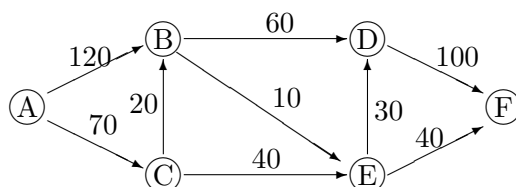


### 3.3 最大流問題

#### 3.3.1 概要

[問題 3.2] 下図で与えられたネットワークにおいて、節点 1 から節点 6 まで最大どれだけ流すことができるか。ただし、枝に与えられた値はその枝の容量を表す。



ソースを  $s$  , シンクを  $t$  , 枝  $(i, j)$  の容量を  $u_{i,j}$  とする。最大流問題を線形計画問題として定式化すると以下のとおり。  $x_{i,j}$  は枝  $(i, j)$  に流れる量を表す変数。

目的関数： \_\_\_\_\_ → 最大化

$$\text{制約条件： } \sum_{(s,j) \in E} x_{s,j} - \sum_{(i,s) \in E} x_{i,s} = f \quad (3.3)$$

$$\text{_____} \quad (3.4)$$

$$\text{_____} \quad (3.5)$$

$$0 \leq x_{i,j} \leq u_{i,j} \quad ((i, j) \in E) \quad (3.6)$$

ソースとシンク以外の接点に対する制約条件 (式 (3.4)) を \_\_\_\_\_ という。最後の制約条件 (式 (3.6)) は、\_\_\_\_\_ という。また、流れ保存則と容量制約条件を満たしている  $x = \{x_{ij}\}$  を \_\_\_\_\_ と呼ぶ。また、そのときの  $f$  の値を \_\_\_\_\_ と呼ぶ。

ここでは、フロー増加法と呼ばれるアルゴリズムについて説明する。

以後、ネットワークには枝  $(i, j)$  と枝  $(j, i)$  が同時に存在することはないと仮定する。

#### 3.3.2 フロー増加法

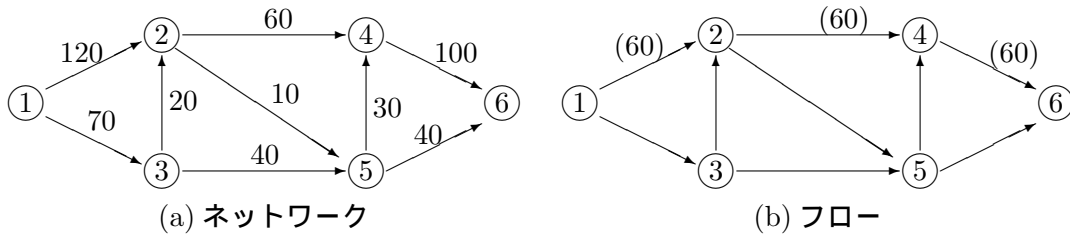
あるフロー  $x = \{x_{ij}\}$  が得られているとする。

残余ネットワーク: 元のネットワーク  $G = (V, E)$  の各枝  $(i, j) \in E$  を、容量  $u_{ij}^x = u_{ij} - x_{ij}$  を持つ枝  $(i, j)$  と容量  $u_{ji}^x = x_{ij}$  を持つ枝  $(j, i)$  に置き換えたネットワーク ( $u_{ij}^x = 0$  の場合は枝  $(i, j)$  は \_\_\_\_\_ とする。)  $G^x = (V, E^x)$  と書く。また、残余ネットワークにおける  $u_{ij}^x$  の値を \_\_\_\_\_ と呼ぶ。

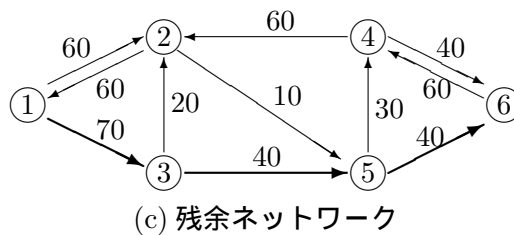
フロー増加路： 残余ネットワークにおけるソースからシンクへの路．

残余ネットワークは，与えられたフロー以上にどれだけ流せるかを示すネットワークとなる．したがって，フロー増加路が存在するときには，その路にそって，さらにフローを増加させることができる．

たとえば，下図 (a) で示すネットワークで，(b) で示す節点 1 から節点 6 のフローが得られているとする．

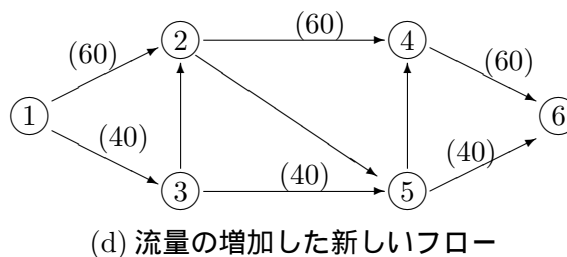


この場合，残余ネットワークは次の図 (c) のようになり，太矢印で示すような増加路が存在する．



したがって，フロー増加路に沿ってフローを増加させることができる．また，フロー増加路によって追加できるフローの量は，増加路に含まれる枝の残余容量の \_\_\_\_\_ である．この例の場合は，\_\_\_\_ である．

元々得られていたフローに，増加路に沿ったフローを追加すると，図 (d) のような流量の大きい新しいフローが得られる．



以上の作業を，増加路が存在しないフローが得られるまで繰り返して最大流を求める方法を，フロー増加法という．具体的な手順は以下のとおり．

### フロー増加法

- (Step 0) 初期フローを得る (たとえば, すべての枝について  $x_{ij} = 0$ )
- (Step 1) 残余ネットワークを作り, フロー増加路を見つける. 存在しなければ終了.
- (Step 2) フロー増加路に沿って, フローを追加する. 追加するフローの量は, フロー増加路に含まれる枝の \_\_\_\_\_ . Step 1 に戻る.

フロー増加法では, 残余ネットワークからフロー増加路を見つける必要がある. 上の例のように小さなネットワークの場合は増加路を見つけることはたやすいが, 大きなネットワークでは見て探すことは困難. フロー増加路の見つけ方 (アルゴリズム) については, 次節で扱う.

[課題 3.2] 前ページに示したネットワークの例 (図 (a)) に対して,

- (1) 図 (d) のフローに対する残余ネットワークを図示せよ.
- (2) フロー増加路を見つけ, さらに流量の増加したフローを図示せよ.

### 3.3.3 フロー増加路を得る方法 — ラベリング法

ラベリング法: ネットワーク (グラフ)  $G = (V, E)$  とソース  $s \in V$  とシンク  $t \in V$  が与えられたときに, ソースからシンクへの路を見つけるアルゴリズム.

ラベリング法では, ソースから到達可能な点に順次ラベルをつけて行く. シンクにラベルがついたとき, ラベルを逆にたどればソースからシンクへの路が求まる. 具体的には以下のとおり.

### ラベリング法

- (Step 0) 初期状態として  $L = \{s\}$ ,  $S = \{\}$  とし, 全ての節点  $i \in V$  に対してラベルを  $p(i) = 0$  とする.
- (Step 1) 節点  $\hat{i} \in L \setminus S$  を一つ選び,  $S \leftarrow S \cup \{\hat{i}\}$  とする.
- (Step 2) すべての枝  $(\hat{i}, j) \in E$  について, 以下を行う.
- $$j \notin L \text{ ならば } L \leftarrow L \cup \{j\}, p(j) \leftarrow \hat{i}$$
- (Step 3)  $t \in L$  または  $L = S$  ならば終了. そうでなければ, Step 1 にもどる.

- 集合  $L$  は, ラベルづけされた ( \_\_\_\_\_ ) 節点の集合

- 集合  $S(\subseteq L)$  は, \_\_\_\_\_ の節  
点の集合

したがって,  $t \in L$  で終了した場合はソースからシンクへの路 (つまり増加路) が求まっている. 一方,  $L = S$  でラベリング法が終了した場合は, 経路が存在しないことを意味するので, フロー増加法も終了となる.

例として, 先の図 (c) の残余ネットワークを考えよう (増加路を見つけるためには, 残余ネットワークに対してラベリング法を適用することに注意.)

(0) 初期状態は  $L = \{1\}, S = \{\}, p(1) = p(2) = \cdots = p(6) = 0$ .

(1)  $L \setminus S = \{1\}$  より  $\hat{i} = 1$  を選択,  $S = \{1\}$ .

$(1, 2), (1, 3) \in E$  より  $L \leftarrow \{1, 2, 3\}, p(1) = p(2) = 1$ .

(2)  $L \setminus S = \{2, 3\}$  より  $\hat{i} = 2$  を選択,  $S = \{1, 2\}$ .

\_\_\_\_\_  $\in E$  より  $L \leftarrow \{\text{_____}\}, \text{_____}$ .

(3)  $\hat{i} = 3$  を選択,  $S = \{1, 2, 3\}$ .

\_\_\_\_\_  $\in E$  だが, \_\_\_\_\_.

(4)  $\hat{i} = 5$  を選択,  $S = \text{_____}$ .

\_\_\_\_\_  $\in E$  より  $L = \text{_____}, \text{_____}$ .

ここで,  $6 \in L$  なので終了.

この場合は, 終了条件より節点 1 から節点 6 への路が求まっている. 路は  $p(6) = 5, p(5) = 2, p(2) = 1$  と逆にたどることにより \_\_\_\_\_ が得られる. 路を求める過程を表で表すと以下ようになる.

|     | 1        | 2        | 3        | 4 | 5        | 6 |
|-----|----------|----------|----------|---|----------|---|
| (0) | <u>0</u> | 0        | 0        | 0 | 0        | 0 |
| (1) |          | <u>1</u> | 1        | 0 | 0        | 0 |
| (2) |          |          | <u>1</u> | 0 | 2        | 0 |
| (3) |          |          |          | 0 | <u>2</u> | 0 |
| (4) |          |          |          | 5 |          | 5 |

下線の引いてあるのは, 次の繰り返しにおいて選ばれる節点を示している.

[課題 3.3] 課題 3.2 の (2) において, ラベリング法を用いて増加路を求めよ.

[課題 3.4] さらに, 前課題で求めたフローの残余ネットワークを図示し, ラベリング法を用いて増加路を求めよ. さらに流量を増やすことは可能か?

### 3.3.4 フロー増加法の正当性と最大流最小カット定理

ラベリング法を用いたフロー増加法によって，最大流が求まることを示そう．

カット：節点集合  $V$  を，ソースを含む集合  $S$  と，シンクを含む集合  $T$  に分割したものをカットと呼び， $(S, T)$  と書く．カット  $(S, T)$  に対して， $i \in S, j \in T, (i, j) \in E$  のとき  $(i, j) \in (S, T)$  と書く．逆に， $i \in T, j \in S, (i, j) \in E$  のとき  $(i, j) \in (T, S)$  と書く．

カットの容量：全ての枝  $(i, j) \in (S, T)$  の容量  $u_{ij}$  の和をカット  $(S, T)$  の容量と呼び， $C(S, T)$  と書く．つまり，

$$C(S, T) = \sum_{(i, j) \in (S, T)} u_{ij}$$

$x$  を任意のフロー， $f$  をその流量， $(S, T)$  を任意のカットとする．すると，

$$C(S, T) = \sum_{(i, j) \in (S, T)} \text{——} \geq \sum_{(i, j) \in (S, T)} \text{——}$$

また，

$$f = \sum_{(i, j) \in (S, T)} x_{ij} - \sum_{(i, j) \in (T, S)} x_{ij} \leq \sum_{(i, j) \in (S, T)} x_{ij}$$

が成り立つ．したがって，

$$f \leq C(S, T)$$

言い替えれば，

$$\max_x f \leq \min_{(S, T)} C(S, T) \quad (3.7)$$

となる．

さて，ラベリング法を用いたフロー増加法が終了したとき，フロー  $x^*$ ，流量  $f^*$  が得られていたとする．また，ラベリング法の終了条件より，残余ネットワーク  $G^x = (V, E^x)$  に対して，ソースを含みシンクを含まない節点集合  $S^*$  が得られているはずである． $T^* = V \setminus S^*$  と定義すると， $(S^*, T^*)$  はカットとなる． $S^*$  は，残余ネットワークにおいて，ソースから到達可能な節点の集合であるから， $E^x$  には  $S^*$  内の節点から  $T^*$  内の節点への枝は存在しない．したがって，枝  $(i, j) \in E$  に対して，

- $(i, j) \in (S^*, T^*)$  ならば \_\_\_\_\_ である ( $x_{ij}^* < u_{ij}$  と仮定すると，残余ネットワークには  $u_{ij}^x = \text{——}$  の容量を持つ  $S^*$  から  $T^*$  への枝  $(i, j)$  が存在することになり，矛盾．)
- $(i, j) \in (T^*, S^*)$  ならば \_\_\_\_\_ である ( $x_{ji}^* > 0$  と仮定すると，残余ネットワークには  $u_{ji}^x = \text{——}$  の容量を持つ  $S^*$  から  $T^*$  への枝  $(j, i)$  が存在することになり，矛盾．)

したがって、フロー  $x^*$  の流量  $f^*$  は、

$$f^* = \sum_{(i,j) \in (S^*, T^*)} x_{ij}^* - \sum_{(i,j) \in (T^*, S^*)} x_{ij}^* = \sum_{(i,j) \in (S^*, T^*)} u_{ij} = C(S^*, T^*)$$

を満たしている。式 (3.7) より、 $f^*$  が最大流量であることがわかり、 $x^*$  は最大流となっている。

以上より、次の定理が成り立つ。

**定理 3.1 (最大流最小カット定理)** 任意のネットワークにおいて、フローの最大流量とカット容量の最小値は等しくなる。

### 3.3.5 フロー増加法の計算量と改良

フロー増加法の計算量について考えてみよう。 $n$ : 節点数、 $m$ : 枝数、 $U$ : 枝容量の最大値、枝容量は全て整数とすると、

- 一回のラベリング法の計算量は枝の本数に比例:  $O(m)$ 。
- フロー増加法の各繰返しで、少なくとも流量は 1 増える。また、最大流量は高々  $mU$ 。したがって、フロー増加法の繰返し回数は高々  $mU$ 。
- 全体では、 $O(m^2U)$  となる。

$U$  が非常に大きいとき、非効率である。改良方法としては、以下の方法がある。

- 複数のフロー増加路が存在する場合、必ず最も短い (枝数の少ない) 路を選ぶことにすると、フロー増加法の繰返し回数が  $mn/2$  以下になることが知られている。全体では  $O(m^2n)$  となり  $U$  によらず、多項式時間アルゴリズムとなる。
- 1 度に複数のフロー増加路を求めて、たくさんフローを増加させる。たとえば、ある繰返しでは枝数が 3 のフロー増加路をすべて求め、全ての増加路に沿ってフローを追加する。次の繰返しでは、枝数が 3 の増加路は存在しないので、今度は枝数 4 の増加路を全て求める。すると、繰返しの回数は高々節点数  $n$ 。複数の増加路を 1 度に求める計算量は  $O(mn)$  で実行可能であることが知られている。全体では  $O(mn^2)$  となる。

#### Q & A

Q. 「フロー増加法」には、考案者の名前のついた呼び方はないのですか？

A. あります！「フォード-ファルカーソン法」と呼ばれたりもします。