数理計画法 第8回目資料 — ネットワーク計画 その2

塩浦 昭義*

平成14年11月27日

3　最小費用フロー問題

3.1　定式化

入力:　有向グラフ $G = (V, E)$

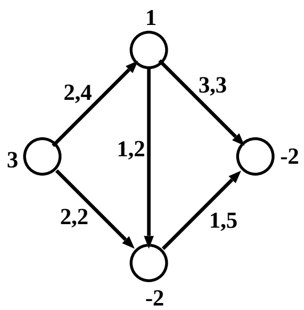
各頂点 $i \in V$ の**需要供給量** b_i (i は $b_i > 0$ のとき**供給点**, $b_i < 0$ のとき**需要点**)

ただし $\sum_{i \in V} b_i = 0$ を満たす ($\sum_{i \in V} b_i \neq 0$ のときは許容解は存在しない)

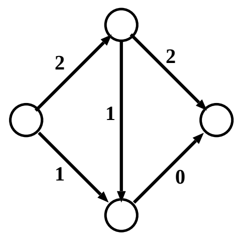
各枝 $(i, j) \in E$ の**容量** $u_{ij} (\geq 0)$, **コスト** c_{ij}

$$
\begin{array}{|lll}
\text{最小化} & \displaystyle\sum_{(i,j) \in E} c_{ij} x_{ij} & \\
\text{条件} & \displaystyle\sum_{j:(k,j) \in E} x_{kj} - \sum_{i:(i,k) \in E} x_{ik} = b_k \qquad (k \in V) & \cdots \quad (1) \\
 & 0 \leq x_{ij} \leq u_{ij} \quad ((i,j) \in E) & \cdots \quad (2)
\end{array}
$$

$(x_{ij} \mid (i, j) \in E)$ は**フロー** $\overset{\text{定義}}{\iff}$ $(x_{ij} \mid (i, j) \in E)$ は上記のＬＰの許容解



問題例 1. 各頂点の数値は需要供給量 b_i.
各枝の数値は (コスト c_{ij}, 容量 u_{ij}).



フローの例: コストは 13

★ 解説

変数 x_{ij}　—　枝 $(i, j) \in E$ を流れるフロー量

制約式 (1)　—　(頂点 k から出る枝のフロー量の総和) − (k に入る枝のフロー量の総和)
= (供給量) (もしくは − (需要量))

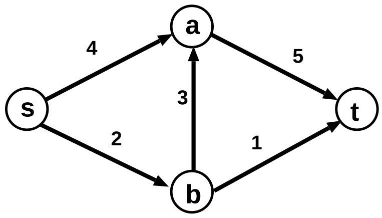
制約式 (2)　—　各枝を流れるフローは非負かつ容量を越えてはならない

※ 最小費用フロー問題は線形計画問題の特殊ケース ⟹ 単体法で解くことができる!

*東北大学大学院 情報科学研究科 shioura@dais.is.tohoku.ac.jp

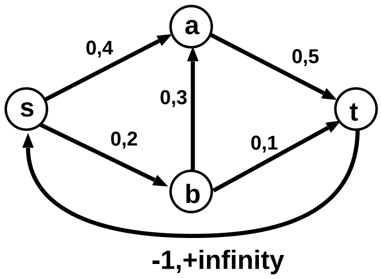
3.2 最大フロー問題との関係

- **最大フロー問題は最小費用フロー問題の特殊ケース**



最大フロー問題の例

$\Longrightarrow$



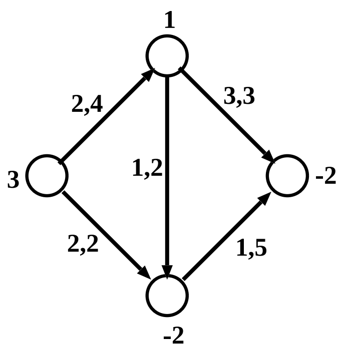
最小費用フロー問題への変換

最大フロー問題のグラフの供給点 s から需要点 t への枝 (t, s) を付加.
枝 (t, s) のコストは -1, 容量は十分大きな値とする.
他の枝については, 容量はそのまま, コストを 0 とする.

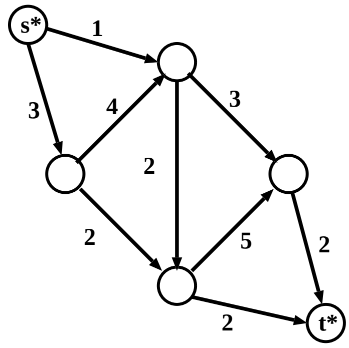
★ 2 つの問題の対応関係

最大フロー問題の許容解	$=$	最小費用フロー問題の許容解 から x_{ts} を除いたもの
フローの流量	$=$	枝 (t, s) のフローの値 $=$ フローのコスト $\times$ ($-$ 1)
フローをたくさん流す	$\Longleftrightarrow$	フローのコストを小さくする

- **最小費用フロー問題の許容解は最大フロー問題を解いて得られる**



問題例 1



許容解を求めるための人工問題 (最大フロー問題)

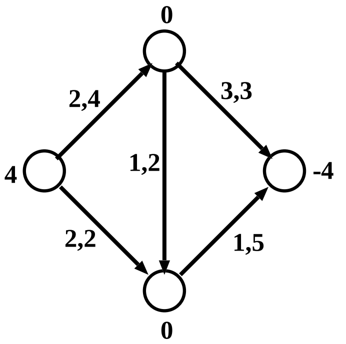
★ 許容解を求めるための人工問題の構成方法

(i): 元のグラフ $G = (V, E)$ に新たな頂点 s^*, t^* を追加.
(ii): 各供給点 $j \in V$ に対して枝 (s^*, j) を追加, 容量は b_j とする.
(iii): 各需要点 $i \in V$ に対して枝 (i, t^*) を追加, 容量は $-b_i$ とする.
(iv): 元々の枝の容量は元のまま.

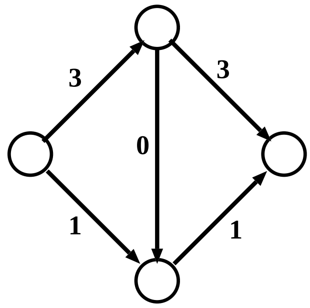
★ 2 つの問題の対応関係

最大フローの流量が $\sum\{b_j \mid b_j > 0\}$ $(= \sum\{-b_i \mid b_i < 0\})$ に等しい
$\Longleftrightarrow$ 枝 (s^*, j) のフロー量は b_j, 枝 (i, t^*) のフロー量は $-b_i$ に等しい
$\Longleftrightarrow$ 元のグラフ上のフローは各頂点の需要供給量の制約を満たす
$\Longleftrightarrow$ 元のグラフ上のフローは最小費用フロー問題の許容解

3.3 残余ネットワークと負閉路消去法



問題例 2



フロー $\boldsymbol{x}$. コストは 18. 最適か？

どうやって最小費用フローであることを判定する? — **残余ネットワーク**の利用

★ 残余ネットワークの構築法

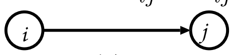
$\boldsymbol{x}$: 現在のフロー $\Longrightarrow$

x に関する残余ネットワーク $G^x = (V, E^x)$, $E^x = F^x \cup R^x$

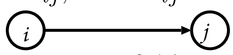
$F^x = \{(i,j) \mid (i,j) \in E, x_{ij} < u_{ij}\}$, F^x の各枝 (i,j) のコスト $c^x_{ij} = c_{ij}$, 容量 $u^x_{ij} = u_{ij} - x_{ij}$

$R^x = \{(j,i) \mid (i,j) \in E, x_{ij} > 0\}$, R^x の各枝 (j,i) のコスト $c^x_{ji} = -c_{ij}$, 容量 $u^x_{ji} = x_{ij}$

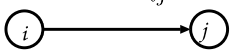
フロー量 $x_{ij} < u_{ij}$



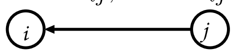
$\Longrightarrow$ 同じ向き, コスト c_{ij}, 容量 $u_{ij} - x_{ij}$ の枝を加える



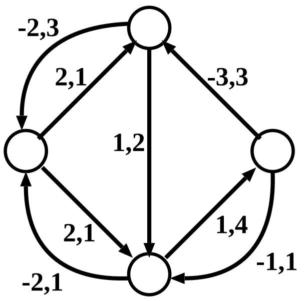
フロー量 $x_{ij} > 0$



$\Longrightarrow$ 逆の向き, コスト $-c_{ij}$, 容量 x_{ij} の枝を加える



残余ネットワークの例



残余ネットワーク (各枝の数値: コスト, 容量)

残余ネットワークにおいて閉路が存在 $\Longrightarrow$ 現在のフロー $\boldsymbol{x}$ の流量を変更することが可能

★ フローの更新方法

$\boldsymbol{x}$: 現在のフロー, $C \subseteq E^x$: 残余ネットワーク上の閉路

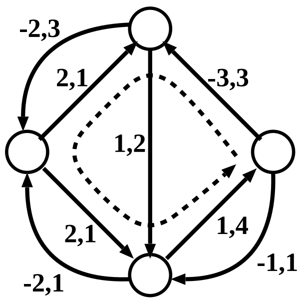
$\alpha = \min\{u^x_{ij} \mid (i,j) \in C\}$ (閉路 C 上の枝の容量の最小値, **閉路の容量**)

$\gamma = \sum\{c^x_{ij} \mid (i,j) \in C\}$ (閉路 C 上の枝のコストの和, **閉路のコスト**)

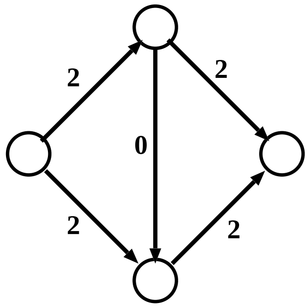
元のグラフの各枝 $(i,j) \in E$ に対し, $\begin{cases} (i,j) \in C \cap F^x & \Longrightarrow \quad x_{ij} := x_{ij} + \alpha \\ (j,i) \in C \cap R^x & \Longrightarrow \quad x_{ij} := x_{ij} - \alpha \\ \text{それ以外} & \Longrightarrow \quad x_{ij} \text{ は不変} \end{cases}$

$\Longrightarrow$ フロー $\boldsymbol{x}$ が変化, 新しいフローのコスト ＝ 元のフローのコスト ＋ $\gamma\alpha$

残余ネットワークにおいてコストが負の閉路が存在 $\Longrightarrow$ 現在のフロー $\boldsymbol{x}$ のコストを減らすことが可能



$\Longrightarrow$



容量が $\min\{3,3,1,4\}=1$, コストが $(-3)+(-2)+2+1=-2$ の閉路が存在.

更新後のフロー量.
フローのコストは 18 から 16 へ減少.

以上のアイデアをもとに最小費用フローを求めるアルゴリズムを構築 $\Longrightarrow$ 負閉路消去法

★ 負閉路消去法

ステップ 0: 人工問題を作って許容解 $\boldsymbol{x}$ を求める.
ステップ 1: $\boldsymbol{x}$ に関する残余ネットワーク $G^x=(V,E^x)$ を構築する.
ステップ 2: G^x における コストが負の閉路 C を求める.
存在しなければ現在のフロー $\boldsymbol{x}$ は最小費用フローなので終了.
ステップ 3: ステップ 2 で求めた閉路 C を用いてフロー $\boldsymbol{x}$ を更新.
ステップ 4: ステップ 1 へ戻る.

3.4 負閉路消去法の正当性

次のことを示せば良い:

現在のフローのコストが最小でない $\Longrightarrow$ 残余ネットワークにコストが負の閉路が存在する

フロー $\boldsymbol{x}$ に関する残余ネットワーク $G^x=(V,E^x)$ を考える.

- $(y_{ij}\mid(i,j)\in E^x)$ は残余ネットワーク G^x における**循環フロー** $\overset{\text{定義}}{\Longleftrightarrow}$ 下記の条件を満たす:

$$\sum_{j:(k,j)\in E^x} y_{kj}-\sum_{i:(i,k)\in E^x} y_{ik}=0 \quad (k\in V), \qquad 0\le y_{ij}\le u^x_{ij} \quad ((i,j)\in E^x)$$

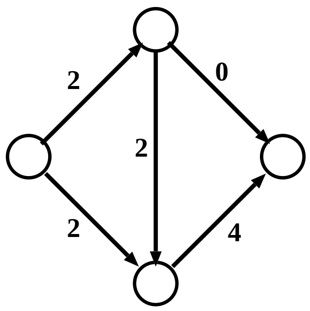
補題 1: 任意のフロー $\boldsymbol{x'}$ に対し, $(y_{ij}\mid(i,j)\in E^x)$ を次のように定義する:

$$y_{ij}=\begin{cases} x'_{ij}-x_{ij} & ((i,j)\in E^x,\ x'_{ij}-x_{ij}>0),\\ x_{ji}-x'_{ji} & ((i,j)\in E^x,\ x'_{ji}-x_{ji}<0),\\ 0 & (\text{その他の } (i,j)\in E^x). \end{cases} \tag{1}$$

このとき, $(y_{ij}\mid(i,j)\in E^x)$ は残余ネットワークにおける循環フローであり, かつそのコストは $\boldsymbol{x'}$ のコストと $\boldsymbol{x}$ のコストの差に等しい:

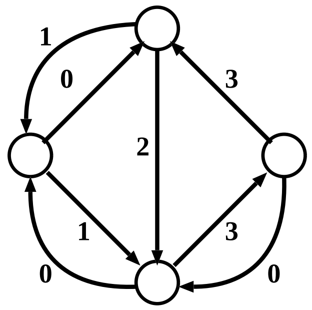
$$\sum_{(i,j)\in E^x} c^x_{ij}y_{ij}=\sum_{(i,j)\in E} c_{ij}x'_{ij}-\sum_{(i,j)\in E} c_{ij}x_{ij}.$$

証明: 残余ネットワークの定義及び式 (1) を使って簡単に証明できる. □



問題例 2 のフロー $\boldsymbol{x'}$

$\Longrightarrow$



3 ページ右上のフロー $\boldsymbol{x}$ に対して, 式 (1) により求めた $(y_{ij} \mid (i,j) \in E^x)$. 3 ページ真ん中の補助ネットワークにおける循環フローになっていることを確認されたい.

補題 2: 残余ネットワークにおける循環フロー $(y_{ij} \mid (i,j) \in E^x)$ は閉路上のフローの和として表せる. すなわち, 残余ネットワーク上の閉路 $C_1, C_2, \ldots, C_k$ および正の値 $\alpha_1, \alpha_2, \ldots, \alpha_k$ が存在して,

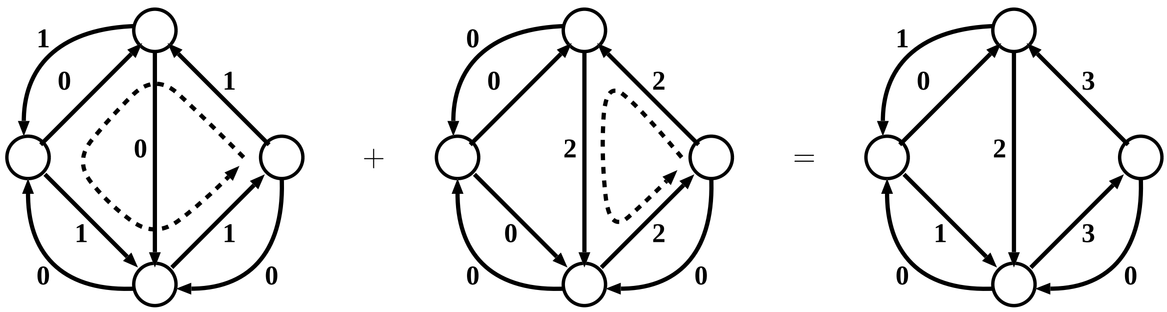
$$y_{ij} = y_{ij}^{(1)} + y_{ij}^{(2)} + \cdots + y_{ij}^{(k)} \qquad ((i,j) \in E^x)$$

が成り立つ. ここで, $(y_{ij}^{(h)} \mid (i,j) \in E^x)$ $(h = 1, 2, \ldots, k)$ は次の式により与えられる:

$$y_{ij}^{(h)} = \begin{cases} \alpha_h & ((i,j) \in E^x \cap C_h), \\ 0 & (\text{その他の } (i,j) \in E^x). \end{cases} \tag{2}$$

証明: 略. □

右上の循環流は次のように閉路上のフローの和として表せる.



定理 3 [負閉路による最適性条件]: 最小費用フロー問題のフロー $\boldsymbol{x}$ は最適解 $\Longleftrightarrow$ $\boldsymbol{x}$ に関する残余ネットワークにおいてコストが負の閉路が存在しない.

証明: [$\Longrightarrow$] 対偶を示す. $\boldsymbol{x}$ に関する残余ネットワークにおいてコストが負の閉路が存在するならば, 先に説明したやり方でフローを変更し, コストを減少させることが出来る. したがって, フロー $\boldsymbol{x}$ は最適解ではない.

[$\Longrightarrow$] 対偶を示す. フロー $\boldsymbol{x}$ は最適解ではないと仮定し, x' を最適フローとおく. この二つのフローに対して, 式 (1) を用いて $(y_{ij} \mid (i,j) \in E^x)$ を定義すると, 補題 1 よりこれは $\boldsymbol{x}$ に関する残余ネットワークの循環フローとなる. また, 補題 2 により, すなわち, 残余ネットワーク上の閉路 $C_1, C_2, \ldots, C_k$ およ

び正の値 $\alpha_1, \alpha_2, \ldots, \alpha_k$ が存在して, 式 (2) で定義される閉路上のフロー $(y_{ij}^{(h)} \mid (i,j) \in E^x)$ により

$$y_{ij} = y_{ij}^{(1)} + y_{ij}^{(2)} + \cdots + y_{ij}^{(k)} \qquad ((i,j) \in E^x)$$

が成り立つ. さらに, 補題 1, 2 より

$$\begin{aligned}
&\sum_{h=1}^{k} \alpha_h \times [\text{閉路 } C_h \text{ のコスト}] \\
&= \sum_{h=1}^{k} [\text{循環フロー } (y_{ij}^{(h)} \mid (i,j) \in E^x) \text{ のコスト}] \\
&= [\text{循環フロー } (y_{ij} \mid (i,j) \in E^x) \text{ のコスト}] \\
&= [\text{フロー } (x'_{ij} \mid (i,j) \in E) \text{ のコスト}] - [\text{フロー } (x_{ij} \mid (i,j) \in E) \text{ のコスト}] \ < \ 0.
\end{aligned}$$

したがって, 少なくともひとつの閉路 C_h について, そのコストは負となる. □

3.5 負閉路消去法の反復回数とアルゴリズムの改良

n : 頂点数, m : 枝数, $C = \max\{|c_{ij}| \mid (i,j) \in E\}$, $U = \max\{u_{ij} \mid (i,j) \in E\}$
※ 以下, 枝のコストと容量は整数値と仮定

★負閉路消去法の反復回数

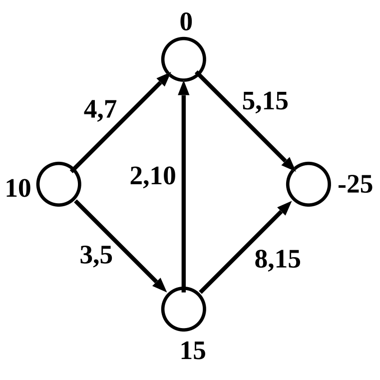
- 負閉路消去法は各反復でコストを少なくとも 1 減少させる
- フローのコストの絶対値は mCU 以下 ($-mCU \leq$ フローのコスト $\leq mCU$)

$\Longrightarrow$ 反復回数は高々 $2mCU$ — 入力のビット長 $n, m, \log C, \log U$ に関する多項式ではない!

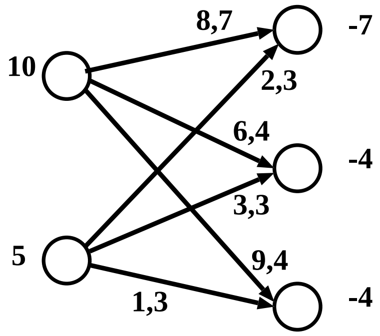
★ 反復回数を減らすには — ステップ 2 で「特別な」閉路を求める

- その 2: コストの減少量を最大にする閉路 (**最有益閉路**) $\Longrightarrow$ 反復回数は $\mathrm{O}(m \log(nCU))$
- その 2: [コスト]/[枝数] の値を最小にする閉路 (**最小平均閉路**) $\Longrightarrow$ 反復回数は $\mathrm{O}(nm \log(nC))$
  $\Longrightarrow$ さらに $\mathrm{O}(nm^2 \log n)$ へと改良することも可能 (**強多項式時間アルゴリズム**)
- その他, 様々な多項式時間アルゴリズムが存在

レポート問題　下記の最小費用フロー問題を負閉路消去法を用いて解け.



問題 1



問題 2