非対称巡回セールスマン問題に対する解法の研究

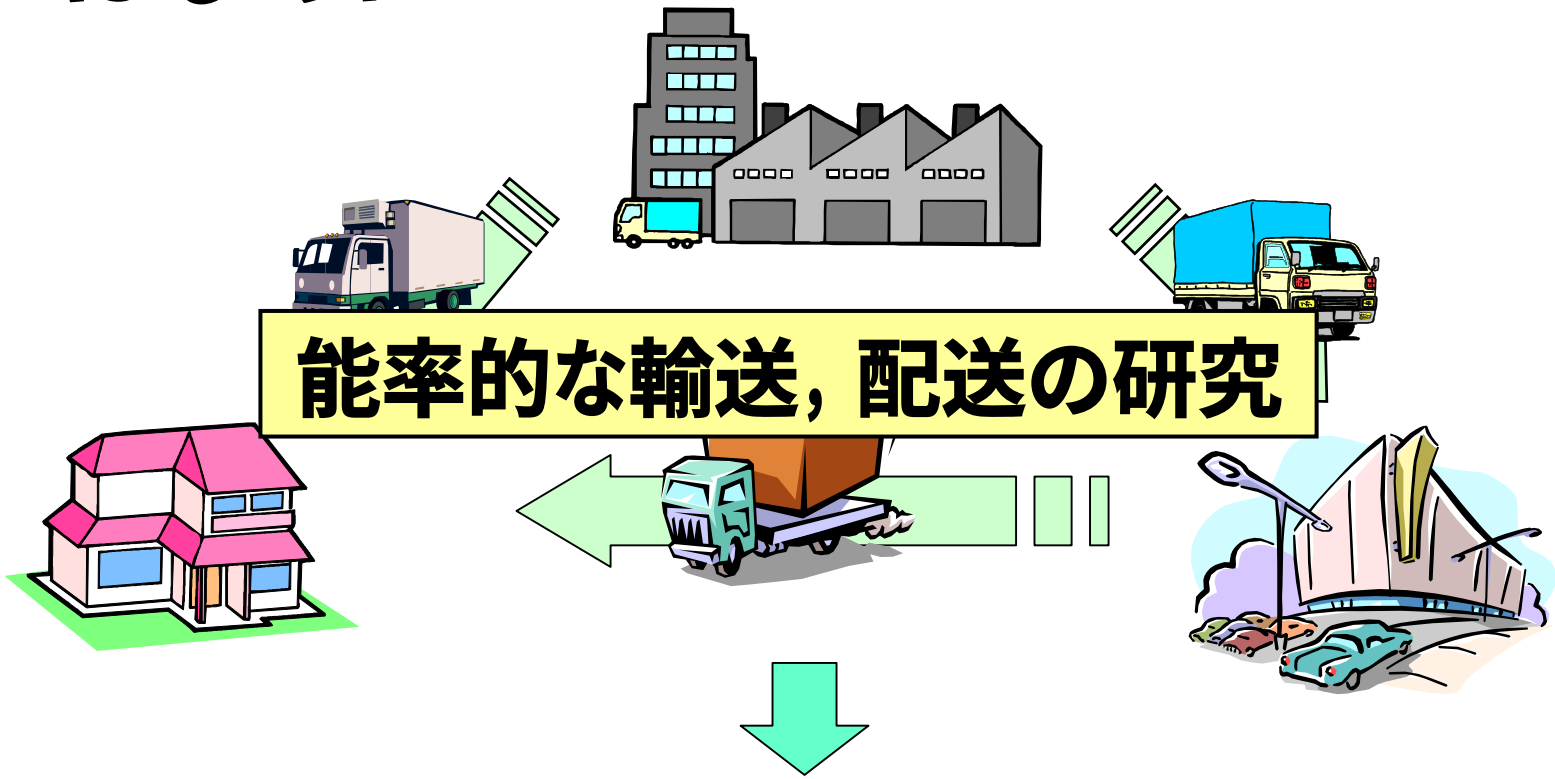
東京理科大学工学部経営工学科
沼田研究室

4400045　菊池　健之
4400099　藤野　悠

目次

- 1. はじめに
- 2. ATSPの定式化
- 3. ATSPの厳密解法について
- 4. プログラムの実装
- 5. 提案する処理方法
- 6. 実験
- 7. まとめ
- 8. 今後の課題

1．はじめに



巡回セールスマン問題
(Traveling Salesman Problem：TSP)

1．はじめに（つづき）

TSPとは，

指定された地点すべてを1回ずつ回るときの
費用の和が最小になる巡回路を求める問題

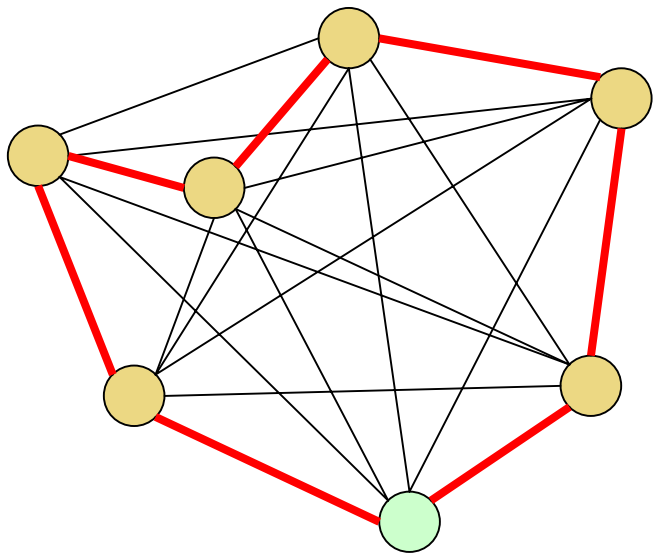


図1　巡回路

1．はじめに（つづき）

c_{ij}：都市 i から都市 j へ行くのにかかる費用

すべて $c_{ij} = c_{ji}$ → 対称型巡回セールスマン問題（STSP）

$c_{ij} \neq c_{ji}$ がある → 非対称巡回セールスマン問題（ATSP）

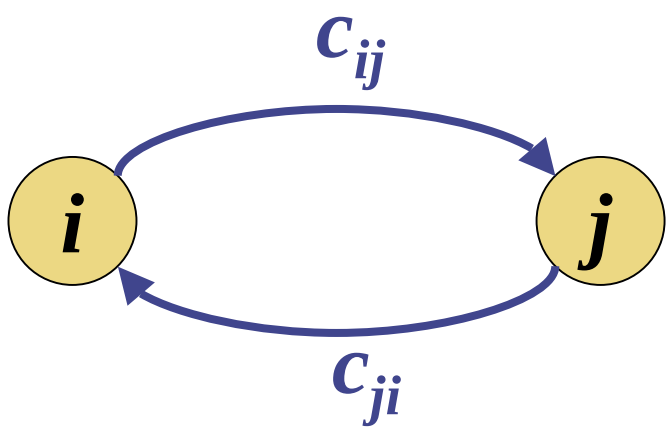


図2　都市 i j 間 j i 間の費用

●トラックの積み降ろしを基本とする配送経路問題

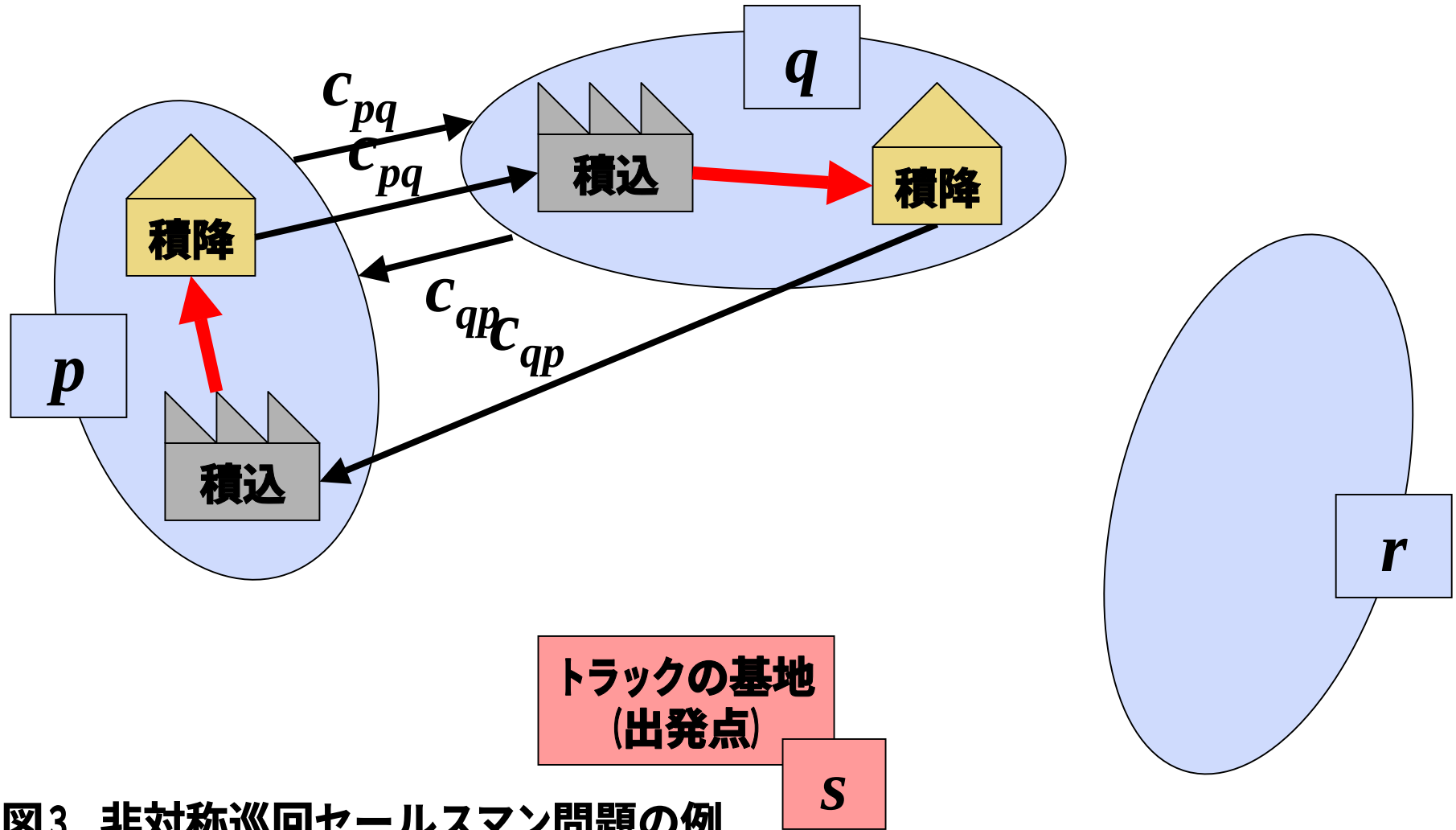
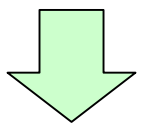


図3　非対称巡回セールスマン問題の例

1．はじめに (つづき)

沼田研究室では

ATSPに対する近似解法の研究を行なってきている[2.3]．



その近似解法の精度を評価するために，厳密解を求める必要が生じた．

1．はじめに (つづき)

研究内容

- 分枝限定法を基本とした厳密解法の実装
- それにおけるヒューリスティックな部分の工夫
- これらを基にした設定時間内に解き終わる解法の提案
- その解法の性能評価

2．ATSPの定式化

・記号の説明

n ：都市数

$V = \{1, 2, \ldots, n\}$：都市の集合

c_{ij} ：都市 i から都市 j への費用

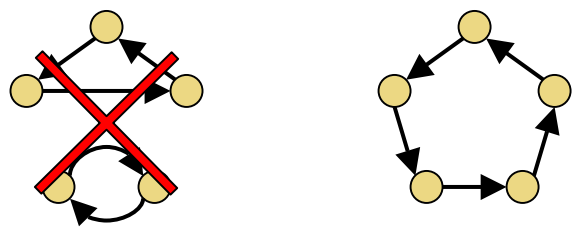
z ：総費用

$$x_{ij} = \begin{cases} 1 & (\text{都市 } i \text{ から都市 } j \text{ へ移動する}) \\ 0 & (\text{都市 } i \text{ から都市 } j \text{ へ移動しない}) \end{cases}$$

2．ATSPの定式化（つづき）

$$
\text{(ATSP)}\quad
\begin{cases}
\text{min.} & z = \sum_{i \in V} \sum_{j \in V - i} c_{ij} x_{ij} \\
\text{s.t.} & \sum_{i \in V - j} x_{ij} = 1 \quad (\ \forall j \in V\ ) & (1) \\
& \sum_{j \in V - i} x_{ij} = 1 \quad (\ \forall i \in V\ ) & (2) \\
& \sum_{i \in S} \sum_{j \in V - S} x_{ij} \geq 1 \quad (\ \forall S \subset V \quad , S \neq V\ , \phi\ ) & (3) \\
& x_{ij} \in \{1,0\} \quad (\ \forall i, \forall j \in V\ ) & (4)
\end{cases}
$$

(3)式



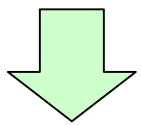
[4]

3．ATSPの厳密解法

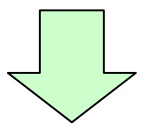
3．1．ATSPの厳密解法

ATSP → NP困難

厳密解を求めるためには列挙を
含まざるを得ないと信じられている



列挙を能率的に行なって解きたい

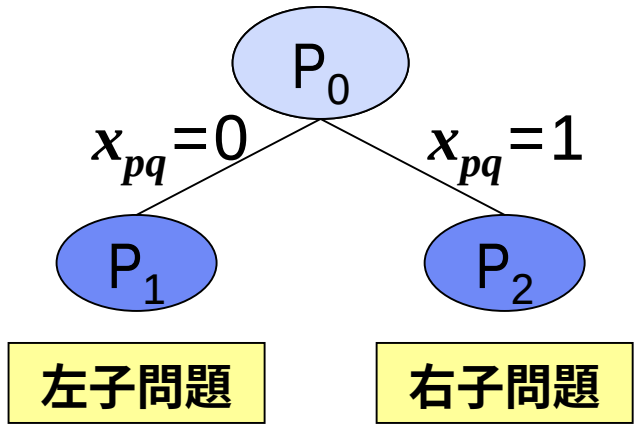


分枝限定法

3．1．ATSPの厳密解法（つづき）

●分枝限定法

厳密解を求める解法のひとつ



・分枝操作

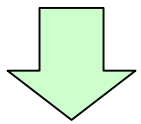
→元の問題を子問題に分けていくこと

・限定操作

→それ以降分枝しても最適解が見つからないと断定されたとき，分枝操作を停止すること

3．1．ATSPの厳密解法（つづき）

限定操作が行なわれるのはどういう場合か？



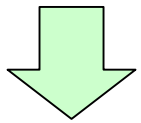
下界値≧ATSPの暫定値のとき

下界値：　最適値が取る可能性のある最小の値

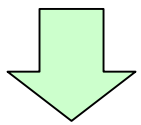
暫定値：　暫定的な巡回路の最小総費用

3．1．ATSPの厳密解法（つづき）

限定操作を行なうには下界値が必要



下界値は緩和問題を解くことで得られる



ATSPの場合，緩和問題は割当問題
（Assignment Problem：AP）となる

逐次最短路法などで
能率良く解くことができる

3．1．ATSPの厳密解法（つづき）

・緩和問題(AP)の定式化

(AP)

$$\text{min.} \quad \sum_{i \in V} \sum_{j \in V - i} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{i \in V - j} x_{ij} = 1 \quad (\forall j \in V) \qquad (5)$$

$$\sum_{j \in V - i} x_{ij} = 1 \quad (\forall i \in V) \qquad (6)$$

$$x_{ij} \geq 0 \quad (\forall i, \forall j \in V) \qquad (7)$$

最小となる目的関数値がATSPの下界値となる

3．1．ATSPの厳密解法（つづき）

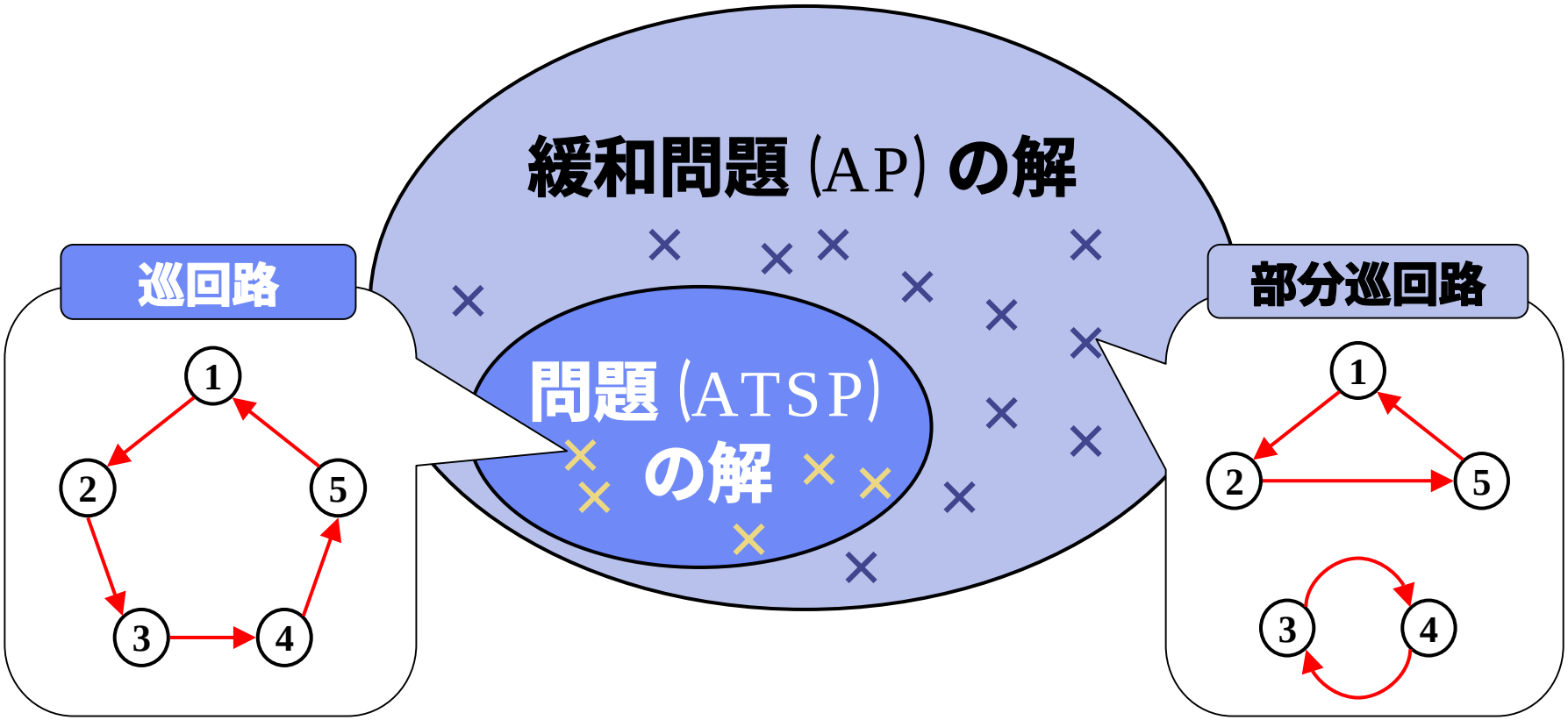


図4　ATSPの解と緩和問題の解

3．1．ATSPの厳密解法（つづき）

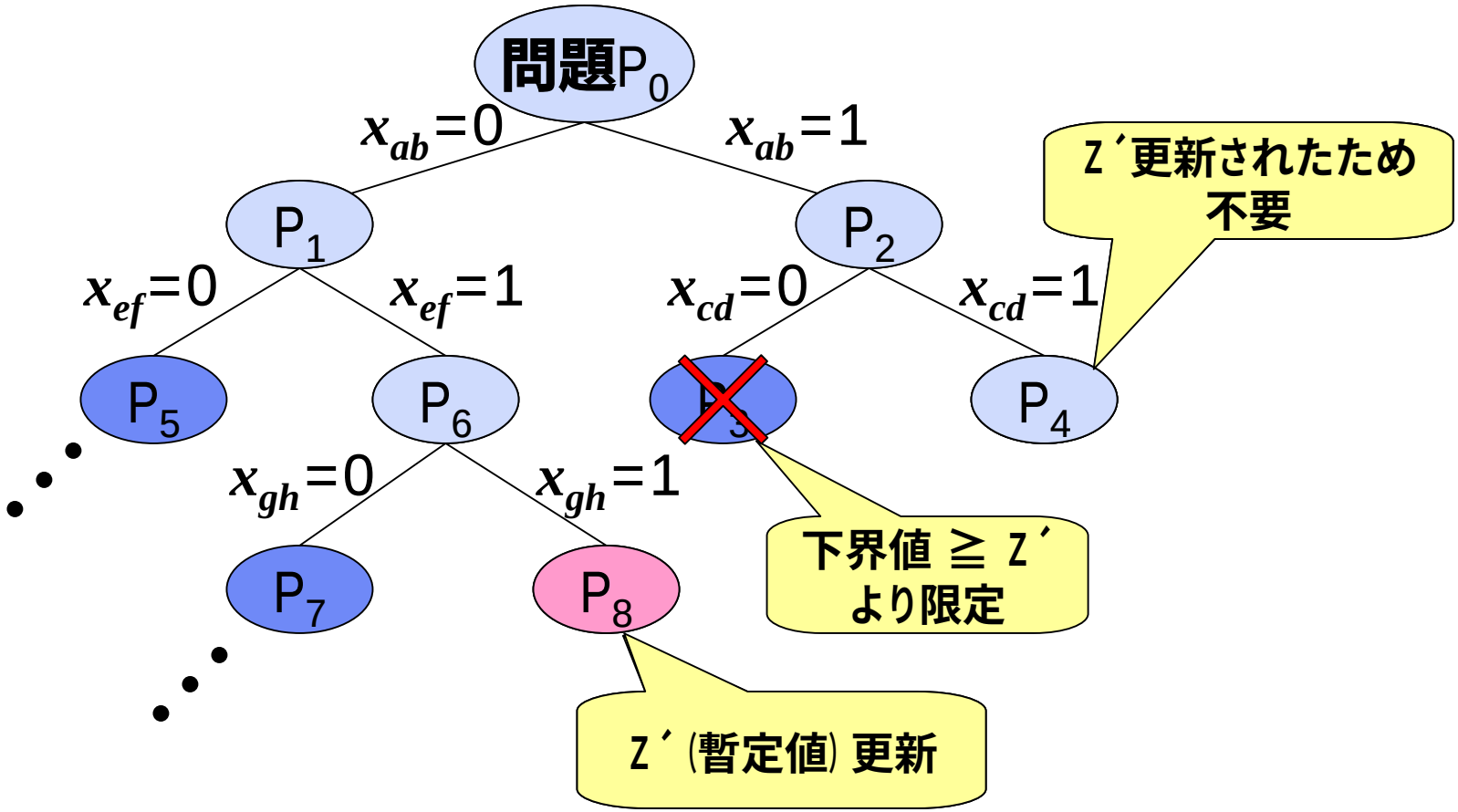


図5　分枝限定法のイメージ

4．プログラムの実装
4．1．実装の概略

分枝限定法を用いた
ATSPの厳密解法のプログラム

4．1．1．問題プール

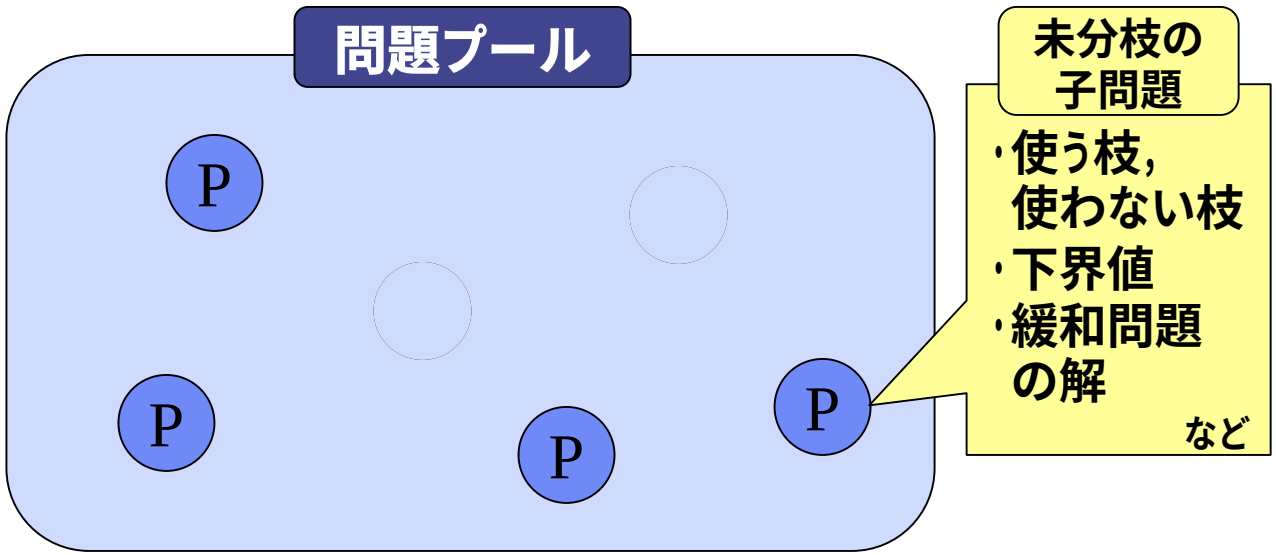
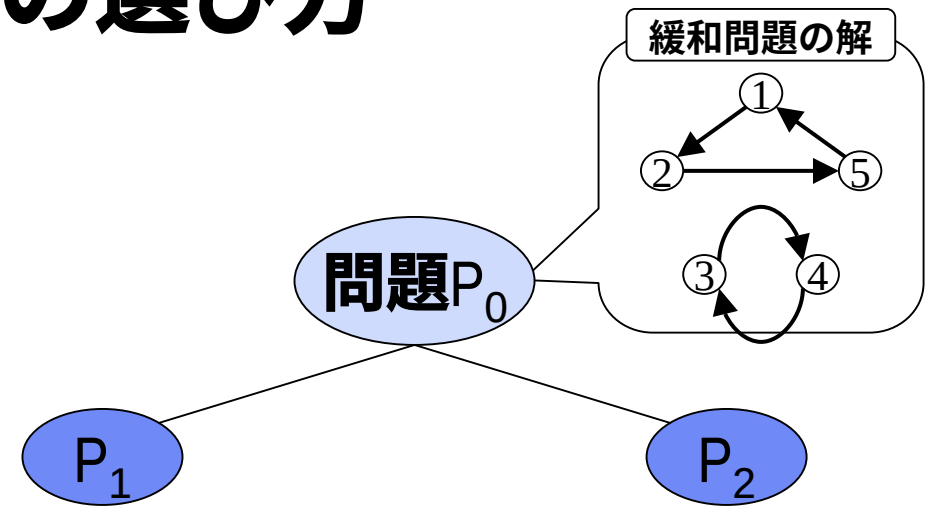


図6　問題プールのイメージ

4．1．2．枝の選び方



左子問題の下界値が大きくなるような枝を選択

（左子問題の緩和問題の最適値）

左子問題を早い段階で限定することができ
計算時間の短縮につながる

4．1．3．費用の書き換え

・使う枝が決定したときの処理

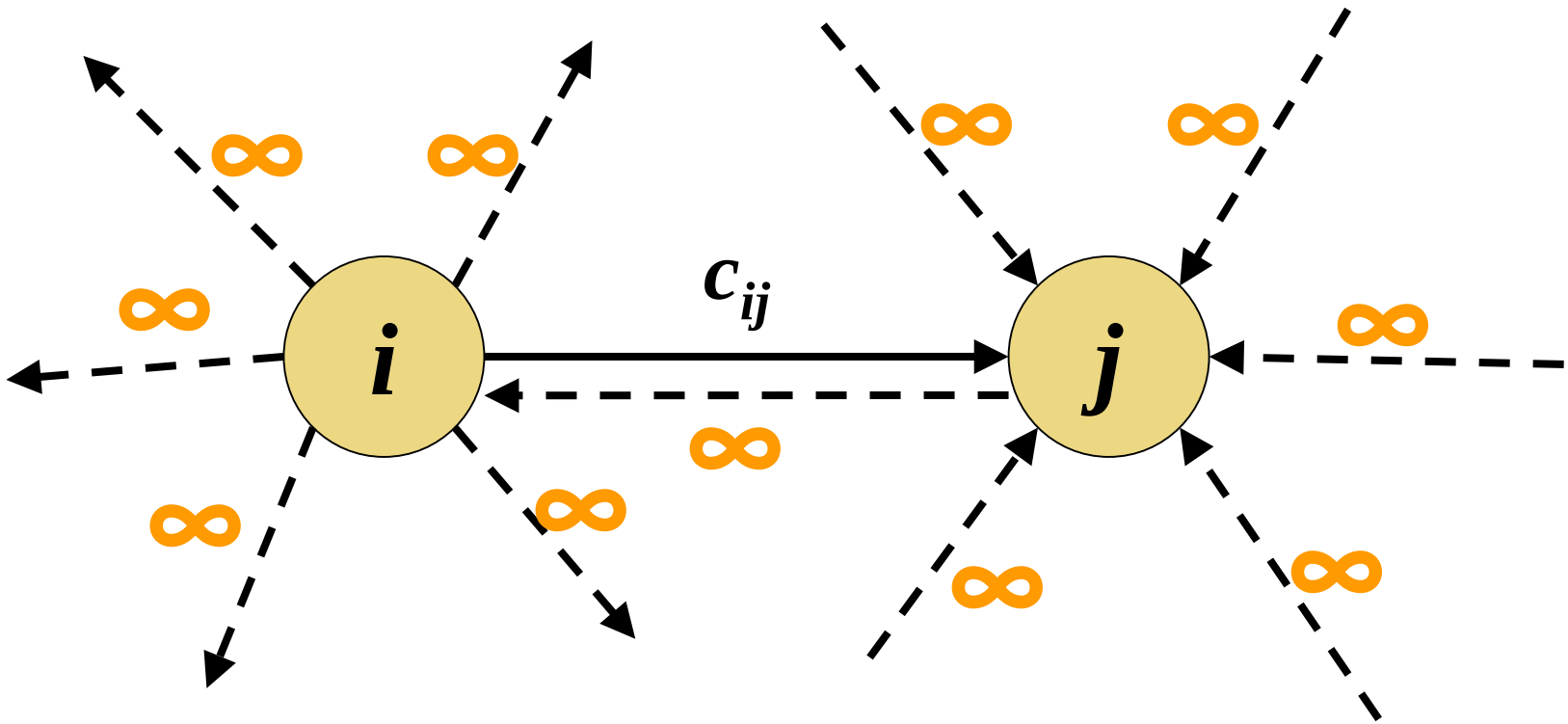


図7　費用の書き換え

4．1．4．限定操作をした問題の処理

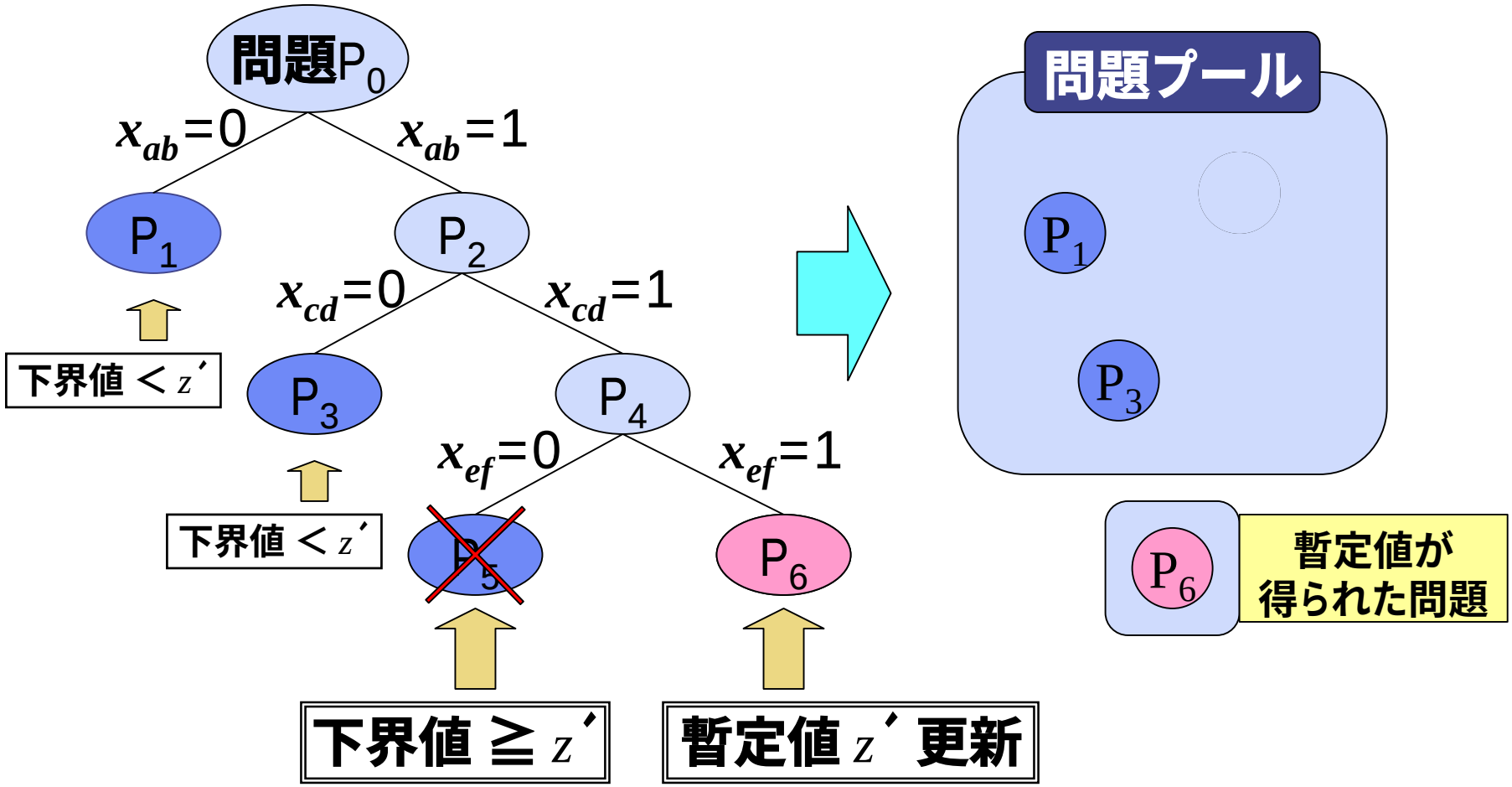


図8　限定操作のイメージ

4．1．5．未分枝の問題の選択

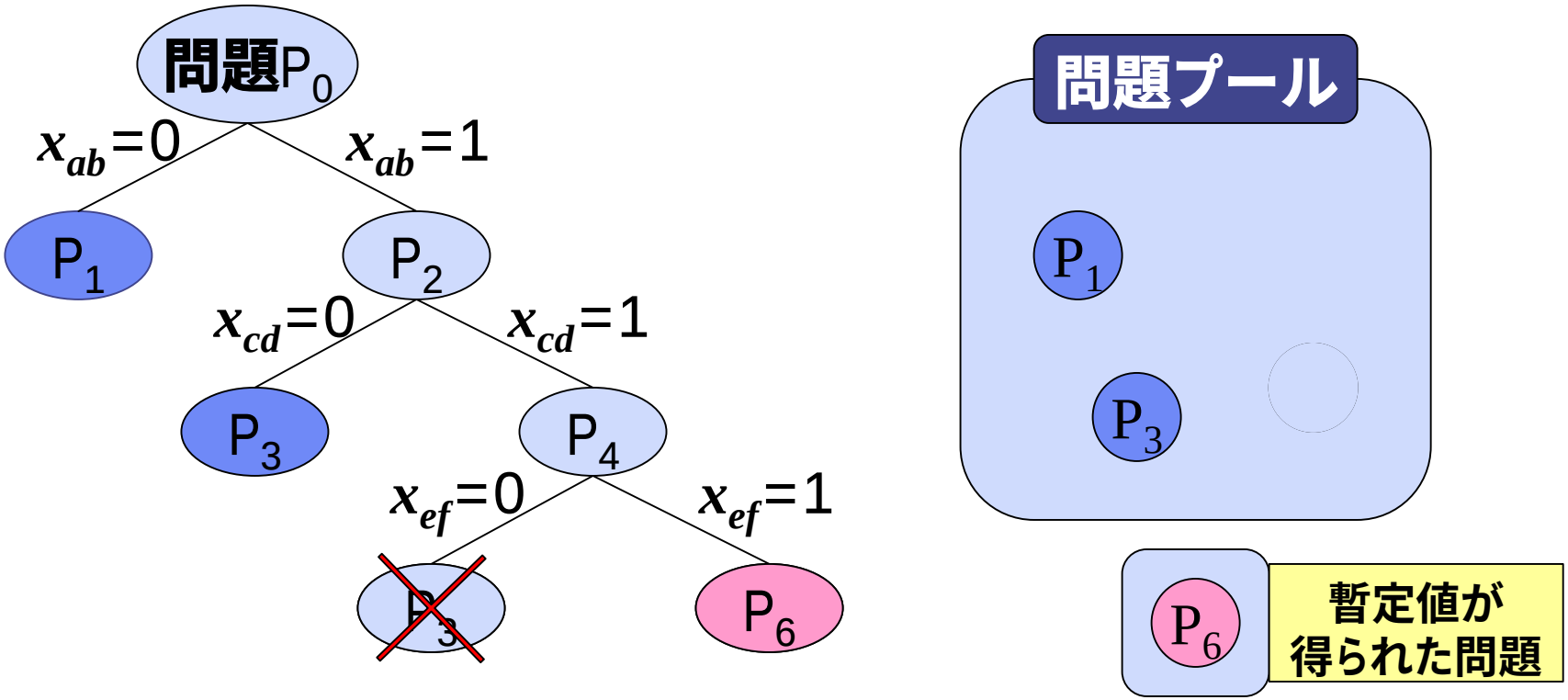
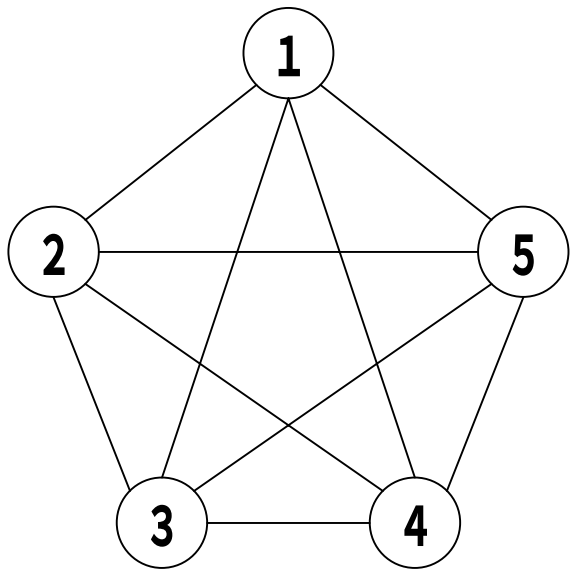


図9　未分枝問題の選択

4．2．ATSPの厳密解法の例



i \ j	1	2	3	4	5
1	-	28	45	39	45
2	23	-	45	25	10
3	43	46	-	28	38
4	15	63	33	-	77
5	3	37	50	72	-

図10　ATSPの例 [1]

4．2．ATSPの厳密解法の例（つづき）

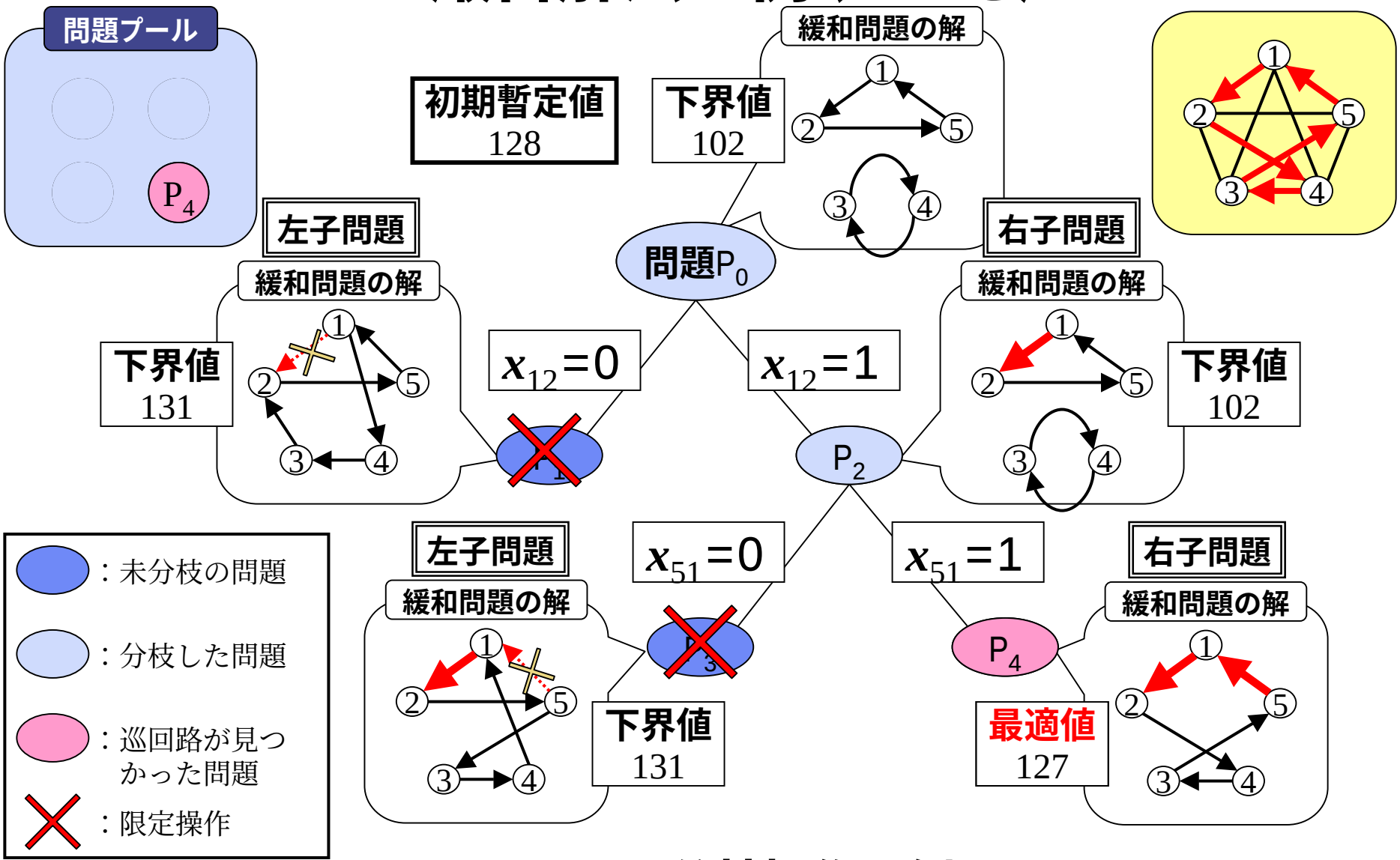


図11　分枝操作の例

4．3．ATSPの解法の流れ

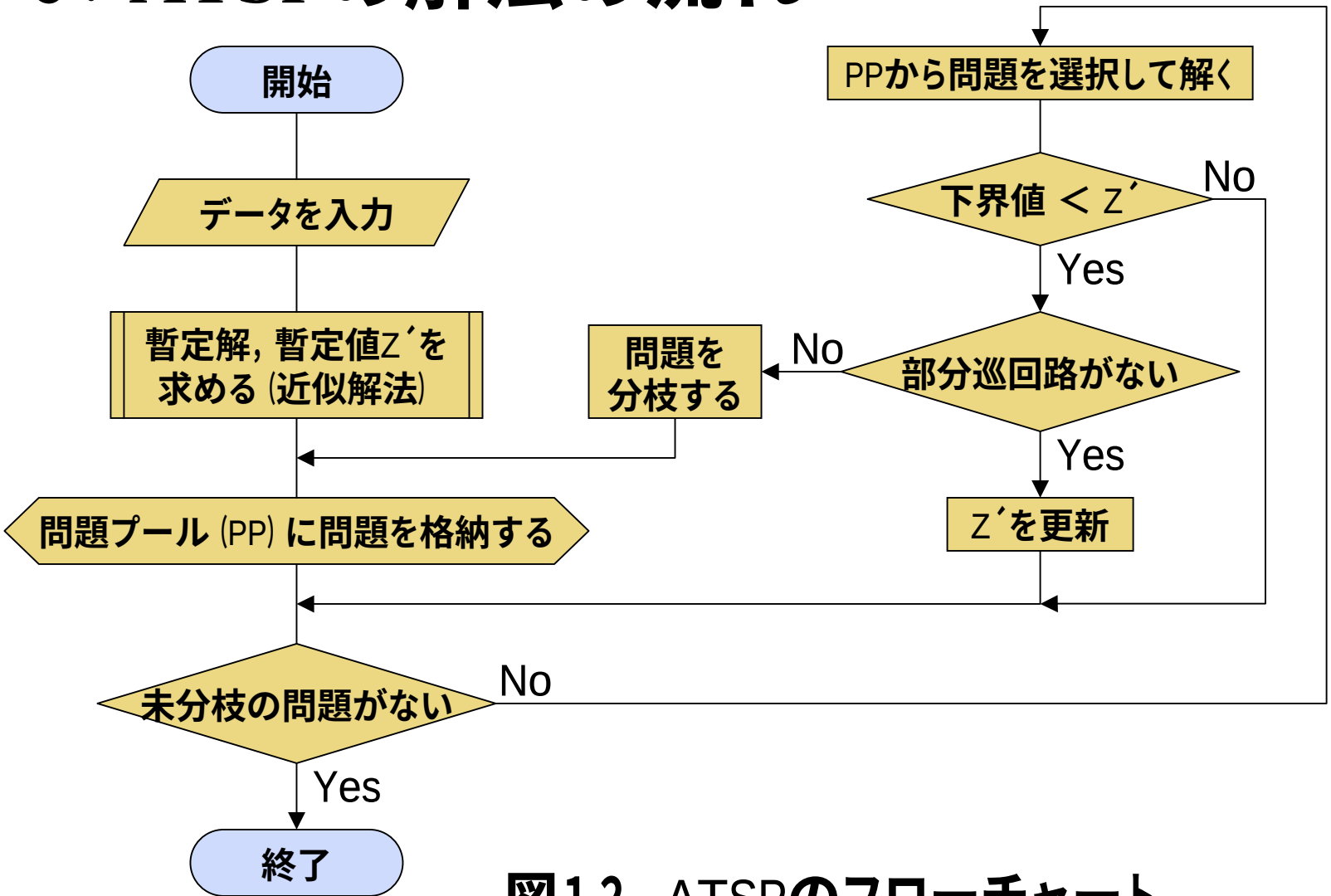


図12　ATSPのフローチャート

5．提案する処理方法

一定時間で最適値が得られない場合に必ず巡回路を得られるようにするための処理方法を提案する．

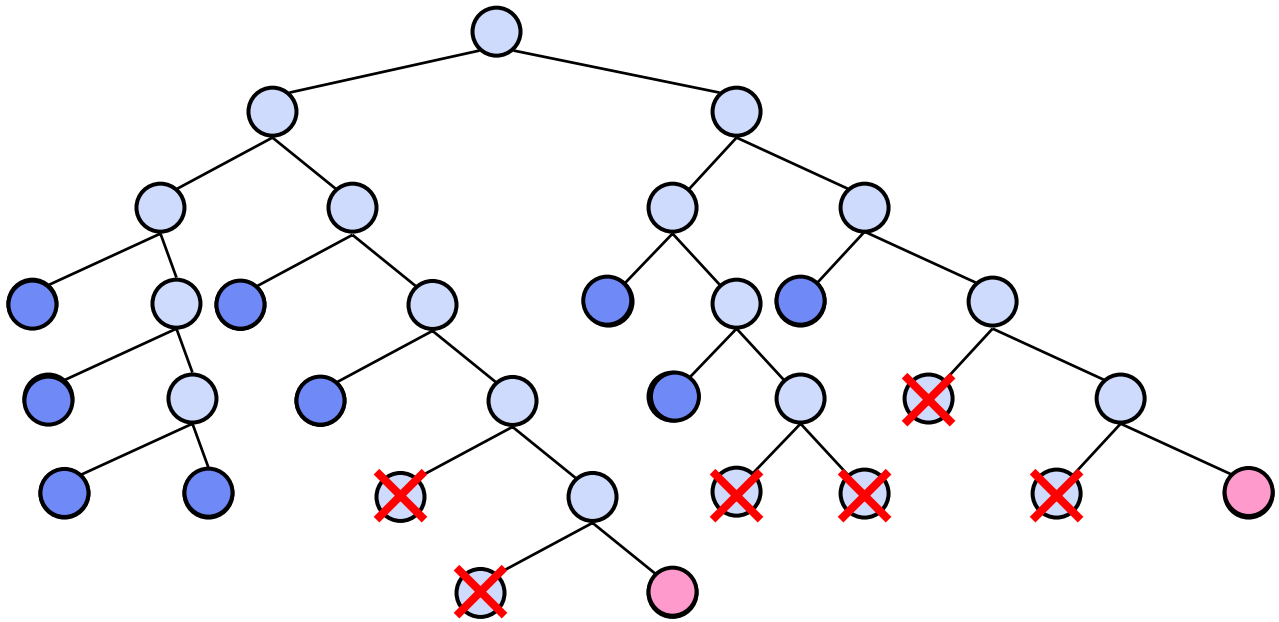
Step1: **実行時間が1分を超えたとき分枝操作を中止**

Step2: **未分枝の問題から下界値の小さいものを100個選択**

Step3: **100個すべてについて巡回路をつくり，その中で費用が最小となるものを求める**

Step4: **求めた最小値と暫定値を比較し，値の小さい方を出力**

5．提案する処理方法（つづき）



Step4: **求めた最小値と暫定値を比較し，値の小さい方を出力**

6．実験

6．1．実験内容

- ベンチマーク問題集 (TSPLIB) のATSPのデータを用いて実行し，厳密解 (巡回路の最小総費用) を求める．
- ランダムに発生させたATSPの費用行列を用いて実行し，厳密解を求める．
- 提案した処理方法により解を求め，得られた解の精度を評価する．

6．2．結果と考察

表1．実験結果（ベンチマーク問題・厳密解法）

データ名	都市数	最良値	出力結果	初期暫定値	生成子問題数	計算時間 [秒]
br17	17	39	39	39	1643594	762 (約13分)
ftv35	36	1473	1473	1473	3150	7
p43	43	5620	5620	5620	428752	1486 (約24分)
ftv44	45	1613	1613	1613	1958	8
ry48p	48	14422	14422	14466	693588	3518 (約59分)
ftv55	56	1608	1608	1608	71970	534 (約9分)
ftv64	65	1839	1839	1854	142428	1633 (約27分)
ft70	70	38673	38673	38721	190282	2691 (約45分)
ftv70	71	1950	1950	1966	619424	9276(約2時間35分)

- **出力結果は最良値** (Best known solutions) **と一致した**
- **計算時間は都市数だけでは決まらない**

6．2．結果と考察 (つづき)

表2．実験結果 (ランダム･厳密解法)

都市数	回数	最適値	初期暫定値	計算時間[秒]
50	1	173	193	2
	2	199	209	3
	3	200	216	5
100	1	246	246	26
	2	204	249	56
	3	224	257	67
200	1	275	357	626
	2	370	370	327
	3	287	364	737
300	1	331	433	2042
	2	343	446	1128
	3	331	440	989

- 300都市の問題まで解くことができた
- 計算時間は都市数にほぼ比例

6．2．結果と考察 (つづき)

表3．実験結果 (ベンチマーク問題･提案した処理方法)

データ名	都市数	最適値	初期暫定値	100個中の最小値	出力結果
br17	17	39	39	40	39
p43	43	5620	5620	5637	5620
ry48p	48	14422	14466	14884	14466
ftv55	56	1608	1608	1629	1608
ftv64	65	1839	1854	1851	1851
ft70	70	38673	38721	38997	38679
ftv70	71	1950	1966	2077	1966

- 100個の中の最小値は最適値に近い値が得られた．

7．まとめ

- 分枝限定法を用いたATSPの厳密解を求めるプログラムを作成した．
- ベンチマーク問題集における最良値（Best known solutions）が最適値であることがわかった．
- 費用をランダムに発生させたATSPを300都市の問題まで解くことができた．
- 近似解法を評価する基準となるものが得られた．

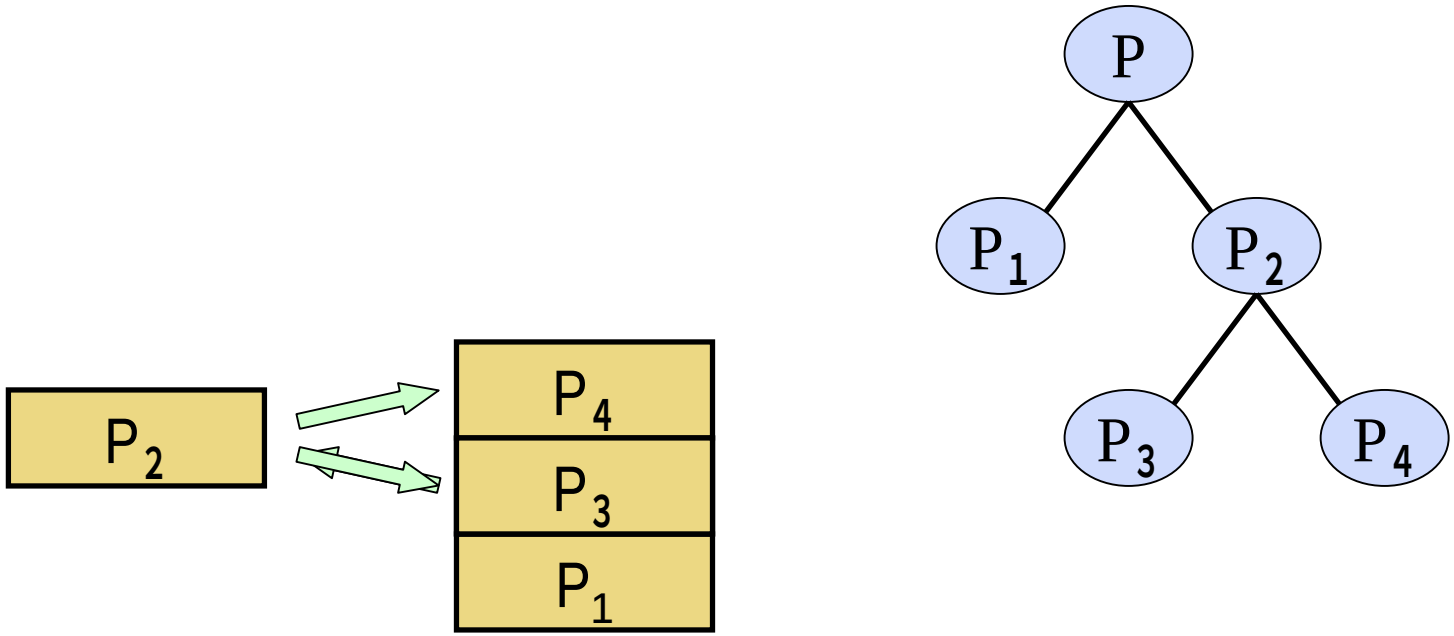
8．今後の課題

- 計算時間の短縮
- 大規模問題への対応
- 他の分枝方法との比較検討
- 提案した処理方法における巡回路の作り方の改善

参考文献

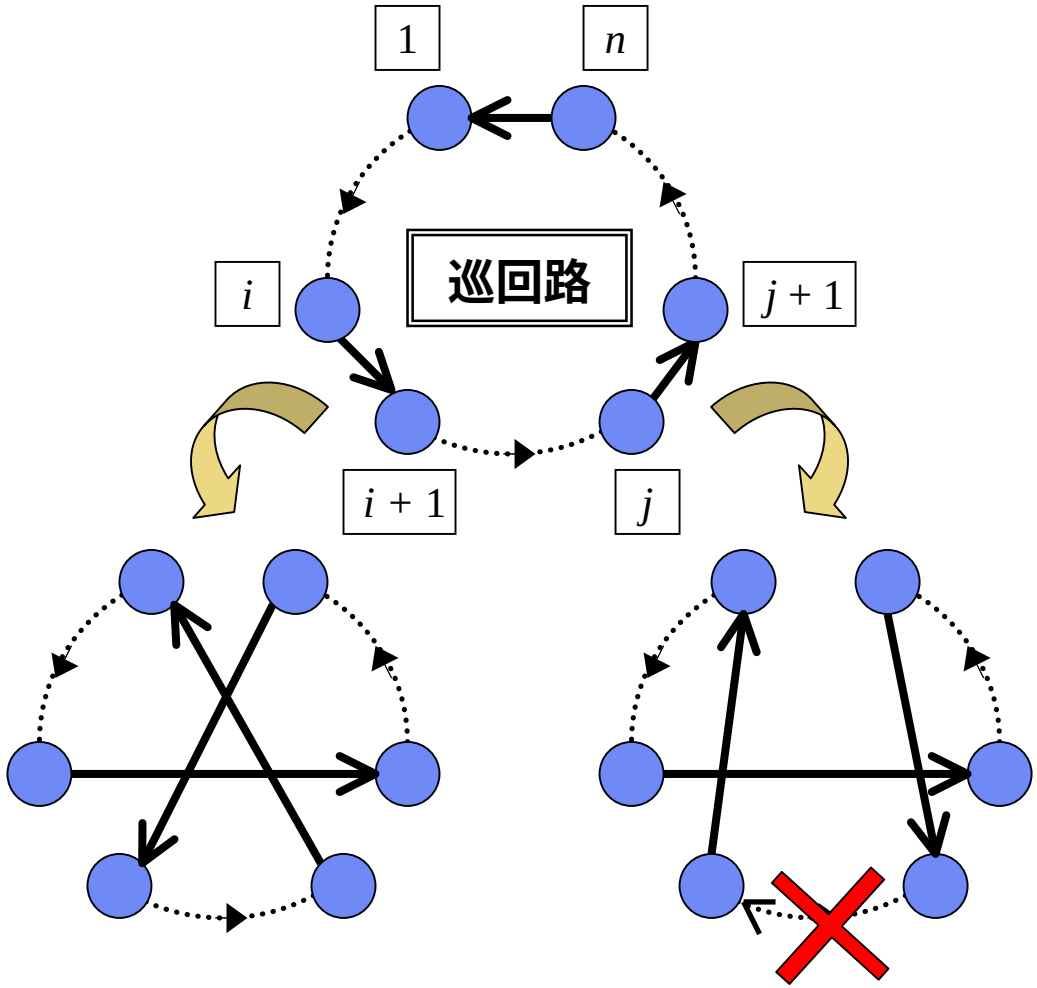
- [1] 今野 浩, 鈴木 久敏：整数計画法と組合せ最適化, 日科技連,1982
- [2] 吉村 央紀:「非対称巡回セールスマン問題に対する近似解法の提案」平成14年度東京理科大学大学院工学研究科経営工学専攻修士論文,2003.
- [3] 升岡 慎吾:「非対称巡回セールスマン問題における近似解法の評価」平成１４年度東京理科大学工学部経営工学科卒業論文, 2003.
- [4] 松井 知己:「組合せ最適化問題入門」
  http://www.misojiro.t.u-tokyo.ac.jp/~tomomi/comb-opt/intro.ppt P 17,18
- [5] 杉原 厚吉, 茨木 俊秀, 浅野 孝夫, 山下 雅史：
  アルゴリズム工学-計算困難問題への挑戦-, 共立出版 , 2001.
- [6] 山本芳嗣,久保 幹雄：巡回セールスマン問題への招待, 朝倉書店 , 2001
- [7] 久保 幹雄,松井知己：組合せ最適化 [短編集] , 朝倉書店 , 1999

配列のイメージ



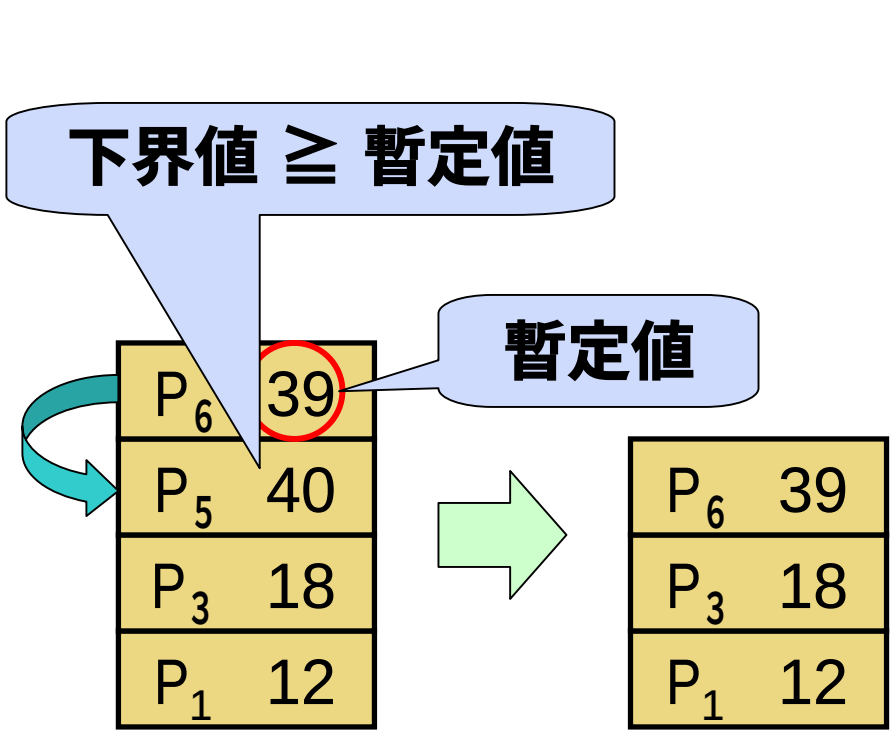
図a　分枝した問題のため方のイメージ

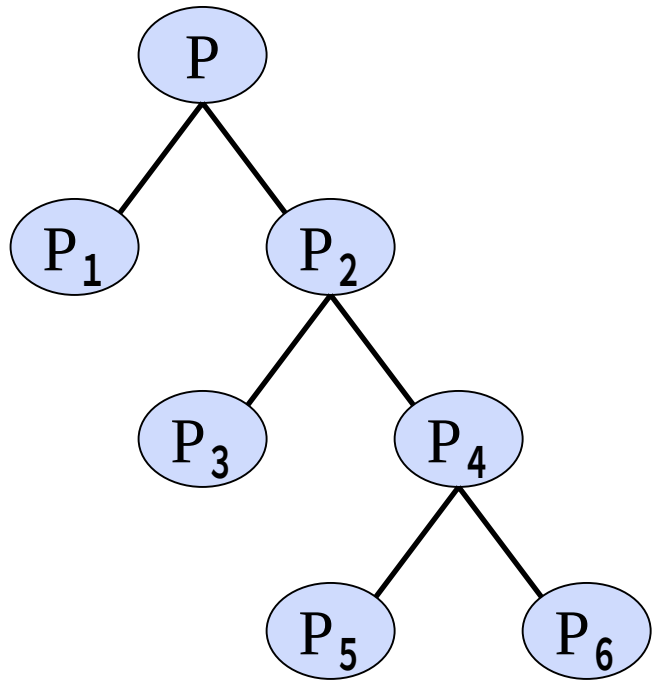
3-opt法



図b　ATSPにおける3-opt法

限定操作をした問題の処理

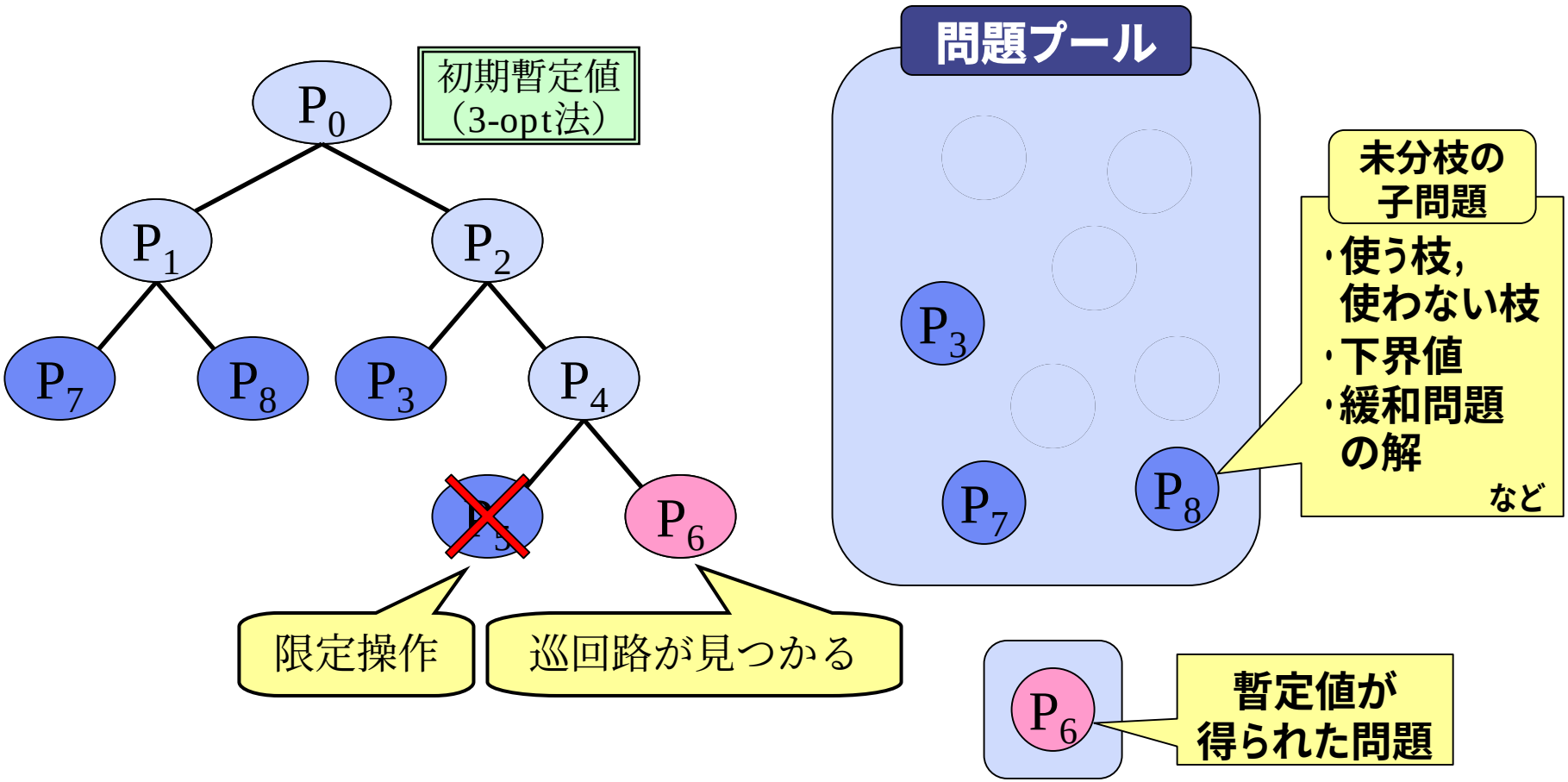




問題　下界値

図c　問題のつめ方

問題プール

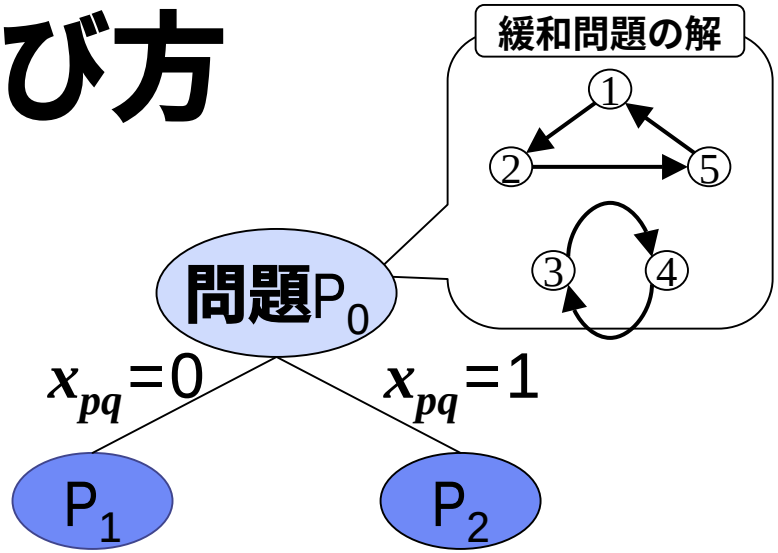


図d　問題プールのイメージ

枝の選び方

$$\theta(p,q) = \min_{j \neq q} \bar{c}_{ij} + \min_{i \neq p} \bar{c}_{iq}$$

$\bar{c}_{ij}$ ：被約費用



（ P_1の最適値 ≧ ① P_1の緩和問題の最適値 ）≧ ② P_0の緩和問題の最適値 + $\theta(p,q)$ [1]

（P_1の下界値）

P_0の緩和問題の解の枝の中から $\max \theta(p,q)$を達成する枝(p,q)を選択

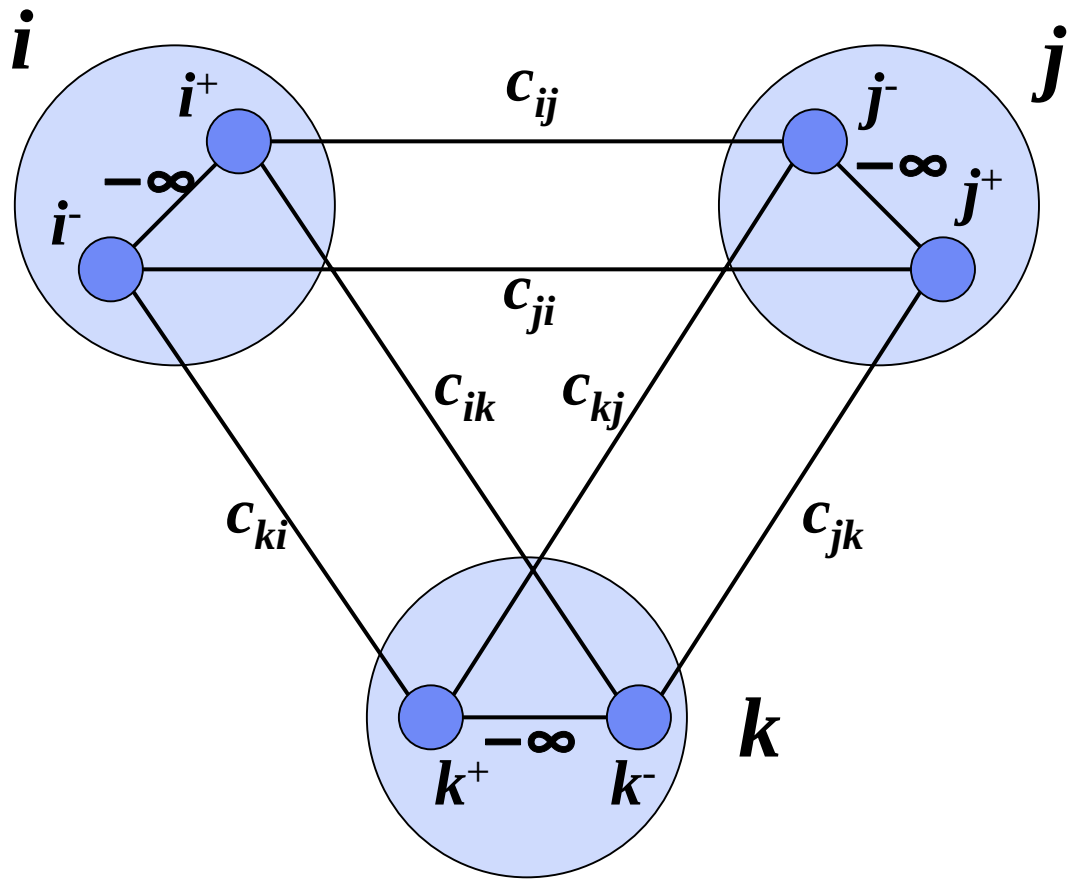
STSPへの変換

$c_{i^+j^-} = c_{ij}$　　$c_{j^-i^+} = c_{ij}$

$c_{j^+i^-} = c_{ji}$　　$c_{i^-j^+} = c_{ji}$

$c_{i^+i^-} = -\infty$　　$c_{i^-i^+} = -\infty$

それ以外はすべて $+\infty$



図e　STSPへの変換

実行画面

図f　プログラムの実行画面

限定操作の説明

下界値：最適値がこの値より小さくなることはないと保障する値

暫定値：暫定値は上界値でもあり、
最適値がこの値より大きくなることはないと保障する値

下界値≧暫定値