最適化 2 (第 6 回)

番原睦則 (banbara@i.nagoya-u.ac.jp)

名古屋大学情報学研究科

連絡

期末テスト

- 2020 年 1 月 31 日 (金) の講義中
- 60 分テスト，70 点満点
- 教科書 3.1, 3.2, 5.1, 5.2, 5.3
- 講義スライドの内容

2 月 7 日の部屋の変更

- 工学部３号館３２１講義室

概要

ネットワーク計画

簡単に言うと，重み付きグラフ上での最適化

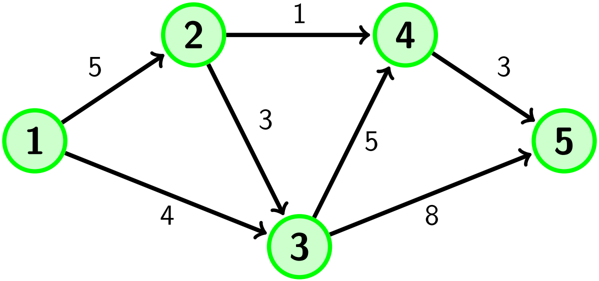
- ネットワーク計画問題のなかには，線形計画問題として定式化できる問題もある．
- 問題のネットワーク構造を利用することにより，効率の良いアルゴリズムを構成できる場合がある．

ネットワーク計画問題の代表例

- 最短路問題
- **最大流問題**
- 最小費用流問題

最大流問題

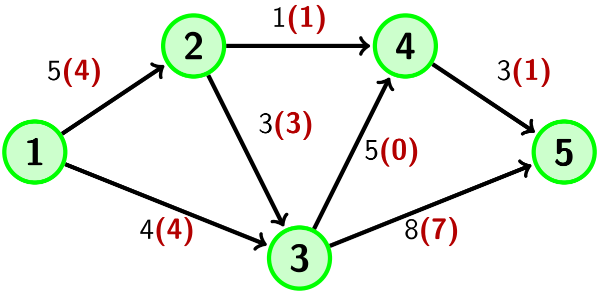
ネットワーク $G = (V, E)$ の各枝 $(i, j) \in E$ が容量 u_{ij} をもつとき，ある節点 $s \in V$(ソース) から別の節点 $t \in V$(シンク) への**流量** f を最大にする**フロー** * $\boldsymbol{x} = (x_{ij})$ を求める問題



*最大流

最大流問題

ネットワーク $G = (V, E)$ の各枝 $(i, j) \in E$ が容量 u_{ij} をもつとき，ある節点 $s \in V$(ソース) から別の節点 $t \in V$(シンク) への**流量** f を最大にする**フロー**[*] $\boldsymbol{x} = (x_{ij})$ を求める問題



- $s = 1$, $t = 5$ の最適フローの例．この時の流量は $f = 8$

[*] 最大流

最大流問題の定式化

線形計画問題としての定式化

目的関数: $f \longrightarrow$ 最大

制約条件:

$$\sum_{\{j \mid (s,j)\in E\}} x_{sj} - \sum_{\{j \mid (j,s)\in E\}} x_{js} = f$$

$$\sum_{\{j \mid (i,j)\in E\}} x_{ij} - \sum_{\{j \mid (j,i)\in E\}} x_{ji} = 0 \qquad (i \in V\backslash\{s,t\})$$

$$\sum_{\{j \mid (t,j)\in E\}} x_{tj} - \sum_{\{j \mid (j,t)\in E\}} x_{jt} = -f$$

$$0 \le x_{ij} \le u_{ij} \qquad (i,j) \in E$$

- 制約条件を満たす $\boldsymbol{x} = (x_{ij})$ を**フロー**と呼び，それに対する f を**流量**という．
- シンプレックス法などでも解けるが，ネットワーク構造を利用した効率的なアルゴリズムが知られている．

残余ネットワーク

残余ネットワーク $G^x = (V, E^x)$

あるフロー $\boldsymbol{x} = (x_{ij})$ が与えられたとき，ネットワーク $G = (V, E)$ の各枝 (i, j) を

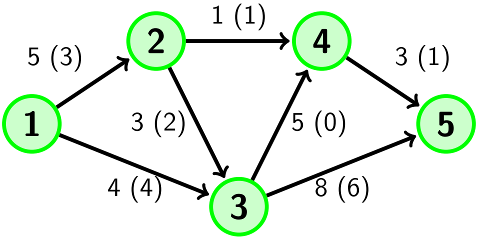
- 容量 $\boldsymbol{u^x_{ij}} = u_{ij} - x_{ij}$ をもつ枝 (i, j)
- 容量 $\boldsymbol{u^x_{ji}} = x_{ij}$ をもつ枝 (j, i)

の 2 つの枝で置き換えたネットワークを，フロー $\boldsymbol{x}$ に対する**残余ネットワーク**といい，$G^x = (V, E^x)$ で表す．

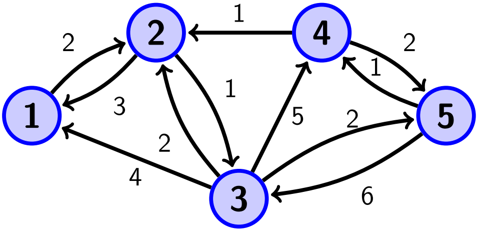
- u^x_{ij} および u^x_{ji} をフロー $\boldsymbol{x}$ に対する枝 (i, j) の**残余容量**という．
- 残余容量が 0 になる枝は省略．

残余ネットワークの例

- ネットワークとフロー (括弧内の数字はフロー)



- フローに対する残余ネットワーク



フロー増加路

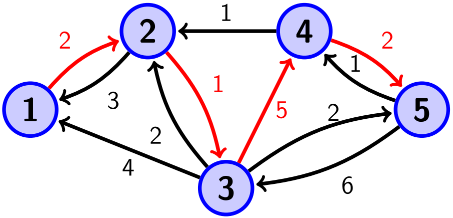
フロー増加路

残余ネットワークにおけるソースからシンクへの路を，**フロー増加路**を呼ぶ.

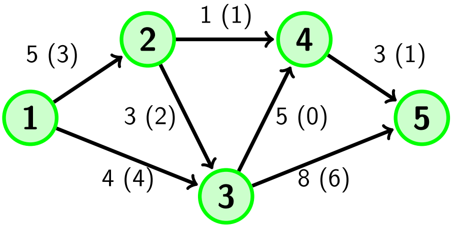
- フロー増加路に沿ってフローを追加することによって，元のネットワーク上での流量を増加できる.
- 追加できるフローの量は，その路に含まれる残余容量の最小値

フロー増加路の例

- フロー増加路を見つける．追加できる量は $\min\{u^x_{12}, u^x_{23}, u^x_{34}, u^x_{45}\} = 1$

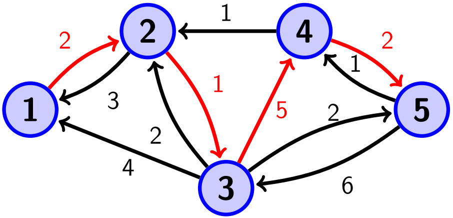


- フロー増加路に沿ってフローを追加し，新しいフローを得る．

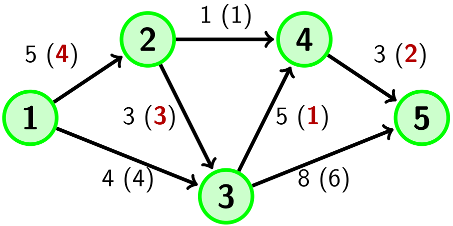


フロー増加路の例

- フロー増加路を見つける．追加できる量は
  $\min\{u^x_{12}, u^x_{23}, u^x_{34}, u^x_{45}\} = 1$



- フロー増加路に沿ってフローを追加し，新しいフローを得る．



フロー増加法

- 最大流問題に対する代表的なアルゴリズム
- これまで数多くの改良版 (多項式時間アルゴリズム) が提案

フロー増加法の計算手順

0. 適当な初期フロー $\boldsymbol{x}$ を定める．
   例えば，$\forall(i,j) \in E$ に対して，$x_{ij} = 0$ とする．
1. 残余ネットワーク $G^x = (V, E^x)$ において，ソース s からシンク t へのフロー増加路を見つける．なければ計算終了．
2. フロー増加路に沿って可能な限りフローを追加し，新しいフロー $\boldsymbol{x}$ を得る．ステップ (1) に戻る．

- フロー増加路を見つけるには?
  $\Longrightarrow$ **ラベリング法**

ラベリング法

- L: ソース s から到達可能であることが判明している節点集合
- S: 先の到達可能性を走査済の節点集合．$S \subseteq L$
- $p(j)$: フロー増加路において，j の直前の節点

ラベリング法の計算手順

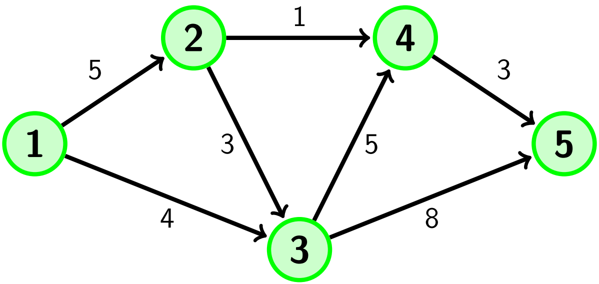
0. $L := \{s\}$, $S := \emptyset$ とする．すべての節点 $i \in V$ に対して $p(i) := 0$ とする．
1. $t \in L$ (フロー増加路を発見) または $L = S$ (フロー増加路は存在しない) ならば終了．そうでなければ，節点 $i \in L \backslash S$ を一つ選び，$S := S \cup \{i\}$ とする．
2. 残余ネットワークにおける節点 i を始点とする枝 (i, j) のすべてに対して

$$j \notin L \text{ ならば } L := L \cup \{j\},\ p(j) := i$$

とする．ステップ (1) に戻る．

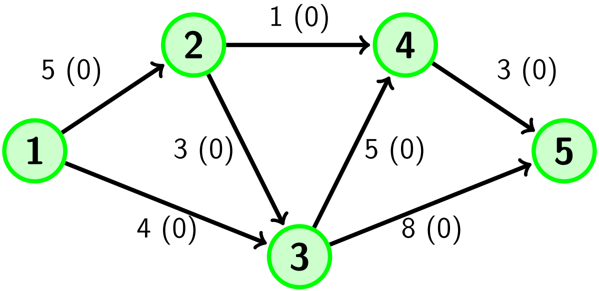
フロー増加法の計算例

ラベリング法を用いたフロー増加法を以下の例に適用する．ただし，$s = 1$, $t = 5$ とする．



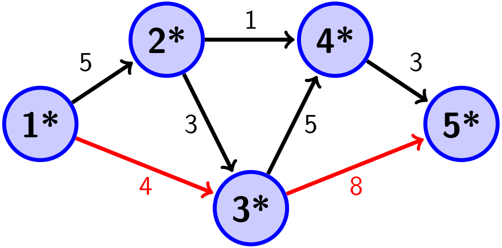
初期化

$\forall(i,j) \in E$ に対して，$x_{ij} = 0$ とする．



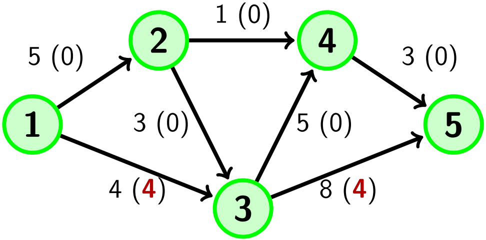
反復 1

1. 残余ネットワークに対してラベリング法を適用



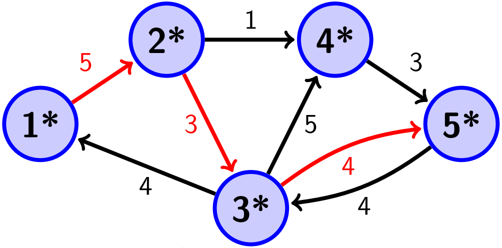
$p(1)$	0
$p(2)$	1
$p(3)$	1
$p(4)$	2
$p(5)$	3

2. フロー増加路に沿ってフローを追加し，新しいフローを得る．



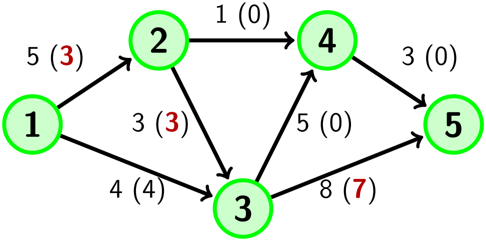
反復 2

1. 残余ネットワークに対してラベリング法を適用



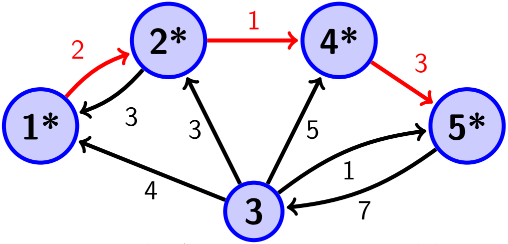
$p(1)$	0
$p(2)$	1
$p(3)$	2
$p(4)$	2
$p(5)$	3

2. フロー増加路に沿ってフローを追加し，新しいフローを得る．



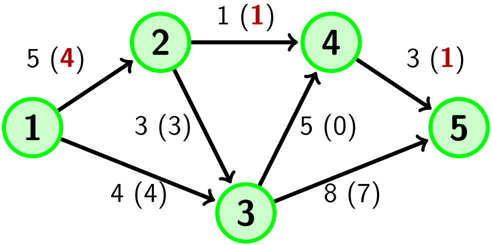
反復 3

1. 残余ネットワークに対してラベリング法を適用



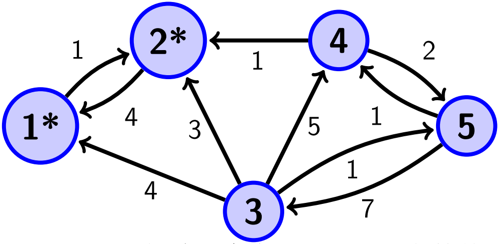
$p(1)$	0
$p(2)$	1
$p(3)$	0
$p(4)$	2
$p(5)$	4

2. フロー増加路に沿ってフローを追加し，新しいフローを得る．



反復 4

1. 残余ネットワークに対してラベリング法を適用



$p(1)$	0
$p(2)$	1
$p(3)$	0
$p(4)$	0
$p(5)$	0

2. フロー増加路が存在しないため計算終了．最大流の流量は 8