

# アルゴリズムとデータ構造⑩

## ～ 最大流 ～

鹿島久嗣

# 最大流

ネットワーク：

辺に容量をもった、入口と出口をもつ有向グラフ

■ グラフ  $G = (V, E)$  : 頂点を辺 (= 枝) でつないだもの

–  $V$  : 頂点集合 (有限集合)

–  $E$  : 辺の集合 ( $V$  上の2項関係 ;  $E \subseteq V \times V$ )  
直積集合

• 辺  $e = (v, w)$  に向きがある場合が有向グラフ

■ ネットワーク：

– 有向グラフで、各辺  $e = (v, w)$  が容量  $c(e) \geq 0$  をもつ

– 特別な頂点  $v_s$  (入口) と  $v_t$  (出口) をもつ

※ 以下、簡単のため始点からは出る方向の辺のみ、  
終点には入る方向の辺のみがあるとする (これは本質的でない)

ネットワークのフロー：

ネットワークの入り口から出口までモノを流す

- ネットワーク上で入口から出口までモノを流すことを考える

- フロー  $f: E \rightarrow \mathbb{R}_+$

–  $f(e) \geq 0$  は辺  $e$  の上を通す物量のようなもの

- ただし、 $f(e) \leq c(e)$ ：辺の容量より多くは通せない

- (出入り口以外の) 各頂点  $v$  において以下の関係が成立

$$\sum_{e \in V^+(v)} f(e) = \sum_{e \in V^-(v)} f(e)$$

出入りのバランスが  
取れている

–  $V^+(v)$  は  $v$  に入る辺； $V^-(v)$  は  $v$  から出る辺

# ネットワークの最大流問題：

容量制約を満たしながら、できるだけモノを流す

## ■ 最大流問題：

– 入口から出る量（= 出口に入る量）を最大化する

$$\max_f \sum_{e \in V^-(v_s)} f(v_s) \left( = \max_f \sum_{e \in V^+(v_t)} f(v_t) \right)$$

$$\text{s. t. } 0 \leq f(e) \leq c(e) \quad \forall e \in E$$

– 問題の解として決定すべきはフロー  $f$ （各辺に通す量）

# 最大流問題のアルゴリズム： フォード・ファルカーソン

- フォード・ファルカーソンの基本的な考え方：  
現在の解（フロー）に新たなフローを逐次的に足していく
  - 現在のフロー  $f$  があるとする
  - ある条件（後述）を満たす  $v_s$  から  $v_t$  へのパス  $p$  をみつける
    - そのパスに沿って追加のフロー  $\Delta f$  を流す
$$f(e) \leftarrow f(e) + \Delta f(e) \text{ for } \forall e \in p$$
  - 条件（後述）を満たすパスがなくなるまで以上を繰り返す

## フォード・ファルカーソンにおける解の更新： フローを追加できる「余裕」のあるパスを見つける

- $v_s$  から  $v_t$  への（辺の向きを無視した）パス  $p$  を考える
- パスに含まれる各辺  $e = (v, w) \in p$  に対して2つの場合：
  1. 正順：パスの向きと、辺の向きが一致している場合
  2. 逆順：一致していない場合
- 各辺  $e$  について「余裕」  $g(e)$  を考える
  - 正順の場合：  $g(e) = c(e) - f(e)$ ：新たに流せる余裕
  - 逆順の場合：  $g(e) = f(e)$ ：逆向きに流せる（減らせる）余裕がある

フォード・ファルカーソンにおける解の更新：  
フローを追加できる「余裕」のあるパスを見つける

- 「パスの余裕」をパス上の辺の余裕の最小値で定義する：

$$g(p) = \min_{e \in p} g(e)$$

- $g(p)$ が正であれば、このパスに沿ってさらに $g(p)$ の物流を新たに流せるはず

– 正順の辺については物流を増やす

$$f(e) \leftarrow f(e) + g(p)$$

– 逆順の辺については物流を減らす（逆向きに流す）

$$f(e) \leftarrow f(e) - g(p)$$

## 余裕のあるパスの発見： たとえば、幅優先探索

---

- $g(p)$ が正であるようなパス $p$ を見つけるには？
  - フォード・ファルカーソンでは $p$ の見つけ方は決められていない
- たとえば、幅優先探索で $g(e)$ が正である（余裕のある）辺をつなげていく（=エドモンズ・カーブ）
  - できるだけ短いパスから見つけることになる

# マッチング

# 2部グラフ：

## 2つの集合の関係を表すグラフ

---

### ■ 2部グラフ

- 頂点集合が2つの集合 $U$ と $V$ に分かれている
- 枝は2つの集合間にしか存在しない

## 2部グラフのマッチング：

2グループ間で成立するペアの数を最大化する

---

### ■ マッチング問題：

– 2部グラフにおいて、以下の条件を満たす辺の集合を選ぶ

- すべての頂点は、たかだか 1 つの辺にしか含まれない

– もっとも大きい辺の集合を見つける

### ■ 例：合コン的な何か

– 2つのグループ間で、各人が（互いに）パートナーを組んでいいと思う人の間に辺があるとする

– 成立するペアの数を最大化したい

# マッチング問題のアルゴリズム： 最大流に帰着できる

---

- 最大流問題に帰着可能

- $v_s$  から  $U$  の各頂点に容量1の辺

- $U$  と  $V$  の間の辺に容量1

- $U$  から  $v_t$  の各頂点に容量1の辺

- フォード・ファルカーソンを適用すればマッチングが得られる

- フォード・ファルカーソンでは、すべての辺の容量が整数であれば、その解も整数

- この場合、 $U$  と  $V$  の間の辺を流れる量は0か1

# 割り当て問題： マッチング問題の一般化

---

- マッチングを多対多の対応に一般化
- 学生とゼミのマッチング：
  - 学生は2つのゼミに所属
  - 各ゼミは最大10名まで受け入れ可能
- 最大流に帰着できる：
  - 2部グラフの構成法：
    - $v_s$ から $U$ の各頂点に容量2の辺
    - $U$ から $v_t$ の各頂点に容量10の辺

## 割り当て問題のアルゴリズム： やはり最大流に帰着できる

- 懸念：最大流が保証するのは、容量を超えないということだけなので、各学生が必ず2つのゼミに所属することは保証できない？
  - 条件が満たされているなら最大流の総流量は、学生数の2倍になっているはず
    - そうでないなら、そもそも不可能
  - できるならやっているはずなので、解が見つかるということは少なくともそのような解のひとつを見つけていることになる