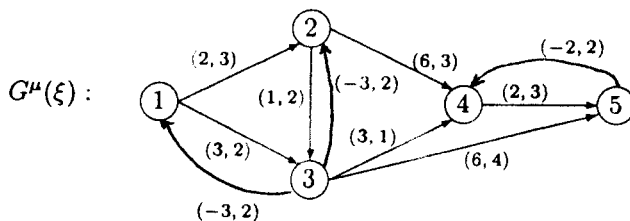
$v(\phi_2) = -1$

$\delta = \min \{1, 2, 4\} = 1$

Actualizamos el flujo en la gráfica original:



Por lo tanto, $v(\xi) = 2$ Con costo $c_3 = 16$
 Gráfica incremental correspondiente a este flujo:



Observamos que ya no existen circuitos absorbentes (esto puede verificarse utilizando algún algoritmo de ruta más corta).

1.2.2 Algoritmo de Busacker y Gowen

Se basa en la determinación de rutas más cortas.

- (1) Empezar con un flujo ξ de valor cero.
- (2) Construir la gráfica $G^\mu(\xi)$.
- (3) Encontrar el camino más corto P^* en $G^\mu(\xi)$ de o a d .
- (4) Calcular la máxima cantidad de flujo δ que pueda mandarse a través de P^*

$$\delta = \min \{b_{ij}^\mu : (i^\mu, j^\mu) \in P^*\}$$

- (a) Si el valor del flujo $\xi + \delta \circ P^*$ es menor que v , actualizar ξ con $\xi + \delta \circ P^*$ en G y regres a (2).
- (b) Si el valor del flujo $\xi + \delta \circ P^*$ es v , ALTO, es el flujo de costo mínimo requerido. Actualizar ξ con $\xi + \delta \circ P^*$ en G .
- (c) Si $h = \xi + \delta \circ P^* > v$, ALTO, $\xi + (\delta - h + v) \circ P^*$ es el flujo de costo mínimo requerido. Actualizar en G .

Teorema 6. Dado que ξ es un flujo de costo mínimo en G de valor v y P^* es la ruta más corta (costo mínimo) de o a d en la gráfica incremental $G^\mu(\xi)$, entonces $\xi + 1 \circ (P^*)$ es el flujo de costo mínimo en G de valor $v + 1$.

Demostración: Primeramente obsérvese que $\xi + 1 \circ (P^*)$ es de valor $v + 1$. Por otro lado, puesto que ξ es de costo mínimo, la gráfica incremental $G^\mu(\xi)$ no contiene circuitos negativos.

Para demostrar que $\xi + 1 \circ (P^*)$ también es de costo mínimo basta probar que la gráfica incremental $G^\mu(\xi + 1 \circ (P^*))$ tampoco contiene circuitos negativos.

Las gráficas $G^\mu(\xi)$ y $G^\mu(\xi + 1 \circ (P^*))$ coinciden, excepto posiblemente, en los arcos de P^* . Considérese la partición de los arcos de P^* en los conjuntos D (directo) e I (inverso).

Si

$$\xi_{ij} + 1 < b_{ij} \quad \forall (i, j) \in D$$

y

$$\xi_{ij} - 1 > 0 \quad \forall (i, j) \in I$$

es claro que $G^\mu(\xi)$ y $G^\mu(\xi + 1 \circ (P^*))$ contienen los mismos arcos y por lo tanto $G^\mu(\xi + 1 \circ (P^*))$ no contiene circuitos negativos.

Ahora, si existe algún $(i, j) \in D$ tal que $\xi_{ij} + 1 = b_{ij}$, entonces este arco que pertenece a $G^\mu(\xi)$, no pertenece a $G^\mu(\xi + 1 \circ (P^*))$ y el arco $(j, i) \in G^\mu(\xi + 1 \circ (P^*))$. Supóngase que este arco pertenece a un circuito ϕ de $G^\mu(\xi + 1 \circ (P^*))$, sean los vértices de ϕ : $i, x_1, x_2, \dots, x_k, j, i$. Puesto que P^* es una ruta más corta de o a d en $G^\mu(\xi)$, entonces el arco (i, j) es una ruta más corta de i a j , ya que pertenece a P^* . Por lo tanto, en la gráfica $G^\mu(\xi)$ se tiene que

$$c_{ij} \leq c_{i1} + c_{12} + \dots + c_{kj}$$

luego

$$c[\phi] = c_{i1} + c_{12} + \dots + c_{kj} - c_{ij} \geq 0$$

y por lo tanto, ϕ no es circuito negativo. Análogamente puede demostrarse que $G^\mu(\xi + 1 \circ (P^*))$ no contiene circuitos negativos si existe algún arco $(i, j) \in I$ tal que $\xi_{ij} - 1 = 0$. ■

Justificación del algoritmo de Busaker y Gowen: En el paso (4) del algoritmo el flujo $\xi + \delta \circ P^*$ es de valor $v' + \delta$ si ξ es de valor v' . Esto puede verse si consideramos el flujo $\xi + \delta \circ P^*$ de la siguiente manera:

$\xi_1 = \xi + 1 \circ P^*$ que es un flujo de valor $v' + 1$,

$\xi_2 = \xi_1 + 1 \circ P^*$ que es un flujo de valor $(v' + 1) + 1 = v' + 2$,

⋮

$\xi_\delta = \xi_{\delta-1} + 1 \circ P^*$ que es un flujo de valor $(v' + \underbrace{1 + \dots + 1}_{\delta-1}) + 1 = v' + \delta$.

Por lo tanto, el flujo $\xi + \delta \circ P^*$ es de valor $v' + \delta$. Por el teorema 6, cada uno de los flujos ξ_i es un flujo de valor $v' + i$ a costo mínimo, pues ξ es un flujo de valor v' a costo mínimo. Por lo tanto, si $v' + \delta = v$ entonces el flujo $\xi + \delta \circ P^*$ es un flujo de valor v a costo mínimo. Si $v' + \delta > v$ entonces si $h = \xi + \delta \circ P^*$, el flujo $\xi + (\delta - h + v) \circ P^*$ es un flujo de valor v , pues

$$\xi + (\delta - h + v) \circ P^* = \xi + (\delta - \xi - \delta + v) \circ P^* = \xi + (v - \xi) \circ P^*$$

y por el teorema 6, es un flujo a costo mínimo.

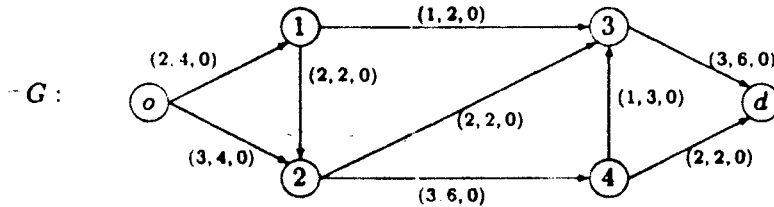
Observación 4. El primer algoritmo es un algoritmo "primal", ya que reorienta el flujo de valor v hasta obtener el "óptimo" (costo mínimo) manteniendo factibilidad (flujo factible). El segundo algoritmo es el método "dual", es decir, construye el flujo de costo mínimo ξ^* de valor v empezando con un flujo de costo mínimo que tiene algún valor $v_0 < v$, agregando flujo a la red para obtener un nuevo flujo a costo mínimo de valor $v_1 > v_0$, etc. hasta obtener ξ^* de valor v . Siempre manteniendo "optimalidad" (flujo a costo mínimo). Observamos que siempre es posible empezar con un flujo de costo mínimo, ya que el flujo cero es el flujo de costo mínimo de valor cero.

Ejemplo

Mandar un flujo de valor 2 a costo mínimo.

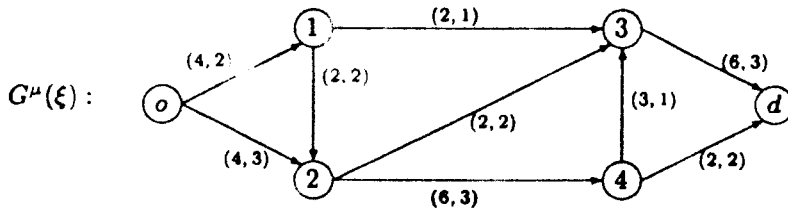
Comenzamos considerando un flujo de valor cero.

$$(q_{ij}, c_{ij}, \xi_{ij})$$



Construimos la gráfica incremental y buscamos el camino más corto de o a d .

$$(c_{ij}^\mu, q_{ij}^\mu)$$

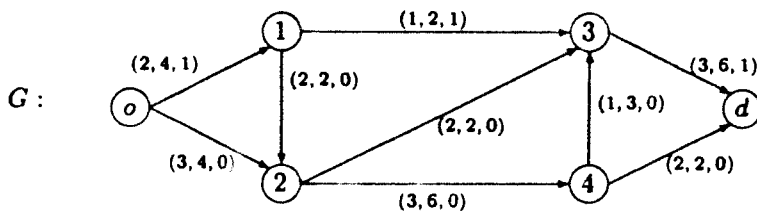


$P^* = \{(o, 1), (1, 3), (3, d)\}$ cuyo costo es 12.

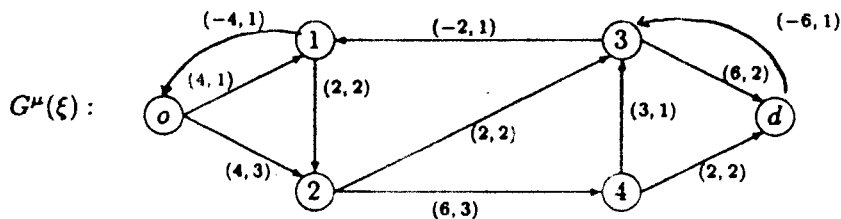
Calculamos la máxima cantidad de flujo que puede circular:

$$\delta = \min \{2, 1, 3\} = 1$$

Tenemos un flujo de valor 1 a costo mínimo que es igual a 12.



Como todavía no tenemos un flujo de valor 2 continuamos el algoritmo, construyendo la gráfica incremental.



$P^* = \{(o, 2), (2, 4), (4, d)\}$ cuyo costo es 12.

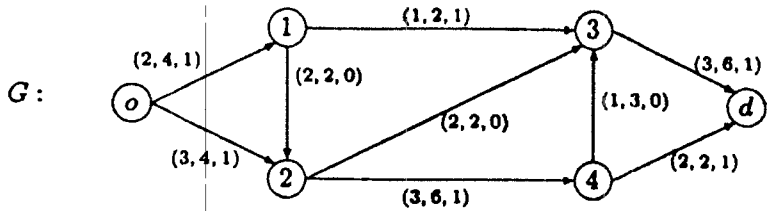
$$\delta = \min \{3, 3, 2\} = 2$$

$$\xi_1 + 2 \circ P^* = 1 + 2 \circ P^* = 3 > v$$

$$\Rightarrow h = 1 + 2 \circ P^*$$

$$\Rightarrow \xi_1 + (\delta - h + v) \circ P^* = 1 + (2 - 3 + 2) \circ P^* = 1 + 1 \circ P^* = 2 = v$$

Por lo tanto, tenemos ya un flujo de valor 2 a costo mínimo igual a 24.



1.2.3 Simplex Especializado en Redes

Este método resuelve el problema de minimizar costos en una red en la que cada vértice i en G tiene asociado un número entero r_i , que representa los recursos disponibles de un artículo (si $r_i > 0$) o la demanda del artículo (si $r_i < 0$). Los vértices con $r_i > 0$ se llaman *orígenes* y los nodos con $r_i < 0$ se llaman *destinos*. Si $r_i = 0$ entonces ningún artículo está disponible en el vértice i y ninguno se requiere; en este caso, se dice que el vértice i es un nodo *intermedio* o de *transbordo*. Asociado con cada arco (i, j) se tiene el número x_{ij} , que representa la cantidad de flujo sobre el arco (se supone que $x_{ij} \geq 0$), y el número entero c_{ij} , que es el costo unitario de transporte a lo largo del arco.

Supondremos que la oferta total en la red es igual a la demanda total, es decir, que

$$\sum_{i \in N} r_i = 0$$

Observación 5. Si $\sum_{i \in N} r_i > 0$ entonces se añade un vértice ficticio $n + 1$ con demanda

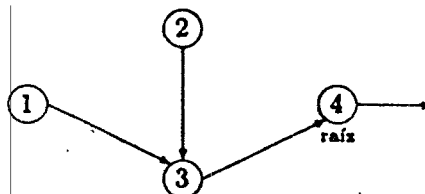
$$r_{m+1} = - \sum_{i \in N} r_i$$

y con arcos con costo cero desde cada vértice origen al nuevo vértice.

El problema de flujo a costo mínimo en una red tiene la siguiente representación matemática:

$$\begin{cases} \min w = & \sum_{(i,j) \in A} c_{ij} x_{ij} \\ \text{s.a.} & \sum_{j \in N} x_{ij} - \sum_{k \in N} x_{ki} = r_i \quad \forall i \in N \\ & x_{ij} \geq 0 \quad \forall (i,j) \in A \end{cases} \quad (1.2)$$

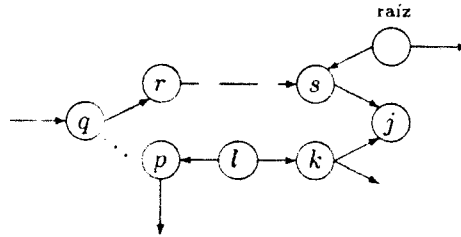
Observación 6. El método simplex siempre inicia con una matriz de restricciones de rango total, y la matriz de restricciones de este problema tiene rango $n - 1$, por lo tanto, se requiere de una variable artificial de tal forma que el rango de la nueva matriz sea n . Gráficamente esta variable artificial se interpreta como la adición de un arco que empieza en el nodo n y termina en el espacio. Este arco con un sólo extremo se llama *arco raíz* y el vértice al que es incidente se llama *raíz*. Véase la figura siguiente.



Los vértices pendientes se denominan *terminales*.

Observación 7. Una matriz básica B para el problema (1.2) está caracterizada por un *árbol de expansión enraizado*. Es decir, un árbol de expansión con un arco raíz.

Ahora, considérese la subgráfica básica G_B correspondiente a un árbol de expansión enraizado y fijémosnos en cualquier arco no básico (r, s) . Como G_B es un árbol, se sabe que al añadir este arco no básico se forma un único ciclo.



Asignando al ciclo la orientación determinada por el arco (r, s) se tiene:

$$\begin{aligned} & a_{rs} + a_{sj} - a_{kj} - a_{lk} + a_{lp} + \dots + a_{qr} \\ &= (e_r - e_s) + (e_s - e_j) - (e_k - e_j) - (e_l - e_k) + (e_l - e_p) + \dots + (e_q - e_r) \\ &= 0 \end{aligned}$$

donde e_i son vectores unitarios en \mathbf{R}^m con un 1 la i -ésima posición.

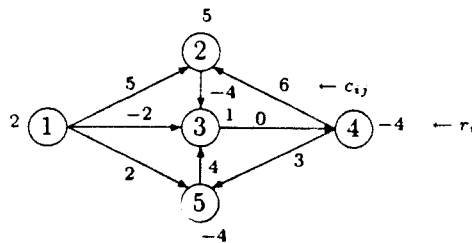
Como tenemos una suma igual a cero, podemos despejar a_{rs} :

$$a_{rs} = -a_{sj} + a_{kj} + a_{lk} - a_{lp} - \dots - a_{qr}$$

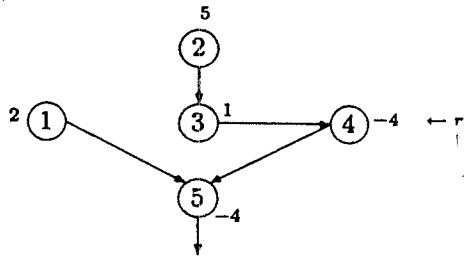
Hemos representado el arco no básico (r, s) en términos de arcos básicos. Resumiendo, primero se determina el ciclo único formado al añadir el arco no básico al árbol. Después se da al ciclo una orientación de acuerdo al arco no básico. Un arco básico en el ciclo tiene un coeficiente -1 si es directo con respecto a la orientación y el coeficiente es $+1$ si es indirecto con respecto a la orientación. Los otros arcos básicos fuera del ciclo tienen un coeficiente 0.

Una vez que se tiene una matriz básica, el cálculo de la solución básica factible se inicia de los vértices terminales hacia la raíz. La solución se calcula resolviendo la restricción de conservación de flujo utilizando los arcos del árbol.

Ejemplo 4. Considérese la siguiente red



Añadiendo el arco artificial del vértice 5, supóngase que se selecciona la base factible siguiente:



El arco (1,5) apunta hacia afuera del vértice 1 y, en consecuencia, x_{15} tiene un +1 en la restricción 1. Así,

$$x_{15} = r_1 = 2$$

Examinando el vértice 2, vemos que es un nodo terminal y por lo tanto, x_{23} puede calcularse de manera similar:

$$x_{23} = r_2 = 5$$

Después se observa que el nodo 3 tienen todas las variables de sus arcos incidentes con valores asignados excepto una. Luego, la ecuación de conservación puede usarse para el nodo 3 y resolver para la variable restante:

$$\begin{aligned} x_{34} - x_{23} &= r_3 = 1 \\ x_{34} &= 1 + 5 = 6 \end{aligned}$$

Ahora puede resolverse para x_{45} :

$$\begin{aligned} x_{45} - x_{34} &= r_4 = -4 \\ x_{45} &= -4 + 6 = 2 \end{aligned}$$

Finalmente se resuelve para x_5 :

$$\begin{aligned} x_5 - x_{15} - x_{45} &= r_5 = -4 \\ x_5 &= -4 + 2 + 2 = 0 \end{aligned}$$

Vamos ahora a presentar los pasos generales del método simplex aplicado a redes.

- (0) Supongamos que tenemos una solución básica factible inicial.
- (1) Calcular el vector dual $\mathbf{w} = (w_i)_{i \in N}$ de la manera siguiente:

Se empieza con la variable dual para el nodo raíz en el valor cero, y después se procede hacia afuera de la raíz, hacia las terminales del árbol usando la relación de que $w_i - w_j = c_{ij}$ a lo largo de los arcos básicos en el árbol.

- (2) Calcular para todo arco no básico

$$z_{ij} - c_{ij} = w_i - w_j - c_{ij}$$

- (a) Si para todo arco no básico (i, j) se tiene que $z_{ij} - c_{ij} \leq 0$. ALTO la solución es óptima.
- (b) Para los arcos que tienen $z_{ij} - c_{ij} > 0$. Escoger el arco (r, s) para el cual $z_{rs} - c_{rs} = \max \{z_{ij} - c_{ij}\}$. El arco (r, s) es candidato a entrar a la base.
- (3) Considerar el árbol junto con el arco no básico (r, s) . Asignar al ciclo formado una orientación de acuerdo a (r, s) . Hacer circular por el ciclo Δ unidades de flujo.

- En el arco (r, s) circulan Δ unidades.
- En los arcos básicos que son directos con respecto a la orientación, sumar Δ unidades a la solución.
- En los arcos básicos que son inversos con respecto a la orientación, restar Δ unidades a la solución.
- En los arcos básicos que no pertenecen al ciclo no alterar su solución.

(4) Sea Y el conjunto de arcos básicos a los que se les restó Δ unidades de flujo. Entonces

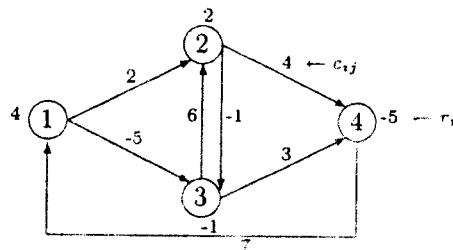
$$\Delta = x_{pq} = \min \{x_{ij} : a_{ij} \in Y\}$$

(Si hay más de uno, escoger solamente uno) la nueva solución básica factible será determinada, i.e., algunas variables básicas tendrán valor cero). El arco $(p, q) \in Y$ sale de la base.

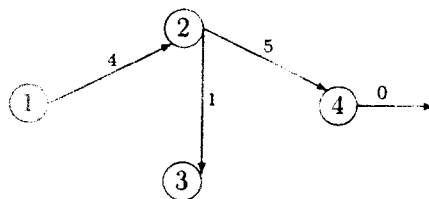
(5) Se tiene ahora una nueva solución básica factible al hacer las operaciones señaladas en el paso (3) con el valor de Δ calculado en (4). Ir a (1).

Ejemplo

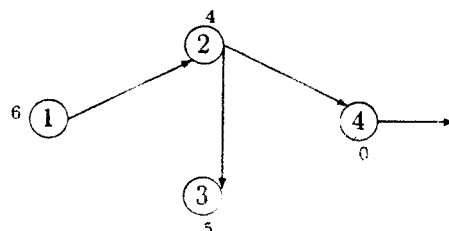
El siguiente ejemplo fue tomado del libro "Programación Lineal y flujo en Redes" de [1]. Encontrar el flujo a costo mínimo en la siguiente red:



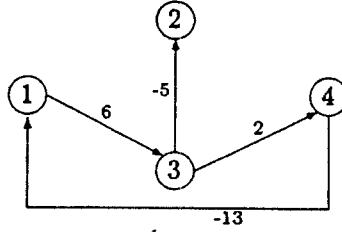
Primera iteración:
Solución primal:



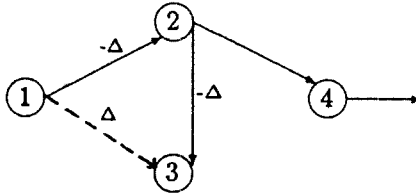
Solución dual:



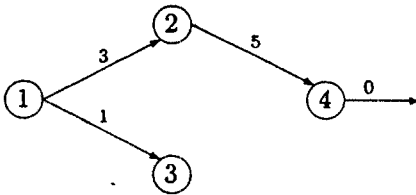
Cálculo de $z_{ij} - c_{ij}$ para los arcos no básicos:



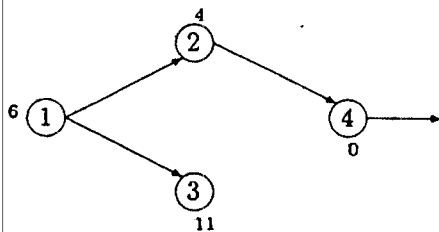
Pivoteamos



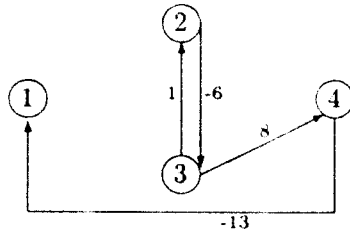
Segunda iteración:
Solución primal:



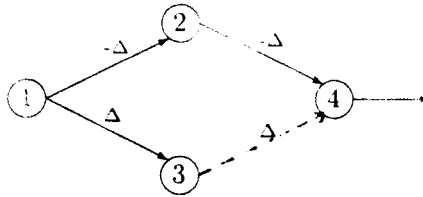
Solución dual:



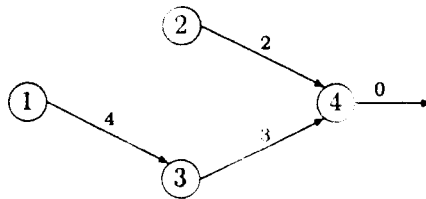
Cálculo de $z_{ij} - c_{ij}$ para los arcos no básicos:



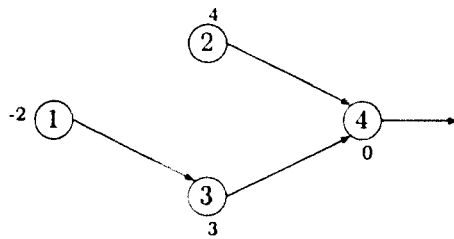
Pivoteamos



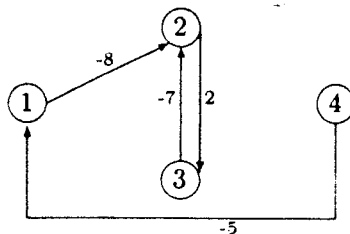
Tercera iteración:
Solución primal:



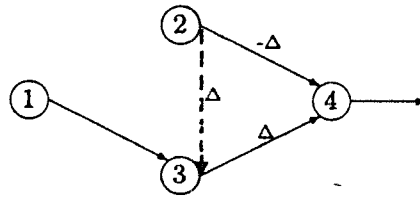
Solución dual:



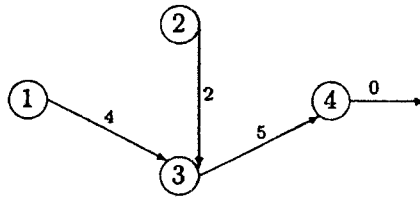
Cálculo de $z_{ij} - c_{ij}$ para los arcos no básicos:



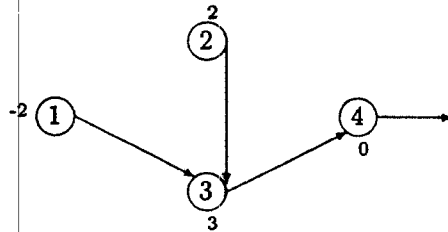
Pivoteamos



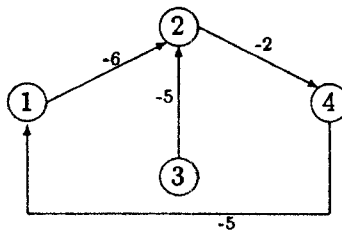
Cuarta iteración:
Solución primal:



Solución dual:



Cálculo de $z_{ij} - c_{ij}$ para los arcos no básicos



Como todos los arcos no básicos presentan que $z_{ij} - c_{ij} \leq 0$, se tiene ya, la solución óptima:
 $w_{\min} = -7$, $x_{13} = 4$, $x_{23} = 2$, $x_{34} = 5$.

Capítulo 2

Planteamiento del Problema

Se presentará ahora, el planteamiento y formulación matemática de la generalización del problema de flujo a costo mínimo, considerando en este caso, que el número de bienes que deben circular en la red es mayor que uno. El planteamiento del problema es similar al que resuelve el algoritmo simplex especializado en redes para un solo bien, desarrollado en el capítulo anterior.

Supóngase, por ejemplo, que un agricultor produce trigo, frijol y arroz, y desea distribuir estos productos a diferentes expendios y cada uno de éstos solicita diferentes cantidades de cada cereal. El agricultor desea satisfacer las demandas de los expendios, la disponibilidad con la que él cuenta, sin sobrepasar las capacidades de los camiones transportadores, de tal manera que el costo total de distribución sea mínimo. Éste es tan solo un ejemplo que puede formularse matemáticamente como un problema de flujo de bienes múltiples a costo mínimo. El planteamiento será descrito con dos formulaciones: la de nodo-arco y la de arco-cadena. basadas en Tomlin [29].

2.1 Formulación Nodo-Arco

Para la formulación de este problema supongamos que tenemos una red $G = (N, A)$ con n nodos y m arcos, cada arco (i, j) tiene una capacidad entera no negativa b_{ij} , el costo (no negativo) por unidad de flujo en el arco (i, j) es c_{ij} ; supongamos que en la red fluirán q bienes e identifiquemos, en N para cada bien k ($k = 1, \dots, q$), los conjuntos de vértices orígenes, transbordos y destinos, S_k , R_k y T_k respectivamente, y por último supongamos que cada bien k tiene un requerimiento entero r_i^k en el vértice i (que puede ser positivo si se trata de una disponibilidad o negativo si se trata de una demanda).

Observación 1. Algunos autores ([1], [3] y [6]) consideran que a cada bien le corresponde un costo diferente en un mismo arco, es decir, c_{ij}^k representa el costo (no negativo) por unidad de flujo del bien k en el arco (i, j) . Sin embargo no se pierde generalidad si suponemos que el costo en un arco es el mismo para todos los bienes.

Sea y_{ij}^k la cantidad de flujo del bien k en el arco (i, j) . Las restricciones del problema son las siguientes:

(i) La cantidad de flujo total en el arco (i, j) no debe sobrepasar la capacidad del arco, matemáticamente esta restricción se representa por:

$$\sum_{k=1}^q y_{ij}^k \leq b_{ij} \quad \forall (i, j) \in A \tag{2.1}$$

(ii) La cantidad de flujo de cada bien debe conservarse en los nodos, matemáticamente tenemos:

$$\sum_{j \in N} (y_{ij}^k - y_{ji}^k) = r_i^k \quad \forall k = \overline{1, q}, i \in N \tag{2.2}$$

donde r_i^k es tal que:

$$\begin{aligned} r_i^k &> 0 && \text{si } i \in S_k \\ r_i^k &= 0 && \text{si } i \in R_k \\ r_i^k &< 0 && \text{si } i \in T_k \end{aligned}$$

Para que el problema esté equilibrado supongamos que

$$\sum_{i \in N} r_i^k = 0 \quad \forall k$$

esto significa que la cantidad demandada para cada bien es precisamente la cantidad disponible existente.

Para cada k , $\sum_{j \in N} (y_{ij}^k - y_{ji}^k)$ puede representarse por $A_k y^k$, donde A_k es la matriz de incidencia nodo-arco.

Sean y_k el vector de flujo para cada k , d^k el vector obtenido del lado derecho de (2.2) y b el vector de capacidades, entonces las restricciones (2.1) y (2.2) pueden reescribirse de la siguiente manera:

$$\begin{aligned} Iy^1 + Iy^2 + \dots + Iy^q + Is &= b \\ A_1 y^1 &= d^1 \\ A_2 y^2 &= d^2 \\ &\vdots \\ A_q y^q &= d^q \\ y^k, s &\geq 0 \quad \forall k \end{aligned} \tag{2.3}$$

donde I es la matriz identidad de $m \times m$ y s es el vector de las variables de holgura, que se añaden para obtener la igualdad en las restricciones (2.1).

La función objetivo de este problema es:

$$\min w = c'y^1 + \dots + c'y^q$$

donde c' es el vector de costo.

De esta manera, la formulación matemática al problema de flujo de bienes múltiples a costo mínimo queda establecida por:

$$\left\{ \begin{aligned} \min w = & c'y^1 + c'y^2 + \dots + c'y^q \\ \text{s.a. } & Iy^1 + Iy^2 + \dots + Iy^q + Is = b \\ & A_1 y^1 = d^1 \\ & A_2 y^2 = d^2 \\ & \vdots \\ & A_q y^q = d^q \\ & y^k, s \geq 0 \quad \forall k \end{aligned} \right. \tag{2.4}$$

donde s es el m -vector asociado a las variables de holgura.

El problema tiene $(m + nq)$ restricciones y $(mq + m)$ variables, incluyendo las de holgura.

Adelantándonos un poco en el método de resolución, podemos observar que el problema tiene la estructura especial que plantean Dantzig y Wolfe para resolver algunos problemas lineales (véase el apéndice B), así pues, vamos a aplicar el primer paso de este principio, el cual consiste en hacer la siguiente transformación del problema:

Sea $W_k = \{w_{k1}, \dots, w_{kN_k}\}$ el conjunto de soluciones de puntos extremos del subsistema $A_k y^k = d^k$ donde N_k es el número de soluciones de puntos extremos para cada k .

Ahora, y^k que satisface $A_k y^k = d^k$, puede representarse como combinación convexa de los elementos de W_k

$$y^k = \sum_{j=1}^{N_k} \lambda_{kj} w_{kj}$$

donde

$$\sum_{j=1}^{N_k} \lambda_{kj} = 1$$

y

$$\lambda_{kj} \geq 0$$

Sustituyendo y^k en (2.4) tenemos el siguiente problema equivalente,

$$\left\{ \begin{array}{l} \min w = \sum_{j=1}^{N_1} \lambda_{1j} (c' w_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj} (c' w_{qj}) \\ \text{s.a.} \quad \sum_{j=1}^{N_1} \lambda_{1j} (I w_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj} (I w_{qj}) + I s = b \\ \sum_{j=1}^{N_1} \lambda_{1j} = 1 \\ \vdots \\ \sum_{j=1}^{N_q} \lambda_{qj} = 1 \\ \lambda_{kj}, s \geq 0 \quad j = \overline{1, N_k}, k = \overline{1, q} \end{array} \right. \quad (2.5)$$

Este problema tiene $m + q$ renglones y un número muy grande de variables, pues si para cada k tenemos N_k soluciones de puntos extremos, entonces se tienen $\sum_{k=1}^q N_k$ número de variables, sin embargo, esto no genera mayor dificultad, pues el principio de descomposición resuelve el problema trabajando con subproblemas de dimensiones mucho menores.

2.2 Formulación Arco-Cadena

Supongamos que tenemos una enumeración de los arcos e_1, \dots, e_m de la red, y sean $C_1^k, \dots, C_{N_k}^k$ todas las cadenas que unen los orígenes y los destinos, para un bien dado k .

Observación 2. El concepto de *cadena* debe interpretarse como *camino* si la red es dirigida, pues el flujo no puede atravesar un arco en sentido inverso.

Así, se propone el siguiente cambio de variable para establecer la equivalencia entre los dos problemas (I) y (II). Como x_j^k (en (II)) es el flujo del bien k en la cadena j , y $\lambda_{kj}w_{kj}$ es la cantidad de flujo del bien k en la cadena w_{kj} , entonces

$$x_j^k = \lambda_{kj}w_{kj}$$

Tomemos,

$$x_k = (x_1^k, \dots, x_{N_k}^k)^t = (\lambda_{k1}w_{k1}, \dots, \lambda_{kN_k}w_{kN_k})^t$$

Veamos la función objetivo:

$$\begin{aligned} \sum_{k=1}^q \sum_{j=1}^{N_k} \sum_{i=1}^m c_i a_{ij}^k x_j^k &= \sum_{k=1}^q \sum_{j=1}^{N_k} \sum_{i=1}^m c_i a_{ij}^k \lambda_{kj} w_{kj} \\ &= \sum_{k=1}^q \sum_{j=1}^{N_k} \lambda_{kj} w_{kj} \sum_{i=1}^m c_i a_{ij}^k \end{aligned}$$

$\sum_{i=1}^m c_i a_{ij}^k$ representa el costo de la cadena j del bien k .

Entonces,

$$\begin{aligned} \sum_{k=1}^q \sum_{j=1}^{N_k} \sum_{i=1}^m c_i a_{ij}^k x_j^k &= \sum_{k=1}^q \sum_{j=1}^{N_k} \lambda_{kj} w_{kj} c_j^k \\ &= \sum_{j=1}^{N_1} \lambda_{1j} (w_{1j} c_j^1) + \dots + \sum_{j=1}^{N_q} \lambda_{qj} (w_{qj} c_j^q) \end{aligned}$$

el costo es el mismo para todos los bienes, así pues, tenemos que

$$\sum_{k=1}^q \sum_{j=1}^{N_k} \sum_{i=1}^m c_i a_{ij}^k x_j^k = \sum_{j=1}^{N_1} \lambda_{1j} (c_j w_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj} (c_j w_{qj})$$

Como w_{kj} es una cadena y c_j es el costo de esta cadena, no se pierde generalidad si cambiamos c_j por c' (el vector de costos), pues $c'w_{kj} = c_j w_{kj}$.

Por lo tanto,

$$\sum_{k=1}^q \sum_{j=1}^{N_k} \sum_{i=1}^m c_i a_{ij}^k x_j^k = \sum_{j=1}^{N_1} \lambda_{1j} (c'w_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj} (c'w_{qj})$$

Y la función objetivo en ambos problemas es la misma.

Ahora, veamos las restricciones. El primer tipo de restricciones es de la forma:

$$A^1 x_1 + \dots + A^q x_q \leq b.$$

Fijémonos en un término cualquiera:

$$A^k x_k = \sum_{j=1}^{N_k} \sum_{i=1}^m a_{ij}^k x_j^k = \sum_{j=1}^{N_k} \sum_{i=1}^m a_{ij}^k (\lambda_{kj} w_{kj}) = \sum_{j=1}^{N_k} \lambda_{kj} w_{kj} \sum_{i=1}^m a_{ij}^k = \sum_{j=1}^{N_k} \lambda_{kj} (I w_{kj}).$$

Por lo tanto,

$$A^1 x_1 + \dots + A^q x_q = \sum_{j=1}^{N_1} \lambda_{1j} (I w_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj} (I w_{qj}) \leq b.$$

Las siguientes restricciones son de la forma:

$$u'_k x_k = r_k.$$

Entonces,

$$u'_k x_k = u'_k (\lambda_{k1} w_{k1}, \dots, \lambda_{kN_k} w_{kN_k})^t = \sum_{j=1}^{N_k} \lambda_{kj} w_{kj} = r_k.$$

Esto es la suma de los flujos en las cadenas del origen al destino del bien k , y λ_{kj} es la fracción de flujo para cada cadena, entonces debe suceder que

$$\sum_{j=1}^{N_k} \lambda_{kj} = 1.$$

Como $x_j^k \geq 0$, entonces $\lambda_{kj} w_{kj} \geq 0$ ahora, w_{kj} nunca será negativo, pues w_{kj} representa cadenas del origen al destino, así pues, $w_{kj} \geq 0$ y por lo tanto, $\lambda_{kj} \geq 0$. De esta manera queda establecido que las dos formulaciones son equivalentes.

Capítulo 3

Un Método de Resolución Basado en el Método de Dantzig-Wolfe

Es preciso plantear un método de resolución para el problema de flujo de bienes múltiples ya que los algoritmos que existen para el problema de flujo de un solo bien no pueden aplicarse. Estos algoritmos toman ventaja de la propiedad de total unimodularidad¹ que su matriz de restricciones presenta, la cual se pierde cuando se trata de varios bienes. Sin embargo, la matriz de restricciones del problema de flujo de bienes múltiples presenta una estructura especial que permite aplicar el principio de descomposición de Dantzig-Wolfe, por lo tanto, el método de resolución aquí presentado está basado precisamente en este principio, el cual está descrito en [1] y en [13], y en este trabajo se explica en el apéndice B.

El principio de descomposición aplicado al problema de flujo de bienes múltiples ha sido desarrollado por Bazarra [1] y por Tomlin [29]; en el primero se expone únicamente para la formulación nodo-arco, y en el segundo, se presenta para ambas formulaciones. El desarrollo del método aquí expuesto, está basado en [29] y los algoritmos propuestos están basados en el algoritmo presentado en [1].

3.1 Formulación Nodo-Arco

En el capítulo anterior, sección 2.1, vimos la formulación matemática del problema de flujo de bienes múltiples a costo mínimo y también la aplicación del primer paso del principio de descomposición de Dantzig-Wolfe. Así pues, vamos a presentar este método de resolución tomando el planteamiento que resultó de haber aplicado la transformación $y^k = \sum_{j=1}^{N_k} \lambda_{kj} w_{kj}$, ésta producía el problema siguiente:

¹Una matriz A es *totalmente unimodular* si cualquier submatriz cuadrada tiene determinante $+1$, -1 ó 0 . Esto permite garantizar que las soluciones al problema serán enteras, si los términos independientes del problema son enteros.

$$\left\{ \begin{array}{l} \min \varpi = \sum_{j=1}^{N_1} \lambda_{1j}(c'w_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj}(c'w_{qj}) \\ \text{s.a.} \quad \sum_{j=1}^{N_1} \lambda_{1j}(Iw_{1j}) + \dots + \sum_{j=1}^{N_q} \lambda_{qj}(Iw_{qj}) + Is = b \\ \sum_{j=1}^{N_1} \lambda_{1j} = 1 \\ \vdots \\ \sum_{j=1}^{N_q} \lambda_{qj} = 1 \\ \lambda_{kj}, s \geq 0 \quad j = \overline{1, N_k}, k = \overline{1, q} \end{array} \right. \quad (3.1)$$

Supongamos que tenemos una solución básica factible al problema transformado (3.1). Asociemos a los primeros m renglones las variables duales π_{ij} y a los q renglones restantes las variables duales $\alpha_1, \dots, \alpha_q$.

Sea $\pi' = (\pi_{ij})$ entonces el problema dual es:

$$\max z = b\pi' + \sum_{k=1}^q \alpha_k$$

$$\pi' Iw_{kj} + \alpha_k \leq c'w_{kj} \quad (3.2)$$

$$\pi' \leq 0 \quad (3.3)$$

$$\pi', \alpha_k \text{ s.t.s}$$

Observación 1. Dados estos dos problemas lineales, primal y dual, los coeficientes de costo reducido del problema primal son reflejados en el problema dual en su vector de términos independientes, por lo tanto, para una solución básica factible del primal, si alguna restricción del dual no se satisface, entonces en el problema primal, el coeficiente de costo reducido correspondiente es negativo y así, la variable asociada a él es candidata a entrar en la base. Si todas las restricciones del dual se satisfacen, entonces los costos reducidos del primal son no negativos, y por lo tanto, su solución es óptima.

Con base en esto, en el problema (3.1), la columna k, j puede entrar en la base si la k, j -ésima restricción del primer tipo de restricciones, (3.2), del problema dual no se satisface, es decir, si

$$\pi' Iw_{kj} + \alpha_k > c'w_{kj}$$

es decir,

$$c'w_{kj} - \pi' Iw_{kj} < \alpha_k$$

$$(c' - \pi')w_{kj} < \alpha_k \quad (3.4)$$

$(c' - \pi')w_{kj}$ puede considerarse como el costo del valor w_{kj} , tomando como $(c' - \pi')$ el costo asociado a los arcos de la red. Como w_{kj} es solución de punto extremo del sistema $A_k y^k = d^k$, w_{kj} representa el valor de un flujo del bien k .

Para que (3.4) sea cierto, es necesario que el costo mínimo sea menor que α_k , pues si así es, entonces cualquier otro costo satisfará (3.4), ya que si nos contentamos con cualquier costo que satisfaga (3.4), corremos el peligro de que algún otro valor de costo, en particular el mínimo, sea mayor que α_k . Por lo tanto, w_{kj} se encuentra al resolver el subproblema:

$$\left\{ \begin{array}{l} \min v = (c - \pi)'w_k \\ \text{s.a.} \quad A_k w_k = d^k \\ w_k \geq 0 \end{array} \right. \quad (3.5)$$

Una vez encontrado el valor de w_{kj} , se sustituye en (3.4) y se verifica que se satisfaga la desigualdad. Si es así, entonces la variable λ_{kj} es candidato a entrar en la base del problema (3.1). Y tenemos que el subproblema es un problema de flujo de un solo bien a costo mínimo.

Si al resolver todos los subproblemas se tiene que (3.4) no se satisface, entonces observamos si se verifica el segundo tipo de restricción (3.3) del problema dual. Esto quiere decir, que si algún π_{ij} es positivo entonces la variable de holgura correspondiente s_{ij} es candidato a entrar en la base del problema (3.1).

Si las restricciones duales se satisfacen entonces la solución es óptima, por el razonamiento visto en la observación anterior.

Para resolver el problema (3.1) partimos del supuesto de que conociamos de antemano una solución básica factible, sin embargo, no siempre es posible tener de inmediato una primera solución, por lo que podemos obtenerla aplicando el método simplex, es decir, añadiendo variables artificiales x_1, \dots, x_q a las últimas q restricciones del problema y minimizando la suma de éstas. Basta con añadir las variables artificiales a las últimas q restricciones, pues éstas son del tipo ecuación y sin embargo las primeras m son del tipo desigualdad, a las que se añadieron variables de holgura para obtener las igualdades.

Por lo tanto, tenemos el problema asociado:

$$\left\{ \begin{array}{llll} \min \varphi = & x_1 & + \dots & + x_q \\ \text{s.a.} & \sum_j \lambda_{1j} w_{1j} & + \dots & + \sum_j \lambda_{qj} w_{qj} + Is & = b \\ & \sum_j \lambda_{1j} & & & + x_1 & = 1 \\ & & \dots & & & \vdots \\ & & & \sum_j \lambda_{qj} & & + x_q & = 1 \\ & & & & & \lambda_{kj}, s, x_k & \geq 0 \end{array} \right. \quad (3.6)$$

La función objetivo puede escribirse en función de las variables λ_{kj} , sumando los últimos q renglones y despejando la suma de x_i , tenemos:

$$\min \varphi = x_1 + \dots + x_q = q - \sum_j \lambda_{1j} - \dots - \sum_j \lambda_{qj}$$

Para resolver este problema, utilizamos nuevamente el principio de descomposición de Dantzig-Wolfe, pero ahora tenemos que las variables artificiales y las de holgura (vector s) constituyen una solución básica inicial al problema (3.6).

Asociamos variables duales al problema asociado: σ_{ij} para los primeros m renglones y $\gamma_1, \dots, \gamma_q$ para los últimos q renglones. Sea $\sigma' = (\sigma_{ij})$; el problema dual queda planteado así:

$$\begin{aligned} \max z &= b\sigma' + \sum_{k=1}^q \gamma_k \\ \sigma' w_{kj} + \gamma_k &\leq -1 & (3.7) \\ \sigma' &\leq 0 & (3.8) \\ \gamma_k &\leq 0 & (3.9) \\ &\sigma', \gamma_k \text{ s.r.s.} \end{aligned}$$

Si la k, j -ésima restricción de (3.7) no se satisface, es decir, si

$$-1 - \sigma' w_{kj} - \gamma_k < 0$$

entonces la columna k, j puede entrar en la base; de manera análoga, es necesario conocer los valores σ', γ_k y w_{kj} para verificar esta restricción dual. σ' y γ_k los conocemos por ser variables duales, y w_{kj} lo obtenemos al resolver el subproblema:

$$\begin{cases} \min v = -\sigma w_k \\ A_k w_k = d^k \\ w_k \geq 0 \end{cases}$$

Si alguna restricción de (3.8) no se satisface, es decir, si algún σ_{ij} es positivo entonces la correspondiente variable de holgura s_{ij} entra en la base. Si alguna restricción de (3.9) no se satisface, es decir, si γ_k es positivo entonces la variable artificial correspondiente x_k puede entrar en la base nuevamente. Si todas las restricciones duales se satisfacen entonces la solución es óptima, y según sea ésta, se pasa o no a la fase II del método simplex.

Presentamos a continuación un algoritmo para la formulación nodo-arco, basado en el desarrollado en [1]; en la siguiente sección, destinada para la presentación del método de resolución para la formulación arco-cadena se expone también un algoritmo análogo a éste.

Algoritmo

(0) Sea w_{k1} una solución básica factible inicial del problema (3.1). Calcúlese la inversa de la matriz básica, los valores $c'w_{k1}$ para cada k y el vector $w_B = B^{-1}\hat{b}$, donde $\hat{b} = (b \ 1)^t$, y 1 es un q -vector con todas sus componentes iguales a 1. Sea $\varpi = \sum_{k=1}^q c'w_{k1}$. Sea (π, α) el vector de variables duales correspondientes a la solución básica factible.

(1) Fórmese el arreglo simplex revisado:

(π, α)	ϖ
B^{-1}	w_B

- (a) Si algún $\pi_{ij} > 0$ entonces la correspondiente s_{ij} es un candidato para entrar en la base. Ir a (3).
- (b) Si $\pi_{ij} \leq 0 \ \forall (i, j) \in A$, considérese el siguiente k -ésimo subproblema

$$\begin{cases} \min v = (c - \pi)'w_k \\ \text{s.a.} \quad A_k w_k = d^k \\ w_k \geq 0 \end{cases}$$

Este es un problema de flujo de un solo bien.

Sea \tilde{w}_{kj} la solución a este subproblema. Ir a (2).

(2) Calcule $M = c' \tilde{w}_{kj} - \pi' I \tilde{w}_{kj} - \alpha_k$

(a) Si $M < 0$ entonces λ_{kj} es un candidato para entrar en la base. Ir a (3).

(b) Si $M \geq 0 \quad \forall k = \overline{1, q}$ ALTO se dispone ya de la solución óptima.

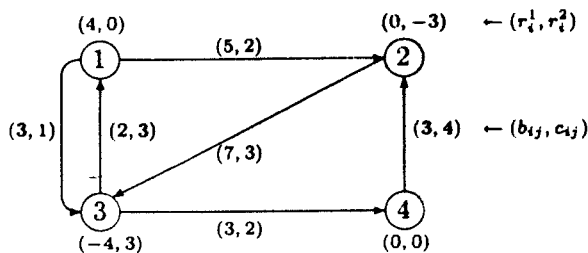
(3) Una vez identificada la variable candidato, se actualiza su columna de manera adecuada:

- $B^{-1} \begin{pmatrix} e_{ij} \\ 0 \end{pmatrix}$ si la variable candidato es s_{ij} , donde e_{ij} es un m -vector columna unitario con un 1 en el renglón asociado al arco (i, j) .
- $B^{-1} \begin{pmatrix} \tilde{w}_{kj} \\ e_k \end{pmatrix}$ si la variable candidato es λ_{kj} , donde e_k es un q -vector columna unitario con un 1 en el renglón asociado al bien k .

Se pivotea, tomando $-M$ el coeficiente de costo reducido asociado a λ_{kj} . Esto actualiza la inversa de la base, las variables duales y el lado derecho. Ir a (1).

3.1.1 Ejemplo

Consideremos la red siguiente para dos bienes.



El bien 1 tiene los nodos 1 y 3 como origen y destino, respectivamente, y el bien 2 tiene los nodos 3 como origen y 2 como destino.

Supongamos que empezamos con las siguientes soluciones factibles (esto para evitar el método de dos fases):

$$w_{11} = \begin{pmatrix} y_{12}^1 \\ y_{13}^1 \\ y_{23}^1 \\ y_{31}^1 \\ y_{34}^1 \\ y_{42}^1 \end{pmatrix} = \begin{pmatrix} 4 \\ 0 \\ 4 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad w_{21} = \begin{pmatrix} y_{12}^2 \\ y_{13}^2 \\ y_{23}^2 \\ y_{31}^2 \\ y_{34}^2 \\ y_{42}^2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 3 \end{pmatrix}$$

La solución de base consiste de las holguras, λ_{11} y λ_{21} . La matriz básica y su inversa son:

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 4 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$B^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -4 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -4 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$c'w_{11} = 20 \quad c'w_{21} = 18$$

$$\varpi = 20 + 18 = 38$$

$$w_B = B^{-1}\hat{b} = (1 \ 3 \ 3 \ 2 \ 0 \ 0 \ 1 \ 1)^t$$

Iteración 1.

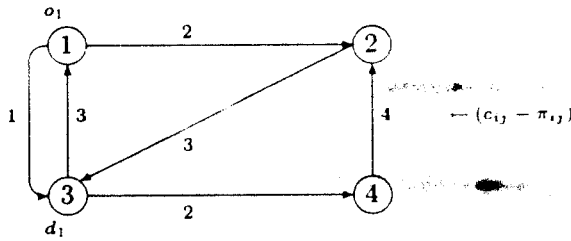
Formamos el arreglo simplex revisado:

(π, α)	ϖ
B^{-1}	w_B

	π_{12}	π_{13}	π_{23}	π_{31}	π_{34}	π_{42}	α_1	α_2	
ϖ	0	0	0	0	0	0	20	18	38
s_{12}	1	0	0	0	0	0	-4	0	1
s_{13}	0	1	0	0	0	0	0	0	3
s_{23}	0	0	1	0	0	0	-4	0	3
s_{31}	0	0	0	1	0	0	0	0	2
s_{34}	0	0	0	0	1	0	0	0	0
s_{42}	0	0	0	0	0	1	0	-3	0
λ_{11}	0	0	0	0	0	0	1	-3	1
λ_{21}	0	0	0	0	0	0	0	1	1

Primero observamos que todo $\pi_{ij} \leq 0$, verificamos entonces, si existe un candidato de cualquier subproblema para entrar en la base.

Subproblema 1



La solución óptima para este subproblema es

$$\tilde{w}_{12} = (0 \ 4 \ 0 \ 0 \ 0 \ 0)^t$$

Por lo tanto, calculamos el valor de \$M\$:

$$M = c' \tilde{w}_{12} - \pi' \tilde{w}_{12} - \alpha_1 = 4 - 0 - 20 = -16 < 0$$

Como es negativo, entonces \$\lambda_{12}\$ es candidato a entrar en la base.

Actualizamos la columna de \$\lambda_{12}\$

$$B^{-1} \begin{pmatrix} \tilde{w}_{12} \\ 1 \\ 0 \end{pmatrix} = (-4 \ 4 \ -4 \ 0 \ 0 \ 0 \ 1 \ 0)^t$$

Pivoteamos:

	\$\pi_{12}\$	\$\pi_{13}\$	\$\pi_{23}\$	\$\pi_{31}\$	\$\pi_{34}\$	\$\pi_{42}\$	\$\alpha_1\$	\$\alpha_2\$	
\$w\$	0	0	0	0	0	0	20	18	38
\$s_{12}\$	1	0	0	0	0	0	-4	0	1
\$s_{13}\$	0	1	0	0	0	0	0	0	3
\$s_{23}\$	0	0	1	0	0	0	-4	0	3
\$s_{31}\$	0	0	0	1	0	0	0	0	2
\$s_{34}\$	0	0	0	0	1	0	0	0	0
\$s_{42}\$	0	0	0	0	0	1	0	-3	0
\$\lambda_{11}\$	0	0	0	0	0	0	1	-3	1
\$\lambda_{21}\$	0	0	0	0	0	0	0	1	1

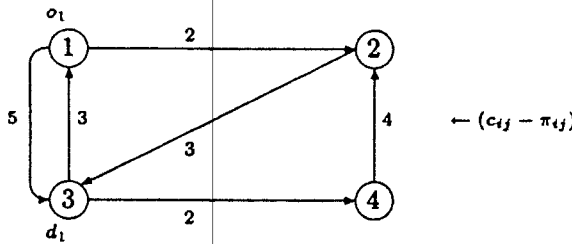
16
-4
4
-4
0
0
0
1
0

Iteración 2.

	π_{12}	π_{13}	π_{23}	π_{31}	π_{34}	π_{42}	α_1	α_2	
w	0	-4	0	0	0	0	20	18	26
s_{12}	1	1	0	0	0	0	-4	0	4
λ_{12}	0	1/4	0	0	0	0	0	0	3/4
s_{23}	0	0	1	0	0	0	-4	0	6
s_{31}	0	0	0	1	0	0	0	0	2
s_{34}	0	0	0	0	1	0	0	0	0
s_{42}	0	0	0	0	0	1	0	-3	0
λ_{11}	0	-1/4	0	0	0	0	1	-3	1/4
λ_{21}	0	0	0	0	0	0	0	1	1

Todo $\pi_{ij} \leq 0$. Verificamos los candidatos de los subproblemas:

Subproblema 1.



El óptimo para este subproblema es

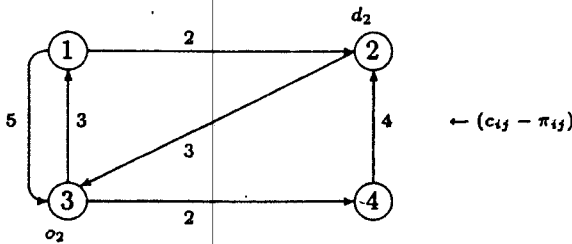
$$\tilde{w}_{13} = (0 \ 4 \ 0 \ 0 \ 0 \ 0)^t$$

Calculamos el valor de M :

$$M = c' \tilde{w}_{13} - \pi' \tilde{w}_{13} - \alpha_1 = 4 + 16 - 20 = 0 \geq 0$$

Como es no negativo, no hay candidato del subproblema 1.

Subproblema 2.



El óptimo para este subproblema es

$$\tilde{w}_{22} = (3 \ 0 \ 0 \ 3 \ 0 \ 0)^t$$

Calculamos el valor de M :

$$M = c' \tilde{w}_{22} - \pi' \tilde{w}_{22} - \alpha_2 = 15 - 0 - 18 = -3 < 0.$$

Como es negativo, λ_{22} es candidato a entrar en la base. Actualizamos su columna:

$$B^{-1} \begin{pmatrix} \tilde{w}_{22} \\ 0 \\ 1 \end{pmatrix} = (3 \ 0 \ 0 \ 3 \ -3 \ -3 \ 0 \ 1)^t$$

Pivoteamos:

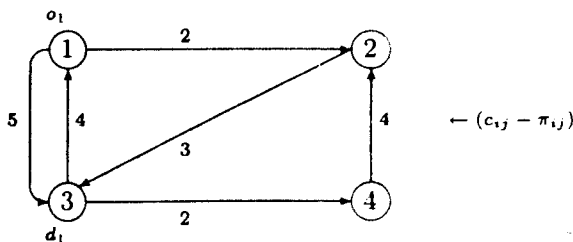
	π_{12}	π_{13}	π_{23}	π_{31}	π_{34}	π_{42}	α_1	α_2		
ω	0	-4	0	0	0	0	20	18	26	3
s_{12}	1	1	0	0	0	0	-4	0	4	3
λ_{12}	0	1/4	0	0	0	0	0	0	3/4	0
s_{23}	0	0	1	0	0	0	-4	0	6	0
s_{31}	0	0	0	1	0	0	0	0	2	3
s_{34}	0	0	0	0	1	0	0	0	0	-3
s_{42}	0	0	0	0	0	1	0	-3	0	-3
λ_{11}	0	-1/4	0	0	0	0	1	-3	1/4	0
λ_{21}	0	0	0	0	0	0	0	1	1	1

Iteración 3.

	π_{12}	π_{13}	π_{23}	π_{31}	π_{34}	π_{42}	α_1	α_2		
ω	0	-4	0	-1	0	0	20	18	24	
s_{12}	1	1	0	-1	0	0	-4	0	2	
λ_{12}	0	1/4	0	0	0	0	0	0	3/4	
s_{23}	0	1	1	0	0	0	-4	0	6	
λ_{22}	0	0	0	1/3	0	0	0	0	2/3	
s_{34}	0	0	0	1	1	0	0	0	2	
s_{42}	0	0	0	1	0	1	0	-3	2	
λ_{11}	0	-1/4	0	0	0	0	1	-3	1/4	
λ_{21}	0	0	0	-1/3	0	0	0	1	1/3	

Como todo $\pi_{ij} \leq 0$ ningún s_{ij} es candidato a entrar en la base.

Subproblema 1.



El óptimo para este subproblema es

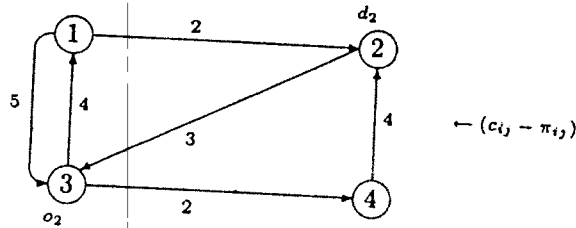
$$\tilde{w}_{14} = (0 \ 4 \ 0 \ 0 \ 0 \ 0)^t$$

El valor de M está dado por:

$$M = c' \tilde{w}_{14} - \pi' \tilde{w}_{14} - \alpha_1 = 4 + 16 - 20 = 0 \geq 0$$

Como es no negativo, no hay candidato del subproblema 1.

Subproblema 2.



El óptimo para este subproblema es

$$\tilde{w}_{23} = (3 \ 0 \ 0 \ 3 \ 0 \ 0)^t$$

Para este caso, M tiene un valor de:

$$M = c' \tilde{w}_{23} - \pi' \tilde{w}_{23} - \alpha_2 = 15 + 3 - 18 = 0 \geq 0$$

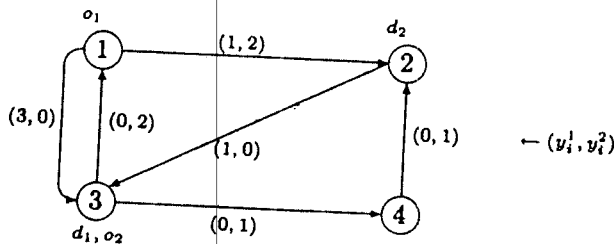
Como es no negativo, no hay candidato del subproblema 2.

Como para todo k se tiene que $M \geq 0$, se cuenta ya con la solución óptima:

$$\varpi_{\min} = 24$$

$$\begin{aligned} y^1 &= \lambda_{11} \tilde{w}_{11} + \lambda_{12} \tilde{w}_{12} \\ &= \frac{1}{4} (4 \ 0 \ 4 \ 0 \ 0 \ 0)^t + \frac{3}{4} (0 \ 4 \ 0 \ 0 \ 0 \ 0)^t \\ &= (1 \ 3 \ 1 \ 0 \ 0 \ 0)^t \end{aligned}$$

$$\begin{aligned} y^2 &= \lambda_{21} \tilde{w}_{21} + \lambda_{22} \tilde{w}_{22} \\ &= \frac{1}{3} (0 \ 0 \ 0 \ 0 \ 3 \ 3)^t + \frac{2}{3} (3 \ 0 \ 0 \ 3 \ 0 \ 0)^t \\ &= (2 \ 0 \ 0 \ 2 \ 1 \ 1)^t \end{aligned}$$



Algoritmo

(0) Sea x^k una solución básica factible inicial del problema (3.10). Calcúlese la inversa de la matriz básica, los valores $c' A^k x^k$ para cada k y el vector $x_B = B^{-1} \hat{b}$, donde $\hat{b} = (b \quad r^k)^t$, así como los valores $\alpha_k = \frac{c' A^k x^k}{r^k}$. Sea $w = \sum_{k=1}^q c' x^k$. Sea (π, α) el vector de variables duales correspondientes a la solución básica factible.

(1) Fórmese el arreglo simplex revisado:

(π, α)	w
B^{-1}	w_B

(a) Si algún $\pi_i > 0$, entonces la correspondiente s_i es un candidato para entrar en la base. Ir a (3).

(b) Si $\pi_i \leq 0 \quad \forall i = \overline{1, m}$, considérese el k -ésimo subproblema que consiste en encontrar la cadena más corta tomando como distancias los valores $(c_i - \pi_i)$ para cada arco.

Sea C_j^k la cadena más corta para el subproblema k . Ir a (2).

(2) Sea $M = \sum_{i=1}^m (c_i - \pi_i) a_{ij}^k - \alpha_k$, donde a_{ij}^k son las entradas de la cadena C_j^k en la matriz de incidencia arco-cadena.

(a) Si $M < 0$, entonces x_j^k es candidato a entrar en la base. Ir a (3).

(b) Si $M \geq 0 \quad \forall k$, ALTO se dispone ya de la solución óptima.

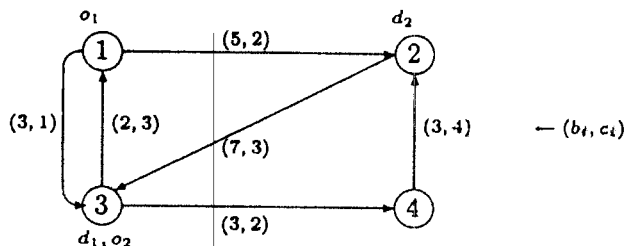
(3) Identificar la variable candidato, se actualiza su columna de manera adecuada:

- $B^{-1} \begin{pmatrix} \mathbf{v}_i \\ \mathbf{0} \end{pmatrix}$ si la variable candidato es s_i , donde \mathbf{v}_i es un m -vector unitario con un 1 en el renglón asociado al arco i .
- $B^{-1} \begin{pmatrix} a_{ij}^k \\ \mathbf{v}_k \end{pmatrix}$ si la variable candidato es x_j^k , donde \mathbf{v}_k es un q -vector unitario con un 1 en el renglón asociado al bien k .

Se pivotea, tomando $-M$ el coeficiente de costo reducido asociado a x_j^k . Esto actualiza la inversa de la base, las variables duales y el lado derecho. Ir a (1).

3.2.1 Ejemplo

Consideremos la red del ejemplo 1. Donde el bien 1 tiene un requerimiento de 4 unidades y el bien 2 de 3 unidades.



Establezcamos la siguiente enumeración de los arcos:

arco	número
(1, 2)	1
(1, 3)	2
(2, 3)	3
(3, 1)	4
(3, 4)	5
(4, 2)	6

Supongamos que comenzamos con la siguiente solución factible:

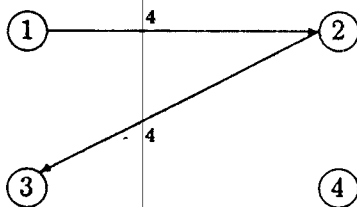
$$x_1^1 = 4, x_1^2 = 3$$

donde, x_1^1 es la cantidad de flujo del bien 1 en la cadena 1, la cual está descrita por

$$C_1^1 = \{1, (1, 2), 2, (2, 3), 3\}$$

y cuya representación en la matriz de incidencia es la columna

$$a_{i1}^1 = (1, 0, 1, 0, 0, 0)^t$$

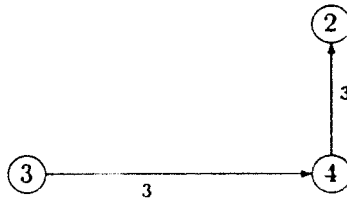


Análogamente, x_1^2 es la cantidad de flujo del bien 2 en la cadena 1:

$$C_1^2 = \{3, (3, 4), 4, (4, 2), 2\}$$

representada en la matriz de incidencia por la columna

$$a_{i1}^2 = (0, 0, 0, 0, 1, 1)^t$$



La solución de base consiste en las variables de holgura, de x_1^1 y de x_1^2 . La inversa de la matriz básica es:

$$B^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$c'A^1x^1 = 20$$

$$c'A^2x^2 = 18$$

donde $x^k = (x_j^k)$

$$x_B = B^{-1}\hat{b} = (1, 3, 3, 2, 0, 0, 4, 3)^t$$

$$\varpi = 20 + 18 = 38$$

Observe que para esta solución básica factible, tenemos que los valores de la variables duales son:

$$\pi_i = 0 \quad \forall i$$

$$\alpha_1 = \frac{c'A^1x^1}{r_1} = \frac{20}{4} = 5$$

$$\alpha_2 = \frac{c'A^2x^2}{r_2} = \frac{18}{3} = 6$$

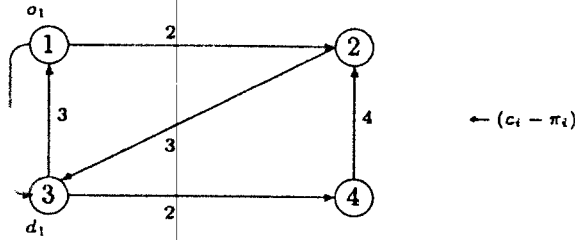
Iteración 1.

Formamos el arreglo simplex revisado:

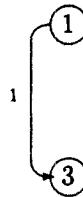
	π_1	π_2	π_3	π_4	π_5	π_6	α_1	α_2	
ϖ	0	0	0	0	0	0	5	6	38
s_1	1	0	0	0	0	0	-1	0	1
s_2	0	1	0	0	0	0	0	0	3
s_3	0	0	1	0	0	0	-1	0	3
s_4	0	0	0	1	0	0	0	0	2
s_5	0	0	0	0	1	0	0	-1	0
s_6	0	0	0	0	0	1	0	-1	0
x_1^1	0	0	0	0	0	0	1	0	4
x_1^2	0	0	0	0	0	0	0	1	3

Como $\pi_i \leq 0$ ningún s_i es candidato a entrar en la base.

Subproblema 1.



El camino más corto de 1 a 3 es



$$a_{i2}^1 = (0, 1, 0, 0, 0, 0)^t$$

El valor de M está dado por:

$$M = \sum_{i=1}^6 c_i a_{i2}^1 - \sum_{i=1}^6 \pi_i a_{i2}^1 - \alpha_1 = 1 - 0 - 5 = -4 < 0$$

Como es negativo, entonces x_2^1 entra en la base. Actualizamos su columna:

$$B^{-1} \begin{pmatrix} a_{i2}^1 \\ v_1 \end{pmatrix} = (-1, 1, -1, 0, 0, 0, 1, 0)^t$$

Pivoteamos

	π_1	π_2	π_3	π_4	π_5	π_6	α_1	α_2	
ω	0	0	0	0	0	0	5	6	38
s_1	1	0	0	0	0	0	-1	0	1
s_2	0	1	0	0	0	0	0	0	3
s_3	0	0	1	0	0	0	-1	0	3
s_4	0	0	0	1	0	0	0	0	2
s_5	0	0	0	0	1	0	0	-1	0
s_6	0	0	0	0	0	1	0	-1	0
x_1^1	0	0	0	0	0	0	1	0	4
x_2^1	0	0	0	0	0	0	0	1	3

4

-1

1

-1

0

0

1

0

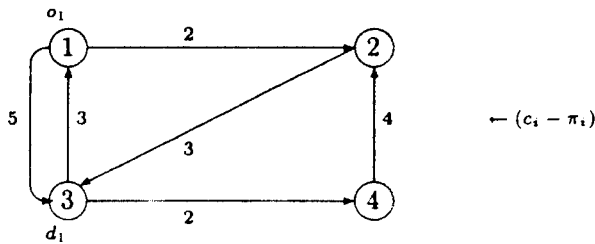
0

Iteración 2.

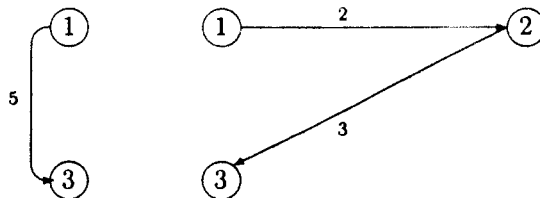
	π_1	π_2	π_3	π_4	π_5	π_6	α_1	α_2	
\bar{w}	0	-4	0	0	0	0	5	6	26
s_1	1	1	0	0	0	0	-1	0	4
x_2^1	0	1	0	0	0	0	0	0	3
s_3	0	1	1	0	0	0	-1	0	6
s_4	0	0	0	1	0	0	0	0	2
s_5	0	0	0	0	1	0	0	-1	0
s_6	0	0	0	0	0	1	0	-1	0
x_1^1	0	-1	0	0	0	0	1	0	1
x_1^2	0	0	0	0	0	0	0	1	3

Como $\pi_i \leq 0 \quad \forall i$ entonces ningún s_i puede entrar en la base.

Subproblema 1.



Los caminos más cortos de 1 a 3 son



$$a_{i3}^1 = (0, 1, 0, 0, 0, 0)^t \quad a_{i4}^1 = (1, 0, 1, 0, 0, 0)^t$$

Ambos caminos tienen una longitud mínima igual a 5, así pues, para ambos se tiene que

$$M = 5 - 5 = 0$$

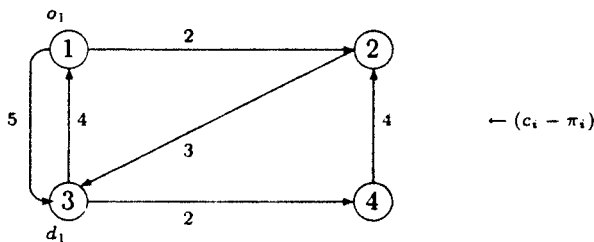
Por lo tanto, no hay candidato del subproblema 1.

Iteración 3.

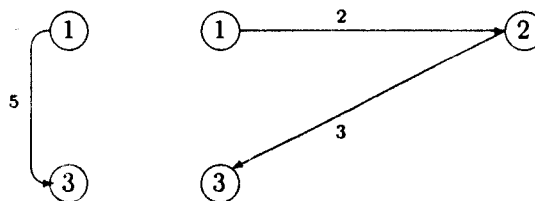
	π_1	π_2	π_3	π_4	π_5	π_6	α_1	α_2	
ϖ	0	-4	0	-1	0	0	5	6	24
s_1	1	1	0	-1	0	0	-1	0	2
x_2^1	0	1	0	0	0	0	0	0	3
s_3	0	1	1	0	0	0	-1	0	6
x_2^2	0	0	0	1	0	0	0	0	2
s_5	0	0	0	1	1	0	0	-1	0
s_6	0	0	0	1	0	1	0	-1	0
x_1^1	0	-1	0	0	0	0	1	0	1
x_1^2	0	0	0	-1	0	0	0	1	1

$\pi_i \leq 0 \quad \forall i$, así ningún s_i es candidato a entrar en la base.

Subproblema 1.



Los caminos más cortos de 1 a 3 son



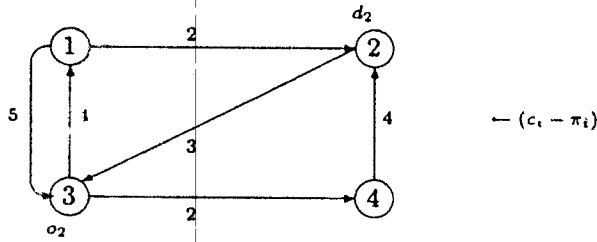
$$a_{i5}^1 = (0, 1, 0, 0, 0, 0)^t \quad a_{i6}^1 = (1, 0, 1, 0, 0, 0)^t$$

Ambos tienen una longitud mínima igual a 5, así pues, para ambos tenemos que

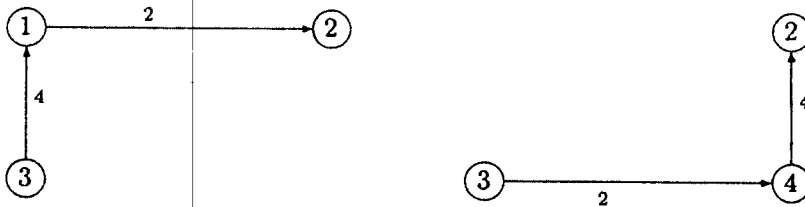
$$M = 5 - 5 = 0$$

Por lo tanto, no hay candidato del subproblema 1.

Subproblema 2.



Los caminos más cortos de 3 a 2 son



$$a_{i3}^2 = (1, 0, 0, 1, 0, 0)^t \quad a_{i4}^2 = (0, 0, 0, 0, 1, 1)^t$$

Ambos tienen una longitud mínima igual a 6, así pues, para ambos tenemos que

$$M = 6 - 6 = 0$$

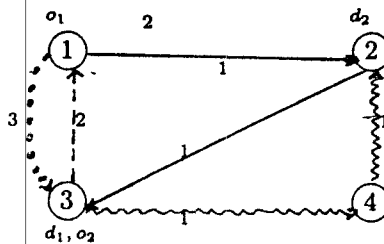
Por lo tanto, no hay candidato del subproblema 2.

Como $M \geq 0$ para todo k se tiene que la solución es óptima.

$$\omega_{\min} = 24$$

$$x^1 = (x_1^1, x_2^1) = (1, 3)$$

$$x^2 = (x_1^2, x_2^2) = (1, 2)$$



$$\begin{array}{ll} x_1^1 : \text{—————} & x_2^1 : \text{.....} \\ x_1^2 : \text{~~~~~} & x_2^2 : \text{-----} \end{array}$$

Capítulo 4

Planteamiento y Resolución de Dos Casos Particulares

El estudio de casos particulares siempre será importante en cualquier tipo de problemas que se presenten, pues precisamente por la estructura especial que tengan, puede hacerse un desarrollo específico y presentarse resultados para ellos. Es por esto que en este trabajo se presentan dos casos particulares del problema de flujo de bienes múltiples a costo mínimo; en ellos se analiza la estructura que tienen y al tomar ventaja de ella se demostrará que estos problemas particulares son equivalentes a problemas de flujo a costo mínimo de un solo bien. El presente capítulo está basado en [3] y en [4].

4.1 Problema de Transporte de Bienes Múltiples (Propiedad Topológica)

Se remite al lector al apéndice C donde se hace un repaso del planteamiento del problema de transporte de un solo bien.

Evans en [3] establece que una condición para que el problema de transporte de bienes múltiples pueda ser transformado en un problema de transporte de un solo bien con capacidades en los arcos, es que el número de orígenes o destinos no exceda de dos.

Primero presentamos el planteamiento del problema de transporte de bienes múltiples.

Supongamos que tenemos una red de transporte (no se consideran vértices intermedios o de transbordo) con m orígenes y n destinos en la que fluirán q número de bienes.

Sea x_{ij}^k la cantidad de flujo del bien k en el arco (i, j) , c_{ij}^k el costo por unidad de flujo del bien k en el arco (i, j) , a_i^k la disponibilidad del bien k en el origen i , b_j^k la demanda del bien k en el destino j , y u_{ij} la capacidad del arco (i, j) .

Denotaremos un problema de transporte de bienes múltiples con m orígenes, n destinos y q bienes como PTBM(m, n, q). La formulación matemática es:

$$\left\{ \begin{array}{l} \min z = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^q c_{ij}^k x_{ij}^k \\ \text{s.a.} \quad \sum_{j=1}^n x_{ij}^k = a_i^k \quad \forall i, k \\ \sum_{i=1}^m x_{ij}^k = b_j^k \quad \forall j, k \\ \sum_{k=1}^q x_{ij}^k + s_{ij} = u_{ij} \quad \forall i, j \\ x_{ij}^k, s_{ij} \geq 0 \quad \forall i, j, k \end{array} \right.$$

donde s_{ij} son variables de holgura.

El problema de transporte de un solo bien con capacidades en los arcos, con m orígenes y n destinos se denotará por $PTC(m, n)$ y su formulación matemática es:

$$\left\{ \begin{array}{l} \min w = \sum_{i=1}^m \sum_{j=1}^n h_{ij} y_{ij} \\ \text{s.a.} \quad \sum_{j=1}^n y_{ij} = e_i \quad \forall i \\ \sum_{i=1}^m y_{ij} = d_j \quad \forall j \\ y_{ij} + t_{ij} = v_{ij} \quad \forall i, j \\ y_{ij}, t_{ij} \geq 0 \quad \forall i, j \end{array} \right.$$

donde y_{ij} es la cantidad de flujo en el arco (i, j) , h_{ij} es el costo por unidad de flujo en el arco (i, j) , e_i es la disponibilidad en el origen i , d_j es la demanda en el destino j , v_{ij} es la capacidad del arco (i, j) y t_{ij} son las variables de holgura.

Por tratarse de problemas de transporte, podemos suponer que la suma de las disponibilidades es igual a la suma de las demandas, es decir,

$$\sum_{i=1}^m a_i^k = \sum_{j=1}^n b_j^k \quad \forall k$$

y

$$\sum_{i=1}^m e_i = \sum_{j=1}^n d_j$$

Como ya se mencionó anteriormente, la condición para que el $PTBM(m, n, q)$ pueda transformarse en un problema de transporte de un solo bien es que $m \leq 2$ o $n \leq 2$. Los dos casos son análogos, el resultado en este trabajo está presentado para el caso en que $n = 2$.

Teorema 1. *Todo $PTBM(m, 2, q)$ es equivalente al $PTC(m, q + 1)$.*

Es decir, cualquier problema de transporte de bienes múltiples de m orígenes, 2 destinos y q bienes, puede transformarse en un problema de transporte de un solo bien con m orígenes y $(q + 1)$ destinos.

Por "equivalente" se entenderá una correspondencia uno-a-uno entre los conjuntos de variables $\{x_{ij}^k, s_{ij}\}$ y $\{y_{ij}, t_{ij}\}$ de tal manera que las restricciones de un problema puedan obtenerse a partir de las restricciones del otro.

Demostración: Evans propone una demostración constructiva, es decir, se formulará el $PTBM(m, 2, q)$ como un $PTC(m, q + 1)$ y se dará una correspondencia uno-a-uno entre las variables.

Observemos que cada origen tiene únicamente dos opciones para enviar las disponibilidades de los bienes. Así, si se envían p unidades del bien k del origen i al destino 1, es claro que el resto (si lo hay) de la disponibilidad ($a_i^k - p$) se envía al destino 2 y viceversa. Así pues, si satisfacemos primero las demandas del destino 2, las del destino 1 quedarán satisfechas de manera inmediata.

Con esta observación construyamos una red de transporte de la siguiente manera:

Consideremos los m orígenes de la red original y construyamos q destinos, de cada origen salen q arcos con capacidades a_i^k , de tal manera que cada arco con capacidad a_i^k está dirigido del origen i al destino k , esto podemos ilustrarlo en la figura 1:

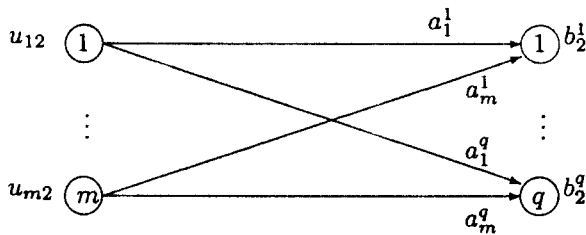


Figura 1.

Cada origen tendrá una disponibilidad u_{i2} y cada destino una demanda b_2^k , esto es porque queremos satisfacer las demandas del destino 2, así si por el arco (i, k) fluye una cantidad p ($p \leq a_i^k$) querrá decir que p unidades de la disponibilidad a_i^k van a dar al destino 2 y el resto al destino 1.

Si y_{ik} es la cantidad de flujo en el arco (i, k) y t_{ik} la holgura, entonces podemos establecer la relación siguiente:

$$\begin{aligned} y_{ik} &= x_{i2}^k - \\ t_{ik} &= x_{i1}^k \end{aligned}$$

Ahora, si añadimos un destino más y m arcos de los m orígenes al destino $q + 1$, este destino tendrá una demanda de $\sum_{i=1}^m u_{i2} - \sum_{k=1}^q b_2^k$ y las capacidades de estos arcos serán $u_{i1} + u_{i2} - \sum_{k=1}^q a_i^k$.

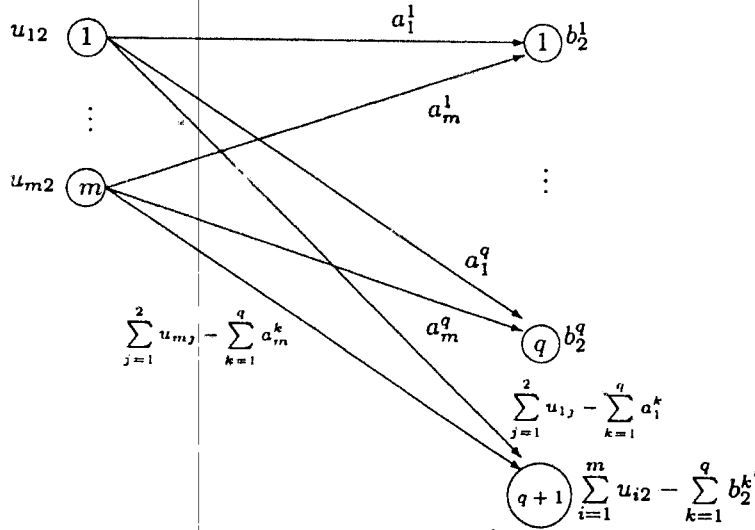


Figura 2.

Así, si un arco $(i, q+1)$ lleva un flujo de p unidades, entonces la holgura en el arco $(i, 2)$ será de p unidades y la holgura en el arco $(i, q+1)$ representa la holgura del arco $(i, 1)$. Por lo tanto, podemos establecer la relación:

$$y_{i,q+1} = s_{i2}$$

$$t_{i,q+1} = s_{i1}$$

Así hemos establecido una correspondencia uno-a-uno entre los conjuntos de variables.

Matemáticamente el problema representado en la red de la figura 2 debe cumplir las siguientes restricciones:

$$\sum_{k=1}^{q+1} y_{ik} = u_{i2} \quad \forall i = \overline{1, m} \quad (4.1)$$

$$\sum_{i=1}^m y_{ik} = \sum_{i=1}^m a_i^k - b_1^k = b_2^k \quad \forall k = \overline{1, q} \quad (4.2)$$

$$\sum_{i=1}^m y_{i,q+1} = \sum_{i=1}^m u_{i2} - \sum_{k=1}^q b_2^k \quad (4.3)$$

$$y_{ik} + t_{ik} = a_i^k \quad \forall i = \overline{1, m}, k = \overline{1, q} \quad (4.4)$$

$$y_{i,q+1} + t_{i,q+1} = \sum_{j=1}^2 u_{ij} - \sum_{k=1}^q a_i^k \quad \forall i = \overline{1, m} \quad (4.5)$$

donde las ecuaciones (4.1) a (4.5) determinan las restricciones de un PTC($m, q+1$), con las disponibilidades, demandas y capacidades de arcos expresadas en función de los parámetros del PTBM($m, 2, q$). Es decir, hemos obtenido las restricciones del PTC($m, q+1$) a partir de las restricciones del PTBM($m, 2, q$).

Ahora, con las correspondencias anteriormente establecidas vamos a obtener de (4.1) a (4.5) las restricciones del PTBM($m, 2, q$). Sustituyendo en (4.1):

$$\sum_{k=1}^q x_{i2}^k + s_{i2} = u_{i2} \quad \forall i = \overline{1, m} \quad (4.6)$$

Sustituyendo en (4.2):

$$\sum_{i=1}^m x_{i2}^k = b_2^k \quad \forall k = \overline{1, q} \quad (4.7)$$

Sustituyendo en (4.3):

$$\sum_{i=1}^m s_{i2} = \sum_{i=1}^m u_{i2} - \sum_{k=1}^q b_2^k \quad (4.8)$$

Sustituyendo en (4.4):

$$x_{i2}^k + x_{i1}^k = a_i^k \quad \forall i = \overline{1, m}, k = \overline{1, q} \quad (4.9)$$

Sustituyendo en (4.5):

$$s_{i2} + s_{i1} = \sum_{j=1}^2 u_{ij} - \sum_{k=1}^q a_i^k \quad \forall i = \overline{1, m} \quad (4.10)$$

Ahora, sumando (4.9) sobre i y restando (4.7):

$$\begin{aligned} \sum_i x_{i2}^k + \sum_i x_{i1}^k &= \sum_i a_i^k \\ - \sum_{i=1}^m x_{i2}^k &= -b_2^k \end{aligned}$$

Obtenemos:

$$\sum_{i=1}^m x_{i1}^k = \sum_i a_i^k - b_2^k = b_1^k \quad (4.11)$$

Restando (4.10) de (4.6):

$$\begin{aligned} \sum_{k=1}^q x_{i2}^k + s_{i2} &= u_{i2} \\ -s_{i1} - s_{i2} &= -u_{i1} - u_{i2} + \sum_{k=1}^q a_i^k \end{aligned}$$

Obtenemos:

$$\sum_{k=1}^q x_{i2}^k - s_{i1} = -u_{i1} + \sum_{k=1}^q a_i^k$$

despejando $\sum_{k=1}^q x_{i2}^k$:

$$\sum_{k=1}^q x_{i2}^k = s_{i1} - u_{i1} + \sum_{k=1}^q a_i^k \quad (4.12)$$

Sumando sobre k (4.9), se tiene

$$\sum_{k=1}^q x_{i2}^k + \sum_{k=1}^q x_{i1}^k = \sum_{k=1}^q a_i^k \quad (4.13)$$

Sustituyendo (4.12) en (4.13):

$$(s_{i1} - u_{i1} + \sum_{k=1}^q a_i^k) + \sum_{k=1}^q x_{i1}^k = \sum_{k=1}^q a_i^k$$

Obtenemos:

$$\sum_{k=1}^q x_{i1}^k + s_{i1} = u_{i1} \quad (4.14)$$

Si tomamos las ecuaciones (4.6), (4.7), (4.9), (4.11), y (4.14), adicionalmente con las restricciones de no negatividad, obtenemos el siguiente conjunto de restricciones:

$$\left\{ \begin{array}{l} \sum_{k=1}^q x_{i2}^k + s_{i2} = u_{i2} \quad \forall i = \overline{1, m} \\ \sum_{i=1}^m x_{i2}^k = b_2^k \quad \forall k = \overline{1, q} \\ \sum_{i=1}^m x_{i1}^k + x_{i2}^k = a_i^k \quad \forall i = \overline{1, m}, k = \overline{1, q} \\ \sum_{i=1}^m x_{i1}^k = b_1^k \\ \sum_{k=1}^q x_{i1}^k + s_{i1} = u_{i1} \\ x_{ij}^k, s_{ij} \geq 0 \end{array} \right.$$

que son precisamente las restricciones del PTBM($m, 2, q$) obtenidas a partir de las restricciones del PTC($m, q + 1$).

Veamos ahora la relación entre las funciones objetivo. La función objetivo para el PTC($m, q + 1$) es:

$$\min w = \sum_{i=1}^m \sum_{j=1}^{q+1} h_{ij} y_{ij}$$

Y tenemos la correspondencia: $y_{ij} = x_{i2}^j$ (para $j = \overline{1, q}$) y $y_{i, q+1} = s_{i2}$, por lo tanto, la función objetivo queda:

$$\min w = \sum_{i=1}^m \sum_{k=1}^q h_{ik} x_{i2}^k + h_{i, q+1} s_{i2}$$

Pero para x_{i2}^k tenemos asociados los costos c_{i2}^k y para s_{i2} el costo asociado es cero, por lo tanto, tenemos:

$$\min w = \sum_{i=1}^m \sum_{k=1}^q c_{i2}^k x_{i2}^k$$

Por otro lado, la función objetivo del PTBM($m, 2, q$) es

$$\min z = \sum_{i=1}^m \sum_{j=1}^2 \sum_{k=1}^q c_{ij}^k x_{ij}^k = \sum_{i=1}^m \sum_{k=1}^q c_{i1}^k x_{i1}^k + \sum_{i=1}^m \sum_{k=1}^q c_{i2}^k x_{i2}^k$$

Es decir,

$$\min z = \sum_{i=1}^m \sum_{k=1}^q c_{i1}^k x_{i1}^k + w$$

Obsérvese que los valores de x_{i1}^k están totalmente determinados por los valores de x_{i2}^k , pues un origen i para un bien dado k , tiene disponibilidad a_i^k y como solamente tenemos dos destinos, cada origen tiene solamente dos opciones para enviar su disponibilidad, por lo tanto, si se decide que $x_{i2}^k = p$ unidades, entonces $x_{i1}^k = a_i^k - x_{i2}^k = a_i^k - p$, es decir, x_{i1}^k queda totalmente determinado a partir del valor de x_{i2}^k . Por lo tanto, la función objetivo del PTBM($m, 2, q$) puede ser vista como

$$\min z = w + cte.$$

Esto implica que ambos problemas, el PTC($m, q + 1$) y el PTBM($m, 2, q$), proporcionan el mismo costo de valor mínimo. Por lo tanto, ambos problemas son equivalentes. ■

Observación 1. El $\text{PTC}(m, q + 1)$ satisface que la suma de disponibilidades es igual a la suma de demandas; si sumamos el lado derecho de (4.2) y de (4.3) que representan las demandas, tenemos que

$$\sum_{k=1}^q \sum_{i=1}^m a_i^k - \sum_{k=1}^q b_1^k + \sum_{i=1}^m u_{i2} - \sum_{k=1}^q b_2^k = \sum_{k=1}^q \sum_{i=1}^m a_i^k - \sum_{k=1}^q \sum_{j=1}^2 b_j^k + \sum_{i=1}^m u_{i2}$$

Por hipótesis sabemos que $\sum_{i=1}^m a_i^k = \sum_{j=1}^2 b_j^k \quad \forall k$, así tenemos que:

$$\sum_{k=1}^q \sum_{i=1}^m a_i^k - \sum_{k=1}^q b_1^k + \sum_{i=1}^m u_{i2} - \sum_{k=1}^q b_2^k = \sum_{i=1}^m u_{i2}$$

Es decir, la suma de las disponibilidades (u_{i2}) es igual a la suma de las demandas.

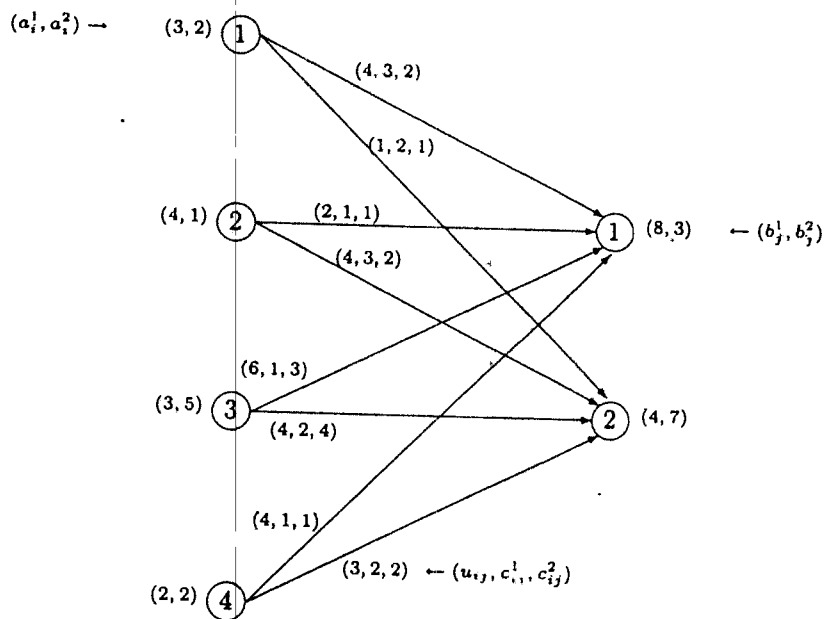
Observación 2. La transformación del $\text{PTBM}(m, 2, q)$ se planteó para satisfacer las demandas del destino 2 y por lo tanto, de manera inmediata se satisfacen las demandas del destino 1; de forma análoga, podría plantearse el $\text{PTC}(m, q + 1)$ en función de satisfacer las demandas del destino 1, es decir, hacer la correspondencia:

$$\begin{aligned} y_{ik} &= x_{i1}^k \\ t_{ik} &= x_{i2}^k \\ y_{i,q+1} &= s_{i1} \\ t_{i,q+1} &= s_{i2} \end{aligned}$$

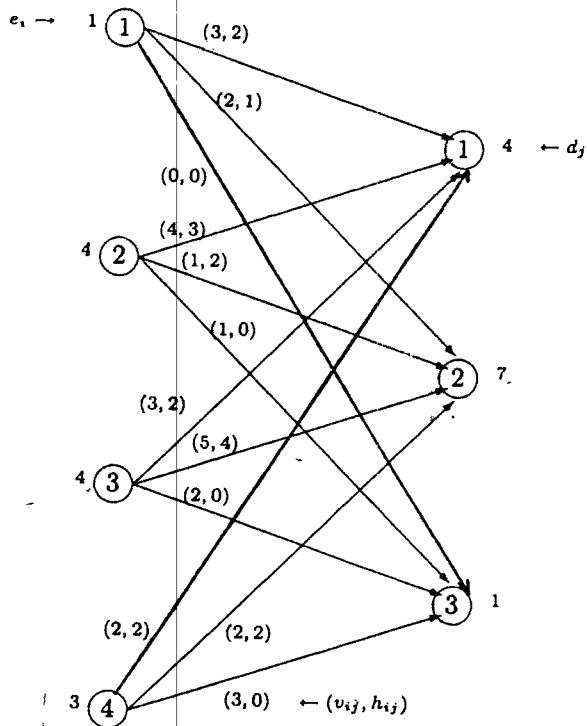
Observación 3. La condición para que la transformación sea válida es que el número de orígenes o de destinos no exceda de 2; por lo tanto, sin pérdida de generalidad, todo $\text{PTBM}(2, n, q)$ puede transformarse en un $\text{PTC}(q + 1, n)$ y la demostración es análoga a la presentada en esta sección.

4.1.1 Ejemplo

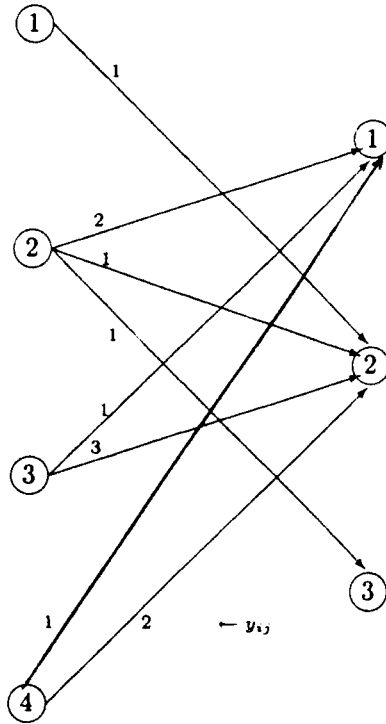
Considérese el siguiente PTBM(4,2,2):



Que se transforma en el siguiente PTC(4,3):



La solución óptima a este problema está dada por:



$$w_{\min} = 29$$

(para la resolución se utilizó el algoritmo de Busacker y Gowen, añadiendo un origen y un destino ficticios. Véase el capítulo I sección I.2)

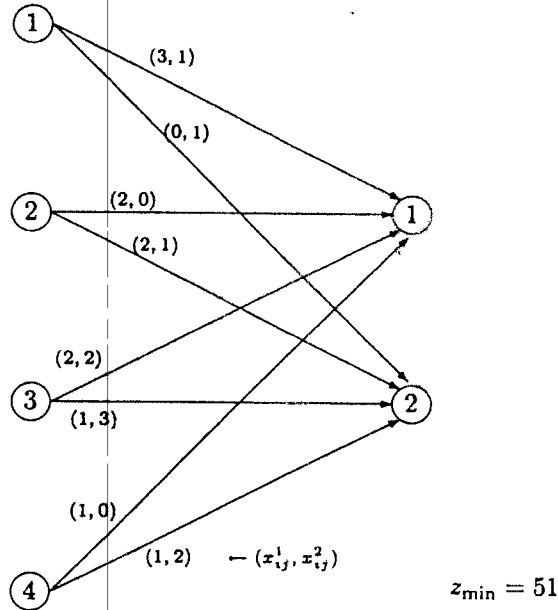
$$\begin{array}{ll} y_{11} = 0 = x_{12}^1 & t_{11} = 3 = x_{11}^1 \\ y_{12} = 1 = x_{12}^2 & t_{12} = 1 = x_{11}^2 \\ y_{13} = 0 = s_{12} & t_{13} = 0 = s_{11} \end{array}$$

$$\begin{array}{ll} y_{21} = 3 = x_{22}^1 & t_{21} = 2 = x_{21}^1 \\ y_{22} = 2 = x_{22}^2 & t_{22} = 0 = x_{21}^2 \\ y_{23} = 1 = s_{22} & t_{23} = 0 = s_{21} \end{array}$$

$$\begin{array}{ll} y_{31} = 1 = x_{32}^1 & t_{31} = 2 = x_{31}^1 \\ y_{32} = 3 = x_{32}^2 & t_{32} = 2 = x_{31}^2 \\ y_{33} = 0 = s_{32} & t_{33} = 2 = s_{31} \end{array}$$

$$\begin{array}{ll} y_{41} = 1 = x_{42}^1 & t_{41} = 1 = x_{41}^1 \\ y_{42} = 2 = x_{42}^2 & t_{42} = 0 = x_{41}^2 \\ y_{43} = 0 = s_{42} & t_{43} = 3 = s_{41} \end{array}$$

Por lo tanto, la solución óptima para el PTBM(4,2,2) es:



Observación 4. En el ejemplo anterior, el costo mínimo del PTC(4,3) fue de 29, y el costo mínimo del PTBM(4,2,2) fue de 51. La diferencia se debe al costo asociado al flujo enviado al destino 1.

4.2 Una Propiedad Topológica de la Red

En [4] Evans establece una transformación de un problema de flujo de bienes múltiples en un problema de flujo de un solo bien, en este caso la condición que debe satisfacer el problema de bienes múltiples es que su red asociada cumpla con una propiedad topológica (especificada en el teorema 2). Esto lo hace muy interesante, pues la transformación (y la equivalencia de los dos problemas) no depende del número de bienes ni de la dimensión de la red, sino únicamente de su estructura.

El problema transformado (flujo de un solo bien) tiene una matriz totalmente unimodular, lo que garantiza que existe una solución óptima entera si los requerimientos son enteros (la cual podrá encontrarse en un número finito de pasos utilizando alguno de los métodos vistos en el capítulo 1); así, el cálculo de las soluciones puede hacerse utilizando un algoritmo de marcaje. Un algoritmo que sirve para resolver el problema es el algoritmo simplex especializado en redes. Véase el capítulo 1, sección 2, subsección 3.

Nos referiremos a la formulación nodo-arco del problema de flujo de bienes múltiples:

$$\left\{ \begin{array}{l} \min w = \sum_{k=1}^q \sum_{(i,j) \in A} c_{ij} y_{ij}^k \\ \text{s.a.} \quad \sum_{k=1}^q y_{ij}^k + s_{ij} = b_{ij} \quad (i,j) \in A \\ \sum_{j \in N} y_{ij}^k - \sum_{j \in N} y_{ji}^k = r_i^k \quad i \in N, \forall k \\ y_{ij}^k, s_{ij} \geq 0 \quad \forall (i,j) \in A, k = \overline{1, q} \end{array} \right. \quad (4.15)$$

donde s_{ij} es la variable de holgura asociada a la restricción de capacidad.

Sea $i_0 \in N$. Definimos $A(i_0)$ como el conjunto $\{(i,j) \in A : i = i_0 \text{ o } j = i_0\}$, así $\bar{A}(i_0) = A - A(i_0)$.

Definimos un nuevo problema, llamado $P(i_0)$, como sigue:

$$\begin{aligned} \min z &= \sum_k \sum_{(i,j) \in A} c_{ij} y_{ij}^k \\ \sum_k y_{i_0 j}^k + s_{i_0 j} &= b_{i_0 j} \quad \forall (i_0, j) \in A(i_0) \end{aligned} \quad (4.16)$$

$$-\sum_k y_{j i_0}^k - s_{j i_0} = -b_{j i_0} \quad \forall (j, i_0) \in A(i_0) \quad (4.17)$$

$$\sum_{j \in N} y_{ij}^k - \sum_{j \in N} y_{ji}^k = r_i^k \quad \forall k, i \neq i_0, i \in N \quad (4.18)$$

$$\sum_{j \in N} s_{ij} - \sum_{j \in N} s_{ji} = \sum_{j \in N} b_{ij} - \sum_{j \in N} b_{ji} - \sum_k r_i^k \quad i \neq i_0, i \in N \quad (4.19)$$

$$y_{ij}^k, s_{ij} \geq 0 \quad \forall k = \overline{1, q}, (i,j) \in A \quad (4.20)$$

Observación 5. A toda matriz cuyas columnas contengan únicamente un 1, un -1 y ceros en sus restantes entradas, puede asociársele una gráfica, pues este tipo de matrices definen una matriz de incidencia nodo-arco. De esta forma, $P(i_0)$ representa un problema de flujo de un solo bien, pues cada variable aparece en exactamente dos restricciones con signos opuestos, es decir, la matriz del problema es una matriz de incidencia nodo-arco y por lo tanto, puede asociársele una red; es de un solo bien, pues la matriz de restricciones es una matriz de incidencia.

Observación 6. $P(i_0)$ satisface que la suma de disponibilidades es igual a la suma de demandas. Sumando el lado derecho de las restricciones (4.16) a (4.19):

$$\begin{aligned} & \sum_{j \in N} b_{i_0 j} - \sum_{j \in N} b_{j i_0} + \sum_{i \neq i_0} \sum_k r_{i_0}^k + \sum_{j \in N} \sum_{i \neq i_0} b_{i j} - \sum_{j \in N} \sum_{i \neq i_0} b_{j i} - \sum_{i \neq i_0} \sum_k r_{i_0}^k \\ = & \sum_{j \in N} b_{i_0 j} - \sum_{j \in N} b_{j i_0} + \sum_{j \in N} \sum_{i \neq i_0} b_{i j} - \sum_{j \in N} \sum_{i \neq i_0} b_{j i} \end{aligned}$$

Pero $i_0 \in N$

$$\begin{aligned} = & \sum_{j \in N} b_{i_0 j} - \sum_{j \in N} b_{j i_0} + \sum_{j \neq i_0} \sum_{i \neq i_0} b_{i j} - \sum_{j \neq i_0} \sum_{i \neq i_0} b_{j i} + \sum_{i \neq i_0} b_{i i_0} - \sum_{i \neq i_0} b_{i_0 i} \\ = & \sum_{j \neq i_0} \sum_{i \neq i_0} b_{i j} - \sum_{j \neq i_0} \sum_{i \neq i_0} b_{j i} \\ = & 0 \end{aligned}$$

Nótese que $P(i_0)$ es una relajación de (4.15) ya que las siguientes restricciones no pertenecen al problema:

$$\sum_k y_{ij}^k + s_{ij} = b_{ij} \quad (i, j) \in \bar{A}(i_0) \quad (4.21)$$

$$y_{i_0 j}^k - y_{j i_0}^k = r_{i_0}^k \quad \forall k \quad (4.22)$$

Observación 7. Claramente, cualquier solución de $P(i_0)$ es también una solución de (4.15) si y sólo si (4.21) y (4.22) se satisfacen.

Pues si una solución de $P(i_0)$ es también solución de (4.15) entonces claramente se satisfacen las restricciones (4.21) y (4.22). Inversamente, si (4.21) y (4.22) se satisfacen y se tiene una solución de $P(i_0)$ entonces esta solución satisface todas las de (4.15), pues las restricciones de (4.15) son las restricciones de $P(i_0)$ además de las restricciones (4.21) y (4.22).

En el siguiente teorema se establece una condición suficiente para que los dos problemas sean equivalentes.

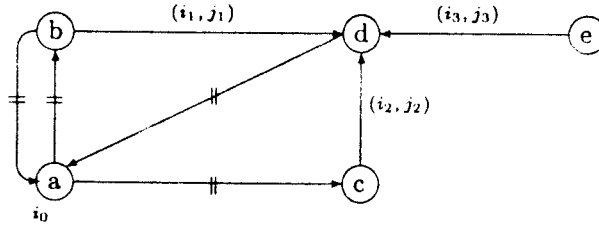
Teorema 2. Una solución óptima de $P(i_0)$ lo es también para (4.15) si existe una permutación de arcos en $\bar{A}(i_0)$, (i_1, j_1) , (i_2, j_2) , ..., (i_n, j_n) tales que

(i) (i_1, j_1) es incidente a un nodo, tal que todos los otros arcos incidentes están solamente en $A(i_0)$ y

(ii) (i_r, j_r) es incidente a un nodo, tal que todos los otros arcos incidentes están en $A(i_0) \cup J_r$, para $r = 2, 3, \dots, n$; donde $J_r = \{(i_s, j_s) : s < r\}$.

En este caso $P(i_0)$ y (4.15) se dicen equivalentes.

Ejemplo 1. Considérese la red siguiente:



El nodo a se escoge como i_0 . Los arcos marcados \parallel pertenecen al conjunto $A(i_0)$. Fijémonos en el arco (i_1, j_1) , este arco es incidente al nodo b , y este nodo tiene todos sus otros arcos incidentes pertenecientes a $A(i_0)$. Ahora el arco (i_2, j_2) es incidente al nodo d , los otros arcos incidentes a este nodo, uno pertenece a $A(i_0)$ y el otro pertenece a $J_2 = \{(i_1, j_1)\}$. Por último, el arco (i_3, j_3) es incidente al nodo d , cuyos arcos incidentes pertenecen, uno a $A(i_0)$ y los otros dos a $J_3 = \{(i_1, j_1), (i_2, j_2)\}$.

Puede concluirse que para esta red, existe un nodo i_0 tal que se satisfacen las condiciones del teorema 2, y así, cualquier problema de flujo de bienes múltiples asociado a esta red, será equivalente al problema $P(i_0)$.

Demostración del teorema 2: Se demostrará que si las condiciones (i) y (ii) se satisfacen, entonces las restricciones (4.21) y (4.22) también se satisfacen.

Sumemos la restricción (4.18) sobre todo $i \neq i_0$:

$$\sum_{i \neq i_0} \sum_{j \in N} y_{ij}^k - \sum_{i \neq i_0} \sum_{j \in N} y_{ji}^k = \sum_{i \neq i_0} r_i^k$$

Por hipótesis sabemos que

$$\begin{aligned} \sum_{i \in N} r_i^k &= \sum_{i \neq i_0} r_i^k + r_{i_0}^k = 0 \\ \Rightarrow \sum_{i \neq i_0} r_i^k &= -r_{i_0}^k \end{aligned}$$

Es decir, tenemos que:

$$\sum_{i \neq i_0} \sum_{j \in N} y_{ij}^k - \sum_{i \neq i_0} \sum_{j \in N} y_{ji}^k = -r_{i_0}^k$$

Para un bien dado k , tenemos un nodo l con $l \neq i_0$ para el cual consideramos el valor y_{li}^k (en $\sum_{j \in N} y_{ij}^k$) y cuando nos fijamos en i_1 tenemos el valor $-y_{i_1}^k$ (en $\sum_{j \in N} y_{ji}^k$) esto implica que para los arcos donde ninguno de los nodos extremos es i_0 , los flujos en ese arco se cancelan y los únicos que se conservan son los que tienen a i_0 como nodo extremo, es decir, se tiene que:

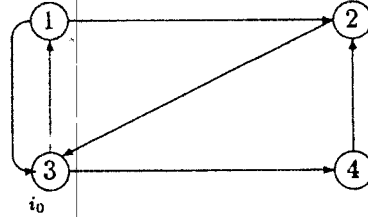
$$\sum_{j \in N} y_{ji_0}^k - \sum_{j \in N} y_{i_0j}^k = -r_{i_0}^k$$

que es precisamente la restricción (4.22) multiplicada por -1 .

Por lo tanto, (4.22) se satisface para toda k .

Hagamos un paréntesis para ver un ejemplo de lo anterior:

Ejemplo 2. Considérese la red siguiente, y tomemos $i_0 = 3$



Para un bien k tenemos:

$$\begin{aligned} \sum_{j \in N} \sum_{i \neq i_0} y_{ij}^k &= \sum_{j \in N} y_{1j}^k + \sum_{j \in N} y_{2j}^k + \sum_{j \in N} y_{4j}^k \\ &= y_{12}^k + y_{13}^k + y_{23}^k + y_{42}^k \end{aligned}$$

$$\begin{aligned} - \sum_{j \in N} \sum_{i \neq i_0} y_{ji}^k &= - \sum_{j \in N} y_{j1}^k - \sum_{j \in N} y_{j2}^k - \sum_{j \in N} y_{j4}^k \\ &= -y_{31}^k - y_{12}^k - y_{42}^k - y_{34}^k \end{aligned}$$

Sumando:

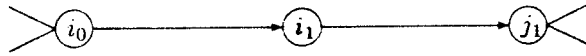
$$\begin{aligned} \sum_{j \in N} \sum_{i \neq i_0} y_{ij}^k - \sum_{j \in N} \sum_{i \neq i_0} y_{ji}^k &= y_{12}^k + y_{13}^k + y_{23}^k + y_{42}^k - y_{31}^k - y_{12}^k - y_{42}^k - y_{34}^k \\ &= y_{13}^k + y_{23}^k - y_{31}^k - y_{34}^k \\ &= \sum_{j \in N} y_{j3}^k - \sum_{j \in N} y_{3j}^k = -r_3^k \end{aligned}$$

Regresando a la demostración, supongamos que se verifica (i). Tenemos dos casos a considerar:

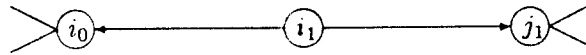
- (a) Cada arco incidente a i_1 excepto para (i_1, j_1) está en $A(i_0)$.
- (b) Cada arco incidente a j_1 excepto para (i_1, j_1) está en $A(i_0)$.

Consideremos el caso (a), éste tiene tres subcasos:

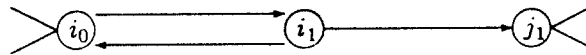
(a.1) $(i_0, i_1) \in A(i_0)$



(a.2) $(i_1, i_0) \in A(i_0)$



(a.3) (i_0, i_1) y $(i_1, i_0) \in A(i_0)$



Veamos el subcaso (a.1):

Aplicamos la restricción (4.16) al arco $(i_0, i_1) \in A(i_0)$

$$\sum_k y_{i_0 i_1}^k + s_{i_0 i_1} = b_{i_0 i_1} \quad (4.23)$$

Aplicamos la restricción (4.18) al nodo i_1

$$y_{i_1 j_1}^k - y_{i_0 i_1}^k = r_{i_1}^k \quad (4.24)$$

Aplicamos la restricción (4.19) al nodo i_1

$$s_{i_1 j_1} - s_{i_0 i_1} = b_{i_1 j_1} - b_{i_0 i_1} - \sum_k r_{i_1}^k \quad (4.25)$$

Sumamos (4.24) sobre k y sumamos (4.25):

$$\sum_k y_{i_1 j_1}^k - \sum_k y_{i_0 i_1}^k + s_{i_1 j_1} - s_{i_0 i_1} = \sum_k r_{i_1}^k + b_{i_1 j_1} - b_{i_0 i_1} - \sum_k r_{i_1}^k$$

Cancelando $\sum_k r_{i_1}^k$ y por (4.23) nos queda:

$$\sum_k y_{i_1 j_1}^k + s_{i_1 j_1} = b_{i_1 j_1}$$

Que es precisamente la restricción (4.21) aplicada al arco $(i_1, j_1) \in \bar{A}(i_0)$.

Los otros dos subcasos son análogos a éste.

El caso (b) es análogo al caso (a).

Ahora supongamos que la condición (ii) se verifica y demostraremos que también la restricción (4.21).

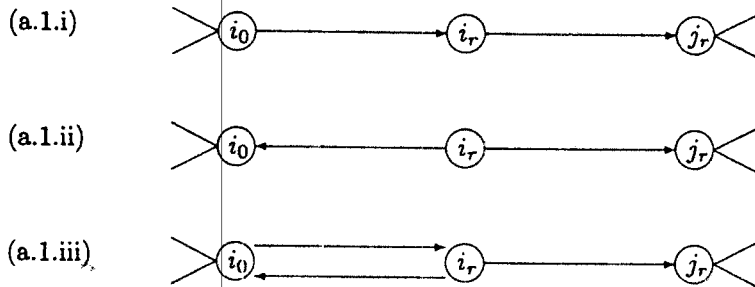
Tenemos dos casos:

- (a) Cada arco incidente a i_r , excepto para (i_r, j_r) , está en $A(i_0) \cup J_r$, donde $J_r = \{(i_s, j_s) : s < r\}$.
- (b) Cada arco incidente a j_r , excepto para (i_r, j_r) , está en $A(i_0) \cup J_r$.

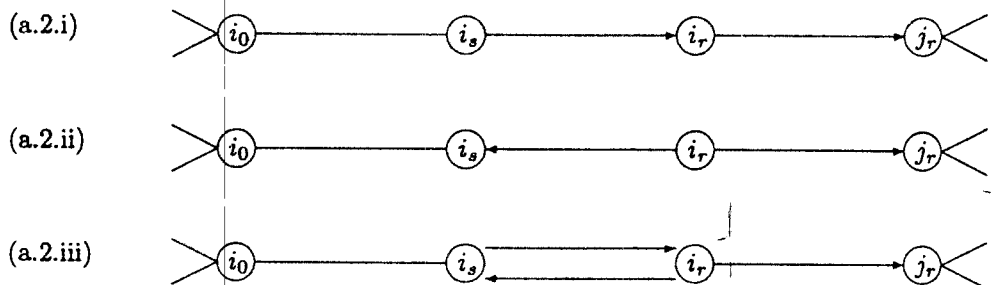
Veamos el caso (a):

Se tienen dos subcasos:

- (a.1) Cada arco incidente a i_r está en $A(i_0)$, el cual tiene tres posibilidades:



- (a.2) Cada arco incidente a i_r está en J_r , el cual tiene tres posibilidades:



Las demostraciones de todos los casos son análogas.

Demostraremos el caso (a.2.i).

Aplicando (4.18) al nodo i_r y sumando sobre k :

$$\sum_k y_{i_r j_r}^k - \sum_k y_{i_s i_r}^k = \sum_k r_{i_r}^k$$

Aplicamos la restricción (4.19) al nodo i_r

$$s_{i_r j_r} - s_{i_s i_r} = b_{i_r j_r} - b_{i_s i_r} - \sum_k r_{i_r}^k$$

Sumamos estas dos expresiones y nos queda:

$$\sum_k y_{i_r j_r}^k - \sum_k y_{i_s i_r}^k + s_{i_r j_r} - s_{i_s i_r} = b_{i_r j_r} - b_{i_s i_r}$$

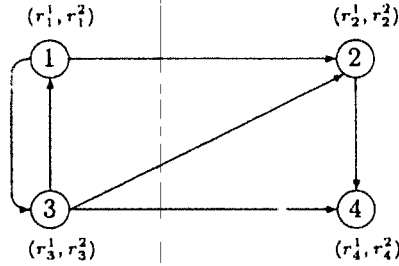
Pero, el arco (i_s, i_r) puede ser visto como (i_s, j_s) que pertenece a J_r y por hipótesis de inducción este arco satisface la restricción (4.21). Por lo tanto, nos queda:

$$\sum_k y_{i_r j_r}^k + s_{i_r j_r} = b_{i_r j_r}$$

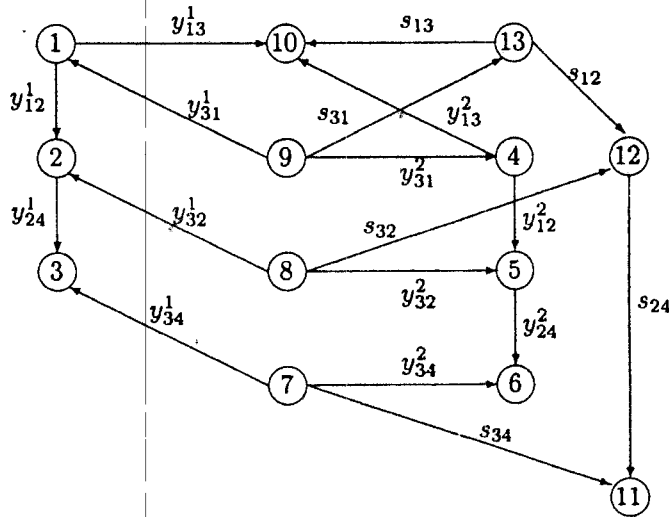
Que es precisamente la restricción (4.21) aplicada al arco $(i_r, j_r) \in \bar{A}(i_0)$. Así, queda demostrado el teorema. ■

Observación 8. El conjunto de condiciones establecidas en el teorema 2 no son necesarias, pues como se ve en la demostración, una solución de $P(i_0)$ puede satisfacer la restricción (4.22) sin que (i) y (ii) se verifiquen.

Ejemplo 3. Consideremos la red siguiente para dos bienes:



Tomemos $i_0 = 3$, entonces el problema $P(3)$ tiene la siguiente red asociada:



Las disponibilidades y demandas se determinan por el lado derecho de las restricciones (4.16) a (4.19).

Observación 9. En el ejemplo anterior $P(1)$, $P(2)$, $P(4)$ no satisfacen las condiciones del teorema 2.

Veámoslo para $P(2)$

$$A(2) = \{(1, 2), (3, 2), (2, 4)\}$$

El arco $(3, 4)$ es incidente al nodo 4 que tiene solamente un arco incidente que pertenece a $A(2)$. Por lo tanto, se satisface la condición (i) del teorema 2.

Veamos ahora si se satisface la condición (ii). El arco $(3, 1)$ es incidente al nodo 3 que no tiene todos sus arcos incidentes pertenecientes ni a $A(2)$ ni a $J_2 = \{(3, 4)\}$; el arco $(1, 3)$ es incidente al nodo 3 que no tiene todos sus arcos incidentes pertenecientes ni a $A(2)$ ni a $J_2 = \{(3, 4)\}$, por lo tanto, la condición (ii) no se satisface.

De manera similar se observa que $P(1)$ y $P(4)$ no satisfacen las condiciones del teorema 2.

Con esta observación la pregunta inmediata es ¿Cómo encontrar en una red, un nodo i_0 tal que se satisfagan las condiciones del teorema 2?, para resolver este problema, Evans propone un algoritmo (de tipo glotón) en el que van considerándose todos los nodos (uno por uno) hasta encontrar uno que satisfaga las condiciones del teorema 2 o que pueda afirmarse que la red en cuestión no las satisface.

Algoritmo para encontrar un nodo i_0 que satisfaga las condiciones del teorema 2.

- (0) Todos los nodos y arcos están inicialmente sin etiquetar. Escoger el nodo que tenga más número de arcos incidentes (en realidad cualquier nodo puede servir). Sea éste i_0 .

Etiquetar los arcos incidentes y nodos adyacentes a estos arcos.

- (1) (Paso iterativo)

Escoger cualquier nodo etiquetado. Si a lo más un arco incidente está sin etiquetar, etiquetar este arco y su otro nodo incidente. Repetir.

- Si esta condición no se satisface, escoger otro nodo etiquetado y repetir.
- Si ningún nodo etiquetado satisface esta condición ALTO, las condiciones del teorema 2 no se satisfacen.

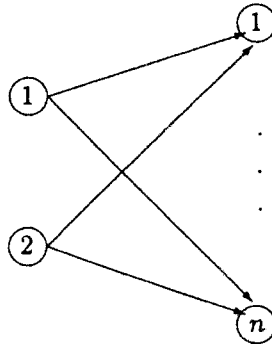
En este caso borrar i_0 de la lista de los nodos, escoger otro nodo como i_0 , borrar todas las etiquetas e ir al paso 0.

Se continúa hasta que se encuentra un nodo que satisfaga las condiciones del teorema 2 o hasta que todos los nodos hayan sido considerados.

Justificación: En el paso (0), todos los arcos que reciben etiqueta conforman el conjunto $A(i_0)$. La primera aplicación del paso 1 busca que un arco satisfaga la condición (i), es decir, un arco (i_1, j_1) tal que todos los demás arcos incidentes a i_1 (o j_1) pertenecen a $A(i_0)$. Si no existe tal arco, claramente la condición (i) no se satisface. La repetición del paso iterativo busca que los demás arcos satisfagan la condición (ii), es decir, arcos (i_r, j_r) tales que todos los arcos incidentes a i_r (o j_r) pertenecen o a $A(i_0)$ o a J_r , donde $J_r = \{(i_s, j_s) : s < r\}$

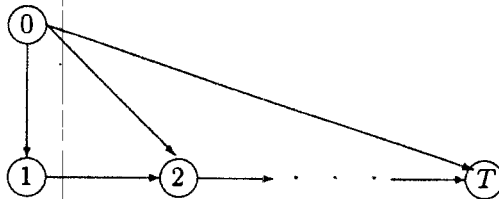
La elección de un nodo etiquetado puede hacerse de manera arbitraria, pues en cualquier momento un arco es etiquetado, así el número de arcos incidentes a cualquier nodo no etiquetado no puede incrementarse, además porque se tiene una red finita. ■

Observación 10. Si en la red asociada al problema de transporte de bienes múltiples con dos orígenes (o dos destinos) tomamos como i_0 a un origen (o destino, según sea el caso) entonces la red satisface las condiciones del teorema 1.



Tomemos por ejemplo, $i_0 = 1$ (el origen 1), entonces aplicando el algoritmo, todos los arcos de la forma $(1, k)$ para $k = \overline{1, n}$ son etiquetados, así como todos los destinos; tomemos cualquier destino k , este tiene solamente dos arcos incidentes $(1, k)$ y $(2, k)$, y el segundo es el único sin etiqueta, por lo tanto, puede recibirla. Después de un número finito de pasos, todos los arcos y nodos están etiquetados. De esta forma, la red satisface las condiciones del teorema 2.

Observación 11. Para la red asociada al problema de planeación de la producción dinámica de bienes múltiples de un nivel con T periodos¹, el vértice 0 satisface las condiciones del teorema 2.



Sea $i_0 = 0$, entonces todos los arcos de la forma $(0, k)$ para $k = \overline{1, T}$ son etiquetados, así como todos los vértices vecinos a 0. El vértice 1 tiene solamente dos arcos incidentes, a saber, $(0, 1)$ y $(1, 2)$, y éste último es el único sin etiqueta, por lo tanto, puede recibirla. De aquí en adelante, fijémonos en cualquier vértice etiquetado k para $1 < k < T$, éste vértice tiene tres arcos incidentes, $(0, k)$, $(k-1, k)$ y $(k, k+1)$ y el último arco es el único sin etiqueta, por lo tanto, puede recibirla; al final, para el vértice T , éste tiene ya sus dos arcos incidentes etiquetados. Por lo tanto, se satisfacen las condiciones del teorema 2 para este tipo de redes.

Por lo visto hasta ahora, no todas las redes satisfacen las condiciones del teorema 2, es por esto que a continuación se presenta una clasificación de las redes para las cuales es posible aplicar la transformación propuesta por Evans.

Proposición 1. En toda gráfica que constituya una cadena, puede encontrarse un nodo i_0 que satisfaga las condiciones del teorema 2.

¹Los vértices de 0 a T representan los periodos de tiempo, un arco de 0 a k representa la producción en el periodo k , y el arco $(k, k+1)$ representa el inventario que va acumulándose. Todos los arcos tienen capacidades.

Demostración: Aplicando el algoritmo para encontrar un nodo i_0 , escogemos i_0 como cualquier nodo extremo de la cadena. Etiquetamos su arco incidente y el único nodo vecino de i_0 , sea éste " i_1 ", por ser una cadena, i_1 tiene dos arcos incidentes y solamente uno que no está etiquetado, por lo tanto, puede etiquetarse este arco y el nodo vecino a i_1 , sea éste " i_2 ". Nuevamente, i_2 tiene un solo arco sin etiqueta y por lo tanto puede recibirla. Continuando de esta forma, el otro nodo extremo es etiquetado, pues se trata de una cadena, que además es finita.

Así, i_0 satisface las condiciones del teorema 2. ■

Proposición 2. *En toda gráfica que constituya un ciclo elemental, puede encontrarse un nodo i_0 que satisfaga las condiciones del teorema 2.*

Demostración: Aplicando el algoritmo, escogemos i_0 como cualquier nodo del ciclo. Como tenemos un ciclo elemental, i_0 tiene únicamente dos arcos incidentes; los etiquetamos, así como a sus nodos vecinos. Sean éstos i_1 e i_2 . Sin pérdida de generalidad, tomemos i_1 , este nodo tiene solamente un arco sin etiqueta (pues tenemos un ciclo elemental) por lo tanto, podemos etiquetarlo y también al nodo vecino, digamos i_3 . Nuevamente i_3 satisface las condiciones del algoritmo y podemos continuar de esta manera hasta etiquetar al nodo i_l que es vecino del nodo i_2 , ya que el ciclo es finito y elemental. El nodo i_l tiene solamente un arco sin etiqueta, por lo tanto, puede recibirla y como el nodo i_2 ya estaba etiquetado, tenemos todos los nodos y arcos etiquetados.

Así i_0 satisface las condiciones del teorema 2. ■

Sean, ahora, $G_1 = (N_1, A_1)$ y $G_2 = (N_2, A_2)$ dos gráficas que satisfacen las condiciones del teorema 2.

Sea

$$G = G_1 \cup G_2 = (N, A)$$

donde $N = N_1 \cup N_2$ y $A = A_1 \cup A_2$

Sea $i \in N$, el grado del nodo i en la gráfica G puede calcularse por:

$$g_G(i) = a_i + b_i - c_i$$

donde

$$a_i = g_{G_1}(i)$$

$$b_i = g_{G_2}(i)$$

y c_i es el número de arcos que pertenecen a $A_1 \cap A_2$ y cuyo vértice extremo es i .

Sea I_0^1 el conjunto cuyos elementos son los vértices $i \in N_1$ que son candidatos a ser i_0 , es decir, cada vértice de I_0^1 satisface las condiciones del teorema 2 para la gráfica G_1 . De manera análoga se define I_0^2 .

Sea

$$I_0 = I_0^1 \cap I_0^2$$

Proposición 3. *Sea G una gráfica que es unión de dos gráficas G_1 y G_2 que satisfacen las condiciones del teorema 2. Si*

$$(i) I_0 \neq \emptyset$$

$$(ii) \forall i \in N - I_0 \text{ se tiene que } a_i - c_i = 0 \text{ ó } b_i - c_i = 0$$

o bien, existe a lo más un $i \in N - I_0$ tal que $a_i - c_i = 1$ ó $b_i - c_i = 1$, pero todos los demás $j \in N - I_0$, $j \neq i$, tienen que $a_j - c_j = 0$ ó $b_j - c_j = 0$

(iii) $\forall i \in I_0$ se tiene que $a_i - c_i = 0$ ó $b_i - c_i = 0$

o bien, existe a lo más un $i \in I_0$ tal que $a_i - c_i \geq 1$ ó $b_i - c_i \geq 1$, pero todos los demás $j \in I_0$, $j \neq i$, tienen que $a_j - c_j = 0$ ó $b_j - c_j = 0$

Entonces existe un vértice $i \in I_0$ tal que se satisfacen las condiciones del teorema 2 para la gráfica G .

Demostración: Supongamos ciertas las hipótesis. Sea i_0 algún elemento de I_0 : Si existe a lo más un $i \in I_0$ tal que $a_i - c_i \geq 1$ ó $b_i - c_i \geq 1$, entonces sea éste i_0 . Si $\forall i \in I_0$ se tiene que $a_i - c_i = 0$ ó $b_i - c_i = 0$, entonces cualquier elemento de I_0 puede tomarse como i_0 .

Aplicamos el algoritmo para verificar que se satisfagan las condiciones del teorema 2. Etiquetamos todos los arcos incidentes a i_0 , así como sus vértices vecinos. Fijémonos en cualquier vértice etiquetado $i \in N$: Si $a_i - c_i = 0$, esto significa que el grado del vértice i en G es igual al grado del vértice i en la gráfica G_1 . Por lo tanto, como para G_1 se satisfacen las condiciones del teorema 2, este vértice, en algún paso del algoritmo, presentará solamente un arco sin etiqueta, la cual puede recibir. De manera análoga, si $b_i - c_i = 0$ el grado de i en G es igual al grado de i en G_2 , nuevamente, este vértice tendrá un solo arco sin etiqueta, pues G_2 satisface las condiciones del teorema 2.

Si i es tal que $a_i - c_i = 1$, esto significa que el grado de i en G es uno más que el grado de i en G_1 . Por lo tanto, este vértice tiene dos arcos sin etiqueta (uno que le corresponde por ser elemento de G_1 y el arco "de más", digámosle así). Sin embargo, consideremos los vértices vecinos a i correspondientes a estos dos arcos, llamémosles j_1 y j_2 . Para ambos vértices se tiene que $a_{j_k} - c_{j_k} = 0$ ó $b_{j_k} - c_{j_k} = 0$, para $k = 1, 2$ (esto es porque i es el único vértice que presenta que $a_i - c_i = 1$) y por el razonamiento anterior, todos los arcos incidentes a j_1 y a j_2 son etiquetados. Por lo tanto, el vértice i también presentará que todos sus vértices incidentes están ya etiquetados. De manera análoga, se demuestra que si i es tal que $b_i - c_i = 1$ entonces i tendrá todos sus arcos incidentes etiquetados. En un momento dado, todos los vértices y arcos de G están etiquetados, y esto demuestra que la gráfica G satisface las condiciones del teorema 2. ■

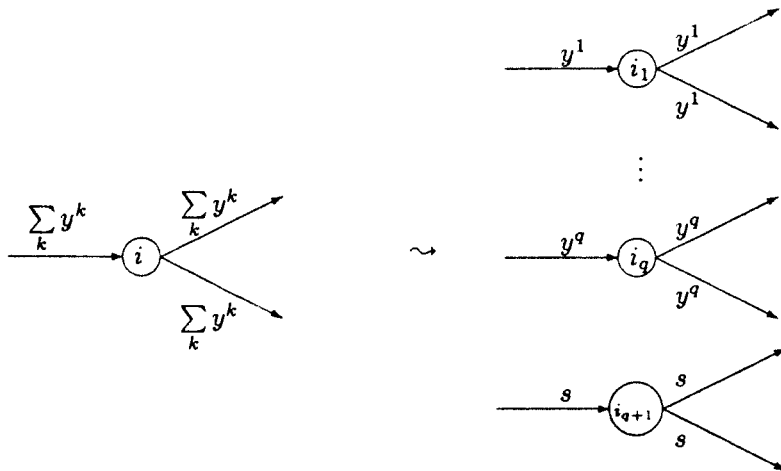
Observación 12. Denotemos por (N_0, A_0) la gráfica asociada al problema $P(i_0)$. Como las restricciones (4.16) y (4.17) están asociadas a cada arco perteneciente a $A(i_0)$ se tienen $|A(i_0)|$ restricciones, de la restricción (4.18) hay $q(|N| - 1)$ ecuaciones, pues esta restricción se aplica para todos los bienes y para todos los nodos excepto para i_0 , para la restricción (4.19) se tienen $|N| - 1$ ecuaciones, pues esta se aplica a todos los nodos excepto para i_0 . Así, se tiene que

$$\begin{aligned} |N_0| &= |A(i_0)| + q(|N| - 1) + |N| - 1 \\ &= |A(i_0)| + (q + 1)(|N| - 1) \end{aligned}$$

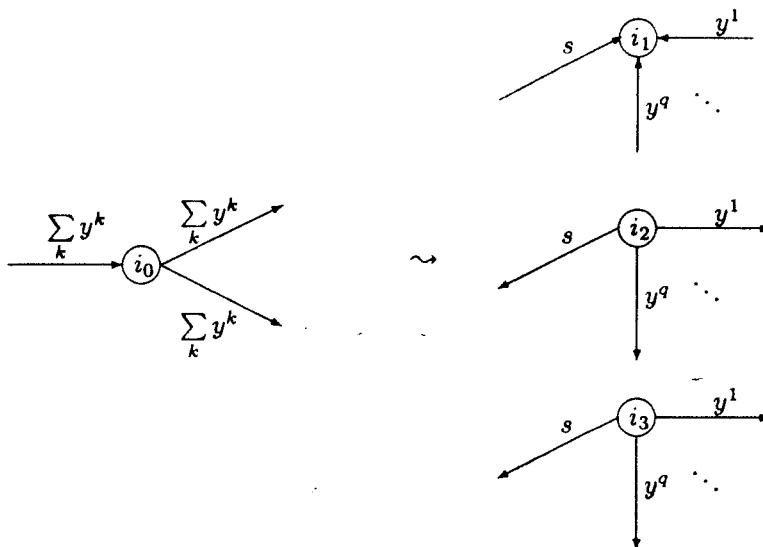
El número de arcos que tendrá la red (N_0, A_0) está relacionado con todas las variables de (4.15), más la holgura, es decir,

$$|A_0| = |A|(q + 1)$$

Observación 13. Dado un nodo de (4.15) diferente de i_0 , por la restricción (4.18) le corresponde q nodos y por la restricción (4.19) le corresponde uno más. Cada uno de estos nodos tiene arcos incidentes a él según los tenga en la red (4.15), por ejemplo,

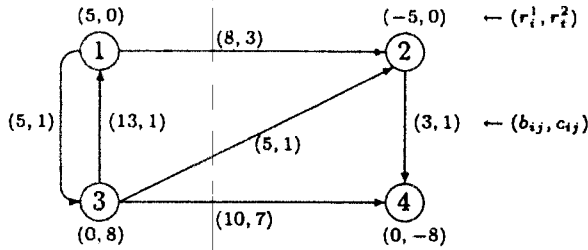


Para el vértice i_0 , por las restricciones (4.16) y (4.17) le corresponde $|A(i_0)|$ número de nodos en (N_0, A_0) . Cada uno de ellos tendrá q arcos incidentes, que corresponden al flujo de cada uno de los bienes, más uno, asociado a la holgura. Por ejemplo,

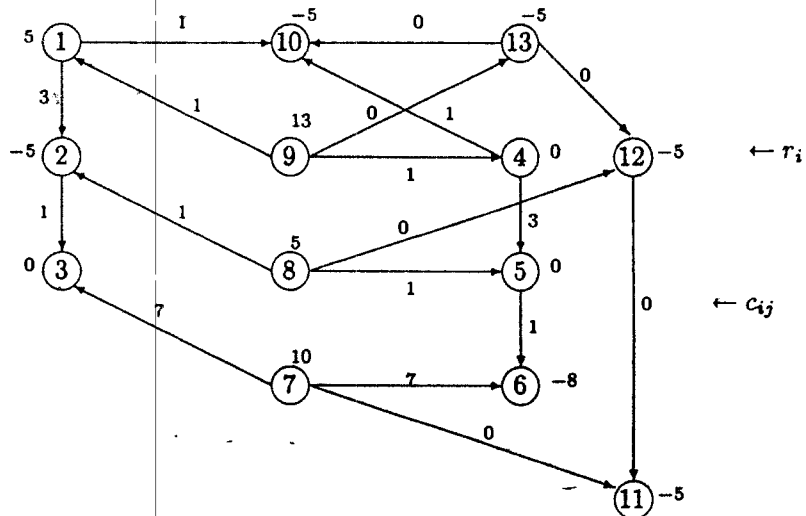


4.2.1 Ejemplo

Considérese el siguiente problema de flujo de dos bienes:



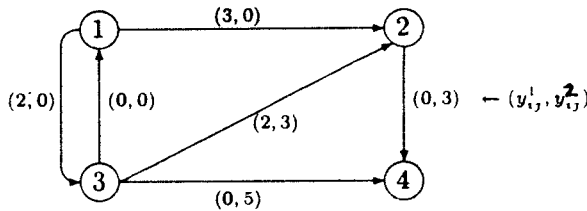
Como ya se había visto en el ejemplo 3, tomando $i_0 = 3$, esta red satisface las condiciones del teorema 2, por lo tanto la red asociada a $P(i_0)$ para este caso es la siguiente:



Aplicamos el algoritmo simplex especializado en redes, después de 4 iteraciones se encuentra la solución óptima: $z_{\min} = 81$ $x_{12} = 3$, $x_{1,10} = 2$, $x_{5,6} = 3$, $x_{7,6} = 5$, $x_{7,11} = 5$, $x_{8,2} = 2$, $x_{8,5} = 3$, $x_{9,13} = 13$, $x_{13,10} = 3$, $x_{13,12} = 5$, todas las demás iguales a cero.

Cada x_{ij} está asociada a un y_{ij}^k o a una s_{ij} , así, tenemos que:

$$\begin{aligned} x_{12} &= s_{24} = 3 \\ x_{1,10} &= y_{13}^1 = 2 \\ x_{56} &= y_{24}^2 = 3 \\ x_{76} &= y_{34}^2 = 5 \\ x_{7,11} &= s_{34} = 5 \\ x_{82} &= y_{32}^1 = 2 \\ x_{85} &= y_{32}^2 = 3 \\ x_{9,13} &= s_{31} = 13 \\ x_{13,10} &= s_{13} = 3 \\ x_{13,12} &= s_{12} = 8 \end{aligned}$$



4.2.2 Un Caso Particular

Este caso se refiere al hecho de que en una gráfica que satisface las condiciones del teorema 2, la gráfica asociada al problema $P(i_0)$ puede resultar no conexa, sin embargo como se verá más adelante esto no causará mayor problema, pues la solución óptima del problema (4.15) será la suma (unión) de las soluciones óptimas de cada componente conexa del problema $P(i_0)$.

Primero se establece la siguiente proposición que indica bajo qué circunstancias la gráfica asociada al problema $P(i_0)$ no será conexa.

Proposición 4. Si el nodo i_0 es un punto de articulación, en una gráfica que satisface las condiciones del teorema 2, entonces la gráfica asociada a $P(i_0)$ es no conexa. Además, si al eliminar i_0 de la gráfica se forman p componentes conexas entonces la gráfica asociada a $P(i_0)$ tendrá p componentes conexas.

Demostración: Consideremos la gráfica del problema (4.15), supongamos que al eliminar i_0 la gráfica se descompone en p componentes conexas. Podemos expresar los conjuntos N y A de la siguiente manera:

$$N = N_1 \cup N_2 \cup \dots \cup N_p$$

donde

$$N_l = \{i \in N : i \text{ está en la } l\text{-ésima componente conexa}\} \cup \{i_0\}$$

y

$$A = A_1 \cup A_2 \cup \dots \cup A_p$$

donde

$$A_l = \{(i, j) \in A : i, j \in N_l\}$$

Así, se tiene que

$$N_1 \cap N_2 \cap \dots \cap N_p = \{i_0\}$$

y

$$(i, i_0), (i_0, i) \in A_l \quad i \in N_l$$

De esta forma, tenemos que

$$A(i_0) = A_1(i_0) \cup A_2(i_0) \cup \dots \cup A_p(i_0)$$

donde

$$A_l(i_0) = \{(i, i_0), (i_0, i) \in A(i_0) : i \in N_l\}$$

El conjunto de restricciones de $P(x_0)$ podemos separarlo, es decir, podemos escribir:

$$\begin{aligned} \sum_k y_{i_0j}^k + s_{i_0j} &= b_{i_0j} \quad \forall (i_0, j) \in A_1(i_0) \\ -\sum_k y_{ji_0}^k - s_{ji_0} &= -b_{ji_0} \quad \forall (j, i_0) \in A_1(i_0) \\ \sum_{j \in N} y_{ij}^k - \sum_{j \in N} y_{ji}^k &= r_i^k \quad \forall k, i \neq i_0, i \in N_1 \\ \sum_{j \in N} s_{ij} - \sum_{j \in N} s_{ji} &= \sum_{j \in N} b_{ij} - \sum_{j \in N} b_{ji} - \sum_k r_i^k \quad i \neq i_0, i \in N_1 \\ &\vdots \\ \sum_k y_{i_0j}^k + s_{i_0j} &= b_{i_0j} \quad \forall (i_0, j) \in A_p(i_0) \\ -\sum_k y_{ji_0}^k - s_{ji_0} &= -b_{ji_0} \quad \forall (j, i_0) \in A_p(i_0) \\ \sum_{j \in N} y_{ij}^k - \sum_{j \in N} y_{ji}^k &= r_i^k \quad \forall k, i \neq i_0, i \in N_p \\ \sum_{j \in N} s_{ij} - \sum_{j \in N} s_{ji} &= \sum_{j \in N} b_{ij} - \sum_{j \in N} b_{ji} - \sum_k r_i^k \quad i \neq i_0, i \in N_p \end{aligned}$$

Por ser $P(i_0)$ un problema de flujo de un solo bien, cada variable aparece exactamente dos veces con signos contrarios, además cada variable aparece las dos veces en un mismo bloque de restricciones, es decir, no puede suceder que una variable aparezca una vez en el bloque de restricciones asociadas a (N_s, A_s) y que ella misma con el signo contrario aparezca en el bloque asociado a (N_r, A_r) , esto es porque $N_r \cap N_s = \{i_0\}$ y $A_r(i_0) \cap A_s(i_0) = \emptyset$

Por lo tanto, la matriz de incidencia nodo-arco del problema $P(i_0)$ está conformada por bloques:

$$\begin{pmatrix} B_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & B_p \end{pmatrix}$$

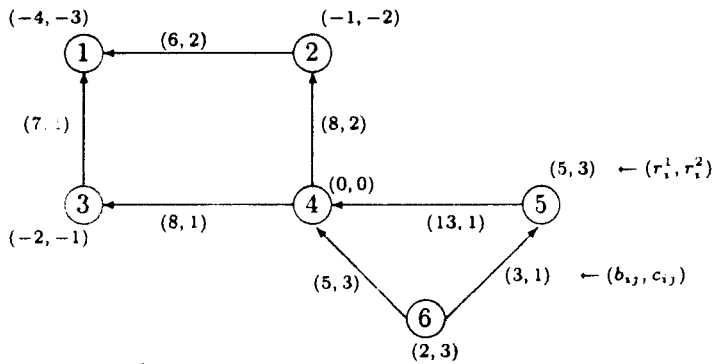
donde B_i es la matriz de restricciones del bloque i . Y se tienen p bloques, pues se tienen p conjuntos de restricciones. Puede concluirse que la gráfica asociada a $P(i_0)$ tiene p componentes conexas. ■

Observación 14. En cada componente conexa de $P(i_0)$, la suma de disponibilidades es igual a la suma de las demandas. De esta manera puede resolverse cada componente conexa por separado y la unión de las soluciones óptimas de cada una de ellas será la solución óptima para (4.15).

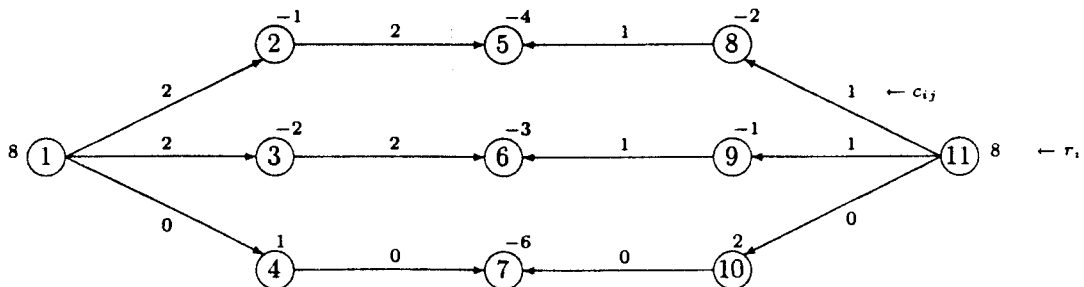
Observación 15. Cada componente conexa tiene $|A_i(i_0)| + (q + 1)(|N_i| - 1)$ nodos y $|A_i|(q + 1)$ arcos.

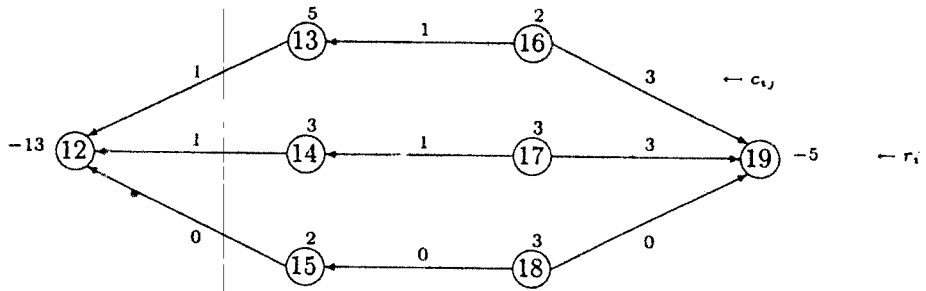
Ejemplo

Considérese la red siguiente para un problema de flujo de dos bienes:



Tomando $i_0 = 4$ se verifica que esta red satisface las condiciones del teorema 2; si eliminamos el nodo 4 la gráfica se descompone en dos componentes conexas, así $P(i_0)$ constará de dos gráficas, estas son:

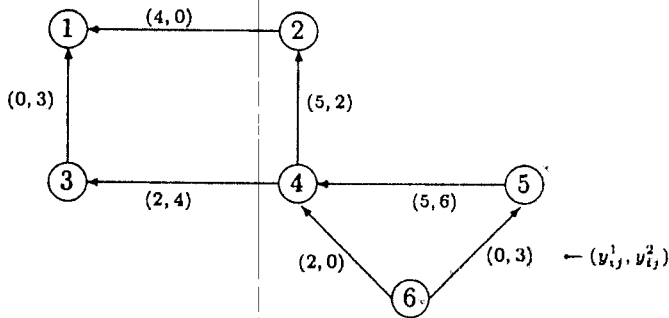




Resolviendo la primera red con el algoritmo simplex especializado en redes, encontramos, después de dos iteraciones, la solución óptima: $z_{\min} = 31$, con $y_{42}^1 = 5$, $y_{42}^2 = 2$, $y_{21}^1 = 4$, $y_{31}^2 = 3$, $y_{43}^1 = 2$, $y_{43}^2 = 4$

Ahora, resolvemos la segunda red, y encontramos en la segunda iteración la solución óptima, $z_{\min} = 20$, con $y_{54}^1 = 5$, $y_{54}^2 = 6$, $y_{65}^2 = 3$, $y_{64}^1 = 2$

Por lo tanto, para la red original tenemos la siguiente solución óptima:



$$w_{\min} = 31 + 20 = 51$$

Capítulo 5

Caracterización de una Condición Bajo la Cual Existen Soluciones Enteras para el Problema en General

Una matriz A se dice totalmente unimodular si cualquier submatriz cuadrada de A tiene determinante igual a 0, +1 ó -1. Por lo tanto, un sistema de ecuaciones $Ax = b$ con A una matriz totalmente unimodular, tendrá soluciones enteras si el vector b es entero, sin embargo, aunque b fuera entero, si A no es totalmente unimodular no hay ninguna garantía para la enterabilidad de las soluciones del sistema de ecuaciones.

Con respecto al problema de flujo de bienes múltiples a costo mínimo, en el capítulo 3 se expuso un método de resolución para el mismo. Sin embargo, si podemos garantizar que las soluciones del problema van a ser siempre enteras, entonces podría plantearse un método de resolución de redes que resultaría ser más eficiente que el método propuesto en dicho capítulo.

En este capítulo se expondrá bajo qué condiciones un problema de flujo de bienes múltiples a costo mínimo tiene soluciones enteras.

En [7] y [6] se presentan dos diferentes condiciones bajo las cuales el problema de flujo de bienes múltiples presenta tal característica. En este capítulo se analiza el resultado publicado en [6].

5.1 Preliminares

Nuevamente basamos el desarrollo utilizando la formulación nodo-arco del problema de flujo de bienes múltiples.

Sea $G = (N, A)$ la red asociada al problema. Supongamos que $|N| = n$ y $|A| = m$.

Como vimos en el capítulo II, el problema es:

$$\begin{aligned} \min w &= \sum_k c' y^k \\ \sum_k y^k &\leq b \end{aligned} \quad (5.1)$$

$$\begin{aligned} A_k y^k &= r^k \quad \forall k = \overline{1, q} \\ y^k &\geq 0 \end{aligned} \quad (5.2)$$

donde y^k es el m -vector de flujo en los arcos para el bien k , r^k el $(n-1)$ -vector de requerimientos enteros del bien k , b el m -vector de capacidades enteras en los arcos y A_k la matriz resultante de eliminar un renglón de la matriz de incidencia nodo arco, así A_k es de dimensión $(n-1) \times m$ y de rango $(n-1)$, y c' es el m -vector de costos en los arcos.

Observación 1. Cada A_k es totalmente unimodular, mientras que la matriz del sistema completo (5.1), (5.2) no lo es.

Para cada bien k , existe una matriz básica de la matriz A_k , que corresponde a un árbol de expansión de G . Sean estas bases B_k , así (5.2) puede reescribirse como:

$$B_k y_B^k + R_k y_R^k = r^k \quad (5.3)$$

Si despejamos y_B^k , obtenemos

$$y_B^k = B_k^{-1} r^k - U^k y_R^k \quad (5.4)$$

donde

$$U^k = B_k^{-1} R_k$$

Observación 2. Las matrices B_k no son matrices básicas de todo el problema, sino únicamente de las matrices A_k , y es por esto que podemos asegurar que cada B_k está asociada, graficamente, a un árbol de expansión.

En la restricción (5.1) podemos separar las variables y_k , tal como lo hicimos en (5.3), es decir,

$$\sum_k y_B^k + \sum_k y_R^k \leq b \quad (5.5)$$

Sustituyendo y_B^k en (5.5)

$$\begin{aligned} \sum_k (B_k^{-1} r^k - U^k y_R^k) + \sum_k y_R^k &\leq b \\ \sum_k B_k^{-1} r^k - \sum_k U^k y_R^k + \sum_k y_R^k &\leq b \\ - \sum_k U^k y_R^k + \sum_k y_R^k &\leq b - \sum_k B_k^{-1} r^k \\ \sum_k (y_R^k - U^k y_R^k) &\leq b - \sum_k B_k^{-1} r^k \\ \sum_k (E_k - U^k) y_R^k &\leq b - \sum_k B_k^{-1} r^k \end{aligned}$$

obtenemos:

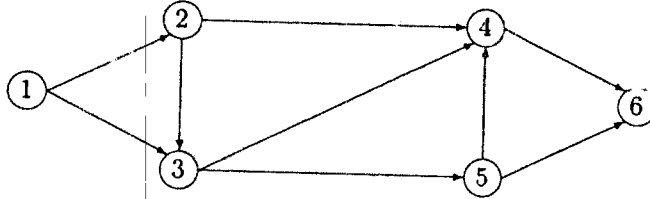
$$\sum_k S_k y_R^k \leq b - \sum_k B_k^{-1} r^k \quad (5.6)$$

donde $S_k = E_k - U^k$ y las columnas de E_k son vectores unitarios que corresponden a las componentes de y_R^k .

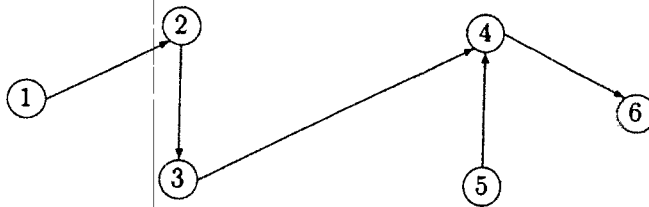
Observación 3. Las restricciones (5.4) y (5.6) son equivalentes a las restricciones (5.1) y (5.2), únicamente hemos separado el conjunto de variables y^k en dos conjuntos, a saber, y_B^k y y_R^k y las restricciones (5.4) y (5.6) representan cómo deben calcularse las variables.

Observación 4. Las columnas de S_k son vectores que representan ciclos elementales en G .

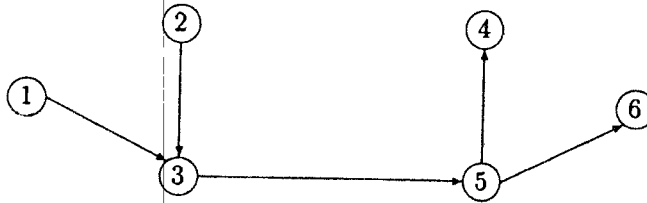
Ejemplo 1. Consideremos la siguiente red para dos bienes:



Para cada bien consideremos las siguientes matrices básicas (correspondientes a las matrices A_k) que están asociadas a los siguientes árboles de expansión:



$$B_1 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix} \quad R_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$



$$B_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \quad R_2 = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

Para el bien 1, la matriz U^1 está determinada por:

$$U^1 = B_1^{-1}R_1 = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ 0 & -1 & -1 & -1 & -1 \\ 0 & 0 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

Para el bien 2, tenemos

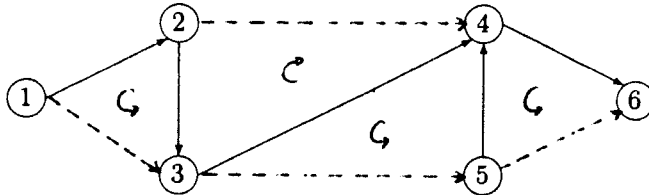
$$U^2 = B_2^{-1}R_2 = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & -1 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Como cada renglón está asociado a un arco, los renglones de las U^k son arcos básicos, así podemos insertar los arcos no básicos con vectores renglón cero. Por lo tanto, podemos encontrar la matriz S_k :

$$S_1 = E_1 - U^1 = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix} - \begin{bmatrix} 1 & & & & & \\ & 1 & 1 & & & \\ & & 1 & 1 & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & -1 & 1 \end{bmatrix} = \begin{bmatrix} -1 & & & & & \\ 1 & & & & & \\ -1 & -1 & & & & \\ & 1 & & & & \\ -1 & -1 & & & & \\ & & 1 & & & \\ & & & -1 & & \\ & & & & 1 & -1 \\ & & & & & 1 \end{bmatrix}$$

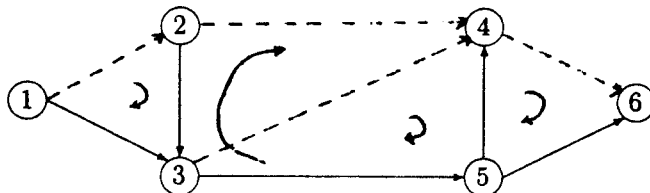
Las entradas restantes son ceros.

Graficamente, S_1 representa los siguientes ciclos:



$$S_2 = E_2 - U^2 = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix} - \begin{bmatrix} 1 & & & & & \\ & -1 & 1 & & & \\ & & 1 & 1 & & \\ & & & 1 & 1 & -1 \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix} = \begin{bmatrix} 1 & & & & & \\ -1 & & & & & \\ 1 & -1 & & & & \\ & 1 & & & & \\ -1 & -1 & & & & \\ & & 1 & & & \\ -1 & -1 & & & & 1 \\ & & & -1 & & \\ & & & & 1 & -1 \end{bmatrix}$$

Los ciclos que S_2 representa son:



Observación 5. Para que las variables y_B^k tomen valores enteros, es necesario que el lado derecho de (5.4) sea entero. Ahora, $B_k^{-1}r^k$ toma valores enteros, porque B_k es totalmente unimodular¹ y r^k es un vector entero; $U^k y_R^k$ es entero solamente si y_R^k es entero, pues la matriz U^k tiene elementos enteros, ya que U^k es el producto de multiplicar dos matrices con elementos enteros ($U^k = B_k^{-1}R_k$).

Por lo tanto, para que y_B^k tome valores enteros es necesario que y_R^k sea entero. Por la restricción (5.6) y_R^k es entero si la matriz $S = [S_1 | \dots | S_q]$ es unimodular (y en particular, que sea totalmente unimodular), pues el lado derecho de la desigualdad es entero.

Todo este razonamiento puede resumirse en la siguiente proposición.

Proposición 1. Si S es totalmente unimodular entonces las variables del problema de flujo de bienes múltiples a costo mínimo, toman valores enteros.

Demostración: Sean B_k matrices básicas de las matrices A_k . Si S es totalmente unimodular, por la restricción (5.6), y_R^k tomará valores enteros y por la restricción (5.4) también y_B^k será entero. Por lo tanto, todas las variables del problema de flujo de bienes múltiples a costo mínimo toman valores enteros. ■

La cuestión inmediata es determinar bajo qué condiciones la matriz S será totalmente unimodular, para poder tener la garantía de la enterabilidad de las soluciones del problema. Para determinar esta caracterización vamos a presentar algunos conceptos.

Definamos una matriz H cuyas columnas son las representaciones vectoriales de todos los ciclos dirigidos (ciclos elementales con una orientación arbitraria) y uniones de ciclos arco-ajenos (uniones de ciclos que no tienen arcos en común) de la gráfica G y cuyos renglones son los arcos de G .

Observación 6: La matriz S es una submatriz de H .

Proposición 2. Si H es totalmente unimodular entonces las variables del problema de flujo de bienes múltiples a costo mínimo, toman valores enteros.

Demostración: Obsérvese primero que bajo la hipótesis, S es totalmente unimodular, pues cualquier submatriz cuadrada A de S lo es también de H y por ser ésta totalmente unimodular, entonces el determinante de A es 0, +1 ó -1.

Entonces por la proposición 1, las variables tomarán valores enteros. ■

- Nuestra pregunta anterior sobre las condiciones para que S sea totalmente unimodular se traduce, ahora en determinar cuáles son los requerimientos para que la matriz H , de una gráfica dada, sea totalmente unimodular; y así tendremos la garantía de que para el problema asociado a esa red, las soluciones son enteras.

Antes de establecer tal caracterización vamos primero a presentar un resultado obtenido por Commoner, que nos va a servir para la demostración que caracterizará a la matriz H .

¹Para obtener la inversa de una matriz se divide entre el determinante, que en este caso, por ser la matriz totalmente unimodular es 0, +1 ó -1 y que a su vez por ser matriz básica el determinante es diferente de 0.

5.2 Desarrollo de Commoner

Sea A una matriz cualquiera tal que todas sus entradas son 0, -1 y $+1$. Se construye una gráfica bipartita $K(A)$ asociando un conjunto de vértices a los renglones y el otro conjunto de vértices a las columnas; un arco entre un par de vértices existe si la correspondiente entrada es diferente de cero, tal arco está dirigido del vértice renglón al vértice columna si la entrada es $+1$ y si la entrada es -1 , está dirigido del vértice columna al vértice renglón.

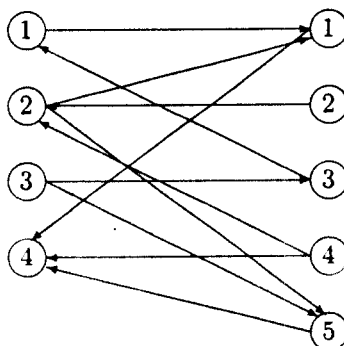
Consideremos de $K(A)$ cualquier ciclo y asignémosle una orientación arbitraria, con respecto a esta orientación cada arco del ciclo tendrá un $+1$ si es directo y un -1 si es inverso.

Definición 1. El *signo* de un ciclo es el producto de los números asociados a los arcos del ciclo, según la orientación dada.

Ejemplo 2. Consideremos la siguiente matriz

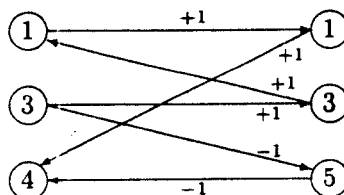
$$A = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ 1 & -1 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ -1 & 0 & 0 & -1 & -1 \end{bmatrix}$$

Entonces la gráfica $K(A)$ es:



Sea el siguiente ciclo de $K(A)$ con la orientación determinada por las etiquetas en los arcos

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



El signo de este ciclo es $(+1)^4(-1)^2 = +1$.

Teorema 1 (Commoner). *Si todo ciclo elemental de $K(A)$ tiene signo $+1$ entonces A es totalmente unimodular.*

Para el lector interesado, la demostración de este teorema se encuentra en [2].

5.3 Presentación de la Operación Binaria “*”

Presentaremos algunos conceptos adicionales antes de establecer la caracterización de H .

Nota. En lo sucesivo hablaremos de H como un conjunto de vectores columna, porque aún cuando H fue definida como una matriz, también puede ser vista como un conjunto de vectores columna cuyos elementos son todos los ciclos elementales y uniones de ciclos arco-ajenos. De esta forma, cuando se escriba $f \in H$ se entenderá que f es una columna de H o bien un elemento del conjunto H .

Sea f un vector que corresponde a una columna de H .

Definición 2. $f(e_j)$ es el coeficiente del arco e_j en f , y se determina de la siguiente manera, asignamos al ciclo f una orientación arbitraria, entonces $f(e_j) = +1$ si el arco e_j es directo y $f(e_j) = -1$ si el arco e_j es inverso con respecto a la orientación dada.

Observación 7. Un ciclo f puede ser representado por la expresión

$$f : \sum_j f(e_j)e_j$$

Definición 3. $|f|$ denotará el conjunto de arcos que pertenecen al ciclo en G , es decir,

$$|f| = \{e_j : f(e_j) \neq 0\}$$

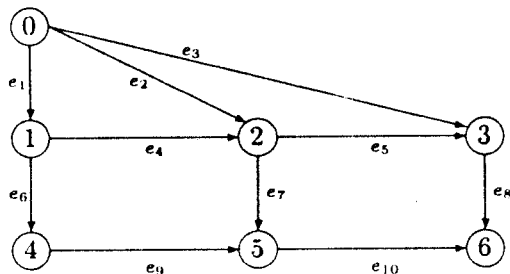
Definase ahora, una operación binaria $*$ entre elementos de H .

Sean f y g dos vectores columna de H con una orientación arbitraria dada para cada uno. Sea $e \in |f| \cap |g|$

$$f * g = \begin{cases} f + g & \text{si } |f| \cap |g| = \emptyset \\ f + g & \text{si } f(e) = +1 \text{ y } g(e) = -1 \text{ o viceversa} \\ f + (-g) & \text{si } f(e) = g(e) = +1 \text{ o } f(e) = g(e) = -1 \end{cases}$$

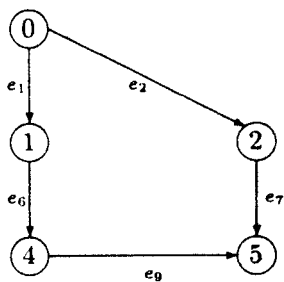
donde $+$ es el operador suma ordinario.

Ejemplo 3. Consideremos la siguiente gráfica:

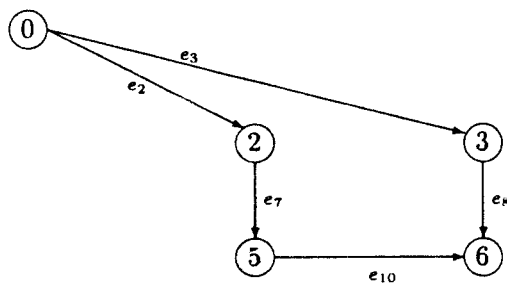


Sean

$$f : e_1 + e_6 + e_9 - e_7 - e_2$$



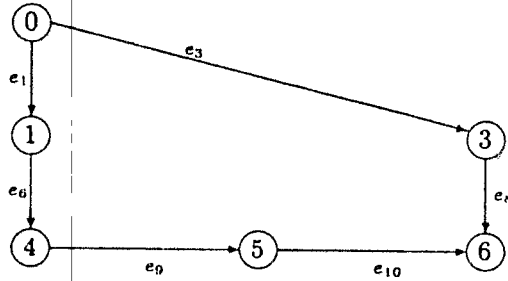
$$g : e_2 + e_7 + e_{10} - e_8 - e_3$$



$$|f| \cap |g| = \{e_2, e_7\}$$

Como e_2 y e_7 tienen coeficientes diferentes en ambos ciclos entonces

$$\begin{aligned} f * g : & \quad e_1 \quad -e_2 \quad \quad +e_6 \quad -e_7 \quad \quad +e_9 \\ & \quad \quad e_2 \quad -e_3 \quad \quad +e_7 \quad -e_8 \quad \quad +e_{10} \\ = & \quad e_1 \quad \quad -e_3 \quad +e_6 \quad \quad -e_8 \quad +e_9 \quad +e_{10} \end{aligned}$$

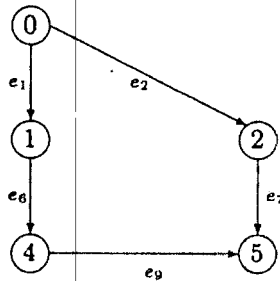


Obsérvese que el resultado de $f * g$ es también un elemento de H .

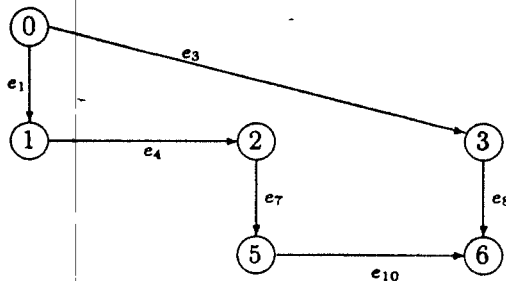
Definición 4. El sistema algebraico $(H, *)$ se dice que es *cerrado*, si para cualesquiera $f, g \in H$ se tiene que existe una orientación para ambos ciclos, tales que $f * g \in H$.

Observación 8. Para algunas gráficas esta propiedad de cerradura no siempre se satisface, por ejemplo, en la gráfica anterior, sean

$$f : e_1 + e_6 + e_9 - e_7 - e_2$$



$$g : e_1 + e_4 + e_7 + e_{10} - e_8 - e_3$$

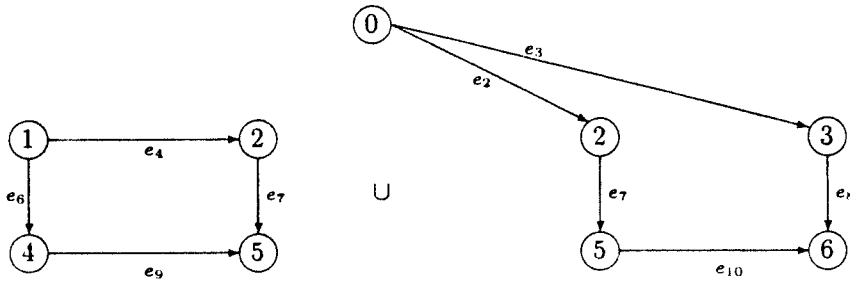


$$|f \cap g| = \{e_1, e_7\}$$

e_1 tiene coeficientes iguales en ambos ciclos, por lo tanto,

$$\begin{aligned}
 f * g: & \quad e_1 & -e_2 & & +e_6 & -e_7 & & +e_9 \\
 & -e_1 & & +e_3 & -e_4 & & -e_7 & +e_8 & -e_{10} \\
 = & & -e_2 & +e_3 & -e_4 & +e_6 & -2e_7 & +e_8 & +e_9 & -e_{10}
 \end{aligned}$$

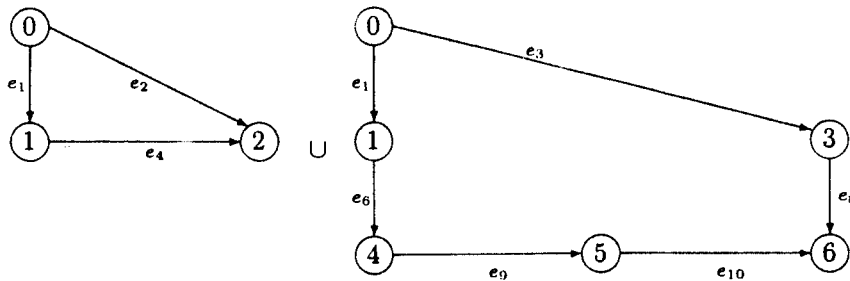
Como el coeficiente de e_7 es 2, se concluye que el resultado es una unión de ciclos que no son arco-ajenos, y este tipo de ciclos no son elementos de H , así se tiene que $f * g \notin H$.



Ahora, e_7 tiene coeficientes diferentes en f y g , así tenemos que

$$\begin{aligned}
 f * g: & \quad e_1 & -e_2 & & +e_6 & -e_7 & & +e_9 \\
 & e_1 & & -e_3 & +e_4 & & +e_7 & -e_8 & +e_{10} \\
 = & 2e_1 & -e_2 & -e_3 & +e_4 & +e_6 & & -e_8 & +e_9 & +e_{10}
 \end{aligned}$$

Igualmente tenemos coeficiente de valor 2 en el arco e_1 , esto significa que el resultado $f * g$ es una unión de ciclos que no son arco-ajenos, es decir, $f * g \notin H$.



Obsérvese que no existe una orientación adecuada para ninguno de los ciclos tal que permita tener que $f * g \in H$. Por lo tanto, $(H, *)$ no es cerrado para esta gráfica.

Una forma gráfica-teórica de interpretar el operador $*$, es asignar la etiqueta (+) o (-) a los arcos e_j en cada ciclo dependiendo si el coeficiente de e_j es +1 ó -1, respectivamente.

Proposición 3. Sea $E = \{e_j : e_j \in |f| \cap |g|\}$. Todo elemento de E tiene o etiquetas idénticas, o etiquetas diferentes en ambos ciclos, si y sólo si $f * g \in H$.

Demostración: Necesidad. Si $E = \emptyset$ entonces $f * g \in H$ pues como no hay arcos en común $f * g$ es unión de ciclos arco-ajenos. Supongamos $E \neq \emptyset$ y supongamos que todo elemento de E tiene etiquetas iguales, entonces $f * g = f + (-g)$ y por lo tanto, todos los arcos que son elementos de E se cancelan, así tenemos un ciclo o unión de ciclos arco-ajenos y por lo tanto $f * g \in H$. Análogamente, si todo elemento de E tiene etiquetas diferentes entonces $f * g = f + g$ y todos los elementos de E se cancelan y el resultado es un ciclo o unión de ciclos arco-ajenos, es decir, $f * g \in H$.

Suficiencia. Supongamos que para cualesquiera f y $g \in H$, se tiene que $f * g \in H$. La demostración se hará por contradicción: supongamos que existe un arco $e_1 \in E$ tal que tiene etiquetas iguales en ambos ciclos y que existe otro arco $e_2 \in E$ tal que tiene etiquetas diferentes en los ciclos. Sin pérdida de generalidad, supongamos que $f(e_1) = g(e_2) = +1$ y que $f(e_2) = +1$ y $g(e_2) = -1$. El ciclo f puede representarse como:

$$f : e_1 + e_2 + \sum_{\substack{e \in f \\ e \neq e_1, e_2}} f(e)e$$

de igual manera el ciclo g se representa por:

$$g : e_1 - e_2 + \sum_{\substack{e \in g \\ e \neq e_1, e_2}} g(e)e$$

Se define la operación $*$ para el arco e_1 , como éste tiene etiquetas iguales, se tiene que:

$$\begin{aligned} f * g &= f + (-g) \\ &: e_1 + e_2 + \sum_{\substack{e \in f \\ e \neq e_1, e_2}} f(e)e - e_1 + e_2 - \sum_{\substack{e \in g \\ e \neq e_1, e_2}} g(e)e \\ &= 2e_2 + \sum_{\substack{e \in f \\ e \neq e_1, e_2}} f(e)e - \sum_{\substack{e \in g \\ e \neq e_1, e_2}} g(e)e \end{aligned}$$

El coeficiente del arco e_2 es 2, es decir, el resultado es una unión de ciclos que no son arco-ajenos, i.e. $f * g \notin H$. Esto contradice la hipótesis. De forma análoga, si se define la operación $*$ para el arco e_2 se tiene que:

$$\begin{aligned} f * g &= f + g \\ &: e_1 + e_2 + \sum_{\substack{e \in f \\ e \neq e_1, e_2}} f(e)e + e_1 - e_2 + \sum_{\substack{e \in g \\ e \neq e_1, e_2}} g(e)e \\ &= 2e_1 + \sum_{\substack{e \in f \\ e \neq e_1, e_2}} f(e)e + \sum_{\substack{e \in g \\ e \neq e_1, e_2}} g(e)e \end{aligned}$$

Y se observa que el resultado es una unión de ciclos que no son arco-ajenos, pues el coeficiente de e_1 es 2, esto implica que $f * g \notin H$. Contradicción a la hipótesis. Por lo tanto, todos los elementos de E tienen o etiquetas iguales o etiquetas diferentes en ambos ciclos. ■

5.4 Caracterización Algebraica

Ahora sí, ya estamos preparados para presentar en qué casos la matriz H es totalmente unimodular, y así tener la garantía de la enterabilidad de las soluciones al problema de flujo de bienes múltiples a costo mínimo; la caracterización se presenta a continuación desde un punto de vista algebraico.

Teorema 2. Si el sistema algebraico $(H, *)$ es cerrado entonces $(H, *)$ es un grupo.

Demostración: Supongamos que $(H, *)$ es cerrado.

- Asociatividad:

Sean $f, g, h \in H$, por ser cerrado, $f * g = k \in H$ y $g * h = l \in H$

$$\begin{aligned} (f * g) * h &= (f \pm g) * h \\ &= (f \pm g) \pm h \\ &= f \pm (g \pm h) \\ &= f * (g \pm h) \\ &= f * (g * h) \end{aligned}$$

Pues la operación de adición es asociativa.

Por lo tanto, $*$ es asociativa.

- Existencia del idéntico:

El vector $\underline{0} \in H$ es el elemento identidad, pues como $|f| \cap |\underline{0}| = \emptyset$ se tiene que

$$\begin{aligned} f * \underline{0} &= f + \underline{0} = f \\ \underline{0} * f &= \underline{0} + f = f \end{aligned}$$

- Existencia del inverso:

El inverso de f es $f^{-1} = -f \in H$.

Tenemos que $|f| \cap |-f| = |f|$ y todos los arcos tienen etiquetas diferentes, entonces

$$\begin{aligned} f * f^{-1} &= f * (-f) = f + (-f) = \underline{0} \\ f^{-1} * f &= (-f) * f = (-f) + f = \underline{0} \end{aligned}$$

Por lo tanto, $(H, *)$ es grupo. ■

Teorema 3. Si el sistema algebraico $(H, *)$ es grupo entonces la matriz H es totalmente unimodular.

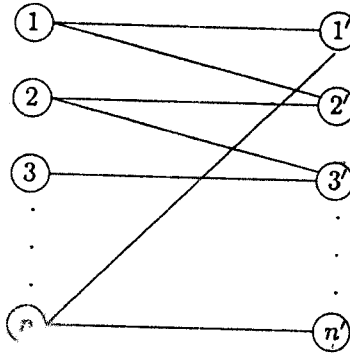
Demostración: Supongamos que $(H, *)$ es grupo. La demostración se hará por contradicción; supongamos que H no es totalmente unimodular entonces, por el teorema de Commoner (utilizando la contrapuesta), existe un ciclo en $K(H)$ con signo -1 . Se demostrará que esto implica que H no es grupo.

Construyamos la gráfica bipartita $K(H)$, según el desarrollo de Commoner. Un ciclo elemental de $2n$ vértices de $K(H)$ tiene la siguiente representación matricial

$$A = \begin{matrix} & \begin{matrix} 1' & 2' & 3' & \cdots & (n-1)' & n' \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ \vdots \\ n-1 \\ n \end{matrix} & \begin{bmatrix} 1 & a_{12} & & & & \\ & a_{22} & a_{23} & & & \\ & & a_{33} & & & \\ & & & \ddots & & \\ & & & & a_{n-1,n-1} & a_{n-1,n} \\ a_{n1} & & & & & a_{nn} \end{bmatrix} \end{matrix}$$

donde $a_{ij} = \pm 1$, según la correspondiente entrada en H y las entradas restantes son cero.

El ciclo representado en la siguiente figura corresponde al ciclo de la matriz A .



El arco $(1, 1')$ se toma en dirección de 1 a $1'$, las otras direcciones se determinan por los valores de a_{ij} .

Orientemos el ciclo con la dirección determinada por el arco $(1, 1')$, los arcos que pertenecen a la diagonal principal tendrán una etiqueta $(+)$ si su dirección es del vértice renglón al vértice columna, es decir, si su entrada $a_{ii'} = +1$, y tendrán etiqueta $(-)$ si su correspondiente entrada es $a_{ii'} = -1$. Los arcos que están fuera de la diagonal principal tendrán etiqueta $(+)$ si están dirigidos del vértice columna al vértice renglón, es decir, $a_{ij'} = -1$ y tendrán etiqueta $(-)$ si $a_{ij'} = +1$. Por lo tanto, el signo del ciclo está determinado por:

$$\begin{aligned}
 \text{signo} &= \underbrace{(+1)(-1)^r}_{\text{diagonal principal}} \underbrace{(+1)(-1)^s}_{\text{fuera de la diagonal ppl.}}
 \end{aligned}$$

Así, el ciclo tendrá $\text{signo} = -1$ si r es impar y s par, o bien si r es par y s impar. En la matriz A esto se traduce en tener un número impar de elementos negativos en la diagonal principal y un número par de elementos positivos fuera de ella, o bien, tener un número par de elementos negativos en la diagonal principal y un número impar de elementos positivos fuera de ella.

Multipliquemos las columnas de la matriz A por -1 , de tal manera que en cada renglón se tenga un $+1$ y un -1 .

Obsérvese que el signo del ciclo no cambia, pues al multiplicar una columna por -1 estamos cambiando el signo a dos elementos: uno que pertenece a la diagonal principal y otro que está fuera de ella, de esta forma si teníamos, por ejemplo, un número impar de elementos negativos en la diagonal principal y un número par de elementos positivos fuera, entonces al multiplicar una columna por -1 tendremos un número par de elementos negativos en la diagonal principal y un número impar de elementos positivos fuera de ella, y el signo del ciclo sigue siendo -1 .

Graficamente, lo que nos queda es que en los vértices renglón tenemos un arco entrante y un arco saliente, lo que, según sea la orientación del ciclo, significa que ambos arcos tienen etiquetas $(+)$ o bien $(-)$. Como el ciclo es par (se tienen $2n$ arcos) y el signo del ciclo es -1 debe existir un vértice renglón tal que sus dos arcos incidentes sean entrantes o salientes, así, en la orientación del ciclo tendrán etiquetas diferentes. En la matriz A esto significa que siempre existirá un renglón con sus dos entradas diferentes de cero de igual signo.

Fijémonos en el renglón que tiene sus dos entradas iguales y en sus columnas adyacentes:

$$\begin{array}{c} f_1 \quad f_2 \quad f_3 \\ e_1 \left[\begin{array}{ccc} \pm 1 & \mp 1 & \\ e_2 & \pm 1 & \pm 1 \\ e_3 & & \mp 1 \quad \pm 1 \end{array} \right] \end{array}$$

Como A es un ciclo de la gráfica $K(H)$, los renglones están asociados a arcos de G y las columnas a ciclos de H entonces podemos operar los ciclos con $*$ y como $(H, *)$ es un grupo, la regla de asociatividad debe ser válida:

$$(f_2 * f_1) * f_3 = f_2 * (f_1 * f_3)$$

donde

$$\begin{aligned} f_1 &: \pm e_1 + e \\ f_2 &: \mp e_1 \pm e_2 \\ f_3 &: \pm e_2 \mp e_3 \end{aligned}$$

operando tenemos,

$$\begin{aligned} f_2 * f_1 &: \pm e_1 \pm e_2 \mp e_1 + e \\ &= \pm e_2 + e \end{aligned}$$

$$\begin{aligned} (f_2 * f_1) * f_3 &: \pm e_2 + e \mp e_2 \pm e_3 \\ &= \pm e_3 + e \end{aligned}$$

Ahora calculamos

$$f_1 * f_3 : \pm e_1 \pm e_2 \mp e_3 + e$$

Pero

$$f_2 * (f_1 * f_3) \notin H$$

pues los arcos en común, e_1 y e_2 , no tienen ambos etiquetas iguales o diferentes.

Obsérvese que $f_1 * f_3$ es una unión de ciclos arco-ajenos; buscar una orientación para $f_1 * f_3$ de tal manera que $f_2 * (f_1 * f_3) \in H$ se traduce en multiplicar una de las dos columnas asociadas a f_1 y a f_3 por -1 , por ejemplo,

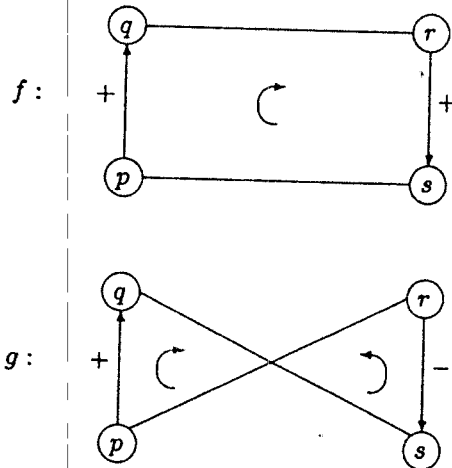
$$\begin{array}{c} \downarrow \\ e_1 \left[\begin{array}{ccc} f_1 & f_2 & f_3 \\ \pm 1 & \mp 1 & \\ e_2 & \pm 1 & \mp 1 \\ e_3 & & \pm 1 \quad \pm 1 \end{array} \right] \end{array}$$

Pero ahora tenemos que el renglón asociado a e_3 tiene sus dos entradas iguales y podríamos aplicar el razonamiento anterior y llegaríamos otra vez a que la asociatividad no es válida. Por lo tanto, $(H, *)$ no es grupo, esto contradice la hipótesis. Por lo tanto, H es totalmente unimódular. ■

Con estos teoremas, para tener la garantía de que una red de bienes múltiples tiene soluciones enteras, tenemos que verificar que sistema $(H, *)$ sea cerrado. Sin embargo, esto implica verificar la cerradura para todo par de ciclos y esto es un trabajo muy laborioso que llevaría demasiado tiempo. Otra opción, es verificar que $(H, *)$ no sea cerrado, es decir, buscar un par de ciclos f y g , tales que $f * g \notin H$, para ello se presenta la siguiente proposición.

Proposición 4. Si $(H, *)$ no es cerrado entonces debe existir un par de arcos (p, q) y (r, s) tales que exista un par de cadenas arco-ajenas, una de p a s y otra de q a r , y también un par de cadenas arco-ajenas de p a r y de q a s , sin que ninguna de las cuatro cadenas contenga los arcos (p, q) y (r, s) .

Demostración: Supongamos que $(H, *)$ no es cerrado entonces existen dos ciclos f y g tales que $f * g \notin H$. Tomemos dos arcos que estén contenidos en $E = \{e_j : e_j \in |f| \cap |g|\}$, tales que uno de ellos tenga etiquetas iguales y el otro tenga etiquetas diferentes en ambos ciclos (este tipo de arcos existen por la proposición 3) sean éstos (p, q) y (r, s) , sin pérdida de generalidad, supongamos que el arco (p, q) es el que tiene las etiquetas iguales en ambos ciclos, digamos $(+)$ y el arco (r, s) tiene etiquetas diferentes, digamos $(+)$ en f y $(-)$ en g .



Siguiendo la orientación del ciclo en f , como ambos arcos son directos, entonces hay una cadena de p a s y otra de q a r , sin que ninguna contenga los arcos (p, q) y (r, s) , además como f es un ciclo elemental o unión de ciclos arco-ajenos, estas cadenas son arco-ajenas. En el ciclo g , como el arco (p, q) es directo y el arco (r, s) es inverso, hay dos cadenas una de p a r y otra de q a s , ambas cadenas arco-ajenas y sin contener los arcos (p, q) y (r, s) . ■

A pesar de que la verificación de la cerradura o no cerradura de H no es un trabajo sencillo, ni siquiera computacionalmente hablando, pueden existir redes con una estructura especial que permiten demostrar que $(H, *)$ es o no cerrado. Para ejemplificar esto, vamos a analizar las siguientes redes.

Evans establece en [6] que las redes asociadas al problema de transporte $PTBM(m, 2, q)$ (o $PTBM(2, n, q)$) (figura 1) y al problema de planeación de producción dinámica de bienes múltiples de un solo nivel con T periodos (figura 2) satisfacen que para cualesquiera dos ciclos f y $g \in H$ se verifica fácilmente que $f * g \in H$.

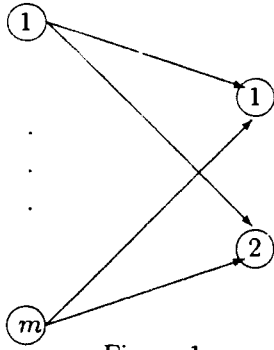


Figura 1

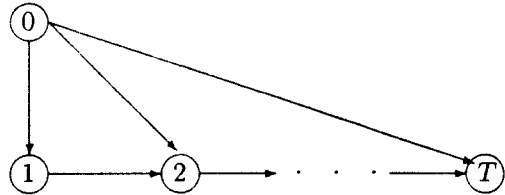


Figura 2

Proposición 5. En toda red asociada al $PTBM(m, 2, q)$ (o $PTBM(2, n, q)$) sucede que $(H, *)$ es cerrado.

Demostración: Primero obsérvese que cualquier ciclo elemental de la gráfica tiene exactamente cuatro arcos, pues se necesita uno del origen r al destino 1, un segundo del destino 1 al origen s , un tercero del origen s al destino 2 y un cuarto del destino 2 al origen r .

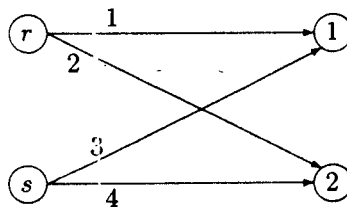
Sean f y g dos ciclos independientes de $PTBM(m, 2, q)$ entonces

$$p = ||f| \cap |g|| = 0, 1, 2.$$

Pues si $p = 4$ entonces $f = g$ y no serían independientes. Por otro lado, cualquier conjunto de tres arcos es un subconjunto de un ciclo de $PTBM(m, 2, q)$ y forma un árbol sobre el conjunto de vértices incidentes de $PTBM(m, 2, q)$. Entonces existe un único arco que completa el ciclo, por lo tanto, $f = g$ o uno de los dos no es ciclo, así $p \neq 3$.

Si $p = 0$ ó 1 entonces claramente $f * g \in H$, de hecho esto es verdad para cualquier gráfica.

Queda solamente por considerar el caso en que $p = 2$. Sea f cualquier ciclo elemental de $PTBM(m, 2, q)$



Sea g un ciclo tal que tenga dos arcos en común con f , éstos pueden ser (1,3) o (2,4) únicamente. Para ambos casos, $f * g \in H$, pues podemos asignar la orientación a los ciclos por la dirección determinada del arco del primer origen del ciclo hacia el destino 1. Entonces, se tendrá que los arcos en la intersección tienen etiquetas iguales.

Para el caso en que f y/o g son uniones de ciclos arco-ajenos podemos considerar para cada ciclo de la unión la misma orientación dada anteriormente y se verifica que $f * g \in H$. ■

Proposición 6. *La red asociada al problema de planeación de producción dinámica de bienes múltiples de un nivel y T periodos presenta que $(H, *)$ es cerrado.*

Demostración: Para cualquier par de ciclos elementales, la orientación puede ser la determinada por la dirección de los arcos de la forma (i, j) con $i < j$, es decir, éstos tendrán etiquetas $(+)$, de esta forma puede verificarse que $f * g \in H$.

Para el caso en que alguno de los dos ciclos es unión de ciclos arco-ajenos, siempre es posible encontrar una orientación (la misma descrita anteriormente para cada uno de los ciclos que componen el ciclo) tal que al operarlos con $*$, el resultado sea también un elemento de H . ■

Observación 9. Para la generalización del problema anterior a niveles múltiples, $(H, *)$ es no cerrado, como se verificó en la observación 8 referente al ejemplo 3.

Conclusiones

El propósito de este trabajo fue el estudio y análisis de una generalización del problema de flujo de bienes múltiples a costo mínimo. Éste consistió en la presentación (planteamiento) del problema y en la exposición de un método de resolución, así como el análisis de dos casos particulares del problema; además, de exponer las condiciones bajo las cuales se garantiza la existencia de soluciones enteras para el problema, de tal manera que pudiera plantearse algún otro método de resolución (de redes) que sea más eficiente que el aquí tratado.

Para el planteamiento matemático, se presentaron dos tipos de formulaciones: la de nodo-arco y la de arco-cadena. Podría decirse que éstas son las únicas formas para plantear el problema; sin embargo, dentro de las mismas formulaciones pueden adaptarse cambios o re-planteamientos en miras de que el método de resolución sea menos laborioso. Dentro de la bibliografía estudiada se encontraron varios métodos y algoritmos de resolución, tanto para el caso de minimizar costos, como para el de maximizar la cantidad de flujo. En ambos casos se planteó el problema con la formulación nodo-arco o arco-cadena, con o sin modificaciones. En este trabajo se expuso uno de estos métodos debido a que el problema presenta la estructura especial requerida para aplicar el principio de descomposición de Dantzig-Wolfe, el cual proporciona un procedimiento sistemático para resolver problemas lineales de gran escala. Además, la resolución de los subproblemas no causó mayor dificultad, pues se verificó que éstos fuesen problemas del tipo flujo de un solo bien o de encontrar la ruta más corta, para los cuales existen varios métodos de resolución ampliamente conocidos. También otro de los métodos de resolución encontrados (el propuesto por [30]), está basado en este principio. Éste utiliza la formulación arco-cadena con una modificación que permite que el cálculo de la base y del vector de términos independientes sea menos complicado.

El estudio de problemas particulares es importante, pues precisamente por la característica especial que presentan, puede replantearse el problema para tener un método de resolución que haga que el cálculo de la solución sea más eficiente. En este trabajo, para los dos casos particulares presentados, se demostró una equivalencia con el problema de flujo de un solo bien, para el cual existen varios algoritmos que lo resuelven.

Este trabajo constituye parte de un proyecto más amplio que comprendería el estudio de los métodos de marcaje en redes existentes para el problema, la comparación de los diferentes métodos de resolución, la implementación en computadora y la aplicación a un problema concreto. Además, podría hacerse un análisis de sensibilidad a la solución óptima, correspondiente al método de resolución empleado.

Apéndice A

Conceptos Básicos

A continuación se introducen algunos conceptos que se manejan en este trabajo, la sección A.1 fue tomada de [10] y las secciones A.2 y A.3 de [18].

A.1 Grupos

Definición 1. Una *operación binaria* $*$ en un conjunto, es una regla que asigna a cada par ordenado de elementos de un conjunto, algún elemento del conjunto.

Definición 2. Una operación binaria $*$ en un conjunto S es *asociativa* si y sólo si

$$(a * b) * c = a * (b * c) \quad \forall a, b, c \in S.$$

Observación 1. Al definir una operación binaria $*$ en un conjunto S debemos estar seguros de que

- i) se asigne exactamente un elemento a cada par ordenado posible de elemento de S
- ii) para cada par ordenado de elementos de S , el elementos asignado esté en S .

Definición 3. Un grupo $(G, *)$ es un conjunto G , junto con una operación binaria $*$ en G , tal que se satisfacen los siguientes axiomas:

- i) La operación binaria $*$ es asociativa.
- ii) Existe un elemento e en G tal que

$$e * x = x * e = x \quad \forall x \in G.$$

Este elemento e es un *elemento identidad* para $*$ en G .

- iii) Para cada a en G existe un elemento a' en G con la propiedad de que

$$a' * a = a * a' = e.$$

El elemento a' es un *inverso de a respecto a $*$* .

A.2 Teoría de Gráficas

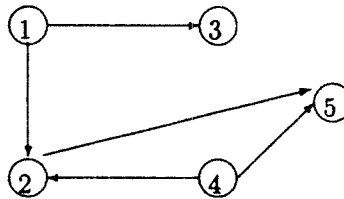
Definición 4. Una *gráfica* $G = (N, A)$ es la pareja constituida por:

- Un conjunto finito N llamado conjunto de *vértices* y
- un conjunto $A \subset N \times N$ llamado conjunto de *arcos*.

Ejemplo 1.

$$G = (\{1, 2, 3, 4, 5\}, \{(1, 2), (1, 3), (2, 2), (2, 5), (4, 2), (4, 5)\}).$$

Por abuso del lenguaje confundiremos, en general, una gráfica y su representación gráfica. La representación gráfica de G es:



Definición 5. Una gráfica en la cual la dirección de los arcos no está especificada se llamará una *gráfica no dirigida* y se denotará $G = (N, U)$, en este caso, U es el conjunto de *aristas*.

Definición 6. Una *red* es una gráfica con “cierta información” en los arcos y se denotará $R = (N, A, d_i)$ donde $d_i : A \rightarrow \mathbf{R}$ o bien $d : N \rightarrow \mathbf{R}$.

Definición 7. Dado un arco $e = (x, y)$, llamaremos a x el vértice *inicial* y a y el vértice *terminal* del mismo; x y y serán los vértices *extremos* del arco e .

Definición 8. Un vértice x y un arco e se dicen *incidentes* si el vértice x es un vértice extremo del arco e .

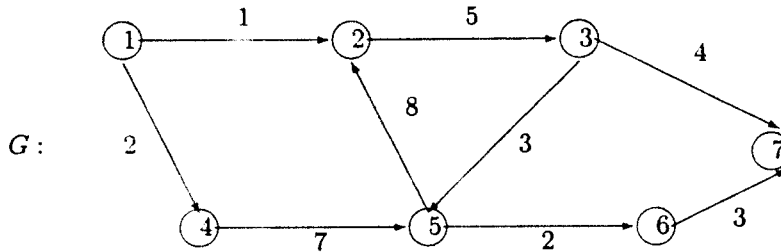
Definición 9. Un *camino* c es una sucesión alternada de vértices y arcos tal que para cada arco e en la sucesión, el vértice que lo precede en la sucesión corresponde a su vértice inicial y el vértice que lo sucede en la sucesión corresponde a su vértice terminal.

Definición 10. Una *cadena* C es una sucesión alternada de vértices y arcos tal que los vértices que encuadran a cada arco corresponden a sus vértices extremos. El vértice x_1 se llamará vértice *inicial* de la cadena y el vértice x_{n+1} se llamará vértice *terminal* de ésta.

Observación 2. Todo camino es una cadena.

Definición 11. La *longitud* de una cadena C es $\sum_{e \in C} \ell(e)$, donde $\ell(e)$ es la longitud del arco e .

Ejemplo 2.



Un camino de G es

$$c_1 = \{1, (1, 4), 4, (4, 5), 5, (5, 2), 2, (2, 3), 3, (3, 7), 7\}$$

$$\ell(c_1) = 2 + 7 + 8 + 5 + 4 = 26$$

Una cadena de G es

$$C_2 = \{1, (1, 2), 2, (5, 2), 5, (4, 5), 4\}$$

$$\ell(C_2) = 1 + 8 + 7 = 16$$

Definición 12. Cadena (Camino) *simple* es aquella que no pasa dos veces por el mismo arco.

Definición 13. Cadena (Camino) *elemental* es aquella que no pasa dos veces por el mismo vértice.

Observación 3. Toda cadena elemental es simple.

Definición 14. Un *ciclo* es una cadena cuyos vértices inicial y terminal coinciden.

Definición 15. Un *circuito* es un camino cuyos vértices inicial y terminal coinciden.

Observación 4. Todo circuito es un ciclo.

Ejemplo 3.

En la gráfica anterior, un ciclo de G es

$$\{1, (1, 4), 4, (4, 5), 5, (5, 2), 2, (1, 2), 1\}.$$

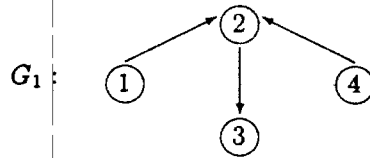
Un circuito de G es

$$\{2, (2, 3), 3, (3, 5), 5, (5, 2), 2\}.$$

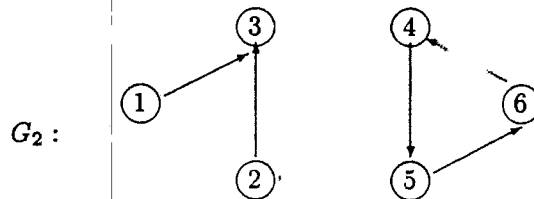
Definición 16. Una gráfica se dice *conexa* si existe una cadena que une todo par de vértices distintos en la gráfica.

Ejemplo 4.

La gráfica G_1 es conexa:



La gráfica G_2 no es conexa, ya que no existe cadena que una los vértices 2 y 4.



Una gráfica puede ser vista como un conjunto de gráficas conexas. Cada una de estas gráficas conexas se llama *componente conexa* de la gráfica original.

Definición 17. Una *subgráfica* de G se denotará $G_S = (N_S, A_S)$, donde

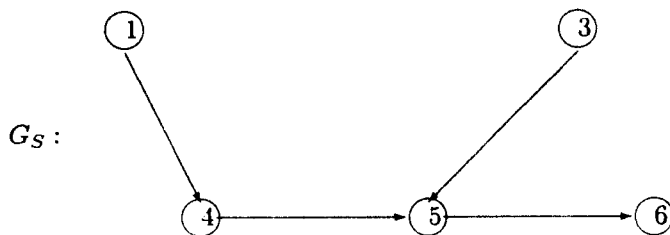
$$N_S \subset N$$

y $\forall x_i, x_j \in N_S, (x_i, x_j) \in A_S.$

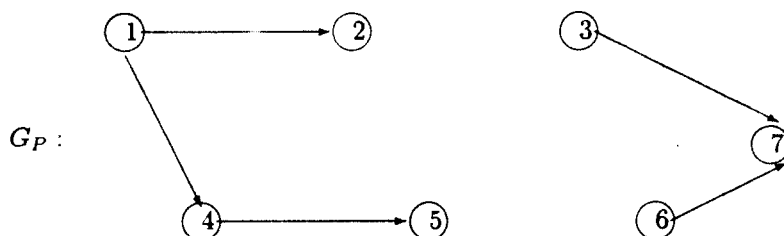
Definición 18. Una *gráfica parcial* de G se denotará $G_P = (N, A_P)$, donde $A_P \subset A.$

Ejemplo 5.

Una subgráfica de G es:



Una gráfica parcial de G es



Definición 19. Llamaremos conjunto de vértices *sucesores* de un vértice x al conjunto:

$$\Gamma^+(x) = \{y : (x, y) \in A\}$$

y se llamará *grado exterior* a la cardinalidad del conjunto de sucesores, i.e.

$$g^+(x) = |\Gamma^+(x)|.$$

Definición 20. Llamaremos conjunto de vértices *predecesores* de un vértice x al conjunto:

$$\Gamma^-(x) = \{z : (z, x) \in A\}$$

y se llamará *grado interior* a la cardinalidad del conjunto de predecesores, i.e.

$$g^-(x) = |\Gamma^-(x)|.$$

Ejemplo 6.

En la gráfica G , tenemos que:

$$\Gamma^+(5) = \{2, 6\},$$

$$\Gamma^-(5) = \{3, 4\},$$

$$\Gamma^+(7) = \emptyset,$$

$$\Gamma^-(1) = \emptyset.$$

Definición 21. El *grado* de un vértice x se denotará $g_G(x)$ y se define

$$g_G(x) = g^+(x) + g^-(x) = |\Gamma^+(x)| + |\Gamma^-(x)|$$

Definición 22. Un vértice x se llama *pendiente* si $g_G(x) = 1$.

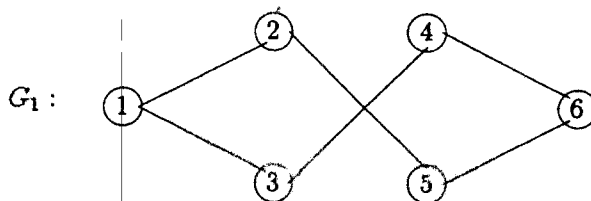
Definición 23. Una gráfica $G = (N_1, N_2, A)$ es *bipartita* si

$$N = N_1 \cup N_2, \quad N_1 \cap N_2 = \emptyset$$

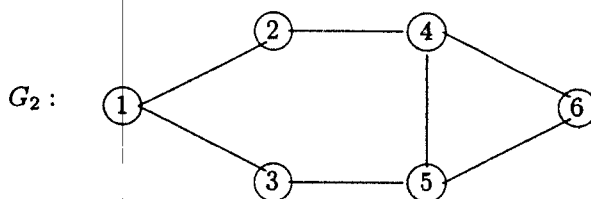
$$\forall (x, y) \in A, \quad x \in N_1, y \in N_2 \text{ o bien } x \in N_2, y \in N_1.$$

Ejemplo 7.

La gráfica G_1 es bipartita, pues $N_1 = \{1, 4, 5\}$ y $N_2 = \{2, 3, 6\}$



La gráfica G_2 no es bipartita, pues si lo fuera, ponemos el vértice 1 en N_1 por tanto los vértices 2 y 3 deben estar en N_2 , lo cual implica que los vértices 4 y 5 deben estar en N_1 y esto es una contradicción, pues si el vértice 4 está en N_1 el vértice 5 debe estar en N_2 (el vértice 5 no puede estar en N_1 y en N_2).



Definición 24. Un *árbol* es una gráfica conexa y sin ciclo. O bien, sin ciclo y $(n - 1)$ arcos, donde n es el número de vértices. O bien, conexa y $(n - 1)$ arcos.

Definición 25. Un *árbol de expansión* de una gráfica G es un árbol que contiene todos los vértices de la gráfica G .

Claramente, no puede existir un árbol de expansión en una gráfica con más de una componente conexa y toda gráfica conexa contiene un árbol de expansión.

Definición 26. Un vértice a de una gráfica es una *raíz* si existe en G un camino que una a a con x , $\forall x \in X$.

Definición 27. Una gráfica $G = (X, U)$ sobre $n \geq 2$ vértices es una *arborescencia de raíz a* si

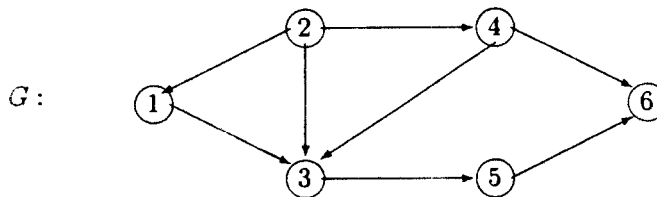
- i) a es una raíz de G ,
- ii) G es un árbol.

Observación 5. Una arborescencia es un árbol, pero el recíproco es falso en general.

Definición 28. La *matriz de incidencia* de $G = (N, A)$ donde $|N| = n$ y $|A| = m$ es una matriz $n \times m$ tal que

$$a_{ij} = \begin{cases} +1, & \text{si el vértice } i \text{ es el vértice inicial del arco } j, \\ -1, & \text{si el vértice } i \text{ es el vértice terminal del arco } j, \\ 0, & \text{en otro caso.} \end{cases}$$

Ejemplo 8.



La matriz de incidencia de G es:

$$A = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{pmatrix}$$

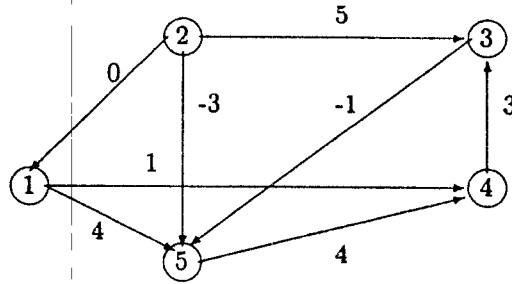
Definición 29. *Punto de articulación* es un vértice tal que al suprimirlo de la gráfica aumenta el número de componentes conexas de la misma.

A.3 Ruta más Corta

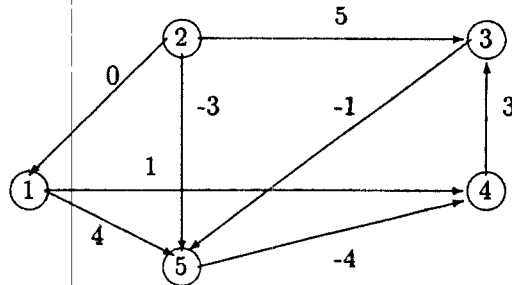
Definición 30. Un circuito C tal que $\sum_{u \in C} d(u) < 0$ se llama *circuito absorbente*.

Observación 6. Sea C un circuito absorbente de una red $R = (N, A, d)$ supongamos que existe un camino c_o^x de o a un vértice x del circuito C , y un camino c_x^d de un vértice x del circuito al d . Entonces el camino obtenido recorriendo el camino c_o^x después el circuito C , k veces y c_x^d al final, es un camino de o a d de longitud negativa tan grande como queramos (tomando k suficientemente grande en valor absoluto). El problema de búsqueda de un camino más corto no tiene solución finita en este caso.

Ejemplo 9. Circuito no absorbente



Circuito absorbente



$$d(C) = -2 < 0$$

Observación 7. Como un ejemplo de valores negativos en una gráfica consideremos lo siguiente: al construir una carretera que conecte lugares por donde pasaba un tren, podemos anotar el tiempo que se ahorra al hacer el cambio.

$$7h. - 10h. = -3 \quad \text{se tiene un ahorro}$$

$$7h. - 7h. = 0$$

$$7h. - 5h. = 2 \quad \text{se tiene un gasto}$$

Teorema 1. (Principio de Optimalidad) Sea C un camino más corto en la red $R = (N, A, d)$ de un vértice o a un vértice d y sea c_{xy}^C la porción de ese camino situada entre dos vértices x y y . Entonces c_{xy}^C es un camino más corto de x a y .

Demostración: Si existiera de x a y un camino \bar{c} más corto que c_{xy}^C , la unión de c_o^C, \bar{c}, c_y^d daría un camino más corto que C , lo que es imposible. ■

Algoritmo General

La paternidad de este algoritmo es delicada para establecer, hay que citar seguramente a Ford, Fulkerson y Dantzig, tal vez otros deban citarse.

Este algoritmo permite poner en evidencia un circuito absorbente si existe, o bien, una arborescencia de caminos más cortos de raíz o .

El principio del algoritmo consiste en “mejorar” una arborescencia de raíz o hasta obtener una arborescencia de caminos más cortos.

Se ha escogido presentar este algoritmo por las razones siguientes:

- Este algoritmo es simple para presentar y fácil de comprender.
- En sus principios es muy similar al algoritmo simplex.
- El procedimiento central de este algoritmo consiste en mejorar una solución realizable inicial. Este método es, pues, muy interesante cuando se trata de *actualizar* una arborescencia de caminos más cortos si las longitudes de algunos arcos han sido modificadas. Con este objeto, este procedimiento será incorporado a algunos algoritmos de resolución de problemas más complejos de optimización en redes (por ejemplo, flujo a costo mínimo).

Algoritmo.

(0) Determinar una gráfica parcial de la gráfica original que sea una arborescencia de raíz “ o ”.

- Si “ o ” no es raíz de G , terminar.
- Si “ o ” es raíz de G , sea X el conjunto de arcos de la arborescencia encontrada. Sea $\pi(x)$ la longitud del camino de “ o ” a x en la arborescencia (N, X) .

(1) Buscar un arco u en $A - X$ tal que $\delta = -\pi(T(u)) + \pi(I(u)) + d(u) < 0$

- (a) Si no existe un arco tal, (N, X) es una arborescencia de caminos más cortos de raíz “ o ”. Terminar.
- (b) Si $(N, X \cup \{u\})$ contiene un circuito C , este circuito es absorbente. Terminar.
- (c) Si $(N, X \cup \{u\})$ no contiene circuito ir a (2).

(2) Sea:

$$x = T(u)$$

v es el arco de (N, X) cuyo extremo terminal es x .

Poner:

$$X = X \cup \{u\} - \{v\}$$

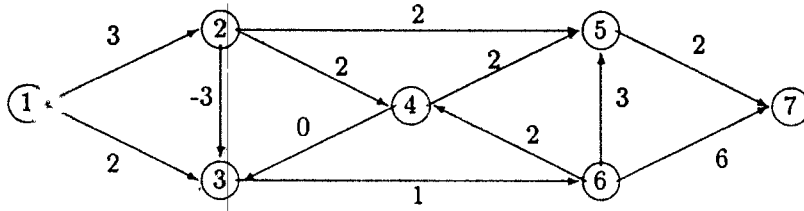
$$N' = \{y \in N : y = x \text{ o bien } y \text{ es un descendiente de } x^1 \text{ en } (N, X)\}$$

$$\pi(y) = \pi(y) + \delta \quad \forall y \in N'$$

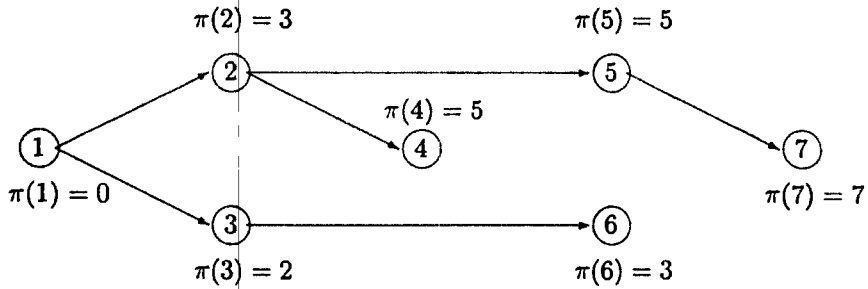
Ir a (1).

¹ *Descendiente de x* es todo vértice al cual podemos acceder a partir de x .

Ejemplo 10.

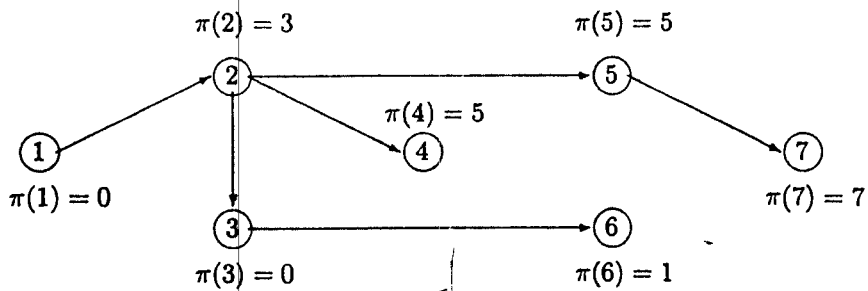


Comenzamos considerando la arborescencia siguiente:



Sea $u = (2, 3)$
 $\delta = -2 + 3 + (-3) = -2 < 0$

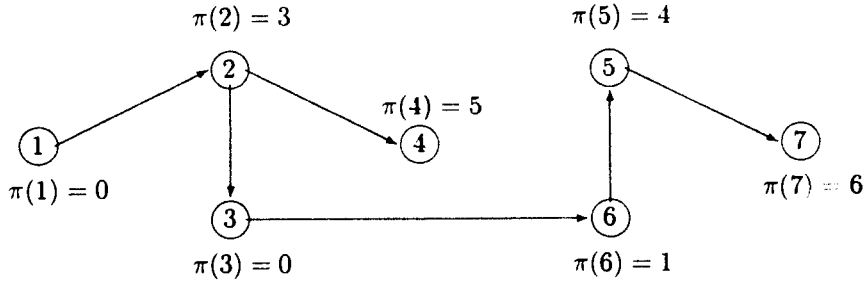
$x = T(u) = 3$
 $v = (1, 3)$
 $N' = \{3, 6\}$
 $\pi(3) = 2 - 2 = 0$
 $\pi(6) = 3 - 2 = 1$



Sea $u = (6, 5)$
 $\delta = -5 + 1 + 3 = -1 < 0$

$x = 5$
 $v = (2, 5)$
 $N' = \{5, 7\}$
 $\pi(5) = 5 + (-1) = 4$

$$\pi(7) = 7 - 1 = 6$$



Sea $u = (6, 4)$

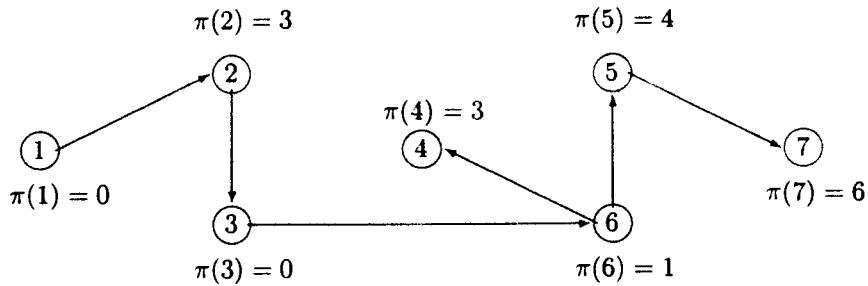
$$\delta = -5 + 1 + 2 = -2 < 0$$

$$x = 4$$

$$v = (2, 4)$$

$$N' = \{4\}$$

$$\pi(4) = 5 - 2 = 3$$



Sea $u = (1, 3)$

$$\delta = 0 - 0 + 2 > 0$$

Sea $u = (2, 5)$

$$\delta = -4 + 3 + 2 > 0$$

Sea $u = (4, 5)$

$$\delta = -4 + 3 + 2 > 0$$

Sea $u = (4, 3)$

$$\delta = 0 + 3 + 0 > 0$$

Sea $u = (2, 4)$

$$\delta = -3 + 3 + 2 > 0$$

Sea $u = (6, 7)$

$$\delta = -6 + 1 + 6 > 0$$

Como no existe un arco u tal que $\delta < 0$, se tiene que (N, X) es la arborescencia de caminos más cortos.

Justificación: a) A lo largo del desarrollo del proceso, $\pi(x)$ queda igual a la longitud del camino de o a x incluido en la arborescencia (N, X) . En cada etapa $\sum_{x \in N} \pi(x)$ decrece estrictamente, de donde no podemos encontrar una arborescencia ya encontrada anteriormente. En consecuencia, puesto que el número de gráficas parciales de una gráfica dada que son arborescencias, es finito, el algoritmo es finito.

b) Si terminamos en el paso (1-b) el circuito C está constituido por el arco u y por los arcos que pertenecen a la arborescencia, i.e.

$$\begin{aligned}\pi(T(u)) - \pi(I(u)) &> d(u) \\ \pi(T(v)) - \pi(I(v)) &> d(v) \quad \forall v \in C - \{u\}\end{aligned}$$

y por consecuencia $0 > \sum_{v \in C} d(v)$ es decir, es un circuito absorbente.

c) Si terminamos en el paso (1-a):

El algoritmo termina con una arborescencia si $\forall u \in X$

$$\begin{aligned}-\pi(T(u)) + \pi(I(u)) + d(u) &\geq 0 \\ \Rightarrow \pi(T(u)) - \pi(I(u)) - d(u) &\leq 0 \\ \Rightarrow \pi(I(u)) + d(u) &\geq \pi(T(u))\end{aligned}\tag{A.1}$$

Supongamos $\pi(x)$ al final del algoritmo, no es la longitud de una ruta más corta de o a x .

Por construcción $\pi(x)$ es la longitud de un camino de o a x .

Sea P un camino más corto de o a x entonces

$$d(P) < \pi(x)\tag{A.2}$$

Sea y el predecesor de x en el camino P , entonces:

$$d(P) = \pi(y) + d(y, x) < \pi(x)$$

por (A.2) lo cual es una contradicción a (A.1).

Por lo tanto, $\pi(x)$ es la longitud de una ruta más corta de o a x , por lo tanto, la arborescencia generada es una arborescencia de rutas más cortas de raíz " o ".

Observación 8. No hemos precisado cómo determinar en este algoritmo si la red $(N, X \cup \{u\})$ contiene un circuito. La manera más sencilla de proceder es buscar si la extremidad terminal de " u " es un predecesor en la arborescencia (N, X) de la extremidad inicial. Si estamos seguros de que no hay circuito absorbente en la red, podemos evitarnos esta prueba.

Apéndice B

Principio de Descomposición de Dantzig-Wolfe

El principio de descomposición es un procedimiento sistemático para resolver programas lineales de gran escala o programas lineales que contienen restricciones de estructura especial. Las restricciones se dividen en dos conjuntos: restricciones generales y restricciones de estructura especial.

La estrategia del principio de descomposición consiste en operar sobre dos programas lineales separados: uno sobre el conjunto de las restricciones generales y otro sobre el conjunto de las restricciones especiales. La información se pasa entre uno y otro de los programas lineales hasta alcanzar un punto en el que se tiene la solución óptima del problema original. El programa lineal sobre las restricciones generales se llama el *programa maestro*, y el programa lineal sobre las restricciones especiales se llama el *subproblema*. El programa maestro le pasa un nuevo conjunto de coeficientes de costo al subproblema y recibe una nueva columna basada en tales coeficientes.

Este apéndice está basado en [1].

Principio de Descomposición

Frecuentemente encontramos problemas de programación lineal de grandes dimensiones que presentan la siguiente estructura:

$$\begin{array}{rllll} \min w = & c_1x_1 & +c_2x_2 & +\cdots & +c_Tx_T \\ \text{s.a.} & A_1x_1 & +A_2x_2 & +\cdots & +A_Tx_T & = b \\ & B_1x_1 & & & & \leq b_1 \\ & & B_2x_2 & & & \leq b_2 \\ & & & \ddots & & \vdots \\ & & & & B_Tx_T & \leq b_T \\ & & & & x_i & \geq 0 \quad i = \overline{1} \overline{T} \end{array} \quad (\text{B.1})$$

Se mostrará que este tipo de problemas pueden resolverse eficientemente con un principio de descomposición que requiere la solución de una serie de de problemas de programación lineal cuyas dimensiones son más pequeñas que la de (B.1).

Observación 1. Problemas con la estructura anterior surgen con frecuencia en redes de optimización con varios bienes y en la asignación de recursos escasos entre actividades que compiten entre sí. Problemas con esta estructura pueden resolverse mediante el algoritmo de descomposición. Sin embargo, como se verá posteriormente, la estructura diagonal puede aprovecharse aún más.

El conjunto de puntos $x_i \geq 0$ que satisfacen $B_i x_i \leq b_i$ forma un conjunto convexo cerrado y que contiene solamente un número finito de puntos extremos. Si el conjunto está estrictamente acotado se trata, entonces de un poliedro y cualquier punto en el interior del conjunto convexo puede representarse por una combinación convexa de los puntos extremos. Supondremos que estamos en este caso.

Denotemos por x_{ij}^* los puntos extremos del conjunto convexo de soluciones factibles de $B_i x_i \leq b_i$. Supongamos que tenemos t_i de estos puntos extremos. Entonces cualquier solución factible $x_i \geq 0$ de $B_i x_i \leq b_i$ puede reescribirse

$$\begin{aligned} x_i &= \sum_{j=1}^{t_i} \lambda_{ij} x_{ij}^* \\ \sum_{j=1}^{t_i} \lambda_{ij} &= 1 \\ \lambda_{ij} &\geq 0, \quad j = \overline{1, T_i} \end{aligned}$$

Observación 2. Cualquier x_i de la forma dada anteriormente es automáticamente una solución factible de $B_i x_i \leq b_i$, pues cualquier combinación convexa de puntos extremos será un elemento del conjunto convexo de soluciones factibles.

Sustituyendo x_i , el problema de optimización (B.1) puede transformarse en el siguiente problema de variables λ_{ij} :

$$\begin{aligned} \min w &= \sum_{j=1}^{t_1} (c_1 x_{1j}) \lambda_{1j} + \dots + \sum_{j=1}^{t_T} (c_T x_{Tj}) \lambda_{Tj} \\ \text{s.a.} \quad &\sum_{j=1}^{t_1} (A_1 x_{1j}) \lambda_{1j} + \sum_{j=1}^{t_T} (A_T x_{Tj}) \lambda_{Tj} = b \\ &\sum_{j=1}^{t_1} \lambda_{1j} = 1 \\ &\vdots \\ &\sum_{j=1}^{t_T} \lambda_{Tj} = 1 \\ &\lambda_{ij} \geq 0 \quad i = \overline{1, T}, \quad j = \overline{1, t_i} \end{aligned} \tag{B.2}$$

Puesto que t_i , el número de puntos extremos del conjunto $\{B_i x_i \leq b_i, x_i \geq 0\}$, generalmente es muy grande, tratar de enumerar explícitamente todos los puntos extremos x_{ij}^* , y resolver explícitamente este problema sería una tarea muy difícil. En consecuencia, se intentará encontrar una solución óptima del problema (y, por lo tanto, del problema original) sin enumerar explícitamente todos los puntos extremos.

Aplicación del Método Simplex Revisado.

Supóngase que se tiene una solución básica factible del sistema anterior con una base B . Obsérvese que cada base debe contener al menos una variable λ_{ij} de cada bloque i . Aún más supóngase que B^{-1} , $\bar{b} = B^{-1} \begin{pmatrix} b \\ \mathbf{1} \end{pmatrix}$, y $(\mathbf{w} \ \alpha) = (w_1, \dots, w_m, \alpha_1, \dots, \alpha_T) = \hat{c}_B B^{-1}$ son conocidos, en donde \hat{c}_B es el costo de las variables básicas ($\hat{c}_{ij} = c_i x_{ij}$ para λ_{ij}) y \mathbf{w} y α son las variables duales correspondientes a las restricciones del problema (B.2). Esto puede resumirse en el arreglo simplex revisado siguiente:

(\mathbf{w}, α)	$\hat{c}_B B^{-1}$
B^{-1}	\bar{b}

El método simplex revisado procede concluyendo que la solución presente es óptima, o bien decidiendo incrementar una variable no básica.

Esta solución es óptima si $z_{ij} - \hat{c}_{ij} \leq 0$ para cada variable (naturalmente $z_{ij} - \hat{c}_{ij} = 0$ para cada variable básica). En particular, en optimalidad deben satisfacerse las siguientes condiciones:

$$\lambda_{ij} \text{ no básica} \Rightarrow 0 \geq z_{ij} - \hat{c}_{ij} = \mathbf{w}A_i x_{ij} + \alpha_i - c_i x_{ij} \quad (\text{B.3})$$

El que las condiciones (B.3) se cumplan o no puede verificarse fácilmente resolviendo los siguientes subproblemas:

$$\begin{array}{ll} \max z = & (\mathbf{w}A_i - c_i)x_i + \alpha_i \\ \text{s.a.} & B_i x_i \leq b_i \\ & x_i \geq 0 \end{array} \quad (\text{B.4})$$

Sea x_{ik} un punto extremo óptimo. Si el valor objetivo óptimo es tal que

$$z_{ik} - \hat{c}_{ik} = \mathbf{w}A_i x_{ik} + \alpha_i - c_i x_{ik} \leq 0$$

entonces la condición (B.3) se cumple para el subproblema i . De lo contrario, λ_{ik} puede introducirse en la base. Cuando todos los subproblemas tengan $z_{ij} - \hat{c}_{ij} \leq 0$, se tendrá entonces la solución óptima del problema original. Si las restricciones maestras son del tipo de desigualdad, entonces antes de terminar debe verificarse los valores $z_j - \hat{c}_j$ para las variables de holgura no básicas, es decir, para una restricción maestra i del tipo \leq , con variables de holgura asociadas s_i , se obtiene:

$$z_{s_i} - c_{s_i} = (\mathbf{w}, \alpha) \begin{pmatrix} \mathbf{e}_i \\ 0 \end{pmatrix} - 0 = w_i$$

En resumen, se resuelve cada subproblema i . Si el subproblema i da un punto óptimo tal que $\mathbf{w}A_i x_{ik} + \alpha_i - c_i x_{ik} > 0$, entonces el punto extremo es un candidato para entrar a la base maestra. Si no ocurre ninguna de estas condiciones, entonces en el presente no hay ninguna columna que sea candidato para entrar a la base maestra del subproblema i . Si ningún subproblema da un candidato para entrar a la base maestra, se tiene entonces la solución óptima. En caso contrario, debe seleccionarse uno de los varios candidatos para entrar a la base maestra. Al seleccionar el candidato, se actualiza la columna entrante, se pivotea sobre el arreglo maestro, y se repite el proceso.

Observación 3. El conjunto de problemas en (B.4) puede presentar alguna estructura especial que permita aplicar algún algoritmo ya conocido para su resolución. Por ejemplo, puede tratarse de problemas de flujo o de rutas, para los cuales ya se expusieron algunos métodos de resolución.

Apéndice C

El Problema de Transporte

C.1 Presentación del Problema

Uno de los problemas típicos de programación lineal es el llamado *problema de transporte*; el cual consiste en minimizar el costo de transportar un artículo de lugares donde se dispone (orígenes) hacia lugares donde es demandado (destinos). Nos basamos en [26] para la presentación de este apéndice.

El planteamiento del problema es el siguiente:

Supóngase que en m lugares de expedición (orígenes) se encuentran a_1, a_2, \dots, a_m unidades de un artículo que deben ser enviadas a n consumidores (destinos) en cantidades de b_1, b_2, \dots, b_n unidades. Se dan los costos $c_{ij} \geq 0$ de transportar una unidad del artículo del i -ésimo origen al j -ésimo destino. Sea $x_{ij} \geq 0$ la cantidad de unidades del artículo que se transporta del i -ésimo origen al j -ésimo destino.

Las restricciones del problema son las siguientes

1) La cantidad que sale de los orígenes debe ser la que se dispone:

$$\sum_{j=1}^n x_{ij} = a_i \quad \forall i = \overline{1, m}$$

2) La cantidad que llega a los destinos debe ser la que se demanda:

$$\sum_{i=1}^m x_{ij} = b_j \quad \forall j = \overline{1, n}$$

3) Las variables deben ser no negativas.

$$x_{ij} \geq 0 \quad \forall i = \overline{1, m} \quad j = \overline{1, n}$$

El objetivo es

$$\begin{aligned} \min w &= c_{11}x_{11} + c_{12}x_{12} + \dots + c_{mn}x_{mn} \\ &= \sum_{i=1}^m \sum_{j=1}^n c_{ij}x_{ij}. \end{aligned}$$

Así el planteamiento matemático del problema de transporté es

$$\begin{aligned} \min w &= \sum_{i=1}^m \sum_{j=1}^n c_{ij}x_{ij} \\ \text{s.a.} \quad &\sum_{j=1}^n x_{ij} = a_i \quad \forall i = \overline{1, m} \\ &\sum_{i=1}^m x_{ij} = b_j \quad \forall j = \overline{1, n} \\ &x_{ij} \geq 0 \quad \forall i = \overline{1, m} \quad j = \overline{1, n} \end{aligned}$$

La resolución del problema se divide en dos etapas: 1) La determinación de la solución factible inicial y 2) La construcción de las iteraciones sucesivas, o sea, la aproximación a la solución óptima.

Supongamos que los valores de las disponibilidades y de las demandas son enteros no negativos y además que la suma de las disponibilidades es igual a la suma de las demandas, es decir que

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

C.2 Resolución

Describimos ahora, el primer paso de la resolución, que consiste en encontrar una primera solución factible.

Supongamos que tenemos los datos iniciales del problema en la tabla siguiente

a_1	x_{11}	c_{11}	x_{12}	c_{12}	...	x_{1j}	c_{1j}	...	x_{1n}	c_{1n}
a_2	x_{21}	c_{21}	x_{22}	c_{22}	...	x_{2j}	c_{2j}	...	x_{2n}	c_{2n}
\vdots	\vdots	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\ddots	\vdots	\vdots
a_i	x_{i1}	c_{i1}	x_{i2}	c_{i2}	...	x_{ij}	c_{ij}	...	x_{in}	c_{in}
\vdots	\vdots	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\ddots	\vdots	\vdots
a_m	x_{m1}	c_{m1}	x_{m2}	c_{m2}	...	x_{mj}	c_{mj}	...	x_{mn}	c_{mn}
a_i b_j	b_1		b_2		...	b_j		...	b_n	

Vamos a construir una solución factible inicial por el método llamado *esquina noroeste*. Llenamos la tabla a partir de la esquina superior izquierda, moviéndonos luego por el renglón hacia la derecha, o bien por la columna hacia abajo. En la casilla (1, 1), x_{11} va ser igual al menor entre los números a_1 y b_1 , es decir,

$$x_{11} = \min\{a_1, b_1\}$$

- Si $a_1 > b_1$, entonces $x_{11} = b_1$ y la primera columna está *completa*, es decir, las necesidades del primer destino están satisfechas por completo. El valor b_1 se cambia por cero y a_1 por $a_1 - b_1$, y la columna uno la "cancelamos", o sea, ya no la tomamos en cuenta.

Nos movemos luego por el primer renglón, escribiendo en la casilla vecina (1, 2) el menor entre los números $a_1 - b_1$ y b_2 , o sea,

$$x_{12} = \min\{a_1 - b_1, b_2\}$$

- Si $b_1 > a_1$, se *completa* entonces, el primer renglón y por lo tanto lo cancelamos, así $x_{11} = a_1$, el valor a_1 se cambia por cero y b_1 por $b_1 - a_1$.

Continuamos por la casilla (2, 1) en la cual escribimos

$$x_{21} = \min\{a_2, b_1 - a_1\}$$

Continuamos con este proceso hasta que en una etapa dada, quedan satisfechas las demandas b_j y agotadas las disponibilidades a_i .

Después de construida la solución inicial todas las variables quedan divididas en dos grupos: x_{kl} que son las variables básicas y x_{rs} que son las variables no básicas.

El método de resolución es un método primal-dual, así una vez que tenemos una solución factible, calculamos las variables duales:

En una tabla de $(m + 1) \times (n + 1)$ calculamos u_i las variables duales asociadas a los orígenes y v_j asociadas a los destinos. Fijamos un valor cualquiera para u_1 y a partir de éste calculamos todos los demás de tal manera que, para las casillas asociadas a las variables básicas, se satisfaga que

$$u_k + v_l = c_{kl}$$

Una vez que se tienen los valores para u_i y v_j , llenamos la tabla siguiente, llamada de *costos indirectos*

$$c'_{ij} = \begin{array}{c|cccc} & v_i & v_1 & v_2 & \cdots & v_n \\ \hline u_j & & & & & \\ \hline u_1 & & u_1 + v_1 & u_1 + v_2 & \cdots & u_1 + v_n \\ \hline u_2 & & u_2 + v_1 & u_2 + v_2 & \cdots & u_2 + v_n \\ \hline \vdots & & \vdots & \vdots & \ddots & \vdots \\ \hline u_m & & u_m + v_1 & u_m + v_2 & \cdots & u_m + v_n \end{array}$$

Y sea c_{ij} la tabla con los costos reales del problema y calculamos:

$$c_{ij} - c'_{ij}$$

Observación 1. Las entradas asociadas a las variables básicas tendrán un valor de cero, pues al calcular la resta para estas casillas se tiene que

$$c_{kl} - c'_{kl} = c_{kl} - (u_k + v_l) = c_{kl} - c_{kl} = 0$$

- Si todas las entradas (no básicas) son positivas o cero entonces la solución es óptima.
- Si existe algún valor $c_{rs} - c'_{rs} < 0$ entonces escogemos la entrada que tenga el valor mínimo y la variable asociada a éste es candidato a entrar a la base.

Una vez identificada la variable entrante, enviamos Δ unidades de flujo en esa casilla y para balancear la solución restamos y sumamos Δ intercaladamente sobre el ciclo formado en la tabla donde figura la solución factible y la casilla entrante, por ejemplo:

$x_{kl} + \Delta$	\cdots	\cdots	$x_{kl} - \Delta$
\vdots			\vdots
\vdots	$x_{kl} - \Delta$	\cdots	$x_{kl} + \Delta$
\vdots	\vdots		
$x_{kl} - \Delta$	Δ		

Observación 2. Si al “caminar” sobre un ciclo restando y sumando Δ unidades, en una columna o renglón existe una o más variables básicas intermedias entre un vértice y otro del ciclo, éstas no deben ser alteradas.

Sea Y el conjunto de todas las variables básicas para las cuales se les restó Δ unidades, entonces

$$\Delta = x_{pq} = \min\{x_{kl} : x_{kl} \in Y\}$$

Y la variable x_{pq} sale de la base.

Se renueva la solución factible y se procede nuevamente al cálculo de las variables duales. Se continua el procedimiento hasta llegar a la solución óptima.

C.3 Ejemplo

Considérese la siguiente tabla de transporte donde se tienen dos orígenes, A y B , y tres destinos, 1, 2 y 3. Los valores en la tabla determinan las disponibilidades, demandas y costos de transporte. Encontrar el plan de distribución de tal manera que el costo sea mínimo.

		1	2	3
A	150	6	10	4
B	90	12	2	8
	a_i / b_j	60	70	110

Primero encontramos una solución factible por el método de la esquina noroeste.

Comenzamos llenando la casilla (1, 1)

$$x_{11} = \min\{150, 60\} = 60$$

La primera columna está completa y por lo tanto, se cancela.

Pasamos a la casilla (1, 2)

$$x_{12} = \min\{150 - 60, 70\} = 70$$

La segunda columna está completa y por lo tanto, se cancela.

Luego, la casilla (1, 3)

$$x_{13} = \min\{90 - 70, 110\} = 20$$

Como en la tercera columna resulta el resto igual a 90, pasamos al llenado de la casilla (2, 3) en la cual tenemos

$$x_{23} = \min\{90, 90\} = 90$$

Y contamos ya, con la primera solución factible

$$x_{ij} = \begin{array}{c} A \\ B \end{array} \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 150 & 60 & 70 & 20 \\ 90 & 0 & 0 & 90 \\ \hline a_i & & & \\ \hline & b_j & & \end{array}$$

con un costo de

$$\begin{aligned} w &= 60(6) + 70(10) + 20(4) + 90(8) \\ &= 1860 \end{aligned}$$

Calculamos las variables duales

v_j	$v_1 = 6$	$v_2 = 10$	$v_3 = 4$
u_i			
$u_1 = 0$			
$u_2 = 12$			

Así, calculamos los costos indirectos:

$$c'_{ij} = \begin{array}{c|ccc} & v_j & & \\ \hline u_i & v_1 = 6 & v_2 = 10 & v_3 = 4 \\ \hline u_1 = 0 & 6 & 10 & 4 \\ u_2 = 4 & 10 & 14 & 8 \end{array}$$

Entonces

$$\begin{aligned} c_{ij} - c'_{ij} &= \begin{array}{c|ccc} 6 & 10 & 4 \\ \hline 12 & 2 & 8 \end{array} - \begin{array}{c|ccc} 6 & 10 & 4 \\ \hline 10 & 14 & 8 \end{array} \\ &= \begin{array}{c|ccc} 0 & 0 & 0 \\ \hline 2 & -12 & 0 \end{array} \end{aligned}$$

Por lo tanto, x_{22} es candidato a entrar a la base:

60	$70 - \Delta$	$20 + \Delta$
0	Δ	$90 - \Delta$

Así,

$$\Delta = 70$$

Y la variable x_{12} sale de la base. La nueva solución factible es

$$x_{ij} = \begin{array}{c} A \\ B \end{array} \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 150 & 60 & 0 & 90 \\ 90 & 0 & 70 & 20 \\ \hline a_i & & & \\ \hline & b_j & & \end{array}$$

Con un costo de

$$\begin{aligned} w &= 60(6) + 90(4) + 70(2) + 20(8) \\ &= 1020 \end{aligned}$$

Con esta nueva solución, calculamos las variables duales:

v_j	$v_1 = 6$	$v_2 = -2$	$v_3 = 4$
u_i			
$u_1 = 0$			
$u_2 = 4$			

Los costos indirectos:

$$c'_{ij} = \begin{array}{c|ccc} & v_j & & \\ \hline u_i & & & \\ \hline u_1 & v_1 = 6 & v_2 = -2 & v_3 = 4 \\ \hline u_1 = 0 & 6 & -2 & 4 \\ \hline u_2 = 4 & 10 & 2 & 8 \\ \hline \end{array}$$

Tenemos que

$$\begin{aligned} c_{ij} - c'_{ij} &= \begin{array}{|c|c|c|} \hline 6 & 10 & 4 \\ \hline 12 & 2 & 8 \\ \hline \end{array} - \begin{array}{|c|c|c|} \hline 6 & -2 & 4 \\ \hline 10 & 2 & 8 \\ \hline \end{array} \\ &= \begin{array}{|c|c|c|} \hline 0 & 12 & 0 \\ \hline 2 & 0 & 0 \\ \hline \end{array} \end{aligned}$$

Como todas las entradas son no negativas se tiene que la solución es óptima. Es decir,

$$\begin{aligned} w_{\min} &= 1020 \\ x_{11} &= 60 \\ x_{12} &= 0 \\ x_{13} &= 90 \\ x_{21} &= 0 \\ x_{22} &= 70 \\ x_{23} &= 20 \end{aligned}$$

Bibliografía

- [1] M. S. Bazaraa y J. J. Jarvis. "Programación Lineal y Flujo en Redes". *Limusa.*, 1981.
- [2] F. G. Commoner. *A Sufficient Condition for a Matrix to be Totally Unimodular*. *Networks* **3**. (1973) 351-365.
- [3] J. R. Evans. *A Combinatorial Equivalence between a Class of Multicommodity Flow Problems and the Capacitated Transportation Problem*. *Math. Programming* **10**. (1976) 401-404.
- [4] J. R. Evans. *A Single-Commodity Transformation for Certain Multicommodity Network Flow Problem*. *Opns. Res.* **26**. (1978) 673-680.
- [5] J. R. Evans. *The Simplex Method for Integral Multicommodity Networks*. *Naval Res. Logist. Quart.* Próxima aparición.
- [6] J. R. Evans & J. J. Jarvis. *Network Topology and Integral Multicommodity Flow Problem*. *Networks* **8**. (1978) 107-119.
- [7] J. R. Evans, J. J. Jarvis & R. Duke. *Graphic Matroids and the Multicommodity Transportation Problem*. *Math. Programming* **13**. (1977) 323-328.
- [8] L. Ford & D. Fulkerson. "Flows in Networks". *Princeton University Press.*, 1962.
- [9] L. Ford & D. Fulkerson. *A Suggested Computation for Maximal Multicommodity Network Flows*. *Management Sci.* **5**. (1958) 97-101.
- [10] J. B. Fraleigh "Álgebra abstracta". *Addison Wesley Iberoamericana.*, 1987.
- [11] M. D. Grigoriadis & W. White. *A Partitioning Algorithm for the Multicommodity Network Flow Problem*. *Math. Programming* **3**. (1972) 157-177.
- [12] R. Grinold. *A Multicommodity Max-Flow Algorithm*. *Opns. Res.* **16**. (1968) 1234-1238.
- [13] G. Hadley. "Linear Programming" *Addison Wesley Publishing Co. INC.*, 1963.
- [14] J. Hartman & L. Lasdon. *A Generalized Upper Bounding Algorithm for Multicommodity Network Flow Problems*. *Networks* **1**. (1972) 333-354.
- [15] J. Kennington. *A Survey of Linear Cost Multicommodity Network Flows*. *Opns. Res.* **2**. (1978) 209-236.
- [16] J. Kennington & R. Helgason. "Algorithms for Network Programming". *Wiley-Interscience Publication.*, 1980.
- [17] D. Kleitman. *An Algorithm for Certain Multi-Commodity Flow Problems*. *Networks* **1**. (1971) 75-90.
- [18] R. Peniche. "Apuntes de Programación Lineal y Análisis de Redes". No publicado.
- [19] J. Prawda. "Métodos y modelos de Investigación de Operaciones". *Limusa I.*, 1979.

- [20] W. Rothfarb, N. P. Shein & I. T. Frisch. *Common Terminal Multicommodity Flow*. Opns. Res. 16. (1968) 202-205.
- [21] W. Rothfarb & I. T. Frisch. *On the 3-Commodity Flow Problem*. SIAM J Appl. Math. 17. (1968) 46-58.
- [22] B. Rothschild & A. Whinston. *On Two Commodity Network Flows*. Opns. Res. 14. (1966) 377-387.
- [23] B. Rothschild & A. Whinston. *Feasibility of Two Commodity Network Flows*. Opns. Res. 14. (1966) 1125-1129.
- [24] B. Rothschild, A. Whinston & J. Kent. *Computing Two-Commodity Flows*. Opns. Res. 16. (1968) 446-450.
- [25] M. Sakarovitch. *Two Commodity Network Flows and Linear Programming*. Math. Programming 4. (1973) 1-20.
- [26] M. Simonnard. "Programación Lineal". *Parainfo*. 1972.
- [27] H. Soroush & P. B. Mirchandani. *The Stochastic Multicommodity Flow Problem*. Networks 20. (1990) 121-155.
- [28] D. Tang. *Comments on Feasibility conditions of Simultaneous Flows in a Network*. Opns. Res. 13. (1965) 143-146.
- [29] J. A. Tomlin. *Minimum-cost Multicommodity Networks Flows*. Opns. Res. 14. (1966) 45-51.
- [30] A. Weintraub, F. Barahona, J.C. Rojas & E. Muñoz. *Solving Multicommodity Flow Problems* Investigación Operativa 3. (1993) 201-213.