

経営・経済のための最適化理論

第9回

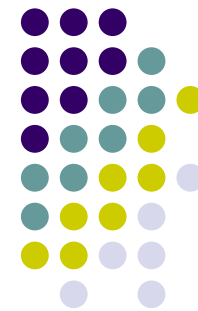
最大流問題

塩浦昭義

東京工業大学 経営工学系

shioura.a.aa@m.titech.ac.jp

様々な条件を満たす フローを求める



これまで扱ってきたフローの条件

- 容量条件: フローは非負, 上界値以下)
- 流量保存条件: 頂点から出て行くフローの総和
= 入るフローの総和

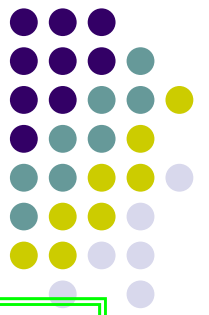
より一般的な条件

- 容量条件: フローは下界値以上
- 需要供給条件:
頂点から出て行くフローの総和
− 入るフローの総和 = (ゼロとは限らない) 定数

このような条件を満たすフローの

存在判定および計算(存在の場合)は, 最大流問題に帰着可能

需要供給条件と上下界条件を満たすフローを求める問題



入力： 有向グラフ $G = (V, E)$

各枝 $(i, j) \in E$ のフローの**下界と上界** $\ell_{ij}, u_{ij} \geq 0$

各頂点 $i \in V$ の**供給・需要量** b_i (ただし b_i の和は0)

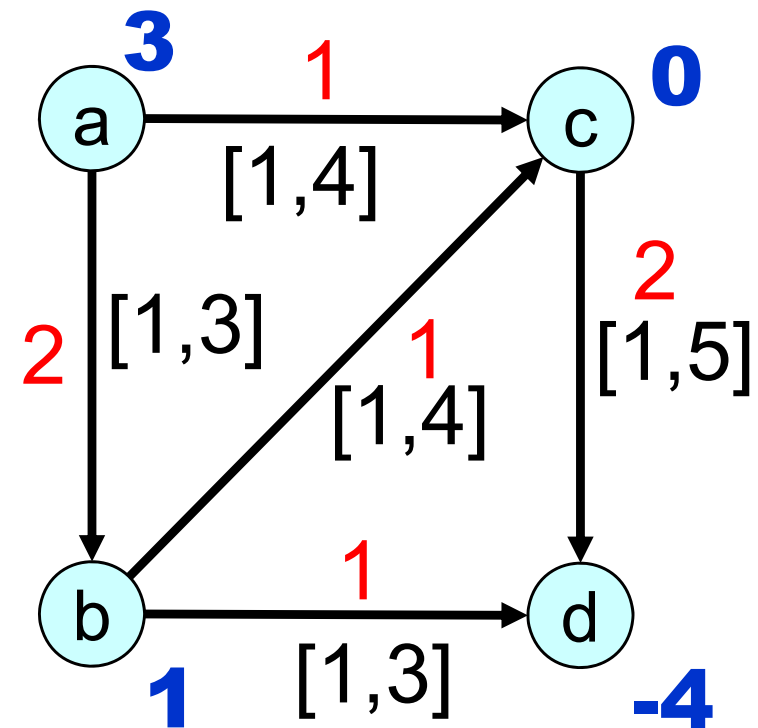
出力： 次の条件を満たすフロー
(もしくは「解なし」と答える)

- 各頂点 $i \in V$ での**供給・需要条件**
(i から流出するフロー量)

$$-(i \text{ に流入するフロー量}) = b_i$$

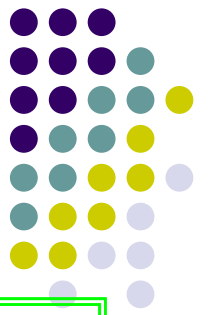
- 各枝 (i, j) の**容量条件**

$$\ell_{ij} \leq x_{ij} \leq u_{ij}$$



条件を満たすフローが存在

需要供給条件と上下界条件を満たすフローを求める問題



入力： 有向グラフ $G = (V, E)$

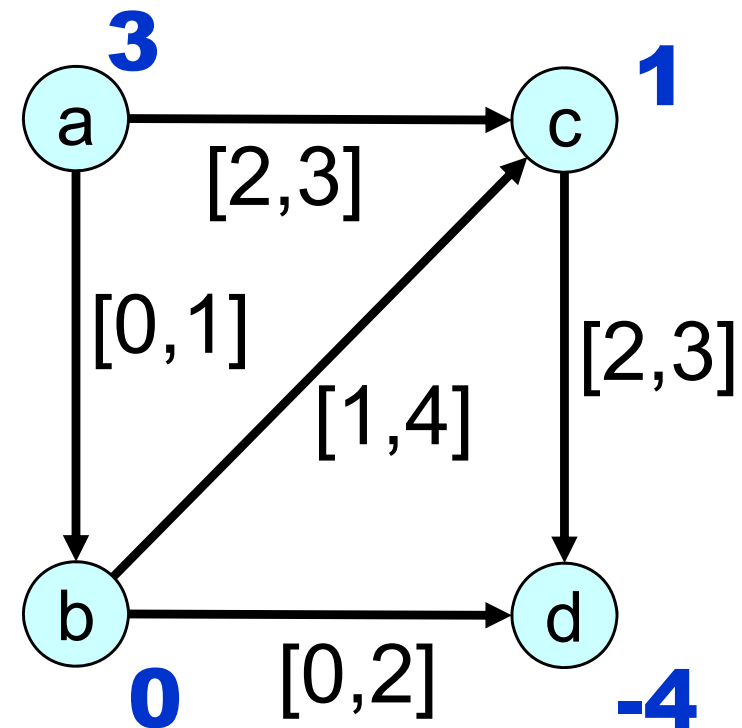
各枝 $(i, j) \in E$ のフローの**下界と上界** $\ell_{ij}, u_{ij} \geq 0$

各頂点 $i \in V$ の**供給・需要量** b_i (ただし b_i の和は0)

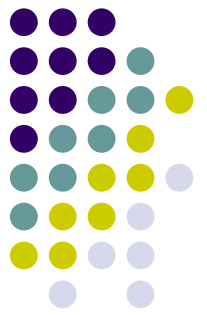
出力： 次の条件を満たすフロー
(もしくは「解なし」と答える)

- 各頂点 $i \in V$ での**供給・需要条件**
(i から流出するフロー量)
− (i に流入するフロー量) = b_i

- 各枝 (i, j) の**容量条件**
 $\ell_{ij} \leq x_{ij} \leq u_{ij}$

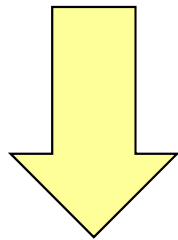


条件を満たすフローは
存在しない

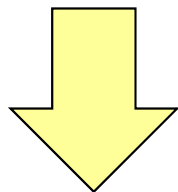


帰着の手順

需要供給条件と上下界条件を満たすフローを求める問題



需要供給条件と上界条件
(下界値=0)を満たすフロー
を求める問題

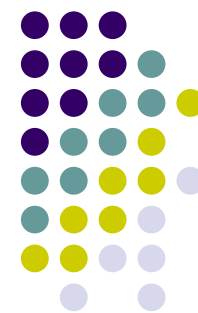


最大流問題

- 各頂点 $i \in V$ での**供給・需要条件**
(i から流出するフロー量)
—(i に流入するフロー量) $= b_i$
- 各枝 (i, j) の**容量条件**
$$\ell_{ij} \leq x_{ij} \leq u_{ij}$$

- 各頂点 $i \in V$ での**供給・需要条件**
(i から流出するフロー量)
—(i に流入するフロー量) $= b_i$
- 各枝 (i, j) の**容量条件**
$$0 \leq x_{ij} \leq u_{ij}$$

供給・需要条件を満たすフロー の計算への帰着



アイデア：変数 x_{ij} を $y_{ij} = x_{ij} - \ell_{ij}$ に置き換えた問題を考える

- 容量条件 $\ell_{ij} \leq x_{ij} \leq u_{ij} \rightarrow 0 \leq y_{ij} \leq u_{ij} - \ell_{ij}$

- 需要供給条件

$$\sum_{(i,j):i \text{ から出る枝}} x_{ij} - \sum_{(k,i):i \text{ に入る枝}} x_{ki} = b_i$$

$$\rightarrow \sum_{(i,j):i \text{ から出る枝}} (y_{ij} + \ell_{ij}) - \sum_{(k,i):i \text{ に入る枝}} (y_{ki} + \ell_{ki}) = b_i$$

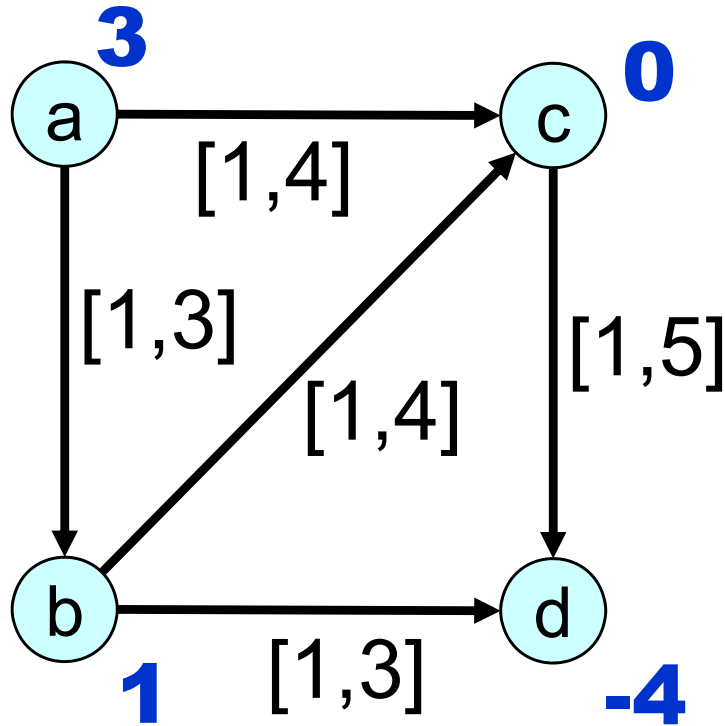
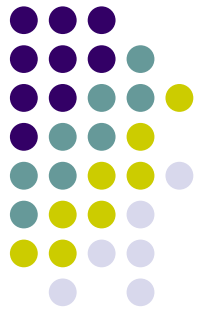
$$\rightarrow \sum_{(i,j):i \text{ から出る枝}} y_{ij} - \sum_{(k,i):i \text{ に入る枝}} y_{ki} \\ = b_i - \sum_{(i,j):i \text{ から出る枝}} \ell_{ij} + \sum_{(k,i):i \text{ に入る枝}} \ell_{ki}$$

∴ 下界が0 のフローを求める問題に帰着できた.

よって, 上記の条件を満たす y_{ij} を計算

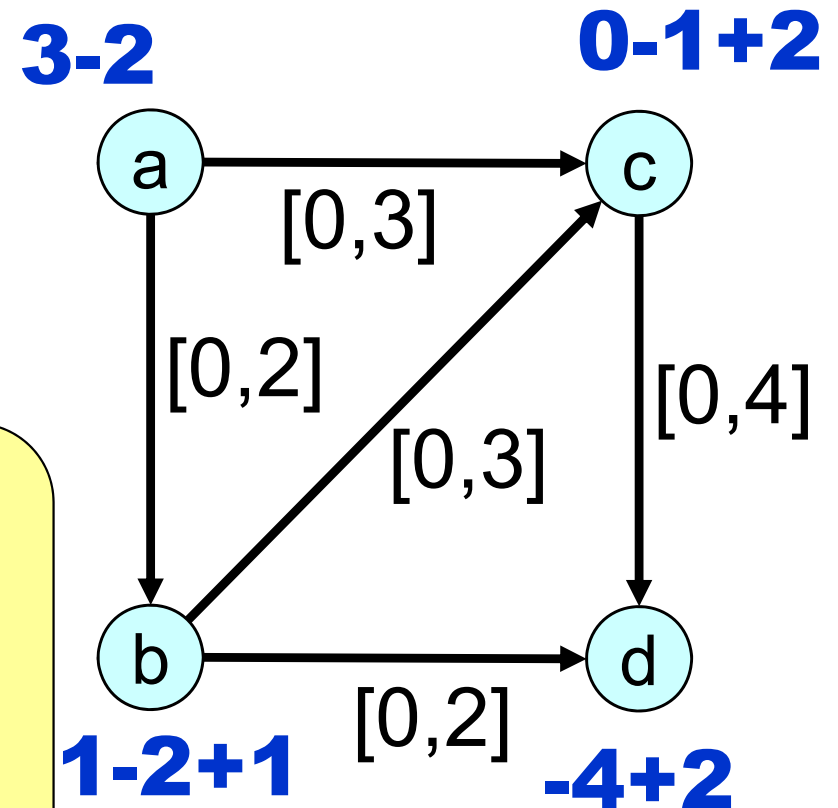
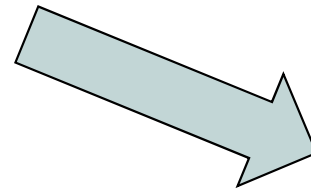
$\rightarrow x_{ij} = y_{ij} + \ell_{ij}$ とおけば, 元の問題のフローが得られる.

帰着の具体例



$$0 \leq y_{ij} \leq u_{ij} - \ell_{ij}$$

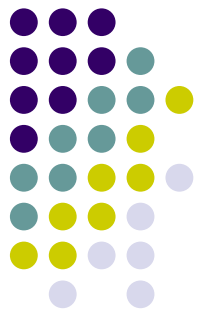
$$\sum_{(i,j): i \text{ から出る枝}} y_{ij} - \sum_{(k,i): i \text{ に入る枝}} y_{ki} = b_i - \sum_{(i,j): i \text{ から出る枝}} \ell_{ij} + \sum_{(k,i): i \text{ に入る枝}} \ell_{ki}$$



イメージ:

各枝において,
下界値の分だけフローを無理矢理流す
→ 各頂点で貯まった・不足した量を±0
にするために, フローをさらに流す

供給・需要を満たすフローを求める



入力： 有向グラフ $G = (V, E)$

各枝 $(i, j) \in E$ の容量 $u_{ij} \geq 0$

各頂点 $i \in V$ の供給・需要量 b_i (ただし b_i の和は0)

($b_i > 0 \rightarrow i$ は供給点, $< 0 \rightarrow i$ は需要点, $= 0 \rightarrow i$ は通過点)

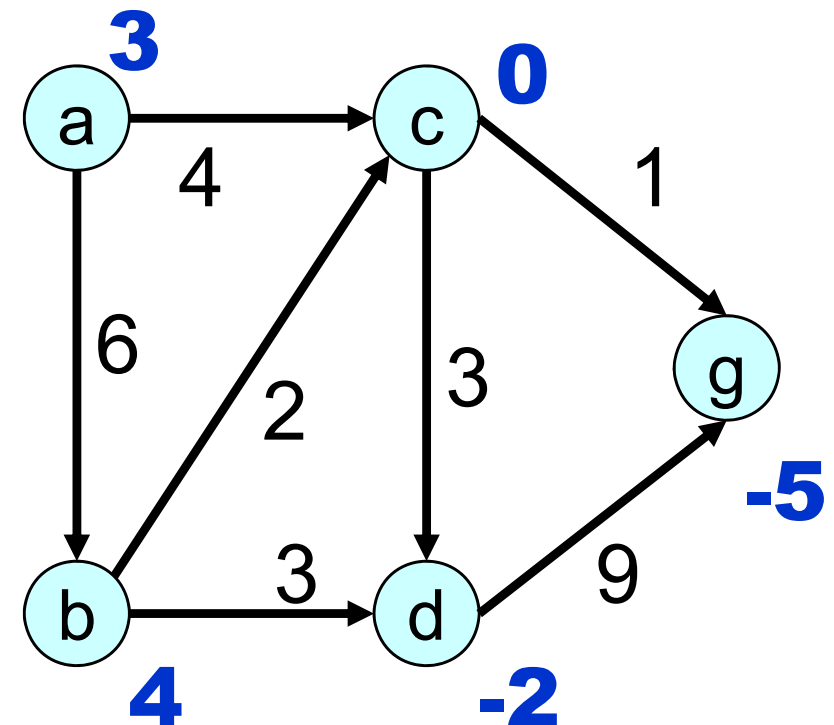
出力： 次の条件を満たすフロー
(もしくは「解なし」と答える)

- 各頂点 $i \in V$ での供給・需要条件
(i から流出するフロー量)

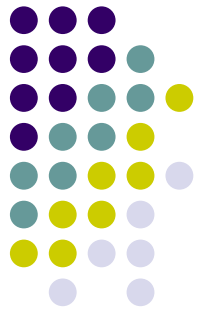
$$-(i \text{ に流入するフロー量}) = b_i$$

- 各枝 (i, j) の容量条件

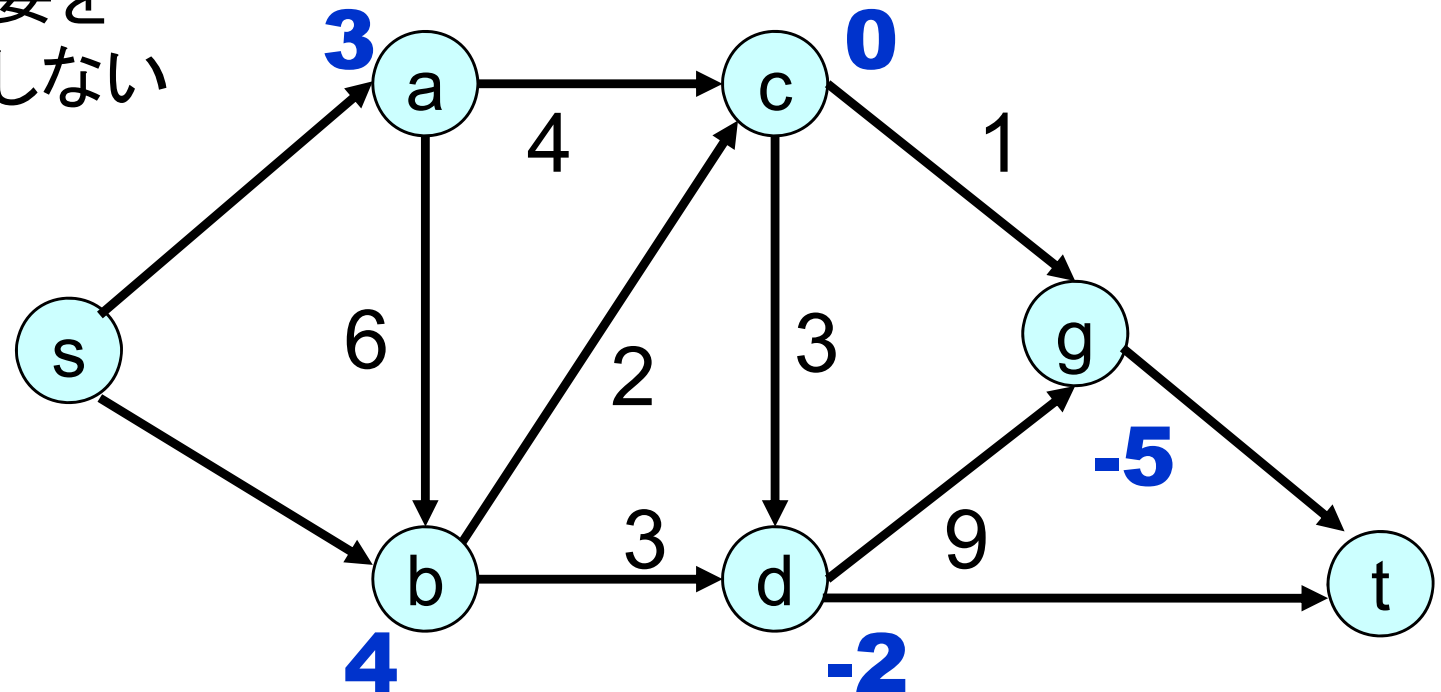
$$0 \leq x_{ij} \leq u_{ij}$$

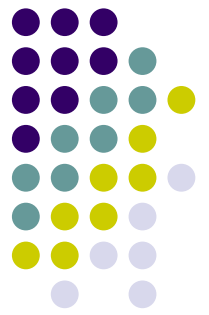


最大流問題への帰着



- (1) 新たな頂点 s (ソース), t (シンク) を追加
- (2) $b_i > 0$ ならば枝 (s, i) を追加, 容量は $+b_i$
- (3) $b_i < 0$ ならば枝 (i, t) を追加, 容量は $-b_i$
- (4) 最大フローを求める.
- (5) 各枝 (s, i) に対し $x_{si} = b_i \rightarrow$ 供給・需要を満たすフローが得られる
それ以外 $\rightarrow$ 供給・需要を満たすフローは存在しない





フローの整数性に関する性質

定理: 需要供給量 b_i と上下界 u_{ij}, ℓ_{ij} が整数

→ 全ての条件を満たすフローが存在するとき,
整数値フローが存在

(証明) 最大流問題に関する整数性を使う

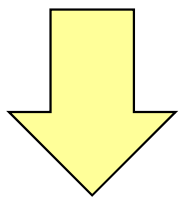
定理: 枝容量がすべて整数のとき,
整数値の最大フローが存在



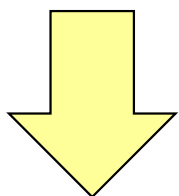
フローの整数性に関する性質

(証明)

需要供給条件と
上下界条件を満たす
フローを求める



需要供給条件と
上界条件を満たす
フローを求める



最大流問題

整数値フローが
存在

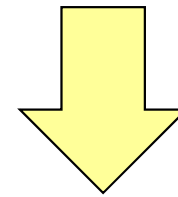


整数値フローが
存在

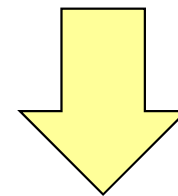


整数値フローが
存在

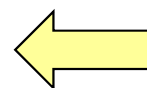
需要供給量 b_i と
上下界 u_{ij}, ℓ_{ij} が
整数



需要供給量 $b_i - \sum \ell_{ij} + \sum \ell_{ki}$
と上界 $u_{ij} - \ell_{ij}$ が整数



上界が整数





応用：数値表のデータを丸める

Ahuja, Magnanti, Orlin: Network Flows (1993)

	X	Y	Z	行の 合計
A	3.1	6.8	7.3	17.2
B	9.6	2.4	0.7	12.7
C	3.6	1.2	6.5	11.3
列の 合計	16.3	10.4	14.5	

各データを
切り上げ・切り捨てにより
整数に丸めたい

	X	Y	Z	行の 合計
A	3	7	7	17
B	10	2	1	13
C	4	1	7	11
列の 合計	16	10	15	

行・列の合計に関して
つじつまの合う丸めは可能か？
単なる四捨五入では駄目



応用：数値表のデータを丸める

Ahuja, Magnanti, Orlin: Network Flows (1993)

	X	Y	Z	行の 合計
A	3.1	6.8	7.3	17.2
B	9.6	2.4	0.7	12.7
C	3.6	1.2	6.5	11.3
列の 合計	16.3	10.4	14.5	

各データを
切り上げ・切り捨てにより
整数に丸めたい

	X	Y	Z	行の 合計
A	3	7	7	17
B	10	2	1	13
C	3	1	7	11
列の 合計	16	10	15	

行・列の合計に関して
つじつまの合う丸めは可能か？
それほど簡単には分らない

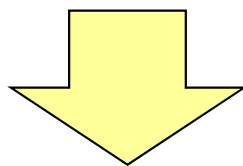
数値表のデータを丸める： 一般の場合



Ahuja, Magnanti, Orlin: Network Flows (1993)

$m \times n$ 個の実数値データ a_{ij} ($i = 1, 2, \dots, m, j = 1, 2, \dots, n$)

$b_i = \sum_{j=1}^n a_{ij}$ 第*i*行のデータの和, $c_j = \sum_{i=1}^m a_{ij}$ 第*j*列のデータの和



求めたいもの:

$m \times n$ 個の整数値 $\tilde{a}_{ij}$ ($i = 1, 2, \dots, m, j = 1, 2, \dots, n$)

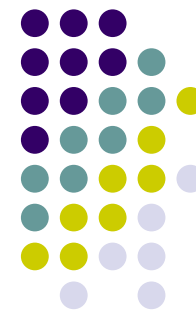
$\tilde{b}_i = \sum_{j=1}^n \tilde{a}_{ij}$ 第*i*行の数値の和, $\tilde{c}_j = \sum_{i=1}^m \tilde{a}_{ij}$ 第*j*列の数値の和

満たすべき条件:

$\tilde{a}_{ij} = \lfloor a_{ij} \rfloor$ または $\lceil a_{ij} \rceil$, $\tilde{b}_i = \lfloor b_i \rfloor$ または $\lceil b_i \rceil$, $\tilde{c}_j = \lfloor c_j \rfloor$ または $\lceil c_j \rceil$

定理: このような「丸め」は常に存在

(証明) 需要供給条件, 上下界条件を満たすフローの問題へ帰着.

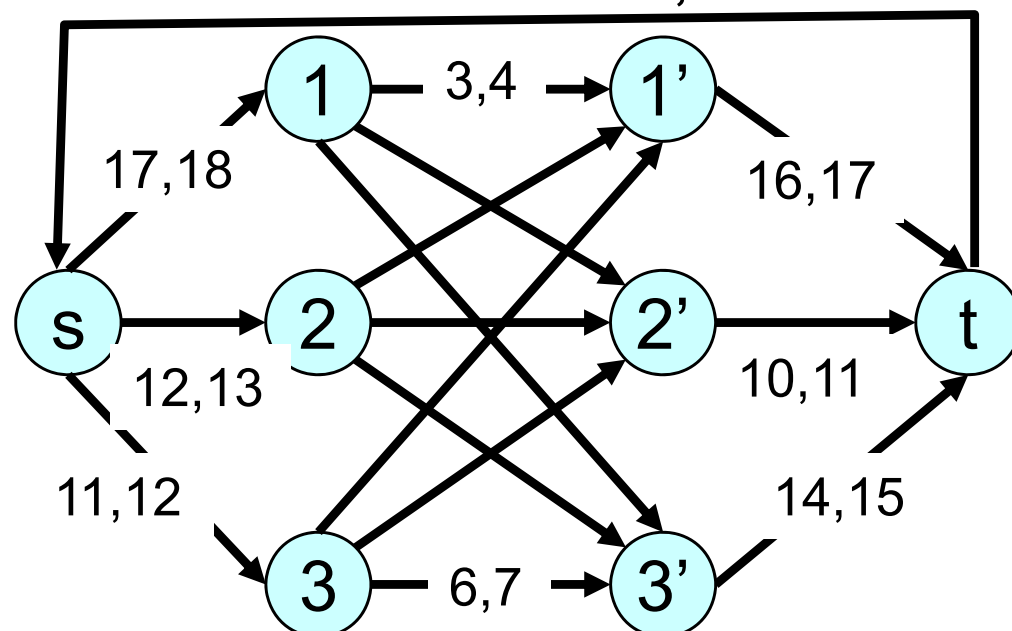


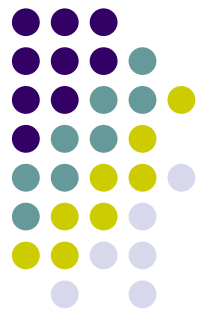
帰着に使うグラフ

頂点集合 $\{1, 2, \dots, m\} \cup \{1', 2', \dots, n'\} \cup \{s, t\}$

枝集合 $\{(i, j') \mid i = 1, 2, \dots, m, j = 1, 2, \dots, n\}$ $[a_{ij}] \leq x_{ij'} \leq [a_{ij}]$
 $\cup \{(s, i) \mid i = 1, 2, \dots, m\}$ $[b_i] \leq x_{si} \leq [b_i]$
 $\cup \{(j', t) \mid j = 1, 2, \dots, n\}$ $[c_j] \leq x_{j't} \leq [c_j]$
 $\cup \{(t, s)\}$ $[\sum_{i,j} a_{ij}] \leq x_{ts} \leq [\sum_{i,j} a_{ij}]$

	X	Y	Z	行の 合計
A	3.1	6.8	7.3	17.2
B	9.6	2.4	0.7	12.7
C	3.6	1.2	6.5	11.3
列の 合計	16.3	10.4	14.5	





丸めの存在の証明

頂点集合 $\{1, 2, \dots, m\} \cup \{1', 2', \dots, n'\} \cup \{s, t\}$

枝集合 $\{(i, j') | i = 1, 2, \dots, m, j = 1, 2, \dots, n\}$ $\lfloor a_{ij} \rfloor \leq x_{ij'} \leq \lceil a_{ij} \rceil$
 $\cup \{(s, i) | i = 1, 2, \dots, m\}$ $\lfloor b_i \rfloor \leq x_{si} \leq \lceil b_i \rceil$
 $\cup \{(j', t) | j = 1, 2, \dots, n\}$ $\lfloor c_j \rfloor \leq x_{j't} \leq \lceil c_j \rceil$
 $\cup \{(t, s)\}$ $\lfloor \sum_{i,j} a_{ij} \rfloor \leq x_{ts} \leq \lceil \sum_{i,j} a_{ij} \rceil$

元の実数値データは、全ての条件を満たすフローを与える

∴ 全ての条件を満たす整数値のフローが存在

これは、「丸め」に対応 ■