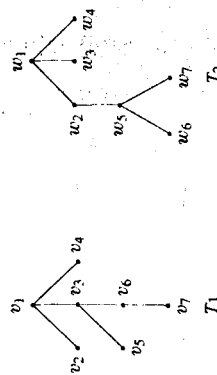


32. Determine si los siguientes árboles con raíz son isomorfos. Si los árboles son isomorfos, proporcione un isomorfismo. Si los árboles no son isomorfos, proporcione un invariante no compartido por los árboles.

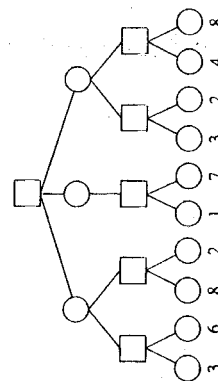


Sección 7.9

33. Determine el valor de la posición del gato mediante la función de evaluación del ejemplo 7.9.1.



34. Proporcione una función de evaluación de una posición del gato distinta a la del ejemplo 7.9.1. Intente una distinción más fina entre las posiciones con respecto de la función de evaluación del ejemplo 7.9.1.
35. Evalúe cada vértice en el siguiente árbol de juego. Los valores de los vértices terminales están dados.



36. Evalúe la raíz del árbol del ejercicio 35 mediante el procedimiento minimax con poda alfa-beta. Suponga que los hijos se evalúan de izquierda a derecha. Para cada vértice cuyo valor sea calculado, escriba el valor en el vértice. Coloque un signo de verificación en la raíz de cada subárbol podado.

8

MODELO DE REDES Y REDES DE PETRI

8.1	MODELOS DE REDES
8.2	UN ALGORITMO DE FLUJO MÁXIMO
8.3	EL TEOREMA DEL FLUJO MÁXIMO Y CORTE MÍNIMO
8.4	ACOPLAMIENTO
	RINCÓN DE SOLUCIÓN DE PROBLEMAS: ACOPLAMIENTO
8.5	REDES DE PETRI
	NOTAS
	CONCEPTOS BÁSICOS DEL CAPÍTULO
	AUTOEVALUACIÓN DEL CAPÍTULO

¡Cásate conmigo, y nunca miraré a otro caballo!

— de A Day at the Races



En este capítulo analizamos dos temas, modelos de redes y redes de Petri, que utilizan las gráficas dirigidas. La mayor parte de este capítulo se dedica al problema de maximizar el flujo que pasa a través de una red. La red podría ser una red de transporte a través de la cual fluyen las mercancías, una red de tuberías a través de las cuales fluye el petróleo, una red de computadoras a través de la cual fluyen los datos, o varias otras posibilidades. En cada caso, el problema consiste en determinar un flujo máximo. Muchos otros problemas, que aparentemente no serían problemas de flujo, pueden modelarse como problemas de flujo en una red.

La maximización del flujo en una red es un problema que pertenece a la teoría de gráficas y a la investigación de operaciones. El problema del agente de ventas viajero es otro ejemplo de un problema en teoría de gráficas e investigación de operaciones. La investigación de operaciones estudia la muy amplia categoría de problemas de optimización del desempeño de un sistema. Algunos problemas típicos de la investigación de operaciones son los relativos a redes, asignación de recursos y a asignación de personal.

Las redes de Petri modelan sistemas en los que el procesamiento puede ocurrir de manera concurrente. El modelo proporciona un marco de referencia para tratar cuestiones como la posible aparición de estancamientos en un sistema o el hecho de exceder la capacidad de los componentes de un sistema.

8.1 MODELOS DE REDES

Consideremos la gráfica dirigida de la figura 8.1.1, la cual representa una red de tuberías de petróleo. El petróleo se descarga en el muelle a y se bombea a través de la red hasta la refinería z . Los vértices b, c, d y e representan estaciones de bombeo intermedias. Las aristas dirigidas representan subterráneas del sistema y muestran la dirección en que puede fluir el petróleo. Las etiquetas sobre las aristas muestran las capacidades de las subterráneas. El problema consiste en determinar una forma de maximizar el flujo del muelle a a la refinería y calcular el valor de este flujo máximo. La figura 8.1.1 es un ejemplo de red de transporte.

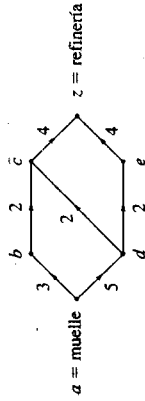


FIGURA 8.1.1 Una red de transporte.

DEFINICIÓN 8.1.1

Una red de *transporte* (o más simple, una *red*) es una gráfica dirigida, simple, con pesos, que satisfice:

- Un vértice fijo, la *fuerza*, no tiene aristas de entrada.
- Un vértice fijo, el *sumidero* (o *destino*), no tiene aristas de salida.
- El peso C_{ij} de la arista dirigida (i, j) , llamado la *capacidad* de (i, j) , es un número no negativo.

EJEMPLO 8.1.2

La gráfica de la figura 8.1.1 es una red de transporte. La fuerza es el vértice a y el sumidero es el vértice z . La capacidad de la arista (a, b) , C_{ab} , es 3, y la capacidad de la arista (b, c) , C_{bc} , es 2. □

En este capítulo, si G es una red, denotaremos la fuerza por a y el sumidero por z .

Un *flujo* en una red asigna un flujo en cada arista dirigida que no excede la capacidad de dicha arista. Además, se supone que el flujo de entrada a un vértice v , que no sea la fuerza ni el sumidero, es igual al flujo de salida de v . La siguiente definición precisa estas ideas.

DEFINICIÓN 8.1.3

Sea G una red de transporte. Sea C_{ij} la capacidad de la arista dirigida (i, j) . Un *flujo* F en G asigna a cada arista dirigida (i, j) un número no negativo F_{ij} tal que:

- $F_{ij} \leq C_{ij}$
 - Para cada vértice j , que no sea la fuerza ni el sumidero,
- $$\sum_i F_{ij} = \sum_j F_{ji}, \quad (8.1.1)$$

[En una suma como (8.1.1), a menos que se indique lo contrario, se supone que la suma se realiza sobre todos los vértices i . Además, si (i, j) no es una arista, hacemos $F_{ij} = 0$.] F_{ij} es el *flujo en la arista* (i, j) . Para cualquier vértice j ,

$$\sum_i F_{ij}$$

es el *flujo de entrada* a j , y

$$\sum_j F_{ji}$$

es el *flujo de salida* de j .

La propiedad expresada por la ecuación (8.1.1) es la *conservación del flujo*. En el ejemplo del bombeo de petróleo de la figura 8.1.1, la conservación del flujo indica que el petróleo no se utiliza ni se suministra en las estaciones de bombeo b, c, d y e .

EJEMPLO 8.1.4

Las asignaciones

$$F_{ab} = 2, \quad F_{bc} = 2, \quad F_{ca} = 3, \quad F_{ad} = 3, \\ F_{dc} = 1, \quad F_{de} = 2, \quad F_{ce} = 2,$$

definen un flujo para la red de la figura 8.1.1. Por ejemplo, el flujo de entrada del vértice d ,

$$F_{ad} = 3,$$

es igual al flujo de salida del vértice d ,

$$F_{dc} + F_{de} = 1 + 2 = 3.$$

En la figura 8.1.2 hemos trazado de nuevo la red de la figura 8.1.1 para mostrar el flujo del ejemplo 8.1.4. Una arista e se etiqueta " x, y " si la capacidad de e es x y el flujo en e es y . Esta notación se utilizará en todo el capítulo.

Observe que en el ejemplo 8.1.4, el flujo de salida de la fuerza a ,

$$F_{ab} + F_{ad}$$

es igual al flujo de entrada del sumidero z ,

$$F_{cz} + F_{ez}.$$

ambos iguales a 5. El siguiente teorema demuestra que el flujo de salida de la fuerza siempre es igual al flujo de entrada del sumidero.

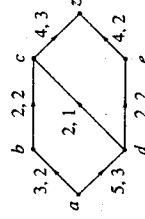


FIGURA 8.1.2

Flujo en una red. Las aristas se etiquetan con x, y para indicar la capacidad x y el flujo y .

TEOREMA 8.1.5

Dado un flujo F en una red, el flujo de salida de la fuente a es igual al flujo de entrada del sumidero z ; es decir,

$$\sum_i F_{ai} = \sum_j F_{jz}.$$

Demostración. Sea V el conjunto de vértices. Tenemos

$$\sum_{j \in V} \left(\sum_{i \in V} F_{ij} \right) = \sum_{j \in V} \left(\sum_{i \in V} F_{ji} \right),$$

pues cada suma doble es

$$\sum_{a \in E} F_a.$$

donde E es el conjunto de aristas. Ahora,

$$\begin{aligned} 0 &= \sum_{j \in V} \left(\sum_{i \in V} F_{ij} - \sum_{i \in V} F_{ji} \right) \\ &= \left(\sum_{i \in V} F_{iz} - \sum_{j \in V} F_{ji} \right) + \left(\sum_{i \in V} F_{io} - \sum_{j \in V} F_{ji} \right) \\ &\quad + \sum_{\substack{j \in V \\ j \neq a, z}} \left(\sum_{i \in V} F_{ij} - \sum_{i \in V} F_{ji} \right) \\ &= \sum_{i \in V} F_{iz} - \sum_{i \in V} F_{ai} \end{aligned}$$

pues $F_{ji} = 0 = F_{io}$ para toda $i \in V$, y (definición 8.1.3b)

$$\sum_{i \in V} F_{ij} - \sum_{i \in V} F_{ji} = 0 \quad \text{si } j \in V - \{a, z\}.$$

En vista del teorema 8.1.5, podemos enunciar la siguiente definición.

DEFINICIÓN 8.1.6

Sea F un flujo en una red G . El valor

$$\sum_i F_{ai} = \sum_j F_{jz}$$

es el *valor del flujo* F .

EJEMPLO 8.1.7

El valor del flujo en la red de la figura 8.1.2 es 5.

El problema de una red de transporte G se puede establecer así: Determinar un flujo máximo en G ; es decir, entre todos los flujos posibles en G , determinar un flujo F tal que el valor de F sea máximo. En la siguiente sección daremos un algoritmo que resuelve este problema de manera eficiente. Concluimos esta sección con algunos ejemplos más.

EJEMPLO 8.1.8

Una red de bombeo

La figura 8.1.3 representa una red de bombeo por medio de la cual se envía agua a dos ciudades, A y B , desde tres pozos, w_1 , w_2 y w_3 . Las capacidades de los sistemas intermedios aparecen sobre las aristas. Los vértices b , c y d representan estaciones de bombeo intermedias. Modelar este sistema como una red de transporte.

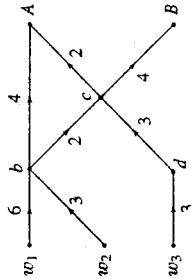


FIGURA 8.1.3 Una red de bombeo. El agua de las ciudades A y B se obtiene de los pozos w_1 , w_2 y w_3 . Las capacidades aparecen sobre las aristas.

Para obtener una fuente y un sumidero fijos, podemos obtener una red de transporte equivalente, uniendo las fuentes en una **superfuente** y los sumideros en un **supersumidero** (véase la figura 8.1.4). En la figura 8.1.4, ∞ representa una capacidad ilimitada.

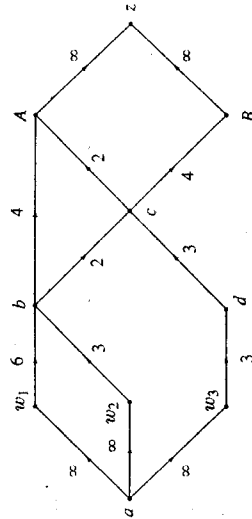


FIGURA 8.1.4 La red de la figura 8.1.3 con una fuente y un sumidero fijos.

EJEMPLO 8.1.9

Una red de flujo de tráfico

Es posible ir de la ciudad A a la ciudad C directamente o pasar por la ciudad B. Durante el periodo de 6:00 a 7:00 p.m. los tiempos promedio de viaje son

A a B (15 minutos)

B a C (30 minutos)

A a C (30 minutos).

Las capacidades máximas de las carreteras son

A a B (3000 vehículos)

B a C (2000 vehículos)

A a C (4000 vehículos).

Representar el flujo de tráfico de A a C durante el periodo de 6:00 a 7:00 p. m. como una red.

Un vértice representa una ciudad en un instante dado (véase la figura 8.1.5). Una arista conecta X, t_1 con Y, t_2 si podemos salir de la ciudad X en el instante t_1 p.m. y llegar a la ciudad Y en el instante t_2 p.m. La capacidad de una arista es la capacidad de la ruta. Existen aristas de capacidad infinita que conectan a A, t_1 con A, t_2 y B, t_1 con B, t_2 para indicar que una cantidad arbitraria de autos puede esperar en la ciudad A o la ciudad B. Por último, introducimos una superfuente y un sumidero.

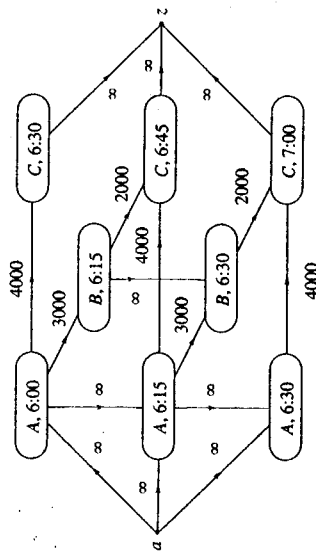
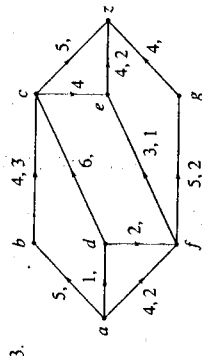
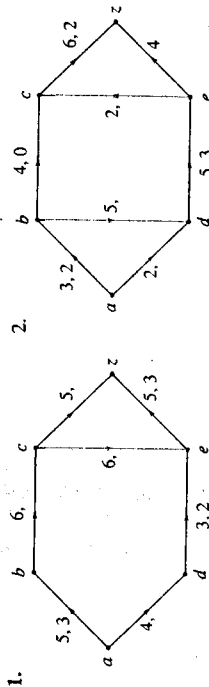


FIGURA 8.1.5 Una red que representa el flujo de tráfico de la ciudad A a la ciudad C durante el periodo de 6:00 a 7:00 p.m.

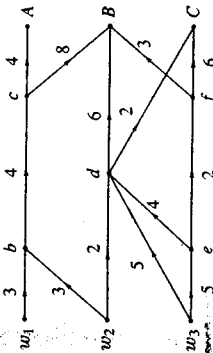
Algunas variantes del problema de flujo de redes se han utilizado en el diseño de redes de computadoras eficientes (véase [Jones; Kleinrock]). Al modelar una red de computadoras, un vértice es un mensaje o un centro de intercambio, una arista representa un canal por el cual se pueden transmitir los datos entre los vértices, un flujo es el número promedio de bits por segundo que se transmiten en un canal, y la capacidad de una arista es la capacidad del canal correspondiente.

Ejercicios

En los ejercicios 1-3, escriba los flujos faltantes sobre las aristas de modo que el resultado sea un flujo en la red dada. Determine los valores de los flujos.



4. La siguiente gráfica representa una red de bombeo en la que el petróleo de tres pozos, w_1 , w_2 y w_3 , se entrega a tres refinерías, A, B y C. Las capacidades de los sistemas intermedios aparecen sobre las aristas. Los vértices b, c, d, e y f representan estaciones de bombeo intermedias. Modele este sistema como una red.



5. Modele el sistema del ejercicio 4 como una red, suponiendo que el pozo w_1 puede bombear a lo más 2 unidades, el pozo w_2 a lo más 4 unidades, y el pozo w_3 a lo más 7 unidades.

6. Modele el sistema del ejercicio 5 como una red, suponiendo, además de las limitaciones sobre los pozos, que la ciudad A necesita 4 unidades, que B necesita 3 unidades, y que C necesita 4 unidades.

7. Modele el sistema del ejercicio 6 como una red, suponiendo, además de las restricciones sobre los pozos y las necesidades de las ciudades, que la estación intermedia d puede bombear a lo más 6 unidades.

8. Existen dos rutas de la ciudad A a la ciudad D. Una ruta pasa por la ciudad B y la otra por la ciudad C. Durante el periodo de 7:00 a 8:00 a.m., los tiempos promedio de recorrido son

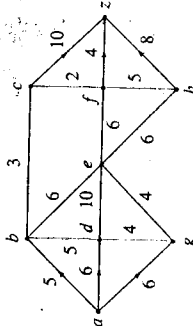
A a B	(30 minutos)
A a C	(15 minutos)
B a D	(15 minutos)
C a D	(15 minutos)

Las capacidades máximas de estas rutas son

A a B	(1000 vehículos)
A a C	(3000 vehículos)
B a D	(4000 vehículos)
C a D	(2000 vehículos)

Represente como una red el flujo de tráfico de A a D durante el periodo de 7:00 a 8:00 a.m.

9. En el siguiente sistema, queremos maximizar el flujo de a a z . Las capacidades aparecen sobre las aristas. El flujo entre dos vértices, ninguno de los cuales sea a o z , puede tener cualquier dirección. Modele este sistema como una red.



10. Proporcione un ejemplo de una red con exactamente dos flujos máximos, donde cada F_{ij} sea un entero no negativo.
11. ¿Cuál es el máximo número de aristas que puede tener una red?

8.2 UN ALGORITMO DE FLUJO MÁXIMO

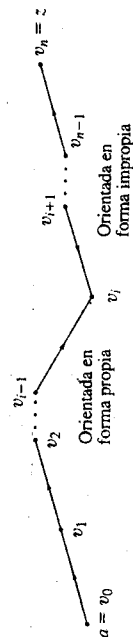
Si G es una red de transporte, un flujo máximo en G es un flujo con valor máximo. En general, existirán varios flujos con el mismo valor máximo. En esta sección daremos un algoritmo para determinar un flujo máximo. La idea fundamental es sencilla: comenzar con cierto flujo inicial e incrementar de manera iterativa el valor del flujo hasta que no pueda mejorarse más. El flujo resultante será entonces un flujo máximo.

Podemos considerar como flujo inicial aquél en el que el flujo en cada arista es igual a cero. Para incrementar el valor de un flujo dado, debemos determinar un camino de la fuente al sumidero e incrementar el flujo a lo largo de este camino.

En este punto, será útil introducir cierta terminología. En esta sección, G denota una red con fuente a , sumidero z y capacidad C . Por el momento, consideraremos a las aristas de G no dirigidas, y sea

$$P = (v_0, v_1, \dots, v_n), \quad v_0 = a, \quad v_n = z,$$

un camino de a a z en esta gráfica no dirigida. (Todos los caminos de esta sección harán referencia a la gráfica no dirigida subyacente.) Si una arista e en P está dirigida de v_{i-1} a v_i , decimos que e está **orientada en forma propia (con respecto de P)**; en caso contrario, decimos que e está **orientada en forma impropia (con respecto de P)** (véase la figura 8.2.1).



Camino P

FIGURA 8.2.1 Aristas orientadas en forma propia e impropia. La arista (v_{i-1}, v_i) está orientada en forma propia, pues está orientada en la dirección de a a z . La arista (v_i, v_{i+1}) está orientada en forma impropia, pues no está en la dirección de a a z .

Si podemos determinar un camino P de la fuente al sumidero en donde cada arista de P esté orientada en forma propia y el flujo en cada arista es menor que la capacidad de la arista, es posible aumentar el valor del flujo.

EJEMPLO 8.2.1

Consideremos el camino de a a z de la figura 8.2.2. Todas las aristas de P están orientadas en forma propia. El valor del flujo en esta red se puede incrementar en 1, como muestra la figura 8.2.3.



FIGURA 8.2.2 Un camino tal que todas sus aristas están orientadas en forma propia.

FIGURA 8.2.3 Después de incrementar el flujo de la figura 8.2.2 en 1.

También es posible incrementar el flujo en ciertos caminos de la fuente al sumidero que tengan aristas orientadas en forma propia e impropia. Sea P un camino de a a z y sea x un vértice en P que no sea a ni z (véase la figura 8.2.4). Existen cuatro posibilidades para

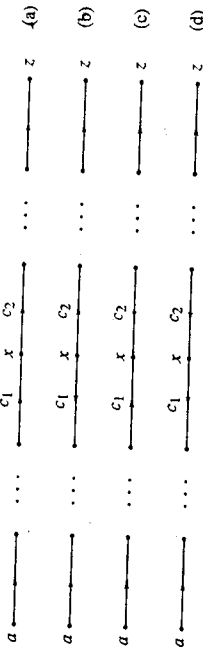


FIGURA 8.2.4 Las cuatro orientaciones posibles de las aristas incidentes en x .

las orientaciones de las aristas e_1 y e_2 incidentes en x . En el caso (a), ambas aristas están orientadas en forma propia. En este caso, si incrementamos el flujo en cada arista en Δ , el flujo de entrada en x seguirá siendo igual al flujo de salida de x . En el caso (b), si incrementamos el flujo en e_2 en Δ , debemos *disminuir* el flujo en e_1 en Δ de modo que el flujo de entrada en x siga siendo igual al flujo de salida en x . El caso (c) es análogo al caso (b), excepto que incrementamos el flujo en e_1 en Δ y disminuimos el flujo en e_2 en Δ . En el caso (d), disminuimos el flujo en ambas aristas en Δ . En cada caso, las asignaciones resultantes de las aristas dan como resultado un flujo. Por supuesto, para realizar estas alteraciones, debemos tener un flujo menor que la capacidad en una arista orientada en forma propia y un flujo distinto de cero en una arista orientada en forma impropia.

EJEMPLO 8.2.2

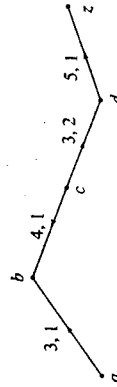


FIGURA 8.2.5 Un camino con una arista orientada en forma impropia: (c, b) .

Consideremos el camino de a a z de la figura 8.2.5. Las aristas (a, b) , (c, d) y (d, z) están orientadas en forma propia y la arista (c, b) está orientada en forma impropia. Disminuimos el flujo en 1 en la arista orientada en forma impropia (c, b) y aumentamos el flujo en 1 en las aristas orientadas en forma propia (a, b) , (c, d) y (d, z) (véase la figura 8.2.6). El valor del nuevo flujo es 1 más que el valor del flujo original.

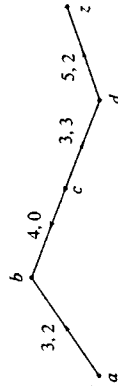


FIGURA 8.2.6 Después de incrementar en 1 el flujo de la figura 8.2.5.

Resumimos el método de los ejemplos 8.2.1 y 8.2.2 como un teorema.

TEOREMA 8.2.3

Sea P un camino de a a z en una red G tal que:

(a) Para cada arista (i, j) de P , orientada en forma propia,

$$F_{ij} < C_{ij}$$

(b) Para cada arista (i, j) de P , orientada en forma impropia,

$$0 < F_{ij}$$

Sea

$$\Delta = \min X,$$

donde X consta de los números $C_{ij} - F_{ij}$ para las aristas (i, j) de P orientadas en forma propia, y F_{ij} para las aristas (i, j) de P orientadas en forma impropia. Definimos

$$F_{ij}^* = \begin{cases} F_{ij} & \text{si } (i, j) \text{ no está en } P \\ F_{ij} + \Delta & \text{si } (i, j) \text{ está orientada en forma propia en } P \\ F_{ij} - \Delta & \text{si } (i, j) \text{ está orientada en forma impropia en } P. \end{cases}$$

Entonces F^* es un flujo cuyo valor es Δ unidades mayor que el valor de F .

Demostración. (Véanse las figuras 8.2.2, 8.2.3, 8.2.5 y 8.2.6.) El argumento de que F^* es un flujo aparece antes del ejemplo 8.2.2. Como la arista (a, v) en P se incrementa en Δ , el valor de F^* es Δ unidades mayor que el valor de F . ■

En la siguiente sección mostraremos que si no existen caminos que satisfagan las condiciones del teorema 8.2.3, el flujo es máximo. Así, es posible construir un algoritmo con base en el teorema 8.2.3. La idea es

1. Iniciar con un flujo (por ejemplo, un flujo tal que el flujo en cada arista sea 0).
2. Buscar un camino que satisfaga las condiciones del teorema 8.2.3. Si no existe tal camino, se habrá terminado y el flujo es máximo.
3. Se incrementa el flujo del camino en Δ , donde Δ se define como en el teorema 8.2.3, y se regresa a la línea 2.

En el algoritmo formal, buscamos un camino que satisfaga las condiciones del teorema 8.2.3 a la vez que se lleva un registro de las cantidades

$$C_{ij} - F_{ij}, F_{ij}$$

ALGORITMO 8.2.4

Determinación de un flujo máximo en una red

Este algoritmo determina un flujo máximo en una red. La capacidad de cada arista es un entero no negativo.

Entrada: Una red con fuente a , sumidero z , capacidad C , vértices $a = v_0, \dots, v_n = z$ y n

Salida: Un flujo máximo F

```

procedure max_flow( $a, z, C, v, n$ )
// la etiqueta de  $v$  es ( $pred(v), val(v)$ )
// se inicia con un flujo nulo
for cada arista  $(i, j)$  do
1.  $F_{ij} := 0$ 
2. while true do
3.  $begin$ 
4. // se eliminan todas las etiquetas
5. for  $i := 0$  to  $n$  do
6.  $begin$ 
7.  $pred(v_i) := null$ 
8.  $val(v_i) := null$ 
9.  $end$ 
10. // se etiqueta  $a$ 
11.  $predessor(a) := -$ 
12.  $val(a) := \infty$ 
13. //  $U$  es el conjunto de vértices etiquetados no examinados
14.  $U := \{a\}$ 
15. // se continúa hasta que  $z$  es etiquetado
16. while  $val(z) = null$  do
17.  $begin$ 
18. if  $U = \emptyset$  then // el flujo es máximo
19. return( $F$ )
20. se elige  $v$  en  $U$ 
21.  $U := U - \{v\}$ 
22.  $\Delta := val(v)$ 
23. for cada arista  $(v, w)$  con  $val(w) = null$  do
24. if  $F_{vw} < C_{vw}$  then
25.  $begin$ 
26.  $predessor(w) := v$ 
27.  $val(w) := \min\{\Delta, C_{vw} - F_{vw}\}$ 
28.  $U := U \cup \{w\}$ 
29.  $end$ 
30. for cada arista  $(w, v)$  con  $val(w) = null$  do
31. if  $F_{wv} > 0$  then
32.  $begin$ 
33.  $predessor(w) := v$ 
34.  $val(w) := \min\{\Delta, F_{wv}\}$ 
35.  $U := U \cup \{w\}$ 
36.  $end$ 
37.  $end$ 

```

// se determina un camino P de a a z para el cual se revisa el flujo

```

35.  $w_0 := z$ 
36.  $k := 0$ 
37. while  $w_k \neq a$  do
38.  $begin$ 
39.  $w_{k+1} := predessor(w_k)$ 
40.  $k := k + 1$ 
41.  $end$ 
42.  $P := (w_k, w_{k+1}, \dots, w_1, w_0)$ 
43.  $\Delta := val(z)$ 
44. for  $i = 1$  to  $k$  do
45.  $begin$ 
46.  $e := (w_i, w_{i-1})$ 
47. if  $e$  está orientada en forma propia en  $P$  then
48.  $F_e := F_e + \Delta$ 
49. else
50.  $F_e := F_e - \Delta$ 
51. else
52.  $end$ 
53.  $end$  max_flow

```

Se deja como ejercicio demostrar que el algoritmo 8.2.4 concluye (ejercicio 19). Si se permite que las capacidades sean números racionales no negativos, el algoritmo también concluye; sin embargo, si se permiten capacidades reales no negativas arbitrarias y que las aristas de la línea 20 se revisen en cualquier orden, el algoritmo podría no concluir (véase [Ford, páginas 21-22]).

El algoritmo 8.2.4 se conoce como el **algoritmo de etiquetado**. Ilustraremos el algoritmo con dos ejemplos.

EJEMPLO 8.2.5

En este análisis, si el vértice v satisface

$$predessor(v) = p \quad y \quad val(v) = t,$$

la etiqueta de v sobre la gráfica será (p, t) .

En las líneas 1 y 2, inicializamos el flujo como 0 en cada arista (véase la figura 8.2.7). A continuación, en las líneas 5-9, hacemos todas las etiquetas iguales a $null$. Luego, en las líneas 10 y 11, etiquetamos el vértice a como $(-, \infty)$. En la línea 12 hacemos $U = \{a\}$. Luego entramos al ciclo while (línea 13).

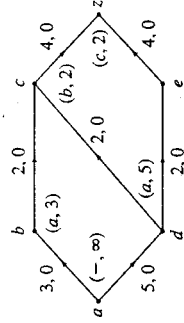


FIGURA 8.2.7 Después del primer etiquetado. El vértice v se etiqueta ($predessor(v), val(v)$).

Como z no está etiquetado y U no es vacío, pasamos a la línea 17, donde elegimos el vértice a en U y lo eliminamos de U en la línea 18. En este momento, $U = \emptyset$. Hacemos Δ igual a ∞ [$= \text{val}(a)$] en la línea 19. En la línea 20 examinamos las aristas (a, b) y (a, d) pues b y d no están etiquetados. Para la arista (a, b) tenemos

$$F_{ab} = 0 < C_{ab} = 3.$$

En las líneas 23 y 24, etiquetamos el vértice b como $(a, 3)$, ya que

$$\text{predesor}(b) = a$$

y

$$\text{val}(b) = \min(\Delta, 3 - 0) = \min(\infty, 3 - 0) = 3.$$

En la línea 25, agregamos b a U . De manera análoga, etiquetamos el vértice d como $(a, 5)$ y agregamos d a U . En este momento, $U = \{b, d\}$.

Ahora regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos un vértice en U . Suponga que elegimos a . Eliminamos b de U en la línea 18. Hacemos Δ igual a 3 [$= \text{val}(b)$] en la línea 19. En la línea 20 examinamos la arista (b, c) . En las líneas 23 y 24 etiquetamos el vértice c como $(b, 2)$ ya que

$$\text{predesor}(c) = b$$

y

$$\text{val}(c) = \min(\Delta, 2 - 0) = \min(3, 2 - 0) = 2.$$

En la línea 25 agregamos c a U . En este punto, $U = \{c, d\}$.

Ahora regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos un vértice en U . Suponga que elegimos a . Eliminamos c de U en la línea 18. Hacemos Δ igual a 2 [$= \text{val}(c)$] en la línea 19. En la línea 20 examinamos la arista (c, z) . En las líneas 23 y 24 etiquetamos el vértice z como $(c, 2)$. En la línea 25 agregamos z a U . En este punto, $U = \{d, z\}$.

Ahora regresamos a la parte superior del ciclo while (línea 13). Como z está etiquetado, pasamos a la línea 35. En las líneas 35-42, seguimos los predecesores desde z para determinar el camino

$$P = (a, b, c, z)$$

de a a z . En la línea 42 hacemos Δ igual a 2. Como cada arista de P está orientada en forma propia, en la línea 48 incrementamos el flujo en cada arista de P en $\Delta = 2$ unidades, para obtener la figura 8.2.8.

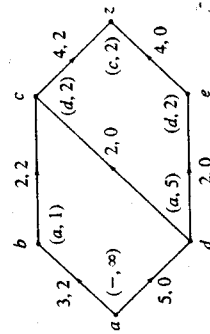


FIGURA 8.2.8 Después de incrementar el flujo en el camino (a, b, c, z) en 2 unidades y del segundo etiquetado.

Ahora regresamos a la parte superior del ciclo while (línea 3). Luego, en las líneas 5-9 hacemos todas las etiquetas iguales a *null*. Entonces, en las líneas 10 y 11, etiquetamos el vértice a como $(-, \infty)$ (véase la figura 8.2.8). En la línea 12 hacemos $U = \{a\}$. Luego entramos al ciclo while (línea 13).

Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos el vértice a en U y lo eliminamos de U en la línea 18. En las líneas 23 y 24 etiquetamos el vértice b como $(a, 1)$ y el vértice d como $(a, 5)$. Agregamos b y d a U , de modo que $U = \{b, d\}$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos un vértice en U . Suponga que elegimos b . Eliminamos b de U en la línea 18. En la línea 20 examinamos la arista (b, c) . Como $F_{bc} = C_{bc}$, no etiquetamos el vértice c en este punto. Ahora, $U = \{d\}$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos el vértice d en U y lo eliminamos de U en la línea 18. En las líneas 23 y 24 etiquetamos el vértice c como $(d, 2)$ y al vértice e como $(d, 2)$. Agregamos c y e a U , de modo que $U = \{c, e\}$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos un vértice en U . Suponga que elegimos c en U y lo eliminamos de U en la línea 18. En las líneas 23 y 24 etiquetamos el vértice z como $(c, 2)$. Agregamos z a U , de modo que $U = \{z, e\}$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z está etiquetado, pasamos a la línea 35. En la línea 42 tenemos que

$$P = (a, d, c, z).$$

Como cada arista de P está orientada en forma propia, en la línea 48 incrementamos el flujo en cada arista de P en $\Delta = 2$ unidades para obtener la figura 8.2.9.

El lector debe verificar que la siguiente iteración del algoritmo produce el etiquetado de la figura 8.2.9. Al incrementar el flujo en $\Delta = 2$ unidades se obtiene la figura 8.2.10.

Ahora regresamos a la parte superior del ciclo while (línea 3). A continuación, en las líneas 5-9 hacemos todas las etiquetas iguales a *null*. Entonces, en las líneas 10 y 11 etiquetamos el vértice a como $(-, \infty)$ (véase la figura 8.2.10). En la línea 12 hacemos $U = \{a\}$. Luego entramos al ciclo while (línea 13).

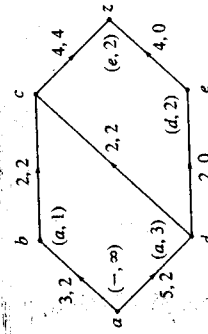


FIGURA 8.2.9 Después de incrementar el flujo en el camino (a, d, c, z) en 2 unidades y del tercer etiquetado.

Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos el vértice a en U y lo eliminamos de U en la línea 18. En las líneas 23 y 24 etiquetamos el vértice b como $(a, 1)$ y el vértice d como $(a, 1)$. Agregamos b y d a U , de modo que $U = \{b, d\}$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos un vértice en U . Suponga que elegimos b . Eliminamos b de U en la línea 18. En la línea 20 examinamos la arista (b, c) . Como $F_{bc} = C_{bc}$, no etiquetamos el vértice c . Ahora, $U = \{d\}$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos el vértice d en U y lo eliminamos de U en la línea 18. En la línea 20 examinamos las aristas (d, c) y (d, e) . Como $F_{dc} = C_{dc}$ y $F_{de} = C_{de}$, no etiquetamos el vértice c ni el vértice e . Ahora, $U = \emptyset$.

Luego regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado, pasamos a la línea 15. Como U es vacío, el algoritmo termina. El flujo de la figura 8.2.10 es máximo.

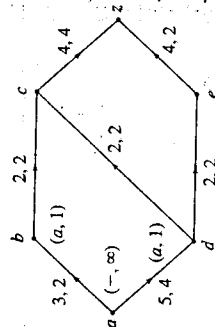


FIGURA 8.2.10 Después de incrementar el flujo en el camino (a, d, e, z) en 2 unidades y del cuarto y último etiquetado. El flujo es máximo. \square

Nuestro último ejemplo muestra la forma de modificar el algoritmo 8.2.4 para generar un flujo máximo a partir de un flujo dado.

EJEMPLO 8.2.6

Reemplazar el flujo nulo de las líneas 1 y 2 del algoritmo 8.2.4 con el flujo de la figura 8.2.11 y determinar luego un flujo máximo.

Después de inicializar el flujo dado, pasamos a las líneas 5-9, donde hacemos todas las etiquetas iguales a $null$. Luego, en las líneas 10 y 11, etiquetamos el vértice a como $(-, \infty)$ (véase la figura 8.2.11). En la línea 12 hacemos $U = \{a\}$. Luego entramos al ciclo while (línea 13).

Como z no está etiquetado y U no es vacío, pasamos a la línea 17, donde elegimos el vértice a en U y lo eliminamos de U en la línea 18. En las líneas 23 y 24, etiquetamos el vértice b como $(a, 1)$ y el vértice d como $(a, 1)$. Agregamos b y d a U de modo que $U = \{b, d\}$.

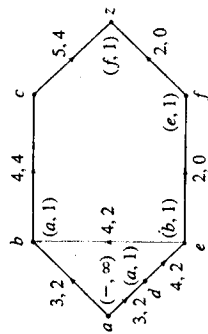


FIGURA 8.2.11 Después del etiquetado.

Ahora regresamos a la parte superior del ciclo while (línea 13). Como z no está etiquetado y U no es vacío, nos movemos a la línea 17, donde elegimos un vértice en U . Suponga que elegimos a . Eliminamos a de U en la línea 18. En la línea 20 examinamos las aristas (b, c) y (e, b) . Como $F_{bc} = C_{bc}$, no etiquetamos el vértice c . En las líneas 30 y 31, el vértice e se etiqueta $(b, 1)$ ya que

$$\text{val}(e) = \min\{\text{val}(b), F_{eb}\} = \min\{1, 2\} = 1.$$

Luego regresamos a la parte superior del ciclo while (línea 13). Finalmente etiquetamos z (véase la figura 8.2.11) y en la línea 42 determinamos el camino

$$P = (a, b, e, f, z).$$

Las aristas (a, b) , (e, f) y (f, z) están orientadas en forma propia, de modo que el flujo en cada una se incrementa en 1. Como la arista (e, b) está orientada en forma impropia, su flujo se decrementa en 1 unidad. Obtenemos el flujo de la figura 8.2.12.

Otra iteración del algoritmo produce el flujo máximo de la figura 8.2.13. \square

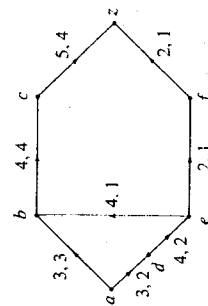


FIGURA 8.2.12 Después de incrementar el flujo del camino (a, b, e, f, z) en 1 unidad. Observe que la arista (e, b) está orientada en forma impropia, por lo que su flujo se disminuye en 1 unidad.

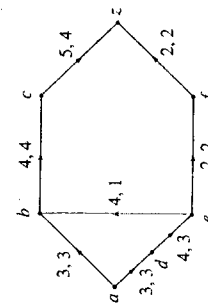
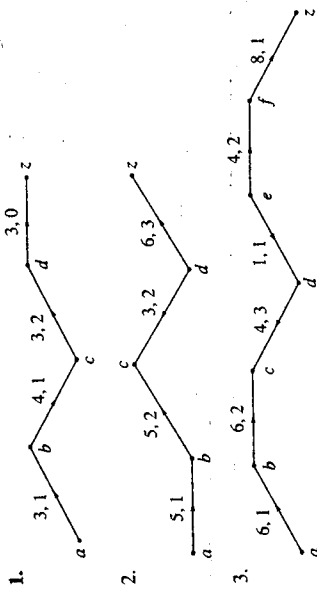


FIGURA 8.2.13 Después de incrementar el flujo del camino (a, d, e, f, z) en 1 unidad. El flujo es máximo.

Ejercicios

En los ejercicios 1-3 se da un camino de la fuente a al sumidero z en una red. Determine el máximo incremento posible en el flujo, que se pueda obtener alterando los flujos en las aristas en el camino.

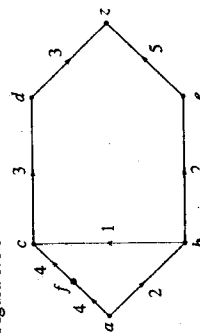


En los ejercicios 4-12, utilice el algoritmo 8.2.4 para determinar un flujo máximo en cada red.

4. Figura 8.1.4

5. Figura 8.1.5

6.



7. Ejercicio 5, sección 8.1

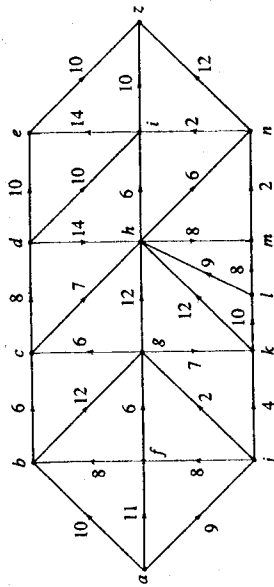
8. Ejercicio 6, sección 8.1

9. Ejercicio 7, sección 8.1

10. Ejercicio 8, sección 8.1

11. Ejercicio 9, sección 8.1

12.



En los ejercicios 13-18, determine un flujo máximo en cada red, comenzando con el flujo dado.

13. Figura 8.1.2

14. Ejercicio 1, sección 8.1

15. Ejercicio 2, sección 8.1

16. Ejercicio 3, sección 8.1

17. La figura 8.1.4, con los flujos

$$\begin{aligned} F_{a,w_1} &= 2, & F_{w_1,b} &= 2, & F_{ba} &= 0, & F_{ca} &= 0, & F_{Az} &= 0, \\ F_{a,w_2} &= 0, & F_{w_2,b} &= 0, & F_{bc} &= 2, & F_{cb} &= 4, & F_{Bz} &= 4, \\ F_{a,w_3} &= 2, & F_{w_3,d} &= 2, & F_{dc} &= 2. \end{aligned}$$

18. La figura 8.1.4, con los flujos

$$\begin{aligned} F_{a,w_1} &= 1, & F_{w_1,b} &= 1, & F_{ba} &= 4, & F_{ca} &= 2, & F_{Az} &= 6, \\ F_{a,w_2} &= 3, & F_{w_2,b} &= 3, & F_{bc} &= 0, & F_{cb} &= 1, & F_{Bz} &= 1, \\ F_{a,w_3} &= 3, & F_{w_3,d} &= 3, & F_{dc} &= 3. \end{aligned}$$

19. Muestre que el algoritmo 8.2.4 concluye.

8.3 EL TEOREMA DEL FLUJO MÁXIMO Y CORTE MÍNIMO

En esta sección mostraremos que al concluir el algoritmo 8.2.4, el flujo en la red es máximo. Mientras tanto, definiremos y analizaremos los cortes en las redes.

Sea G una red y consideremos el flujo F al concluir el algoritmo 8.2.4. Algunos vértices están etiquetados y otros no. Sea $P(\bar{P})$ el conjunto de vértices etiquetados (no etiquetados). (Recuerde que \bar{P} denota el complemento de P .) Entonces la fuente a está en P y el sumidero z está en \bar{P} . El conjunto S de aristas (v, w) , con $v \in P$ y $w \in \bar{P}$ es un corte, y la suma de las capacidades de las aristas en S es la **capacidad del corte**. Veremos que este corte tiene una capacidad mínima y, como un corte mínimo corresponde a un flujo máximo (teorema 8.3.9), el flujo F es máximo. Comenzamos con la definición formal de corte.

En esta sección, G es una red con fuente a y sumidero z . La capacidad de la arista (i, j) es C_{ij} .

DEFINICIÓN 8.3.1

Un **corte** (P, \bar{P}) en G consta de un conjunto P de vértices y el complemento \bar{P} de P , con $a \in P$ y $z \in \bar{P}$.

EJEMPLO 8.3.2

Consideremos la red G de la figura 8.3.1. Si hacemos $P = \{a, b, d\}$, entonces $\bar{P} = \{c, e, f, z\}$ y (P, \bar{P}) es un corte en G . Como muestra la figura, a veces indicamos un corte trazando una línea punteada para separar los vértices.

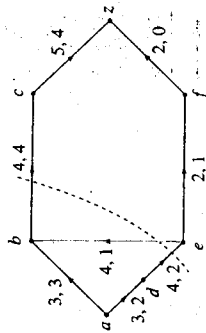


FIGURA 8.3.1 Un corte en una red. La línea punteada divide los vértices en los conjuntos $P = \{a, b, d\}$ y $\bar{P} = \{c, e, f, z\}$ produciendo el corte (P, \bar{P}) . □

EJEMPLO 8.3.3

La figura 8.2.10 muestra el etiquetado al final del algoritmo 8.2.4 para una red particular. Si $P(\bar{P})$ denota el conjunto de vértices etiquetados (no etiquetados), obtenemos el corte que se muestra en la figura 8.3.2.

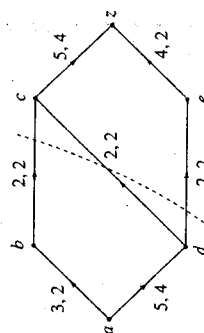


FIGURA 8.3.2 Una red al finalizar el algoritmo 8.2.4. El corte (P, \bar{P}) , $P = \{a, b, d\}$ se obtiene al definir P como el conjunto de vértices etiquetados. □

Ahora definiremos la capacidad de un corte.

DEFINICIÓN 8.3.4

La capacidad del corte (P, \bar{P}) es el número

$$C(P, \bar{P}) = \sum_{i \in P} \sum_{j \in \bar{P}} C_{ij}.$$

EJEMPLO 8.3.5

La capacidad del corte de la figura 8.3.1 es

$$C_{bc} + C_{de} = 8.$$

□

EJEMPLO 8.3.6

La capacidad del corte de la figura 8.3.2 es

$$C_{bc} + C_{de} + C_{de} = 6.$$

□

El siguiente teorema muestra que la capacidad de cualquier corte siempre es mayor o igual que el valor de cualquier flujo.

TEOREMA 8.3.7

Sea F un flujo en G y sea (P, \bar{P}) un corte en G . Entonces la capacidad de (P, \bar{P}) es mayor o igual que el valor de F ; es decir,

$$\sum_{i \in P} \sum_{j \in \bar{P}} C_{ij} \geq \sum_{i \in P} F_{ai}. \quad (8.3.1)$$

(La notación $\sum_{i \in P}$ significa la suma sobre todos los vértices i .)

Demostración. Observe que

$$\sum_{j \in P} F_{ji} = \sum_{j \in P} \sum_{i \in P} F_{ij}.$$

pues cada lado de la ecuación es simplemente la suma de F_{ij} sobre todas las $i, j \in P$. Ahora,

$$\begin{aligned} \sum_i F_{ai} &= \sum_{j \in P} \sum_{i \in P} F_{ji} - \sum_{j \in P} \sum_{i \in \bar{P}} F_{ji} \\ &= \sum_{j \in P} \sum_{i \in P} F_{ji} + \sum_{j \in P} \sum_{i \in \bar{P}} F_{ji} - \sum_{j \in P} \sum_{i \in \bar{P}} F_{ji} - \sum_{j \in P} \sum_{i \in \bar{P}} F_{ij} \\ &= \sum_{j \in P} \sum_{i \in \bar{P}} F_{ji} - \sum_{j \in P} \sum_{i \in \bar{P}} F_{ij} \leq \sum_{j \in P} \sum_{i \in \bar{P}} F_{ji} \leq \sum_{j \in P} \sum_{i \in \bar{P}} C_{ij}. \end{aligned}$$

■

EJEMPLO 8.3.8

En la figura 8.3.1, el valor 5 del flujo es menor que la capacidad 8 del corte. □

Un corte mínimo es un corte con capacidad mínima.

TEOREMA 8.3.9 Teorema del flujo máximo y el corte mínimo

Sea F un flujo en G y sea (P, \bar{P}) un corte en G . Si la igualdad se cumple en (8.3.1), entonces el flujo es máximo y el corte es mínimo. Además, la igualdad se cumple en (8.3.1) si y sólo si

$$(a) F_{ij} = C_{ij} \quad \text{para } i \in P, j \in \bar{P}$$

$$(b) F_{ij} = 0 \quad \text{para } i \in \bar{P}, j \in P.$$

Demostración. El primer enunciado es inmediato.

La demostración del teorema 8.3.7 muestra que la igualdad se cumple precisamente cuando

$$\sum_{j \in P} \sum_{i \in P} F_{ij} = 0 \quad \text{y} \quad \sum_{j \in P} \sum_{i \in P} F_{ji} = \sum_{j \in P} \sum_{i \in P} C_{ji}.$$

de modo que la última afirmación también es verdadera.

EJEMPLO 8.3.10

En la figura 8.3.2, el valor del flujo y la capacidad del corte son ambos iguales a 6; por tanto, el flujo es máximo y el corte es mínimo.

Podemos utilizar el teorema 8.3.9 para mostrar que el algoritmo 8.2.4 produce un flujo máximo.

TEOREMA 8.3.11

Al concluir, el algoritmo 8.2.4 produce un flujo máximo. Además, si P (respectivamente, \bar{P}) es el conjunto de vértices etiquetados (respectivamente, no etiquetados) al concluir el algoritmo 8.2.4, el corte (P, \bar{P}) es mínimo.

Demostración. Sea \bar{P} el conjunto de vértices etiquetados (no etiquetados) de G al concluir el algoritmo 8.2.4. Consideremos una arista (i, j) , donde $i \in P, j \in \bar{P}$. Como i está etiquetado, debemos tener

$$F_{ij} = C_{ij};$$

en caso contrario, habríamos etiquetado j en las líneas 23 y 24. Ahora, consideremos una arista (j, i) , donde $j \in \bar{P}, i \in P$. Como i está etiquetado, debemos tener

$$F_{ji} = 0;$$

en caso contrario, habríamos etiquetado j en las líneas 30 y 31. Por el teorema 8.3.9, al concluir el algoritmo 8.2.4, el flujo es máximo y el corte (P, \bar{P}) es mínimo.

Ejercicios

En los ejercicios 1-3, determine la capacidad del corte (P, \bar{P}) . Además, determine si el corte es mínimo.

1. $P = \{a, d\}$ para el ejercicio 1, sección 8.1
2. $P = \{a, d, e\}$ para el ejercicio 2, sección 8.1
3. $P = \{a, b, c, d\}$ para el ejercicio 3, sección 8.1

En los ejercicios 4-16, determine un corte mínimo en cada red.

4. Figura 8.1.1
5. Figura 8.1.4
6. Figura 8.1.5
7. Ejercicio 1, sección 8.1
8. Ejercicio 2, sección 8.1
9. Ejercicio 3, sección 8.1
10. Ejercicio 4, sección 8.1
11. Ejercicio 5, sección 8.1
12. Ejercicio 6, sección 8.1
13. Ejercicio 7, sección 8.1
14. Ejercicio 8, sección 8.1
15. Ejercicio 9, sección 8.1
16. Ejercicio 12, sección 8.2

Los ejercicios 17-22 se refieren a una red G tal que, además de tener capacidades C_{ij} dadas por enteros no negativos, tiene requisitos de flujo mínimo en una arista dados por enteros no negativos m_{ij} . Es decir, un flujo F debe satisfacer

$$m_{ij} \leq F_{ij} \leq C_{ij}$$

para todas las aristas (i, j) .

17. Proporcione un ejemplo de una red G , en la cual $m_{ij} \leq C_{ij}$ para todas las aristas (i, j) , de modo que no exista un flujo.

Defina

$$C(\bar{P}, P) = \sum_{i \in \bar{P}} \sum_{j \in P} C_{ij}, \quad m(\bar{P}, P) = \sum_{i \in \bar{P}} \sum_{j \in P} m_{ij}, \quad m(\bar{P}, P) = \sum_{i \in \bar{P}} \sum_{j \in P} m_{ij}.$$

18. Muestre que el valor V de cualquier flujo satisface

$$m(\bar{P}, \bar{P}) - C(\bar{P}, P) \leq V \leq C(P, \bar{P}) - m(P, P)$$

para cualquier corte (P, \bar{P}) .

19. Muestre que si existe un flujo en G , entonces existe un flujo máximo en G con valor

$$\min \{C(P, \bar{P}) - m(\bar{P}, P) \mid (P, \bar{P}) \text{ es un corte en } G\}.$$

20. Suponga que G tiene un flujo F . Desarrolle un algoritmo para determinar un flujo máximo en G .

21. Muestre que si existe un flujo en G , entonces existe un flujo mínimo en G con valor

$$\max \{m(P, \bar{P}) - C(\bar{P}, P) \mid (P, \bar{P}) \text{ es un corte en } G\}.$$

22. Suponga que G tiene un flujo F . Desarrolle un algoritmo para determinar un flujo mínimo en G .

23. ¿Verdadero o falso? Si F es un flujo en una red G , (P, \bar{P}) es un corte en G y la capacidad de (P, \bar{P}) excede al valor del flujo F , entonces el corte (P, \bar{P}) no es mínimo y el flujo F no es máximo. Si es verdadero, demuéstrelo; en caso contrario, proporcione un contraejemplo.

8.4 ACOPLAMIENTO

En esta sección consideraremos el problema de hacer corresponder los elementos de un conjunto con los elementos de otro conjunto. Veremos que este problema se puede reducir a determinar un flujo máximo en una red. Comenzaremos con un ejemplo.

EJEMPLO 8.4.1

Supongamos que cuatro personas A, B, C y D realizan una solicitud para cinco trabajos J_1, J_2, J_3, J_4 y J_5 . Suponga que el solicitante A está calificado para los trabajos J_1, J_2 y J_3 ; el solicitante B está calificado para los trabajos J_2 y J_3 ; el solicitante C está calificado para los trabajos J_1, J_3, J_4 y J_5 ; y el solicitante D está calificado para los trabajos J_2 y J_3 . ¿Es posible hallar un trabajo para cada solicitante?

La situación se puede modelar mediante la gráfica de la figura 8.4.1. Los vértices representan a los solicitantes y los trabajos. Una arista une un solicitante con un trabajo para el cual el solicitante está calificado. Podemos mostrar que no es posible asociar un trabajo a cada solicitante, si nos fijamos en los solicitantes A, B y D , quienes son calificados para los trabajos J_2 y J_3 . Si A y B reciben un trabajo, no quedaría un trabajo para D . Por tanto, no existe una asignación para A, B, C y D . \square

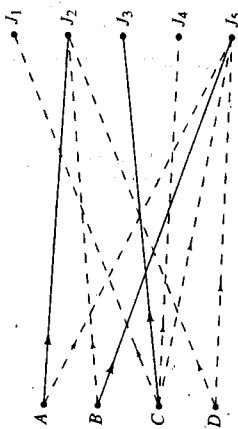


FIGURA 8.4.1 Solicitantes (A, B, C, D) y trabajos (J_1, J_2, J_3, J_4, J_5). Una arista une un solicitante con un trabajo para el cual está calificado. Las líneas continuas muestran un acoplamiento máximo (es decir, el mayor número de solicitantes tiene trabajo).

En el ejemplo 8.4.1, un acoplamiento consiste en hallar trabajos para las personas calificadas. Un acoplamiento máximo encuentra trabajo para el mayor número de personas. En la gráfica de la figura 8.4.1 aparece con líneas continuas un acoplamiento máximo. Un acoplamiento completo encuentra trabajo para todos. Hemos mostrado que la gráfica de la figura 8.4.1 no tiene un acoplamiento completo. A continuación damos las definiciones formales.

DEFINICIÓN 8.4.2

Sea G una gráfica dirigida, bipartita, con conjuntos ajenos de vértices V y W en los cuales las aristas se dirigen de los vértices en V a los vértices en W . (Cualquier vértice en G está en V o en W .) Un **acoplamiento** para G es un conjunto de aristas E sin vértices en común. Un **acoplamiento máximo** para G es un acoplamiento E tal que E contiene el número máximo de aristas. Un **acoplamiento completo** para G es un acoplamiento E con la propiedad de que si $v \in V$, entonces $(v, w) \in E$ para algún $w \in W$.

EJEMPLO 8.4.3

El acoplamiento para la gráfica de la figura 8.4.2, que aparece con líneas continuas, es un acoplamiento máximo y un acoplamiento completo. \square

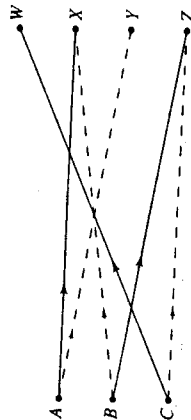


FIGURA 8.4.2 Las líneas continuas muestran un acoplamiento máximo (se utiliza el máximo número de aristas) y un acoplamiento completo (se establece una correspondencia de A, B y C).

En el siguiente ejemplo ilustraremos la forma de utilizar el problema de acoplamiento como un problema de redes.

EJEMPLO 8.4.4

Una red de acoplamiento

Modele el problema de acoplamiento del ejemplo 8.4.1 como una red.

Primero asignamos a cada arista de la gráfica de la figura 8.4.1 la capacidad 1 (véase la figura 8.4.3). A continuación agregamos una superfuente a y aristas de capacidad 1 de a a cada uno de los vértices A, B, C y D . Por último, agregamos un sumidero y aristas de capacidad 1 desde J_1, J_2, J_3, J_4 y J_5 a z . Una red como la de la figura 8.4.3 es una **red de acoplamiento**.

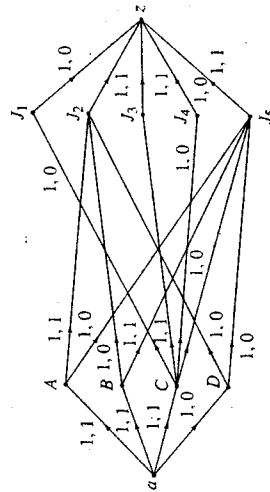


FIGURA 8.4.3 El problema de acoplamiento (figura 8.4.1) como una red de acoplamiento. \square

El siguiente teorema relaciona las redes de acoplamiento y los flujos.

TEOREMA 8.4.5

Sea G una gráfica dirigida, bipartita, con conjuntos ajenos de vértices V y W en la cual las aristas están dirigidas de los vértices de V a los vértices de W . (Cualquier vértice de G está en V o en W .)

- (a) Un flujo en la red de acoplamiento proporciona un acoplamiento en G . El vértice $v \in V$ se asocia con el vértice $w \in W$ si y sólo si el flujo en la arista (v, w) es 1.
- (b) Un flujo máximo corresponde a un acoplamiento máximo.
- (c) Un flujo con valor $|V|$ corresponde a un acoplamiento completo.

Demostración. Sea $a(z)$ la fuente (sumidero) en la red de acoplamiento y supongamos dado un flujo.

Supongamos que la arista (v, w) , $v \in V, w \in W$, tiene flujo 1. La única arista hacia el vértice v es (a, v) . Esta arista debe tener flujo 1; así, el flujo hacia el vértice v es 1. Como el flujo de salida de v también es 1, la única arista de la forma (v, x) que tiene flujo 1 es (v, w) . De manera análoga, la única arista de la forma (x, w) que tiene flujo 1 es (v, w) . Por tanto, si E es el conjunto de aristas de la forma (v, w) con flujo 1, los miembros de E no tienen vértices en común, por lo que E es un acoplamiento para G .

Las partes (b) y (c) son consecuencia del hecho de que el número de vértices en V asociados mediante este acoplamiento es igual al valor del flujo correspondiente. ■

Como un flujo máximo proporciona un acoplamiento máximo, el algoritmo 8.2.4 aplicado a una red de acoplamiento produce un acoplamiento máximo. En la práctica, se puede simplificar la implantación del algoritmo 8.2.4 utilizando la matriz de adyacencia de la gráfica (véase el ejercicio 11).

EJEMPLO 8.4.6

El acoplamiento de la figura 8.4.1 se representa como un flujo en la figura 8.4.3. Como el flujo es máximo, el acoplamiento es máximo. □

Ahora analizaremos la existencia de un acoplamiento completo en una gráfica dirigida bipartita G con conjuntos de vértices V y W . Si $S \subseteq V$, sea

$$R(S) = \{ w \in W \mid v \in S \text{ y } (v, w) \text{ es una arista en } G \}.$$

Suponga que G tiene un acoplamiento completo. Si $S \subseteq V$ debemos tener

$$|S| \leq |R(S)|.$$

Ocurre que si $|S| \leq |R(S)|$ para todos los subconjuntos S de V , entonces G tiene un acoplamiento completo. Este resultado fue planteado por vez primera por el matemático inglés Philip Hall y se conoce como el **teorema de los matrimonios de Hall**, pues si V es un conjunto de hombres y W es un conjunto de mujeres, y si existe una arista de $v \in V$ en $w \in W$ si v y w son compatibles, el teorema proporciona una condición bajo la cual cada hombre puede casarse con una mujer compatible.

TEOREMA 8.4.7

Teorema de los matrimonios de Hall

Sea G una gráfica dirigida, bipartita, con conjuntos ajenos de vértices V y W en la cual las aristas se dirigen de los vértices en V a los vértices en W . (Cualquier vértice en G está en V o en W .) Existe un acoplamiento completo en G si y sólo si

$$|S| \leq |R(S)| \quad \text{para todo } S \subseteq V. \quad (8.4.1)$$

Demostración. Ya hemos señalado que si existe un acoplamiento completo en G , entonces se cumple la condición (8.4.1).

Supongamos que se cumple la condición (8.4.1). Sea $n = |V|$ y sea (P, \bar{P}) un corte mínimo en la red de acoplamiento. Si podemos mostrar que la capacidad de este corte es n , un flujo máximo tendrá valor n . El acoplamiento correspondiente a este flujo máximo sería un acoplamiento completo.

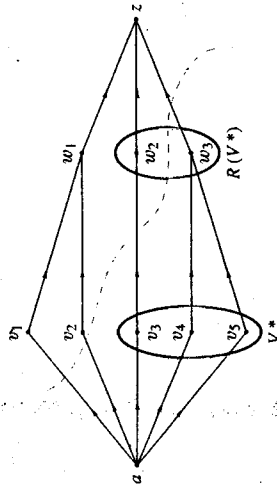


FIGURA 8.4.4 La demostración del teorema 8.4.7. $V = \{v_1, v_2, v_3, v_4, v_5\}$; $n = |V| = 5$; $W = \{w_1, w_2, w_3\}$; el corte es (P, \bar{P}) , donde $P = \{v_1, v_2, v_3, w_1, w_2, w_3\}$; $V^* = V \cap P = \{v_1, v_2, v_3\}$; $R(V^*) = \{w_1, w_2, w_3\}$; $W_1 = R(V^*) \cap P = \{w_1\}$; $W_2 = R(V^*) \cap \bar{P} = \{w_2, w_3\}$; $E = \{(v_1, w_1), (v_2, w_1), (v_3, w_2), (v_4, w_2), (v_5, w_2)\}$. La capacidad del corte es $|E| = 4 < n$. Las aristas de tipo I son (v_1, w_1) y (v_2, w_1) . La arista (v_3, w_2) es la única arista de tipo II, y la arista (v_4, v_5) es la única arista de tipo III.

El argumento es por contradicción. Suponga que la capacidad del corte mínimo (P, \bar{P}) es menor que n . La capacidad de este corte es el número de aristas en el conjunto

$$E = \{(x, y) \mid x \in P, y \in \bar{P}\}$$

(véase la figura 8.4.4). Un miembro de E es de uno de los tres tipos

Tipo I: (a, v) , $v \in V$.

Tipo II: (v, w) , $v \in V, w \in W$.

Tipo III: (w, z) , $w \in W$.

Estimaremos el número de aristas de cada tipo.

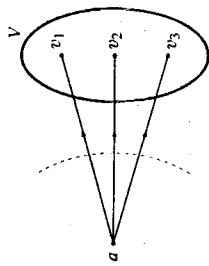


FIGURA 8.4.5 La demostración del teorema 8.4.7 para $n = 3$. Si $V \subseteq \bar{P}$, se puede ver en la figura que la capacidad del corte es n . Como estamos suponiendo que la capacidad del corte es menor que n , este caso no puede ocurrir. Por tanto, $V \cap P$ es no vacío.

Si $V \subseteq \bar{P}$, la capacidad del corte es n (véase la figura 8.4.5); así,

$$V^* = V \cap P$$

es no vacío. Esto implica que existen $n - |V^*|$ aristas en E de tipo I. Separamos $R(V^*)$ en los conjuntos

$$W_1 = R(V^*) \cap P \quad \text{y} \quad W_2 = R(V^*) \cap \bar{P}.$$

Entonces existen al menos $|W_1|$ aristas en E de tipo III. Así, existen menos de

$$n - (n - |V^*|) - |W_1| = |V^*| - |W_1|$$

aristas de tipo II en E . Como cada miembro de W_2 contribuye a lo más con una arista de tipo II,

$$|W_2| < |V^*| - |W_1|.$$

Así,

$$|R(V^*)| = |W_1| + |W_2| < |V^*|,$$

lo cual contradice (8.4.1). Por tanto, existe un acoplamiento completo. ■

EJEMPLO 8.4.8

Para la gráfica de la figura 8.4.1, si $S = \{A, B, D\}$, tenemos que $R(S) = \{J_2, J_3\}$ y

$$|S| = 3 > 2 = |R(S)|.$$

Por el teorema 8.4.7, no existe un acoplamiento completo para la gráfica de la figura 8.4.1. □

EJEMPLO 8.4.9

Existen n computadoras y n unidades de disco. Cada computadora es compatible con $m > 0$ unidades de disco y cada unidad de disco es compatible con n computadoras. ¿Es posible asociar cada computadora con una unidad de disco compatible?

Sea V el conjunto de computadoras y W el conjunto de unidades de disco. Existe una arista de $v \in V$ en $w \in W$ si v y w son compatibles. Observe que cada vértice tiene grado

m . Sea $S = \{v_1, \dots, v_k\}$ un subconjunto de V . Entonces existen km aristas que salen del conjunto S . Si $R(S) = \{w_1, \dots, w_j\}$, entonces $R(S)$ recibe a lo más jm aristas de S . Por tanto,

$$km \leq jm.$$

Ahora,

$$|S| = k \leq j = |R(S)|.$$

Por el teorema 8.4.7, existe un acoplamiento completo. Así, es posible asociar cada computadora con una unidad de disco compatible. □

Ejercicios

1. Muestre que el flujo de la figura 8.4.3 es máximo, exhibiendo un corte mínimo cuya capacidad sea 3.
2. Determine el flujo que corresponde al acoplamiento de la figura 8.4.2. Muestre que este flujo es máximo, exhibiendo un corte mínimo cuya capacidad sea 3.
3. El solicitante A está calificado para los trabajos J_1 y J_2 ; B está calificado para los trabajos J_2, J_3 y J_6 ; C está calificado para los trabajos J_1, J_3, J_5 y J_6 ; D está calificado para los trabajos J_1, J_3 y J_4 ; y E está calificado para los trabajos J_1, J_3 y J_6 .
 - (a) Modele esta situación como una red de acoplamiento.
 - (b) Utilice el algoritmo 8.2.4 para determinar un acoplamiento máximo.
 - (c) ¿Existe un acoplamiento completo?
4. El solicitante A está calificado para los trabajos J_1, J_2, J_3 y J_4 ; B está calificado para los trabajos J_1, J_4 y J_5 ; C está calificado para los trabajos J_1, J_4 y J_5 ; D está calificado para los trabajos J_1 y J_5 ; E está calificado para los trabajos J_2, J_3 y J_6 ; y F está calificado para los trabajos J_4 y J_5 . Responda las partes (a)-(c) del ejercicio 3 para esta situación.
5. El solicitante A está calificado para los trabajos J_1, J_2 y J_4 ; B está calificado para los trabajos J_3, J_4, J_5 y J_6 ; C está calificado para los trabajos J_1 y J_5 ; D está calificado para los trabajos J_1, J_3, J_4 y J_8 ; E está calificado para los trabajos J_1, J_2, J_4, J_5 y J_8 ; F está calificado para los trabajos J_4 y J_6 ; y G está calificado para los trabajos J_3, J_5 y J_7 . Responda las partes (a)-(c) del ejercicio 3 para esta situación.
6. Cinco estudiantes, V, W, X, Y y Z , son miembros de cuatro comités, C_1, C_2, C_3 y C_4 . Los miembros de C_1 son V, X y Y ; los miembros de C_2 son X y Z ; los miembros de C_3 son V, Y y Z ; y los miembros de C_4 son V, W, X y Z . Cada comité debe enviar un representante a la administración. Ningún estudiante puede representar a dos comités.
 - (a) Modele esta situación como una red de acoplamiento.
 - (b) ¿Cuál es la interpretación de un acoplamiento máximo?
 - (c) ¿Cuál es la interpretación de un acoplamiento completo?
 - (d) Utilice el algoritmo 8.2.4 para determinar un acoplamiento máximo.
 - (e) ¿Existe un acoplamiento completo?
7. Muestre que con un orden adecuado de los vértices, la matriz de adyacencia de una gráfica bipartita se puede escribir como

$$\begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix},$$

donde 0 es una matriz que consta únicamente de ceros y A^T es la transpuesta de la matriz A .

En los ejercicios 8-10, G es una gráfica bipartita, A es la matriz del ejercicio 7, y F un flujo en la red de acoplamiento asociada. Etiquete cada entrada en A que represente una ansia con el flujo igual a 1.

8. ¿Cuál tipo de etiquetado corresponde a un acoplamiento?
9. ¿Cuál tipo de etiquetado corresponde a un acoplamiento completo?
10. ¿Cuál tipo de etiquetado corresponde a un acoplamiento máximo?
11. Enuncie el algoritmo 8.2.4, aplicado a una red de acoplamiento, en términos de operaciones de la matriz A del ejercicio 7.

Sea G una gráfica dirigida, bipartita, con conjuntos ajenos de vértices V y W en los cuales las aristas se dirigen de los vértices en V a los vértices en W . (Cualquier vértice en G está en V o en W .) Definimos la deficiencia de G como

$$\delta(G) = \max\{|S| - |R(S)| \mid S \subseteq V\}.$$

12. Muestre que G tiene un acoplamiento completo si y sólo si $\delta(G) = 0$.
13. Muestre que el número máximo de vértices en V que se pueden asociar con vértices de W es $|V| - \delta(G)$.
14. ¿Verdadero o falso? Cualquier acoplamiento está contenido en un acoplamiento máximo. De ser verdadero, demuéstrello; de ser falso, proporcione un contraejemplo.

RINCÓN DE SOLUCIÓN DE PROBLEMAS:

ACOPLAMIENTO

Problema

Sea G una gráfica dirigida, bipartita, con conjuntos ajenos de vértices V y W en los cuales las aristas se dirigen de los vértices en V a los vértices en W . (Cualquier vértice en G está en V o en W .) Sea M_w el grado máximo entre los vértices de W y sea m_v el grado mínimo entre los vértices de V . Muestre que si $0 < M_w \leq m_v$, entonces G tiene un acoplamiento completo.

Para enfrentar el problema

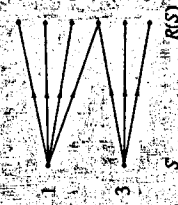
El teorema de los matrimonios de Hall (teorema 8.4.7) dice que una gráfica dirigida, bipartita con conjuntos ajenos de vértices V y W tiene un acoplamiento completo si y sólo si $|S| \leq |R(S)|$ para todo $S \subseteq V$. Así, una posible forma de resolver el problema es mostrar que la condición dada $M_w \leq m_v$ implica que $|S| \leq |R(S)|$ para todo $S \subseteq V$.

Determinación de una solución

Nuestro objetivo es demostrar que si $M_w \leq m_v$, entonces $|S| \leq |R(S)|$ para todo $S \subseteq V$. Comencemos con una gráfica ejemplo G para la cual $M_w \leq m_v$, de hecho, podemos hacer que $M_w = 2$ y $m_v = 3$.



Consideremos un subconjunto ejemplo $S = \{1, 3\}$ de V y las aristas incidentes en los vértices de S :



El hecho de que el grado mínimo de los vértices de V sea 3 significa que para cualquier subconjunto S de V , al menos tres aristas son incidentes en cada vértice de S . En general, habrá al menos $3|S| = m_v|S|$ aristas incidentes en los vértices de S . En nuestro ejemplo, $3|S| = 6$, pero en realidad hay siete aristas incidentes en los vértices de S . La expresión $m_v|S|$ siempre es una cota inferior para el número de aristas incidentes en los vértices de S .

El hecho de que el grado máximo de los vértices de W sea 2 significa que para cualquier subconjunto S de V , a lo más dos aristas son incidentes en cada vértice de $R(S)$. En general, habrá $2|R(S)| = M_w|R(S)|$ aristas incidentes en los vértices de $R(S)$. En nuestro ejemplo, $2|R(S)| = 12$, pero en realidad existen 10 aristas incidentes en los vértices de $R(S)$. Como las aristas incidentes en S son un subconjunto de las aristas incidentes en los vértices de $R(S)$, la expresión $M_w|R(S)|$ es siempre una cota superior para el número de aristas incidentes en los vértices de S .

Tenemos dos formas de estimar el número de aristas incidentes en los vértices de S . La primera forma, utilizando S , nos proporciona una cota inferior $m_v|S|$ sobre el número de aristas, y la segunda, utilizando $R(S)$, nos proporciona una cota superior $M_w|R(S)|$ sobre el número de aristas. Al comparar estas estimaciones obtenemos la desigualdad

$$m_v|S| \leq M_w|R(S)|.$$

No podemos deducir que $|S| \leq |R(S)|$, pero no hemos utilizado la hipótesis

$$M_w \leq m_v.$$

¡Ahí! Al combinar las dos últimas desigualdades, tenemos

$$m_v|S| \leq M_w|R(S)| \leq m_v|R(S)|.$$

Si ahora cancelamos m_v en ambos extremos de la desigualdad, obtenemos

$$|S| \leq |R(S)|$$

que es exactamente la desigualdad que queríamos demostrar.

Solución formal

Sea $S \subseteq V$. Cada vértice en S incide en al menos $m_v|S|$ aristas, así, existen al menos $m_v|S|$ aristas incidentes en los vértices de S . Cada vértice en $R(S)$ incide en a lo más

M_p aristas, así, existen a lo más $M_p |R(S)|$ aristas incidentes en los vértices de $R(S)$. Esto implica que $m_p |S| \leq M_p |R(S)|$. Como $M_p \leq m_p$, tenemos que $|R(S)| M_p \leq |R(S)| m_p$. Por tanto, $m_p |S| \leq m_p |R(S)|$ y $|S| \leq |R(S)|$. Por el teorema 8.4.7, G tiene un acoplamiento completo.

Resumen de técnicas para la solución de problemas

- Analice algunas gráficas ejemplo.
- Al analizar los ejemplos, es conveniente asignar valores distintos a los parámetros del problema, de modo que pueda seguir su pista. (En nuestro ejemplo, hicimos $M_p = 2$ y $m_p = 3$.)
- Intente reducir las condiciones dadas a las de un teorema útil. (En este problema reducimos las condiciones dadas a las del teorema de los matrimonios de Hall.)
- A veces, una desigualdad se puede demostrar estimando el tamaño de cierto conjunto, de alguna forma. Si una estimación proporciona una cota superior M , y otra proporciona una cota inferior m , esto implica que $m \leq M$.

Comentarios

Esta última técnica para resolver problemas proporciona un método para demostrar una desigualdad. De manera análoga, a veces se puede demostrar contando el número de elementos de cierto conjunto de dos formas distintas. Si una forma de contar da como resultado c_1 y la otra da como resultado c_2 , entonces $c_1 = c_2$. Estas técnicas se utilizan ampliamente y no está de más insistir en su utilidad. Por ejemplo, en la sección 4.2 dedujimos una fórmula para $C(n, r)$ contando el número de r -permutaciones de un conjunto de n elementos de dos formas distintas.

EJERCICIO

1. Proporcione un ejemplo de una gráfica bipartita G que tenga un acoplamiento completo pero que no satisfaga la condición $M_p \leq m_p$.

8.5 REDES DE PETRI

Consideremos el programa de computadora de la figura 8.5.1. En general, las instrucciones se procesarían en forma secuencial; es decir, primero $A = 1$, luego $B = 2$, y así sucesivamente. Sin embargo, observe que no existe una razón lógica para evitar que las tres primeras instrucciones ($A = 1$, $B = 2$ y $C = 3$) se procesen en un orden arbitrario o de manera concurrente. Al disminuir el precio del hardware de computadora, y de los procesadores en particular, existe un creciente interés en el procesamiento concurrente, para lograr mayor velocidad y eficiencia. El uso de redes de Petri, gráficas modelo del procesamiento concurrente, es un método para modelar y estudiar el procesamiento concurrente.

$A = 1$
 $B = 2$
 $C = 3$

$A = A + 1$
 $C = B + C$
 $B = A + C$

FIGURA 8.5.1
Un programa de computadora.

DEFINICIÓN 8.5.1

Una red de Petri es una gráfica dirigida $G = (V, E)$, donde $V = P \cup T$ y $P \cap T = \emptyset$. Cualquier arista e en E es incidente en un miembro de P y un miembro de T . El conjunto P es el conjunto de lugares y el conjunto T es el conjunto de transiciones.

De manera informal, una red de Petri es una gráfica dirigida, bipartita, en la cual las dos clases de vértices se llaman lugares y transiciones. En general, en las redes de Petri se permite la existencia de aristas paralelas; sin embargo, para simplificar la exposición, aquí no permitiremos la existencia de aristas paralelas.

EJEMPLO 8.5.2

La figura 8.5.2 muestra un ejemplo de red de Petri. En general, los lugares se dibujan como círculos y las transiciones como barras o cajas rectangulares.

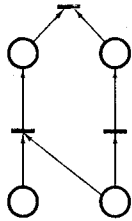


FIGURA 8.5.2 Una red de Petri. Los círculos son lugares y las barras son transiciones. □

DEFINICIÓN 8.5.3

Un marcado de una red de Petri asigna a cada lugar un entero no negativo. Una red de Petri con un marcado es una red de Petri marcada (o simplemente una red de Petri).

Si un marcado asigna el entero no negativo n al lugar p , decimos que existen n elementos en p . Los elementos se representan mediante puntos.

EJEMPLO 8.5.4

La figura 8.5.3 muestra un ejemplo de red de Petri marcada.

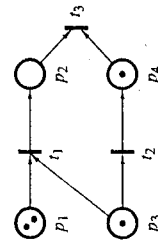


FIGURA 8.5.3 Una red de Petri marcada. □

Al modelar una situación, los lugares representan condiciones, las transiciones representan eventos, y la presencia de al menos un elemento en un lugar (condición) indica que tal condición se cumple.

EJEMPLO 8.5.5

Modelo de red de Petri para un programa de computadora

En la figura 8.5.4 hemos modelado el programa de computadora de la figura 8.5.1. En este caso, los eventos (transiciones) son las instrucciones, y los lugares representan las condiciones bajo las cuales se puede ejecutar una instrucción. □

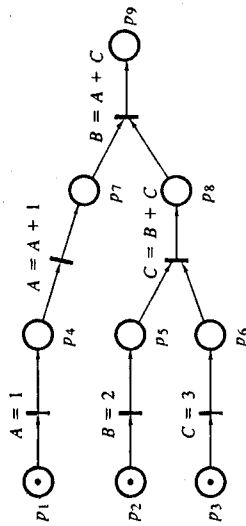


FIGURA 8.5.4 El programa de la figura 8.5.1 como una red de Petri. Los elementos indican que se cumplen las condiciones para ejecutar $A = 1$, $B = 2$ y $C = 3$.

DEFINICIÓN 8.5.6

En una red de Petri, si una arista va del lugar p a la transición t , decimos que p es un *lugar de entrada* para la transición t . Un *lugar de salida* se define de manera análoga. Si cada lugar de entrada de una transición t tiene al menos un elemento, decimos que t está *activada*. La *descarga* de una transición elimina un elemento de cada lugar de entrada y agrega un elemento a cada lugar de salida.

EJEMPLO 8.5.7

En la red de Petri de la figura 8.5.3, los lugares p_1 y p_3 son lugares de entrada para la transición t_1 . Las transiciones t_1 y t_2 están activadas, pero la transición t_3 no. Si descargamos la transición t_1 , obtenemos la red de Petri marcada de la figura 8.5.5. Ahora, la transición t_3 está activada. Si entonces descargamos la transición t_3 , obtenemos la red que se muestra. En este momento, ninguna transición está activada y por tanto ninguna se puede descargar. □

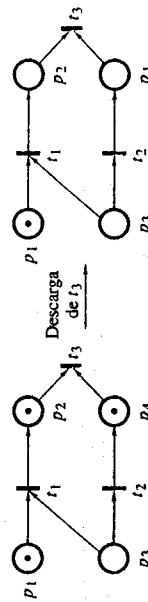


FIGURA 8.5.5 Descarga de la transición t_3 .

DEFINICIÓN 8.5.8

Si una serie de descargas transforma un marcado M en un marcado M' , decimos que M' es *alcanzable* desde M .

EJEMPLO 8.5.9

En la figura 8.5.6, M'' es alcanzable desde M , descargando primero la transición t_1 y luego descargando la transición t_2 . □

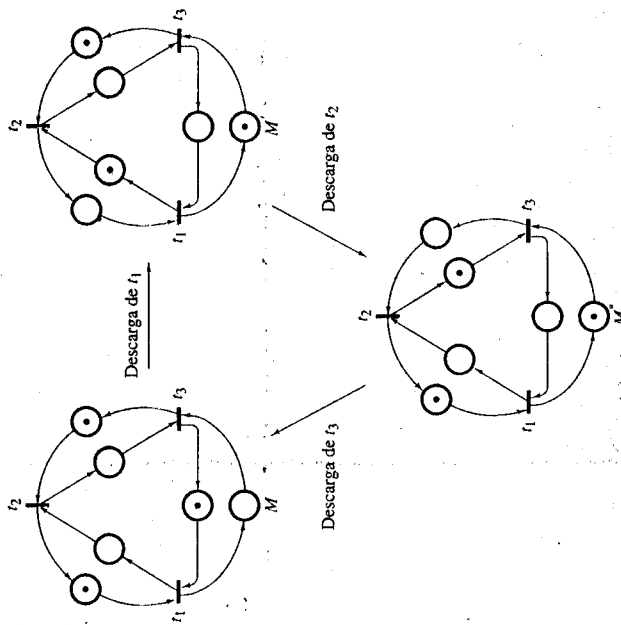


FIGURA 8.5.6 El marcado M'' es alcanzable desde M , descargando primero t_1 y luego t_2 .

Al modelar una situación, la descarga de una transición simula la ocurrencia de ese evento. Por supuesto, un evento puede ocurrir sólo si se cumplen todas las condiciones para su ejecución; es decir, la transición se puede descargar sólo si está activada. Al colocar elementos en los lugares p_1 , p_2 y p_3 de la figura 8.5.4, mostramos que se cumplen las condiciones para ejecutar las instrucciones $A = 1$, $B = 2$ y $C = 3$. El programa está listo para su ejecución. Como las transiciones $A = 1$, $B = 2$ y $C = 3$ están activadas, se pueden descargar en cualquier orden o de manera concurrente. La transición $C = B + C$ sólo se activa si los lugares p_3 y p_4 tienen elementos. Pero estos lugares tendrán elementos sólo si las transiciones $B = 2$ y $C = 3$ se han descargado. En otras palabras, la condición bajo la cual puede ocurrir el evento $C = B + C$ es que $B = 2$ y $C = 3$ hayan sido ejecutadas. De esta forma modelamos las secuencias válidas de ejecución de la figura 8.5.1 y la concurrencia implícita en este programa.

Entre las propiedades más importantes estudiadas en la teoría de las redes de Petri están la *supervivencia* y la *seguridad*. La supervivencia se refiere a la ausencia de estancamientos y la seguridad se relaciona con la capacidad limitada de la memoria.

EJEMPLO 8.5.10

Modelo de red de Petri para un sistema de cómputo compartido

Dos personas comparten un sistema de cómputo que tiene una unidad de disco D y una impresora P . Cada persona necesita D y P . La figura 8.5.7 muestra un posible modelo de red de Petri para esta situación. El marcado indica que D y P están disponibles.

Ahora, suponga que la persona 1 solicita D y luego P (mientras tanto, la persona 2 no las solicita). Las ocurrencias de estos eventos se simulan descargando primero la transición "solicitar D " y descargando después la transición "solicitar P " para la persona 1. La red de Petri resultante aparece en la figura 8.5.8. Cuando la persona 1 termina el procesamiento y libera D y P , lo cual se simula descargando las transiciones "procesar" y luego "liberar D y P ", regresamos a la red de Petri de la figura 8.5.7. Si la persona 2 solicita D y luego P (mientras que la persona 1 no las solicita), obtenemos una secuencia de descarga similar.

De nuevo, suponga que tenemos la situación de la figura 8.5.7. Ahora, suponga que la persona 1 solicita D y que la persona 2 solicita P . Después de descargar las transiciones adecuadas para simular la ocurrencia de estos eventos, obtenemos la red de Petri de la figura 8.5.9. Observe que en este punto, no se pueden descargar las transiciones. La persona 1 está esperando que la persona 2 libere a P , y la persona 2 está esperando que la persona 1 libere D . La actividad dentro del sistema se detiene. Decimos que ha ocurrido un estancamiento.

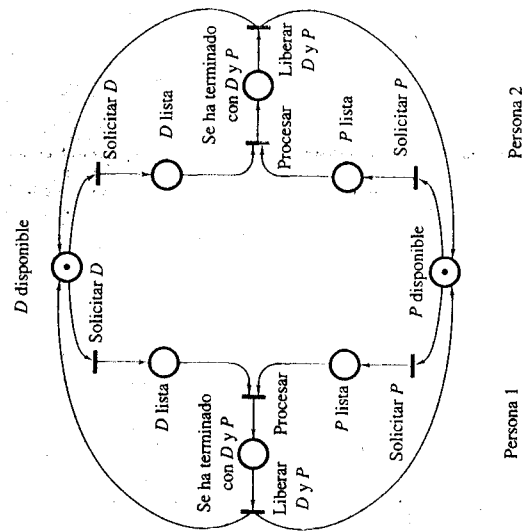


FIGURA 8.5.7 Un modelo de red de Petri para un sistema de cómputo compartido. Cada persona necesita la unidad de disco D y la impresora P . El marcado indica que D y P están disponibles.

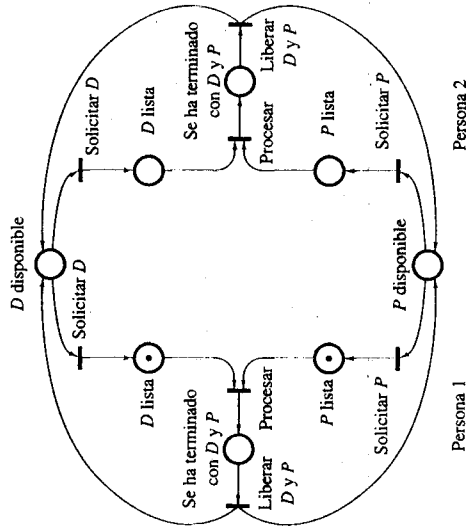


FIGURA 8.5.8 La red de Petri de la figura 8.5.7 después de descargar "solicitar D " y luego "solicitar P " para la persona 1. Después de que la persona 1 termina el procesamiento y libera D y P , lo cual se simula descargando "procesar" y luego "liberar D y P ", obtenemos de nuevo la red de Petri de la figura 8.5.7.

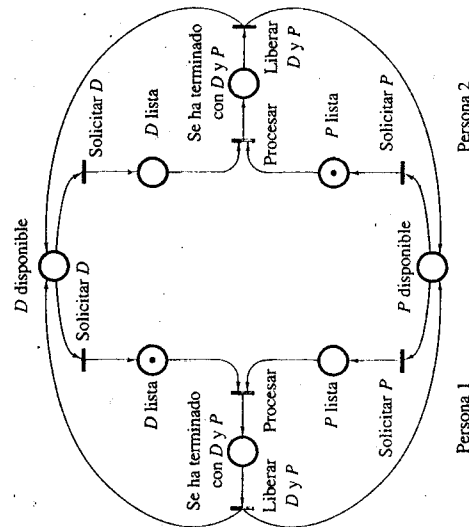


FIGURA 8.5.9 La red de Petri de la figura 8.5.7 después de descargar "solicitar D " para la persona 1 y "solicitar P " para la persona 2. En este punto, la red de Petri está estancada; es decir, ninguna transición se puede descargar.

Formalmente, decimos que una red de Petri marcada está **estancada** si ninguna transición se puede descargar. Una preocupación fundamental en los ambientes de procesamiento concurrente es la forma de evitar los estancamientos. □

El ejemplo 8.5.10 motiva la siguiente definición.

DEFINICIÓN 8.5.11

Un marcado M de una red de Petri está **vivo** si, partiendo de M , sin importar la serie de descargas realizadas, es posible descargar cualquier transición dada mediante alguna secuencia de descargas adicionales.

Si un marcado M está vivo para una red de Petri P , entonces, sin importar la serie de descarga de transiciones, P nunca se estancará. De hecho, podemos descargar cualquier transición mediante cierta secuencia de descargas adicionales.

EJEMPLO 8.5.12

El marcado M de la red de la figura 8.5.6 está vivo. Para ver esto, observe que la única transición del marcado M que se puede descargar es t_1 , la cual produce el marcado M' . La única transición del marcado M' que puede descargarse es t_2 , la cual produce el marcado M'' . La única transición del marcado M'' que se puede descargar es t_3 , la cual nos regresa al marcado M . Así, cualquier secuencia de descarga, que parta del marcado M , produce uno de los marcados M , M' o M'' , y de ahí podemos descargar cualquiera de las transiciones t_1 , t_2 o t_3 , procediendo como en la figura 8.5.6. Por tanto, el marcado M de la red de la figura 8.5.6 está vivo. □

EJEMPLO 8.5.13

El marcado de la figura 8.5.4 no está vivo, pues después de descargar la transición $A = 1$, ya no puede volver a descargarse. □

Si se considera que un lugar tiene capacidad limitada, el hecho de que la red sea acotada nos garantiza que no habrá un desbordamiento de los lugares.

DEFINICIÓN 8.5.14

Un marcado M para una red de Petri está **acotado** si existe algún entero positivo n con la propiedad de que, en cualquier secuencia de descarga, ningún lugar recibe más de n elementos. Si un marcado M está acotado y en cualquier secuencia de descarga ningún lugar recibe más de un elemento, decimos que M es un marcado **seguro**.

Si cada lugar representa un registro capaz de guardar una palabra de computadora y si un marcado inicial es seguro, tenemos garantizado que no se excederá la capacidad de memoria de los registros.

EJEMPLO 8.5.15

Los marcados de la figura 8.5.6 son seguros. El marcado M de la figura 8.5.10 no es seguro pues, como se muestra, si se descarga la transición t_1 , el lugar p_2 entonces tendrá dos elementos. Al enumerar todos los marcados alcanzables desde M , se puede verificar que M está acotada y viva (véase el ejercicio 7).

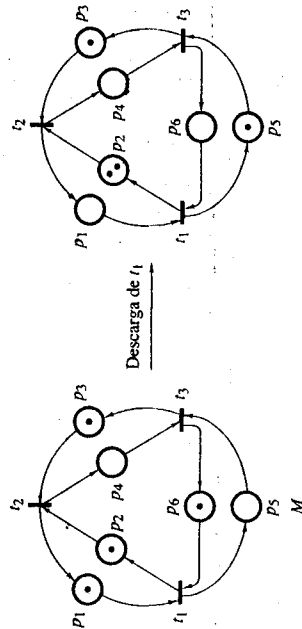


FIGURA 8.5.10 El marcado M no es seguro. Después de descargar t_1 , p_2 tiene dos elementos.

Ejercicios

En los ejercicios 1-3, modele cada programa mediante una red de Petri. Proporcione un marcado que represente la situación anterior a la ejecución del programa.

1. $A = 1$
 $B = 2$
 $C = A + B$
 $C = C + 1$
2. $A = 2$
 $B = A + A$
 $C = 3$
 $D = A + A$
 $C = A + B + C$
3. $A = 1$
 $S = 0$
 $10 \ S = S + A$
 $A = A + 1$
GOTO 10

4. Describa tres situaciones relacionadas con la concurrencia y que puedan modelarse como redes de Petri.

5. Proporcione un ejemplo de una red de Petri marcada en la cual se activen dos transiciones, pero que la descarga de una desactive la otra.

6. Considere el siguiente algoritmo para bañar a un león.

1. Conseguir el león.
2. Conseguir el jabón.
3. Conseguir la tina de baño.
4. Colocar agua en la tina de baño.
5. Colocar al león en la tina de baño.

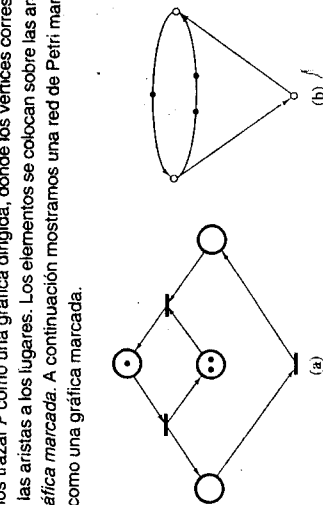
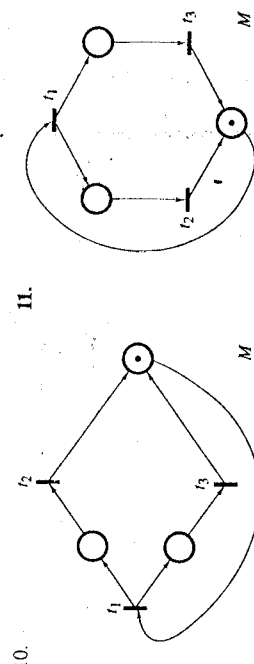
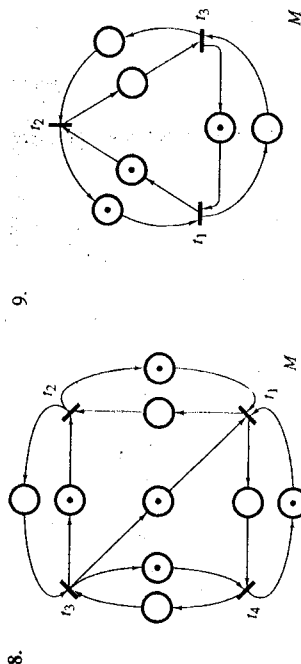
6. Bañar al león con jabón.
7. Enjuagar al león.
8. Sacar al león de la tina de baño.
9. Secar al león.

Modele este algoritmo como una red de Petri. Proporcione un marcado que represente la situación anterior a la ejecución.

7. Muestre que el marcado M de la figura 8.5.10 está vivo y acotado.

Responda las siguientes preguntas para cada una de las redes de Petri marcadas de los ejercicios 8-12.

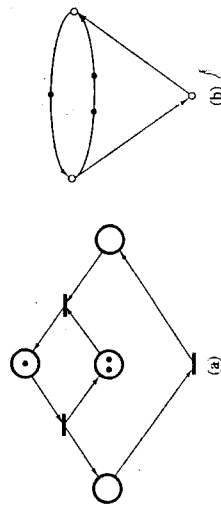
- (a) ¿Cuáles transiciones están activadas?
- (b) Muestre el marcado que resulta de descargar t_1 .
- (c) ¿Está vivo M ?
- (d) ¿Es M seguro?
- (e) ¿Está acotado M ?
- (f) Muestre o describa todos los marcados alcanzables desde M .
- (g) Exhiba un marcado (distinto del marcado que coloca cero elementos en cada lugar) no alcanzable desde M .



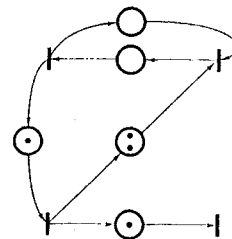
☆ 12.

13. Proporcione un ejemplo de una red de Petri con un marcado seguro, pero no vivo.
14. Proporcione un ejemplo de una red de Petri con un marcado acotado, pero no seguro.
15. El problema de la comida de los filósofos (véase [Dijkstra, 1968]) se relaciona con cinco filósofos sentados ante una mesa redonda. Cada filósofo come o medita. La comida está puesta de manera alternada, con un plato y un palillo para comida china. La comida requiere dos palillos, de modo que si cada filósofo toma el palillo a la derecha del plato, ninguno podrá comer (el sistema se estanca). Modele esta situación como una red de Petri. Su modelo debe estar vivo, de modo que el sistema no se estanque y de modo que, en cada momento, cualquier filósofo tenga la posibilidad de comer o meditar.
16. Desarrolle una red de Petri alternativa a la situación del ejemplo 8.5.10 que evite el estancamiento.

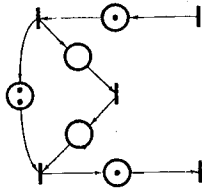
Si cada lugar de una red de Petri marcada P tiene una arista de entrada y una de salida, entonces podemos trazar P como una gráfica dirigida, donde los vértices corresponden a las transiciones y las aristas a los lugares. Los elementos se colocan sobre las aristas. Tal gráfica es una *gráfica marcada*. A continuación mostramos una red de Petri marcada y su representación como una gráfica marcada.



17. ¿Cuáles de las redes de Petri de los ejercicios 8-17 se pueden trazar como gráficas marcadas?
18. Trace la siguiente red de Petri marcada como una gráfica marcada.



19. Trace la siguiente red de Petri marcada como una gráfica marcada.



El número de elementos de un ciclo dirigido simple en una gráfica simple es el número de elementos sobre todas las aristas del ciclo.

20. Muestre que el número de elementos de un ciclo dirigido simple no cambia durante cualquier secuencia de descarga.

- ☆ 21. Muestre que un marcado M para una gráfica marcada G está vivo si y sólo si M coloca al menos un elemento en cada ciclo dirigido simple en G .
 ☆ 22. Muestre que un marcado vivo es seguro para una gráfica marcada G si y sólo si cada arista de G pertenece a un ciclo dirigido simple, con número de elementos igual a 1.
 23. Proporcione un ejemplo de una gráfica marcada con un marcado no vivo, en la cual cada arista pertenece a un ciclo dirigido simple, con número de elementos igual a 1.
 24. Sea G una gráfica marcada. Muestre que cada arista en G está contenida en un ciclo dirigido simple si y sólo si cada marcado para G está acotado.
 ☆ 25. Sea G una gráfica dirigida tal que, si ignoramos la dirección de las aristas en G , G sea conexa como gráfica no dirigida. Muestre que G tiene un marcado vivo y seguro si y sólo si dados cualesquiera dos vértices v y w en G , existe un camino dirigido de v a w .

NOTAS

Las referencias generales con secciones acerca de los modelos de redes son [Berge; Deo; Liu, 1968, 1985; y Tucker]. La obra clásica acerca de las redes es [Ford]; muchos de los resultados de redes, en particular los primeros resultados, se deben a Ford y Fulkerson, los autores de ese libro. [Tarjan] analiza los algoritmos para el flujo en una red y los detalles de su implantación. Las redes de Petri surgieron en la tesis doctoral de C. Petri [Petri] en 1962. Desde entonces ha habido mucha investigación acerca de sus propiedades y un gran interés en su uso para modelar sistemas reales.

El problema de determinar un flujo máximo en una red G , con fuente a , sumidero z y capacidades C_{ij} , se puede parafrasear como sigue:

$$\begin{aligned} &\text{maximizar } \sum_i F_{ai} \\ &\text{sujeta a} \\ &0 \leq F_{ij} \leq C_{ij} \quad \text{para toda } i, j; \\ &\sum_j F_{ij} = \sum_i F_{ji} \quad \text{para toda } i. \end{aligned}$$

Tal problema es un ejemplo de **problema de programación lineal**. En un problema de programación lineal, queremos maximizar (o minimizar) una expresión lineal, como $\sum_i F_{ai}$, sujeta a restricciones lineales en forma de igualdades o desigualdades, como $0 \leq F_{ij} \leq C_{ij}$ y $\sum_j F_{ij} = \sum_i F_{ji}$. Aunque el **algoritmo simplex** es en general una manera eficiente de resolver un problema general de programación lineal, es usual que los problemas de redes de transporte se resuelvan de manera más eficiente mediante el algoritmo 8.2.4. Véase [Hillier] para una exposición del algoritmo simplex.

Suponga que para cada arista (i, j) en una red G , c_{ij} representa el costo de flujo de una unidad a través de la arista (i, j) . Suponga que queremos determinar un flujo máximo, con costo mínimo

$$\sum_i \sum_j c_{ij} F_{ij}.$$

Este problema, llamado el **problema de transporte**, es nuevamente un problema de programación lineal y, como en el caso del problema del flujo máximo, se puede utilizar un algoritmo específico para obtener una solución, de manera más eficiente que el algoritmo simplex (véase [Hillier]).

CONCEPTOS BÁSICOS DEL CAPÍTULO

Sección 8.1

Red (de transporte)

Fuente

Sumidero

Capacidad

Flujo en una red

Flujo en una arista

Flujo de entrada de un vértice

Flujo de salida de un vértice

Conservación del flujo

Dado un flujo F en una red, el flujo de salida de la fuente es igual al flujo de entrada en el sumidero. Este valor común es el valor del flujo F .

Superfuente

Supersumidero

Sección 8.2

Flujo máximo

Arista orientada en forma propia con respecto de un camino

Arista orientada en forma impropia con respecto de un camino

Cómo incrementar el flujo en un camino de la fuente al sumidero cuando:

- (a) para cada arista orientada en forma propia, el flujo es menor que la capacidad y
- (b) cada arista orientada en forma impropia tiene flujo positivo (véase el teorema 8.2.3)

Cómo determinar un flujo máximo en una red (algoritmo 8.2.4)

Sección 8.3

Corte en una red

Capacidad de un corte

La capacidad de cualquier corte es mayor o igual que el valor de cualquier flujo (teorema 8.3.7)

Corte mínimo

Teorema del flujo máximo y el corte mínimo (teorema 8.3.9)

Al concluir el algoritmo del flujo máximo, algoritmo 8.2.4, el conjunto de vértices etiquetados define un corte mínimo

Sección 8.4

Acoplamiento

Acoplamiento máximo

Acoplamiento completo

Red de acoplamiento

Relación entre los flujos y el acoplamiento (teorema 8.4.5)

Teorema de los matrimonios de Hall (teorema 8.4.7)

Sección 8.5

Red de Petri (marcada)

Lugar

Transición

Marcado

Condiciones y eventos

Lugar de entrada

Lugar de salida

Transición activada

Descarga de una transición activada

Marcado alcanzable

Estancamiento

Marcado vivo

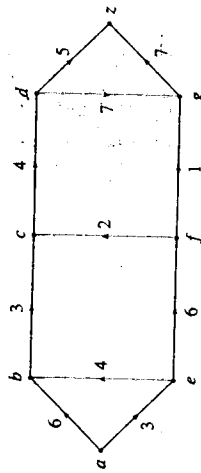
Marcado acotado

Marcado seguro

AUTOEVALUACIÓN DEL CAPÍTULO

Sección 8.1

Los ejercicios 1-4 se refieren a la siguiente red. Las capacidades aparecen sobre las aristas.



1. Explique por qué

$$F_{a,e} = 2, F_{e,b} = 2, F_{b,c} = 3, F_{c,d} = 3, F_{d,z} = 3, F_{a,b} = 1,$$

con las demás $F_{xy} = 0$, es un flujo.

2. ¿Cuál es el flujo de entrada de b ?

3. ¿Cuál es el flujo de salida de c ?

4. ¿Cuál es el valor del flujo F ?

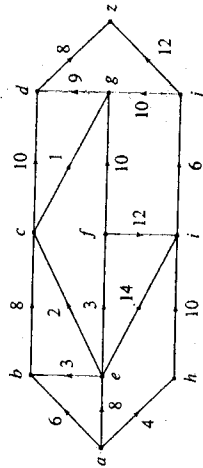
Sección 8.2

5. Para el flujo del ejercicio 1, determine un camino de a a z tal que (a) para cada arista orientada en forma propia, el flujo sea menor que la capacidad y (b) para cada arista orientada en forma impropia tenga flujo positivo.

6. Modifique sólo los flujos en las aristas del camino del ejercicio 5 y determine un flujo con un valor mayor que F .

7. Utilice el algoritmo 8.2.4 para determinar un flujo máximo en la red del ejercicio 1 (comience con el flujo tal que el flujo en cada arista sea igual a cero).

8. Utilice el algoritmo 8.2.4 para determinar un flujo máximo en la siguiente red (comience con el flujo tal que el flujo en cada arista sea igual a cero).



Sección 8.3

9. En cada una de las partes (a)-(d), responda verdadero si la afirmación es verdadera para cualquier red; en caso contrario, responda falso.

(a) Si la capacidad de un corte en una red es igual a C_a , entonces el valor de cualquier flujo es menor o igual a C_a .

(b) Si la capacidad de un corte en una red es igual a C_a , entonces el valor de cualquier flujo es mayor o igual a C_a .

(c) Si la capacidad de un corte en una red es igual a C_a , entonces el valor de algún flujo es mayor o igual a C_a .

(d) Si la capacidad de un corte en una red es igual a C_a , entonces el valor de algún flujo es menor o igual a C_a .

10. Determine la capacidad del corte (P, \bar{P}) en la red del ejercicio 1, donde $P = \{a, b, e, f\}$.

11. ¿Es mínimo el corte (P, \bar{P}) , $P = \{a, b, e, f\}$, en la red del ejercicio 1? Explique.

12. Determine un corte mínimo en la red del ejercicio 8.

Sección 8.4

Los ejercicios 13-16 se refieren a la siguiente situación. El solicitante A está calificado para los trabajos J_1, J_2, J_3, J_4, J_5 ; el solicitante B está calificado para los trabajos J_1, J_2, J_3 ; el solicitante C está calificado para los trabajos J_1, J_3, J_4, J_5 ; y el solicitante D está calificado para los trabajos J_3, J_4, J_5 .

13. Modele esta situación como una red de acoplamiento.

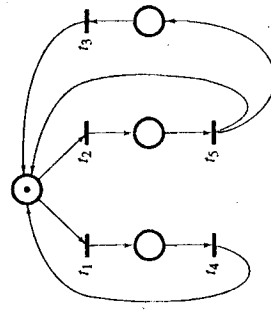
14. Utilice el algoritmo 8.2.4 para determinar un acoplamiento máximo.

15. ¿Existe un acoplamiento completo?

16. Determine un corte mínimo en la red de acoplamiento.

Sección 8.5

Los ejercicios 17-20 se refieren a la siguiente red de Petri.



17. ¿Cuáles transiciones están activadas?

18. Muestre el marcado resultante al descargar t_2 .

19. ¿Es seguro el marcado?

20. ¿Está vivo el marcado?