Bài toán ghép cặp

Graph Matching

Bài toán ghép cặp trên đồ thị

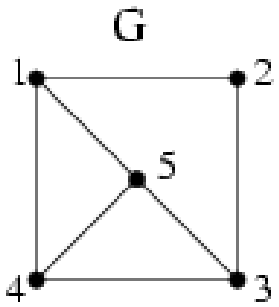
- Gi¶ sö *G*=(*V*,*E*) lµ ®å thÞ v« híng, trong ®ã mçi c¹nh (*v*,*w*) ®îc g¸n víi mét sè thùc *c*(*v*,*w*) gäi lµ träng sè cña nã.
- **§Þnh nghÜa.** *CÆp ghÐp M trªn ®å thÞ G lµ tËp c¸c c¹nh cña ®å thÞ trong ®ã kh«ng cã hai c¹nh nµo cã ®Ønh chung.*
  - *Sè c¹nh trong M - **kÝch thíc**,*
  - *Tæng träng sè cña c¸c c¹nh trong M - **träng lîng** cña cÆp ghÐp.*
  - *CÆp ghÐp víi kÝch thíc lín nhÊt ®îc gäi lµ **cÆp ghÐp cùc ®¹i.***
  - *CÆp ghÐp víi träng lîng lín nhÊt ®îc gäi lµ **cÆp ghÐp lín nhÊt.***
  - *CÆp ghÐp ®îc gäi lµ **®Çy ®ñ (hoµn h¶o)** nÕu mçi ®Ønh cña ®å thÞ lµ ®Çu mót cña Ýt nhÊt mét c¹nh trong cÆp ghÐp.*

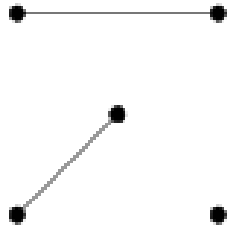
Hai bài toán

- **Bµi to¸n cÆp ghĐp cùc ®¹i**: *T×m cÆp ghĐp víi kÝch thíc lín nhÊt trong ®å thÞ G.*
- **Bµi to¸n cÆp ghĐp lín nhÊt:** *T×m cÆp ghĐp víi träng lîng lín nhÊt trong ®å thÞ G.*

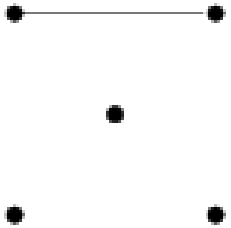
- *Ta h¹n chÕ xĐt c¸c bµi to¸n ®Æt ra trªn ®å thÞ hai phÝa G = (X ∪ Y, E).*

Ví dụ

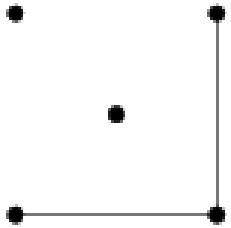




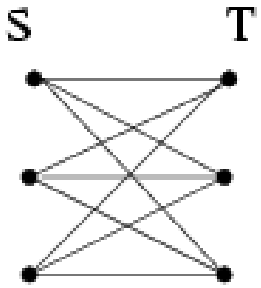
Cặp ghép cực đại

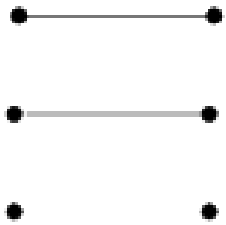


Cặp ghép

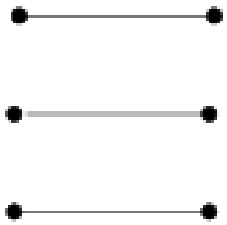


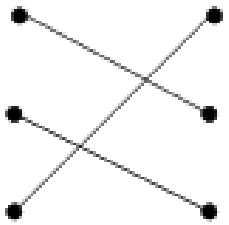
không là cặp ghép





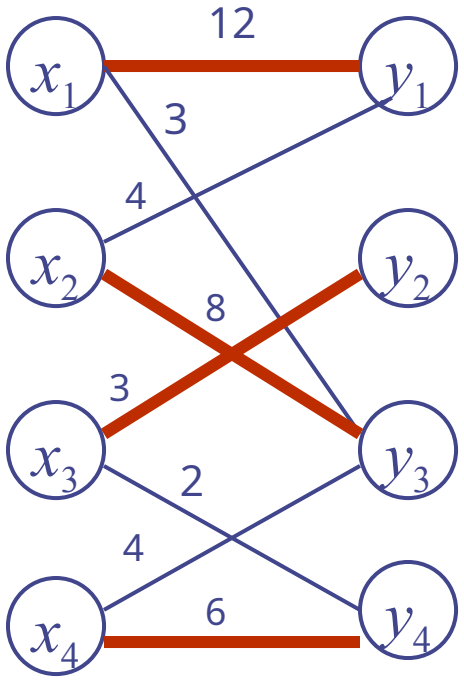
Cặp ghép





Cặp ghép hoàn hảo

Ví dụ



- Cặp ghép lớn nhất:

$$M = \{(x_1, y_1), (x_2, y_3), (x_3, y_2), (x_4, y_4)\}$$

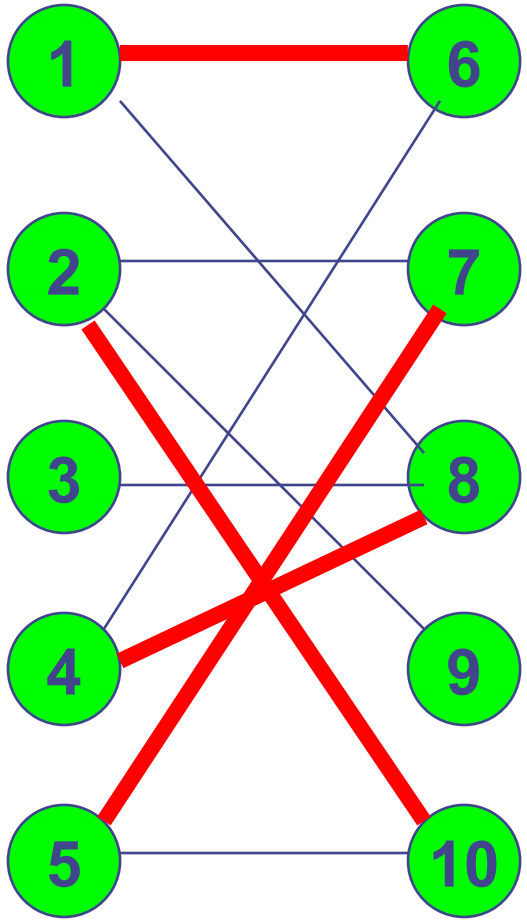
Có trọng lượng 29.

Bµi to¸n cÆp ghÐp cùc ®¹i trªn ®å thÞ hai phÝa

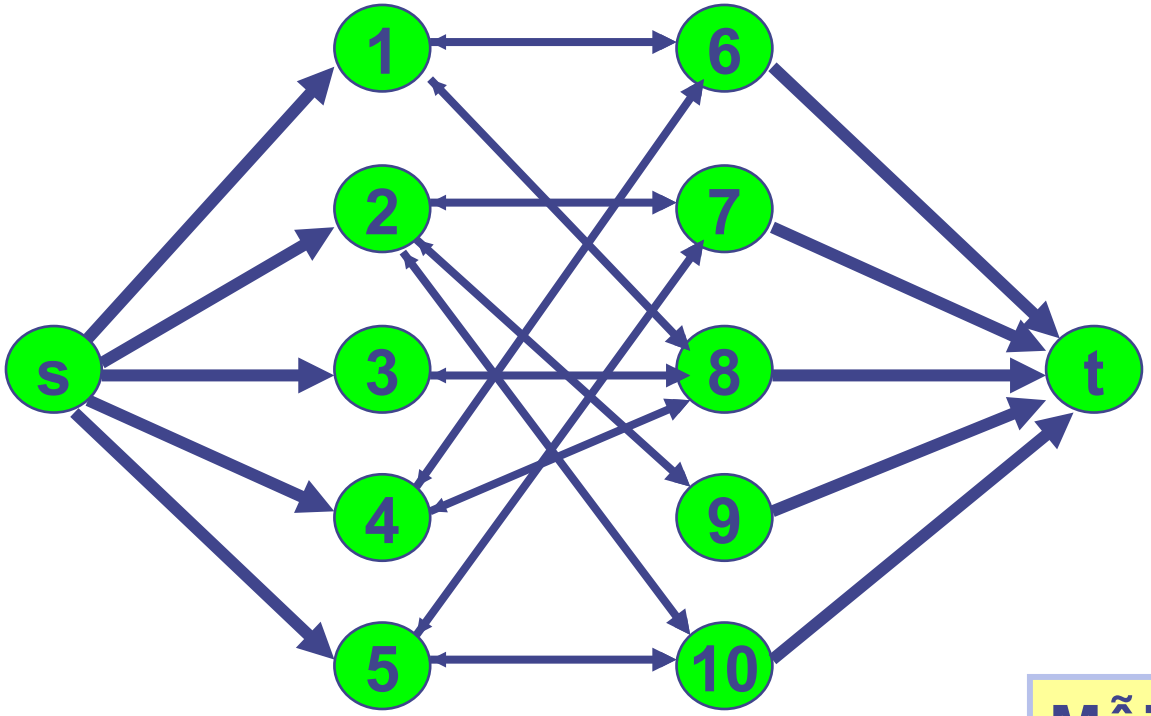
XÐt ®å thÞ hai phÝa
$G = (X \cup Y, E)$.

CÆp ghÐp lµ tËp c¹nh mµ kh«ng cã hai c¹nh nµo cã chung ®Ønh

***Bµi to¸n*: T×m cÆp ghÐp kÝch thíc lín nhÊt**



Qui vÒ Bµi to¸n luång cùc ®¹i

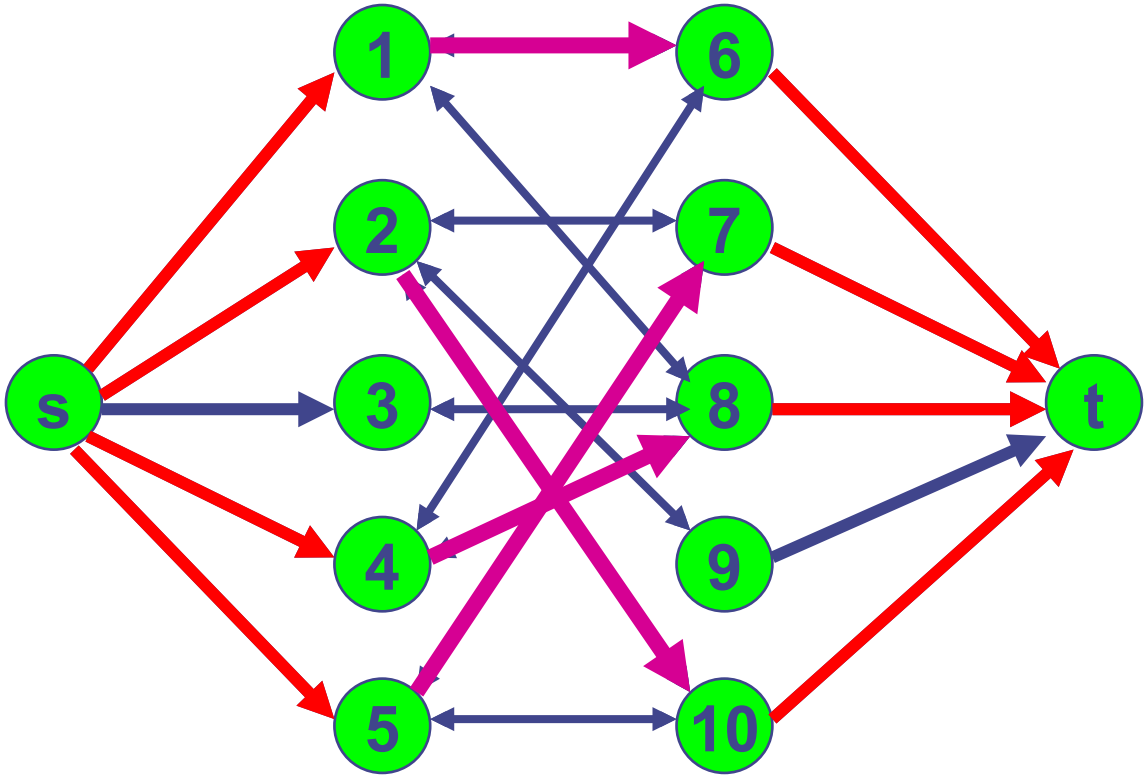


Mỗi cạnh được thay thế bởi cung có kntq 1.

Mỗi cung (s, i) có kntq 1.

Mỗi cung (j, t) có kntq 1.

Tìm luồng cực đại



Luồng cực đại từ s->t có giá trị 4.

Cặp ghép cực đại có kích thước 4.

Bµi to¸n cÆp ghÐp cùc ®¹i trªn ®å thÞ hai phÝa

- Gi¶ sö M lµ mét cÆp ghÐp trªn G.
- NÕu c¹nh $e = (x, y) \in M$, ta nãi e lµ c¹nh cña cÆp ghÐp (hay c¹nh ®Ëm) vµ c¸c ®Ønh x, y lµ c¸c ®Ønh ®Ëm (hay kh«ng tù do).
- NÕu c¹nh $e = (x, y) \notin M$, th× ta nãi e lµ c¹nh nh¹t cßn c¸c ®Ønh x, y lµ c¸c ®Ønh nh¹t (hay tù do).

Đường tăng cặp ghép

- *Mét ®ên g ®i trª n ®å thÞ G mµ trong ®ã hai c¹nh liª n tiÕp lµ kh«ng cïng ®Ëm hay nh¹t sÏ ®îc gäi lµ **®ên g ®i lu©n phiª n** ®Ëm/nh¹t (hay gäi ng¾n gän lµ ®ên g ®i lu©n phiª n).*
- *§ên g ®i lu©n phiª n b¾t ®Çu tõ mét ®Ønh tù do thuéc tËp X vµ kÕt thóc ë mét ®Ønh tù do thuéc tËp Y ®îc gäi lµ **®ên g t¨ng cÆp ghÐp.***

Định lý Berge

- **§Þnh lý 1 (Berge C).** *CÆp ghĐp M lµ cùc ®¹i khi vµ chØ khi kh«ng t×m ®îc ®ên t¨ng cÆp ghĐp.*
- ***CM:***
- ***§iÒu kiÖn cÇn.*** *B»ng ph¶n chøng. Gi¶ sö M lµ cÆp ghĐp cùc ®¹i nh ng vÉn t×m ®îc ®ên t¨ng cÆp ghĐp*

  $$P \equiv x_0, y_1, x_1, y_2, \ldots, x_k, y_0$$

  trong ®ã x_0 vµ y_0 lµ c¸c ®Ønh tù do.

  Gäi E_P lµ tËp c¸c c¹nh cña ®å thÞ n»m trªn ®ên ®i P

  $$E_P = \{ (x_0,y_1), (y_1, x_1), \ldots, (x_k, y_0) \}.$$

  DÔ thÊy sè lîng c¹nh nh¹t trong E_P lµ b»ng sè lîng c¹nh ®Ëm cña nã céng víi 1. §Ó ®¬n gi¶n trong phÇn díi ®©y ta ®ång nhÊt ký hiÖu ® ên ®i P víi tËp c¹nh E_P cña nã. X©y dùng cÆp ghĐp M' theo qui t¾c:

  $$M' = (M \cup P) \setminus (M \cap P).$$

  DÔ thÊy M' còng lµ cÆp ghĐp vµ râ rµng $|M'| = |M| + 1$. M©u thuÉn thu ®îc ®· chøng minh ®iÒu kiÖn cÇn.

Định lý Berge

- **§iÒu kiÖn ®ñ.** Gi¶ sö cÆp ghÐp *M* cha lµ cÆp ghÐp cùc ®¹i. Gäi *M** lµ cÆp ghÐp cùc ®¹i. XÐt ®å thÞ *G'* = (*V*, *M*∪*M**). Râ rµng hai c¹nh liªn tiÕp trong mçi ®êng ®i còng nh mçi chu tr×nh trong *G'* kh«ng thÓ thuéc cïng mét cÆp ghÐp *M* hoÆc *M**. V× vËy, mçi ®êng ®i còng nh mçi chu tr×nh trong *G'* ®Òu lµ ®Ëng lu©n phiªn *M*/*M**. Do |*M**| > |*M*|, nªn râ rµng lµ lu«n t×m ®îc Ýt nhÊt mét ®êng ®i lu©n phiªn *M*/*M** mµ trong ®ã sè lîng c¹nh thuéc *M** lµ lín h¬n sè lîng c¹nh thuéc *M*. §êng ®i ®ã chÝnh lµ ®êng t¨ng cÆp ghÐp trªn ®å thÞ *G*.
- §Þnh lý ®îc chøng minh.
- **Chó ý:** Trong chøng minh ®Þnh lý ta kh«ng sö dông tÝnh hai phÝa cña *G*. Do ®ã, §Þnh lý 1 lµ ®óng víi ®å thÞ v« híng bÊt kú.

ThuËt to¸n t×m cÆp ghÐp cùc ®¹i

- *§Çu vµo:* §å thÞ v« híng $G = (V, E)$.
- *Bíc khëi t¹o.* X©y dùng cÆp ghÐp M trong ®å thÞ G (cã thÓ b¾t ®Çu tõ $M = \emptyset$).
- *Bíc lÆp.*
  - KiÓm tra tiªu chuÈn tèi u: NÕu ®å thÞ G kh«ng chøa ®êng t¨ng cÆp ghÐp th× M lµ cÆp ghÐp cùc ®¹i, thuËt to¸n kÕt thóc.
  - Ngîc l¹i, gäi P lµ mét ®êng t¨ng cÆp ghÐp xuÊt ph¸t tõ ®Ønh tù do $x_0 \in X$, kÕt thóc ë ®Ønh tù do $y_0 \in Y$. T¨ng cÆp ghÐp theo qui t¾c
    $$M := (M \cup P) \setminus (M \cap P),$$
    råi lÆp l¹i bíc lÆp.

Tìm đường tăng

- Tõ ®å thÞ G ta x©y dùng ®å thÞ cã híng $G_M = (X \cup Y, E_M)$ víi tËp cung E_M ®îc b»ng c¸ch ®Þnh híng l¹i c¸c c¹nh cña G theo quy t¾c sau:

  i) NÕu $(x,y) \in M \cap E$, th× $(y,x) \in E_M$;

  ii) NÕu $(x,y) \in E \setminus M$, th× $(x,y) \in E_M$.

  §å thÞ G_M sÏ ®îc gäi lµ ***®å thÞ t¨ng cÆp ghÐp***.

- DÔ thÊy:
  - Đêng t¨ng cÆp ghÐp t¬ng øng víi mét ®êng ®i xuÊt ph¸t tõ mét ®Ønh tù do $x_0 \in X$ kÕt thóc t¹i mét ®Ønh tù do $y_0 \in Y$ trªn ®å thÞ G_M.
  - Ngîc l¹i, mét ®êng ®i trªn ®å thÞ G_M xuÊt ph¸t tõ mét ®Ønh tù do $x_0 \in X$ kÕt thóc t¹i mét ®Ønh tù do $y_0 \in Y$ sÏ t¬ng øng víi mét ®êng t¨ng cÆp ghÐp trªn ®å thÞ G.
- V× vËy, ®Ó xÐt xem ®å thÞ G cã chøa ®êng t¨ng cÆp ghÐp hay kh«ng, cã thÓ thùc hiÖn thuËt to¸n t×m kiÕm theo chiÒu réng trªn ®å thÞ G_M b¾t ®Çu tõ c¸c ®Ønh tù do thuéc tËp X.

Thuật toán

- Sö dông c¸ch t×m ®ªng t¨ng cÆp ghĐp theo nhËn xĐt võa nªu, tõ s¬ ®å tæng qu¸t dÔ dµng x©y dùng thuËt to¸n ®Ó gi¶i bµi to¸n t×m cÆp ghĐp cùc ®¹i trªn ®å thÞ hai phÝa víi thêi gian tÝnh $O(n^3)$, trong ®ã $n = \max(|X|, |Y|)$.

Cài đặt

Cấu trúc dữ liệu

```
Var
    A : Array[1..100,1..100] of Byte; (* Ma trận kề của đồ thi hai phía G *)
    Truoc,                            (* Ghi nhận đường đi *)
    Vo,                               (* Vo[x]- đỉnh được ghép với x∈X *)
    Chong : Array[1..100] of Byte;    (* Chong[y]-đỉnh được ghép với y ∈Y *)
    N, x0, y0, Cnt : Byte;
    Stop : Boolean;

(* Nếu (x, y) ∈ M thì Vo[x]=y; Chong[y]=x.
  Vo[x]=0 => x là đỉnh nhạt; Chong[y]=0 => y là đỉnh nhạt *)
```

Tìm đường tăng

```
Procedure Tim(x:Byte);
  var y: Byte;
   begin
     For y:=1 to N do
      If (A[x,y]=1)and(Truoc[y]=0)and(y0=0) then
     begin
         Truoc[y]:=x;
         If Chong[y]<>0 then Tim(Chong[y])
         else
         begin
                y0:=y;
                Exit;
         end;
      end;
   end;
```

```
Procedure Tim_Duong_Tang;
begin
    Fillchar(Truoc,Sizeof(Truoc),0);
    y0:=0;
    For x0:=1 to N do
    begin
       If Vo[x0]=0 then Tim(x0);
       If y0<>0 then exit;
    end;
    Stop:=true;
end;
```

Thủ tục MaxMatching

```
Procedure Tang;
var temp: Byte;
begin
    Inc(Cnt);
    While Truoc[y0]<>x0 do
    begin
       Chong[y0]:=Truoc[y0];
       Temp:=Vo[Truoc[y0]];
       Vo[Truoc[y0]]:=y0;
       y0:=Temp;
    end;
    Chong[y0]:=x0;
    Vo[x0]:=y0;
end;
```

```
Procedure MaxMatching;
begin
    Stop:=false;
    Fillchar(Vo,Sizeof(Vo),0);
    Fillchar(Chong,Sizeof(Chong),0);
    Cnt:=0;
   While not Stop do
    begin
       Tim_duong_tang;
       If not Stop then Tang;
    end;
end;
```

Bµi to¸n ph©n c«ng

Cã *n* c«ng viÖc vµ *n* thî. Mçi thî ®Òu cã kh¶ n¨ng thùc hiÖn tÊt c¶ c¸c c«ng viÖc. BiÕt

w_{ij} - hiÖu qu¶ ph©n c«ng thî *i* lµm viÖc *j*,

($i, j = 1, 2, \ldots, n$).

CÇn t×m c¸ch ph©n c«ng thî thùc hiÖn c¸c c«ng viÖc sao cho mçi thî chØ thùc hiÖn mét viÖc vµ mçi viÖc chØ do mét thî thùc hiÖn, ®ång thêi tæng hiÖu qu¶ thùc hiÖn c¸c c«ng viÖc lµ lín nhÊt.

Qui về bài toán cặp ghép lớn nhất

X©y dùng ®å thÞ hai phÝa ®Çy ®ñ $G = (X \cup Y, E)$

- $X=\{x_1, x_2,..., x_n\}$ t¬ng øng víi c¸c thî,
- $Y = \{y_1, y_2,..., y_n\}$ - t¬ng øng víi c¸c c«ng viÖc.

Mçi c¹nh (x_i, y_j) ®îc g¸n cho träng sè $w(x_i, y_j) = w_{ij}$.

Khi ®ã trong ng«n ng÷ ®å thÞ, bµi to¸n ph©n c«ng cã thÓ ph¸t biÓu nh sau: T×m trong ®å thÞ G cÆp ghÐp ®Çy ®ñ cã tæng träng sè lµ lín nhÊt. CÆp ghÐp nh vËy ®îc gäi lµ cÆp ghÐp tèi u.

C¬ së thuËt to¸n

Ta gäi mét ***phÐp g¸n nh·n chÊp nhËn ®îc*** *cho c¸c ®Ønh cña ®å thÞ G=(X∪Y,E) lµ mét hµm sè f x¸c ®Þnh tr^{a}n tËp ®Ønh X∪Y: f: X∪Y → R, tho¶ m·n*

$$f(x) + f(y) \geq w(x,y), \quad \forall x \in X, \quad \forall y \in Y.$$

Mét phÐp g¸n nh·n chÊp nhËn ®îc nh vËy dÔ dµng cã thÓ t×m ®îc, ch¼ng h¹n phÐp g¸n nh·n sau ®©y lµ chÊp nhËn ®îc

$$f(x) = \max \{ w(x,y): \ y \in Y \}, \ x \in X,$$

$$f(y) = 0 , \ y \in Y.$$

Đồ thị cân bằng

- Gi¶ sö cã f lµ mét phÐp g¸n nh·n chÊp nhËn ®îc, ký hiÖu

$$E_f = \{(x,y) \in E: \ f(x) + f(y) \ = w(x,y)\}.$$

- Ký hiÖu G_f lµ ®å thÞ con cña G sinh bëi tËp ®Ønh $X \cup Y$ vµ tËp c¹nh E_f. Ta sÏ gäi G_f lµ ***®å thÞ c©n b»ng***.

Tiêu chuẩn tối ưu

§Þnh lý 2. *Gi¶ sö f lµ phÐp g¸n nh·n chÊp nhËn ®îc. NÕu G_f chøa cÆp ghÐp ®Çy ®ñ M^*, th× M^* lµ cÆp ghÐp tèi u.*

Chøng minh.

Gi¶ sö G_f chøa cÆp ghÐp ®Çy ®ñ M^*. Khi ®ã tõ ®Þnh nghÜa G_f suy ra M^* còng lµ cÆp ghÐp ®Çy ®ñ cña ®å thÞ G. Gäi $w(M^*)$ lµ träng lîng cña M^*:

$$w(M^*) = \sum_{e \in M^*} w(e)$$

Do mçi c¹nh $e \in M^*$ ®Òu lµ c¹nh cña G_f vµ mçi ®Ønh cña G kÒ víi ®óng mét c¹nh cña M^*, nªn

$$w(M^*) = \sum_{e \in M^*} w(e) = \sum_{v \in V} f(v)$$

Gi¶ sö M lµ mét cÆp ghÐp ®Çy ®ñ tuú ý cña G, khi ®ã

$$w(M) = \sum_{e \in M} w(e) \le \sum_{v \in V} f(v)$$

Suy ra $w(M^*) \ge w(M)$. VËy M^* lµ cÆp ghÐp tèi u.

Sơ đồ thuật toán

- Ta sÏ b¾t ®Çu tõ mét phÐp g¸n nh·n chÊp nhËn ®îc *f*. X©y dùng ®å thÞ G_f. B¾t ®Çu tõ mét cÆp ghÐp *M* nµo ®ã trong G_f ta x©y dùng cÆp ghÐp ®Çy ®ñ trong G_f. NÕu t×m ®îc cÆp ghÐp ®Çy ®ñ *M**, th× nã chÝnh lµ cÆp ghÐp tèi u. Ngîc l¹i, ta sÏ t×m ®îc cÆp ghÐp cùc ®¹i kh«ng ®Çy ®ñ *M*'. Tõ *M*' ta sÏ t×m c¸ch söa phÐp g¸n nh·n thµnh *f*' sao cho *M*' vÉn lµ cÆp ghÐp cña G_f' vµ cã thÓ tiÕp tôc ph¸t triÓn *M*' trong G_f'., v.v...
- Qu¸ tr×nh ®îc tiÕp tôc cho ®Õn khi thu ®îc cÆp ghÐp ®Çy ®ñ trong ®å thÞ c©n b»ng.

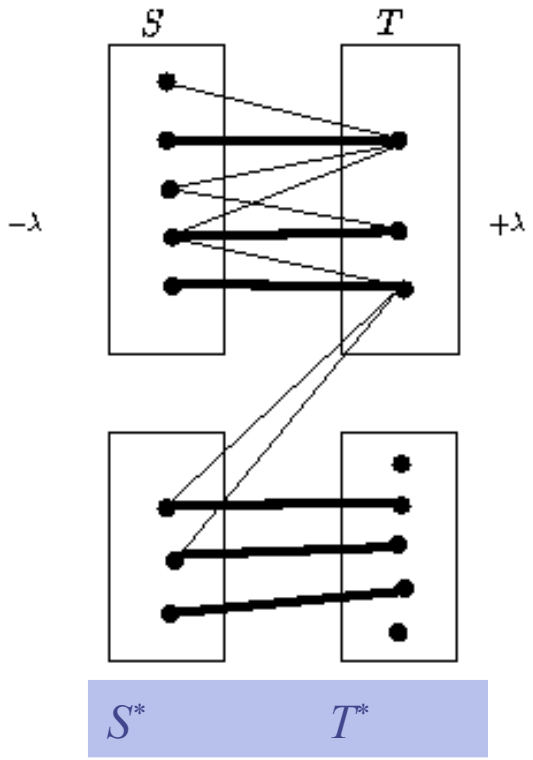
Điều chỉnh nhãn

- Gi¶ sö M lµ cÆp ghÐp cùc ®¹i trong ®å thÞ G_f vµ M cha lµ cÆp ghÐp ®Çy ®ñ cña G. Ta cÇn t×m c¸ch ®iÒu chØnh phÐp g¸n nh·n f tho¶ m·n c¸c yªu cÇu ®Æt ra.
- Thùc hiÖn t×m kiÕm theo chiÒu réng tõ c¸c ®Ønh tù do trong X. Gäi S lµ c¸c ®Ønh ®îc th¨m trong X, cßn T lµ c¸c ®Ønh ®îc th¨m trong Y trong qu¸ tr×nh thùc hiÖn t×m kiÕm. Ký hiÖu $S^* = X \setminus S;\quad T^* = Y \setminus T$.

  $|S| > |T|$ (do mçi ®Ønh trong T ®¹t ®îc tõ mét ®Ønh nµo ®ã trong S).

Điều chỉnh nhãn

- Tõ tÝnh chÊt cña thuËt to¸n t×m kiÕm theo chiÒu réng, rå rµng, kh«ng cã c¹nh nµo tõ *S* ®Õn *T**. §Ó söa ch÷a nh·n, chóng ta sÏ tiÕn hµnh gi¶m ®ång lo¹t c¸c nh·n trong *S* ®i cïng mét gi¸ trÞ λ nµo ®ã, vµ ®ång thêi sÏ t¨ng ®ång lo¹t nh·n cña c¸c ®Ønh trong *T* lªn λ. §iÒu ®ã ®¶m b¶o c¸c c¹nh tõ *S* sang *T* (nghÜa lµ nh÷ng c¹nh mµ mét ®Çu mót thuéc *S* cßn mét ®Çu mót thuéc *T*) kh«ng bÞ lo¹i bá khái ®å thÞ c©n b»ng



C¸c tËp *S* vµ *T* trong thùc hiÖn thuËt to¸n. ChØ vÏ c¸c c¹nh trong G_f.

Điều chỉnh nhãn

- Khi c¸c nh·n trong S bÞ gi¶m, c¸c c¹nh trong G tõ S sang T^* sÏ cã kh¶ n¨ng gia nhËp vµo ®å thÞ c©n b»ng G_f. Ta sÏ t¨ng λ ®Õn khi cã thªm Ýt nhÊt mét c¹nh míi gia nhËp ®å thÞ c©n b»ng. Cã hai kh¶ n¨ng:
  - NÕu c¹nh míi gia nhËp ®å thÞ c©n b»ng gióp ta th¨m ®îc mét ®Ønh kh«ng tù do $y \in T^*$ th× tõ nã ta sÏ th¨m ®îc mét ®Ønh ®îc ghÐp víi nã trong cÆp ghÐp $x \in S^*$, vµ c¶ hai ®Ønh nµy ®îc bæ sung vµo S vµ T t¬ng øng, vµ nh vËy viÖc t×m kiÕm ®êng t¨ng sÏ ®îc tiÕp tôc më réng.
  - NÕu c¹nh míi gia nhËp ®å thÞ c©n b»ng cho phÐp th¨m ®îc mét ®Ønh tù do $y \in T^*$ th× ta t×m ®îc ®êng t¨ng cÆp ghÐp, vµ kÕt thóc mét pha ®iÒu chØnh nh·n.

Điều chỉnh nhãn

Ta gäi ***mét pha ®iÒu chØnh*** lµ tÊt c¶ c¸c lÇn söa nh·n cÇn thiÕt ®Ó t¨ng ®îc kÝch thíc cña cÆp ghÐp *M*.

- V× sau mçi pha ®iÒu chØnh kÝch thíc cña cÆp ghÐp t¨ng lªn 1, nªn ta ph¶i thùc hiÖn nhiÒu nhÊt *n* pha ®iÒu chØnh.
- Trong mçi pha ®iÒu chØnh, do sau mçi lÇn söa nh·n cã Ýt nhÊt hai ®Ønh míi ®îc bæ sung vµo danh s¸ch c¸c ®Ønh ®îc th¨m, nªn ta ph¶i thùc hiÖn viÖc söa nh·n kh«ng qu¸ *n* lÇn. MÆt kh¸c, trong thêi gian $O(n^2)$ ta cã thÓ x¸c ®Þnh ®îc c¹nh nµo tõ *S* sang T^* lµ c¹nh gia nhËp ®å thÞ c©n b»ng (b»ng viÖc duyÖt hÕt c¸c c¹nh). Tõ ®ã suy ra ®¸nh gi¸ thêi gian tÝnh cña thuËt to¸n lµ $O(n^4)$.

Thuật toán

- **Bíc 0:** T×m mét phÐp g¸n nh·n chÊp nhËn ®îc *f*.
- **Bíc 1:** X©y dùng ®å thÞ c©n b»ng G_f.
- **Bíc 2:** T×m cÆp ghÐp cùc ®¹i *M* trong G_f.
- **Bíc 3:** NÕu *M* lµ cÆp ghÐp ®Çy ®ñ th× nã lµ cÆp ghÐp lín nhÊt cÇn t×m. ThuËt to¸n kÕt thóc.
- **Bíc 4:** Gäi *S* lµ tËp c¸c ®Ønh tù do trong *X*. Thùc hiÖn t×m kiÕm tõ c¸c ®Ønh trong *S*. Gäi *T* lµ tËp c¸c ®Ønh cña *Y* ®îc th¨m trong qu¸ tr×nh t×m kiÕm. Bæ sung c¸c ®Ønh trong *X* ®îc th¨m trong qu¸ tr×nh t×m kiÕm vµo *S*.
- **Bíc 5:** TiÕn hµnh ®iÒu chØnh nh·n *f* ta sÏ bæ sung ®îc c¸c c¹nh vµo G_f cho ®Õn khi t×m ®îc ®ªng t¨ng, bæ sung c¸c ®Ønh míi ®îc th¨m vµo *S* vµ *T* t¬ng øng nh ®· m« t¶ ë trªn. T¨ng cÆp ghÐp *M* vµ quay l¹i bíc 3.

Tăng hiệu quả

- §Ó cã ®îc thuËt to¸n víi ®¸nh gi¸ thêi gian tÝnh tèt h¬n, vÊn ®Ò ®Æt ra lµ lµm thÕ nµo cã thÓ tÝnh ®îc gi¸ trÞ λ t¹i mçi lÇn söa nh·n cña pha ®iÒu chØnh mét c¸ch nhanh chãng.
- Ta x¸c ®Þnh ®é lÖch cña c¸c c¹nh theo c«ng thøc

$$slack(x, y) = f(x) + f(y) – c(x, y).$$

Tăng hiệu quả

- Khi ®ã

$$\lambda = \min_{x \in S, y \in T^*} slack(x, y)$$

- Râ rµng viÖc tÝnh trùc tiÕp λ theo c«ng thøc ®ßi hái thêi gian $O(n^2)$. B©y giê, nÕu víi mçi ®Ønh trong T^* ta ghi nhËn l¹i c¹nh víi ®é lÖch nhá nhÊt

$$slack(y_j) = \min_{x_i \in S} slack(x_i, y_j).$$

Tăng hiệu quả

- ViÖc tÝnh gi¸ trÞ ®é lÖch *slack*(y_j) ®ßi hái thêi gian $O(n^2)$ ë ®Çu pha ®iÒu chØnh. Khi tiÕn hµnh pha ®iÒu chØnh ta cã thÓ söa l¹i tÊt c¶ c¸c ®é lÖch trong thêi gian $O(n)$ do chóng bÞ thay ®æi cïng mét gi¸ trÞ (do nh·n cña c¸c ®Ønh trong S gi¶m ®ång lo¹t ®i cïng mét gi¸ trÞ λ). Khi mét ®Ønh x ®îc chuyÓn tõ S^* sang S ta cÇn tÝnh l¹i c¸c ®é lÖch cña c¸c ®Ønh trong T^*, viÖc ®ã ®ßi hái thêi gian $O(n)$. Tuy nhiªn sù kiÖn mét ®Ønh ®îc chuyÓn tõ S^* sang S chØ x¶y ra nhiÒu nhÊt n lÇn.
- Nh vËy, mçi pha ®iÒu chØnh cã thÓ cµi ®Æt víi thêi gian $O(n^2)$. Do cã kh«ng qu¸ n pha ®iÒu chØnh trong thuËt to¸n, nªn c¸ch cµi ®Æt nµy cho ta thuËt to¸n víi thêi gian tÝnh $O(n^3)$.

Ví dụ

- Xét bài toán với ma trận hiệu quả

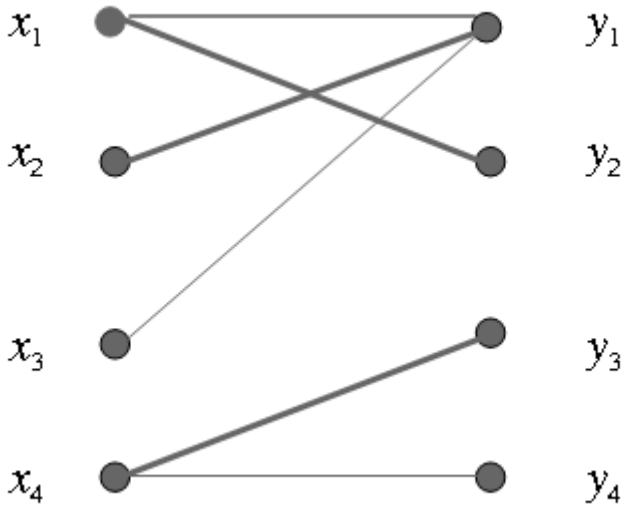
$$W = \begin{vmatrix} 4 & 4 & 1 & 3 \\ 3 & 2 & 2 & 1 \\ 5 & 4 & 4 & 3 \\ 1 & 1 & 2 & 2 \end{vmatrix}$$

Ví dụ

- Bắt đầu từ phép gán nhãn

$f(y)$ / $f(x)$	0	0	0	0
4	4	4	1	3
3	3	2	2	1
5	5	4	4	3
2	1	1	2	2

Đồ thị cân bằng G_f



CÆp ghĐp cùc ®¹i t×m ®îc

$M = \{(x_1,y_2), (x_2,y_1), (x_4, y_4)\}$.

T×m kiÕm theo chiÒu réng b¾t ®Çu tõ ®Ønh tù do x_3 ta cã

$S = \{x_2, x_3\}$, $T = \{y_1\}$

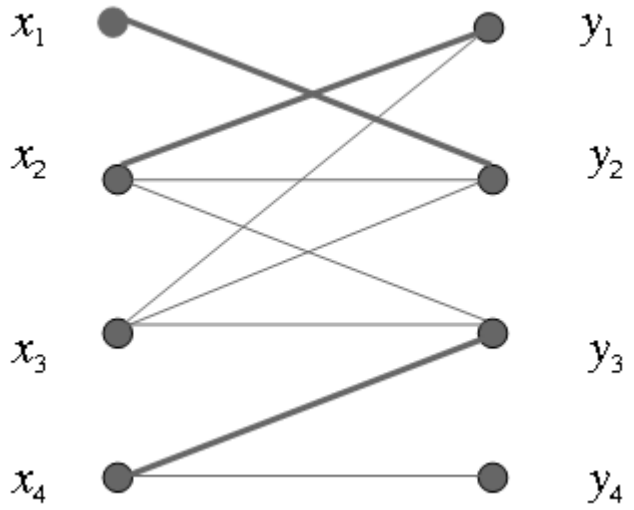
Ví dụ

- TÝnh

  $\lambda = \min \{f(x)+f(y)-w(x,y): x \in \{x_2, x_3\}, y \in \{y_2, y_3, y_4\}\} = 1.$

- TiÕn hµnh söa nh·n, t¬ ®i ®Õn nhDn

$f(y)$ / $f(x)$	1	0	0	0
4	4	4	1	3
2	3	2	2	1
4	5	4	4	3
2	1	1	2	2



Ví dụ

- Theo ®ên t¨ng cÆp ghĐp

  x_3, y_3, x_4, y_4

  ta t¨ng cÆp ghĐp M thµnh cÆp ghĐp ®Çy ®ñ

  $M = \{(x_1,y_2), (x_3,y_1), (x_2,y_3), (x_4,y_4)\},$

  ®ång thêi lµ cÆp ghĐp tèi u víi träng l îng

  $w(M) = 4 + 2 + 5 + 2 = 13.$

Cài đặt trên Pascal

```
const maxn = 170;

type    data1=array [1..maxn,1..maxn] of integer;
        data2=array [1..2*maxn] of integer;
        data3=array [1..2*maxn] of longint;

var   c: data1;
      px, py, q, queue: data2;
      a, b, f: data3;
      n, n2, k, u, z: integer;
```

Khởi tạo

```
procedure init;
var i, j: integer;
begin
     n2:= n+n; fillchar(f,sizeof(f),0);
     for i:=1 to n do
      for j:=1 to n do
       if f[i]<c(i,j) then f[i]:=c(i,j);
     k:=0;
     fillchar(px,sizeof(px),0); fillchar(py,sizeof(py),0);
     for i:=1 to n do
      for j:=1 to n do
       if (py[j]=0) and (f[i]+f[j+n]=c(i,j)) then
       begin
          px[i]:=j; py[j]:=i;  inc(k);
          break;
       end;
end;
```

Tìm đường tăng

```
function FoundIncPath: boolean;
var dq, cq, v, w: integer;
begin
   fillchar(q,sizeof(q),0);
   dq:=1; cq:=1; queue[dq]:=u; q[u]:=u;
   while dq<=cq do
   begin
      v:=queue[dq]; inc(dq);
      if v<=n then
      begin
         for w:=n+1 to n2 do
          if (f[v]+f[w]=c(v,w-n)) and (q[w]=0) then
          begin  inc(cq); queue[cq]:=w; q[w]:=v; end;
      end  else
      if (py[v-n]=0) then begin  FoundIncPath:=true;z:=v;exit; end
      else  begin w:=py[v-n];  inc(cq); queue[cq]:=w; q[w]:=v; end;
   end;
   FoundIncPath:=false;
end;
```

Tìm đỉnh tự do

```
function FreeNodeFound :boolean;
var i:integer;
begin
   for i:=1 to n do
      if px[i]=0 then
      begin
          u:=i;
          FreeNodeFound:=true;
          exit;
      end;
  FreeNodeFound :=false;
end;
```

Tăng cặp ghép và Sửa nhãn

```
procedure Tangcapghep;
var i, j: integer;
    ok: boolean;
begin
   j:=z; ok:=true;
   while j<>u do
   begin
      i:=q[j];
      if ok then
      begin
         px[i]:=j-n;
         py[j-n]:=i;
      end;
      j:=i;
      ok:= not ok;
   end;
   inc(k);
end;
```

```
procedure Suanhan;
var i, j: integer;
    d: longint;
begin
     d:= maxlongint;
     for i:=1 to n do
        if q[i]>0 then
           for j:=n+1 to n2 do
              if q[j]=0 then
                 if d>longint(f[i]+f[j]-c(i,j-n)) then
                       d:=longint(f[i]+f[j]-c(i,j-n));
     for i:=1 to n do
        if q[i]>0 then dec(f[i],d);
     for j:=n+1 to n2 do
        if q[j]>0 then inc(f[j],d);
end;
```

Main Procedure

```
procedure Solve;
begin
    init;
    while FreeNodeFound do
    begin
        while not FoundIncPath do suanhan;
        Tangcapghep;
    end;
end;
```