

# 情報メディア論 2

2010-11

Evolutionary  
Computation

# Introduction to Soft Computing

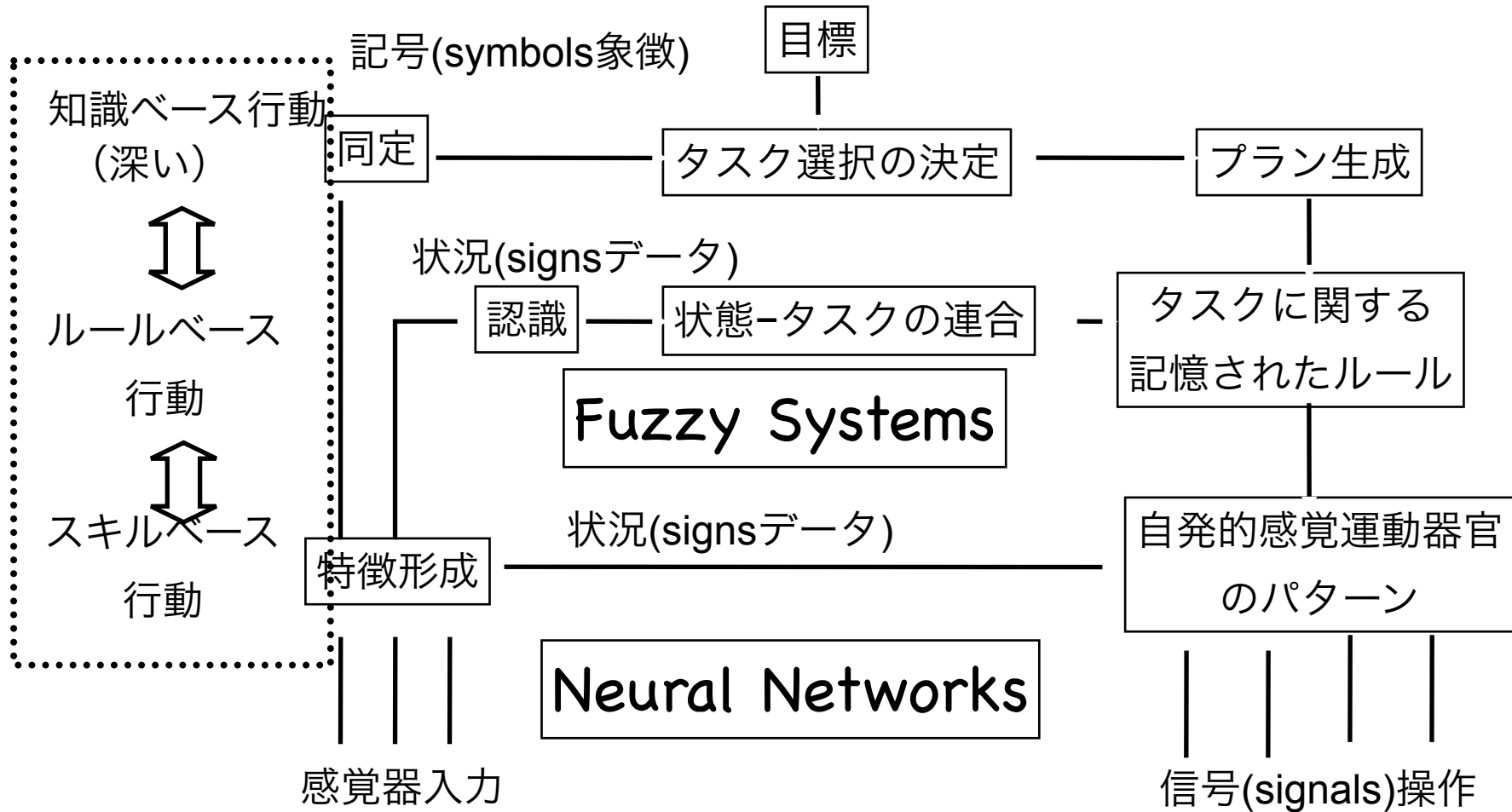
言語表現      ファジィ理論(Fuzzy Theory)

学習能力      ニューラルネットワーク(Artificial  
Neural Network)

探索      進化的アルゴリズム(Evolutionary  
Algorithm)

C.F.

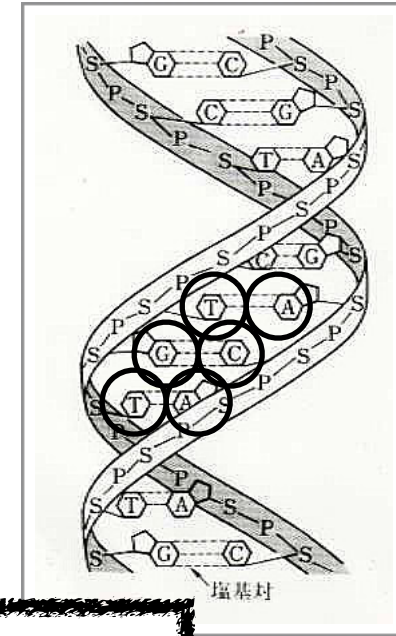
# 人間の行動様式分析 Rusmussen 1980



# 『進化』 計算

- ④ 数理的アプローチ(Science)：集団遺伝学
  - ④ 微分方程式、確率過程をモデルに利用
  - $X(n+1) = a * X(n) * (1 - X(n))$
- ④ 工学的応用(Engineering)：生物進化の計算機上での模擬
  - ④ 適応、学習、最適化問題への適用

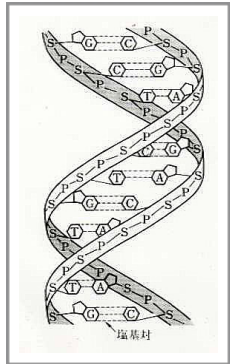
# DNAの構造模型とシステム論的解釈



1. 遺伝情報の記号性(ATCG)
2. 遺伝情報のプログラム・データとしての2面性  
— 自己複製・増殖
3. 突然変異や交叉による新しい遺伝子の生成
4. 表現型の環境への適合度合いによる評価
5. 淘汰に中立な遺伝子による頑健性、創造性

図は<http://www.edo.toride.ibaraki.jp/edotori160/evt031.htm>より

# EC概要



問題の定式化（最適化問題として）

コーディング

適応度（評価）関数の設定

遺伝演算子（演算子）の設定

各種パラメータの設定

アルゴリズムの実行

結果の吟味

# ECアルゴリズム概要 (GAを例に)

個体数 : population

1計算単位=世代 : generation

1001001101

初期個体群  $P(0)$ ,  $t=0$

YES

終了判定

終了

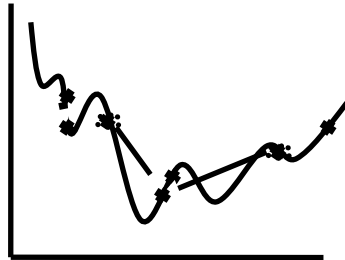
$P(t)$ の各個体の適応度関数評価

選択演算 :  $P(t) \rightarrow P'(t)$

交叉演算 :  $P'(t) \rightarrow P''(t)$

突然変異演算 :  $P''(t) \rightarrow P(t+1)$

$t=t+1$



# 簡単な例題



動作理解のための説明用

all 1問題(改)

目的：前半5bitは0が多い方がよい、

後半5bitは1が多い方がよい

設計例（設定例）

適応度（評価）関数=(0の数in 1st 5bits)+(1の数in the 2nd 5bits)

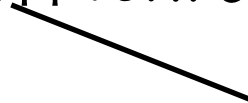
コード：長さ10bit、digital

染色体:chromosome

1001001101

遺伝子:gene

遺伝子座:locus



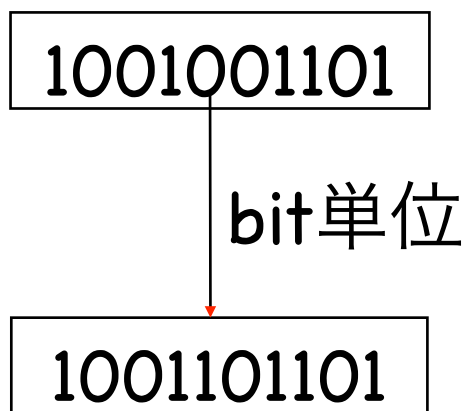


設計例（設定例） 2

遺伝的演算 1：選択（淘汰） エリート保存戦略

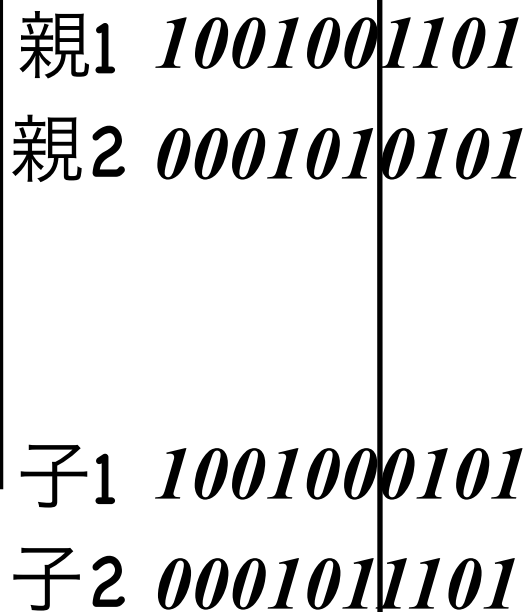
適応度の低い個体 ← 適応度が最大の個体

遺伝的演算 2：突然変異

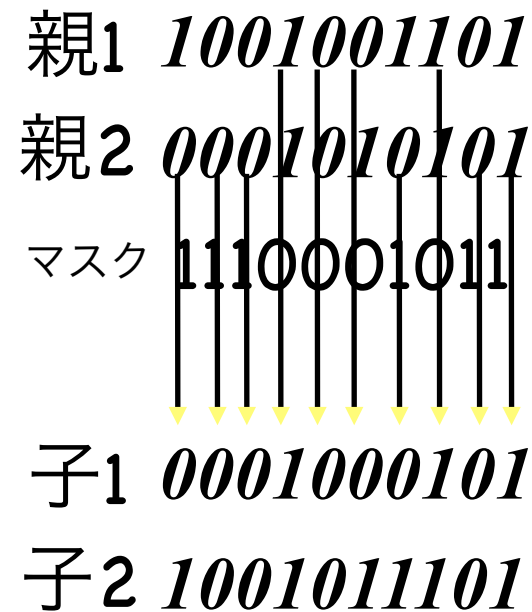


遺伝的演算 3：交叉

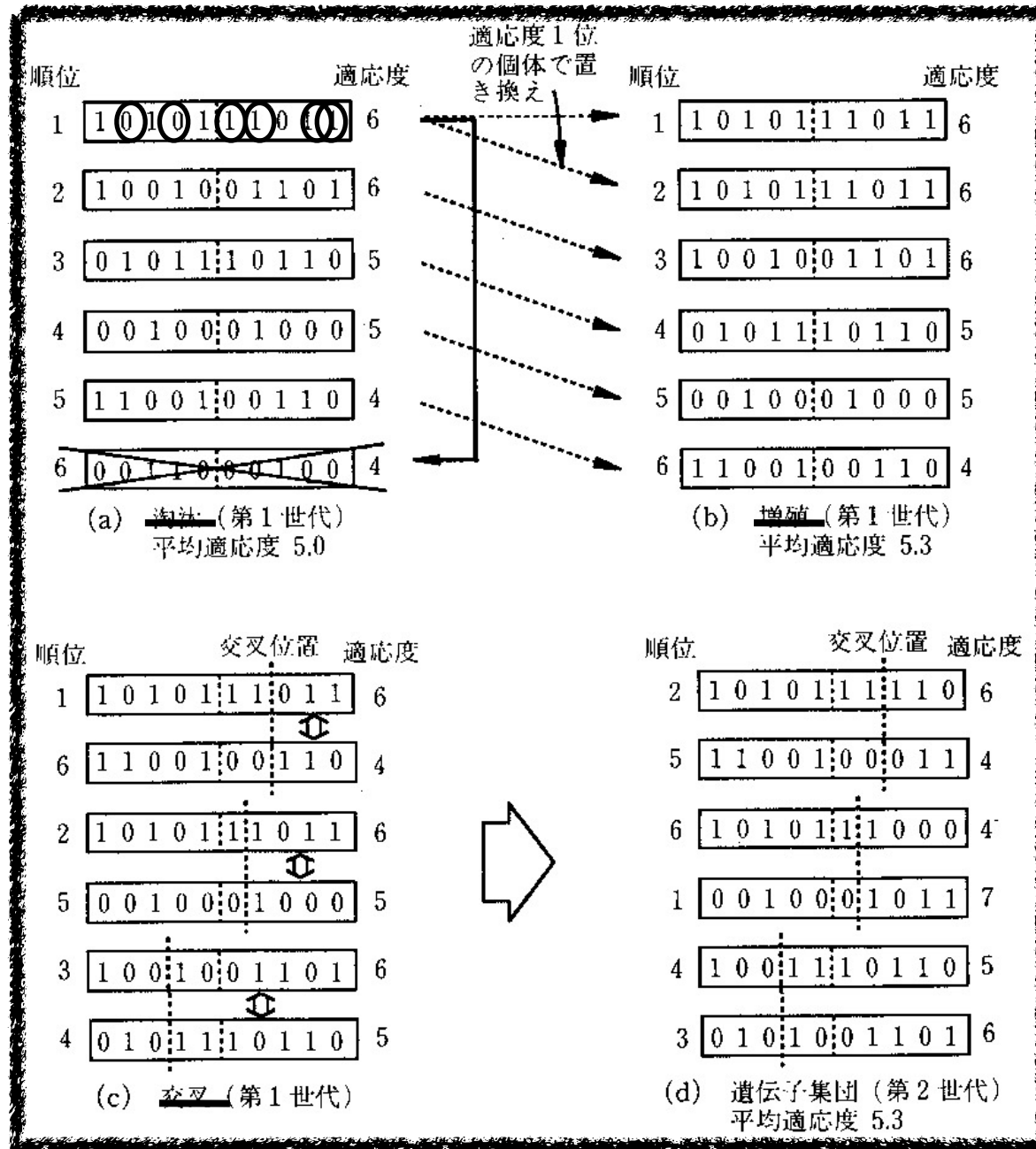
一点交叉



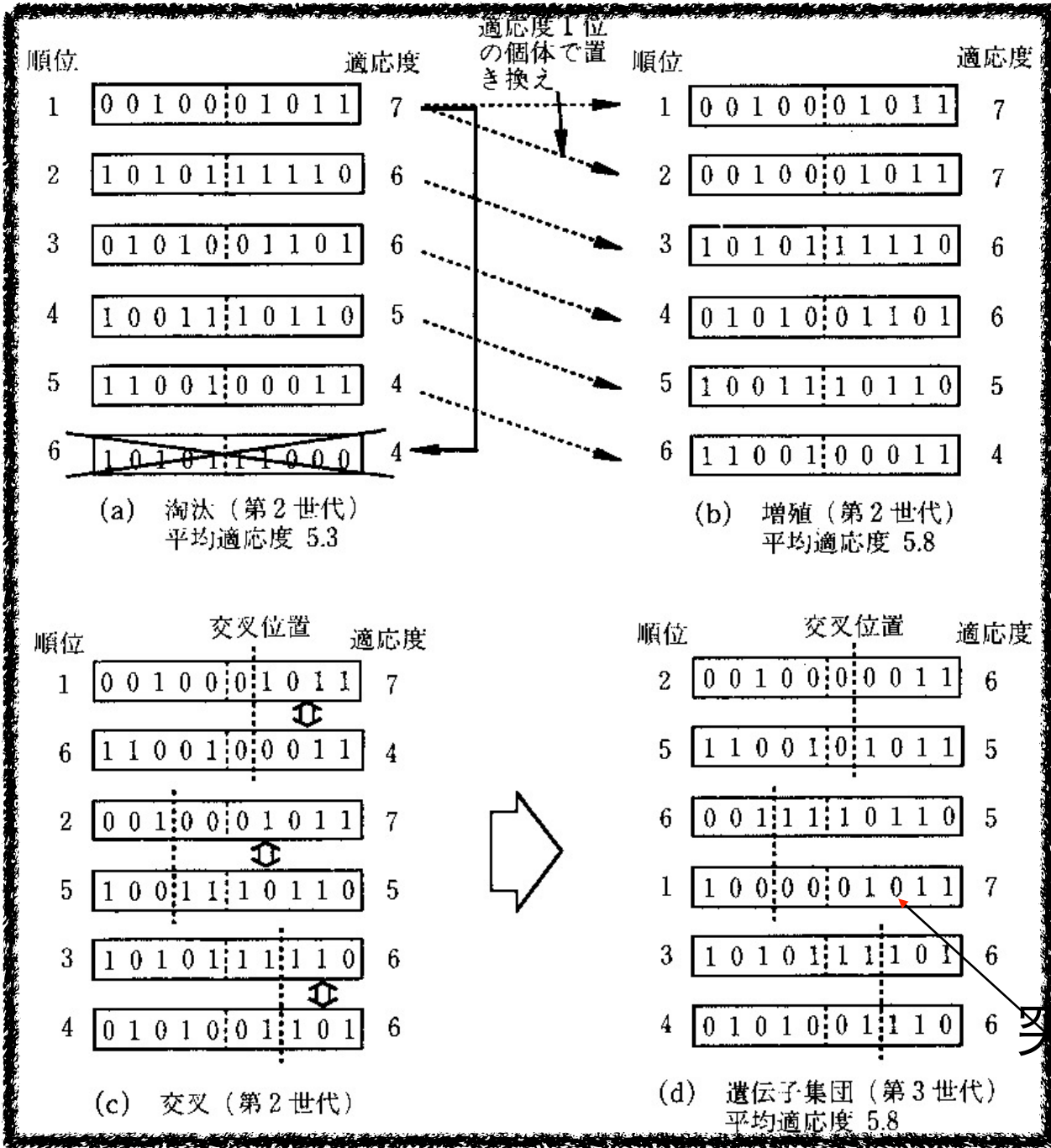
一様交叉



# 動作 1



# 動作 2



# 進化型計算が多く用いられる理由

1. 計算機の性能向上
2. 学習、適応、最適化を考える対象の拡大
3. 進化型計算の枠組みの柔軟さ
  - ◆ 比較的労力を必要とせず、準最適解を得ることができる

# 巡回セールスマン問題

## Traveling Salesman Problem: TSP

N個の都市を一度ずつ訪問する。

総距離を最小にするようなルートを求める。



- 全探索  $(N-1)!$  のオーダーの計算量が必要 → 必ず解が見つかる
- 厳密な最適解を都市数の多項式オーダーで求めるアルゴリズム  
→ 存在しない (と予測 (NP困難))

# コーディングの指針 (小林)

- 完備性(Completeness)

- 解候補をすべて染色体として表現できること

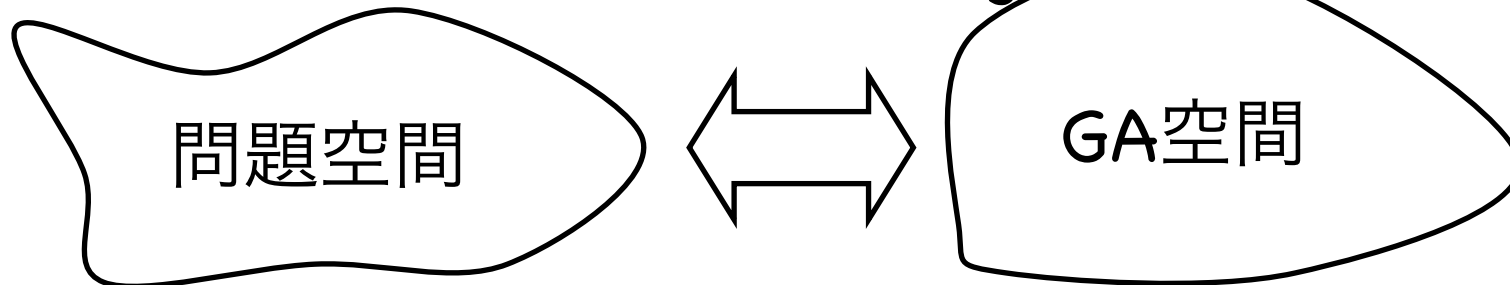
- 健全性(Soundness)

- 染色体はすべて解候補に対応付けられる

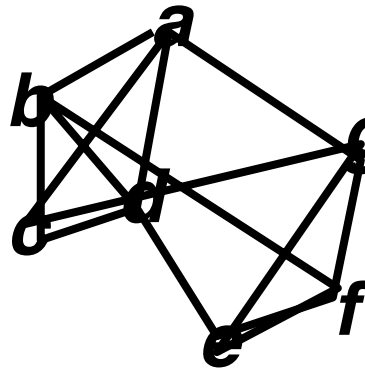
- 非冗長性(Non-redundancy)

- 染色体と解候補は1体1対応

- 形質遺伝性(Character Preservingness)



# コーディング + 遺伝的演算



a-b-c | d-e-f-g

a-c-g | e-f-b-d

| 交叉

a-b-c-e-f-b-d 全ての都市を1度ずつ訪問する

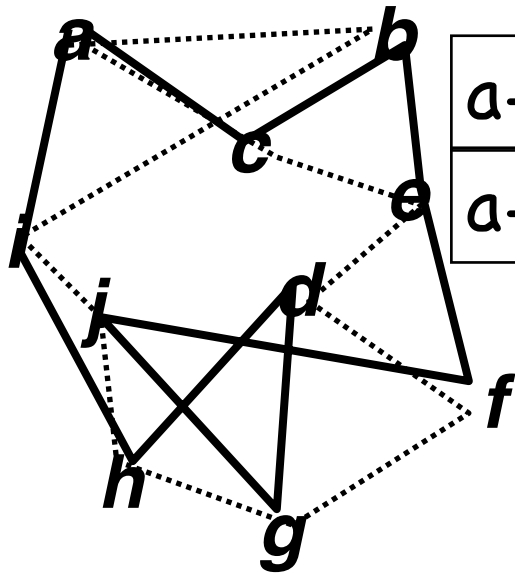
a-c-g-d-e-f-g という制約を満たしていない

致死遺伝子を持つ

— 評価値にペナルティ

| 解探索効率が悪い

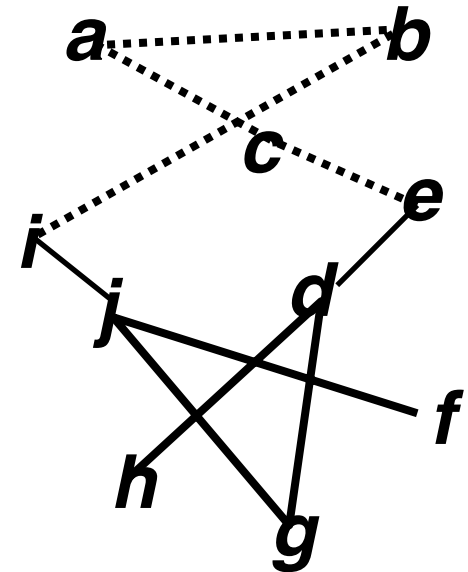
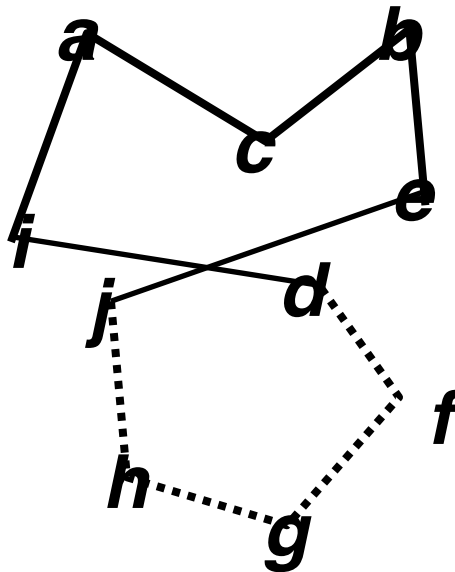
# コーディング+遺伝的演算の例 2



a-c-b-e-f-j-g-d-h-i
a-b-i-j-h-g-f-d-e-c

a-c-b-e-j-h-g-f-d-i
a-b-i-f-j-g-d-h-e-c

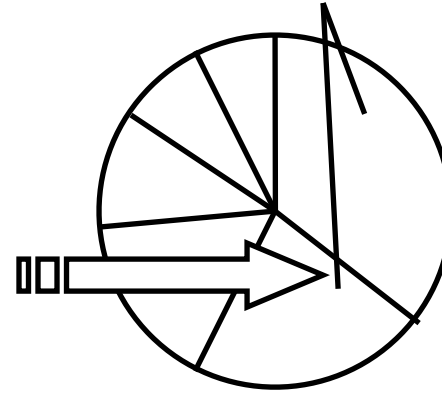
サブツアー交換交叉





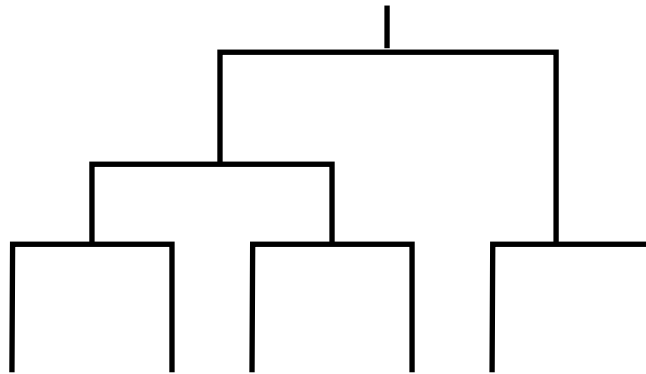
# 選択・淘汰 各個体の適応度

① ルーレット選択戦略



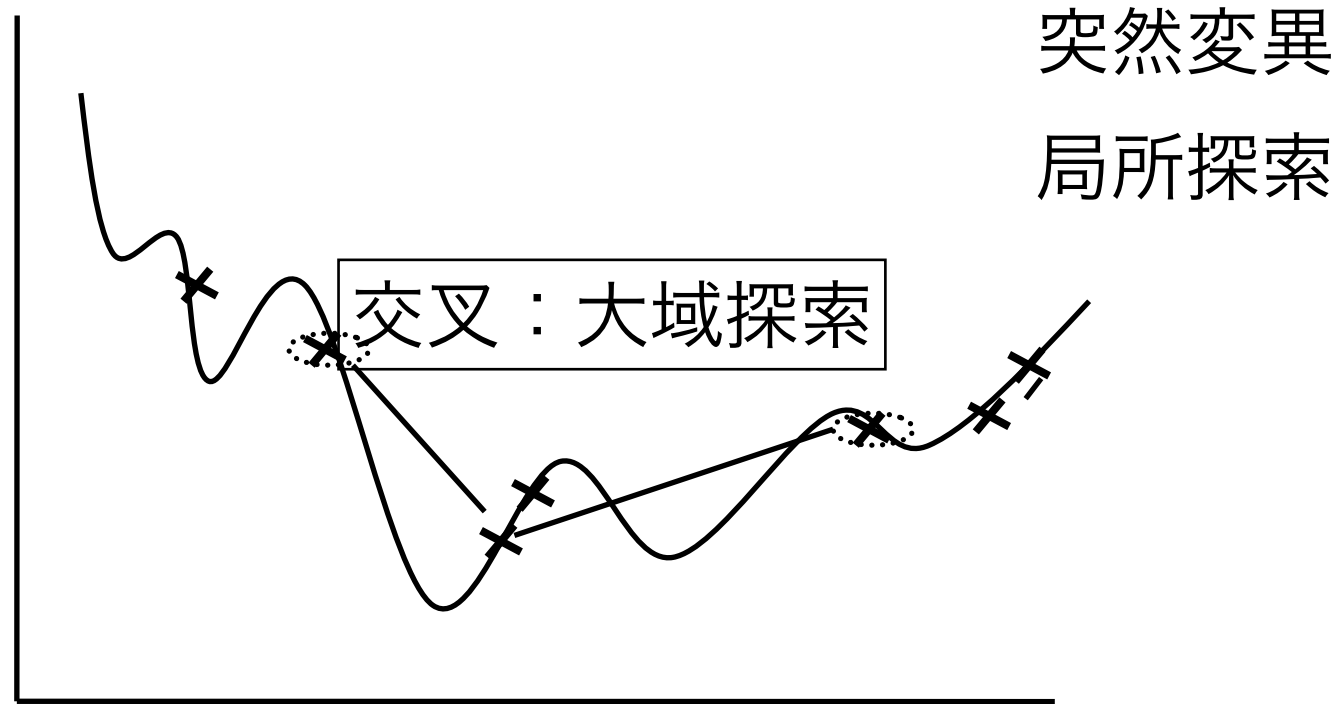
② エリート保存戦略

③ トーナメント選択戦略



# GAの特徴

- 多点（同時・並列）探索
- 多様性の維持



# 最適化手法としてのGA

- ④ 近似解を比較的容易に得ることができる
- ④ NP困難、多峰性などの問題にも適用可
- ④ Trial-and-Error形式の解探索
  - ④ 計算機の高速度により現実的に
- ④ 多様な要請に柔軟に適用可能
  - ④ 適応度（評価）関数の自由度大（計算できれば良い）
  - ④ 領域固有の知識を反映しやすい
  - ④ 領域固有の知識を前提としない