離散最適化基礎論　第 3 回
二部グラフの最大マッチング：アルゴリズム

岡本 吉央
okamotoy@uec.ac.jp

電気通信大学

2020 年 10 月 20 日

最終更新：2020 年 10 月 22 日　10:32

スケジュール 前半 (予定)

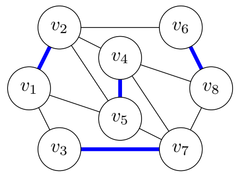
注意：予定の変更もありうる

スケジュール 後半 (予定)

注意：予定の変更もありうる

最大マッチング問題 (maximum matching problem)

定義：最大マッチング問題
- 入力：無向グラフ $G = (V, E)$
- 出力：G の最大マッチング M (を 1 つ)



概要

今日の目標

二部グラフの最大マッチングを見つけるアルゴリズムを理解し，
使えるようになる
- 増加道の発見法
- 最適性の保証法
- (時間があれば) 高速化の手法

目次

二部グラフ

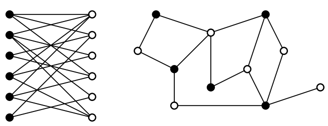
無向グラフ $G = (V, E)$

定義：二部グラフ (bipartite graph)

G が二部グラフであるとは，
- 頂点集合 V を 2 つの集合 A, B に分割できて
- どの辺 $e \in E$ も一端点を A に持ち，もう一端点を B に持つもの

A と B を G の部集合と呼ぶ

二部グラフの例



このとき，二部グラフを $G = (A, B, E)$ と表記することがある

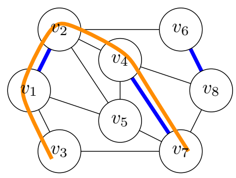
交互道

無向グラフ $G = (V, E)$，マッチング $M \subseteq E$

定義：交互道 (alternating path)

M に関する交互道とは，G における道で，
M の辺と $E - M$ の辺が交互に現れるもの

交互道を交互路と呼ぶこともある



これ は 青のマッチング に関する交互道である

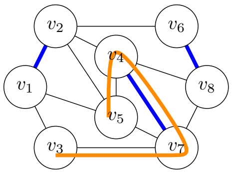
増加道

無向グラフ $G = (V, E)$，マッチング $M \subseteq E$

定義：増加道 (augmenting path)

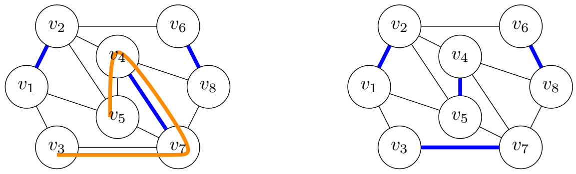
M に関する増加道とは，M に関する交互道で，
その両端点が M の辺と接続しないもの

増加道を増大道，増加路，増大路と呼ぶこともある



これ は 青のマッチング に関する増加道である

増加道に沿ってマッチングを大きくする



辺数 3 の マッチング $\rightsquigarrow$ 増加道に沿って 大きくする $\rightsquigarrow$ 辺数 4 の マッチング

つまり

M に関する増加道が存在する $\Rightarrow$ M は最大マッチングではない

つまり (対偶を考えると)

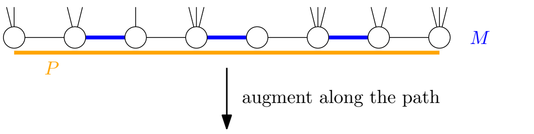
M は最大マッチングである $\Rightarrow$ M に関する増加道が存在しない

最大マッチングと増加道

無向グラフ $G = (V, E)$，マッチング $M \subseteq E$

定理：最大マッチングと増加道の関係 (Berge '57)

M が G の最大マッチング $\Leftrightarrow$ M に関する増加道が存在しない

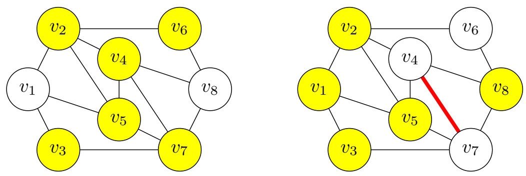


グラフの頂点被覆

無向グラフ $G = (V, E)$

定義：頂点被覆とは？

G の頂点被覆とは頂点部分集合 $C \subseteq V$ で，
G のどの辺も C のある頂点に接続しているもの



$\{v_2, v_3, v_4, v_5, v_6, v_7\}$ は 頂点被覆である

$\{v_1, v_2, v_3, v_5, v_8\}$ は 頂点被覆ではない

頂点被覆の頂点は，それに接続する辺を覆う (被覆する)

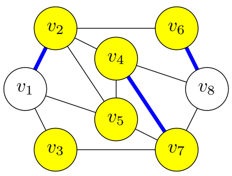
マッチングと頂点被覆の関係：弱双対性

無向グラフ $G = (V, E)$

性質：マッチングと頂点被覆の弱双対性

$\left.\begin{array}{l} G \text{ のマッチング } M \\ G \text{ の頂点被覆 } C \end{array}\right\}$ に対して，$|M| \leq |C|$

例：$|M| = 3$, $|C| = 6$



注：これは二部グラフでなくても成り立つ

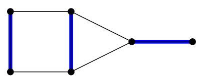
マッチングと頂点被覆の関係：弱双対性 (使い方 (1))

無向グラフ $G = (V, E)$

性質：マッチングと頂点被覆の弱双対性

$\left.\begin{array}{l} G \text{ のマッチング } M \\ G \text{ の頂点被覆 } C \end{array}\right\}$ に対して，$|M| \leq |C|$

次のマッチング M が存在する



このことから，

最大マッチングの辺数 ≥ 3

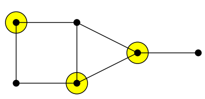
マッチングと頂点被覆の関係：弱双対性 (使い方 (2))

無向グラフ $G = (V, E)$

性質：マッチングと頂点被覆の弱双対性

$\left.\begin{array}{l} G \text{ のマッチング } M \\ G \text{ の頂点被覆 } C \end{array}\right\}$ に対して，$|M| \leq |C|$

次の頂点被覆 C が存在する



弱双対性から

最大マッチングの辺数 ≤ 3

したがって，最大マッチングの辺数 $= 3$

マッチングと頂点被覆の関係：強双対性

二部グラフ $G = (A, B, E)$

定理：Kőnig–Egerváry の定理

G の最大マッチング M と最小頂点被覆 C に対して，$|M| = |C|$

このような定理を最大最小定理 (min-max theorem) と呼ぶ

目次

二部グラフの最大マッチングを見つけるアルゴリズム：基本的な考え方

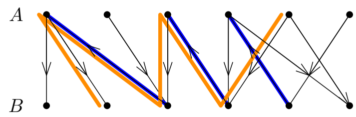
二部グラフ $G = (A, B, E)$

基本的な考え方

G のマッチング M を常に保持する

- M に関する増加道 P を見つけようとする
- 見つかったとき
  - M は最大マッチングではなく，$|M \triangle P| > |M|$
  - $M := M \triangle P$ として，繰り返す
- 見つからないとき
  - M は最大マッチングであるはず
  - $|M| = |C|$ となる頂点被覆 C を見つけ，M (と C) を出力する

増加道の発見法：例



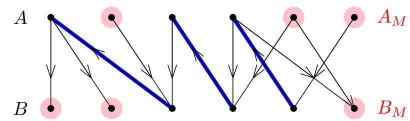
増加道アルゴリズム (1)

入力：二部グラフ $G = (A, B, E)$，マッチング $M \subseteq E$

アルゴリズム：Step 1

有向グラフ $\vec{G} = (V, \vec{E})$ を次のように構成する

- $V = A \cup B$
- $\vec{E} = \{(u, v) \mid u \in A, v \in B, \{u, v\} \notin M\} \cup$
  $\{(v, u) \mid u \in A, v \in B, \{u, v\} \in M\}$

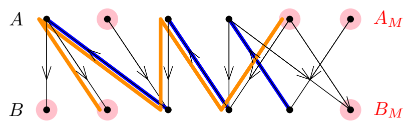


記法：$A_M :=$ M が飽和しない A の頂点全体の集合
$B_M :=$ M が飽和しない B の頂点全体の集合

増加道アルゴリズム (2)

アルゴリズム：Step 2

$\vec{G}$ において，A_M から B_M へ至る有向道を見つける



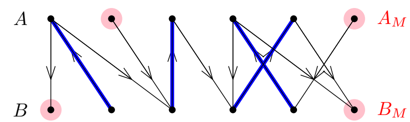
グラフ探索 (例えば，深さ優先探索や幅優先探索) を使うと
$O(|V| + |E|)$ 時間で見つけられる

増加道アルゴリズム (3)

アルゴリズム：Step 3.1

有向道が見つかった場合 (見つかった有向道を P とする)

- M を $M \triangle P$ として Step 1 に戻る



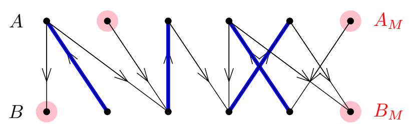
P をたどることで，$O(|V| + |E|)$ 時間で行なえる

増加道アルゴリズム (4)

アルゴリズム：Step 1

有向グラフ $\vec{G} = (V, \vec{E})$ を次のように構成する

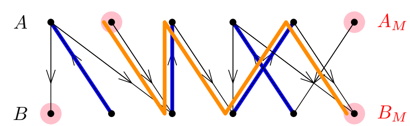
- $V = A \cup B$
- $\vec{E} = \{(u, v) \mid u \in A, v \in B, \{u, v\} \notin M\} \cup$
  $\{(v, u) \mid u \in A, v \in B, \{u, v\} \in M\}$



増加道アルゴリズム (5)

アルゴリズム：Step 2

$\vec{G}$ において，A_M から B_M へ至る有向道を見つける



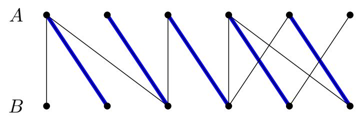
グラフ探索 (例えば，深さ優先探索や幅優先探索) を使うと
$O(|V| + |E|)$ 時間で見つけられる

増加道アルゴリズム (6)

アルゴリズム：Step 3.1

有向道が見つかった場合 (見つかった有向道を P とする)

- M を $M \triangle P$ として Step 1 に戻る



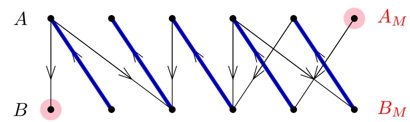
P をたどることで，$O(|V|+|E|)$ 時間で行なえる

増加道アルゴリズム (7)

アルゴリズム：Step 1

有向グラフ $\vec{G} = (V, \vec{E})$ を次のように構成する

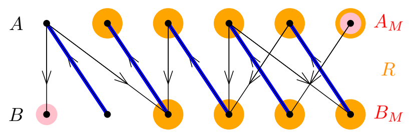
- $V = A \cup B$
- $\vec{E} = \{(u,v) \mid u \in A, v \in B, \{u,v\} \notin M\} \cup$
  $\{(v,u) \mid u \in A, v \in B, \{u,v\} \in M\}$



増加道アルゴリズム (8)

アルゴリズム：Step 2

$\vec{G}$ において，A_M から B_M へ至る有向道を見つける



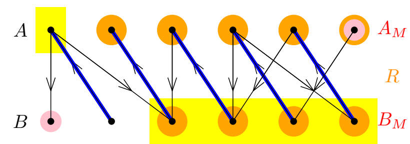
A_M から B_M へ至る有向道が存在しないとき，
A_M から有向道でたどり着ける頂点全体の集合を R とする

増加道アルゴリズム (9)

Step 3.2

A_M から有向道でたどり着ける頂点全体の集合を R とする

- $C = (A - R) \cup (B \cap R)$ とする
- 得られているマッチング M と頂点被覆 C を出力する



増加道アルゴリズム：全体像

入力：二部グラフ $G = (A, B, E)$，マッチング $M \subseteq E$ (例えば，$M = \emptyset$)

Step 1：有向グラフ $\vec{G} = (V, \vec{E})$ を次のように構成する

- $V = A \cup B$
- $\vec{E} = \{(u,v) \mid u \in A, v \in B, \{u,v\} \notin M\} \cup$
  $\{(v,u) \mid u \in A, v \in B, \{u,v\} \in M\}$

Step 2：$\vec{G}$ において，A_M から B_M へ至る有向道を見つける

Step 3.1：有向道が見つかった場合 (見つかった有向道を P とする)

- M を $M \triangle P$ として Step 1 に戻る

Step 3.2：有向道が見つからない場合

- A_M から有向道でたどり着ける頂点全体の集合を R とする
- マッチング M と頂点被覆 $C = (A - R) \cup (B \cap R)$ を出力する

アルゴリズムを設計したら考えるべきこと

アルゴリズムを設計したら考えるべきこと

正当性

- アルゴリズムの出力が正しいこと

計算量

- アルゴリズムの計算量 (の上界) を評価 (算定) すること

増加道アルゴリズムに対して考えるべきこと

正当性

- 出力 M が最大マッチングであること

計算量

- 計算量が $O(|V||E|)$ であること

増加道アルゴリズム：各 Step の計算量

Step 1：有向グラフ $\vec{G} = (V, \vec{E})$ を次のように構成 $O(|V|+|E|)$

- $V = A \cup B$
- $\vec{E} = \{(u,v) \mid u \in A, v \in B, \{u,v\} \notin M\} \cup$
  $\{(v,u) \mid u \in A, v \in B, \{u,v\} \in M\}$

Step 2：$\vec{G}$ において，A_M から B_M へ至る有向道を見つける
$O(|V|+|E|)$

Step 3.1：有向道が見つかった場合 (見つかった有向道を P とする)

- M を $M \triangle P$ として Step 1 に戻る $O(|V|+|E|)$

Step 3.2：有向道が見つからない場合 $O(|V|+|E|)$

- A_M から有向道でたどり着ける頂点全体の集合を R とする
- マッチング M と頂点被覆 $C = (A - R) \cup (B \cap R)$ を出力

考えるべきこと

Step 1 と Step 2 を実行する回数 (反復回数)

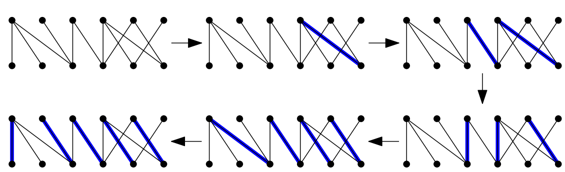
増加道アルゴリズム：反復回数

補題：増加道アルゴリズムの反復回数

増加道アルゴリズムにおける反復回数は $O(|V|)$

証明：アルゴリズムは $M = \emptyset$ として始まる

- Step 3.1 において，$|M|$ は 1 だけ増える
- マッチングの要素数は大きくても $|V|/2 = O(|V|)$
- ∴ Step 3.1 が実行される回数 $\leq |V|/2 = O(|V|)$
- ∴ 反復回数 $= O(|V|)$ □



増加道アルゴリズム：計算量

増加道アルゴリズムの計算量

$=$ (Step 1 の計算量 $+$ Step 2 の計算量) $\times$ 反復回数 $+$
Step 3.1 の計算量 $\times$ (反復回数 $-$ 1) $+$ Step 3.2 の計算量

$= (O(|V|+|E|)+O(|V|+|E|)) \times O(|V|) +$
$O(|V|+|E|) \times O(|V|) + O(|V|+|E|)$

$= O(|V||E|)$ □

注

- グラフが連結であるとき，$|V| \leq |E| - 1$
- グラフが連結でないときは，
  連結成分ごとに最大マッチングを求めればよい

増加道アルゴリズム：正当性 (1)

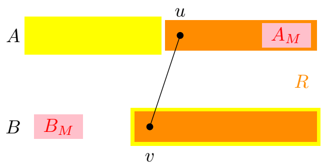
終了したときに得られる M と $C = (A - R) \cup (B \cap R)$ を考える

補題：増加道アルゴリズムの正当性

1. C は G の頂点被覆である
2. $|M| = |C|$

1 の証明：任意の辺 $\{u, v\}$ ($u \in A, v \in B$) を考える

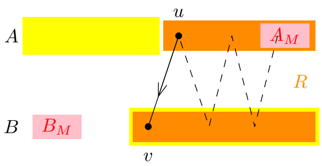
- $u \notin A - R$ であるとき，$v \in B \cap R$ となることを示せばよい



増加道アルゴリズム：正当性 (2)

場合 (1)：$\{u, v\} \notin M$ のとき

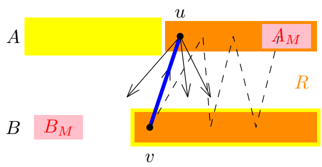
- $\overrightarrow{G}$ で (u, v) と向き付けされる
- $u \in A \cap R$ であるから，A_M から u に有向道でたどり着ける
- $\therefore$ (u, v) を使うことで，A_M から v にも有向道でたどり着ける
- $\therefore v \in B \cap R$



増加道アルゴリズム：正当性 (3)

場合 (2)：$\{u, v\} \in M$ のとき

- $\overrightarrow{G}$ で (v, u) と向き付けされる
- u に接続する他の辺は B に入るように向き付けされている
- $\therefore$ A_M から u にたどり着くには，v を経由する必要がある
- $\therefore$ A_M から v にたどり着ける
- $\therefore v \in B \cap R$



増加道アルゴリズム：正当性 (4)

2 の証明：$|M| = |C|$ を証明する

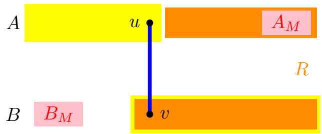
- 弱双対性と 1 より，$|M| \leq |C|$
- $\therefore$ $|M| \geq |C|$ を証明すればよい
- 任意の $v \in B \cap R$ は M に飽和される
  ($\because$ そうでないと，$v \in B_M$ となり，停止したことに矛盾)
- 任意の $u \in A - R$ は M に飽和される
  ($\because$ そうでないと，$u \in A_M \subseteq R$ となり，$u \notin R$ に矛盾)

増加道アルゴリズム：正当性 (4)

- $v \in B \cap R$ と $u \in A - R$ を同時に飽和する M の辺はない
  ($\because$ あるとすると，$u \in R$ となってしまう)
- したがって，

$$
\begin{aligned}
|M| &\geq |B \cap R \text{ を飽和する } M \text{ の辺の集合}| + \\
&\quad\ |A - R \text{ を飽和する } M \text{ の辺の集合}| \\
&= |B \cap R| + |A - R| = |C|
\end{aligned}
$$

□



増加道アルゴリズム：まとめ

定理：増加道アルゴリズム

増加道アルゴリズムによって
二部グラフの最大マッチング問題を $O(|V||E|)$ 時間で解ける

目次

Hopcroft–Karp のアルゴリズム

定理：増加道アルゴリズム

増加道アルゴリズムによって
二部グラフの最大マッチング問題を $O(|V||E|)$ 時間で解ける

↓ 改良 ↓

定理：Hopcroft–Karp のアルゴリズム ('73)

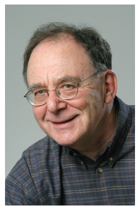
Hopcroft–Karp のアルゴリズムによって
二部グラフの最大マッチング問題を $O(\sqrt{|V|}|E|)$ 時間で解ける

反復回数

増加道アルゴリズム： $O(|V|)$
Hopcroft–Karp のアルゴリズム： $O(\sqrt{|V|})$

ホップクロフト と カープ

John Hopcroft
(1939–)



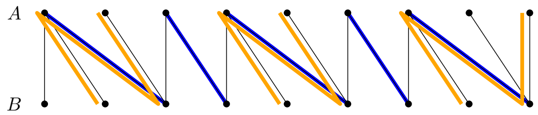
Richard Karp
(1935–)

http://www.cs.cornell.edu/jeh/
https://www2.eecs.berkeley.edu/Faculty/Homepages/karp.html

Hopcroft–Karp のアルゴリズム：基本アイディア

基本アイディア

1. 並列に実行できる増加を一気に行う
2. 増加を「最短増加道」を用いて行う



そのような最短増加道の集合は 幅優先探索によって
$O(|V|+|E|)$ 時間で見つかる

Hopcroft–Karp のアルゴリズム

入力：二部グラフ $G=(A,B,E)$，マッチング $M=\emptyset$

Step 1：有向グラフ $\vec{G}=(V,\vec{E})$ を構成する
- 構成法は，増加道アルゴリズムのときと同じ

Step 2：$\vec{G}$ で A_M から B_M へ至る極大な最短有向道の集合を見つける

Step 3.1：有向道が見つかった場合 (見つかった有向道を $P_1, P_2, \dots, P_k$ とする)
- M を $M \triangle P_1 \triangle P_2 \triangle \cdots \triangle P_k$ として Step 1 に戻る

Step 3.2：有向道が見つからない場合
- A_M から有向道でたどり着ける頂点全体の集合を R とする
- マッチング M と頂点被覆 $C=(A-R)\cup(B\cap R)$ を出力する

Hopcroft–Karp のアルゴリズム：直感

マッチング $M_1=\emptyset$	$\rightsquigarrow$	M_1 に関する最短増加道 P_1
マッチング $M_2=M_1\triangle P_1$	$\rightsquigarrow$	M_2 に関する最短増加道 P_2
マッチング $M_3=M_2\triangle P_2$	$\rightsquigarrow$	M_3 に関する最短増加道 P_3
マッチング $M_4=M_3\triangle P_3$	$\rightsquigarrow$	$\cdots$
マッチング $M_\mu=M_{\mu-1}\triangle P_{\mu-1}$	$\rightsquigarrow$	終了

このとき，
- $|P_1| \leq |P_2| \leq \cdots \leq$ (補題 B)
- また，同じ長さの増加道は頂点を共有しない ($\therefore$ 同時に処理できる) (補題 C)
- $\therefore$ 反復回数 $=P_1, P_2, \dots$ に現れる異なる長さの種類
- $P_{\mu-\sqrt{\mu}}$ の長さ $=O(\sqrt{\mu})$ (定理)
- 残り $\sqrt{\mu}$ 個の長さの種類 $=O(\sqrt{\mu})$
- $\therefore$ 反覆回数 $=O(\sqrt{\mu})=O(\sqrt{|V|})$

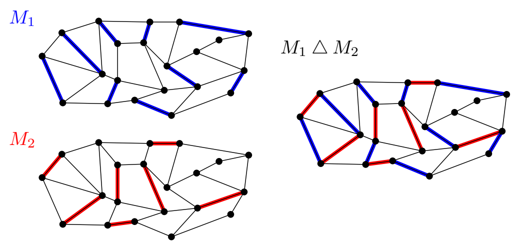
Hopcroft–Karp のアルゴリズム：補題 A

無向グラフ $G=(V,E)$，G のマッチング M_1, M_2，$|M_1| \geq |M_2|$

補題 A

$M_1 \triangle M_2$ には M_2 に関する増加道が 少なくとも $|M_1|-|M_2|$ 個 存在し，
それらは頂点を共有しない

例：$|M_1|=9$，$|M_2|=7$



Hopcroft–Karp のアルゴリズム：補題 A (証明)

証明：$M_1 \triangle M_2$ は偶数長閉路と道で構成される

偶数長の閉路	M_1 の辺数	$=$	M_2 の辺数
偶数長の道	M_1 の辺数	$=$	M_2 の辺数
奇数長の道，端が M_1 の辺	M_1 の辺数	$=$	M_2 の辺数 $+1$
奇数長の道，端が M_2 の辺	M_1 の辺数 $+1$	$=$	M_2 の辺数

したがって，

$|M_1|-|M_2| =$ 端が M_1 の辺である奇数長の道の総数
$-$ 端が M_2 の辺である奇数長の道の総数
$\leq$ 端が M_1 の辺である奇数長の道の総数
$=$ M_2 に関する増加道の総数

そして，それらの増加道は頂点を共有しない □

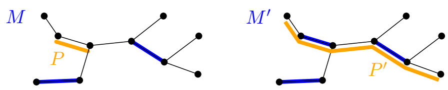
Hopcroft–Karp のアルゴリズム：補題 B

無向グラフ $G=(V,E)$，
マッチング $M\subseteq E$，M に関する最短増加道 P，
マッチング $M'=M\triangle P$，M' に関する増加道 P'

補題 B

$|P'| \geq |P|+2|P\cap P'|$

例：$|P|=1$，$|P'|=5\geq 3=|P|+2|P\cap P'|$

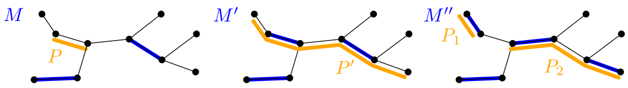


Hopcroft–Karp のアルゴリズム：補題 B (証明)

証明：$M'' = M' \triangle P'$ とする

- このとき，$|M''| = |M| + 2$
- $\therefore$ 補題 A より，$M \triangle M''$ には M に関する増加道が 2 つ以上あり，それらは頂点を共有しない (それらを P_1, P_2 とする)
- このとき，$|P| \leq |P_1|$，$|P| \leq |P_2|$　　　　　　　　　　　　　　($\because P$ は最短)
- また，

$$
\begin{aligned}
M \triangle M'' &= M \triangle (M' \triangle P') = M \triangle ((M \triangle P) \triangle P') \\
&= (M \triangle M) \triangle (P \triangle P') = \emptyset \triangle (P \triangle P') = P \triangle P'
\end{aligned}
$$



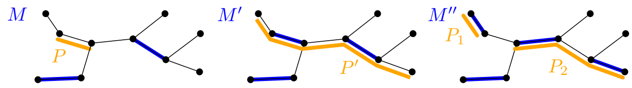
Hopcroft–Karp のアルゴリズム：補題 B (証明 続き)

証明 (続き)：

- したがって，

$$
\begin{aligned}
& |P \triangle P'| = |M \triangle M''| \geq |P_1 \cup P_2| = |P_1| + |P_2| \geq 2|P| \\
\text{かつ，}\ & |P \triangle P'| = |P| + |P'| - 2|P \cap P'|
\end{aligned}
$$

- $\therefore |P| + |P'| - 2|P \cap P'| \geq 2|P|$
- $\therefore |P'| \geq |P| + 2|P \cap P'|$ □



Hopcroft–Karp のアルゴリズム：補題 C

無向グラフ $G = (V, E)$，マッチング $M_1 \subseteq E$，
M_1 に関する最短増加道 P_1，$M_2 = M_1 \triangle P_1$，
M_2 に関する最短増加道 P_2，$M_3 = M_2 \triangle P_2$，$\ldots$，
M_i に関する最短増加道 P_i，$M_{i+1} = M_i \triangle P_i$，$\ldots$

補題 C

$P_1, P_2, \ldots$ の中で長さが同じものは 頂点を共有しない

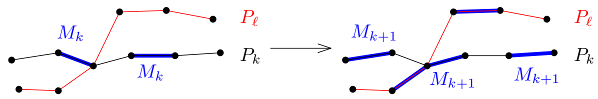
背理法による証明：P_k と P_ℓ ($k < \ell$) は同じ長さで，頂点を共有するものの中で，$\ell - k$ が最小のものであるとする

- 補題 B より，$|P_k| \leq |P_{k+1}| \leq \cdots \leq |P_{\ell-1}| \leq |P_\ell|$
- $|P_k| = |P_\ell|$ なので，$|P_k| = |P_{k+1}| = \cdots = |P_{\ell-1}| = |P_\ell|$
- また，仮定より，$P_{k+1}, \ldots, P_{\ell-1}, P_\ell$ は頂点を共有しない
- $\therefore P_\ell$ は M_{k+1} に関する増加道

Hopcroft–Karp のアルゴリズム：補題 C (証明 続き)

証明 (続き)：

- 補題 B より，$|P_\ell| \geq |P_k| + 2|P_k \cap P_\ell|$
- $|P_k| = |P_\ell|$ なので，$|P_k \cap P_\ell| = 0$ ($\therefore P_k \cap P_\ell = \emptyset$)
- つまり，P_k と P_ℓ は辺を共有しない
- しかし，このとき，P_ℓ は M_{k+1} に関する増加道になれない □



Hopcroft–Karp のアルゴリズム：動作

	$P_1\ P_2\ P_3\ P_4\ P_5\ P_6\ P_7$	$P_8\ P_9$	$P_{10}\ P_{11}$	$P_{12}\ P_{13}$	P_{14}
length	1	3	7	9	11
iteration	1	1	1	1	1

補題 C より，
同じ長さの増加道は 1 つの反復で見つかり，並行して同時に処理できる

- $\therefore$ 反復回数は $P_1, P_2, \ldots$ の中の異なる長さの種類

Hopcroft–Karp のアルゴリズム：定理

$P_1, P_2, \ldots$：Hopcroft–Karp で用いる増加道を順に並べたもの
$M_1 = \emptyset$，$M_2 = M_1 \triangle P_1$，$\ldots$，$M_{i+1} = M_i \triangle P_i$，$\ldots$

定理：Hopcroft–Karp のアルゴリズムの反復回数

$P_1, P_2, \ldots$ の中に現れる増加道の長さの種類 $= O(\sqrt{|V|})$

証明：二部グラフ G の最大マッチングの要素数を μ とする ($\mu = O(|V|)$)

- $r = \lfloor \mu - \sqrt{\mu} \rfloor$ とする (注意：$|M_r| = r - 1$)
- 補題 A より，M_r に関する増加道で頂点を共有しないものが $\mu - r + 1$ 個以上存在する
- $\therefore$ その中で最短のものに含まれる M_r の辺の数 $\leq \dfrac{r-1}{\mu - r + 1}$
- $\therefore |P_r| \leq 2\dfrac{r-1}{\mu - r + 1} + 1 = O(\sqrt{\mu})$

Hopcroft–Karp のアルゴリズム：定理 (証明 続き)

証明 (続き)：(補足：$r = \lfloor \mu - \sqrt{\mu} \rfloor$)

- $\therefore M_{r+1}$ が見つかるまでの反復回数 $= O(\sqrt{\mu})$
- 一方で，$M_{r+1}, \ldots, M_\mu$ までの反復回数 $\leq \mu - r = O(\sqrt{\mu})$
- したがって，アルゴリズムの反復回数 $= O(\sqrt{\mu}) = O(\sqrt{|V|})$ □

目次

今日のまとめ と 次回の予告

今日の目標

二部グラフの最大マッチングを見つけるアルゴリズムを理解し，
使えるようになる

- 増加道の発見法
- 最適性の保証法
- (時間があれば) 高速化の手法

次回の予告

二部グラフでないグラフの最大マッチング

- Tutte の定理
- Berge–Tutte の公式

重要概念

- 奇成分