Bài toán luồng cực đại trong mạng là một trong số bài toán tối ưu trên đồ thị tìm được những ứng dụng rộng rãi trong thực tế cũng như những ứng dụng thú vị trong lý thuyết tổ hợp. Bài toán được đề xuất vào đầu năm 1950, và gắn liên với tên tuổi của hai nhà toán học Mỹ là Ford và Fulkerson. Trong chương này chúng ra sẽ trình bày thuật toán Ford và Fulkerson để giải bài toán đặt ra và nêu một sô ứng dụng của bài toán.

**Mạng. Luồng trong mạng. Bài toán luồng cực đại**

Định nghĩa 1 Ta gọi mạng là đồ thị có hướng G = (V,E), trong đó duy nhất một đỉnh s không có cung đi vào gọi là đỉnh phát, duy nhất một đỉnh t không có cung đi ra gọi là điểm thu và mỗi cung e=(v,w) ∈ E được gán với một số không âm c(e) =c(v,w) gọi là khả năng thông qua của cung e.

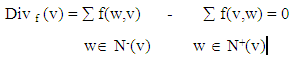
Để thuận tiện cho việc trình bày ta sẽ qui ước rằng nếu không có cung (v,w) thì khả năng thông qua c(v,w) được gán bằng 0.

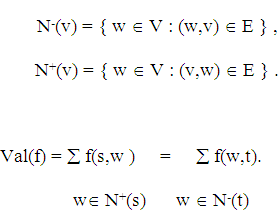
Định nghĩa 2 Giả sử cho mạng G=(V,E). Ta gọi mạng f trong mạng G=(V,E) ;là ánh xạ f: E→R+ gán cho mỗi cung e =(v,w) ∈ E một số thực không âm f(e)=f(v,w), gọi là luồng trên cung e, thoả mãn các điểu kiện sau:

- Luồng trên cung e ∈ E không vượt quá khả năng thông qua của nó:

http://voer.edu.vn/images/stories/m18069_1.1/graphics1.png

- Điều kiện cân bằng luồng trên mỗi đỉnh của mạng: Tổng luồng trên các cung đi vào đỉnh v bằng tổng luồng trên các cung đi ra khỏi đỉnh v, nếu v ≠ s, t:



trong đó N-(v) – tập các đỉnh của mạng mà từ đó có cung đến v, N+(v) - tập các đỉnh của mạng mà từ v có cung đến nó:

Giá trị của luồng f là số

*Bài toán luồng cực đại trong mạng:*

Cho mạng G(V,E). Hãy tìm luồng f\* trong mạng với giá trị luồng val(f\*) là lớn nhất. Luồng như vậy ta sẽ gọi là luồng cực đại trong mạng.

Bài toán như vậy có thể xuất hiện trong rất nhiều ứng dụng thực tế. Chẳng hạn khi cần xác định cường độ lớn nhất của dòng vận tải giữa hai nút của một bản đồ giao thông. Trong ví dụ này lời giải của bài toán luồng cực đại sẽ chỉ cho ta các đoạn đường đông xe nhất và chúng tạo thành "chỗ hẹp" tương ứng với dòng giao thông xét theo hai nút được chọn. Một ví dụ khác là nếu xét đồ thị tương ứng với một hệ thống đường ống dẫn dầu. Trong đó các ống tương ứng với các cung, điểm phát có thể coi là tầu chở dầu, điểm thu là bể chứa, còn những điểm nối giữa các ống là các nút của đồ thị. Khả năng thông qua của các cung tương ứng với tiết diện của các ống. Cần phải tìm luồng dầu lớn nhất có thể bơm từ tàu chở dầu vào bể chứa

**Lát cắt. đường tăng luồng. Định lý ford\_fulkerson**

**Định nghĩa 3**.

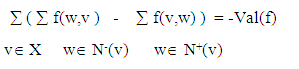
Ta gọi lát cắt (X, X\*) là một cách phân hoạch tập đỉnh V của mạng ra thành hai tập X và X\* = VX, trong đó s ∈ X, t ∈ X\* . Khả năng thông qua của lát cắt (X, X\*) là số

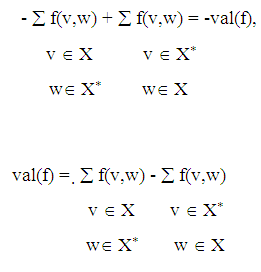
c(X,X\*) = ∑c(v,w) v ∈ X w ∈ X\*

Lát cắt với khả năng thông qua nhỏ nhất được gọi là lát cắt hẹp nhất.

**Bổ đề 1.** Giá trị của luồng f trong mạng luôn nhỏ hơn hoặc bằng khả năng thông qua của lát cắt (X, X\*) bất kỳ trong nó: val(f) ≤ c(X, X\*).

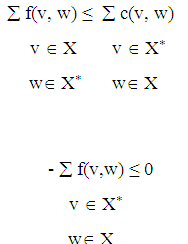
**Chứng minh.** Cộng các điều kiện cân bằng luồng Divf(v)=0 với mọi v∈ X. Khi đó ta có



tổng này sẽ gồm các số hạng dạng f(u,v) với dấu cộng hoặc dấu trừ mà trong đó có ít nhất một trong hai đỉnh u,v phải thuộc tập X. Nếu cả hai đỉnh u,v đều trong tập X, thì f(u,v) xuất hiện với dấu cộng trong Divf(v) và với dấu trừ trong Divf(u), vì thế, chúng triệt tiêu lẫn nhau. Do đó, sau khi giản ước các số hạng như vậy ở vế trái, ta thu được

hay là

Mặt khác, từ điều kiện 1 rõ ràng là



còn

suy ra val(f)≤c(X, X\*). Bổ đề được chứng minh.

**Hệ quả 2.** Giá trị luồng cực đại trong mạng không vượt quá khả năng thông qua của lát cắt hẹp nhất trong mạng.

Ford và Fulkerson đã chứng minh rằng giá trị luồng cực đại trong mạng đúng bằng khả năng thông qua của lát cắt hẹp nhất. Để có thể phát biểu và chứng minh kết quả này chúng ra sẽ cần thêm một số khái niệm.

Giả sử f là một luồng trong mạng G = (V, E). Từ mạng G =(V, E) ta xây dựng đồ thị có trọng số trên cung Gf = (V, Ef), với tập cung Ef và trọng số trên các cung được xác định theo qui tắc sau :

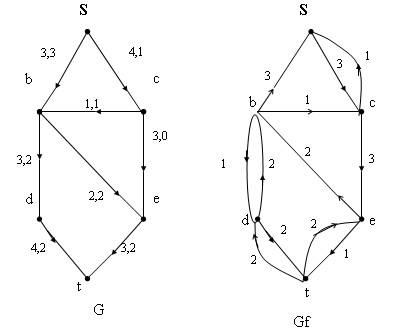
- Nếu e = (v,w) ∈ E với f(v,w) = 0, thì (v,w) ∈ Ef với trọng số c(v,w);

- Nếu e = (v,w) ∈ E với f(v,w) =c(v,w), thì (w,v) ∈ Ef với trọng số f(v,w);

- Nếu e = (v,w) ∈ E với 0 < f(v,w) < c(v,w), thì (v,w) ∈ Ef với trọng số c(v,w) - f (v,w) và (w,v) ∈ Ef với trọng số f(v,w).

Các cung của Gf đồng thời cũng là cung của G được gọi là cung thuận, các cung còn lại được gọi là cung nghịch. Đồ thị Gf được gọi là đồ thị tăng luồng.

**Ví dụ 1**: Các số viết cạnh các cung của G ở hình 9.1 theo thứ tự là khả năng thông qua và luồng trên cung.



*Hình 9.1Mạng G và luồng f. Đồ thị có trọng số Gf tương ứng.*

Giả sử P = (s = v0, v1, . . . , vk = t) là một đường đi từ s đến t trên đồ thị tăng luồng Gf. Gọi ε là giá trị nhỏ nhất của các trọng số của các cung trên đường đi P. Xây dựng luồng f’ trên mạng theo qui tắc sau:

|  |  |
| --- | --- |
|  | f(u,v) + ε , nếu (u,v) ∈ P là cung thuận |
| f’(u,v) = | f(u,v) - ε , nếu (v,u) ∈ P là cung nghịch |
|  | f(u,v), nếu (u,v) ∈ P |

Dễ dàng kiểm tra được rằng f’ được xây dựng như trên là luồng trong mạng và val(f’) = val(f) + ε . Ta sẽ gọi thủ tục biến đổi luồng vừa nêu là tăng luồng dọc theo đường P.

**Định nghĩa 4**. Ta gọi đường tăng luồng f là mọi đường đi từ s đến t trên đồ thị tăng luồng G(f).

**Định lý 3** Các mệnh đề dưới đây là tương đương:

(i) f là luồng cực đại trong mạng;

(ii) không tìm được đường tăng luồng f;

(iii) val(f) = c(X,X\*) với một lát cắt (X, X\*) nào đó.

Chứng minh.

(i) ⇒ (ii). Giả sử ngược lại, tìm được đường tăng luồng P. Khi đó ta có thể tăng giá trị luồng bằng cách tăng luồng dọc theo đường P. Điều đó mâu thuẫn với tính cực đại của luồng f.

(ii) ⇒ (iii). Giả sử không tìm được đường tăng luồng. Ký hiệu X là tập tất cả các đỉnh có thể đến được từ đỉnh s trong đồ thị Gf, và đặt X\*=VX. Khi đó (X, X\*) là lát cắt, và f(v,w) = 0 với mọi v ∈ X\*, w ∈ X nên

val(f) = ∑ f(v,w) - ∑ f(v,w) = ∑ f(v,w) v ∈ X v ∈ X\* v∈ X w∈ X\* w∈ X w∈ X\*

Với v ∈ X, w ∈ X\*, do (v,w) ∈ Gf, nên f(v,w) = c(v,w). Vậy

val(f) = ∑ f(v,w) = ∑ c(v,w) = c(X,X\*) v ∈ X v ∈ Xw∈ X\* w∈ X\*

(iii) ⇒ (i). Theo bổ đề 9.1, val(f) ≤ c(X,X\*) với mọi luồng f và với mọi lát cắt (X,X\*). Vì vậy, từ đẳng thức val(f) = c(X,X\*) suy ra luồng f là luồng cực đại trong mạng.

**Thuật toán tìm luồng cực đại**

**Định lý 3** là cơ sở để xây dựng thuật toán lặp sau đây để tìm luồng cực đại trong mạng: Bắt đầu từ luồng với luồng trên tất cả các cung bằng 0 (ta sẽ gọi luồng như vậy là luồng không), và lặp lại bước lặp sau đây cho đến khi thu được luồng mà đối với nó không còn đường tăng:

Bước lặp tăng luồng (Ford-Fulkerson): Tìm đường tăng P đối với luồng hiện có. Tăng luồng dọc theo đường P.

Khi đã có luồng cực đại, lát cắt hẹp nhất có thể theo thủ tục mô tả trong chứng minh định lý 3. Sơ đồ của thuật toán Ford-Fulkerson có thể mô tả trong thủ tục sau đây:

Procedure Max\_Flow;

(\* Thuật toán Ford-Fulkerson \*)

Begin

(\* Khởi tạo: Bắt đầu từ luồng với giá trị 0 \*)

for u ∈ V do

for v ∈ V do f(u,v) := 0;

stop := false;

while not stop do

if <Tìm được đường tăng luồng P> then <Tăng luồng dọc theo P>

else stop:=true;

End;

Để tìm đường tăng luồng trong G(f) có thể sử dụng thuật toán tìm kiếm theo chiều rộng (hay tìm kiếm theo chiều sâu), bắt đầu từ đỉnh s trong đó không cần xây dựng tường minh đồ thị G(f). Ford-Fulkerson đề nghị thuật toán gán nhãn chi tiết sau đây để giải bài toán luồng cực đại trong mạng. Thuật toán bắt đầu từ luồng chấp nhận được nào đó trong mạng (có thể bắt đầu từ luồng không) , sau đó ta sẽ tăng luồng bằng cách tìm các đường tăng luồng. Để tìm đường tăng luồng ta sẽ áp dụng phương pháp gán nhãn cho các đỉnh. Mỗi đỉnh trong quá trình thực hiện thuật toán sẽ ở một trong ba trạng thái: chưa có nhãn, có nhãn chưa xét, có nhãn đã xét. Nhãn của một đỉnh v gồm hai phần và có một trong hai dạng sau : [,] hoặc []. Phần thứ nhất +p(v) (-p(v)) chỉ ra là cần tăng giảm luồng theo cung (p(v),v)( cung (v, p(v)) còn phần thứ hai chỉ ra lượng lớn nhất có thể tăng hoặc giảm luồng theo cung này. Đầu tiên chỉ có đỉnh s được khởi tạo nhãn và nhãn của nó là chưa xét, còn tất cả các đỉnh còn lại đều chưa có nhãn. Từ s ta gán nhãn cho tất cả các đỉnh kề với nó và nhãn của đỉnh s sẽ trở thành đã xét. Tiếp theo, từ một đỉnh v có nhãn chưa xét ta lại gán nhãn cho tất cả các đỉnh chưa có nhãn kề với nó và nhãn của đỉnh v trở thành đã xét. Quá trình sẽ được lặp lại cho đến khi hoặc là đỉnh t trở thành có nhãn hoặc là nhãn của tất cả các đỉnh có nhãn đầu là đã xét nhưng đỉnh t vẫn không có nhãn. Trong trường hợp thứ nhất ta tìm được đường tăng luồng, còn trong trường hợp thứ hai đối với luồng đang xét không tồn tại đường tăng luồng (tức là luồng đã cực đại). Mỗi khi tìm được đường tăng luồng, ta lại tăng luồng theo đường tìm được, sau đó xoá tất cả các nhãn và đổi với luồng mới thu được lại sử dụng phép gán nhãn các đỉnh để tìm đường tăng luồng. Thuật toán sẽ kết thúc khi nào đối với luồng đang có trong mạng không tìm được đường tăng luồng.

**Hai thủ tục tìm đường tăng luồng có thể mô tả như sau :**

**Procedure Find-path;**

**{ Thủ tục gán nhãn đường tăng luồng**

**p[v], [v] là nhãn của đỉnh v;**

**VT là danh sách các đỉnh có nhãn chưa xét ;**

**c[u,v] là khả năng thông qua của cung (u,v),u,v V;**

**f[u,v] là luồng trên cung (u,v), (u,v V); }**

**BEGIN**

**p[s] := s ; [s] :=; VT := {s};**

**Pathfound := true;**

**While VT <> {} do**

**BEGIN**

**u VT ; ( \* lấy u từ VT \*)**

**For vV do**

**If (v chưa có nhãn) then**

**Begin**

**If (c[u,v] >0) and (f[u,v] < c[u,v] ) then**

**Begin**

**P[v] := u ; [v] := min {[u],c[u,v]-f[u,v] };**

**VT:=VT{v}; (\* nạp v vào danh sách các đỉnh có nhãn \*)**

**If v = t then exit;**

**End**

**Else**

**If (c[v,u] > 0) and (f[v,u] < 0) then**

**Begin**

**P[v] := u ;**

**[v] := min {[u] , f[u,v] };**

**VT:=VT{v};(\* nạp v vào danh sách các đỉnh có nhãn \*)**

**If v = t then exit;**

**End;**

**End;**

**End;**

**PathFound :=false;**

**End;**

**Procedure Inc\_flow ;**

**{ thuật toán tăng luồng theo đường tăng }**

**Begin**

**v := t ; u := t ; tang := [t];**

**while u <> s do**

**begin**

**v := p[u];**

**if v > 0 then f[v,u] := f[v,u] + tang**

**else**

**begin**

**v := -v;**

**f[u,v] :=f[u,v] –tang;**

**end;**

**u := v ;**

**end;**

**Procedure FF;**

**{ thủ tục thể hiện thuật toán Ford\_fulkerson }**

**Begin**

**(\* khởi tạo bắt đầu từ luồng với giá trị 0 \*)**

**For u V do**

**For v V do f[u,v] :=0;**

**Stop := false;**

**While not Stop do**

**begin**

**find\_path;**

**If pathfound then**

**Inc\_flow**

**Else Stop:=true;**

**End;**

**< Luồng cực đại trong mạng là f[u,v], u,v V >**

**< Lát cắt hẹp nhất là (VT , V VT) >**

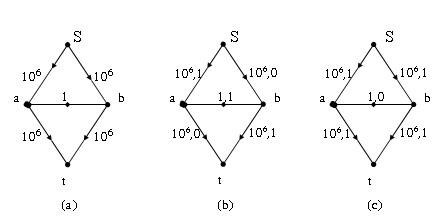
End;

Giả sử là khả năng thông qua của tất cả các khung của đồ thị là các số nguyên. Khi đó sau mỗi lần tăng luồng, giá trị luồng sẽ tăng lên ít nhất là 1. Từ đó suy ra thuật toán Ford\_Fulkerson sẽ dừng sau không quá val(f\*) lần tăng luồng và cho ta luồng cực đại trong mạng. Đồng thời, rõ ràng f\*(u,v) sẽ là số nguyên đối với mỗi cung (u,v) ∈ E. Từ đó ra có các kết quả sau:

**Định lý 4** (Định lý về luồng cực đại trong mạng và lát cắt hẹp nhất). Luồng cực đại trong mạng bằng khả năng thông qua của lát cắt hẹp nhất.

**Định lý 5** (Định lý về tính nguyên). Nếu tất cả các khả năng thông qua là các số nguyên thì luồng tìm được cực đại với luồng trên các cung là các số nguyên.

Tuy nhiên, nếu các khả năng thông qua là các số rất lớn thì giá trị của luồng cực đại cũng có thể là rất lớn và khi đó thuật toán mô tả ở trên sẽ đòi hỏi thực hiện rất nhiều bước tăng luồng. Ví dụ trong hình 2 sẽ minh hoạ cho điều này. Hình 2(a) mô tả mạng cần xét với các khả năng thông qua trên các cung. Hình 2(b) mô tả luồng trên các cung (số thứ hai bên cạnh cung) sau khi thực hiện tăng luồng dọc theo đường tăng luồng (s, a, b, t). Hình 2(c) mô tả luồng trên các cung sau khi thực hiện tăng luồng dọc theo đường tăng luồng (s, b, a, t). Rõ ràng, sau 2.106 lần tăng luồng theo đường (s, a, b, t) và (s, b, a, t) một cách luân phiên ta thu được luồng cực đại.



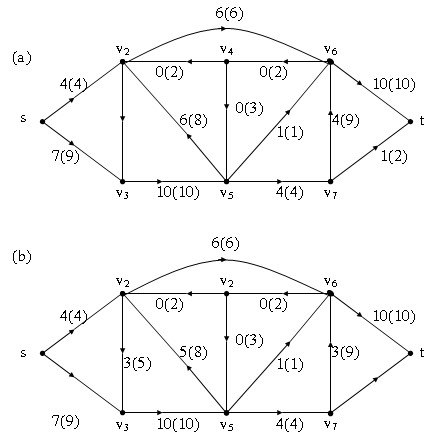
*Hình 2Ví dụ tồi tệ đối với thuật toán Ford\_Fulkerson.*

*Hơn thế nữa, nếu các khả năng thông qua là các số vô tỉ, người ta còn xây dựng được ví dụ để cho thuật toán không dừng, và tệ hơn nếu dãy các giá trị luồng xây dựng theo thuật toán hội tụ thì nó còn không hội tụ đến giá trị luồng cực đại. Như vậy, muốn thuật toán làm việc hiệu quả, việc lựa chọn đường tăng luồng cần được tiến hành hết sức cẩn thận.*

*Edmonds và Karp chỉ ra rằng nếu đường tăng luồng được chọn là đường ngắn nhất từ s đến t trên đồ thị Gf. Điều đó có thể thực hiện, nếu trong thủ tục tìm đường tăng Find\_Path mô tả ở trên, danh sách VT được tổ chức dưới dạng QUEUE (nghĩa là ta thực hiện tìm đường tăng bởi thủ tục tìm kiếm theo chiều rộng) thì thuật toán sẽ kết thúc sau không quá mn/2 lần sử dụng đường tăng luồng. Nếu để ý rằng, tìm kiếm theo chiều rộng trên đồ thị đòi hỏi thời gian O(m+n), thì thuật toán thu được sẽ có độ phức tạp tính toán là O(nm2).*

*Nhờ cách tổ chức tìm đường tăng khéo léo hơn, người ta đã xây dựng được thuật toán với độ phức tạp tính toán tốt hơn như: O(n2m) (Dinic, 1970). O(n3) (Karzanov, 1974), O(n2m2), (Cherkasky, 1977), O(nm log n) (Sleator, - Tarrjan, 1980).*

*Ta kết thúc mục này bởi ví dụ minh hoạ cho thuật toán Ford\_Fulkerson sau đây. Hình 3(a) cho mạng G cùng với thông qua của tất cả các cung và luồng giá trị 10 trong nó. Hai số viết bên cạnh mỗi cung là khả năng thông qua của cung (số trong ngoặc) và luồng trên cung. Đường tăng luồng có dạng (s, v3, v2, v6, v7, t). Ta tính được ε (t) = 1, giá trị luồng tăng từ 10 lên 1. Hình 3 (b) cho luồng thu được sau khi tăng luồng theo đường tăng tìm được.*

**

*Hình 3 Tăng luồng dọc theo đường tăng.*

*Luồng trong hình 3(b) đã là cực đại. Lát cắt hẹp nhất*

*X = { s, v2, v3, v5}, X\* = { v4, v6, v7, t} .*