

# 組合せ最適化問題と遺伝的アルゴリズムによる解法

知能情報処理

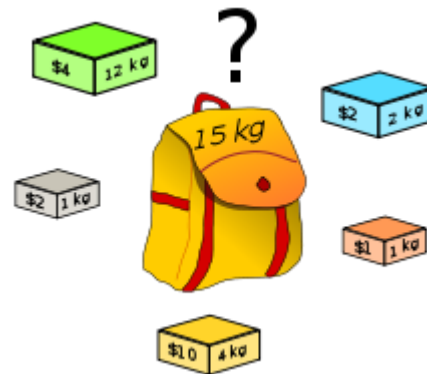
第2回

# 組合せ最適化問題の例(復習)

- 巡回セールスマン問題
  - 最短経路問題
  - 線形計画問題
  - エイトクイーン問題
  - ナップサック問題
- 
- 問題ごとに、制約条件などの形が異なるので、いろいろと工夫が必要
  - ここでは、ナップサック問題を取り上げる

# ナップサック問題

- ナップサックに入れるものの候補が準備されている
  - それぞれの候補のものの価格と重さ(あるいは大きさ)はわかっている
  - ナップサックの容量(重さあるいは大きさ)は制限がある
- トータルで最も価値が高いものの入れ方を求めよ。



# 解法を考える前に...

- 与えられるデータ
  - 入れるものの候補、価値、重さ

品物の番号 $i$	価値 $a[i]$	重さ $b[i]$
0	100	100
1	200	250
2	80	60
3	60	50
4	110	130
5	150	120

- ナップサックの容量  
 $c (= 500 \text{ とする})$

# 解くための準備

- 品物  $i$  を含めるとき  $x[i]=1$ , 含めないとき  $x[i]=0$  とする  
ような変数を用いる
- 解の候補は次のような配列で表現される

$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$
0	1	1	0	0	1

- 上記の場合に、
  - 制約条件を満たしているかどうか
  - 価値の総和
- を求めよ。

# 解を求めるには次のことが必要

- 解候補を(何らかの方法で)設定する。つまり、この場合、6要素の0,1の配列を求める。
- それぞれの6要素の配列に対して、
  - 制約条件を満たしているか  
どんな式？
  - 価値の総和  
どんな式？  
を求める
- 6要素の配列をいろいろと変えて、制約条件を満たしているものの中で最も価値の総和の大きなものを求める

# どうやって解候補を生成するか？

## いくつかの方法例

0	1	1	0	1	0
---	---	---	---	---	---

- 組合せを全部試してみても、最も価値が高い組を探すと→やめておきましょう
- よかったもの(価値の総和が高かった組)を少しだけ変えて、試してみる。もし変えたもののほうがよかったら、それを標準としてさらに、変更を加えていく(山登り法)。
  - 山登り法の弱点は？
- 出発点をいろいろと変えて、山登り法を行ってみる(多点探索)
  - 変数が多かったら、あまり探索できない

# 遺伝的アルゴリズム

- 我々の「個体」は、遺伝子からなる染色体で決まっている(DNA鑑定を知っているでしょう?)
- 我々生物の体の基本的な性質は、「生まれながらにして」決まっている。
- 進化とは?
  - 2人の親→子 (親の染色体の混合: 遺伝子型)
    - それによって決まる体の性質(表現型)により、「環境」に適合していれば繁殖し、適合していなければ淘汰(自然淘汰という)される(消滅する)→**交叉**
    - **突然変異**もたまにある(遺伝子が部分的に別のものに置き換わる)



0	1	1	0	1	0
---	---	---	---	---	---

- これを染色体と考え、与えられた問題を「環境」とみなし、問題に対してよい性質を持つ染色体は、「優れている」と考え、子孫を残す。
- この考えに基づく、最適化アルゴリズムを「遺伝的アルゴリズム」という。
  - 交叉
  - 突然変異
  - 選択
- を伴う

# 遺伝的アルゴリズム

①初期個体集合の生成

②復号化、適応度の計算

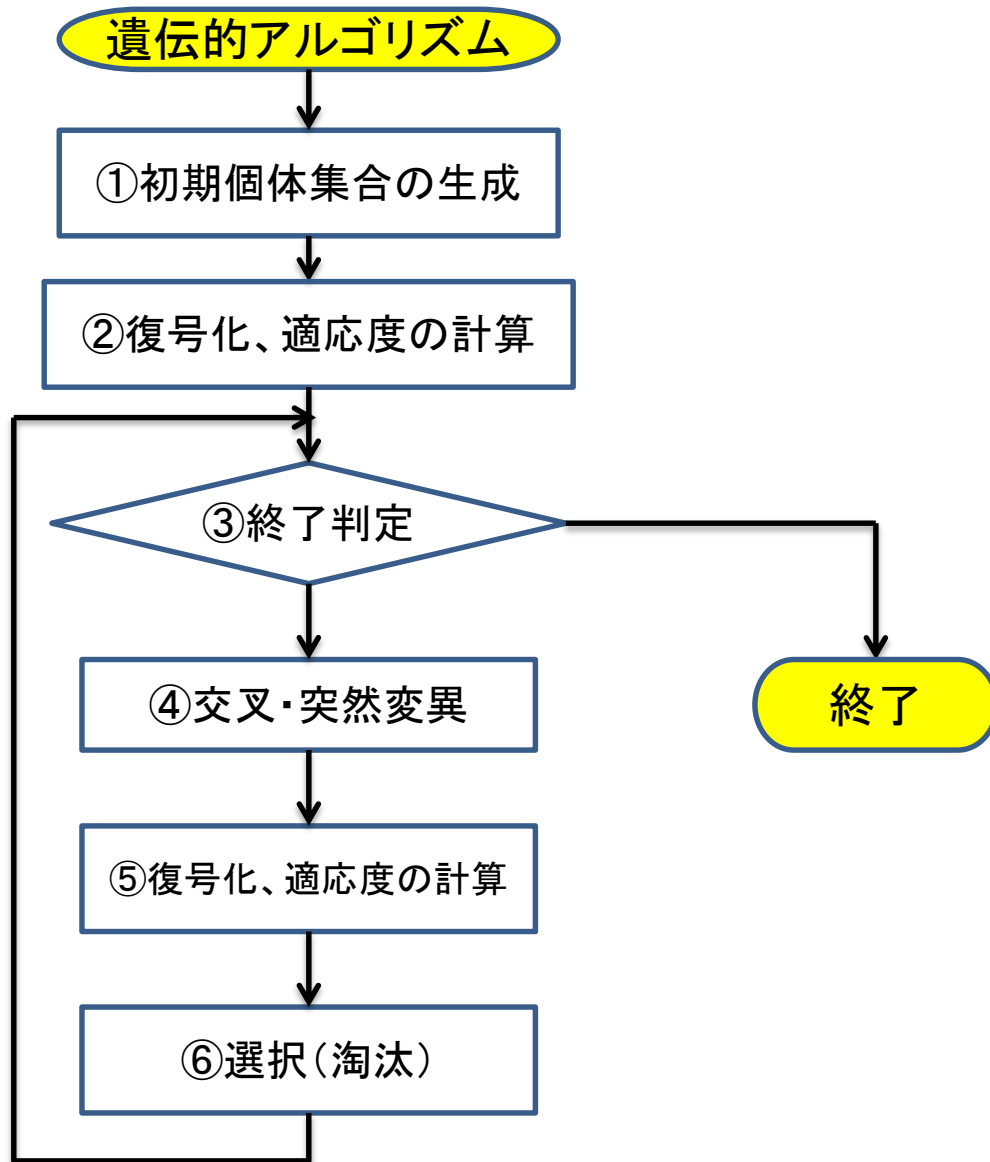
③終了判定

④交叉・突然変異

⑤復号化、適応度の計算

⑥選択(淘汰)

終了



# 初期個体の生成と適応度の定義

# 初期個体の生成と 適応度計算のプログラム

- プログラムソースファイルはHP上。
- 次の手順を踏んで実行せよ
  - Visual C++ Professional を起動
  - 新規作成→プロジェクト→Win32コンソールアプリケーション(名前をたとえばga1とする)→次へ→空のプロジェクトにチェックを入れる→完了
  - ga1 のフォルダの中にさらにga1フォルダがあるので、そこにprog1-1.cppを入れる
  - ソリューションエクスプローラでprog1-1.cppファイルを追加
  - ビルド
  - デバッグなしで開始で、実行してみよ。

# ナップサック問題

- データを定義
  - 入れるものの候補、価値、重さ

$a[6] = \{100, 200, 80, 60, 110, 150\};$

$b$ についても同じようにしてデータを与えよ。

品物の番号 $i$	価値 $a[i]$	重さ $b[i]$
0	100	100
1	200	250
2	80	60
3	60	50
4	110	130
5	150	120

# 課題

ナップサックの中身の総重量と総価値を各個体ごとに表示するプログラムを作れ。