離散最適化基礎論 第 6 回
マトロイドに対する貪欲アルゴリズム

岡本 吉央
okamotoy@uec.ac.jp

電気通信大学

2015 年 11 月 27 日

最終更新：2015 年 11 月 27 日　11:23

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？

⇝ 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

マトロイドに対する貪欲アルゴリズムの応用を見る
- 割当問題 (の一種)
- ジョブ・スケジューリング問題 (の一種)

鍵となる概念：横断マトロイド

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

マトロイドとは？

$\mathcal{I}$ が E 上のマトロイド (matroid) であるとは，次の 3 条件を満たすこと

(I1) $\emptyset \in \mathcal{I}$

(I2) $X \in \mathcal{I}$ かつ $Y \subseteq X$ ならば，$Y \in \mathcal{I}$

(I3) $X, Y \in \mathcal{I}$ かつ $|X| > |Y|$ ならば，
ある $e \in X - Y$ が存在して，$Y \cup \{e\} \in \mathcal{I}$

補足
- (I1) と (I2) は $\mathcal{I}$ が独立集合族であることを意味する
- (I3) を増加公理 (augmentation property) と呼ぶことがある

用語
- $\mathcal{I}$ の要素である集合 $X \in \mathcal{I}$ を，このマトロイドの独立集合と呼ぶ

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

マトロイドと貪欲アルゴリズム

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

これによって解ける問題の例

- 最小全域木問題 (Kruskal のアルゴリズム $=$ 貪欲アルゴリズム)

今日は他の例を見る

目次

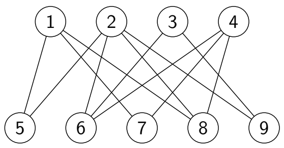
二部グラフ

ここで扱うグラフは，無向グラフで，並列辺や自己閉路を持たない

二部グラフとは？

無向グラフ $G=(V,E)$ が**二部グラフ** (bipartite graph) であるとは，
頂点集合 V の分割 $\{A,B\}$ (つまり，$A \cup B = V, A \cap B = \emptyset$) が存在して，
任意の辺 $e \in E$ に対して，e の一端点が A，他方が B の要素であるもの

$A=\{1,2,3,4\}, B=\{5,6,7,8,9\}$



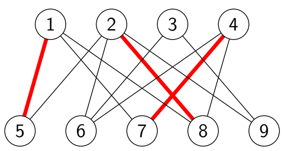
この分割を使って，$G=(A,B;E)$ や $G=(A,B,E)$ と表記することもある

二部グラフのマッチング

無向グラフ $G=(V,E)$

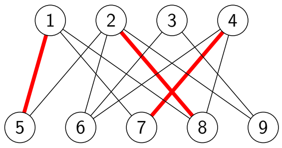
グラフのマッチングとは？ (復習)

G の**マッチング**とは，G の辺部分集合 $M \subseteq E$ で，
任意の頂点 $v \in V$ に対して，v に接続する M の辺が 1 つ以下であるもの



マッチングが飽和する頂点

マッチング M の辺の端点は，M によって**飽和**される (saturated) という



このマッチングが飽和する頂点は $1, 2, 4, 5, 7, 8$ で，
他の頂点は飽和されない

横断マトロイド

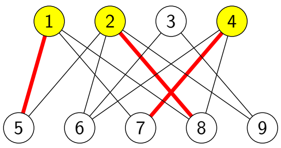
二部グラフ $G = (A, B; E)$

横断マトロイド (transversal matroid) とは？

G から得られる A 上の**横断マトロイド**とは，A 上のマトロイド $\mathcal{I}$ で，

$$X \in \mathcal{I} \quad \Leftrightarrow \quad X \text{ を飽和する } G \text{ のマッチングが存在する}$$

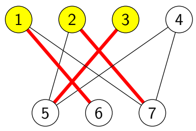
によって定義されるもの

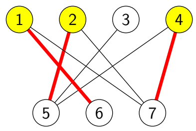


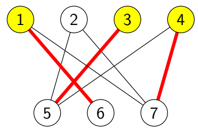
- $A = \{1, 2, 3, 4\}$
- $\{1, 2, 4\} \in \mathcal{I}$

横断マトロイド：例

$A = \{1, 2, 3, 4\}$







台集合を A とする横断マトロイドを考えると，その基族は

$$\mathcal{B} = \{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}\}$$

横断マトロイド：証明 (1)

今からやること

横断マトロイドが確かにマトロイドであることの確認

(I1), (I2) は簡単なので演習問題として，ここでは (I3) を確認する

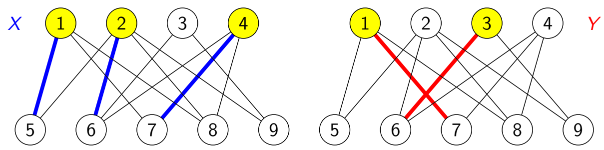
(I3) マトロイドの増加公理

$X, Y \in \mathcal{I}$ かつ $|X| > |Y|$ ならば，
ある $e \in X - Y$ が存在して，$Y \cup \{e\} \in \mathcal{I}$

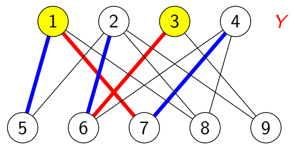
証明：$X, Y \in \mathcal{I}$ かつ $|X| > |Y|$ であると仮定

- 横断マトロイドの定義から，X を飽和するマッチング M と Y を飽和するマッチング N が存在
- $|X| > |Y|$ より，$|M| > |N|$

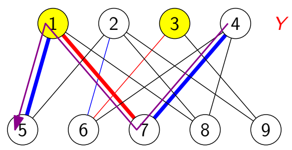
横断マトロイド：証明 (2)



ここで，$(M \cup N) - (M \cap N)$ (つまり，M と N の対称差) を考える



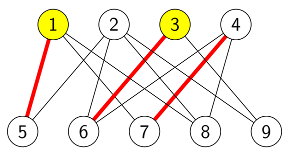
横断マトロイド：証明 (3)



$(M \cup N) - (M \cap N)$ を見ると，
G のどの頂点も M の 1 つ以下の辺と N の 1 つ以下の辺と接続している

- すなわち，$(M \cup N) - (M \cap N)$ の辺をたどると，
  M の辺と N の辺が必ず交互に現れる
- すなわち，たどってできるものは道か閉路である
- $|M| > |N|$ なので，必ず，M の辺を両端に持つ道がどこかに存在
- その道を P とする

横断マトロイド：証明 (4)

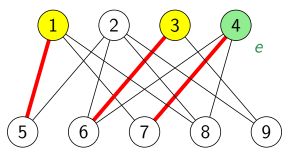


ここで，新しいマッチング N' を以下のように作る

- その道 P においては，
  M の辺を N' に含め，N の辺は N' に含めない
- その他の部分では，
  N の辺を N' に含め，M の辺は N' に含めない

P の両端は M の辺なので，N' は確かにマッチングである

横断マトロイド：証明 (5)



N' が飽和する A の頂点は何であるか，見てみる

- 構成法から，N が飽和する頂点は N' も飽和する
- $N' - N$ の辺は M の辺であるので，
  $N' - N$ の端点は N が飽和していない頂点である
- $|N'| = |N| + 1$ なので，
  そのような頂点は，A の中にちょうど 1 つある
- それを e とすれば，$Y \cup \{e\}$ が N' によって飽和される頂点の集合

$\therefore$ ある $e \in X - Y$ が存在して，$Y \cup \{e\} \in \mathcal{I}$ となる $\square$

横断マトロイド

二部グラフ $G = (A, B; E)$

横断マトロイド (transversal matroid) とは？

G から得られる A 上の横断マトロイドとは，A 上のマトロイド $\mathcal{I}$ で，

$$X \in \mathcal{I} \quad \Leftrightarrow \quad X \text{ を飽和する } G \text{ のマッチングが存在する}$$

によって定義されるもの

今おこなったこと

- 横断マトロイドが確かにマトロイドであることの確認 (証明)

今からおこなうこと

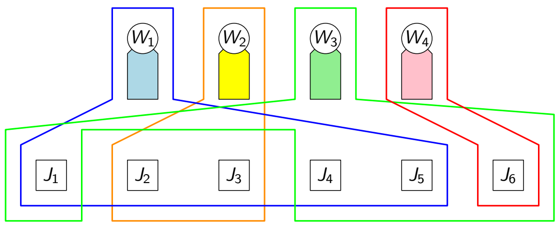
- 横断マトロイドが貪欲アルゴリズムとの関連で現れる様子の観察

目次

割当問題 (の一種)：状況 (1)

次のような状況を考える

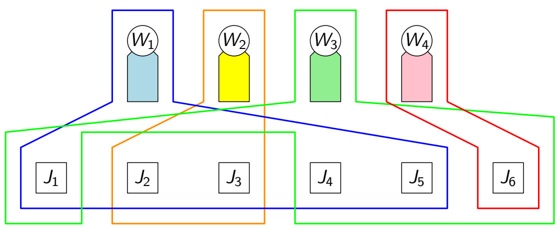
- 仕事：$J_1, J_2, \ldots, J_n$ (n 個)
  - 仕事 J_i を遂行した際に得られる利益 p_i (非負実数)
- 雇用者：$W_1, W_2, \ldots, W_m$ (m 人)
  - 雇用者 W_j が遂行できる仕事の集合 F_j



割当問題 (の一種)：状況 (2)

次のような状況を考える (続き)

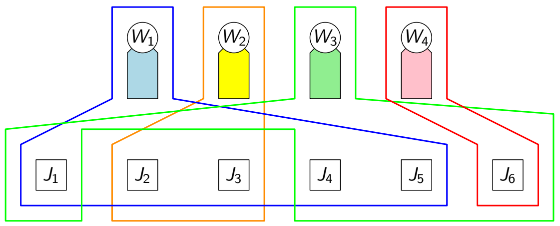
- どの仕事も一人の雇用者で遂行でき，遂行に 1 時間かかる
- 一人の雇用者は 2 つの仕事を同時に遂行できない



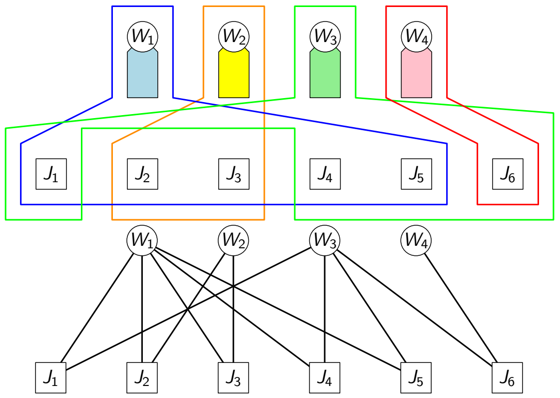
割当問題 (の一種)：問題

問題

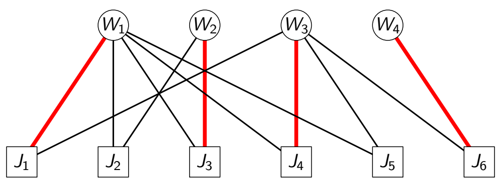
1 時間で得られる利益が最大になるように仕事を遂行できるよう，
雇用者に仕事を割り当てるにはどうすればよいか？



割当問題：二部グラフの構成

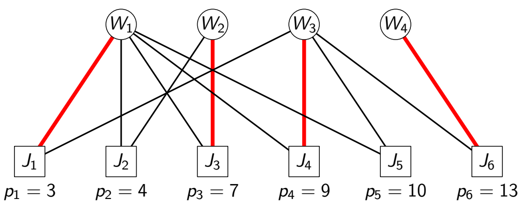


割当問題：仕事の割当 ↔ マッチング



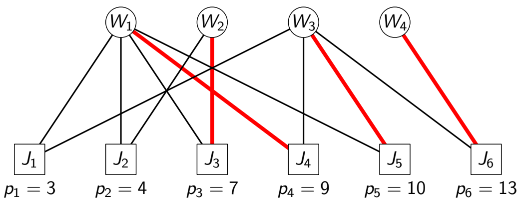
割当問題：マッチングと得られる利益 (1)

得られる利益 $= 3 + 7 + 9 + 13 = 32$



割当問題：マッチングと得られる利益 (2)

得られる利益 $= 7 + 9 + 10 + 13 = 39$

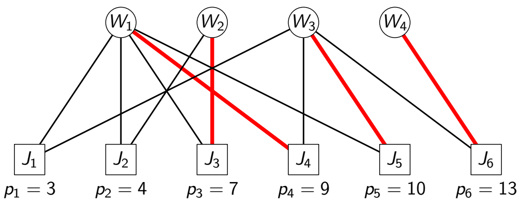


最適な割当

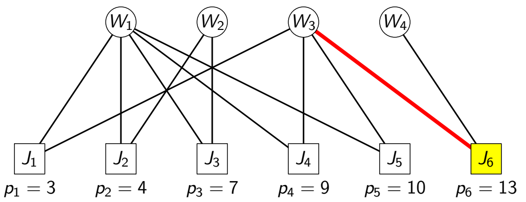
割当問題 → 横断マトロイドの最大独立集合問題

この割当問題は「マトロイドの最大独立集合問題」

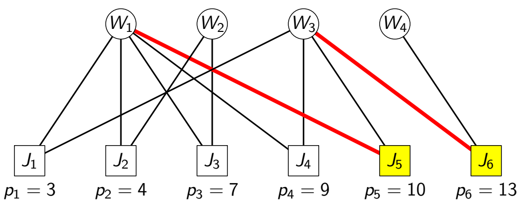
- 台集合 $A = \{J_1, \ldots, J_n\}$ (仕事の集合)
- 考えるマトロイド：A 上の横断マトロイド
  - 二部グラフ $(A, B; E)$
  - $B = \{W_1, \ldots, W_m\}$ (雇用者の集合)
  - $\{J_i, W_j\} \in E \Leftrightarrow J_i \in F_j$ (W_j が遂行できる仕事の集合)
- 要素 $J_i \in A$ の重み $= p_i$



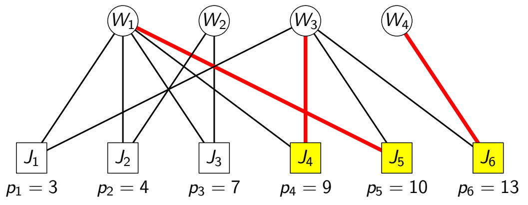
割当問題：貪欲アルゴリズムの動き (1)



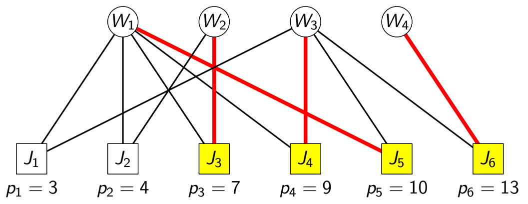
割当問題：貪欲アルゴリズムの動き (2)



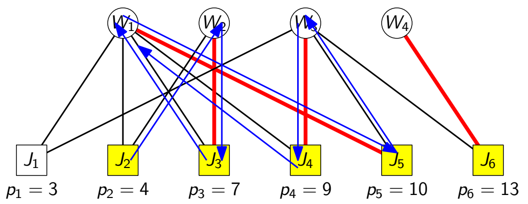
割当問題：貪欲アルゴリズムの動き (3)



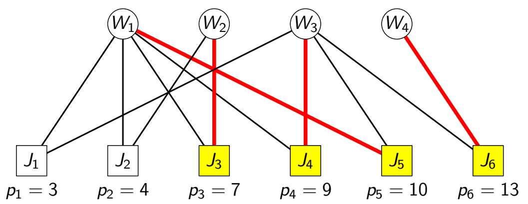
割当問題：貪欲アルゴリズムの動き (4)



割当問題：貪欲アルゴリズムの動き (5)



割当問題：貪欲アルゴリズムの動き (6)



貪欲アルゴリズムによって得られた最適解

目次

ジョブ・スケジューリング問題 (の一種)：状況

次のような状況を考える

1 台の機械でいくつものジョブを処理する

- ジョブ $J_1, J_2, \ldots, J_n$ (n 個)
- どのジョブの処理時間も同じ (1 時間とする)

J_1 J_2 J_3 J_4 J_5 J_6

ジョブ・スケジューリング問題 (の一種)：状況

次のような状況を考える

各ジョブ J_i は次の値を持つ

- 納期 d_i (完了期限)
- コスト c_i

納期までに完了しなかったジョブに対してコストを払う

J_1 J_2 J_3 J_4 J_5 J_6

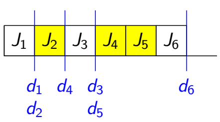
	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2

ジョブ・スケジューリング問題 (の一種)：問題

問題

払うコストを最小にするようなジョブ処理順は何か？

	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2

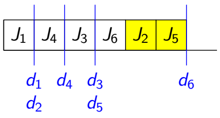


コスト $= 9 + 6 + 4 = 19$

問題

払うコストを最小にするようなジョブ処理順は何か？

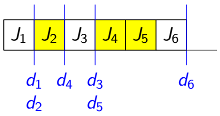
	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2



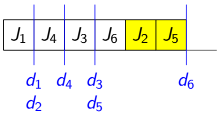
コスト $= 9 + 4 = 13$

$\boxed{\text{遅延したジョブのコスト和最小化}} \Leftrightarrow \boxed{\text{遅延しなかったジョブのコスト和最大化}}$

	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2

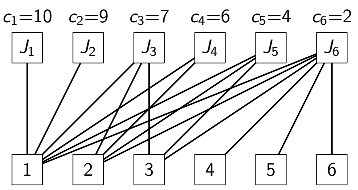


遅延したジョブの
コスト和 $= 19$
遅延しなかったジョブの
コスト和 $= 19$

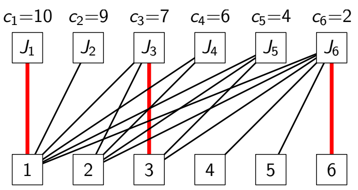


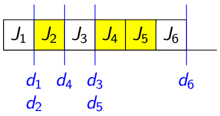
遅延したジョブの
コスト和 $= 13$
遅延しなかったジョブの
コスト和 $= 25$

	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2



	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2

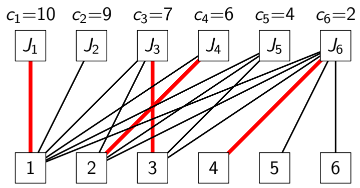


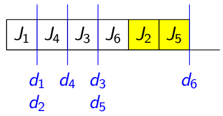


割り当てられたジョブのコスト和 $= 19$

ジョブ・スケジューリング問題：割当とコスト (2)

	J_1	J_2	J_3	J_4	J_5	J_6
納期 d_i	1	1	3	2	3	6
コスト c_i	10	9	7	6	4	2



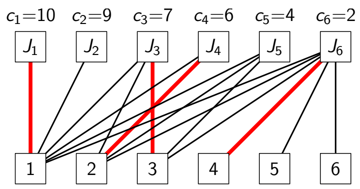


割り当てられたジョブのコスト和 $= 25$

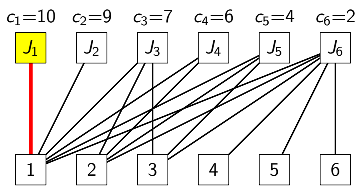
ジョブ・スケジューリング問題 → 横断マトロイドの最大独立集合問題

このスケジューリング問題は「マトロイドの最大独立集合問題」

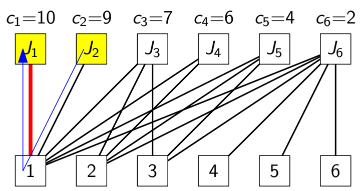
- 台集合 $A = \{J_1, \ldots, J_n\}$ (ジョブの集合)
- 考えるマトロイド：A 上の横断マトロイド
  - 二部グラフ $(A, B; E)$
  - $B = \{1, 2, \ldots, n\}$ (時間帯の集合)
  - $\{J_i, j\} \in E \Leftrightarrow j \leq d_i$
- 要素 $J_i \in A$ の重み $= c_i$



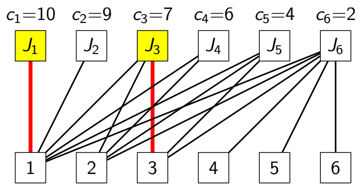
ジョブ・スケジューリング問題：貪欲アルゴリズムの動き (1)



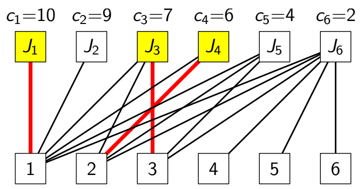
ジョブ・スケジューリング問題：貪欲アルゴリズムの動き (2)



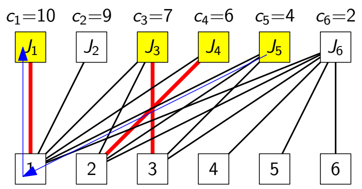
ジョブ・スケジューリング問題：貪欲アルゴリズムの動き (3)



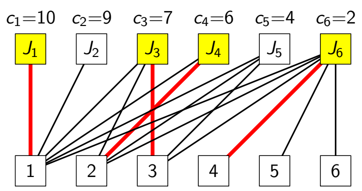
ジョブ・スケジューリング問題：貪欲アルゴリズムの動き (4)



ジョブ・スケジューリング問題：貪欲アルゴリズムの動き (5)

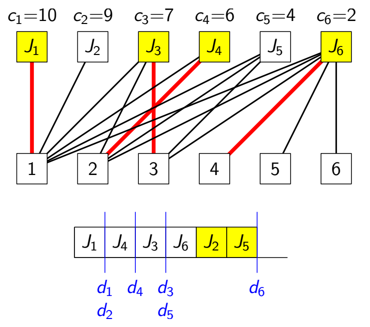


ジョブ・スケジューリング問題：貪欲アルゴリズムの動き (6)



貪欲アルゴリズムによって得られた最適解

ジョブ・スケジューリング問題：得られた最適処理順



目次

今回のまとめ

今日の目標

マトロイドに対する貪欲アルゴリズムの応用を見る
- 割当問題 (の一種)
- ジョブ・スケジューリング問題 (の一種)

鍵となる概念：横断マトロイド

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