情報システム評価学 ―整数計画法―

第6回目：分枝限定法

塩浦昭義（東北大学 大学院情報科学研究科 准教授）

整数計画問題に対する厳密解法

- 整数計画問題は解を全列挙すれば解ける
- しかし，計算時間が膨大で現実には不可能

  ➔ 解の全列挙における無駄を出来るだけ省く
  - 動的計画法：同一の部分問題を繰り返し解かない
  - 分枝限定法：ある部分問題から最適解が得られないことがわかったら，その部分問題は無視する

分枝限定法の考え方

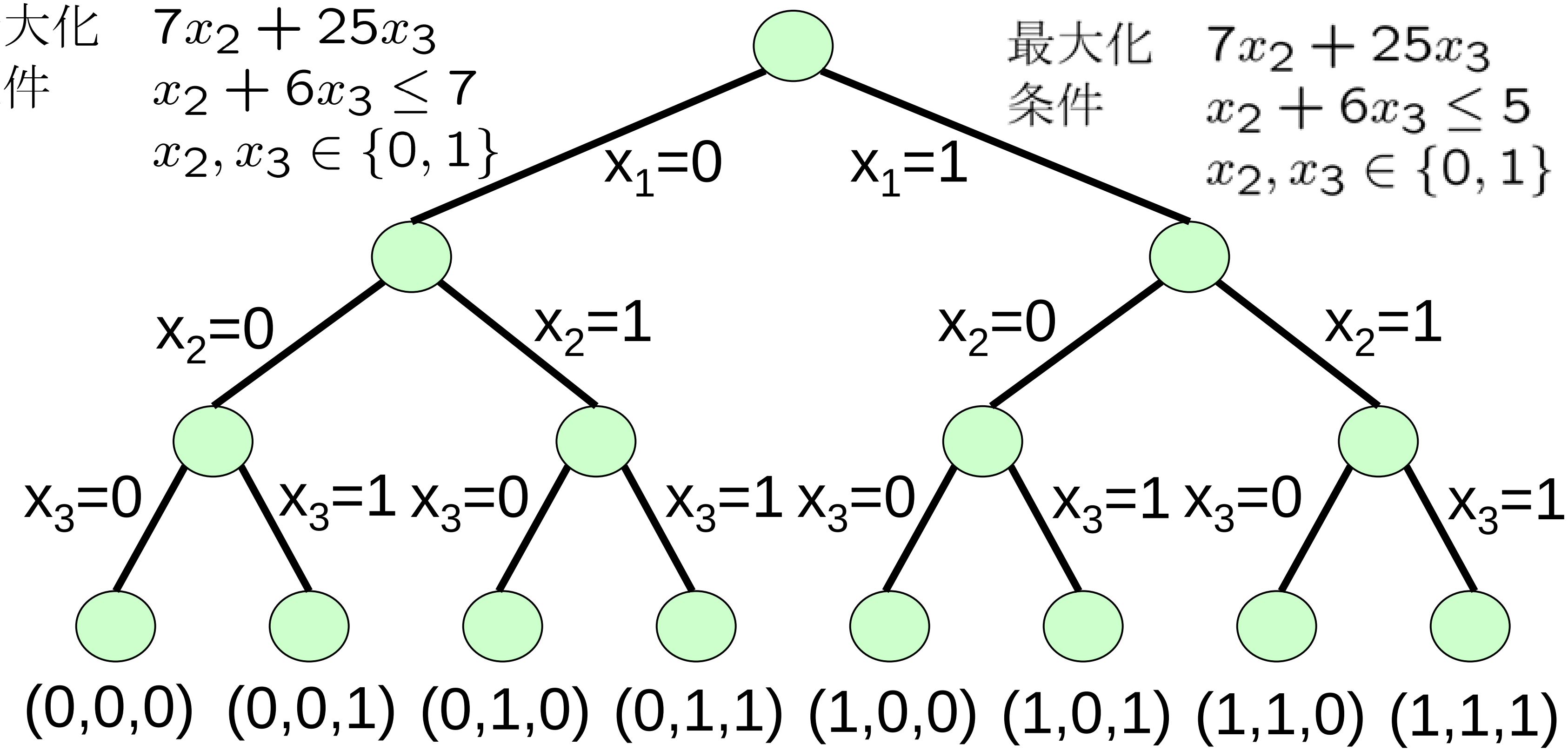
- 整数計画問題（組合せ最適化問題）を，場合分けによって部分問題に分解（分枝操作）
  - 0-1ナップサック問題：各変数について 0 の場合と 1 の場合に分ける
  - 巡回セールスマン問題：次に訪問する都市によって場合分け
- 分枝の進行の様子は探索木により表現可能

0-1ナップサック問題の探索木

$$
\begin{array}{ll}
\text{最大化} & 10x_1 + 7x_2 + 25x_3 \\
\text{条件} & 2x_1 + x_2 + 6x_3 \leq 7 \\
 & x_j \in \{0, 1\} \ (j = 1, 2, 3)
\end{array}
$$

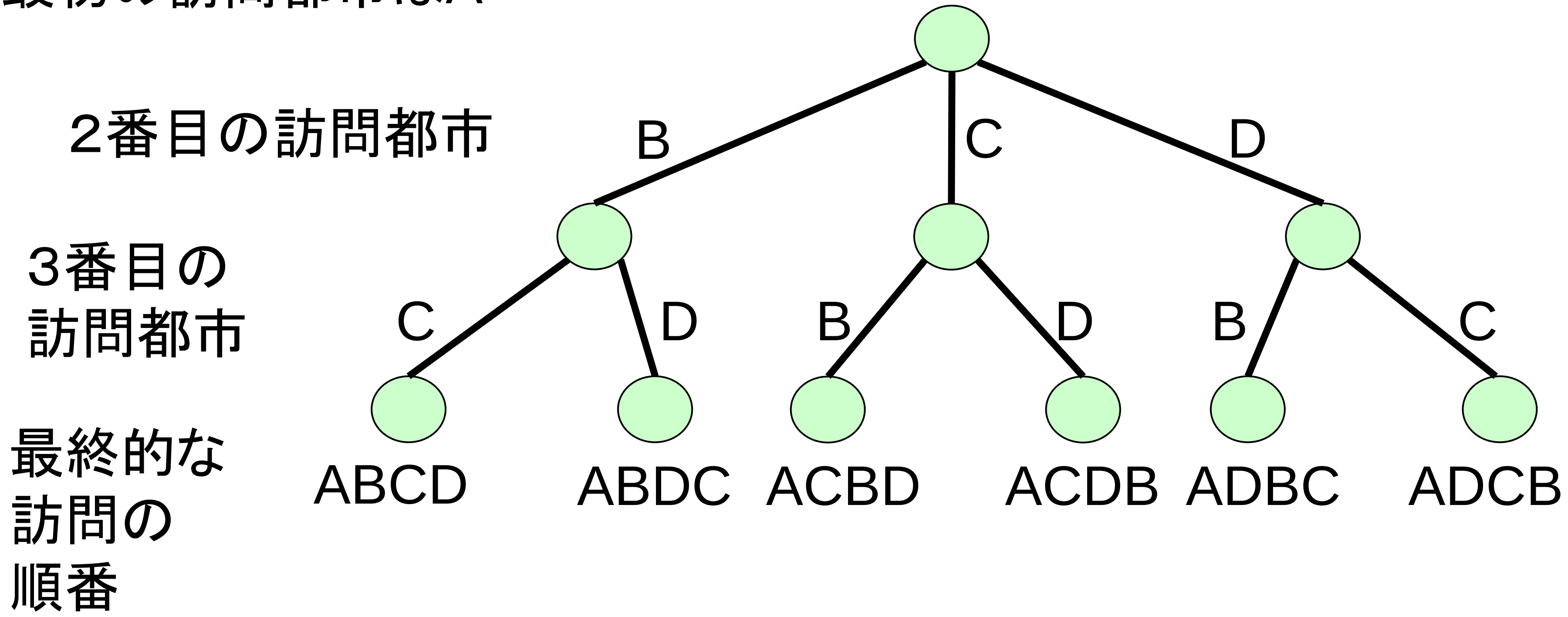
$$
\begin{array}{ll}
\text{最大化} & 7x_2 + 25x_3 \\
\text{条件} & x_2 + 6x_3 \leq 7 \\
 & x_2, x_3 \in \{0, 1\}
\end{array}
$$

$$
\begin{array}{ll}
\text{最大化} & 7x_2 + 25x_3 \\
\text{条件} & x_2 + 6x_3 \leq 5 \\
 & x_2, x_3 \in \{0, 1\}
\end{array}
$$



巡回セールスマン問題の探索木

- 4都市{A,B,C,D}の対称巡回セールスマン問題の場合
- 最初の訪問都市はA



分枝限定法の考え方

- 分枝操作により，たくさんの部分問題が生成される
- 解いても無駄な部分問題が検出されたら，さらなる分枝操作をストップ（限定操作）
- 解いても無駄な部分問題の例
  - 最適解がすでに得られた部
  - 現在の暫定解より良い許容解を得られる可能性がなくなった
  - 許容解が存在しない（実行不可能）
- 暫定解

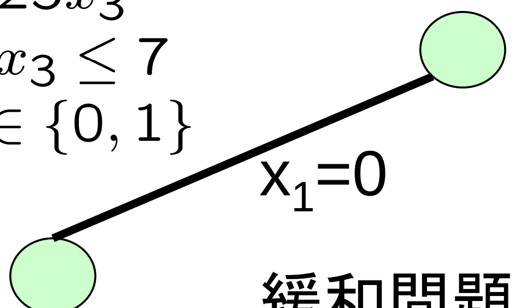
  ＝分枝限定法のそれまでの計算により得られている

  最良の許容解

最適解が得られた部分問題の例

最大化　$10x_1 + 7x_2 + 25x_3$
条件　$2x_1 + x_2 + 6x_3 \leq 7$
$x_j \in \{0, 1\}\ (j = 1, 2, 3)$

最大化　$7x_2 + 25x_3$
条件　$x_2 + 6x_3 \leq 7$
$x_2, x_3 \in \{0, 1\}$



緩和

最大化　$7x_2 + 25x_3$
条件　$x_2 + 6x_3 \leq 7$
$0 \leq x_2, x_3 \leq 1$

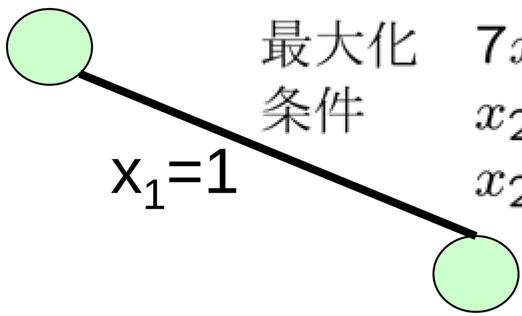
緩和問題の最適解は
$(x_2, x_3) = (1, 1)$（整数解）
➔元の部分問題の最適解

この部分問題をさらに調べても，
より良い解は得られない
➔この部分問題の探索をストップ

$(x_1, x_2, x_3) = (0, 1, 1)$ は暫定解，
目的関数値 ＝32

より良い解が得られない部分問題の例

$(x_1,x_2,x_3)=(0,1,1)$
は暫定解,
目的関数値＝32



最大化 $7x_2 + 25x_3$
条件 $x_2 + 6x_3 \leq 5$
$x_2, x_3 \in \{0, 1\}$

緩和

最大化 $7x_2 + 25x_3$
条件 $x_2 + 6x_3 \leq 5$
$0 \leq x_2, x_3 \leq 1$

緩和問題の最適解は
$(x_2, x_3) = (1, 2/3)$
目的関数値＝23.6666…
これは部分問題の上界値,
≦ 暫定解の目的関数値32

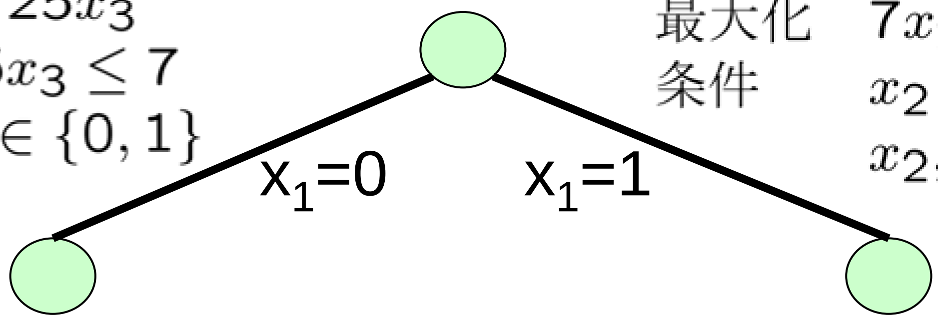
この部分問題をさらに調べても,
暫定解より良い解は得られない
➔この部分問題の探索をストップ

実行不可能な部分問題の例

$$\begin{array}{ll} \text{最大化} & 10x_1 + 7x_2 + 25x_3 \\ \text{条件} & \boxed{8x_1} + x_2 + 6x_3 \le 7 \\ & x_1, x_2, x_3 \in \{0, 1\} \end{array}$$

$$\begin{array}{ll} \text{最大化} & 7x_2 + 25x_3 \\ \text{条件} & x_2 + 6x_3 \le 7 \\ & x_2, x_3 \in \{0, 1\} \end{array}$$

$$\begin{array}{ll} \text{最大化} & 7x_2 + 25x_3 \\ \text{条件} & x_2 + 6x_3 \le -1 \\ & x_2, x_3 \in \{0, 1\} \end{array}$$



この部分問題は実行不可能

➔この部分問題の探索をストップ

分枝限定法の流れ

記号 L: 部分問題のリスト,
x*: 暫定解, z: 暫定解の目的関数値

ステップ0 : L={元問題}, z=－∞, x*は未定義とする.

ステップ1（探索）:

Lが空ならば計算終了. 現在の x* が最適解.

Lが非空ならば, Lから部分問題P'を選び, 削除.

ステップ2（限定操作）:

2-a: P' が実行不可能であることがわかったら, ステップ1へ.

2-b: P'の最適解が得られたら, 必要に応じて x*, z を更新してステップ1へ.

2-c: P' の緩和問題を解いた結果, 暫定解より良い許容解が得られないことがわかったらステップ1へ.

分枝限定法の流れ

ステップ２（限定操作）:

2-a: P' が実行不可能であることがわかったら，ステップ１へ.

2-b: P'の最適解が得られたら，必要に応じて x^*, z を更新して ステップ１へ.

2-c: P' の緩和問題を解いた結果，暫定解より良い許容解が得られないことがわかったらステップ１へ.

ステップ３（分枝操作）:

P'を場合分けによってP'_1, P'_2, …, P'_kに分解.

L にこれらの問題を入れ，ステップ１へ.

分枝限定法の実装における検討事項

分枝限定法の性能は，各々のステップを如何に実現するかによって左右される

どのような順番で部分問題を選ぶか？（部分問題の探索法）

ステップ１（探索）:

Lが空ならば計算終了．現在の x* が最適解.

Lが非空ならば，Lから部分問題P'を選び，削除.

ステップ２（限定操作）:

2-a: P' が実行不可能であることがわか

どのように実行不可能性を判定するか？

2-b: P'の最適解が得られたら，必要に応じて x*, z を更新して ステップ１へ.

2-c: P' の緩和問題を解いた結果，暫定 得られないことがわかったらステップ１へ.

どのような緩和問題を解くか？

ステップ３（分枝操作）:

P'を場合分けによって$P'_1, P'_2, \ldots, P'_k$に分解.

L にこれらの問題を入れ，ステップ１へ.

どのように問題を分解するか？

分枝限定法の実装における検討事項

- 部分問題の探索法
  - どのような順番で部分問題を選ぶか？
- 限定操作のやり方
  - どのように実行不可能性を判定するか？
  - どのような緩和問題を解くか？
- 分枝操作のやり方
  - どのように問題を分解するか？

部分問題の探索法

- 最良優先探索と深さ優先探索が一般的
- 最良優先探索
  - 最も良い許容解が得られそうな部分問題を優先的に選ぶ
  - 緩和問題から得られる上界値などを部分問題の評価値として使用する
  - 利点：計算終了までに解く部分問題の数が少ない
    ➔ 計算時間が少ない
- 深さ優先探索
  - 部分問題の深さがより大きい問題を優先的に選ぶ
  - 利点：実装が簡単，メモリ使用量が少ない，
    暫定解を早く得やすい

整数計画問題に対する分枝限定法の例

解きたい問題（P）

$$\begin{array}{ll} \text{Maximize} & 4x_1 - x_2 \\ \text{subject to} & 7x_1 - 2x_2 \leq 14 \\ & x_2 \leq 3 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1, x_2 : \text{非負整数} \end{array}$$

まず，LP緩和を解いてみる

$$\begin{array}{ll} \text{Maximize} & 4x_1 - x_2 \\ \text{subject to} & 7x_1 - 2x_2 \leq 14 \\ & x_2 \leq 3 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1, x_2 \geq 0 \end{array}$$

最適解は (20/7, 3), 目的関数値=59/7

➔整数解ではないので分枝を行う

整数計画問題に対する分枝限定法の例

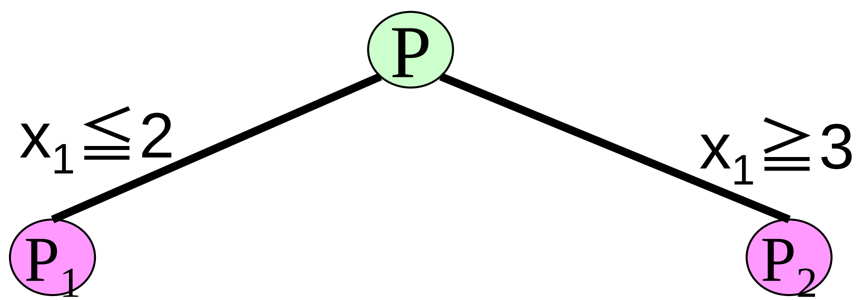
最適解は (20/7, 3), 目的関数値=59/7

➔整数解ではないので分枝を行う

x_1は整数でない➔x_1を利用して分枝

条件 $x_1 \leq \lfloor 20/7 \rfloor = 2$ を追加した子問題 P_1

条件 $x_1 \geq \lceil 20/7 \rceil = 3$ を追加した子問題 P_2



この分枝方法の利点:

問題PのLP緩和の最適解 (20/7, 3)は,

P_1, P_2 のLP緩和の許容解ではない

➔P_1, P_2 のLP緩和から得られる上界値 ≦ Pの上界値

整数計画問題に対する分枝限定法の例

次に問題P_1を選ぶ

$$
\begin{array}{ll}
\text{Maximize} & 4x_1 - x_2 \\
\text{subject to} & 7x_1 - 2x_2 \le 14,\ \ x_2 \le 3 \\
& 2x_1 - 2x_2 \le 3,\ \ x_1 \le 2 \\
& x_1, x_2 : \text{非負整数}
\end{array}
$$

問題P_1のLP緩和を解く

問題PのLP緩和を解いたときの情報が活用できる

➔ 問題P_1のLP緩和は簡単に解ける

最適解(2, ½), 目的関数値=15/2

➔整数解ではないので分枝を行う

x_2は整数でない➔x_2を利用して分枝

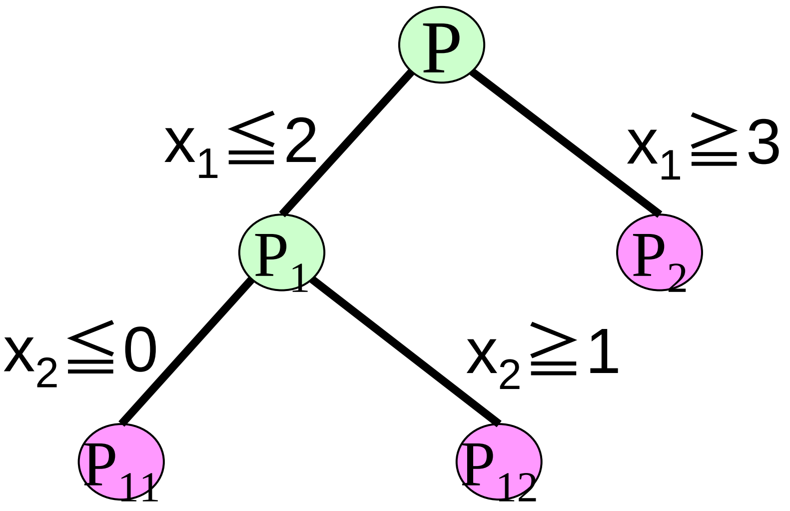
条件 $x_2 \le \lfloor 1/2 \rfloor = 0$ を追加した子問題 P_{11}

条件 $x_2 \ge \lceil 1/2 \rceil = 1$ を追加した子問題 P_{12}

整数計画問題に対する分枝限定法の例

次に問題P_2を選ぶ

$$\begin{array}{ll} \text{Maximize} & 4x_1 - x_2 \\ \text{subject to} & 7x_1 - 2x_2 \leq 14,\ x_2 \leq 3 \\ & 2x_1 - 2x_2 \leq 3,\ x_1 \geq 3 \\ & x_1, x_2 : \text{非負整数} \end{array}$$



問題P_2のLP緩和を解く

➔LP緩和は実行不可能

$$x_2 \leq 3,\ 7x_1 - 2x_2 \leq 14 \Rightarrow x_1 \leq 20/7 < 3$$

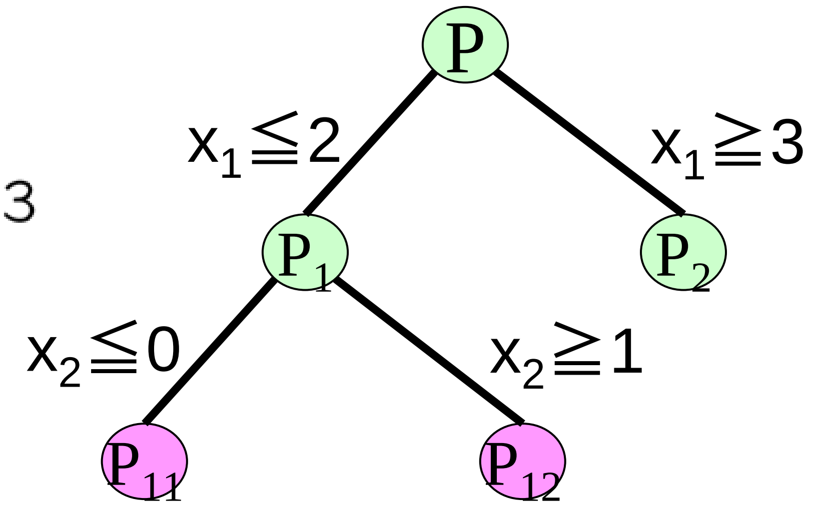
➔問題P_2は実行不可能

➔別の問題を選ぶ

整数計画問題に対する分枝限定法の例

次に問題P_{12}を選ぶ

$$
\begin{array}{ll}
\text{Maximize} & 4x_1 - x_2 \\
\text{subject to} & 7x_1 - 2x_2 \leq 14,\ x_2 \leq 3 \\
 & 2x_1 - 2x_2 \leq 3 \\
 & x_1 \leq 2,\ x_2 \geq 1 \\
 & x_1, x_2 : \text{非負整数}
\end{array}
$$



問題P_{12}のLP緩和を解く

最適解(2, 1)，目的関数値＝7

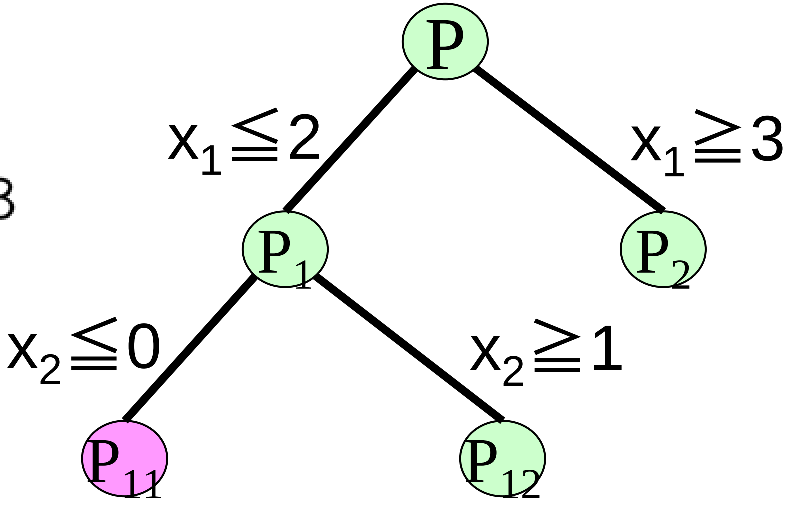
➔整数解なので，P_{12}の最適解である

➔元の問題Pの暫定解として使う

整数計画問題に対する分枝限定法の例

次に問題P_{11}を選ぶ

$$
\begin{array}{ll}
\text{Maximize} & 4x_1 - x_2 \\
\text{subject to} & 7x_1 - 2x_2 \le 14,\ x_2 \le 3 \\
& 2x_1 - 2x_2 \le 3 \\
& x_1 \le 2,\ x_2 \le 0 \\
& x_1, x_2 : \text{非負整数}
\end{array}
$$



問題P_{11}のLP緩和を解く

最適解(3/2, 0)，目的関数値＝6

➔目的関数値6 ≦ 暫定解の目的関数値 7

➔これ以上調べる必要なし

暫定解(2, 1)

目的関数値＝7

部分問題のリストが空になった➔分枝限定法終了

現在の暫定解が最適解

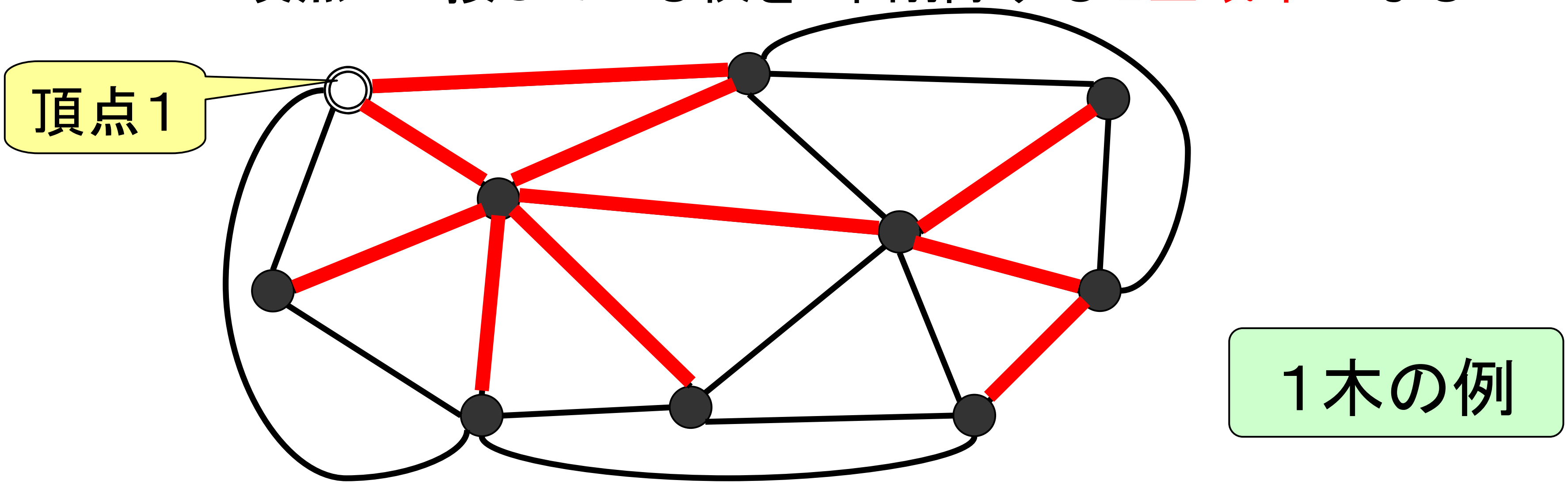
分枝限定法の実装における検討事項

今回の分枝限定法の例の場合:

- 限定操作のやり方 ➔ LP緩和を利用
  - どのように実行不可能性を判定するか？➔ LP緩和が実行不可能ならば, 部分問題も実行不可能
- 分枝操作のやり方
  - 整数でない変数 x_j を利用して分解
  - 複数存在する場合は, 「最も整数から遠い」x_j を利用すると良い（例: $x_1 = 0.9$, $x_2 = 2.5$ ならば x_2 を利用）

巡回セールスマン問題に対する分枝限定法の例

- 限定操作のやり方ー1木による緩和を利用
  - 頂点1にちょうど2本の枝が接している
  - 頂点1に接している枝を1本削除すると全域木になる



最小1木は，そのグラフの巡回路の長さの下界値

∴ある部分問題において，最小1木の長さ≧暫定解の長さ

➔ その部分問題は無視してよい

最小１木の性質

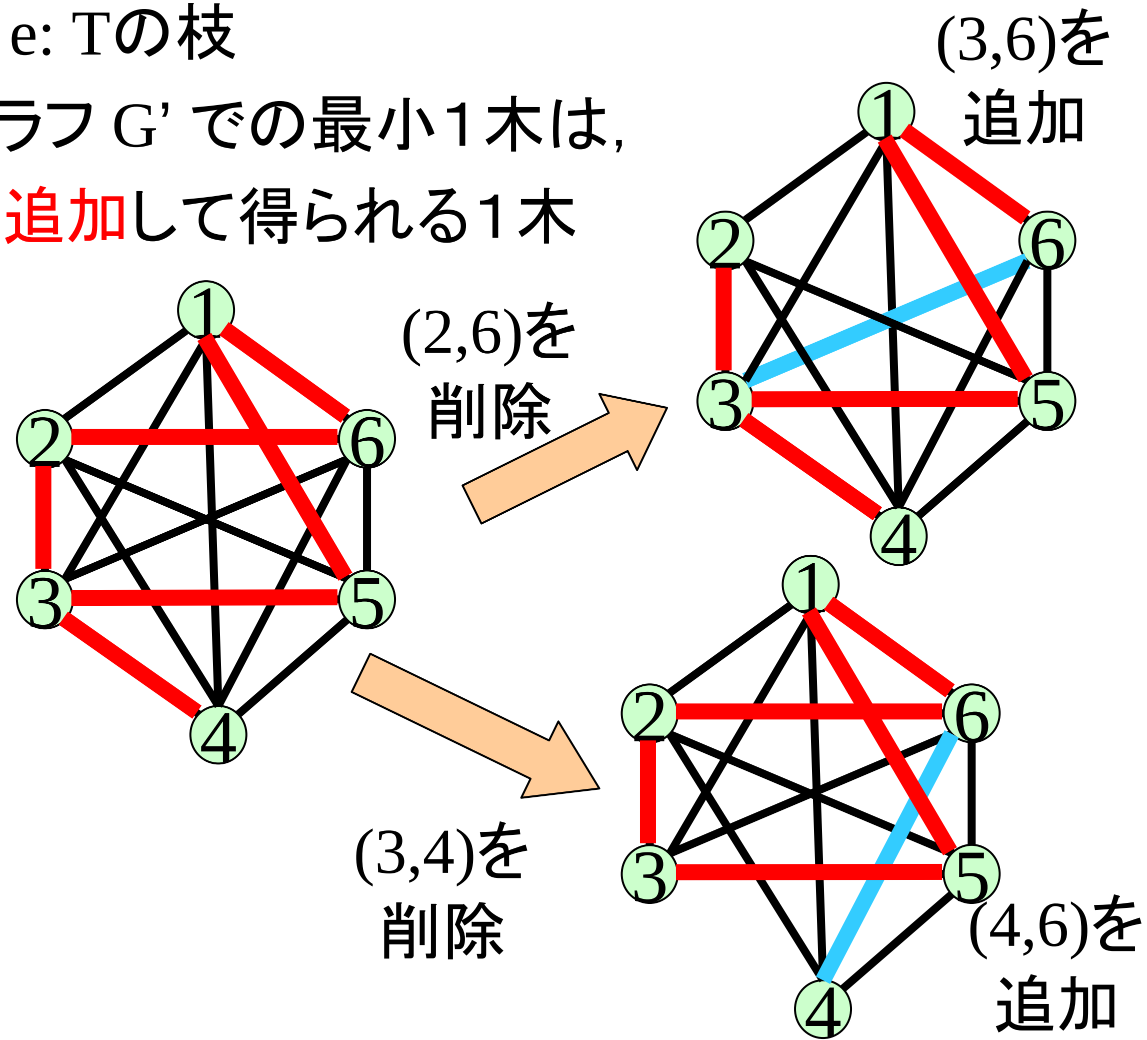
- Ｔ: グラフGの最小１木，e: Tの枝

➔ Gから枝 e を削除したグラフ G’ での最小１木は，

T – {e} にある枝を一本追加して得られる1木

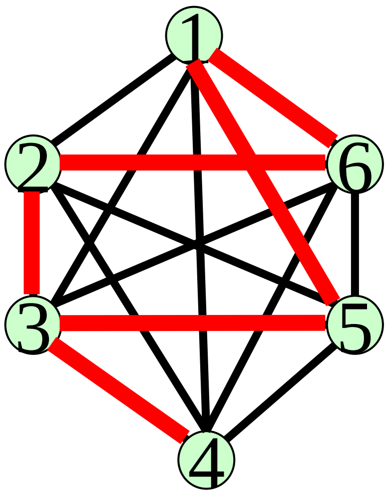
2	3	4	5	6	
99	58	57	18	53	1
	42	52	88	30	2
		30	35	45	3
			95	46	4
				85	5

２頂点間の枝の長さ



巡回セールスマン問題に対する分枝限定法の例

- 解きたい問題：6頂点の巡回セールスマン問題



最小1木の長さ＝208
巡回路ではないので分枝する

2	3	4	5	6	
99	58	57	18	53	1
	42	52	88	30	2
		30	35	45	3
			95	46	4
				85	5

2頂点間の枝の長さ

- 最小1木において，頂点3の次数＞2

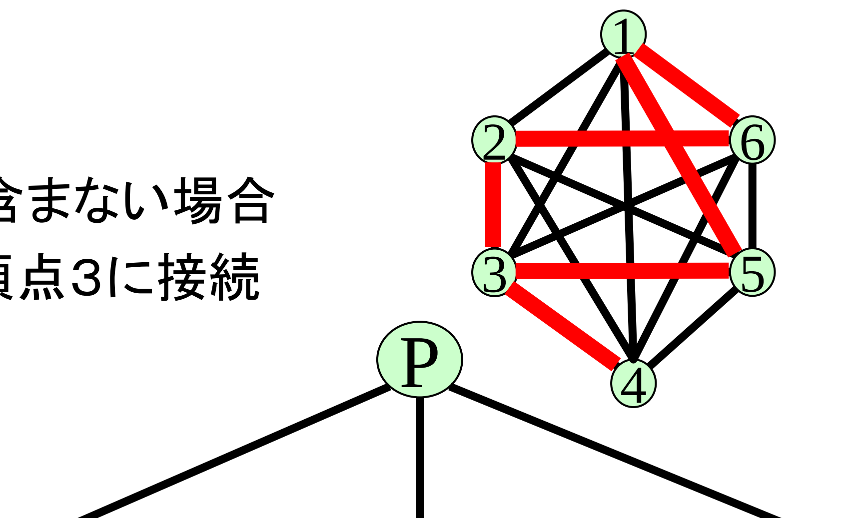
➔ 頂点3に接続する枝(3，5)，(3，4)を使って，3つの部分問題を生成

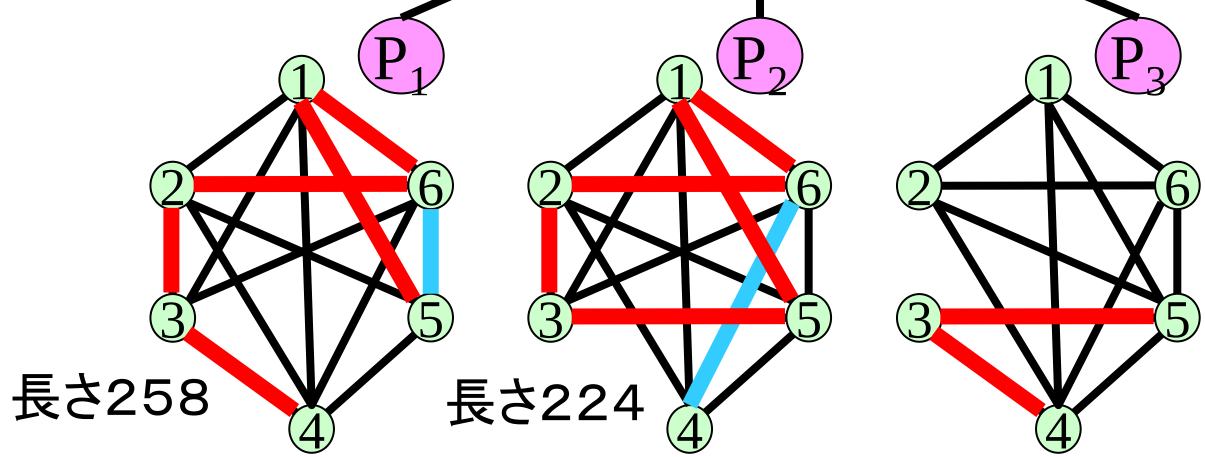
巡回セールスマン問題に対する分枝限定法の例

➔ 頂点3に接続する枝(3, 5), (3, 4)を使って, 3つの部分問題を生成

- P_1: 枝(3, 5)を含まない場合
- P_2: (3, 5)を含み, (3, 4)を含まない場合
- P_3: (3, 5), (3, 4)を含み, 頂点3に接続する他の枝を含まない場合

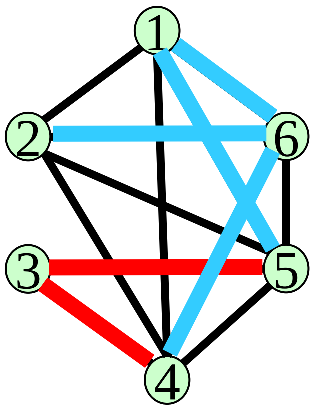
□ P_1, P_2 の最小1木は簡単に計算できる





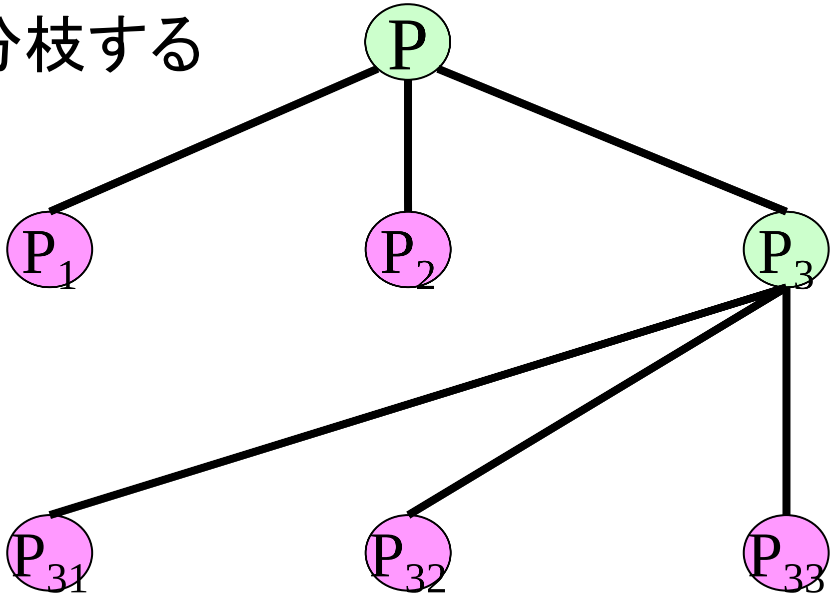
巡回セールスマン問題に対する分枝限定法の例

- 問題 P_3 を選ぶ



最小1木の長さ＝212

巡回路ではないので分枝する



頂点6の次数＞2

➔枝(2, 6), (4, 6)を使って部分問題を生成

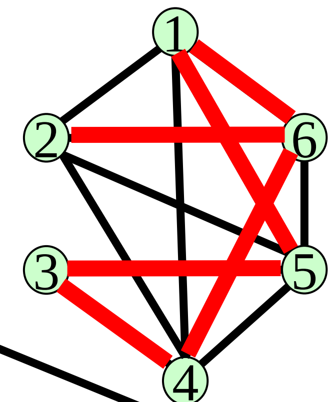
- P_{31}: 枝(2, 6)を含まない場合
- P_{32}: (2, 6)を含み, (4, 6)を含まない場合
- P_{33}: (2, 6), (4, 6)を含み, 頂点6に接続する他の枝を含まない場合

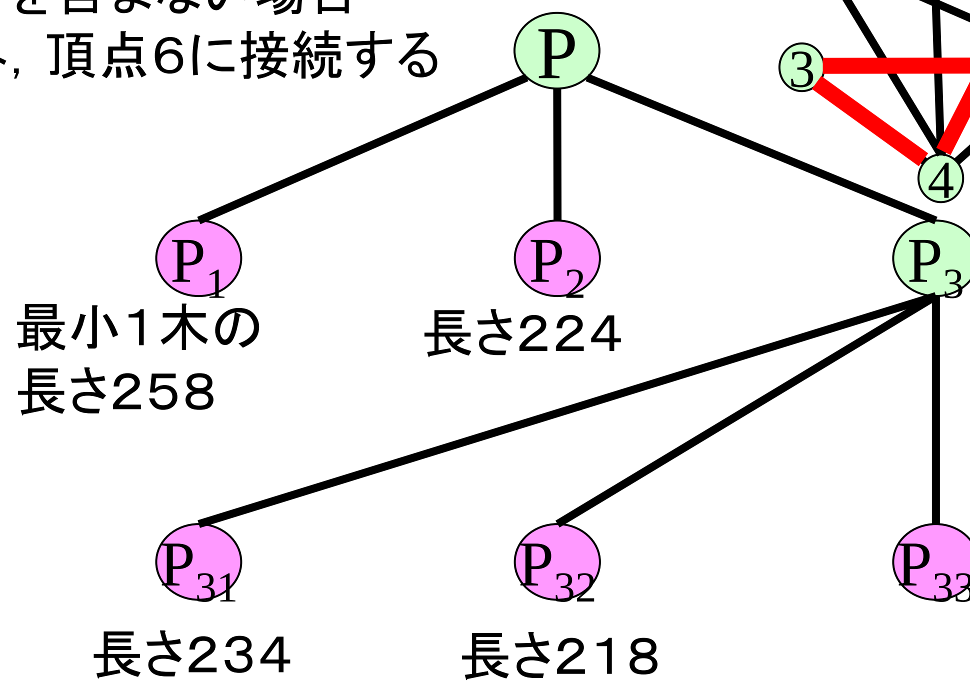
巡回セールスマン問題に対する分枝限定法の例

➔枝(2, 6), (4, 6)を使って部分問題を生成

- P_{31}: 枝(2, 6)を含まない場合
- P_{32}: (2, 6)を含み, (4, 6)を含まない場合
- P_{33}: (2, 6), (4, 6)を含み, 頂点6に接続する他の枝を含まない場合

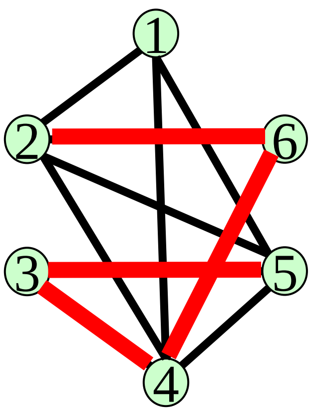
- P_{31}, P_{32} の最小1木は簡単に計算できる



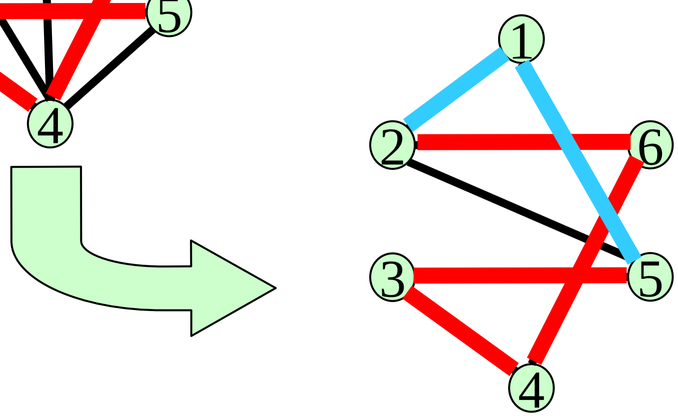


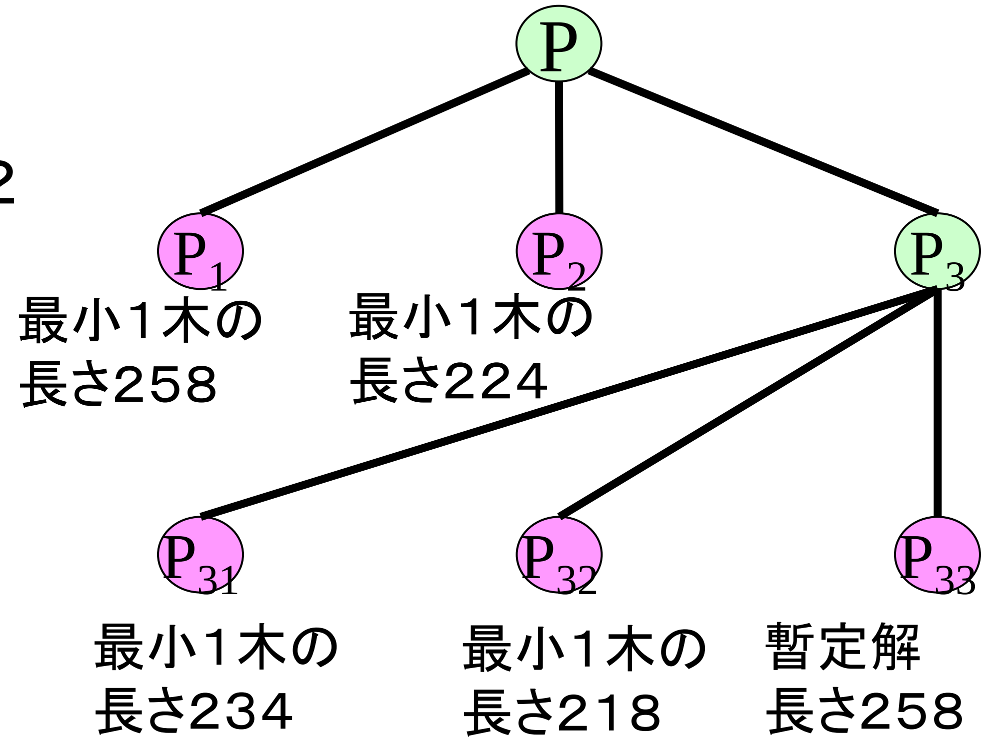
巡回セールスマン問題に対する分枝限定法の例

- 次に問題P_{33} を選ぶ



頂点4は既に次数2
➔他の枝(1,4),
(2,4), (4,5) を削除

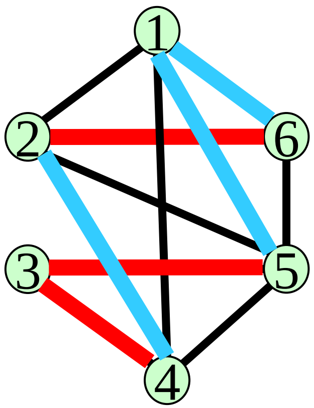


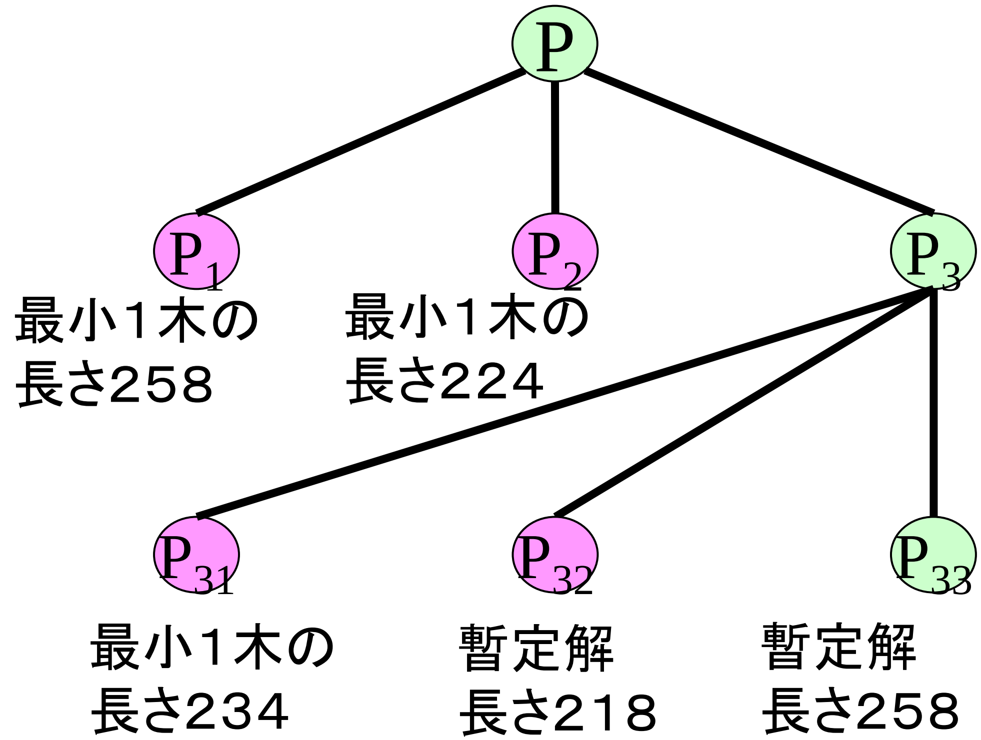


枝を削除したグラフでの最小1木の長さ＝258
これは巡回路➔暫定解とする

巡回セールスマン問題に対する分枝限定法の例

- 次に問題P_{32}を選ぶ





最小1木の長さ＝218
これは巡回路➔暫定解とする

まだ解いていない部分問題において，
最小1木の長さ≧暫定解の長さ218
➔ 他の部分問題は無視して良い ➔ 分枝限定法の計算終了

レポート問題（締切：12月15日）

(1). n = 4 の場合のロットサイズ決定問題を解きなさい．入力は次の通り： $(p_1, p_2, p_3, p_4) = (1,1,1,2)$, $(h_1, h_2, h_3, h_4) = (1, 1, 1, 1)$, $(f_1, f_2, f_3, f_4) = (20, 10, 45, 15)$, $(d_1, d_2, d_3, d_4) = (8, 5, 13, 4)$

(2). 次の整数計画問題を，λ = 7, 8, 9, 10 の場合について動的計画法で解きなさい

$$
\begin{array}{ll}
\text{Minimize} & 5x_1 - 3x_2 + 7x_3 + 10x_4 \\
\text{subject to} & 2x_1 - x_2 + 4x_3 + 5x_4 \geq \lambda \\
 & x_2, x_3 \in \{0, 1\},\ x_1, x_4 : \text{非負整数}
\end{array}
$$

(3). 次の0－1ナップサック問題を解きなさい

$$
\begin{array}{ll}
\text{Maximize} & 3x_1 + 4x_2 + 6x_3 + 5x_4 + 8x_5 \\
\text{subject to} & 412x_1 + 507x_2 + 714x_3 + 671x_4 + 920x_5 \leq 1794 \\
 & x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{array}
$$

レポート問題（締切：12月15日）

(3)のヒント: b = 0, 1, 2, … に対して次の整数計画問題を動的計画法で解き，その結果を使う

$$
\begin{array}{ll}
\text{Minimize} & 412x_1 + 507x_2 + 714x_3 + 671x_4 + 920x_5 \\
\text{subject to} & 3x_1 + 4x_2 + 6x_3 + 5x_4 + 8x_5 \geq b \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{array}
$$

(4). b = 0, 1, …, 12 に対して次の整数計画問題を動的計画法で解きなさい

$$
\begin{array}{ll}
\text{Maximize} & x_1^3 + 2\sqrt{x_2} + 4x_3 + 4x_4 \\
\text{subject to} & 2x_1^2 + 4x_2 + 6x_3 + 5x_4 \leq b \\
& x_1 \leq 3,\ x_2 \leq 2,\ x_4 \leq 1 \\
& x_1, x_2, x_3, x_4 : \text{非負整数}
\end{array}
$$

レポート問題(締切:12月15日)

(5). 次の問題を分枝限定法で解きなさい. LP緩和を使うこと.
LP緩和の計算には図(グラフ)を使って良い.

$$\begin{array}{ll} \text{Maximize} & 9x_1 + 5x_2 \\ \text{subject to} & 4x_1 + 9x_2 \leq 35,\ \ x_1 \leq 6 \\ & x_1 - 3x_2 \geq 1,\ \ 3x_1 + 2x_2 \leq 19 \\ & x_1, x_2 : \text{非負整数} \end{array}$$

(6). 次の0−1ナップサック問題を分枝限定法で解きなさい. LP緩和を使うこと.

$$\begin{array}{ll} \text{Maximize} & 17x_1 + 10x_2 + 25x_3 + 17x_4 \\ \text{subject to} & 5x_1 + 3x_2 + 8x_3 + 7x_4 \leq 12 \\ & x_1, x_2, x_3, x_4 \in \{0, 1\} \end{array}$$

レポート問題（締切：12月15日）

(7). 頂点数5の対称巡回セールスマン問題を分枝限定法で解きなさい．最小1木を緩和問題として使うこと．頂点間の枝の長さは次の通り．

2	3	4	5	
10	2	4	6	1
	9	3	1	2
		5	6	3
			2	4

☆レポートに関する注意☆

少なくとも3問は完全に解くこと．
ただし，動的計画法の問題
および分枝限定法の問題を
それぞれ1問以上解くこと．

分枝限定法の高速化のための工夫

- より良い上界値（緩和問題）を使う
- 良い許容解（近似解）をあらかじめ計算し，暫定解として使う
  - 無駄な部分問題を早く削除できる
  - 良い上界値・許容解の計算に時間を使いすぎると逆効果
- 分枝の工夫
  - $x_1 + x_2 + \ldots + x_k = 1$ と言う制約がある場合には，$x_j = 1$ と言う制約を付加した k 個の部分問題に分割
- 前処理：事前に問題の規模を出来るだけ小さくする
  - 値を固定できる変数を事前に検出
  - より強い条件式の導出
  - 無駄な条件式の削除