情報数理科学特別講義 D ／情報数理科学特別講義 H ／情報数理科学特殊講義 B
第 6 回
厳密アプローチ：前処理による問題の縮小

岡本 吉央

okamotoy@uec.ac.jp

電気通信大学

2012 年 8 月 28 日

最終更新： 2012 年 8 月 27 日　04:53

前処理とは？

前処理 (preprocessing)

入力を変形させるアルゴリズム

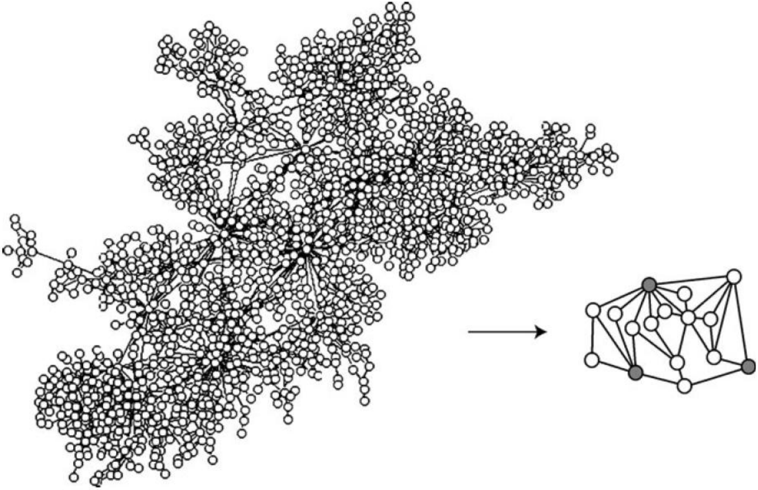
- これで出力は変化しない
- 入力がより解きやすくなる

この講義の目標

次の理解

- 前処理による問題の縮小の典型例を理解
- 前処理による問題の縮小の性能解析を理解

前処理による問題の縮小の威力



(Hüffner, Niedermeier, Wernicke 2008 **より**)

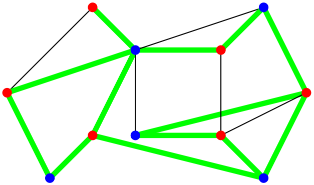
1. 前処理による問題の縮小
2. 最大カット問題
3. 頂点被覆問題
4. ランキング構成問題
5. 今日のまとめ

最大カット問題

定義：最大カット問題 (判定問題版)

- 入力：**連結な**無向グラフ $G = (V, E)$，自然数 k
- 出力：G の頂点への赤と青の色割当 (彩色である必要はない) で，
  2 色辺の数が k 以上のものがあるとき「Yes」
  そうでないとき「No」

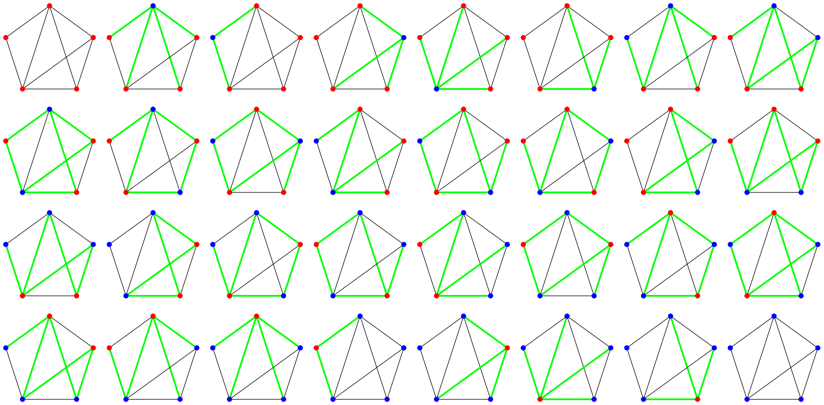
2 色辺：2 端点が 2 色で塗られている辺



2 色辺の数 $= 13$

すぐに思いつく単純なアルゴリズム

すべての色割当を試して，2 色辺の数を計算する



色割当の総数 $= 2^n$ ($n =$ **頂点数**)

出力が Yes であると簡単に分かる場合

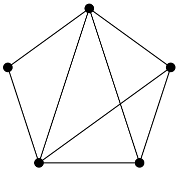
n：入力グラフ G の頂点数

観察 1

$k \leq n-1$ ならば，出力は Yes

<u>証明</u>：G は連結なので，G に全域木が存在 (T とする)

- T の辺数 $= n-1$
- 頂点を 1 つ任意に定め，T におけるそこからの距離の偶奇で色割当
- この色割当における 2 色辺の数 $\geq n-1$ □



出力が Yes であると簡単に分かる場合

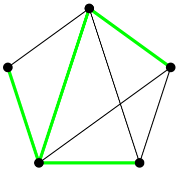
n：入力グラフ G の頂点数

観察 1

$k \leq n-1$ ならば，出力は Yes

証明：G は連結なので，G に全域木が存在 (T とする)

- T の辺数 $= n-1$
- 頂点を 1 つ任意に定め，T におけるそこからの距離の偶奇で色割当
- この色割当における 2 色辺の数 $\geq n-1$ $\square$



出力が Yes であると簡単に分かる場合

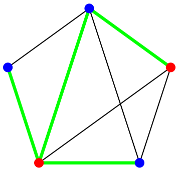
n：入力グラフ G の頂点数

観察 1

$k \leq n-1$ ならば，出力は Yes

証明：G は連結なので，G に全域木が存在 (T とする)

- T の辺数 $= n-1$
- 頂点を 1 つ任意に定め，T におけるそこからの距離の偶奇で色割当
- この色割当における 2 色辺の数 $\geq n-1$ $\square$



問題となる入力の頂点数は小さい

アルゴリズム MaxCut(G, k)

1. 「$k \leq n - 1$」ならば，Yes を出力して停止
2. そうでないとき // $k \geq n$
   すべての色割当を試して解く

- Step 2 で試すすべての色割当の数 $= 2^n \leq 2^k$
- よって，全体の計算量は $O(2^k)$ ぐらい

キーポイント

Step 2 を必要とする入力グラフ G の頂点数は k 以下 (つまり小さい)
$\cdots$ 解くべき問題が縮小された

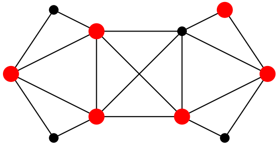
1. 前処理による問題の縮小
2. 最大カット問題
3. 頂点被覆問題
4. ランキング構成問題
5. 今日のまとめ

頂点被覆問題

頂点被覆問題 (判定問題)

- 入力：無向グラフ $G = (V, E)$，自然数 k
- 出力：G が要素数 k 以下の頂点被覆を持てば「Yes」，そうでなければ「No」

G の頂点被覆 (vertex cover)：
頂点部分集合 S で，どの辺の端点も 1 つは S に含まれるもの



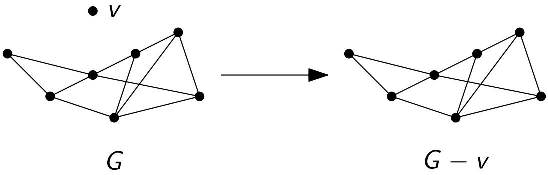
問題の縮小：規則 1

$\langle G, k \rangle$：**入力**

観察 2

v：G **における次数** 0 **の頂点 (孤立点** (isolated vertex))
このとき，次の 2 **つは同値**

1. $\langle G, k \rangle$ **に対する出力が** Yes
2. $\langle G - v, k \rangle$ **に対する出力が** Yes



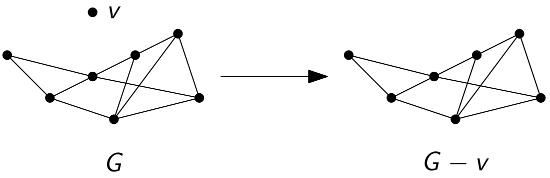
証明：板書 □

問題の縮小：規則 1 (続)

観察 2 より次の縮小規則が得られる

頂点被覆問題：規則 1

入力 $\langle G, k\rangle$ に対して，G に孤立点 v が存在するならば，
$\langle G, k\rangle$ を $\langle G - v, k\rangle$ に置き換え



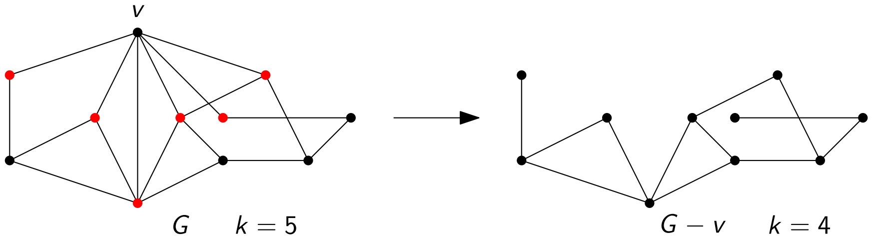
問題の縮小：規則 2

$\langle G, k \rangle$：**入力**

観察 3

v：**G における次数 $k+1$ 以上の頂点**
このとき，次の 2 つは同値

1. $\langle G, k \rangle$ **に対する出力が** Yes
2. $\langle G - v, k-1 \rangle$ **に対する出力が** Yes



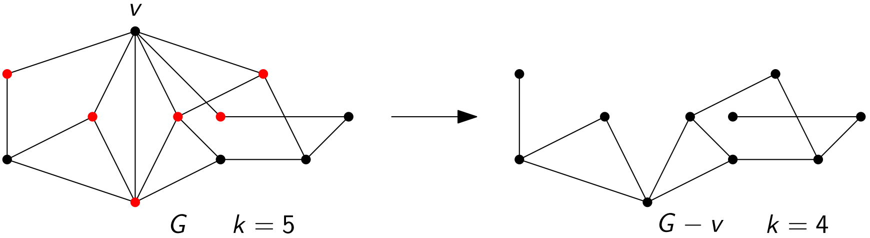
証明：板書

問題の縮小：規則 2 (続)

観察 3 より次の縮小規則が得られる

頂点被覆問題：規則 2

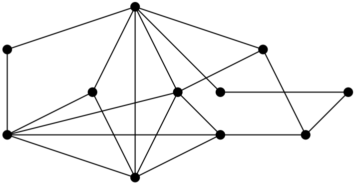
入力 $\langle G, k\rangle$ に対して，G に次数 $k+1$ 以上の頂点 v が存在するならば，$\langle G, k\rangle$ を $\langle G-v, k-1\rangle$ に置き換え



頂点被覆問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k) (Buss)

1. **規則 1 と規則 2 を適用できる限り** $\langle G, k\rangle$ **に適用**
   $//\langle G, k\rangle$ **は変化していく**
2. **適用できなくなったら，**$\langle G, k\rangle$ **を出力**

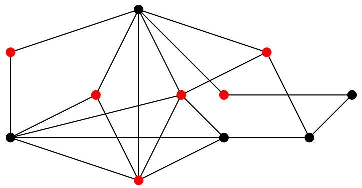


$k = 5$

頂点被覆問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k) (Buss)

1. **規則** 1 **と規則** 2 **を適用できる限り** $\langle G, k\rangle$ **に適用**
   $//\langle G, k\rangle$ **は変化していく**
2. **適用できなくなったら，**$\langle G, k\rangle$ **を出力**

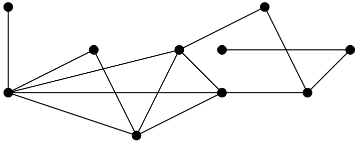


$k = 5$

頂点被覆問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k) (Buss)

1. **規則** 1 **と規則** 2 **を適用できる限り** $\langle G, k\rangle$ **に適用**
   $//\langle G, k\rangle$ **は変化していく**
2. **適用できなくなったら，**$\langle G, k\rangle$ **を出力**

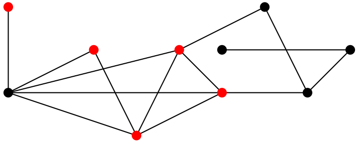


$k = 4$

頂点被覆問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k) (Buss)

1. **規則** 1 **と規則** 2 **を適用できる限り** $\langle G, k\rangle$ **に適用**
   $//\langle G, k\rangle$ **は変化していく**
2. **適用できなくなったら，** $\langle G, k\rangle$ **を出力**

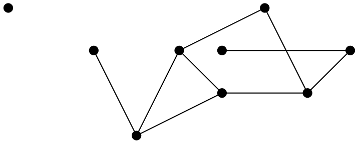


$k = 4$

頂点被覆問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k) (Buss)

1. **規則** 1 **と規則** 2 **を適用できる限り** $\langle G, k\rangle$ **に適用**
   $//\langle G, k\rangle$ **は変化していく**
2. **適用できなくなったら，**$\langle G, k\rangle$ **を出力**

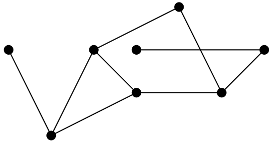


$k = 3$

頂点被覆問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k) (Buss)

1. **規則** 1 **と規則** 2 **を適用できる限り** $\langle G, k \rangle$ **に適用**
   $//\langle G, k \rangle$ **は変化していく**
2. **適用できなくなったら，**$\langle G, k \rangle$ **を出力**

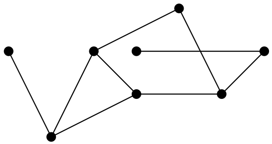


$k = 3$

頂点被覆問題：前処理アルゴリズム (解析)

規則 1 と規則 2 が適用できなくなったときの状況は？

- **各頂点の次数** $\leq k$ (∵ **規則** 2 **が適用不可**)
- **要素数** k **の頂点部分集合が被覆できる辺数** $\leq k \cdot$ **次数の最大値** $\leq k^2$
- ∴ **規則が適用できないグラフの辺数** $> k^2 \Rightarrow$ **出力は** No



$k = 3$

頂点被覆問題：前処理に基づくアルゴリズム

アルゴリズム VertexCover(G, k)

1. $\langle G, k\rangle \leftarrow$ Preprocess(G, k)
2. G **の辺数** $> k^2$ **ならば，**No **を出力して停止**
3. **そうでなければ，要素数** k **のすべての頂点部分集合を試して解く**

Step 3 **に進んだとき，**G **の頂点数** n **は？**

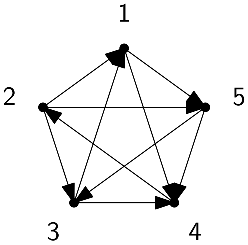
- G **の辺数** $\leq k^2$ (Step 2 **より**)
- G **の各頂点の次数** ≥ 1 (**規則** 1 **より**)
- $\therefore n \leq$ **次数和** $= 2\cdot$ **辺数** $\leq 2k^2$
- **つまり，**G **の頂点数は** $2k^2$ **以下 (小さい)**

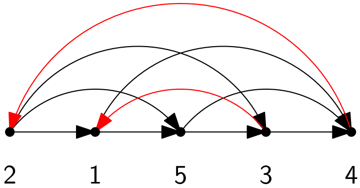
1. 前処理による問題の縮小
2. 最大カット問題
3. 頂点被覆問題
4. ランキング構成問題
5. 今日のまとめ

ランキング構成問題

ランキング構成問題 (判定問題)

- 入力：頂点数 n の完全有向グラフ G，自然数 k
- 出力：G の頂点全体を 1 列に並べる方法で，
  逆向き辺の数が k 以下のものがあれば「Yes」，そうでなければ「No」





よく「the feedback arc set problem in tournaments」とも呼ばれる

ランキング構成問題：性質

G：完全有向グラフ

観察 4

G の頂点全体を 1 列に並べる方法で，逆向き辺がないものが存在
$\Leftrightarrow$ G に有向閉路が存在しない

$\Rightarrow$ の証明：そのような並べ方の存在を仮定すれば，
有向閉路が存在しないことは簡単に分かる (板書)

$\Leftarrow$ の証明：有向閉路が存在しないとき，入次数 0 の頂点が存在すること
を利用 (板書) $\square$

ランキング構成問題：性質 (続)

G：完全有向グラフ

観察 5

G に有向閉路が存在しない $\Leftrightarrow$
G に頂点数 3 の有向閉路 (有向三角形) が存在しない

$\Rightarrow$ の証明：対偶を考えれば，直ちに分かる
$\Leftarrow$ の証明：背理法．頂点数最小の有向閉路を考える (続きは板書) $\square$

ランキング構成問題：前処理の指針

- 有向三角形の中の 1 つの辺は必ず逆向きになる
- 多くの有向三角形が 1 つの辺を共有しているとき，
  それが逆向きになるように並べてはいけない

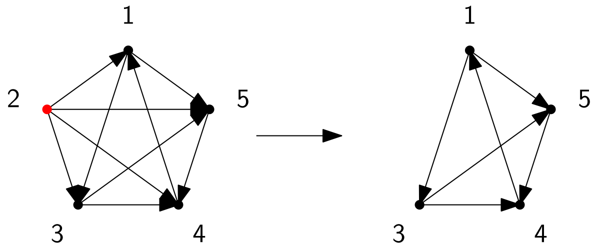
問題の縮小：規則 1

$\langle G, k \rangle$：入力

観察 5

v：G においてどの有向三角形にも含まれない頂点
このとき，次の 2 つは同値

1. $\langle G, k \rangle$ に対する出力が Yes
2. $\langle G - v, k \rangle$ に対する出力が Yes



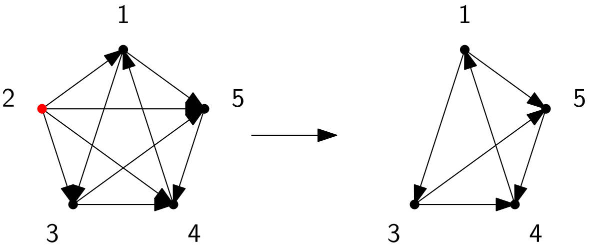
証明：板書 □

問題の縮小：規則 1 (続)

観察 5 より次の縮小規則が得られる

ランキング構成問題：規則 1

入力 $\langle G, k\rangle$ に対して，どの有向三角形にも含まれない頂点 v が存在 $\Rightarrow$ $\langle G, k\rangle$ を $\langle G - v, k\rangle$ に置き換え



問題の縮小：規則 2

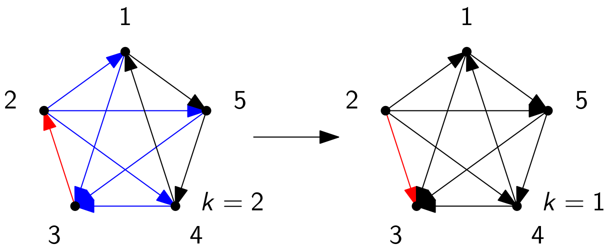
$\langle G, k \rangle$：入力

観察 6

e：G において $k+1$ 個以上の有向三角形に含まれる辺
このとき，次の 2 つは同値

1. $\langle G, k \rangle$ に対する出力が Yes
2. $\langle G', k-1 \rangle$ に対する出力が Yes

ただし，G' は e の向きを反転させたグラフ



問題の縮小：規則 2 (続 1)

「$2 \Rightarrow 1$」の証明：

- G' に対する並べ方で逆向き辺の数が $k-1$ 以下のものが存在
- その並べ方で，e の向きを反転させると G に対する並べ方になり，そこでの逆向き辺の数は k 以下

「$1 \Rightarrow 2$」の証明：

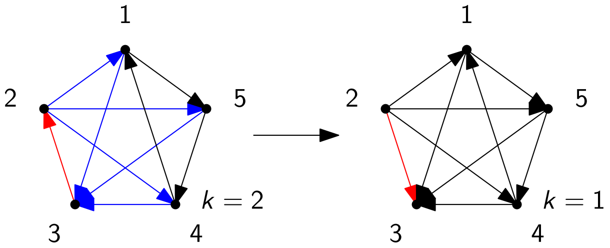
- G に対する並べ方で逆向き辺の数が k 以下のものが存在
- その並べ方において，e の向きは逆向きである　(どうして？)
  - 逆向きでないと，e を含む有向三角形の e 以外のある辺が逆向き
  - $\therefore$ $k+1$ 個以上の辺が逆向きになり，矛盾
- その並べ方で，e の向きを逆にすると，G' に対する並べ方になり，そこでの逆向き辺の数は $k-1$ 以下 $\square$

問題の縮小：規則 2 (続 2)

観察 6 より次の縮小規則が得られる

ランキング構成問題：規則 2

入力 $\langle G, k\rangle$ に対して，$k+1$ 個以上の有向三角形に含まれる辺が存在 $\Rightarrow$ $\langle G, k\rangle$ を $\langle G', k-1\rangle$ に置き換え



ランキング構成問題：前処理アルゴリズム

前処理アルゴリズム Preprocess(G, k)

(Dom, Guo, Hüffner, Niedermeier, Truss, '10)

1. **規則** 1 **と規則** 2 **を適用できる限り** $\langle G, k\rangle$ **に適用**
   $//\langle G, k\rangle$ **は変化していく**
2. **適用できなくなったら，** $\langle G, k\rangle$ **を出力**

ランキング構成問題：前処理アルゴリズム (解析)

規則 1 と規則 2 が適用できなくなったときの状況は？

- **各辺を含む有向三角形の数** $\leq k$ (∵ **規則 2 が適用不可**)
- **逆向き辺の数** $\leq k$ **となる並べ方が存在** $\Rightarrow$ **有向三角形の総数** $\leq k \cdot k = k^2$
- 1 **つの有向三角形の頂点数** $= 3$
- ∴ **有向三角形に含まれる頂点の総数** $\leq 3k^2$
- **すべての頂点はある有向三角形に含まれる** (∵ **規則 1 が適用不可**)
- ∴ **規則が適用できないグラフの頂点数** $> 3k^2 \Rightarrow$ **出力は** No

ランキング構成問題：前処理に基づくアルゴリズム

アルゴリズム Ranking(G, k)

1. $\langle G, k \rangle \leftarrow \text{Preprocess}(G, k)$
2. G **の頂点数** $> 3k^2$ **ならば，**No **を出力して停止**
3. **そうでなければ，すべての並べ方を試して解く**

Step 3 **に進んだとき，**

- G **の頂点数** $\leq 3k^2$ (Step 2 **より**)
- **つまり，**G **の頂点数は小さい**

1. 前処理による問題の縮小
2. 最大カット問題
3. 頂点被覆問題
4. ランキング構成問題
5. 今日のまとめ

第 6 回のまとめ

前処理 (preprocessing)

入力を変形させるアルゴリズム

- これで出力は変化しない
- 入力がより解きやすくなる

例

- 最大カット問題
- 頂点被覆問題
- ランキング構成問題

カーネル化 (kernelization)

この講義で扱ったような前処理は「カーネル化」と呼ばれている

カーネル化 (非形式的な定義)

自然数 k を入力の一部に含むような問題に対する**カーネル化**とは，前処理アルゴリズムで，前処理を施した後の入力の大きさが k のみに依存するようなもの

カーネル化の研究では，前処理を施した後の入力の大きさをできるだけ小さくするような工夫が重要

参考文献

- **前処理 (カーネル化) に関する解説**

  - R. Niedermeier. Invitation to Fixed-Parameter Algorithms. Oxford University Press, 2006.535-537.
  - J. Guo, R. Niedermeier. Invitation to data reduction and problem kernelization. ACM SIGACT News 38(1) (2007) 31-45.
  - H.L. Bodlaender. Kernelization: New upper and lower bound techniques. Lecture Notes in Computer Science 5917 (2009) 17-37.

- **参照文献**

  - J.F. Buss, J. Goldsmith. Nondeterminism within P. SIAM Journal on Computing 22 (1993) 560-572.
  - M. Dom, J. Guo, F. Hüffner, R. Niedermeier, A. Truss. Fixed-parameter tractability results for feedback set problems in tournaments. Journal of Discrete Algorithms 8 (2010) 76-86.
  - F. Hüffner, R. Niedermeier, S. Wernicke. Techniques for practical fixed-parameter algorithms. The Computer Journal 51 (2008) 7-25.

目次