離散最適化基礎論 第 5 回
マトロイドとグラフの全域木

岡本 吉央
okamotoy@uec.ac.jp

電気通信大学

2015 年 11 月 13 日

最終更新：2016 年 8 月 23 日　11:51

スケジュール 前半 (予定)

- ⋆ 休講 (卒研準備発表会) (10/2)
- 1 組合せ最適化問題におけるマトロイドの役割 (10/9)
- ⋆ 休講 (海外出張) (10/16)
- 2 マトロイドの定義と例 (10/23)
- 3 マトロイドの基と階数関数 (10/30)
- 4 グラフとマトロイド (11/6)
- 5 マトロイドとグラフの全域木 (11/13)
- ⋆ 休講 (調布祭) (11/20)
- 6 マトロイドに対する貪欲アルゴリズム (11/27)
- 7 マトロイドのサーキット (12/4)

注意：予定の変更もありうる

スケジュール 後半 (予定)

⋆	休講 (国内出張)	(12/11)
8	マトロイドに対する操作	(12/18)
9	マトロイドの交わり	(12/25)
⋆	冬季休業	(1/1)
10	マトロイド交わり定理	(1/8)
⋆	休講 (センター試験準備)	(1/15)
11	マトロイド交わり定理：アルゴリズム	(1/22)
12	最近のトピック	(1/29)
⋆	授業等調整日 (予備日)	(2/5)
⋆	期末試験	(2/12?)

注意：予定の変更もありうる

テーマ：解きやすい組合せ最適化問題が持つ「共通の性質」

疑問

どうしてそのような違いが生まれるのか？

$\rightsquigarrow$ 解きやすい問題が持つ「共通の性質」は何か？

回答

よく分かっていない

しかし，部分的な回答はある

部分的な回答

問題が「マトロイド的構造」を持つと解きやすい

ポイント

効率的アルゴリズムが設計できる背景に「美しい数理構造」がある

この講義では，その一端に触れたい

今日の目標

グラフの最小全域木問題とマトロイドの関係を探る

- グラフの最小全域木問題 $\leftrightarrow$ マトロイドの最小基問題
- Kruskal のアルゴリズム $\leftrightarrow$ 貪欲アルゴリズム
- 貪欲アルゴリズムの正当性
- 「貪欲アルゴリズムの正当性」におけるマトロイドの必要性

目次

グラフ $G=(V,E)$，G の閉路マトロイド $\mathcal{I}$

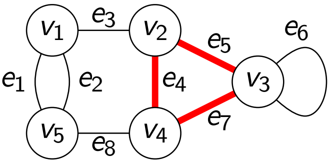
命題：閉路マトロイドをグラフの用語で捉え直す

G の辺部分集合 $X \subseteq E$ に対して

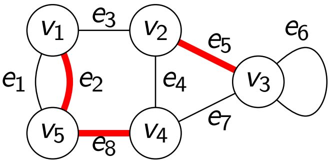
$$X \in \mathcal{I} \quad \Leftrightarrow \quad \text{グラフ } G[X]=(V,X) \text{ が閉路を含まない}$$

例：

$\{e_4, e_5, e_7\} \notin \mathcal{I}$



$\{e_2, e_5, e_8\} \in \mathcal{I}$



無向グラフ $G = (V, E)$

森と木

- G が森 (あるいは林，forest) であるとは，G が閉路を含まないこと
- G が木 (tree) であるとは，G が連結であり，閉路を含まないこと

全域森と全域木

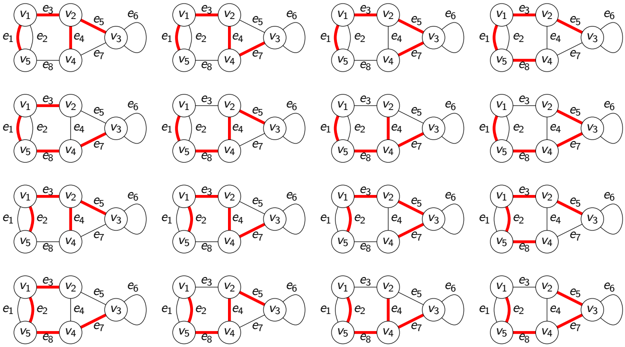
$X \subseteq E$ に対して，$G[X] = (V, X)$ とする

- $G[X]$ が G の全域森 (spanning forest) であるとは，$G[X]$ が閉路を含まないこと
- $G[X]$ が G の全域木 (spanning tree) であるとは，$G[X]$ が連結であり，閉路を含まないこと

例において，G の閉路マトロイドの基族 $\mathcal{B}$ は

$$
\begin{aligned}
\mathcal{B} \;=\; & \{\{e_1, e_3, e_4, e_5\}, \{e_1, e_3, e_4, e_7\}, \{e_1, e_3, e_5, e_7\}, \{e_1, e_3, e_5, e_8\},\\
& \{e_1, e_3, e_7, e_8\}, \{e_1, e_4, e_5, e_8\}, \{e_1, e_4, e_7, e_8\}, \{e_1, e_5, e_7, e_8\},\\
& \{e_2, e_3, e_4, e_5\}, \{e_2, e_3, e_4, e_7\}, \{e_2, e_3, e_5, e_7\}, \{e_2, e_3, e_5, e_8\},\\
& \{e_2, e_3, e_7, e_8\}, \{e_2, e_4, e_5, e_8\}, \{e_2, e_4, e_7, e_8\}, \{e_2, e_5, e_7, e_8\}\}
\end{aligned}
$$

閉路マトロイドの基が
G の全域木に対応
($\because$ G が連結)



連結グラフ G	G の閉路マトロイド
G の全域森	独立集合
G の全域木	基
閉路を含む G の部分グラフ	従属集合
G の閉路	サーキット

復習：E 上のマトロイド $\mathcal{I}$ に対して

- $X \subseteq E$ が $\mathcal{I}$ の従属集合であるとは，$X \notin \mathcal{I}$
- $X \subseteq E$ が $\mathcal{I}$ のサーキットであるとは，
  $X \notin \mathcal{I}$ で，任意の $e \in X$ に対して，$X - \{e\} \in \mathcal{I}$
  (つまり，X は極小な従属集合)

注意：文献では「全域森」を違う意味で使うこともある

目次

グラフの最小全域木問題とは？

連結無向グラフ $G = (V, E)$ と重み $c\colon E \to \mathbb{R}_+$ に対して

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{e \in B} c(e) \\ \text{subject to} & G[B] = (V, B) \text{ は } G \text{ の全域木} \end{array}$$

最小全域木問題を効率よく解く方法

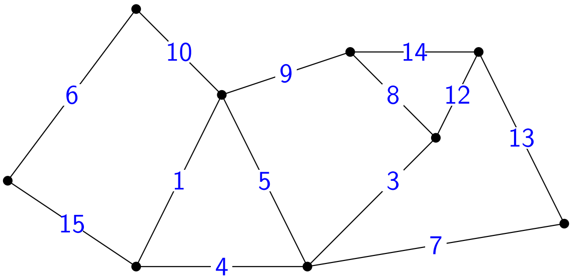
Kruskal のアルゴリズム

1. G の辺を費用の小さい順に並べる
   ($c(e_1) \leq c(e_2) \leq \cdots \leq c(e_n)$ であると仮定する)
2. $B \leftarrow \emptyset$
3. すべての $i \leftarrow 1, \ldots, n$ に対して，以下を繰り返し
$$B \leftarrow \begin{cases} B \cup \{e_i\} & (B \cup \{e_i\} \text{ が閉路を含まないとき}) \\ B & (B \cup \{e_i\} \text{ が閉路を含むとき}) \end{cases}$$
4. B を出力

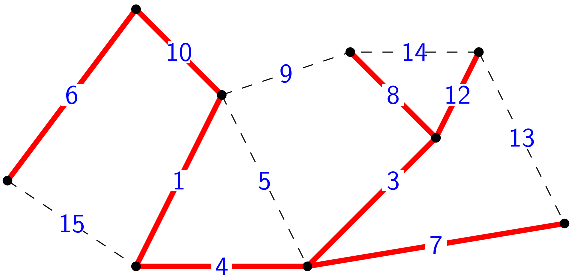
これは正しいアルゴリズム (必ず最小全域木を出力する)

- 証明：『アルゴリズム論第一』か『アルゴリズム論第二』を参照
- 別証明：この講義 (マトロイドを使用)

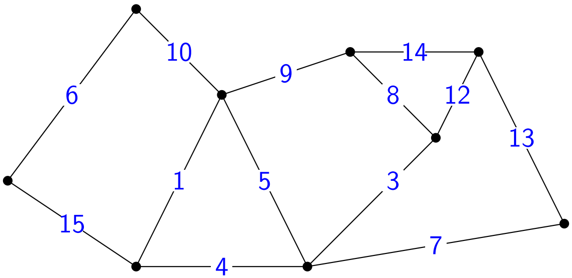
辺の上に書いてある数字が，その辺の重み



辺の上に書いてある数字が，その辺の重み

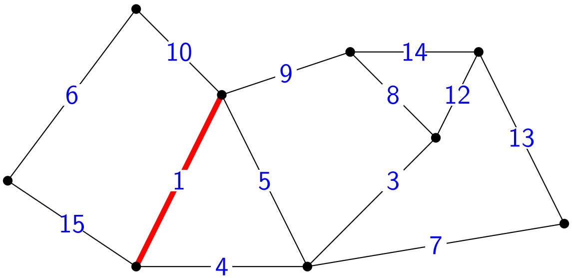


辺の上に書いてある数字が，その辺の重み



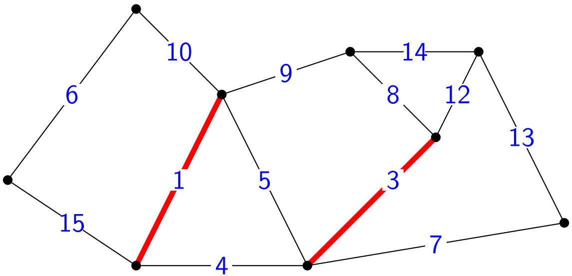
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



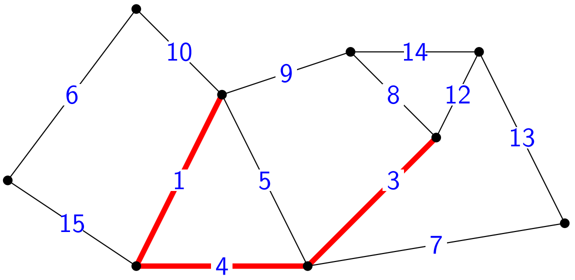
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

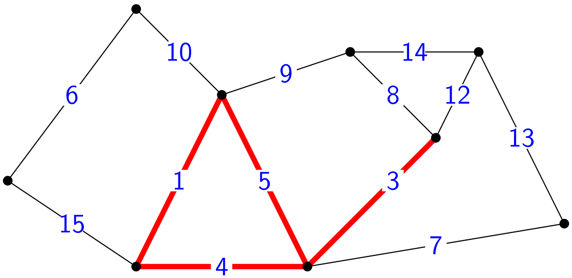
辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

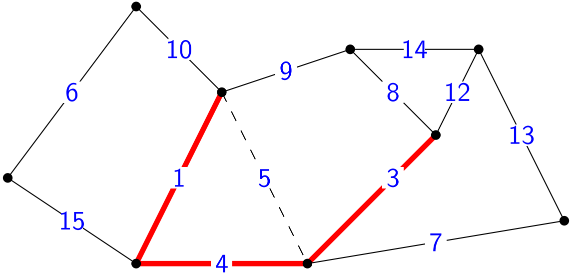
グラフの最小全域木問題：例 — Kruskal のアルゴリズムを実行

辺の上に書いてある数字が，その辺の重み



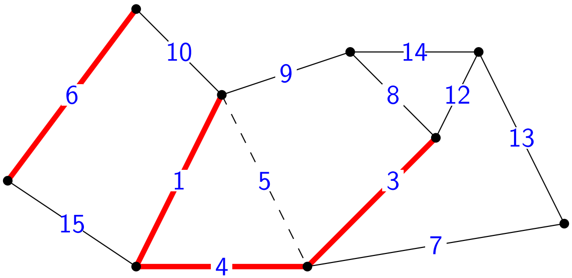
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

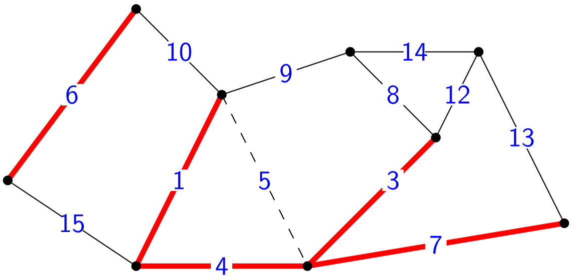
辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

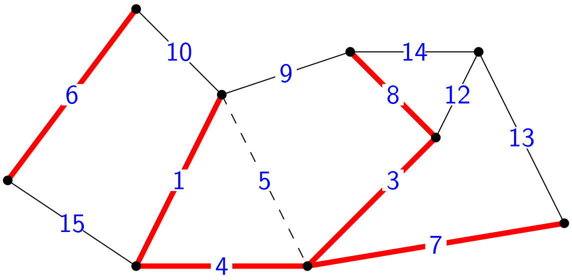
グラフの最小全域木問題：例 — Kruskal のアルゴリズムを実行

辺の上に書いてある数字が，その辺の重み



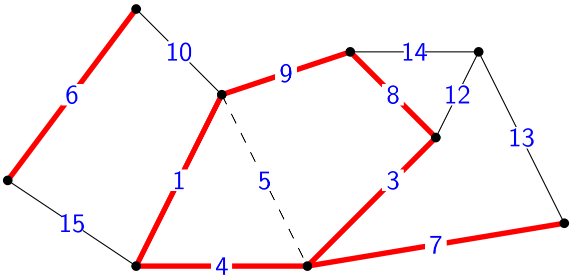
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



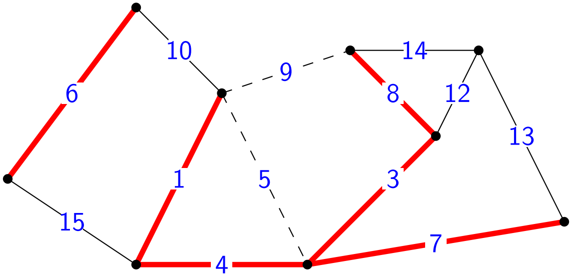
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



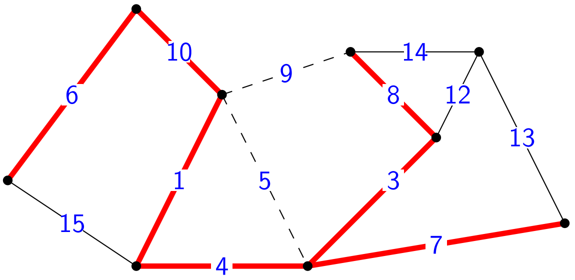
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



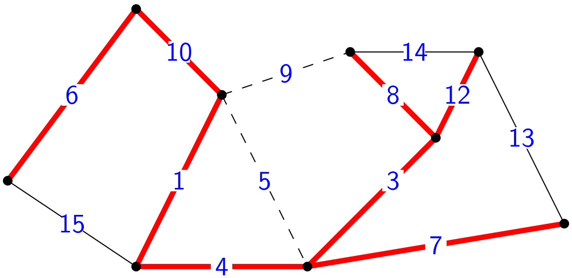
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



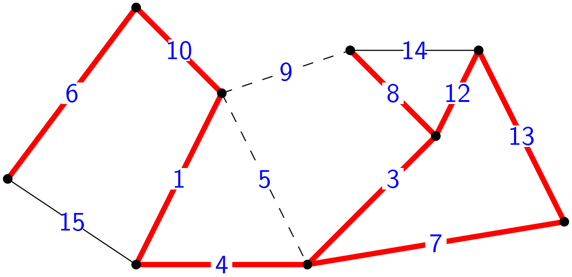
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



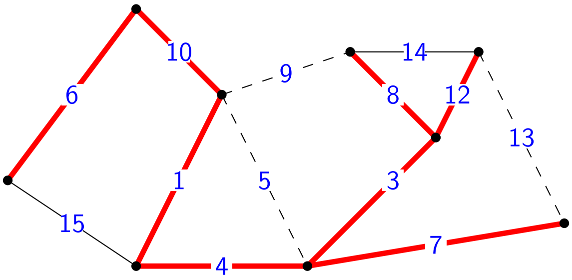
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



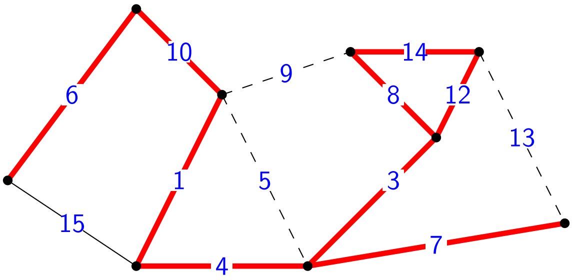
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



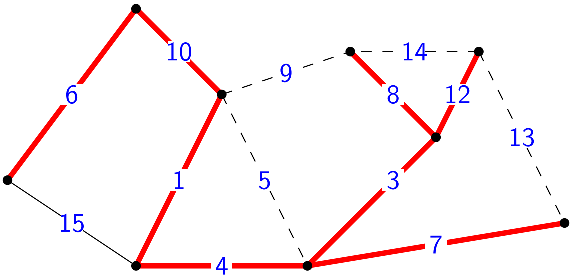
Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

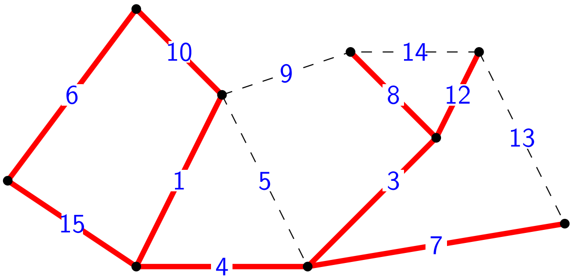
辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

グラフの最小全域木問題：例 — Kruskal のアルゴリズムを実行

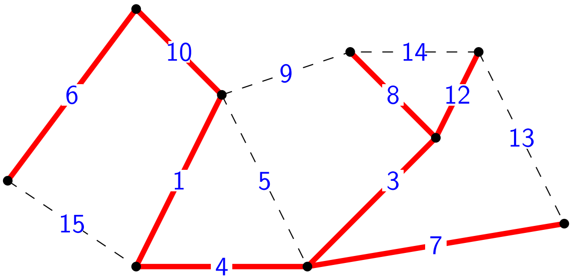
辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

グラフの最小全域木問題：例 — Kruskal のアルゴリズムを実行

辺の上に書いてある数字が，その辺の重み



Kruskal のアルゴリズム：最小全域木を計算する貪欲アルゴリズム

グラフの最小全域木問題とは?: 閉路マトロイドで言い換え

連結無向グラフ $G = (V, E)$ と重み $c \colon E \to \mathbb{R}_+$ に対して

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{e \in B} c(e) \\ \text{subject to} & B \text{ は } G \text{ の閉路マトロイド } \mathcal{I} \text{ の基} \end{array}$$

マトロイドの最小基問題

有限集合 E 上のマトロイド $\mathcal{I}$ と重み $c\colon E \to \mathbb{R}_+$ に対して

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in B} c(e) \\ \text{subject to} \quad & B \text{ は } \mathcal{I} \text{ の基} \end{aligned}$$

マトロイドの最小基問題

有限集合 E 上のマトロイド $\mathcal{I}$ と重み $c\colon E \to \mathbb{R}_+$ に対して

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{e \in B} c(e) \\ \text{subject to} & B \text{ は } \mathcal{I} \text{ の基} \end{array}$$

一方，第 1 回講義で考えた問題は次のもの

マトロイドの最大独立集合問題

有限集合 E 上のマトロイド $\mathcal{I}$ と重み $w\colon E \to \mathbb{R}_+$ に対して

$$\begin{array}{ll} \text{maximize} & \displaystyle\sum_{e \in X} w(e) \\ \text{subject to} & X \in \mathcal{I} \end{array}$$

非空な有限集合 E, E 上のマトロイド $\mathcal{I}$, その基族 $\mathcal{B}$

命題

- 任意の $c\colon E \to \mathbb{R}_+$ を考える
- このとき，$C = \sum_{e \in E} c(e)$ として，$w\colon E \to \mathbb{R}_+$ を次のように定義する

$$\text{任意の } e \in E \text{ に対して，} \quad w(e) = C - c(e)$$

このとき，

$$\underbrace{\min\left\{\sum_{e \in B} c(e) \,\middle|\, B \in \mathcal{B}\right\}}_{\text{最小基問題の最適値}} \quad = \quad C \cdot r(E) - \underbrace{\max\left\{\sum_{e \in X} w(e) \,\middle|\, X \in \mathcal{I}\right\}}_{\text{最大独立集合問題の最適値}}$$

- 補足：$r(E)$ は E の階数 ($= \mathcal{I}$ の基の要素数)
- 注意：任意の $e \in E$ に対して，$w(e) \geq 0$

マトロイドの最小基問題とマトロイドの最大独立集合問題：関係

非空な有限集合 E, E 上のマトロイド $\mathcal{I}$, その基族 $\mathcal{B}$

命題

- 任意の $c\colon E \to \mathbb{R}_+$ を考える
- このとき，$C = \sum_{e \in E} c(e)$ として，$w\colon E \to \mathbb{R}_+$ を次のように定義する

$$\text{任意の } e \in E \text{ に対して，} \quad w(e) = C - c(e)$$

このとき，

$$\underbrace{\min\left\{\sum_{e \in B} c(e) \,\middle|\, B \in \mathcal{B}\right\}}_{\text{最小基問題の最適値}} \quad = \quad C \cdot r(E) - \underbrace{\max\left\{\sum_{e \in X} w(e) \,\middle|\, X \in \mathcal{I}\right\}}_{\text{最大独立集合問題の最適値}}$$

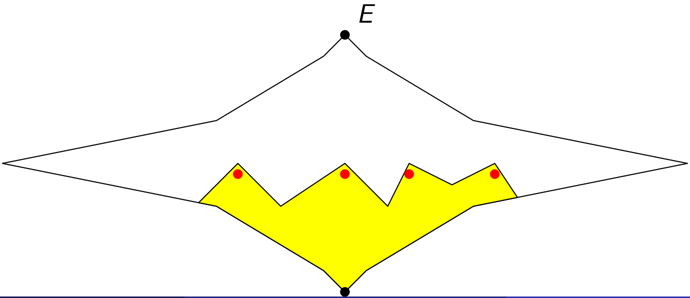
意味

最大独立集合問題が解ければ，最小基問題も解ける

観察 1

最大独立集合問題の最適解として，基であるものが存在する

- 独立集合 $X \in \mathcal{I}$ が最適解であるとする
- このとき，X を含む基 $B \in \mathcal{I}$ が存在 (cf. 演習問題 3.14)
- $X \subseteq B$ なので，$\displaystyle\sum_{e \in X} w(e) \leq \sum_{e \in B} w(e)$
- したがって，B も最適解 □



観察 2

集合 $X, Y \subseteq E$ に対して,

$$\sum_{e \in X} c(e) \le \sum_{e \in Y} c(e) \quad \Leftrightarrow \quad C|X| - \sum_{e \in X} w(e) \le C|Y| - \sum_{e \in Y} w(e)$$

$$\begin{aligned}
& & C|X| - \sum_{e \in X} w(e) &\le C|Y| - \sum_{e \in Y} w(e) \\
&\Leftrightarrow & \sum_{e \in X} (C - w(e)) &\le \sum_{e \in Y} (C - w(e)) \\
&\Leftrightarrow & \sum_{e \in X} c(e) &\le \sum_{e \in Y} c(e) \qquad \square
\end{aligned}$$

- 観察 2 より，

$$\min\left\{\sum_{e\in B} c(e) \,\middle|\, B \in \mathcal{B}\right\} = C \cdot r(E) - \max\left\{\sum_{e\in B} w(e) \,\middle|\, B \in \mathcal{B}\right\}$$

- 観察 1 より

$$\max\left\{\sum_{e\in B} w(e) \,\middle|\, B \in \mathcal{B}\right\} = \max\left\{\sum_{e\in X} w(e) \,\middle|\, X \in \mathcal{I}\right\}$$

- したがって，

$$\min\left\{\sum_{e\in B} c(e) \,\middle|\, B \in \mathcal{B}\right\} = C \cdot r(E) - \max\left\{\sum_{e\in X} w(e) \,\middle|\, X \in \mathcal{I}\right\}$$

□

目次

E 上の独立集合族 $\mathcal{F}$，重み $w\colon E \to \mathbb{R}_+$

最大独立集合問題に対する貪欲アルゴリズム

1. E の要素 e を $w(e)$ の大きい順に並べる
   $(w(e_1) \geq w(e_2) \geq \cdots \geq w(e_n)$ であると仮定する$)$
2. $X \leftarrow \emptyset$
3. すべての $i \leftarrow 1, 2, \ldots, n$ に対して，以下を繰り返し
$$X \leftarrow \begin{cases} X \cup \{e_i\} & (X \cup \{e_i\} \in \mathcal{F} \text{ のとき}) \\ X & (X \cup \{e_i\} \notin \mathcal{F} \text{ のとき}) \end{cases}$$
4. X を出力

第 1 回の講義で紹介した形と違うが，動きは同じ (で効率がよい)

非空な有限集合 E，E 上の独立集合族 $\mathcal{F}$

マトロイドに対する貪欲アルゴリズムの正当性

$\mathcal{F}$ がマトロイド $\Rightarrow$ 任意の重み $w\colon E \to \mathbb{R}_+$ に対して，貪欲アルゴリズムの出力は最大独立集合問題の最適解

つまり，Kruskal のアルゴリズムも正しい

$E = \{e_1, e_2, \ldots, e_n\}$ で，$w(e_1) \geq \cdots \geq w(e_n)$ であるとする

- 貪欲アルゴリズムの出力を B とする (B は $\mathcal{I}$ の基)
- 最適解の 1 つを B^* とする (B^* は $\mathcal{I}$ の基)
- 任意の $i \in \{1, \ldots, n\}$ に対して，次のように集合を定める

$$\begin{aligned} B_i &= B \cap \{e_1, \ldots, e_i\} \\ B_i^* &= B^* \cap \{e_1, \ldots, e_i\} \end{aligned}$$

- r を $\mathcal{I}$ の階数関数とするとき，次が成り立つ

1. $|B_i| = r(\{e_1, \ldots, e_i\})$ (なぜか？)
2. $|B_i^*| \leq r(\{e_1, \ldots, e_i\})$ (なぜか？)

- したがって，$|B_i| \geq |B_i^*|$

E 上のマトロイドの階数関数 $r\colon 2^E \to \mathbb{N}$ とは？

- 任意の $X \in 2^E$ に対して $r(X) = \max\{|Y| \mid Y \subseteq X, Y \in \mathcal{I}\}$
- つまり，$r(X)$ は X の基の要素数

X の基とは?: 次を満たす集合 B_X

- $B_X \subseteq X$，かつ，$B_X \in \mathcal{I}$
- 任意の $e \in X - B_X$ に対して，$B_X \cup \{e\} \notin \mathcal{I}$

階数関数の性質 (第 3 回講義より)

- $X \in \mathcal{I}$ ならば，$r(X) = |X|$
- $X \subseteq Y$ ならば，$r(X) \leq r(Y)$ (単調性)

確認

2. $|B_i^*| \leq r(\{e_1, \ldots, e_i\})$

- B_i^* の定義より，$B_i^* \subseteq B^*$
- $B^* \in \mathcal{I}$ なので，(I2) より，$B_i^* \in \mathcal{I}$
- 階数関数の性質より，$r(B_i^*) = |B_i^*|$
- $B_i^* \subseteq \{e_1, \ldots, e_i\}$ なので，単調性より，$r(B_i^*) \leq r(\{e_1, \ldots, e_i\})$
- したがって，$|B_i^*| = r(B_i^*) \leq r(\{e_1, \ldots, e_i\})$ $\square$

証明 (3)

確認

1. $|B_i| = r(\{e_1, \ldots, e_i\})$

- B_i が $\{e_1, \ldots, e_i\}$ の基であることを示せばよい

- B_i の定義より，$B_i \subseteq \{e_1, \ldots, e_i\}$
- $B_i \subseteq B$ と $B \in \mathcal{I}$ と (I2) より，$B_i \in \mathcal{I}$
- 任意の $e_j \in \{e_1, \ldots, e_i\} - B_i$ を考える
- $B_i \cup \{e_j\} \in \mathcal{I}$ であるとすると，
  貪欲アルゴリズムの第 i 反復で B_i が得られることに矛盾
- したがって，$B_i \cup \{e_j\} \notin \mathcal{I}$
- つまり，B_i は $\{e_1, \ldots, e_i\}$ の基である

したがって，$B_0 = B_0^* = \emptyset$, $w(e_{n+1}) = 0$ としたとき，次が成り立つ

$$
\begin{aligned}
\sum_{e \in B} w(e) &= \sum_{i=1}^{n} (|B_i| - |B_{i-1}|) w(e_i) \\
&= \sum_{i=1}^{n} |B_i| (w(e_i) - w(e_{i+1})) \\
&\geq \sum_{i=1}^{n} |B_i^*| (w(e_i) - w(e_{i+1})) \\
&= \sum_{i=1}^{n} (|B_i^*| - |B_{i-1}^*|) w(e_i) \\
&= \sum_{e \in B^*} w(e)
\end{aligned}
$$

すなわち，B も最適解である □

目次

非空な有限集合 E，E 上の独立集合族 $\mathcal{F}$

定理：マトロイドであることの必要性

任意の重み $w\colon E \to \mathbb{R}_+$ に対して貪欲アルゴリズムの出力が最適解
$\Rightarrow$ $\mathcal{F}$ はマトロイド

証明：対偶を示す

- $\mathcal{F}$ がマトロイドではないと仮定する

証明すること

ある重み $w\colon E \to \mathbb{R}_+$ に対して貪欲アルゴリズムは最適解を出力しない

- $\mathcal{F}$ は独立集合族なので，$\mathcal{F}$ は (I1), (I2) を満たす
- つまり，$\mathcal{F}$ は (I3) を満たさない
- すなわち，$|X| > |Y|$ を満たす $X, Y \in \mathcal{F}$ が存在して，
  任意の $e \in X - Y$ に対して $Y \cup \{e\} \notin \mathcal{F}$ となる

- このとき，$k = |Y|$ として，次のように $w\colon E \to \mathbb{R}_+$ を定義

$$w(e) = \begin{cases} k+2 & (e \in Y \text{ のとき}) \\ k+1 & (e \in X - Y \text{ のとき}) \\ 0 & (\text{その他}) \end{cases}$$

- B を貪欲アルゴリズムの出力とすると，$Y \subseteq B$
- つまり，B は $X - Y$ の要素をどれも含まない
- $\therefore \sum_{e \in B} w(e) = |Y|(k+2) = k(k+2)$
- 一方，$\sum_{e \in X} w(e) \geq |X|(k+1) \geq (|Y|+1)(k+1) = (k+1)(k+1)$
- $k(k+2) < (k+1)(k+1)$ なので，B は最適解ではない □

目次

非空な有限集合 E，独立集合族 $\mathcal{F} \subseteq 2^E$

定理

次の 2 つは同値

1. 任意の重み関数 $w\colon E \to \mathbb{R}_+$ に対して，貪欲アルゴリズムが最適解を出力する
2. 独立集合族 $\mathcal{F}$ がマトロイドである

つまり，

- マトロイド (matroid) とは特殊な性質を持つ独立集合族
- マトロイドに対しては，貪欲アルゴリズムが必ず最適解を出力する

$\rightsquigarrow$ マトロイドはとても性質のよい独立集合族

今回，これを確かに証明した

今日のまとめ

グラフの最小全域木問題とマトロイドの関係を探る

- グラフの最小全域木問題 $\leftrightarrow$ マトロイドの最小基問題
- Kruskal のアルゴリズム $\leftrightarrow$ 貪欲アルゴリズム
- 貪欲アルゴリズムの正当性
- 「貪欲アルゴリズムの正当性」におけるマトロイドの必要性

次回

- 貪欲アルゴリズムの応用

残った時間の使い方

- 演習問題をやる
  - 相談推奨 (ひとりでやらない)
- 質問をする
  - 教員は巡回
- 退室時，小さな紙に感想など書いて提出する ← 重要
  - 内容は何でも OK
  - 匿名で OK

目次