離散最適化基礎論 (2025 年後学期)

高速指数時間アルゴリズム

第 8 回

包除原理 (2)：例

岡本 吉央　(電気通信大学)

okamotoy@uec.ac.jp

2025 年 12 月 9 日

最終更新：2025 年 12 月 10 日　09:01

スケジュール (前半)

1.	高速指数時間アルゴリズムの考え方	(10/7)
∗	休み (体育祭)	(10/14)
2.	分枝アルゴリズム：基礎	(10/21)
3.	分枝アルゴリズム：高速化	(10/28)
4.	分枝アルゴリズム：測度統治法	(11/4)
5.	動的計画法：基礎	(11/11)
6.	動的計画法：例	(11/18)

スケジュール (後半)

7. 包除原理：原理 (11/25)
* 休み (秋ターム試験) (12/2)
8. **包除原理：例** (12/9)
9. 部分集合たたみ込み：原理 (12/16)
* 休み (出張) (12/23)
* 休み (冬季休業) (12/30)
10. 部分集合たたみ込み：例 (1/6)
11. 指数時間仮説：原理 (1/13)
12. 指数時間仮説：証明 (1/20)
13. 最近の話題 (1/27)
* 休み (修士論文発表会) (2/3)

包除原理：一般形

記法

- $A_1, A_2, \ldots, A_n \subseteq U$
- $[n] = \{1, 2, \ldots, n\}$
- 任意の $S \subseteq [n]$ に対して，$A_S = \bigcap_{i \in S} A_i$

定理：包除原理 (一般の n)

$$\left| \overline{\bigcup_{i \in [n]} A_i} \right| = \sum_{S \subseteq [n]} (-1)^{|S|} |A_S|$$

<u>注意</u>：$A_\emptyset = U$

本日の内容

1. **復習：彩色問題**
2. 包除原理に基づく彩色アルゴリズム
3. 染色多項式

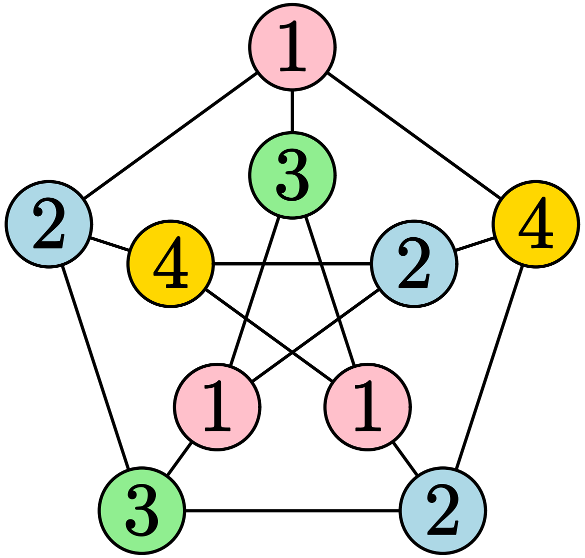
グラフの彩色

無向グラフ $G = (V, E)$

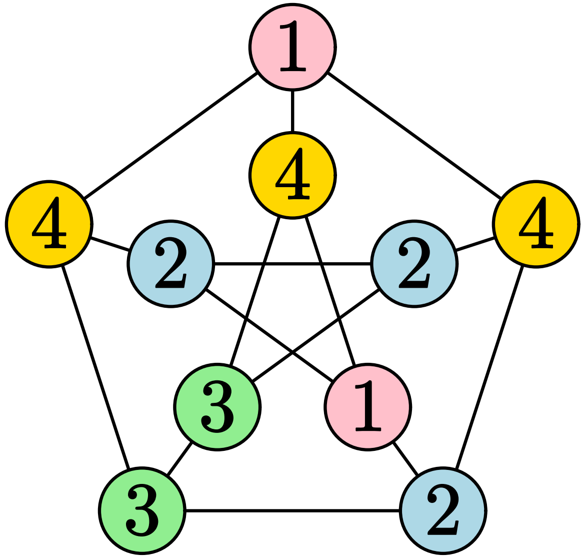
定義：彩色 (coloring)

G の **彩色** (さいしょく) とは,
写像 $c\colon V \to \{1, 2, \ldots\}$ で次を満たすもののこと

$$\{u, v\} \in E \quad \Rightarrow \quad c(u) \neq c(v)$$



彩色である



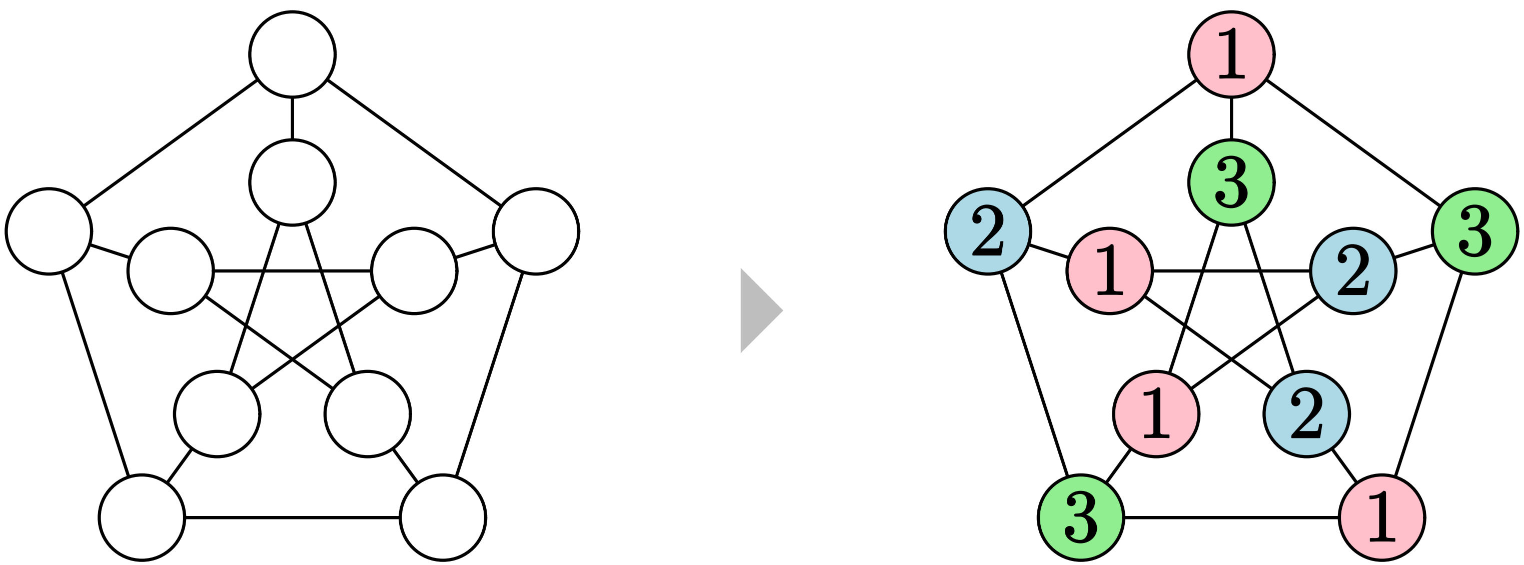
彩色ではない

彩色問題

定義：彩色問題

入力： 無向グラフ $G = (V, E)$
出力： G の彩色 c で，$\max\{c(v) \mid v \in V\}$ が最小のもの

「最小彩色問題」「グラフ彩色問題」とも言う



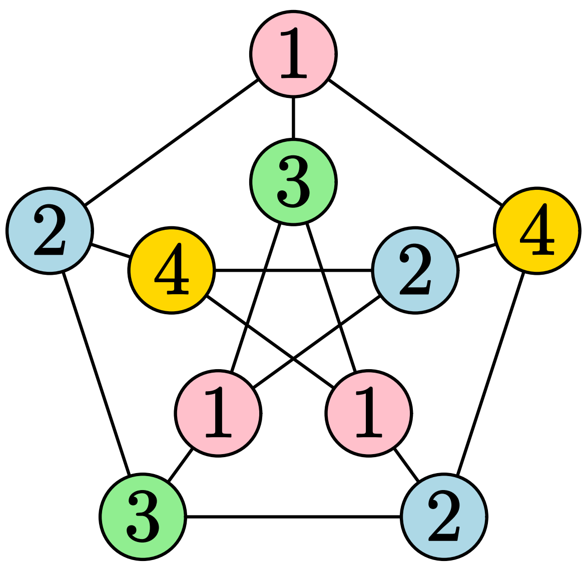
事実：彩色問題は NP 困難 (Karp '72)

彩色と独立集合

無向グラフ $G = (V, E)$, 彩色 $c\colon V \to \{1, 2, \ldots\}$

観察

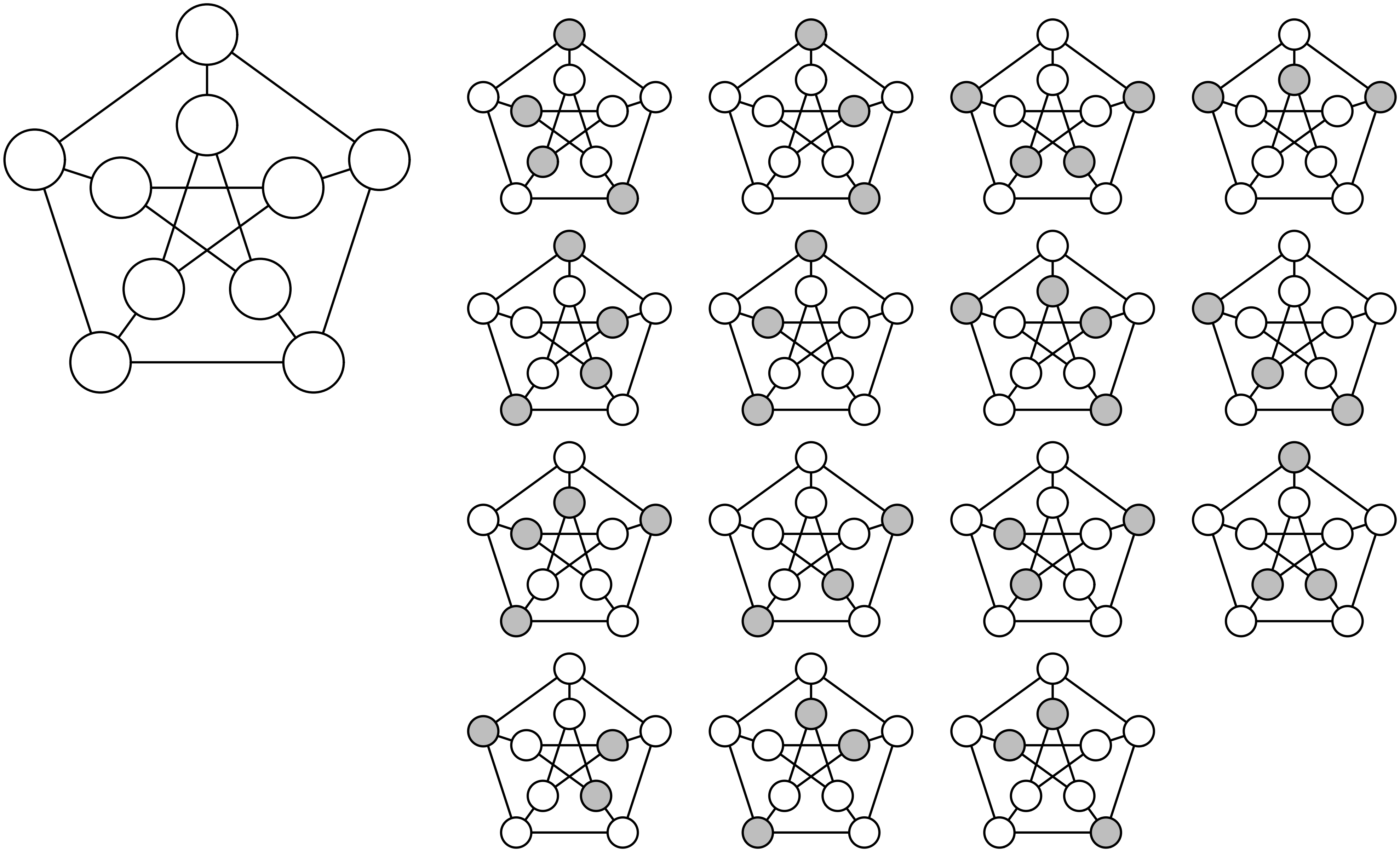
彩色 c によって同じ色で塗られた頂点の集合は
G の独立集合



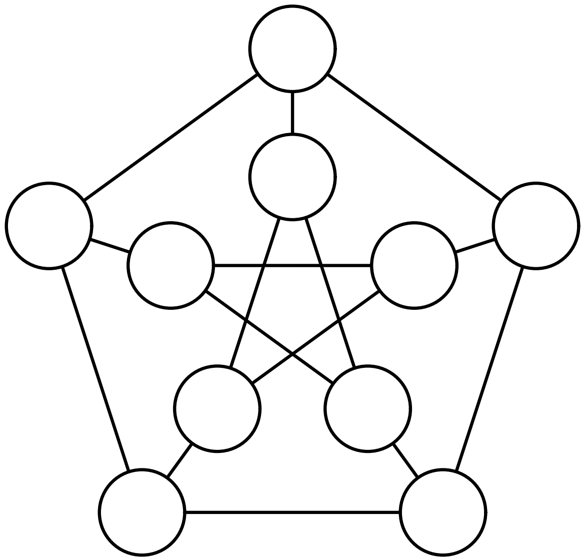
同じ色で塗られた頂点の集合
$= c^{-1}(\{i\})$ $(i \in \{1, 2, \ldots\})$

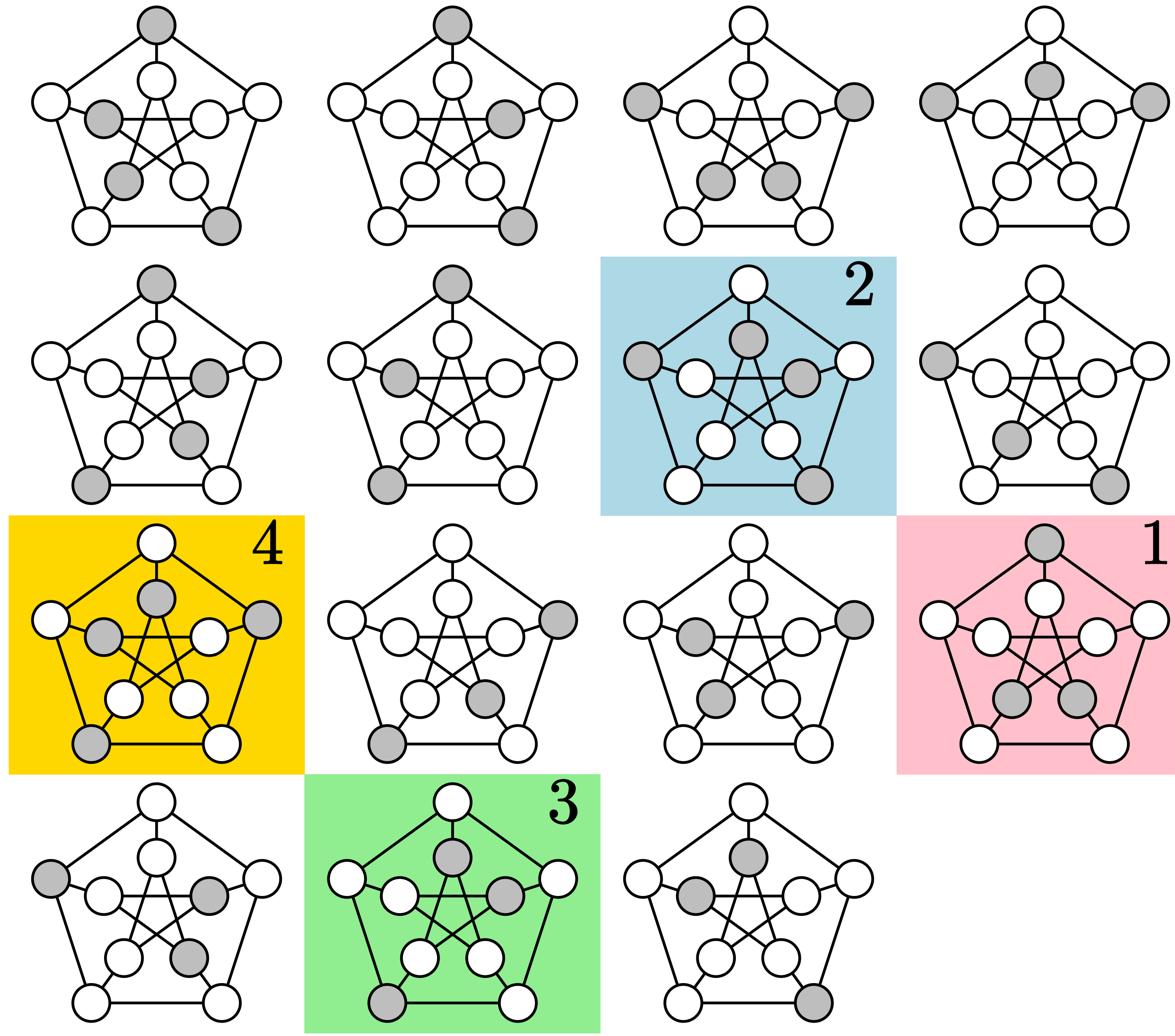
復習：G の **独立集合** とは
G で隣接しない頂点の集合

彩色は独立集合による被覆

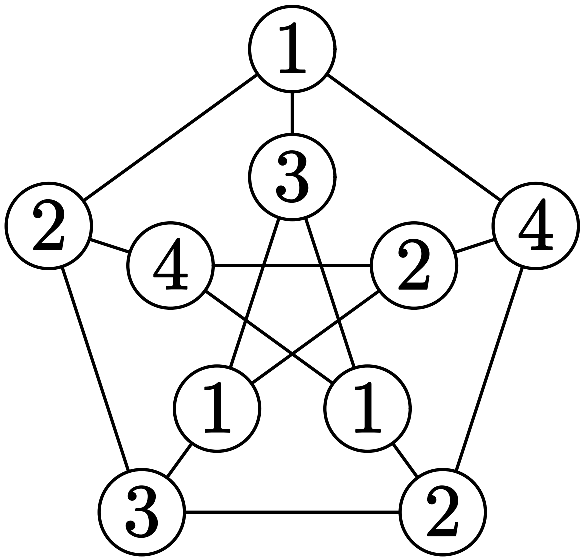


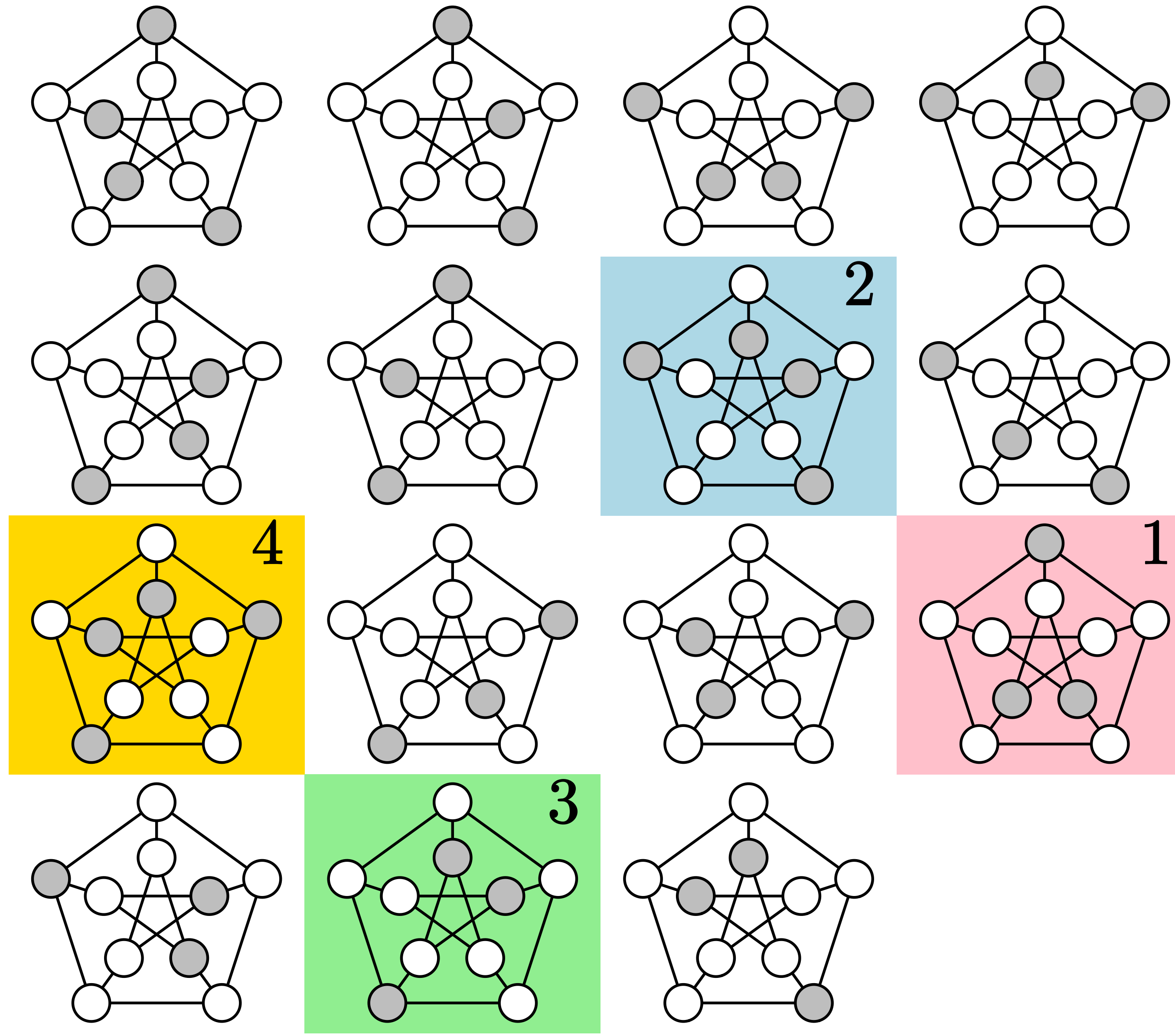
彩色は独立集合による被覆





彩色は独立集合による被覆





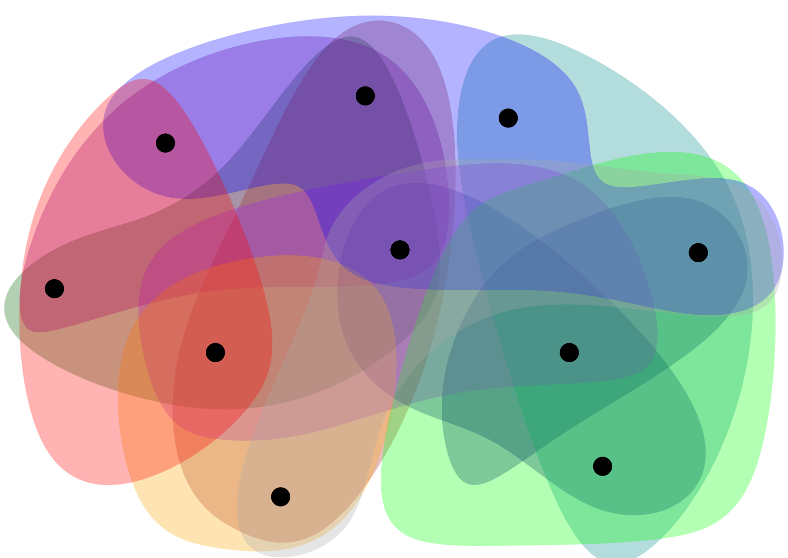
被覆

設定

- 有限集合 V
- 集合族 $\mathcal{S} \subseteq 2^V$

定義：被覆 (cover)

V の **被覆** とは，次を満たす $\mathcal{S}' = \{X_1, X_2, \ldots, X_k\} \subseteq \mathcal{S}$
- $X_1 \cup X_2 \cup \cdots \cup X_k = V$



$\mathcal{S}'$ は V を **被覆する** ともいう
(**覆う**)

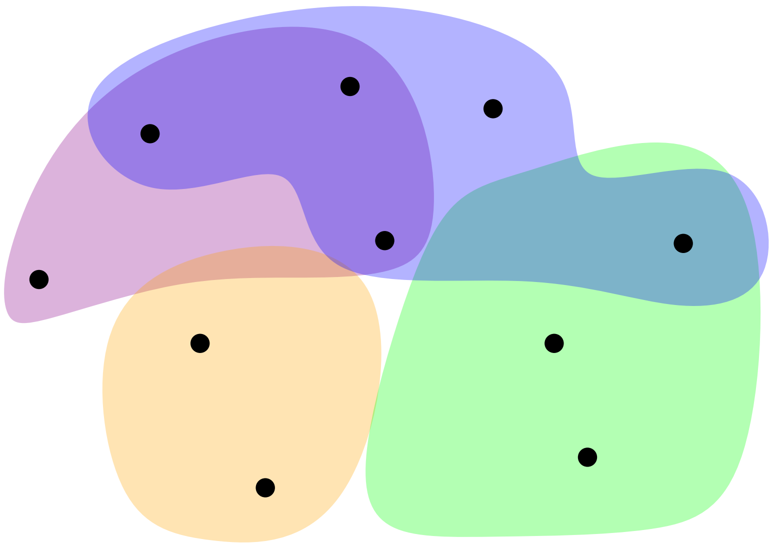
設定

- 有限集合 V
- 集合族 $\mathcal{S} \subseteq 2^V$

定義：被覆 (cover)

V の **被覆** とは，次を満たす $\mathcal{S}' = \{X_1, X_2, \ldots, X_k\} \subseteq \mathcal{S}$

- $X_1 \cup X_2 \cup \cdots \cup X_k = V$



$\mathcal{S}'$ は V を **被覆する** ともいう
(**覆う**)

本日の内容

1. 復習：彩色問題
2. **包除原理に基づく彩色アルゴリズム**
3. 染色多項式

- A. Björklund, T. Husfeldt, M. Koivisto. Set partitioning via inclusion-exclusion. *SIAM Journal on Computing* 39 (2009) pp. 546–563.

彩色問題を包除原理で解く

目標：包除原理を用いて次の定理を導く

定理 (Björklund, Husfeldt, Koivisto ’09)

彩色問題は $O^*(2^n)$ 時間 で解ける
(n はグラフの頂点数)

第 6 回の内容：$O^*(2.4423^n)$ 時間 (動的計画法による)

包除原理によるアルゴリズムの考え方

0. 数え上げ問題として見なす
1. U と A_i を上手に定める
2. $|A_S|$ の計算法を与える

グラフの k 彩色

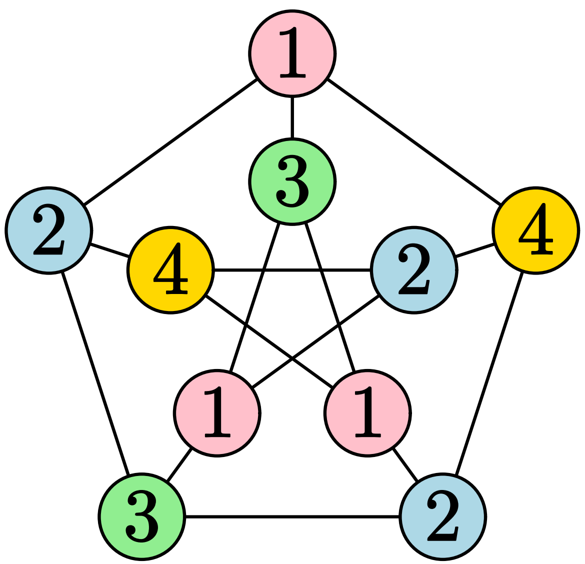
無向グラフ $G=(V,E)$

定義：k 彩色 (k-coloring)

G の k **彩色** とは,
写像 $c\colon V \to \{1,2,\ldots,k\}$ で次を満たすもののこと

$$\{u,v\} \in E \quad\Rightarrow\quad c(u) \neq c(v)$$

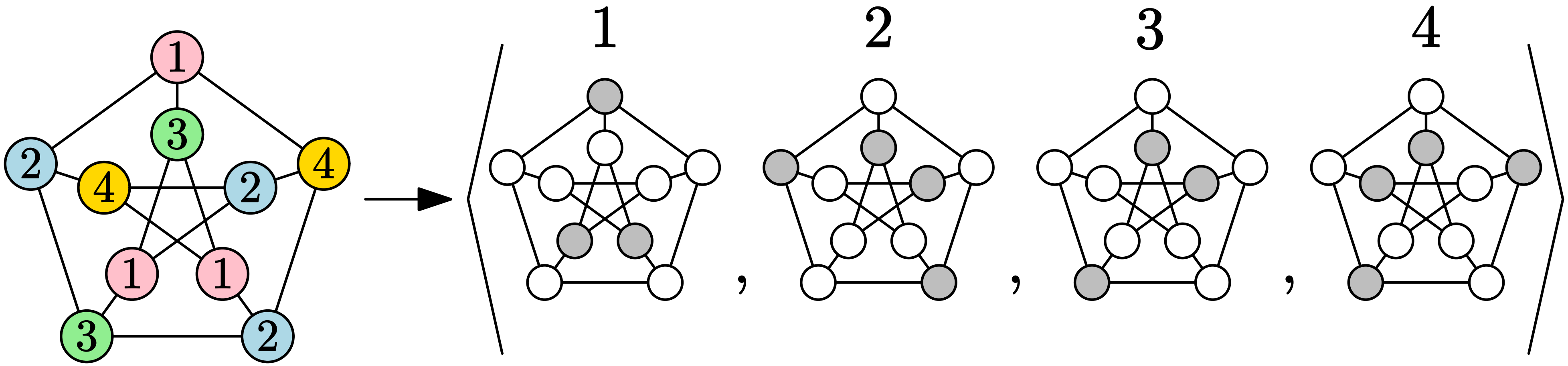
直感：k 色しか使わない彩色



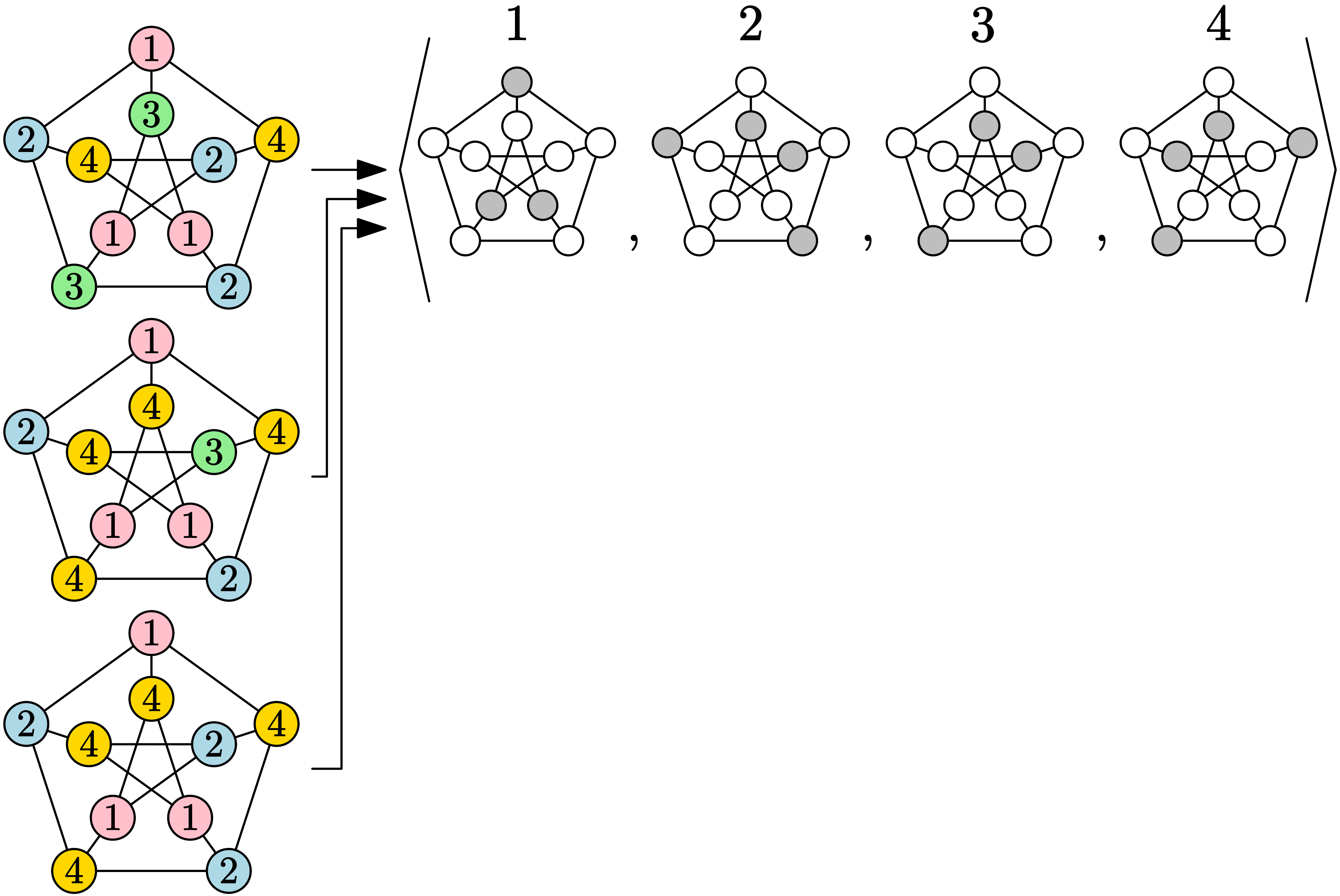
4 彩色であるが,

3 彩色ではない

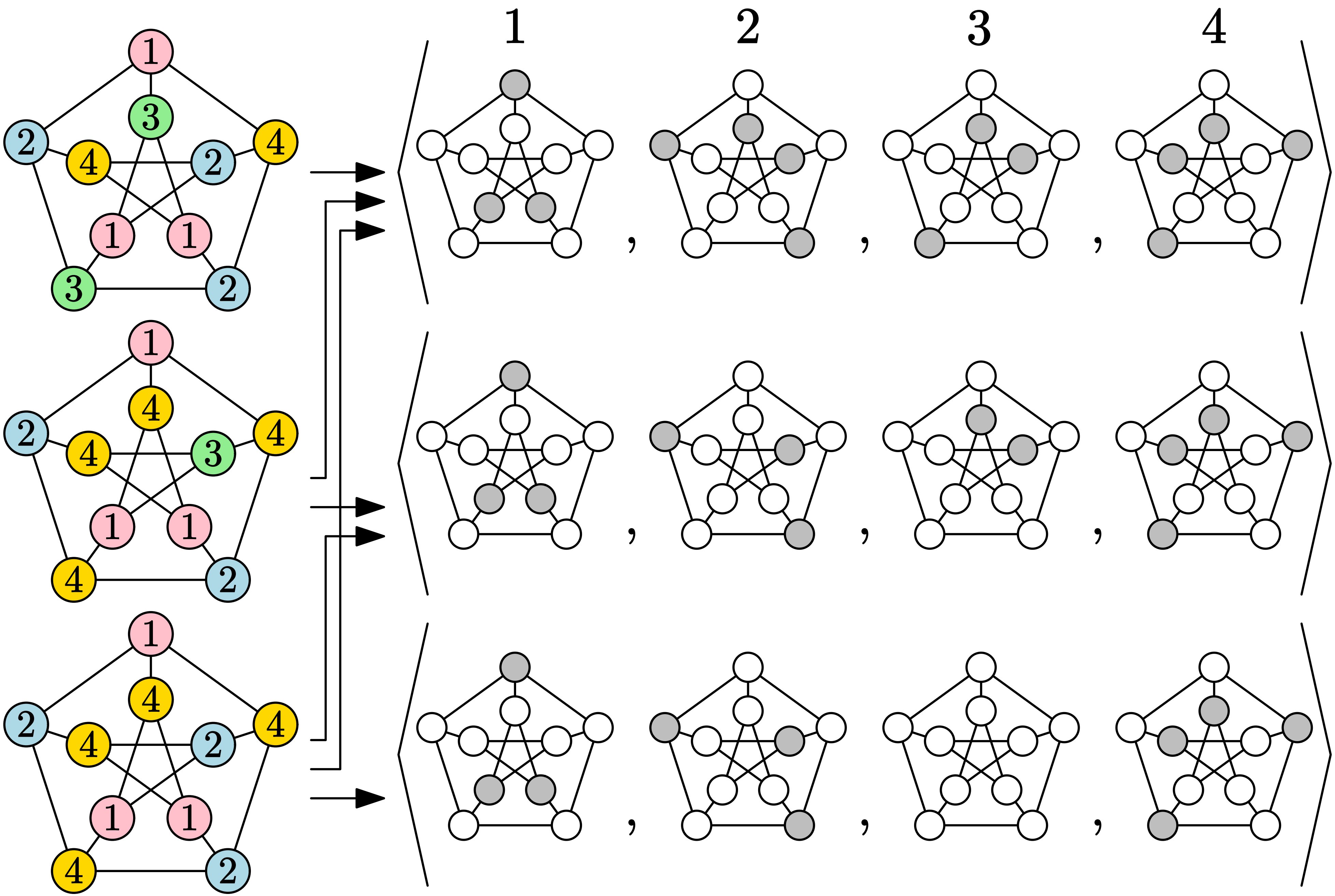
彩色と独立集合被覆列



彩色と独立集合被覆列



彩色と独立集合被覆列



独立集合被覆列

無向グラフ $G = (V, E)$

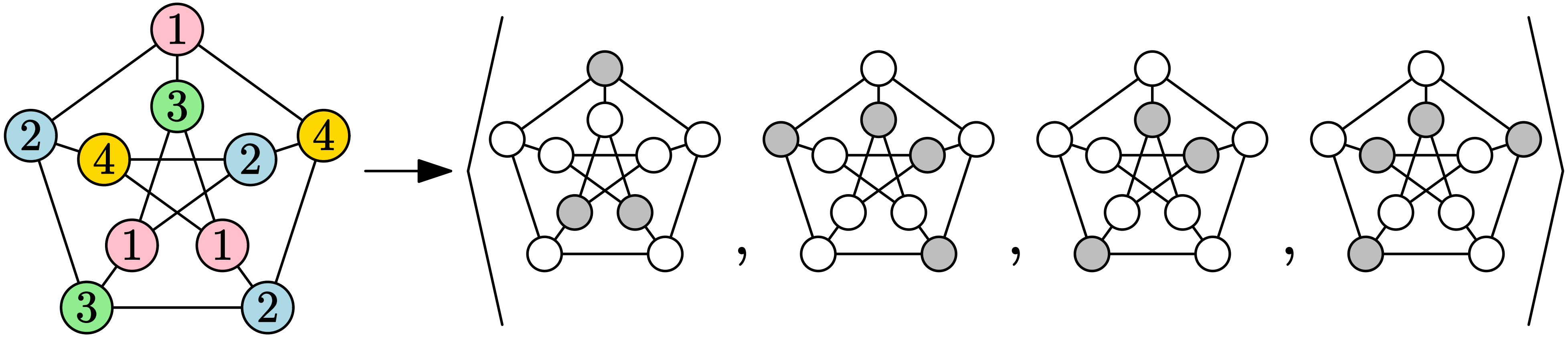
定義：独立集合被覆列

G の **独立集合被覆列** とは,
列 $\langle I_1, I_2, \ldots, I_k \rangle$ で，各 I_j が G の独立集合であり,
$I_1 \cup I_2 \cup \cdots \cup I_k = V$ を満たすもの

注意：$I_1 = I_2$ であってもよい，$I_1 = \emptyset$ であってもよい

観察：G の k 彩色が存在

$\Leftrightarrow$ G の独立集合被覆列で長さ k のものが存在

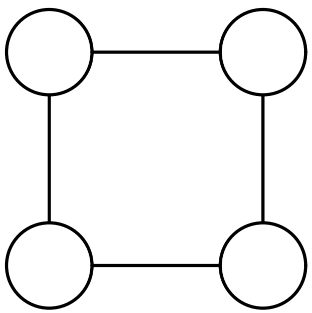


独立集合被覆列の数え上げ

定義：独立集合被覆列の数え上げ問題

入力： 無向グラフ $G = (V, E)$，非負整数 k

出力： G の独立集合被覆列で長さ k のものの 総数



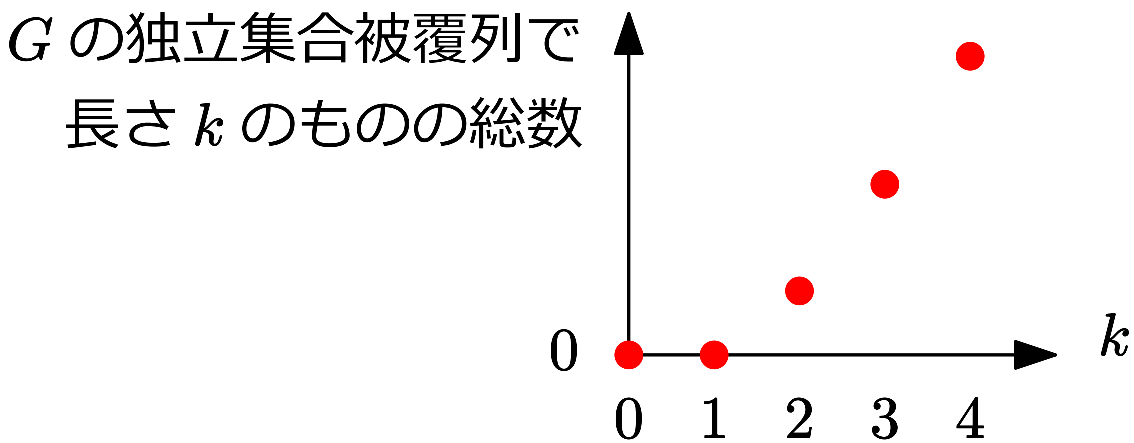
$k = 3$

54

独立集合被覆列を包除原理で数え上げる

実際に導出すること

独立集合被覆列の数え上げ問題は $O^*(2^n)$ 時間 で解ける
(n はグラフの頂点数)



$\rightsquigarrow$ G の最小彩色が使う色数は 2

定理：再掲 (Björklund, Husfeldt, Koivisto ’09)

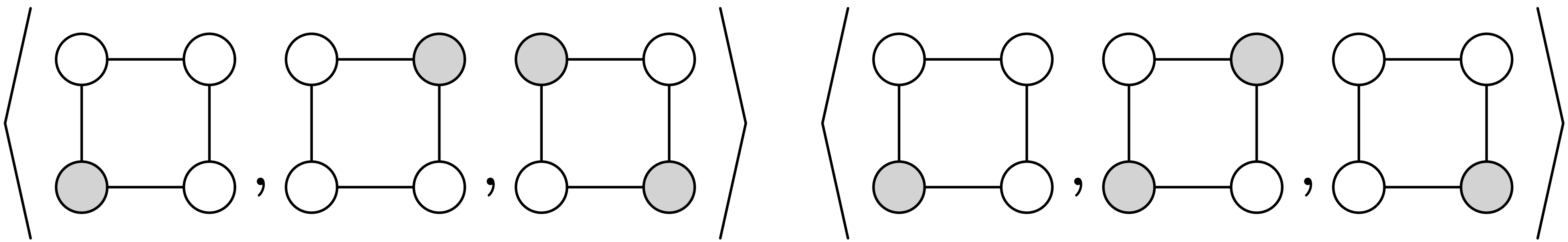
彩色問題は $O^*(2^n)$ 時間 で解ける
(n はグラフの頂点数)

包除原理：U を定める

ポイント：U では 数えすぎる ようにする

$$U = \{\langle I_1, I_2, \ldots, I_k \rangle \mid I_j \text{ は } G \text{ の独立集合 } \forall j \in \{1, 2, \ldots, k\}\}$$

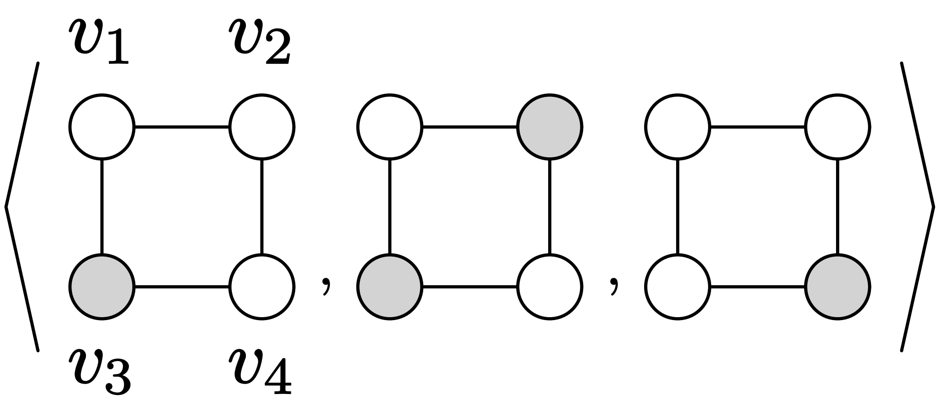
つまり，$I_1 \cup I_2 \cup \cdots \cup I_k = V$ を要請しない



包除原理：A_i を定める

$V=\{v_1, v_2, \ldots, v_n\}$ として，$i \in \{1, 2, \ldots, n\}$ に対して

$$A_i = \left\{ \langle I_1, I_2, \ldots, I_k \rangle \;\middle|\; \begin{array}{l} I_j \text{ は } G \text{ の独立集合 } \forall\, j \in \{1, \ldots, k\} \\ v_i \notin I_1 \cup I_2 \cup \cdots \cup I_k \end{array} \right\}$$



$\in A_1$

このとき

$\langle I_1, I_2, \ldots, I_k \rangle$ が G の独立集合被覆列 $\Leftrightarrow$ $\langle I_1, I_2, \ldots, I_k \rangle \in \overline{A_1 \cup A_2 \cup \cdots \cup A_n}$

包除原理：$|A_S|$ の計算法 (1)

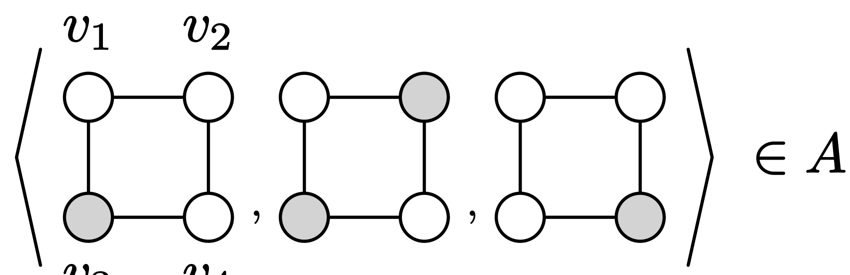
動的計画法！

動的計画法を考えるときの鍵 (数え上げバージョン)

1. 数え上げ対象の持つ **再帰的な構造** を見出す
2. 上の構造から **状態** を適切に定義する
3. 状態の間の **再帰式** を立てる

包除原理：$|A_S|$ の計算法 (1)

動的計画法！



$$\langle I_1, I_2, \ldots, I_k \rangle \in A_S = \bigcap_{i \in S} A_i$$

$$\Leftrightarrow v_i \notin I_j \quad \forall\, i \in S, \forall\, j \in \{1, 2, \ldots, k\}$$

$\Leftrightarrow I_j$ は $G - X$ の独立集合　$\forall\, j \in \{1, 2, \ldots, k\}$

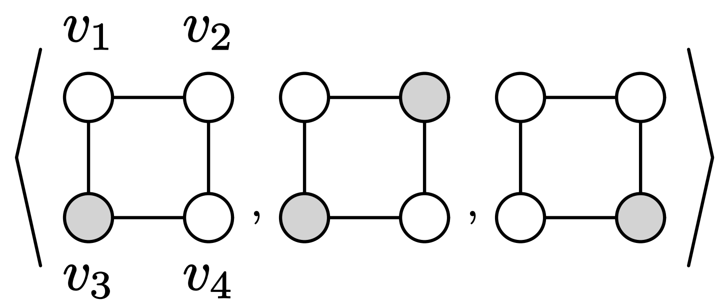
ただし，$X = \{v_i \mid i \in S\}$

動的計画法を考えるときの鍵 (数え上げバージョン)

1. 数え上げ対象の持つ **再帰的な構造** を見出す
2. 上の構造から **状態** を適切に定義する
3. 状態の間の **再帰式** を立てる

包除原理：$|A_S|$ の計算法 (1)

動的計画法！



$\in A_1$

$$\langle I_1, I_2, \ldots, I_k \rangle \in A_S = \bigcap_{i \in S} A_i$$

$\Leftrightarrow v_i \notin I_j \quad \forall\, i \in S, \forall\, j \in \{1, 2, \ldots, k\}$

$\Leftrightarrow I_j$ は $G - X$ の独立集合　$\forall\, j \in \{1, 2, \ldots, k\}$

　ただし，$X = \{v_i \mid i \in S\}$

$\therefore |A_S| = (G - X$ の独立集合の総数$)^k$

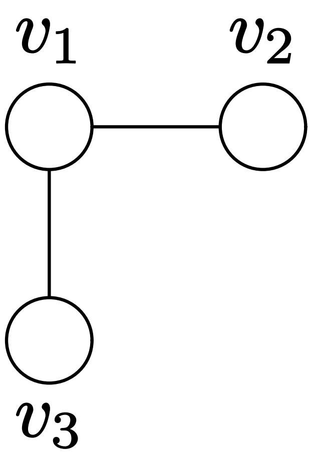
動的計画法を考えるときの鍵 (数え上げバージョン)

1. 数え上げ対象の持つ **再帰的な構造** を見出す
2. 上の構造から **状態** を適切に定義する
3. 状態の間の **再帰式** を立てる

包除原理：$|A_S|$ の計算法 (2)

次の問題を解ければ，$|A_S|$ も計算できる

入力： 無向グラフ $G = (V, E)$
出力： すべての $X \subseteq V$ に対する，$G - X$ の独立集合の総数



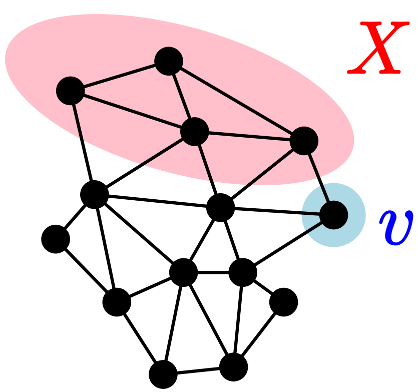
X	$G - X$ の独立集合の総数
$\emptyset$	5
$\{v_1\}$	4
$\{v_2\}$	3
$\{v_3\}$	3
$\{v_1, v_2\}$	2
$\{v_1, v_3\}$	2
$\{v_2, v_3\}$	2
$\{v_1, v_2, v_3\}$	1

包除原理：$|A_S|$ の計算法 (3)

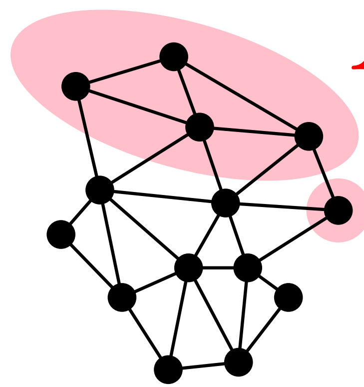
<u>記法</u>：$a(X) = G - X$ の独立集合の総数

$a(X)$ に対する再帰式

$$a(X) = \begin{cases} 1 & (X = V \text{ のとき}) \\ a(X \cup \{v\}) + a(X \cup N[v]) & (v \notin X \text{ のとき}) \end{cases}$$

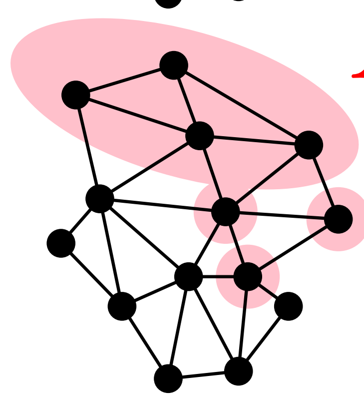


$X \cup \{v\}$



v を含まない独立集合

$X \cup N[v]$

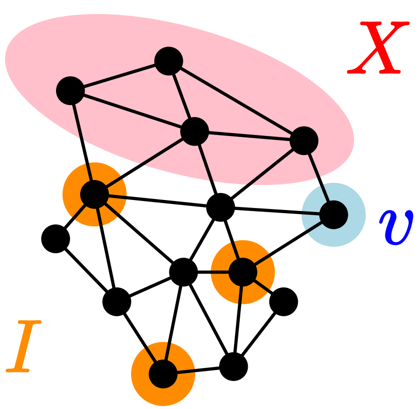


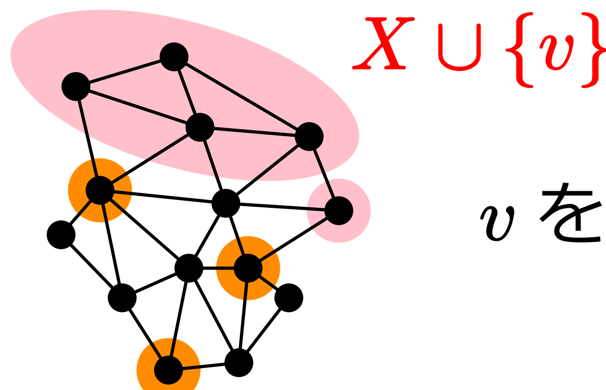
v を含む独立集合

<u>記法</u>：$a(X) = G - X$ の独立集合の総数

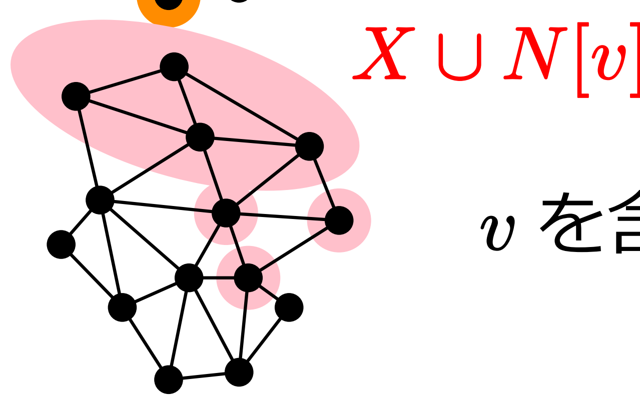
$a(X)$ に対する再帰式

$$a(X) = \begin{cases} 1 & (X = V \text{ のとき}) \\ a(X \cup \{v\}) + a(X \cup N[v]) & (v \notin X \text{ のとき}) \end{cases}$$





v を含まない独立集合



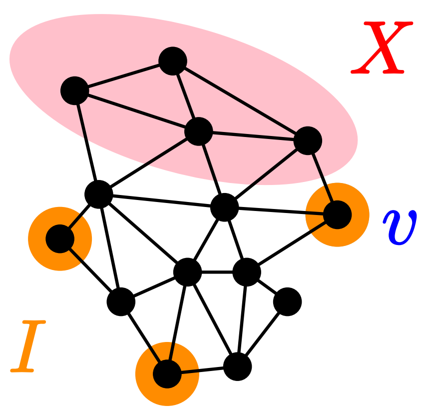
v を含む独立集合

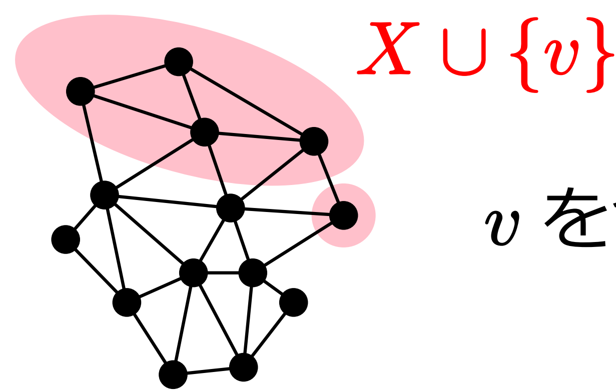
包除原理：$|A_S|$ の計算法 (3)

記法：$a(X) = G - X$ の独立集合の総数

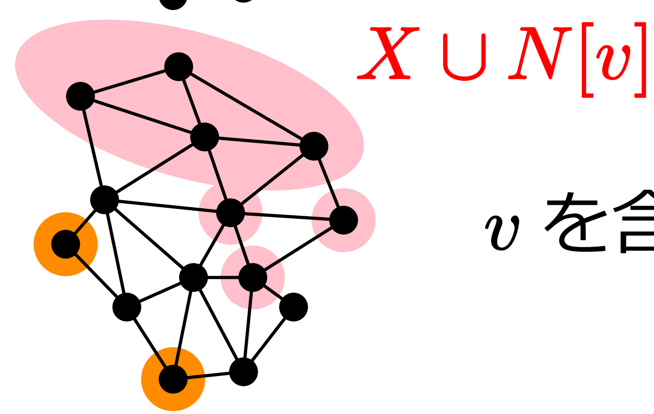
$a(X)$ に対する再帰式

$$a(X) = \begin{cases} 1 & (X = V \text{ のとき}) \\ a(X \cup \{v\}) + a(X \cup N[v]) & (v \notin X \text{ のとき}) \end{cases}$$





v を含まない独立集合



v を含む独立集合

包除原理：$|A_S|$ の計算法 (4) まとめ

アルゴリズム is-dp$(G = (V, E))$ // $n = |V|$

1. $a(V) = 1$
2. すべての $X \subseteq V, X \neq V$ に対して，
   $|X|$ の大きい方から順に，$a(X)$ を再帰式に基づいて計算
3. すべての $X \subseteq V$ に対して $a(X)$ を出力

$\boxed{\text{再帰式}}$： $a(X) = a(X \cup \{v\}) + a(X \cup N[v])$

包除原理：$|A_S|$ の計算法 (4) まとめ

アルゴリズム is-dp$(G = (V, E))$ // $n = |V|$

1. $a(V) = 1$
2. すべての $X \subseteq V, X \neq V$ に対して，
   $|X|$ の大きい方から順に，$a(X)$ を再帰式に基づいて計算
3. すべての $X \subseteq V$ に対して $a(X)$ を出力

$\boxed{\text{再帰式}}$： $a(X) = a(X \cup \{v\}) + a(X \cup N[v])$

$\therefore$ 全体の計算量 $= O^*(2^n)$，全体のメモリ使用量 $= O^*(2^n)$

包除原理：アルゴリズム

アルゴリズム col$(G = (V, E), k)$

1. sum $= 0$ // 初期化
2. $a =$ is-dp(G) // 独立集合の数え上げ
3. 各 $X \subseteq V$ に対して，次を実行
   (a) term $= a(X)^k$
   (b) sum $=$ sum $+ (-1)^{|X|}$ term
4. sum を出力

包除原理：アルゴリズム

アルゴリズム col$(G = (V, E), k)$

1. sum $= 0$ // 初期化
2. $a =$ is-dp(G) // 独立集合の数え上げ
3. 各 $X \subseteq V$ に対して，次を実行
   (a) term $= a(X)^k$
   (b) sum $=$ sum $+ (-1)^{|X|}$ term
4. sum を出力

彩色問題では，$k \leq n$

ビット長 $= O(\log a(X)^k) = O(\log(2^n)^k) = O(nk) = O(n^2)$

包除原理：アルゴリズム

アルゴリズム col$(G=(V,E),k)$

1. sum $=0$　// 初期化
2. $a=$ is-dp(G)　// 独立集合の数え上げ　$O^*(2^n)$
3. 各 $X \subseteq V$ に対して，次を実行　$O^*(2^n)$回の繰返し
   (a) term $= a(X)^k$
   (b) sum $=$ sum $+ (-1)^{|X|}$ term

   $O^*(1)$
4. sum を出力

彩色問題では，$k \le n$

ビット長 $= O(\log a(X)^k) = O(\log (2^n)^k) = O(nk) = O(n^2)$

$\therefore$ 計算量 $= O^*(2^n)$　　メモリ使用量 $= O^*(2^n)$

包除原理を用いて次の定理を導いた

定理 (Björklund, Husfeldt, Koivisto '09)

彩色問題は $O^*(2^n)$ 時間 で解ける
(n はグラフの頂点数)

第 6 回の内容：$O^*(2.4423^n)$ 時間 (動的計画法による)

未解決問題

次を満たす定数 $c < 2$ は存在するか？

彩色問題は $O^*(c^n)$ 時間 で解ける

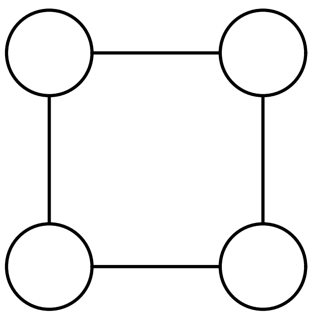
本日の内容

1. 復習：彩色問題
2. 包除原理に基づく彩色アルゴリズム
3. **染色多項式**

k 彩色の数え上げ

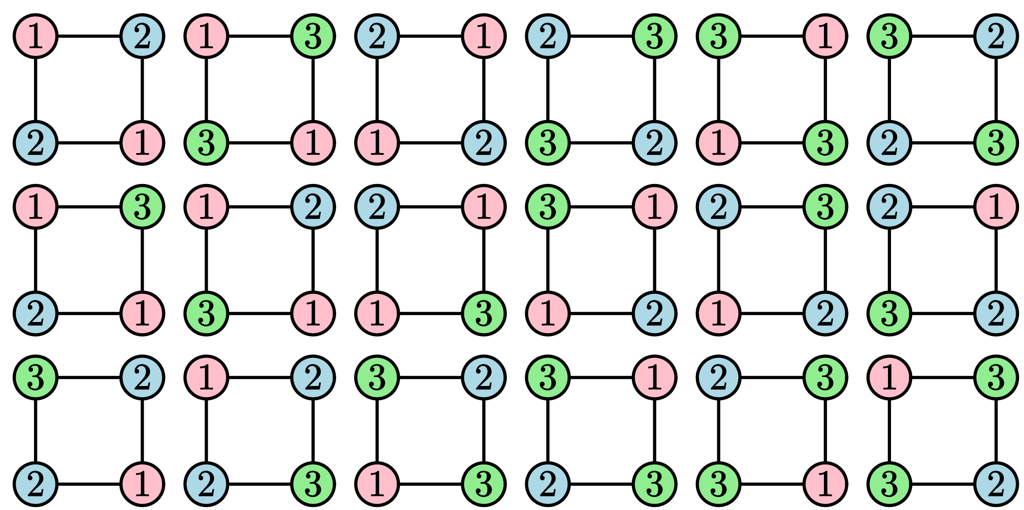
定義：k 彩色の数え上げ問題

入力： 無向グラフ $G = (V, E)$，非負整数 k
出力： G の k 彩色の <u>総数</u>



$k = 3$

18



<u>注意</u>：長さ 3 の独立集合被覆列の総数 $= 54$

k 彩色の数え上げ：予告

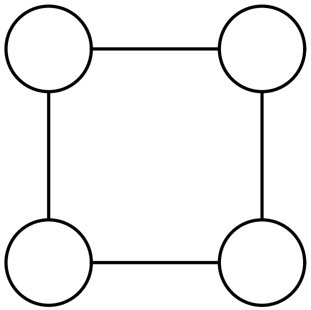
第 10 回の授業で次の定理を導出する (予定)

定理 (Björklund, Husfeldt, Kaski, Koivisto '08)

k 彩色の数え上げ問題は $O^*(2^n)$ 時間 で解ける
(n はグラフの頂点数)

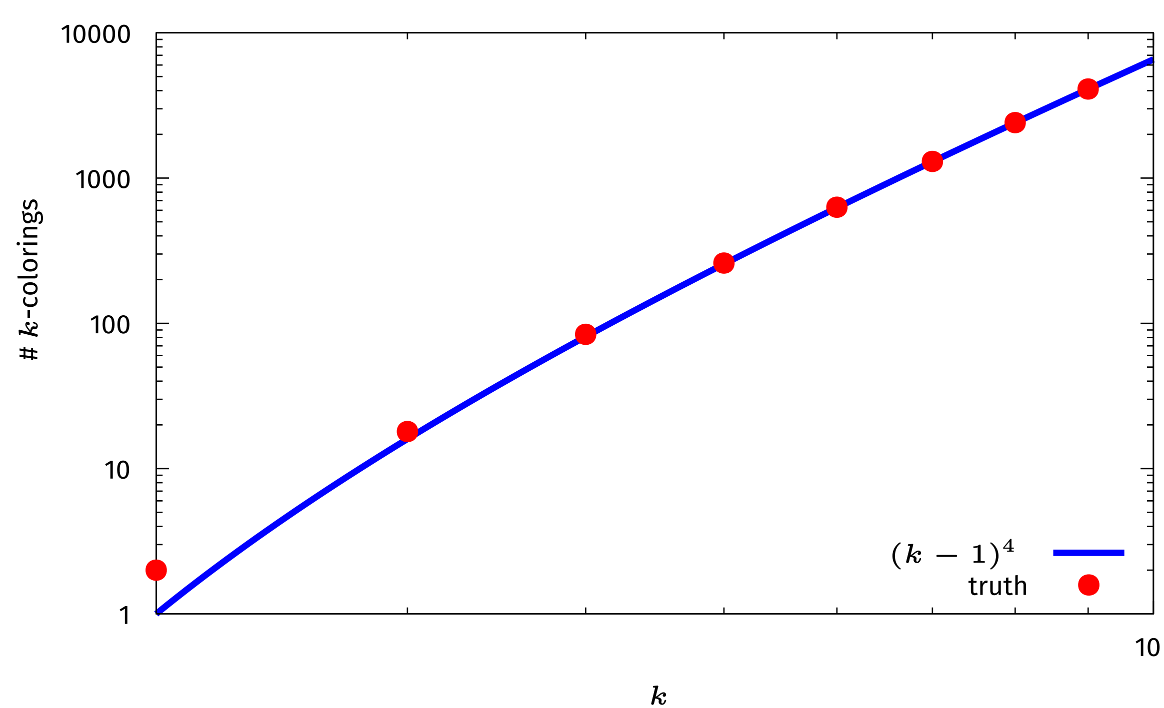
残りの時間で，「k 彩色の数え上げ」が満たす性質を紹介する

k 彩色の数え上げ：例



k	k 彩色の総数
1	0
2	2
3	18
4	84
5	260
6	630
7	1 302
8	2 408
9	4 104

両対数プロット

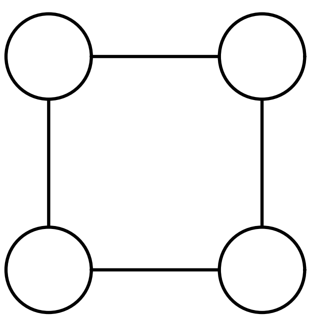


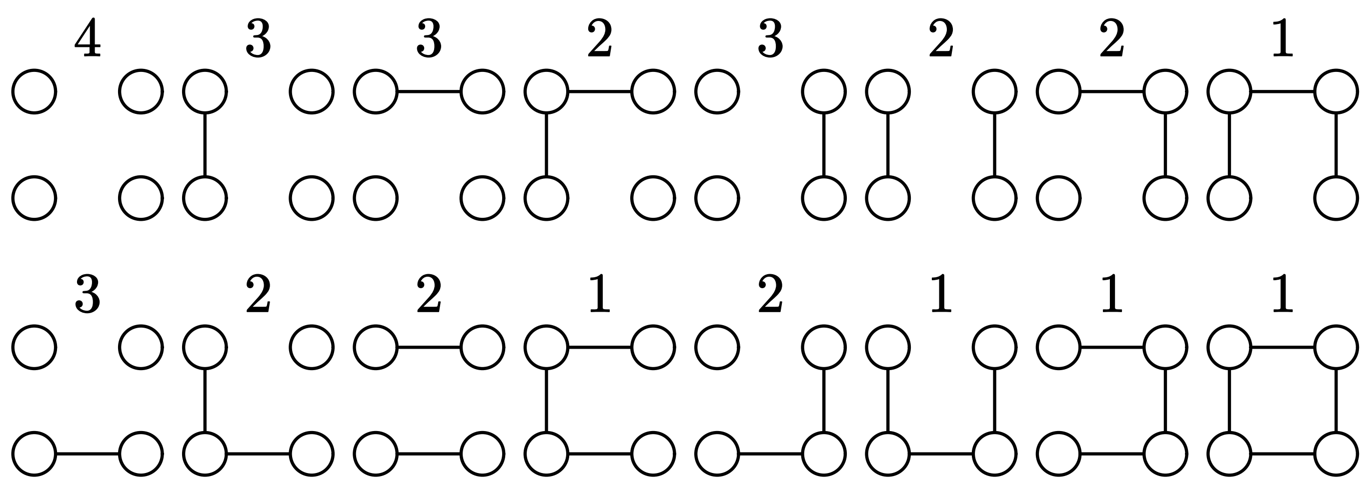
定理：Whitney の公式

無向グラフ $G = (V, E)$ の k 彩色の総数は次で計算できる

$$\sum_{F \subseteq E} (-1)^{|F|} k^{c(F)}$$

ここで，$c(F)$ は無向グラフ (V, F) の連結成分の総数





$\therefore$ k 彩色の総数 $= k^4 - 4k^3 + 6k^2 - 3k$

染色多項式

定理：Whitney の公式

無向グラフ $G = (V, E)$ の k 彩色の総数は次で計算できる

$$\sum_{F \subseteq E} (-1)^{|F|} k^{c(F)}$$

ここで，$c(F)$ は無向グラフ (V, F) の連結成分の総数

帰結：G の k 彩色の総数は k に関する多項式

(G の **染色多項式** (chromatic polynomial) と呼ぶ)

Whitney の公式：包除原理による証明

定理：Whitney の公式

無向グラフ $G=(V,E)$ の k 彩色の総数は次で計算できる

$$\sum_{F \subseteq E} (-1)^{|F|} k^{c(F)}$$

ここで，$c(F)$ は無向グラフ (V,F) の連結成分の総数

<u>証明</u>：包除原理を用いる

包除原理によるアルゴリズムの考え方

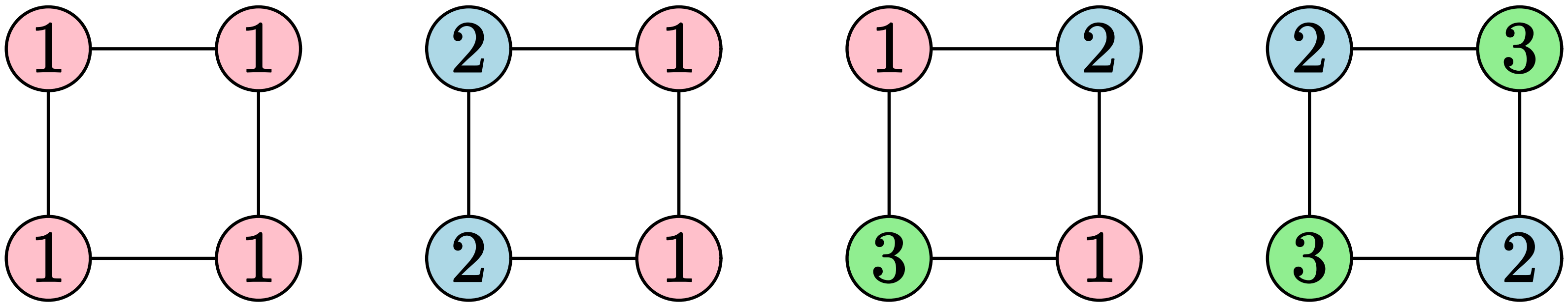
1. U と A_i を上手に定める
2. $|A_S|$ の計算法を与える

Whitney の公式：U を定める

ポイント：U では 数えすぎる ようにする

$$U = \{c \mid c\colon V \to \{1, 2, \ldots, k\}\}$$

つまり，辺 $\{u, v\} \in E$ に対して $c(u) \neq c(v)$ を要請しない



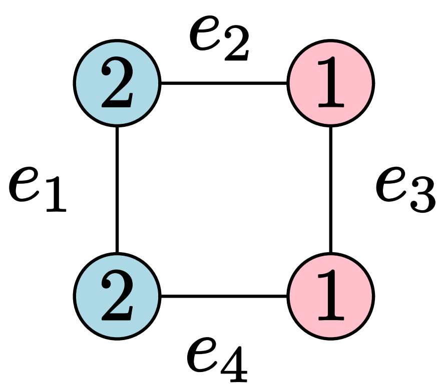
このとき，$|U| = k^{|V|}$

Whitney の公式：A_i を定める

$E = \{e_1, e_2, \ldots, e_m\}$ として，$i \in \{1, 2, \ldots, m\}$ に対して

$$A_i = \{c \mid c \colon V \to \{1, 2, \ldots, k\}, c(u_i) = c(v_i)\}$$

ただし，$e_i = \{u_i, v_i\}$ とする



$\in A_1 \cap A_3$

このとき，c が G の彩色 $\Leftrightarrow c \in \overline{A_1 \cup A_2 \cup \cdots \cup A_m}$

Whitney の公式：$|A_S|$ の計算法

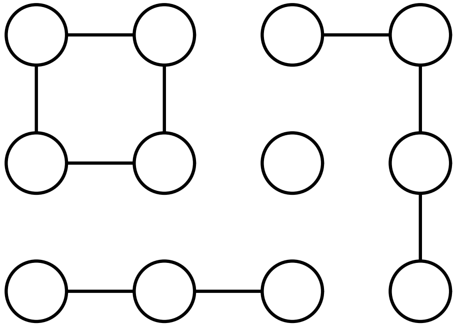
$E = \{e_1, e_2, \ldots, e_m\}$ として，$i \in \{1, 2, \ldots, m\}$ に対して

$$A_i = \{c \mid c\colon V \to \{1, 2, \ldots, k\}, c(u_i) = c(v_i)\}$$

ただし，$e_i = \{u_i, v_i\}$ とする

$S \subseteq \{1, 2, \ldots, m\}$ に対して，$F = \{e_i \in E \mid i \in S\}$ とする

(これは一対一対応)



Whitney の公式：$|A_S|$ の計算法

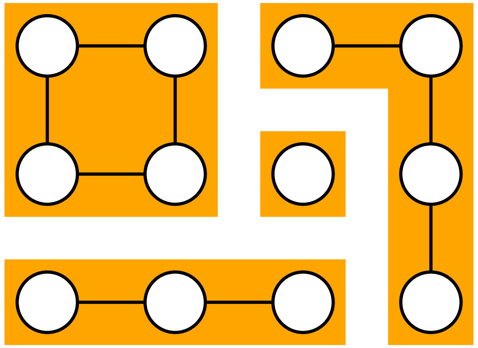
$E = \{e_1, e_2, \ldots, e_m\}$ として，$i \in \{1, 2, \ldots, m\}$ に対して

$$A_i = \{c \mid c \colon V \to \{1, 2, \ldots, k\}, c(u_i) = c(v_i)\}$$

ただし，$e_i = \{u_i, v_i\}$ とする

$S \subseteq \{1, 2, \ldots, m\}$ に対して，$F = \{e_i \in E \mid i \in S\}$ とする

(これは一対一対応)



$c(F) = 4$

$\rightsquigarrow |A_S| = k^{c(F)}$

Whitney の公式 (再掲)

定理：Whitney の公式 (再掲)

無向グラフ $G = (V, E)$ の k 彩色の総数は次で計算できる

$$\sum_{F \subseteq E} (-1)^{|F|} k^{c(F)}$$

ここで，$c(F)$ は無向グラフ (V, F) の連結成分の総数

<u>帰結</u>：k 彩色の総数は $O^*(2^{|E|})$ 時間で 計算できる

<u>第 10 回の授業</u>：k 彩色の総数を $O^*(2^{|V|})$ 時間で 計算する

まとめ

前回と今回

包除原理 (inclusion-exclusion principle) による
アルゴリズムの設計と解析

前回

- 包除原理の説明
  - 二部完全マッチングの数え上げ
  - ハミルトン路の数え上げ

今回

- 包除原理による彩色問題の解法 ($O^*(2^n)$ 時間)

次回の予告

次回と次々回

部分集合畳み込み (subset convolution) による
アルゴリズムの設計と解析

次回

- 部分集合畳み込みの説明
  - 最小シュタイナー木問題 ($O^*(2^{|K|})$ 時間)

次々回

- k 彩色の数え上げ ($O^*(2^n)$ 時間)