

Capítulo 10

MODELOS DE REDES†

- 10.1 Introducción
 - 10.2 Algoritmo de flujo máximo
 - 10.3 Teorema de flujo máximo y corte mínimo
 - 10.4 Asignación por pares
- Rincón de solución de problemas: acoplamiento
Notas
Repaso del capítulo
Autoevaluación del capítulo
Ejercicios para computadora

Sólo sigue la corriente, Shel, sólo sigue la corriente.

DE THE IN-LAWS (LOS FOCKER)

En este capítulo se estudian los modelos de redes, que usan gráficas dirigidas. La mayor parte del capítulo se dedica al problema de maximizar el flujo a través de una red. La red puede ser una red de transporte por la que fluyen bienes, una tubería por la que fluye petróleo, una red de computadoras por la que fluyen datos, o cualquier cantidad de posibilidades diferentes. En cada caso el problema es encontrar un flujo máximo. Muchos otros problemas, que en apariencia no son problemas de flujo, de hecho, se pueden modelar como problemas de flujo en una red.

Maximizar el flujo en una red es un problema que pertenece tanto a la teoría de gráficas como a la investigación de operaciones. El problema del agente viajero proporciona otro ejemplo de un problema de teoría de gráficas e investigación de operaciones. La **investigación de operaciones** estudia la amplia categoría de problemas de optimización del desempeño de un sistema. Los problemas típicos estudiados en investigación de operaciones son problemas de redes, de asignación de recursos y de asignación de personal.

10.1 → Introducción

www

Considere la gráfica dirigida de la figura 10.1.1, que representa una tubería de petróleo. El petróleo se descarga en el muelle *a* y se bombea por toda la red de la refinería *z*. Los vértices *b*, *c*, *d* y *e* representan estaciones de bombeo intermedias. Las aristas dirigidas representan subtuberías del sistema y muestran la dirección en que puede fluir el petróleo. Las etiquetas de las aristas indican las capacidades de las subtuberías. El problema es encontrar una manera de maximizar el flujo del muelle a la refinería y calcular el valor de este flujo máximo. La figura 10.1.1 proporciona un ejemplo de una **red de transporte**.

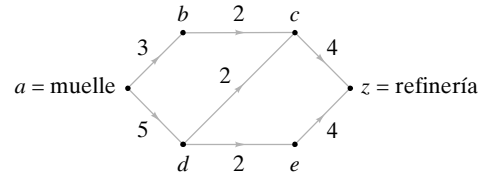


Figura 10.1.1 Una red de transporte.

Definición 10.1.1 ►

Una *red de transporte* (o más sencillo, una *red*) es un gráfica dirigida, ponderada, simple que satisface las siguientes condiciones:

- Un vértice designado, el *origen* o *fuentes*, no tiene aristas entrantes.
- Un vértice designado, *destino* o *sumidero*, no tiene aristas salientes.
- El peso C_{ij} de la arista dirigida (i, j) , llamado *capacidad* de (i, j) , es un número no negativo.

Ejemplo 10.1.2 ►

La gráfica de la figura 10.1.1 es una red de transporte. El origen es el vértice a y el destino es el vértice z . La capacidad de la arista (a, b) , $C_{a,b}$, es 3 y la capacidad de la arista (b, c) , $C_{b,c}$, es 2.

En todo este capítulo, si G es una red, el origen se denotará por a y el destino por z .

Un **flujo** en una red asigna un flujo a cada arista dirigida que no excede la capacidad de esa arista. Más aún, si se supone que el flujo que entra a un vértice v , que no es el origen ni el destino, es igual al flujo que sale de v . La siguiente definición precisa estas ideas.

Definición 10.1.3 ►

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 es la fuente ni el destino,

$$\sum_i F_{ij} = \sum_i F_{ji}. \quad (10.1.1)$$

[En una suma como (10.1.1), a menos que se especifique lo contrario, se supone que la suma se toma sobre todos los vértices i . Además, si (i, j) no es una arista, se hace $F_{ij} = 0$].

F_{ij} recibe el nombre de *flujo en la arista* (i, j) . Para cualquier vértice j ,

$$\sum_i F_{ij}$$

se llama *flujo que entra a j* y

$$\sum_i F_{ji}$$

se llama *flujo que sale de j*.

La propiedad expresada por la ecuación (10.1.1) se llama **conservación del flujo**. En el ejemplo del bombeo de petróleo de la figura 10.1.1, la conservación del flujo significa que el petróleo no se usa ni se suministra en las estaciones de bombeo b, c, d y e .

Ejemplo 10.1.4 ►

Las asignaciones

$$\begin{aligned} F_{ab} = 2, \quad F_{bc} = 2, \quad F_{cz} = 3, \quad F_{ad} = 3, \\ F_{dc} = 1, \quad F_{de} = 2, \quad F_{ez} = 2, \end{aligned}$$

definen un flujo para la red de la figura 10.1.1. Por ejemplo, el flujo que entra al vértice d ,

$$F_{ad} = 3,$$

es el mismo que el flujo que sale del vértice d ,

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

En la figura 10.1.2 se dibujó de nuevo la red de la figura 10.1.1 para mostrar el flujo del ejemplo 10.1.4. Una arista e se etiqueta “ x, y ” si la capacidad de e es x y el flujo en e es y . Esta notación se usará en todo el capítulo.

Advierta que en el ejemplo 10.1.4, el flujo que sale del origen

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

es el mismo que el flujo que entra al destino z ,

$$F_{cz} + F_{ez};$$

ambos valores son 5. El siguiente teorema demuestra que siempre es cierto que el flujo que sale del origen es igual al flujo que entra al destino.

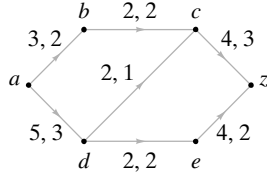


Figura 10.1.2 Flujo en una red. Las aristas tienen etiquetas x, y para indicar la capacidad x y el flujo y .

Teorema 10.1.5

Dado un flujo F en una red, el flujo que sale del origen a es igual al flujo que llega al destino z ; es decir,

$$\sum_i F_{ai} = \sum_i F_{iz}.$$

Demostración Sea V el conjunto de vértices. Se tiene

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

ya que cada doble suma es

$$\sum_{e \in E} F_e,$$

donde E es el conjunto de aristas. Ahora

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

porque $F_{zi} = 0 = F_{ia}$, para toda $i \in V$, y (definición 10.1.3b)

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

A la luz del Teorema 10.1.5, se establece la siguiente definición.

Definición 10.1.6 ▶

Sea F un flujo en una red G . El valor

$$\sum_i F_{ai} = \sum_i F_{iz}$$

se llama el *valor del flujo* F .

Ejemplo 10.1.7 ▶

El valor del flujo en la red de la figura 10.1.2 es 5.

El problema para un red de transporte G se puede establecer como encontrar un flujo máximo en G ; es decir, entre todos los flujos posibles en G , encuentre el flujo F tal que el valor de F sea máximo. En la siguiente sección se da un algoritmo que resuelve con eficiencia este problema. Para terminar la sección se dan ejemplos adicionales.

Ejemplo 10.1.8 ▶**Red de bombeo**

La figura 10.1.3 representa una red de bombeo en la que se entrega agua para dos ciudades, A y B , desde tres pozos w_1 , w_2 y w_3 . Las capacidades de los sistemas intermedios se muestran en las aristas. Los vértices b , c y d representan las estaciones de bombeo intermedias. Modele este sistema como una red de transporte.

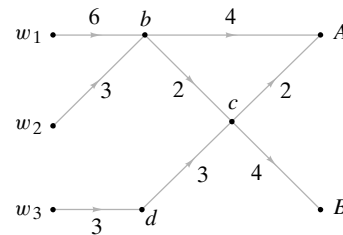


Figura 10.1.3 Red de bombeo.
El agua para las ciudades A y B se entrega desde los pozos w_1 , w_2 y w_3 . Las capacidades se indican en las aristas.

Para obtener el origen y destino designados, se puede obtener una red de transporte equivalente uniendo los orígenes en un **superorigen** y los destinos en un **superdestino** (vea la figura 10.1.4). En ésta, ∞ representa una capacidad ilimitada.

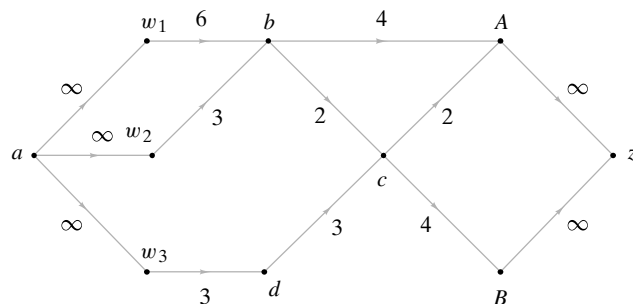


Figura 10.1.4 Red de la figura 10.1.3 con origen y destino designados.

Ejemplo 10.1.9 ►**Una red de flujo de tráfico**

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

A a B	15 minutos
B a C	30 minutos
A a C	30 minutos.

Las capacidades máximas de la rutas son

A a B	3000 vehículos
B a C	2000 vehículos
A a C	4000 vehículos.

Represente el flujo de tráfico de A a C durante el periodo de 6:00 PM a 7:00 PM como una red.

Un vértice representará una ciudad en un tiempo específico (vea la figura 10.1.5). Una arista conecta X, t_1 con Y, t_2 si se puede salir de la ciudad X a las t_1 PM y llegar a la ciudad Y a las t_2 PM. La capacidad de una arista es la capacidad de la ruta. Las aristas de capacidad infinita conectan a A, t_1 con A, t_2 y B, t_1 con B, t_2 para indicar que cualquier número de autos puede esperar en las ciudades A o B. Por último, se introduce un superorigen y un superdestino.

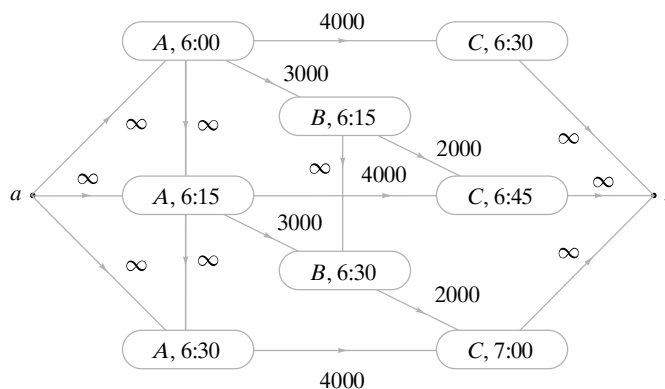


Figura 10.1.5 Red que representa el flujo de tráfico de la ciudad A a la ciudad C durante el periodo de 6:00 PM a 7:00 PM.

Se han usado algunas variaciones del problema de flujo en redes en el diseño de redes de computadoras eficientes (vea [Jones; Kleinrock]). En un modelo de una red de computadoras, un vértice es un mensaje o un centro de conmutación, una arista representa un canal por el que se pueden transmitir datos entre 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.

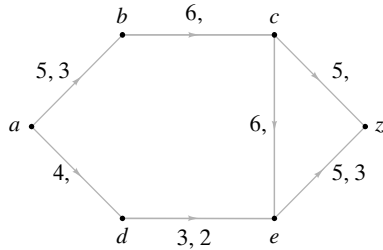
Sección de ejercicios de repaso

1. ¿Qué es una red?
2. ¿Qué es la fuente o el origen en una red?
3. ¿Qué es un sumidero o destino en una red?
4. ¿Qué es una capacidad en una red?
5. ¿Qué es un flujo en una red?
6. ¿Qué es un flujo en una arista?
7. ¿Qué es un flujo en un vértice?
8. ¿Qué es un flujo que sale de un vértice?
9. ¿Qué es conservación del flujo?
10. Dado un flujo en una red, ¿cuál es la relación entre el flujo que sale del origen y el que llega al destino?
11. ¿Qué es un superorigen?
12. ¿Qué es un superdestino?

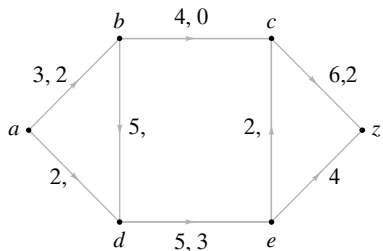
Ejercicios

En los ejercicios 1 al 3, encuentre los flujos en las aristas que faltan de manera que el resultado sea un flujo en la red dada. Determine los valores de los flujos.

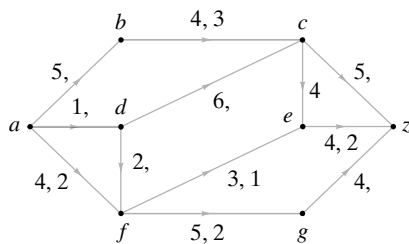
1.



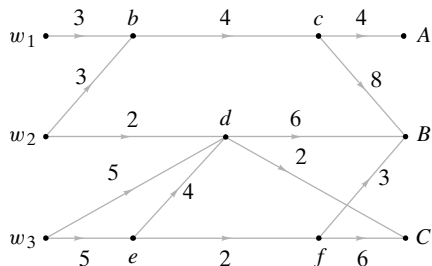
2.



3.



4. La siguiente gráfica representa una red de bombeo por la que se entrega petróleo a tres refinéras, A, B y C, desde tres pozos, w_1 , w_2 y w_3 . Las capacidades de los sistemas intermedios se indican en las aristas. Los vértices b, c, d, e y f representan las 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 cuando mucho 2 unidades, el pozo w_2 , cuando mucho 4 unidades y el pozo w_3 cuando mucho 7 unidades.
6. Modele el sistema del ejercicio 5 como una red suponiendo, además de las restricciones sobre los pozos, que la ciudad A requiere 4 unidades, la ciudad B requiere 3 unidades y la ciudad C, 4 unidades.
7. Modele el sistema del ejercicio 6 como una red suponiendo, además de las restricciones sobre los pozos y los requerimientos de las ciudades, que la estación de bombeo intermedia d puede bombear cuando mucho 6 unidades.
8. Hay dos rutas de la ciudad A a la ciudad D. Una ruta pasa por la ciudad B y la otra ruta pasa por la ciudad C. Durante el periodo de 7:00 AM a 8:00 AM, los tiempos de viaje promedio son

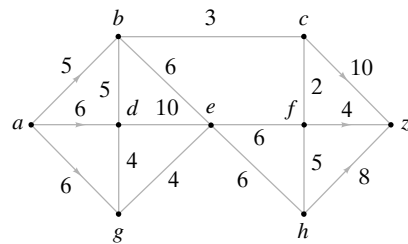
A a B	30 minutos
A a C	15 minutos
B a D	15 minutos
C a D	15 minutos.

Las capacidades máximas de las rutas son

A a B	1000 vehículos
A a C	3000 vehículos
B a D	4000 vehículos
C a D	2000 vehículos.

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

9. En el sistema mostrado, se desea maximizar el flujo de a a z. Las capacidades se indican en las aristas. El flujo entre dos vértices, ninguno de los cuales es a o z, puede ser en cualquier dirección. Modele este sistema como una red.



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

10.2 → Algoritmo de flujo máximo

Si G es una red de transporte, un **flujo máximo** en G es un flujo con valor máximo. En general, habrá varios flujos que tienen el mismo valor máximo. En esta sección se presenta un algoritmo para encontrar un flujo máximo. La idea básica es sencilla: comenzar con un flujo inicial y aumentar, de manera iterativa, el valor del flujo hasta que no sea posible mejorarlo. El flujo obtenido será entonces el flujo máximo.

Se puede tomar el flujo inicial como uno en el que el flujo en cada arista es cero. Para aumentar el valor de un flujo dado, debe encontrarse una trayectoria del origen al destino y aumentar el flujo a lo largo de esta trayectoria.

En este momento resulta útil introducir cierta terminología. En esta sección, G denota una red con origen a , destino z y capacidad C . Por el momento, considere las aristas de G como no dirigidas y sea

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

una trayectoria de a a z en esta gráfica no dirigida. (Todas las trayectorias en esta sección se refieren a la gráfica no dirigida). Si una arista e en P es dirigida de v_{i-1} a v_i , se dice que e tiene la **orientación apropiada (respecto a P)**; de otra manera, se dice que e tiene la **orientación inapropiada (respecto a P)** (vea la figura 10.2.1).

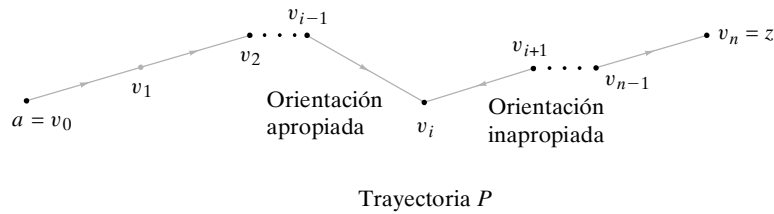


Figura 10.2.1 Aristas con orientación apropiada e inapropiada. La arista (v_{i-1}, v_i) tiene orientación apropiada porque está orientada en la dirección de a a z . La arista (v_i, v_{i+1}) tiene orientación inapropiada porque *no* tiene la dirección de a a z .

Si se puede encontrar una trayectoria P del origen al destino en la que todas las aristas tengan la orientación apropiada y el flujo de cada una sea menor que la capacidad de la arista, es posible aumentar el valor del flujo.

Ejemplo 10.2.1 ►

Considere la trayectoria de a a z en la figura 10.2.2. Todas las aristas en P tienen la orientación adecuada. El valor del flujo en esta red se puede aumentar en 1, como se aprecia en la figura 10.2.3.

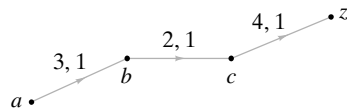


Figura 10.2.2 Una trayectoria cuyas aristas tienen la orientación apropiada.

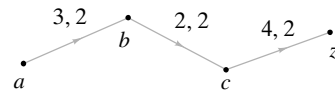


Figura 10.2.3 Después de aumentar en 1 el flujo de la figura 10.2.2.

También es posible aumentar el flujo en ciertas trayectorias del origen al destino, en las que se tienen aristas con orientación apropiada e inapropiada. Sea P una trayectoria de a a z y sea x un vértice en P diferente de a y z (vea la figura 10.2.4). Existen cuatro posibilidades para la orientación de las aristas e_1 y e_2 incidentes en x . En el caso *a*), ambas aristas tienen la orientación correcta. En este caso, si aumentamos en Δ el flujo en cada arista, el flujo que entra a x todavía será igual al flujo que sale de x . En el caso *b*), si se incrementa

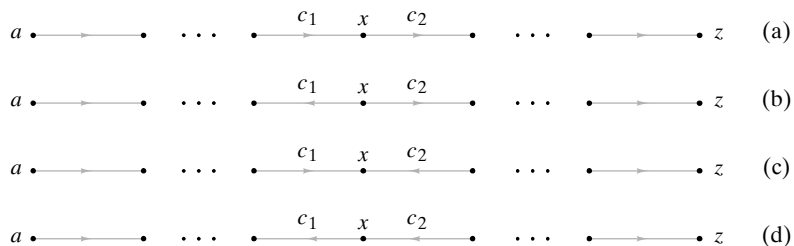


Figura 10.2.4 Las cuatro orientaciones posibles de las aristas incidentes en x .

en Δ el flujo de e_2 , debe *disminuirse* en Δ el flujo en e_1 para que el flujo que llega a x siga siendo igual al que sale. El caso *c*) es similar al caso *b*) excepto que se aumenta en Δ el flujo en e_1 y se disminuye en Δ el flujo en e_2 . En el caso *d*), se disminuye en Δ el flujo en ambas aristas. En todos los casos, las asignaciones de aristas que se obtienen dan un flujo. Por supuesto que para llevar a cabo estas modificaciones debemos tener un flujo menor que la capacidad en un arista con la orientación apropiada y un flujo diferente de cero en la arista con la orientación inapropiada.

Ejemplo 10.2.2 ►

Considere la trayectoria de a a z en la figura 10.2.5. Las aristas (a, b) , (c, d) y (d, z) tienen la orientación apropiada y la arista (c, b) tiene la orientación inapropiada. Se disminuye en 1 el flujo de la arista con orientación inapropiada (c, b) y se aumenta en 1 el flujo de las aristas orientadas apropiadamente (a, b) , (c, d) y (d, z) (vea la figura 10.2.6). El valor del nuevo flujo es 1 unidad mayor que el original.

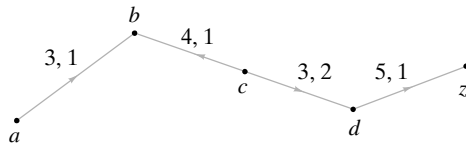


Figura 10.2.5 Una trayectoria con una arista orientada inapropiadamente: (c, b) .

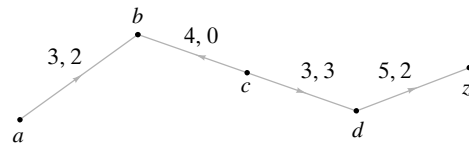


Figura 10.2.6 Después de aumentar en 1 unidad el flujo de la figura 10.2.5.

WWW

Se resume el método de los ejemplos 10.2.1 y 10.2.2 como un teorema.

Teorema 10.2.3

Sea P una trayectoria del vértice a al z en una red G que satisface las siguientes condiciones:

a) Para cada arista (i, j) con orientación apropiada en P ,

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

b) Para cada arista (i, j) con orientación inadecuada en P ,

$$0 < F_{ij}.$$

Sea

$$\Delta = \text{mín } X,$$

donde X consiste en los números $C_{ij} - F_{ij}$ para aristas (i, j) con orientación apropiada en P , y F_{ij} para aristas (i, j) con orientación inapropiada en P . Defina

$$F_{ij}^* = \begin{cases} F_{i,j} & \text{si } (i, j) \text{ no está en } P \\ F_{ij} + \Delta & \text{si } (i, j) \text{ tiene orientación apropiada en } P \\ F_{ij} - \Delta & \text{si } (i, j) \text{ no tiene la orientación apropiada en } P. \end{cases}$$

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

Demostración (Vea las figuras 10.2.2, 10.2.3, 10.2.5 y 10.2.6). El argumento que asegura que F^* es un flujo se presentó justo antes del ejemplo 10.2.2. Como la arista (a, v) en P se incrementa Δ , el valor de F^* es Δ unidades mayor que el valor de F .

En la siguiente sección se demuestra que si no hay trayectorias que satisfagan las condiciones del Teorema 10.2.3, el flujo es máximo. Entonces es posible construir un algoritmo basado en el Teorema 10.2.3. La descripción es la siguiente:

1. Se comienza con un flujo (por ejemplo, uno en el que el flujo de cada arista es 0).
2. Se busca una trayectoria que satisfaga las condiciones del Teorema 10.2.3. Si no existe, hay que detenerse; el flujo es máximo.
3. Se incrementa el flujo en toda la trayectoria por Δ unidades, donde Δ se define como en el Teorema 10.2.3, y se va a la línea 2.

En el algoritmo formal, se busca una trayectoria que satisfaga las condiciones del Teorema 10.2.3 al mismo tiempo que se van registrando las cantidades

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

Algoritmo 10.2.4

Encontrar un flujo máximo en una red

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

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

Salida: Un flujo máximo F

```

flujo_máx( $a, z, C, v, n$ ) {
    //etiqueta de  $v$  es ( $predecesor(v), val(v)$ )
    //inicia con flujo cero
1.   para cada arista  $(i, j)$ 
2.        $F_{ij} = 0$ 
3.   while(verdadero) {
4.       //eliminar todas las etiquetas
5.       for  $i = 0$  to  $n$  {
6.            $predecesor(v_i) = nulo$ 
7.            $val(v_i) = nulo$ 
8.       }
9.       //etiquetar  $a$ 
10.       $predecesor(a) = -$ 
11.       $val(a) = \infty$ 
12.      //U es conjunto de vértices etiquetados no examinados
13.       $U = \{a\}$ 
14.      //continuar hasta etiquetar  $z$ 
15.      while ( $val(z) == nulo$ ) {
16.          if ( $U == \emptyset$ ) //flujo es máximo
17.              return  $F$ 
18.          elegir  $v$  en  $U$ 
19.           $U = U - \{v\}$ 
20.           $\Delta = val(v)$ 
21.          para cada arista  $(v, w)$  con  $val(w) == nulo$ 
22.              if ( $F_{vw} < C_{vw}$ ) {
23.                   $predecesor(w) = v$ 
24.                   $val(w) = \min\{\Delta, C_{vw} - F_{vw}\}$ 
25.                   $U = U \cup \{w\}$ 
26.              }
27.          para cada arista  $(w, v)$  con  $val(w) == nulo$ 
28.              if ( $F_{wv} > 0$ ) {
29.                   $predecesor(w) = v$ 
30.                   $val(w) = \min\{\Delta, F_{wv}\}$ 
31.                   $U = U \cup \{w\}$ 
32.              }
33.      } //fin del ciclo while ( $val(z) == nulo$ )
34.      //encontrar trayectoria  $P$  de  $a$  a  $z$  para revisar flujo
35.       $w_0 = z$ 
36.       $k = 0$ 
37.      while ( $w_k \neq a$ ) {
38.           $w_{k+1} = predecesor(w_k)$ 
39.           $k = k + 1$ 
40.      }

```

```

36.       $P = (w_{k+1}, w_k, \dots, w_0)$ 
37.       $\Delta = \text{val}(z)$ 
38.      for  $i = 1$  to  $k + 1$  {
39.           $e = (w_i, w_{i-1})$ 
40.          if ( $e$  tiene orientación correcta en  $P$ )
41.               $F_e = F_e + \Delta$ 
42.          else
43.               $F_e = F_e - \Delta$ 
44.      }
45.  } //fin del ciclo while(verdadero)

```

Una demostración de que el algoritmo 10.2.4 termina, se deja como ejercicio 19. Si se permite que las capacidades sean números racionales no negativos, el algoritmo también termina; sin embargo, si se permiten números reales no negativos para las capacidades y se permite que las aristas en la línea 17 se examinen en cualquier orden, el algoritmo puede no terminar (vea [Ford, pp. 21–22]).

Con frecuencia nos referimos al algoritmo 10.2.4 como **procedimiento de etiquetado**. Se ilustrará el algoritmo con dos ejemplos.

Ejemplo 10.2.5 ►

En este análisis, si el vértice v satisface

$$\text{predecesor}(v) = p \text{ y } \text{val}(v) = t,$$

la etiqueta de v en la gráfica se indica como (p, t) .

En las líneas 1 y 2, se inicializa el flujo en 0 en cada arista (vea la figura 10.2.7). Después, en las líneas 4 a la 7, las etiquetas se establecen como *nulo*. En las líneas 8 y 9 el vértice a se etiqueta como $(-, \infty)$. En la línea 10 se hace $U = \{a\}$. Después se llega al ciclo “while” (línea 11).

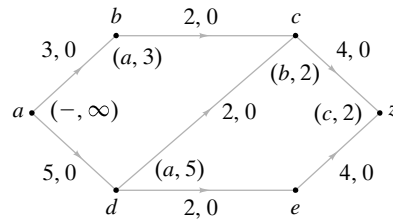


Figura 10.2.7 Después de la primera etiquetada. El vértice v se etiqueta $(\text{predecesor}(v), \text{val}(v))$.

Como z no está etiquetado y U no está vacío, se pasa a la línea 14 donde se elige el vértice a en U y se elimina de U en la línea 15. En este punto, $U = \emptyset$. Se establece que Δ es igual a $\infty [= \text{val}(a)]$ en la línea 16. En la línea 17 se examinan las aristas (a, b) y (a, d) puesto que b y d no están etiquetadas. Para la arista (a, b) se tiene

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

En las líneas 19 y 20, se etiqueta el vértice b como $(a, 3)$ ya que

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

y

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

En la línea 21, se agrega b a U . De manera similar, se etiqueta el vértice d como $(a, 5)$ y se agrega a U . En este punto, $U = \{b, d\}$.

Después se regresa al principio del ciclo “while” (línea 11). Como z no tiene etiqueta y U no está vacío, se pasa a la línea 14, donde se elige un vértice en U . Suponga que se elige b . Se elimina b de U en la línea 15. Se hace Δ igual a 3 [= $val(b)$] en la línea 16. En la línea 17 se examina la arista (b, c) . En las líneas 19 y 20 se etiqueta el vértice c como $(b, 2)$ puesto que

$$predecesor(c) = b$$

y

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

En la línea 21 se agrega c a U . En este punto, $U = \{c, d\}$.

Después se regresa al principio del ciclo “while” (línea 11). Como z no tiene etiqueta y U no está vacío, se pasa a la línea 14, donde se elige un vértice en U . Suponga que se elige c . Se elimina c de U en la línea 15. Se hace Δ igual a 2 [= $val(c)$] en la línea 16. En la línea 17 se examina la arista (c, z) . En las líneas 19 y 20 se etiqueta el vértice z como $(c, 2)$. En la línea 21 se agrega z a U . En este punto, $U = \{d, z\}$.

Después se regresa al principio del ciclo “while” (línea 11). Como z está etiquetada, se procede a la línea 30. En las líneas 30 a la 36, siguiendo los predecesores desde z , se encuentra la trayectoria

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

de a a z . En la línea 37 se hace Δ igual a 2. Como todas las aristas en P tienen la orientación apropiada, en la línea 41 se aumenta el flujo de cada arista en P en $\Delta = 2$ para obtener la figura 10.2.8.

Después se regresa al principio del ciclo “while” (línea 3). En las líneas 4 a la 7 se hacen todas las etiquetas igual a *nulo*. Luego, en las líneas 8 y 9 se etiqueta el vértice a como $(-, \infty)$ (vea la figura 10.2.8). En la línea 10 se hace $U = \{a\}$. Después se llega al ciclo “while” (línea 11).

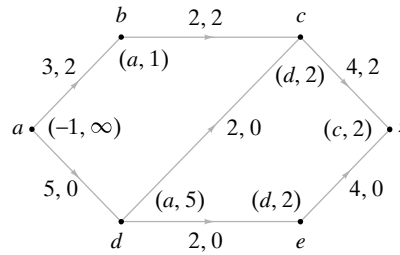


Figura 10.2.8 Después de aumentar el flujo en la trayectoria (a, b, c, z) en 2 unidades y el segundo etiquetado.

Como z no está etiquetado y U es no vacío, se pasa a la línea 14, donde se elige el vértice a en U y se elimina de U en la línea 15. En las líneas 19 y 20 se etiqueta el vértice b como $(a, 1)$ y se etiqueta el vértice d como $(a, 5)$. Se agregan b y d a U de modo que $U = \{b, d\}$.

Después se regresa al principio del ciclo “while” (línea 11). Como z no está etiquetado y U es no vacío, se pasa a la línea 14, donde se elige un vértice en U . Suponga que se selecciona b . Se elimina b de U en la línea 15. En la línea 17 se examina la arista (b, c) . Como $F_{bc} = C_{bc}$, no se etiqueta el vértice c en este punto. Ahora $U = \{d\}$.

Entonces se regresa al principio del ciclo “while” (línea 11). Como z no tiene etiqueta y U es no vacío, se pasa a la línea 14, donde se elige el vértice d en U y se elimina de U en la línea 15. En las líneas 19 y 20 se etiqueta el vértice c como $(d, 2)$ y se etiqueta el vértice e como $(d, 2)$. Se agregan c y e a U de modo que $U = \{c, e\}$.

De nuevo se regresa al principio de ciclo “while” (línea 11). Como z no está etiquetado y U es no vacío, se pasa a la línea 14, donde se elige un vértice en U . Suponga que se selecciona c y se elimina de U en la línea 15. En las líneas 19 y 20 se etiqueta el vértice z como $(c, 2)$. Se agrega z a U de manera que $U = \{z, e\}$.

Se regresa al principio del ciclo “while” (línea 11). Como z está etiquetado, se procede a la línea 30. En la línea 36 se encuentra que

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

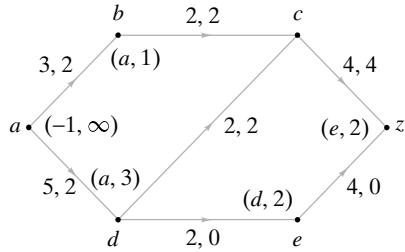


Figura 10.2.9 Después de aumentar el flujo en la trayectoria (a, d, c, z) en 2 unidades y el tercer etiquetado.

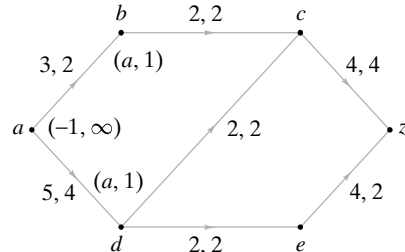


Figura 10.2.10 Después de aumentar el flujo en la trayectoria (a, d, e, z) en 2 unidades y el cuarto etiquetado. El flujo es máximo.

Como todas las aristas en P tienen la orientación adecuada, en la línea 41 se aumenta el flujo en cada una en $\Delta = 2$ para obtener la figura 10.2.9.

Debe verificarse que la siguiente iteración del algoritmo produzca las etiquetas mostradas en la figura 10.2.9. Aumentar el flujo en $\Delta = 2$ produce la figura 10.2.10.

Ahora se regresa al principio del ciclo “while” (línea 3). Después, en las líneas 4 a la 7 se hacen todas las etiquetas iguales a *nulo*. Luego, en las líneas 8 y 9 se etiqueta el vértice a como $(-, \infty)$ (vea la figura 10.2.10). En la línea 10 se hace $U = \{a\}$. Después se llega al ciclo “while” (línea 11).

Como z no tiene etiqueta y U es no vacío, se pasa a la línea 14, donde se elige el vértice a en U y se elimina de U en la línea 15. En las líneas 19 y 20 se etiqueta el vértice b como $(a, 1)$ y se etiqueta el vértice d como $(a, 1)$. Se agregan b y d a U de manera que $U = \{b, d\}$.

Se regresa al principio del ciclo “while” (línea 11). Como z no tiene etiqueta y U es no vacío, se pasa a la línea 14, donde se elige un vértice en U . Suponga que se elige b . Se elimina b de U en la línea 15. En la línea 17, se examina la arista (b, c) . Como $F_{bc} = C_{bc}$, no se etiqueta el vértice c . Ahora $U = \{d\}$.

Se regresa al principio del ciclo “while” (línea 11). Como z no está etiquetado y U es no vacío, se pasa a la línea 14, donde se elige el vértice d en U y se elimina de U en la línea 15. En la línea 17 se examinan las aristas (d, c) y (d, e) . Como $F_{dc} = C_{dc}$ y $F_{de} = C_{de}$, no se etiqueta el vértice c ni el e . Ahora $U = \emptyset$.

Se regresa al principio del ciclo “while” (línea 11). Como z no tiene etiqueta, se pasa a la línea 12. Como U es vacío, el algoritmo termina. El flujo de la figura 10.2.10 es máximo. ◀

El último ejemplo muestra cómo modificar el algoritmo 10.2.4 para generar un flujo máximo a partir de un flujo dado.

Ejemplo 10.2.6 ▶

Sustituya el flujo cero en las líneas 1 y 2 del algoritmo 10.2.4 con el flujo de la figura 10.2.11 y luego encuentre un flujo máximo.

Después de inicializar el flujo dado, se pasa a las líneas 4 a la 7, donde se establecen todas las etiquetas en *nulo*. Después, en las líneas 8 y 9 se etiqueta el vértice a como $(-, \infty)$ (vea la figura 10.2.11). En la línea 10 se hace $U = \{a\}$. Después se llega al ciclo “while” (línea 11).

Como z no tiene etiqueta y U es no vacío, se pasa a la línea 14, donde se elige el vértice a en U y se elimina de U en la línea 15. En las líneas 19 y 20, se etiqueta el vértice b como $(a, 1)$ y se etiqueta el vértice d como $(a, 1)$. Se agregan b y d a U de modo que $U = \{b, d\}$.

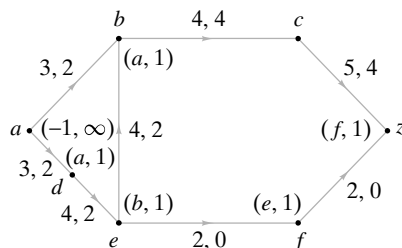


Figura 10.2.11 Después de etiquetar.

Después se regresa al principio del ciclo “while” (línea 11). Como z no tiene etiqueta y U es no vacío, se pasa a la línea 14, donde se elige un vértice de U . Suponga que se elige b . Se elimina b de U en la línea 15. En la línea 17 se examinan las aristas (b, c) y (e, b) . Como $F_{bc} = C_{bc}$, no se etiqueta el vértice c . En las líneas 25 y 26 se etiqueta el vértice e como $(b, 1)$ ya que

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

Se regresa entonces al principio del ciclo “while” (línea 11). Al final se etiqueta z (vea la figura 10.2.11) y en la línea 36 se encuentra la trayectoria

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

Las aristas (a, b) , (e, f) y (f, z) tienen la orientación apropiada, por lo que el flujo en cada una se aumenta en 1. Como la arista (e, b) tiene la orientación inapropiada, su flujo se disminuye en 1. Se obtiene el flujo de la figura 10.2.12.

Otra iteración del algoritmo produce el flujo máximo mostrado en la figura 10.2.13.

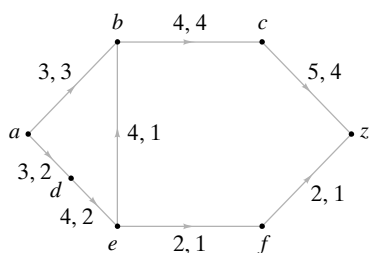


Figura 10.2.12 Después de aumentar en 1 el flujo en la trayectoria (a, b, e, f, z) . Observe que la arista (e, b) está orientada inapropiadamente por lo que su flujo *disminuye* en 1.

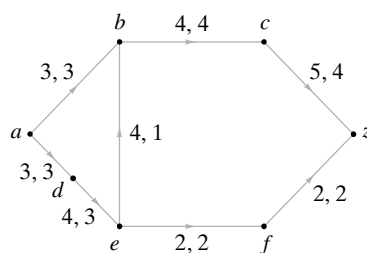


Figura 10.2.13 Después de aumentar en 1 el flujo en la trayectoria (a, d, e, f, z) . El flujo es máximo.

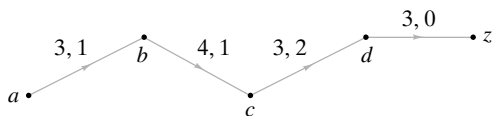
Sección de ejercicios de repaso

1. ¿Qué es un flujo máximo?
2. ¿Qué es una arista con orientación apropiada respecto a una trayectoria?
3. ¿Qué es una arista con orientación inapropiada respecto a una trayectoria?
4. ¿Cuándo se puede incrementar el flujo en una trayectoria del origen al destino?
5. Explique cómo se incrementa el flujo en las condiciones del ejercicio 4.
6. Explique cómo encontrar un flujo máximo en una red.

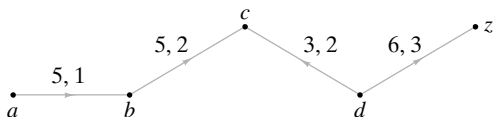
Ejercicios

En los ejercicios 1 al 3 se da una trayectoria del origen a al destino z en una red. Encuentre el incremento máximo posible en el flujo que se puede obtener modificando los flujos en las aristas de la trayectoria.

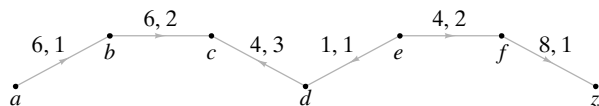
1.



2.



3.

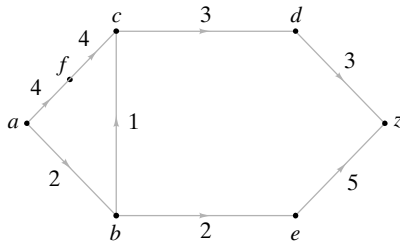


En los ejercicios 4 al 12, utilice el algoritmo 10.2.4 para encontrar un flujo máximo en cada red.

4. Figura 10.1.4

5. Figura 10.1.5

6.



7. Ejercicio 5, sección 10.1
 8. Ejercicio 6, sección 10.1
 9. Ejercicio 7, sección 10.1
 10. Ejercicio 8, sección 10.1
 11. Ejercicio 9, sección 10.1
 12.

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

13. Figura 10.1.2

14. Ejercicio 1, sección 10.1

15. Ejercicio 2, sección 10.1

16. Ejercicio 3, sección 10.1

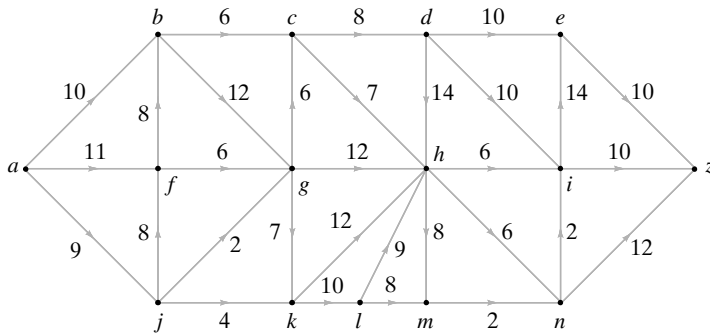
17. Figura 10.1.4 con los flujos

$$\begin{array}{llll} F_{a,w_1} = 2, & F_{w_1,b} = 2, & F_{bA} = 0, & F_{cA} = 0, \\ F_{Az} = 0, & F_{a,w_2} = 0, & F_{w_2,b} = 0, & F_{bc} = 2, \\ F_{cB} = 4, & F_{Bz} = 4, & F_{a,w_3} = 2, & F_{w_3,d} = 2, \\ F_{dc} = 2. & & & \end{array}$$

18. Figura 10.2.4 con los flujos

$$\begin{array}{lll} F_{a,w_1} = 1, & F_{w_1,b} = 1, & F_{bA} = 4, \\ F_{cA} = 2, & F_{Az} = 6, & F_{a,w_2} = 3, \\ F_{w_2,b} = 3, & F_{bc} = 0, & F_{cB} = 1, \\ F_{Bz} = 1, & F_{a,w_3} = 3, & F_{w_3,d} = 3, \\ F_{dc} = 3. & & \end{array}$$

19. Demuestre que el algoritmo 10.2.4 termina.



10.3 → Teorema de flujo máximo y corte mínimo

En esta sección se muestra que a la terminación del algoritmo 10.2.4, el flujo en la red es máximo. Al mismo tiempo se definirán y analizarán los cortes en las redes.

Sea G una red y considere el flujo F a la terminación del algoritmo 10.2.4. 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 el origen a está en P y el destino z está en \bar{P} . El conjunto S de aristas (v, w) , con $w \in P$ y $v \in \bar{P}$, se llama un **corte**, y la suma de las capacidades de las aristas en S se llama **capacidad del corte**. Se verá que este corte tiene capacidad mínima y, como un corte mínimo corresponde a un flujo máximo (Teorema 10.3.9), el flujo F es máximo. Se comienza con la definición formal de corte.

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

Definición 10.3.1 ►

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

Ejemplo 10.3.2 ►

Considere la red G de la figura 10.3.1. Si se hace $P = \{a, b, d\}$, entonces $\bar{P} = \{c, e, f, z\}$ y (P, \bar{P}) es un corte en G . Como se muestra, algunas veces se indica un corte dibujando una línea punteada para dividir los vértices.

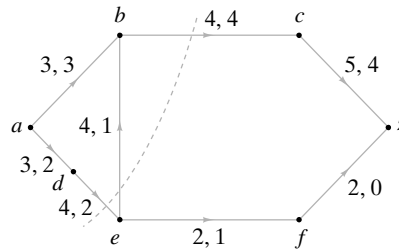


Figura 10.3.1 Un corte en una red. La línea punteada divide los vértices en dos conjuntos $P = \{a, b, d\}$ y $\bar{P} = \{c, e, f, z\}$, lo que produce el corte (P, \bar{P}) .

Ejemplo 10.3.3 ►

La figura 10.2.10 muestra las etiquetas al terminar el algoritmo 10.2.4 para una red en particular. Si P (\bar{P}) denota el conjunto de vértices etiquetados (no etiquetados), se obtiene el corte mostrado en la figura 10.3.2.

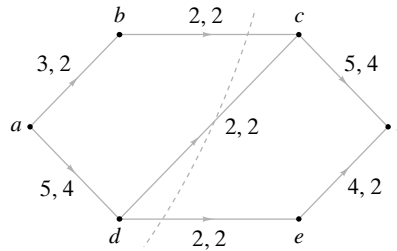


Figura 10.3.2 Una red a la terminación del algoritmo 10.2.4. El corte (P, \bar{P}) , $P = \{a, b, d\}$, se obtiene cuando P es el conjunto de vértices etiquetados.

Ahora se definirá la capacidad de un corte.

Definición 10.3.4 ►

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 10.3.5 ►

La capacidad del corte de la figura 10.3.1 es

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

Ejemplo 10.3.6 ►

La capacidad del corte de la figura 10.3.2 es

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

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

Teorema 10.3.7

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

$$\sum_{i \in P} \sum_{j \in \bar{P}} C_{ij} \geq \sum_i F_{ai}. \quad (10.3.1)$$

(La notación \sum_i significa la suma sobre todos los vértices i).

Demostración Observe que

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

ya que cualquiera de los lados de la ecuación es simplemente la suma de F_{ij} sobre toda $i, j \in P$.

Ahora

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

Ejemplo 10.3.8 ►

En la figura 10.3.1, el valor 5 del flujo es menor que la capacidad de 8 del corte. ◀

Un **corte mínimo** es un corte que tiene capacidad mínima.

Teorema 10.3.9

Flujo máximo, corte mínimo

Sea F un flujo en G y sea (P, \bar{P}) un corte en G . Si se cumple la igualdad en (10.3.1), entonces el flujo es máximo y el corte es mínimo. Más aún, la igualdad se cumple en (10.3.1) si y sólo si

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

y

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

Demostración La primera afirmación se deduce directamente.

La prueba del Teorema 10.3.7 indica que la igualdad se cumple precisamente cuando

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

entonces la última afirmación también se cumple.

Ejemplo 10.3.10 ►

En la figura 10.3.2, el valor del flujo y la capacidad del corte son 6 los dos; por lo tanto, el flujo es máximo y el corte es mínimo. ◀

El Teorema 10.3.9 resulta útil para demostrar que el algoritmo 10.2.4 produce un flujo máximo.

Teorema 10.3.11

Al terminar, el algoritmo 10.2.4 produce un flujo máximo. Todavía más, si P (respectivamente, \bar{P}) es el conjunto de vértices etiquetados (respectivamente, no etiquetados) al terminar el algoritmo 10.2.4, el corte (P, \bar{P}) es mínimo.

Demostración Sea P (\bar{P}) el conjunto de vértices etiquetados (no etiquetados) de G a la terminación del algoritmo 10.2.4. Considere una arista (i, j) , donde $i \in P, j \in \bar{P}$. Como i está etiquetado, debe tenerse

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

de otra manera, se habría etiquetado j en las líneas 19 y 20. Ahora considere una arista (j, i) , donde $j \in \bar{P}, i \in P$. Como i está etiquetado, debe tenerse

$$F_{ji} = 0;$$

de otra manera se habría etiquetado j en las líneas 25 y 26. Por el Teorema 10.3.9, el flujo al terminar el algoritmo 10.2.4 es máximo y el corte (P, \bar{P}) es mínimo.

Sección de ejercicios de repaso

1. ¿Qué es un corte en una red?
2. ¿Qué es la capacidad de un corte?
3. ¿Cuál es la relación entre la capacidad de un corte y el valor de cualquier flujo?
4. ¿Qué es un corte mínimo?
5. Enuncie el teorema del flujo máximo y corte mínimo.
6. Explique cómo demuestra el teorema de flujo máximo y corte mínimo que el algoritmo de la sección 8.2 encuentra correctamente un flujo máximo en una red.

Ejercicios

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

1. $P = \{a, d\}$ para el ejercicio 1, sección 10.1
2. $P = \{a, d, e\}$ para el ejercicio 2, sección 10.1
3. $P = \{a, b, c, d\}$ para el ejercicio 3, sección 10.1

En los ejercicios 4 al 16, encuentre un corte mínimo en cada red.

4. Figura 10.1.1
5. Figura 10.1.4
6. Figura 10.1.5
7. Ejercicio 1, sección 10.1
8. Ejercicio 2, sección 10.1
9. Ejercicio 3, sección 10.1
10. Ejercicio 4, sección 10.1
11. Ejercicio 5, sección 10.1
12. Ejercicio 6, sección 10.1
13. Ejercicio 7, sección 10.1
14. Ejercicio 8, sección 10.1
15. Ejercicio 9, sección 10.1
16. Ejercicio 12, sección 10.2

Los ejercicios 17 al 22 se refieren a una red G que, además de tener capacidades enteras no negativas C_{ij} , tiene requerimientos mínimos m_{ij} de

flujo en enteros no negativos en las aristas. Esto es, un flujo F debe satisfacer

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

para todas las aristas (i, j) .

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

Defina

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

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

18. Demuestre que el valor V de cualquier flujo satisface

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

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

19. Demuestre que si existe un flujo en G , 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 encontrar un flujo máximo en G .