# 離散最適化基礎論 第 1 回
# 離散最適化問題と局所探索法

岡本 吉央
okamotoy@uec.ac.jp

電気通信大学

2013 年 10 月 4 日

最終更新：2013 年 10 月 18 日　16:42





## 概要

**目標**

離散最適化のトピックの 1 つとして局所探索法の理論的側面を取り上げ

- 局所探索法を理論的に解析できるようになる
- 局所探索法の理論における研究課題を理解する

**なぜ講義で取り扱う？**

- 局所探索法は重要なアルゴリズム設計技法だから
- 局所探索法の理論をまとめて講義で扱う機会がなかったから





## スケジュール 前半 (予定)

1. 離散最適化問題と局所探索法 (10/4)
- ⋆ 休講 (海外出張) (10/11)
2. 解の表現 (10/18)
3. 近傍関数と近傍グラフ (10/25)
4. 近傍グラフの性質 (1)：連結性 (11/1)
5. 近傍グラフの性質 (2)：直径 (11/8)
6. 性能保証 (1)：厳密近傍 (11/15)
- ⋆ 休講 (調布祭) (11/22)
- ⋆ 休講 (国内出張) (11/29)

注意：予定の変更もありうる





## スケジュール 後半 (予定)

7. 性能保証 (2)：近似保証 (下界) (12/6)
8. 性能保証 (3)：近似保証 (上界) (12/13)
9. 計算量 (1)：上界と下界 (12/20)
- ⋆ 冬季休業 (12/27, 1/3)
10. 計算量 (2)：クラス PLS と PLS 完全性 (1/10)
- ⋆ 休講 (センター試験準備) (1/17)
11. 計算量 (3)：PLS 完全性 (1/24)
12. 大規模近傍 (1/31)
13. 近似局所探索 (2/7)
- ⋆ 期末試験 (2/14?)

注意：予定の変更もありうる





## 情報

教員
- 岡本 吉央 (おかもと よしお)
- 居室：西 4 号館 2 階 206 号室
- E-mail：okamotoy@uec.ac.jp
- Web：http://dopal.cs.uec.ac.jp/okamotoy/

講義資料
- Web：http://dopal.cs.uec.ac.jp/okamotoy/lect/2013/localsearch/
- 注意：資料の印刷等は各学生が自ら行う
- 講義当日の昼 12 時までに，ここに置かれる
- Twitter (@okamoto7yoshio)：置かれたことを知らせる tweet





## 講義資料

**http://dopal.cs.uec.ac.jp/okamotoy/lect/2013/localsearch/**

- スライド
- 印刷用スライド：8 枚のスライドを 1 ページに収めたもの
- 演習問題

「印刷用スライド」と「演習問題」は各自印刷して持参すると便利





## 授業の進め方

講義 (75 分)
- スライドと板書で進める
- スライドのコピーに重要事項のメモを取る

演習 (15 分)
- 演習問題に取り組む
- 不明な点は教員に質問する

退室 (0 分) ←重要
- コメント (授業の感想，質問など) を紙に書いて提出する (匿名可)
- コメントとそれに対する回答は (匿名で) 講義ページに掲載される

オフィスアワー：金曜 5 限
- 質問など
- ただし，いないときもあるので注意





## 演習問題

演習問題の進め方
- 授業の最後 15 分は演習問題を解く時間
- 残った演習問題は復習・試験対策用
- 注意：「模範解答」のようなものは存在しない

演習問題の種類
- 復習問題：講義で取り上げた内容を反復
- 補足問題：講義で省略した内容を補足
- 追加問題：講義の内容に追加

解答の提出
- 演習問題の解答をレポートとして提出してもよい (すること推奨)
- レポートは採点されない (成績に勘案されない)
- レポートは添削されて，返却される

## 評価

期末試験による

- 出題形式
  - 演習問題と同じ形式の問題を 6 題出題する
  - その中の 4 題は演習問題として提示されたものと同一である
  - 全問に解答する
- 配点：1 題 20 点満点，計 120 点満点
- 成績において，100 点以上は 100 点で打ち切り
- 時間：90 分 (おそらく)
- 持ち込み：A4 用紙 1 枚分 (裏表自筆書き込み) のみ可





## 教科書・参考書

教科書

- 指定しない

全般的な参考書

- W. Michiels, E. Aarts, J. Korst, "Theoretical Aspects of Local Search," Springer, 2007.
- E. Aarts, J. K. Lenstra (eds.), "Local Search in Combinatorial Optimization," John Wiley & Sons, 1997.
- 柳浦睦憲，茨木俊秀，「組合せ最適化—メタ戦略を中心として—」，朝倉書店，2001 年．
- 久保幹雄，J. P. ペドロソ，「メタヒューリスティクスの数理」，共立出版，2009 年．
- その他，研究論文





## この講義の約束

- 私語はしない (ただし，演習時間の相談は OK)
- 携帯電話はマナーモードにする
- 携帯電話は使わない
- 音を立てて睡眠しない

約束が守られない場合は退席してもらう場合あり





## 目次

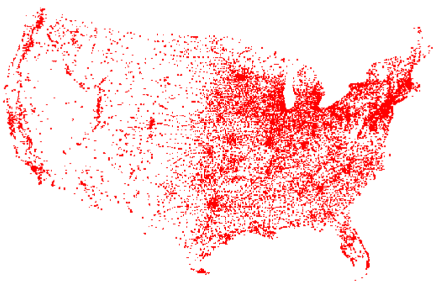






## (1) 巡回セールスマン問題：例

アメリカ合衆国の 13509 都市



http://www.math.uwaterloo.ca/tsp/gallery/idata/usa13509.html





## (1) 巡回セールスマン問題：例 続き

アメリカ合衆国の 13509 都市

http://www.math.uwaterloo.ca/tsp/gallery/itours/usa13509.html





## (1) 巡回セールスマン問題：別の例

プリント基盤の作成 (5934 箇所)

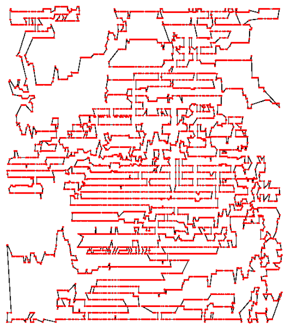
http://www.math.uwaterloo.ca/tsp/gallery/idata/rl5934.html





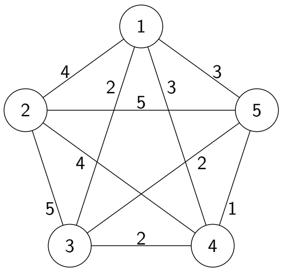
## (1) 巡回セールスマン問題：別の例 続き

プリント基盤の作成 (5934 箇所)



http://www.math.uwaterloo.ca/tsp/gallery/itours/rl5934.html

## (1) 巡回セールスマン問題：小さな例

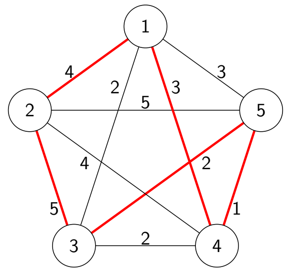



| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |





## (1) 巡回セールスマン問題：小さな例 続き




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |



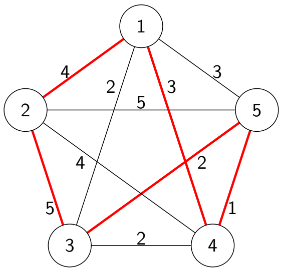


## (1) 巡回セールスマン問題

**巡回セールスマン問題**

**入力**

- $n \times n$ 対称行列 $C = (c(i,j))$




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |



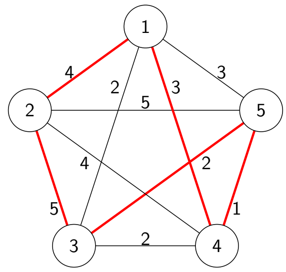


## (1) 巡回セールスマン問題

**巡回セールスマン問題**

**許容解**

- $\{1,\ldots,n\}$ 上の順列 $\tau$ ($1,\ldots,n$ を一列に並べたもの)
(伝統的に，巡回路と呼ばれる)




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |

$\tau(1) = 1,\ \tau(2) = 2,$
$\tau(3) = 3,\ \tau(4) = 5,$
$\tau(5) = 4$



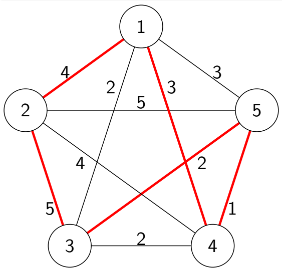


## (1) 巡回セールスマン問題

**巡回セールスマン問題**

**目的**

- $\displaystyle\sum_{i=1}^{n-1} c(\tau(i),\tau(i+1)) + c(\tau(n),\tau(1))$ の最小化




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |

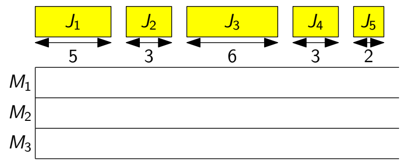
$\tau(1) = 1,\ \tau(2) = 2,$
$\tau(3) = 3,\ \tau(4) = 5,$
$\tau(5) = 4$





## (2) 最終完了時刻最小化スケジューリング問題

$m$ 個の同一な機械 $M_1,\ldots,M_m$，$n$ 個のジョブ $J_1,\ldots,J_n$，
各ジョブ $J_i$ の処理時間 $p_i$




**行うこと：すべてのジョブを機械で処理する**

$$p_1 = 5,\ p_2 = 3,\ p_3 = 6,\ p_4 = 3,\ p_5 = 2$$

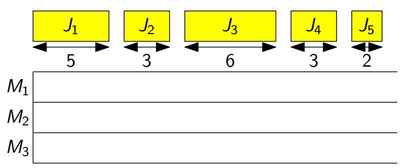




## (2) 最終完了時刻最小化スケジューリング問題

**決めること (スケジューリング，scheduling)**

- 各ジョブをどの機械に割り当てるか？
- 各機械で，割り当てられたジョブをどの順に処理するか？



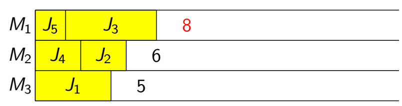

**ジョブの中断はないものとする** (non-preemptive scheduling)





## (2) 最終完了時刻最小化スケジューリング問題

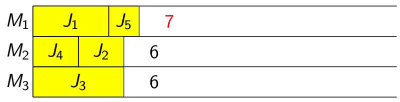
**目的：最終完了時刻の最小化**




**このスケジュールにおける最終完了時刻** $= 8$

## (2) 最終完了時刻最小化スケジューリング問題

目的：最終完了時刻の最小化




このスケジュールにおける最終完了時刻 $= 7$





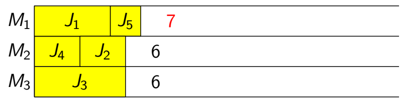
## (2) 最終完了時刻最小化スケジューリング問題

**最終完了時刻最小化スケジューリング問題**

入力
- $m$ 個の同一な機械 $M_1, \ldots, M_m$
- $n$ 個のジョブ $J_1, \ldots, J_n$
- 各ジョブ $J_i$ の処理時間 $p_i$

許容解
- スケジュール (機械へのジョブの割当，処理順)

目的
- 最終完了時刻の最小化

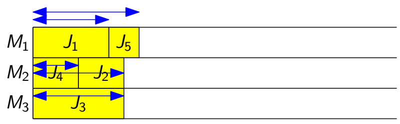







## (3) 完了時刻和最小化スケジューリング問題

各ジョブに着目して，ジョブの完了時刻の和を最小化



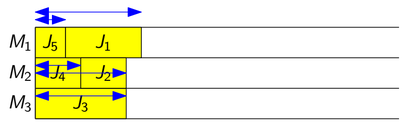

このスケジュールにおける完了時刻和 $= 5 + 7 + 3 + 6 + 6 = 27$





## (3) 完了時刻和最小化スケジューリング問題

各ジョブに着目して，ジョブの完了時刻の和を最小化




このスケジュールにおける完了時刻和 $= 2 + 7 + 3 + 6 + 6 = 24$





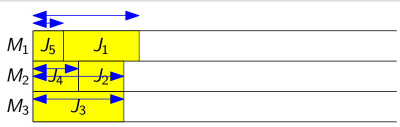
## (3) 完了時刻和最小化スケジューリング問題

**完了時刻和最小化スケジューリング問題**

入力
- $m$ 個の同一な機械 $M_1, \ldots, M_m$
- $n$ 個のジョブ $J_1, \ldots, J_n$
- 各ジョブ $J_i$ の処理時間 $p_i$

許容解
- スケジュール (機械へのジョブの割当，処理順)

目的
- 完了時刻和の最小化



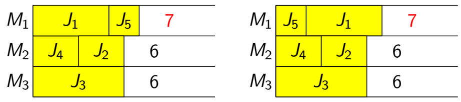





## 注意

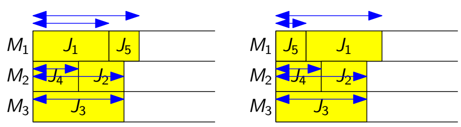
**最終完了時刻最小化スケジューリング問題では**

各機械での処理順が変わっても，最終完了時刻は変わらない




**完了時刻和最小化スケジューリング問題では**

各機械での処理順が変わると，完了時刻和は変わる (かもしれない)

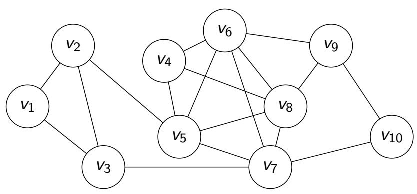







## (4) グラフ等分割問題

偶数個の頂点を持つ無向グラフ $G$



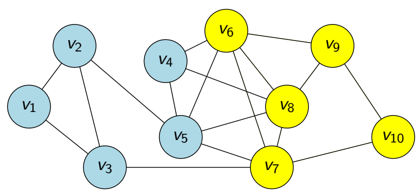

頂点集合 $V$，辺集合 $E$





## (4) グラフ等分割問題

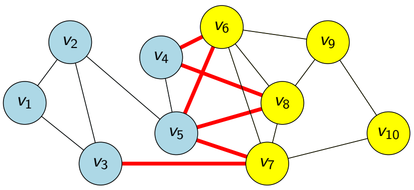
頂点集合 $V$ を二等分割




$V$ を $\boxed{A}$ と $\boxed{B}$ に分割

## (4) グラフ等分割問題

$A$ と $B$ を跨る 辺 の数を最小化



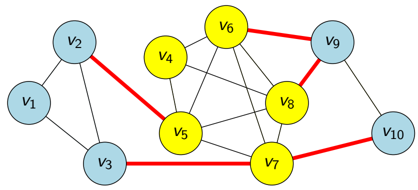

$A$ と $B$ を跨る 辺 の数 $= 6$





## (4) グラフ等分割問題

$A$ と $B$ を跨る 辺 の数を最小化




$A$ と $B$ を跨る 辺 の数 $= 5$





## (4) グラフ等分割問題

**グラフ等分割問題**

**入力**

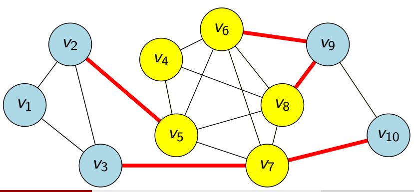
- 無向グラフ $G$ (頂点数は偶数)

**許容解**

- $G$ の頂点集合の二等分割 $\{A, B\}$

**目的**

- $A$ と $B$ を跨る辺の数の最小化








## 離散最適化問題：一般論に向けて

**ここまで，4 つの問題を見てきた**

(1) 巡回セールスマン問題
(2) 最終完了時刻最小化スケジューリング問題
(3) 完了時刻和最小化スケジューリング問題
(4) グラフ等分割問題

**これらの共通点**

次の 3 つを持っている

- 入力
- 許容解
- 目的

それに着目して，「離散最適化問題」の一般論を構築する





## 離散最適化問題とは ?: 一般論

**離散最適化問題 (discrete optimization problem) とは？**

離散最適化問題は以下に挙げる $S, f$ を使って表される最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

のことで，ただし，

- $S$ は 有限 集合
- $f \colon S \to \mathbb{R}$ は関数

である





## 離散最適化問題とは ?: 一般論 続き

**離散最適化問題 (discrete optimization problem) とは？**

離散最適化問題は以下に挙げる $S, f$ を使って表される最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

のことで，ただし，...

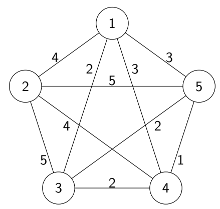
**用語**

- $S$：**許容集合** (feasible set)
  - 「入力」と「何が許容解であるかという条件」によって定まる
- $S$ の各要素：**許容解** (feasible solution)
- $f$：**目的関数** (objective function)
  - 「目的」によって定まる





## 例：巡回セールスマン問題




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |

**許容集合** $=\{(1,2,3,4,5),(1,2,3,5,4),(1,2,5,3,4),(1,5,2,3,4),$
$(5,1,2,3,4),(5,1,2,4,3),(1,5,2,4,3),(1,2,5,4,3),$
$(1,2,4,5,3),(1,2,4,3,5),(1,4,2,3,5),(1,4,2,5,3),$
$(1,4,5,2,3),(1,5,4,2,3),(5,1,4,2,3),(5,4,1,2,3),$
$(4,5,1,2,3),(4,1,5,2,3),(4,1,2,5,3),(4,1,2,3,5),\ldots\}$

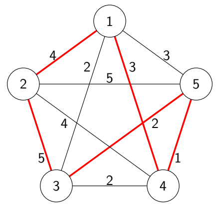




## 例：巡回セールスマン問題 (許容集合をしっかりと書くと)

**許容集合** $= \{$ (1, 2, 3, 4, 5), (1, 2, 3, 5, 4), (1, 2, 5, 3, 4), (1, 5, 2, 3, 4), (5, 1, 2, 3, 4), (5, 1, 2, 4, 3), (1, 5, 2, 4, 3),
(1, 2, 5, 4, 3), (1, 2, 4, 5, 3), (1, 2, 4, 3, 5), (1, 4, 2, 3, 5), (1, 4, 2, 5, 3), (1, 4, 5, 2, 3), (1, 5, 4, 2, 3), (5, 1, 4, 2, 3),
(5, 4, 1, 2, 3), (4, 5, 1, 2, 3), (4, 1, 5, 2, 3), (4, 1, 2, 5, 3), (4, 1, 2, 3, 5), (4, 1, 3, 2, 5), (4, 1, 3, 5, 2), (4, 1, 5, 3, 2),
(4, 5, 1, 3, 2), (5, 4, 1, 3, 2), (5, 1, 4, 3, 2), (1, 5, 4, 3, 2), (1, 4, 5, 3, 2), (1, 4, 3, 5, 2), (1, 4, 3, 2, 5), (1, 3, 4, 2, 5),
(1, 3, 4, 5, 2), (1, 3, 5, 4, 2), (1, 5, 3, 4, 2), (5, 1, 3, 4, 2), (5, 1, 3, 2, 4), (1, 5, 3, 2, 4), (1, 3, 5, 2, 4), (1, 3, 2, 5, 4),
(1, 3, 2, 4, 5), (3, 1, 2, 4, 5), (3, 1, 2, 5, 4), (3, 1, 5, 2, 4), (3, 5, 1, 2, 4), (5, 3, 1, 2, 4), (5, 3, 1, 4, 2), (3, 5, 1, 4, 2),
(3, 1, 5, 4, 2), (3, 1, 4, 5, 2), (3, 1, 4, 2, 5), (3, 4, 1, 2, 5), (3, 4, 1, 5, 2), (3, 4, 5, 1, 2), (3, 5, 4, 1, 2), (5, 3, 4, 1, 2),
(5, 4, 3, 1, 2), (4, 5, 3, 1, 2), (4, 3, 5, 1, 2), (4, 3, 1, 5, 2), (4, 3, 1, 2, 5), (4, 3, 2, 1, 5), (4, 3, 2, 5, 1), (4, 3, 5, 2, 1),
(4, 5, 3, 2, 1), (5, 4, 3, 2, 1), (5, 3, 4, 2, 1), (3, 5, 4, 2, 1), (3, 4, 5, 2, 1), (3, 4, 2, 5, 1), (3, 4, 2, 1, 5), (3, 2, 4, 1, 5),
(3, 2, 4, 5, 1), (3, 2, 5, 4, 1), (3, 5, 2, 4, 1), (5, 3, 2, 4, 1), (5, 3, 2, 1, 4), (3, 5, 2, 1, 4), (3, 2, 5, 1, 4), (3, 2, 1, 5, 4),
(3, 2, 1, 4, 5), (2, 3, 1, 4, 5), (2, 3, 1, 5, 4), (2, 3, 5, 1, 4), (2, 5, 3, 1, 4), (5, 2, 3, 1, 4), (5, 2, 3, 4, 1), (2, 5, 3, 4, 1),
(2, 3, 5, 4, 1), (2, 3, 4, 5, 1), (2, 3, 4, 1, 5), (2, 4, 3, 1, 5), (2, 4, 3, 5, 1), (2, 4, 5, 3, 1), (2, 5, 4, 3, 1), (5, 2, 4, 3, 1),
(5, 4, 2, 3, 1), (4, 5, 2, 3, 1), (4, 2, 5, 3, 1), (4, 2, 3, 5, 1), (4, 2, 3, 1, 5), (4, 2, 1, 3, 5), (4, 2, 1, 5, 3), (4, 2, 5, 1, 3),
(4, 5, 2, 1, 3), (5, 4, 2, 1, 3), (5, 2, 4, 1, 3), (2, 5, 4, 1, 3), (2, 4, 5, 1, 3), (2, 4, 1, 5, 3), (2, 4, 1, 3, 5), (2, 1, 4, 3, 5),
(2, 1, 4, 5, 3), (2, 1, 5, 4, 3), (2, 5, 1, 4, 3), (5, 2, 1, 4, 3), (5, 2, 1, 3, 4), (2, 5, 1, 3, 4), (2, 1, 5, 3, 4), (2, 1, 3, 5, 4),
(2, 1, 3, 4, 5)$\}$

## 例：巡回セールスマン問題



| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |

許容解 $(1, 2, 3, 5, 4)$ の目的関数値 $= 15$





## 最適解

**最適解 (optimal solution) とは？**

離散最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

の**最適解**とは 許容解 $x \in S$ で，次を満たすもののこと

任意の許容解 $x' \in S$ に対して，$f(x) \leq f(x')$

**注意**

- 最適解は複数存在するかもしれない
- $S \neq \emptyset$ ならば，最適解は必ず存在する
  (「連続最適化」では，これが成り立つとは限らない)





## 最適値

**最適値 (optimal value) とは？**

離散最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

の**最適値**とは，最適解の目的関数値のこと

「最適解」と「最適値」は異なる概念であり，明確に使い分ける

**事実 (演習問題)**

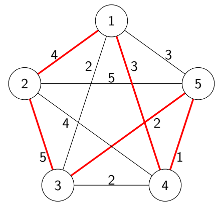
$x_1, x_2 \in S$ が最適解 $\Rightarrow f(x_1) = f(x_2)$

最適解の目的関数値は常に等しい (最適値はただ 1 つしか存在しない)





## 例：巡回セールスマン問題



| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 2 | 3 | 3 |
| 2 | 4 | 0 | 5 | 4 | 5 |
| 3 | 2 | 5 | 0 | 2 | 2 |
| 4 | 3 | 4 | 2 | 0 | 1 |
| 5 | 3 | 5 | 2 | 1 | 0 |

許容解 $(1, 2, 3, 5, 4)$ の目的関数値 $= 15$

**疑問??**

この許容解は最適解だろうか？ (最適値は 15 だろうか？)

確認には忍耐がいる (忍耐だけで済まないかもしれない)

**「許容解 $(1, 2, 3, 5, 4)$ の目的関数値 $= 15$」から分かること**

最適値 $\leq 15$





## 離散最適化問題を解くための原理 と 局所探索法

**どのようにして離散最適化問題を解くのか？**

- 最適値の上界を与える (「最適値 $\leq 15$」のようなもの)
  - 目的関数値が小さな許容解を見つければよい
  - そのための手法がいくつか開発されている (← 局所探索法など)
- 最適値の下界を与える (「最適値 $\geq 10$」のようなもの)
  - 目的関数値がある値よい小さくならないという証明を見つければよい
  - そのための手法がいくつか開発されている
- 上界と下界が一致すれば，それは最適値である
  - 上界を与えていた許容解が最適解となる

離散最適化問題を解く $=$ 離散最適化問題の最適解を発見する

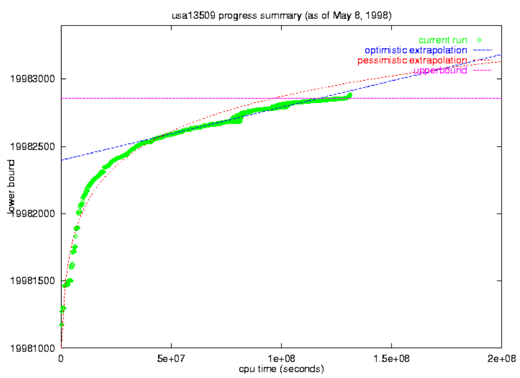
**局所探索法の役割**

「よい許容解を見つける」ための一般的な枠組み





## 上界と下界の一致




http://www.math.uwaterloo.ca/tsp/usa13509/usa13509.html





## 目次

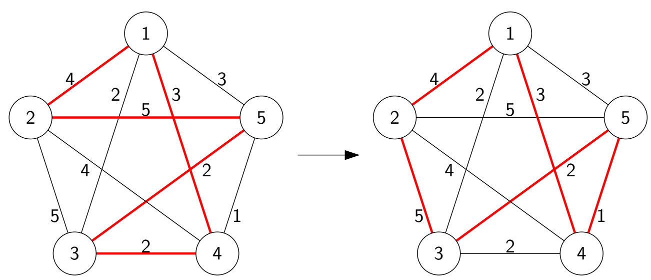






## 局所探索法の基本的なアイディア

- 許容解を常に 1 つ保持しておく
- 保持している許容解に小さな変更を施して，よりよい許容解を得る



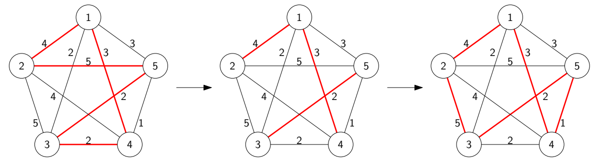
- 小さな変更のことを「局所操作」と呼んでいる
- 1 つの問題に対して，様々な局所操作を考えることができる

## 例 1：巡回セールスマン問題に対する 2opt 操作

**巡回セールスマン問題に対する 2opt 操作**

$\tau$：現在保持している巡回路

1. $\tau$ が使っている辺を 2 つ取り除く
2. $\tau$ が使っていなかった辺を 2 つ追加して，新たな巡回路を得る





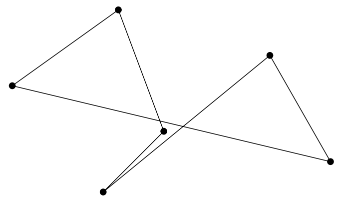


## 例 1：2opt 操作の反復による巡回路の改善

**2opt 操作の反復による巡回路の改善**

$\tau$：現在保持している巡回路

1. 以下を繰り返し
   1. $\tau$ に 2opt 操作を適用して，$\tau$ よりもよい巡回路 $\tau'$ が得られるならば $\tau \leftarrow \tau'$
   2. そうでなければ，繰返しを終了
2. $\tau$ を出力





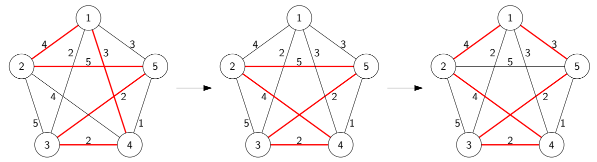


## 例 2：巡回セールスマン問題に対する頂点挿入操作

**巡回セールスマン問題に対する頂点挿入操作**

$\tau$：現在保持している巡回路

1. 頂点を 1 つ選び，$\tau$ においてその頂点をとばす
2. とばした頂点をどこかに挿入して，新たな巡回路を得る





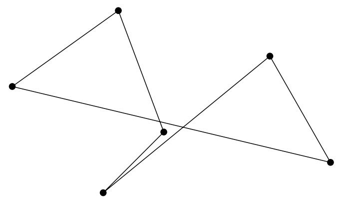


## 例 2：頂点挿入操作の反復による巡回路の改善

**頂点挿入操作の反復による巡回路の改善**

$\tau$：現在保持している巡回路

1. 以下を繰り返し
   1. $\tau$ に頂点挿入操作を適用し，$\tau$ よりよい巡回路 $\tau'$ が得られるならば $\tau \leftarrow \tau'$
   2. そうでなければ，繰返しを終了
2. $\tau$ を出力





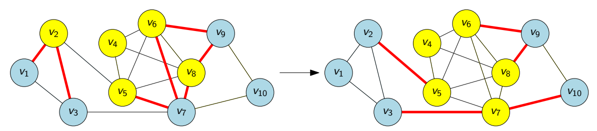


## 例 3：グラフ等分割問題に対する交換操作

**グラフ等分割問題に対する交換操作**

$\{A, B\}$：現在保持している二等分割

1. 頂点 $a \in A$ と頂点 $b \in B$ を選ぶ
2. $a$ を $A$ から $B$ に動かし，$b$ を $B$ から $A$ に動かす





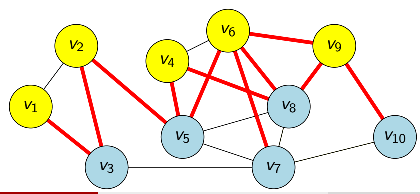


## 例 3：交換操作の反復による二等分割の改善

**交換操作の反復による二等分割の改善**

$\{A, B\}$：現在保持している二等分割

1. 以下を繰り返し
   1. $\{A, B\}$ に交換操作を適用し，$\{A, B\}$ よりよい二等分割 $\{A', B'\}$ が得られるならば $\{A, B\} \leftarrow \{A', B'\}$
   2. そうでなければ，繰返しを終了
2. $\{A', B'\}$ を出力








## 目次







## 局所探索法：一般論に向けて

**ここまで，3 つの局所操作を見てきた**

(1) 巡回セールスマン問題に対する 2opt 操作
(2) 巡回セールスマン問題に対する頂点挿入操作
(3) グラフ等分割問題に対する交換操作

それらを繰り返し適用して，許容解を改善する方法を見てきた

**これらの共通点**

- 変更は部分的に対して適用される
- それを反復することで，改善を行っている
- 改善ができないとき，終了している

それに着目して，「局所探索法」の一般論を構築する

## 局所探索法の一般論に向けて：局所操作から近傍へ

- 一般論に向けて，離散最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

を考える

**ナイーブな視点**

局所操作は 1 つの許容解 $x \in S$ から別の許容解 $x' \in S$ を作る

- 局所操作は $S$ から $S$ への関数か？ $\rightsquigarrow$ そうではなさそう
- $\because$ 局所操作の適用法に依存して，$x$ から多様な許容解が作られる

**正しい視点**

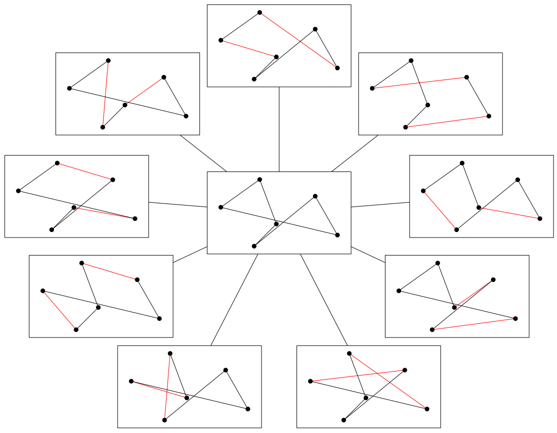
局所操作は 1 つの許容解 $x \in S$ から作られる許容解の集合を定める

- $x$ から $S$ の部分集合 ($x$ の近傍) を定める関数として一般論を構成





## 局所操作から近傍へ







## 近傍関数と近傍

**近傍関数 (neighborhood function) とは？**

離散最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

に対する近傍関数とは，関数 $N\colon S \to 2^S$ で，
任意の $x \in S$ に対して $x \in N(x)$ を満たすもののこと

復習：$2^S$ は $S$ の部分集合を全部集めた集合 ($S$ の冪集合 (べき集合))

**近傍 (neighborhood) とは？**

$N$ に関する許容解 $x \in S$ の近傍とは $N(x)$ のこと





## 局所探索法

**局所探索法 (local search)：一般的枠組み**

0. 許容解 $x \in S$ を適当に見つける
1. 以下を繰り返し
   1. $N(x)$ に $x$ よりもよい解 $x'$ が存在するならば
      (すなわち，ある $x' \in N(x)$ が存在して $f(x') < f(x)$ ならば)
      $x \leftarrow x'$
   2. そうでなければ繰り返しを終了
2. $x$ を出力





## 局所最適解

**局所最適解 (local optimal solution) とは？**

離散最適化問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条　件} & x \in S \end{array}$$

とそれに対する近傍関数 $N\colon S \to 2^S$ を考えたとき，
$x \in S$ が $N$ に関する局所最適解であるとは，次を満たすこと

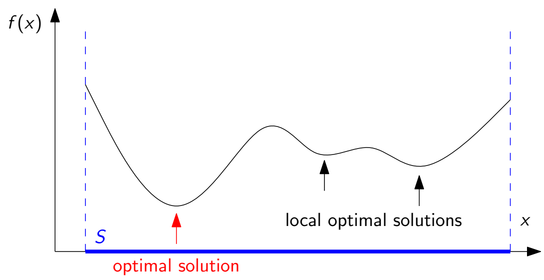
$$\text{任意の } x' \in N(x) \text{ に対して，} f(x) \leq f(x')$$

局所最適解と対比させて，最適解のことを大域最適解と言うことがある





## 連続最適化での局所最適解のイメージ




離散最適化でのイメージは次々回以降に





## 局所最適解の性質 (1)

$S$ を許容集合，$f$ を目的関数，$N$ を近傍関数とする状況を考える

**定理 1.1**

局所探索法の出力は $N$ に関する局所最適解である

**証明**

- $x$ を局所探索法の出力とする
- 局所探索法の繰り返しが終了したことから，
  任意の $x' \in N(x)$ に対して，$f(x') \geq f(x)$
- この条件は $x$ が $N$ に関する局所最適解であるための定義そのもの
- したがって，$x$ は $N$ に関する局所最適解である □





## 局所最適解の性質 (2)

$S$ を許容集合，$f$ を目的関数，$N$ を近傍関数とする状況を考える

**定理 1.2**

最適解は $N$ に関する局所最適解である

**証明**

- $x$ を最適解であるとする
- 最適解の定義から，
  任意の $x' \in S$ に対して，$f(x') \geq f(x)$
- $N(x) \subseteq S$ であるから，
  任意の $x' \in N(x)$ に対しても，$f(x') \geq f(x)$
- この条件は $x$ が $N$ に関する局所最適解であるための定義そのもの
- したがって，$x$ は $N$ に関する局所最適解である □

## 目次







## この講義のねらい

**メインテーマ**

局所探索法はどれほどよい手法なのだろうか？

次の 2 点を**理論的**に解析する

- 性能保証
  - 局所最適解はどれほど最適解に「近い」のか？
- 計算量
  - 局所探索法は素早く終了するのか？
  - 局所最適解は簡単に見つかるのか？

**ねたばらし**

人々が思うほど，(最悪時において) 理論的によいものではない

- 論点：どんなときよくて，どんなとき悪いのか？





## 今日の復習

**重要な概念**

- 離散最適化問題 (一般論)
  - 巡回セールスマン問題，スケジューリング問題，グラフ等分割問題
- 最適解，最適値
- 近傍関数
  - 局所操作 (2opt 操作，頂点挿入操作，交換操作)
- 局所探索法
- 局所最適解

**オンライン参考資料**

- Traveling Salesman Problem
  http://www.math.uwaterloo.ca/tsp/index.html
- 10th DIMACS Implementation Challenge – Graph Partitioning and Graph Clustering –
  http://www.cc.gatech.edu/dimacs10/





## 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