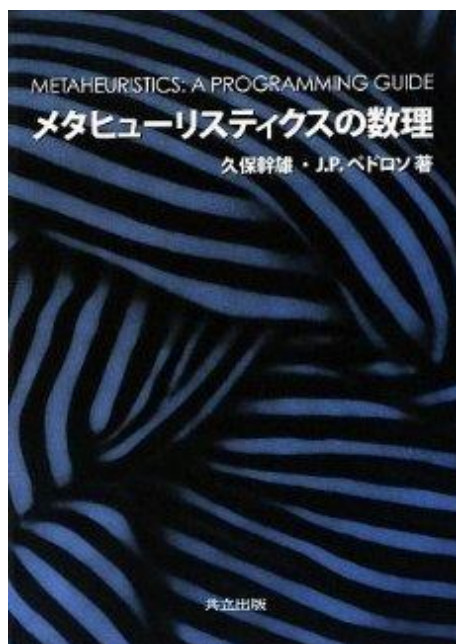


メタヒューリスティクスの数理

2. 代表的なメタ・ヒューリスティクス #3



2013.07.10

M1 若林由弥

内容

1. GRASP法(貪欲ランダム適応型探索法)

2. Ant Colony法(蟻群生法)

3. GA(遺伝的アルゴリズム)

4. 散布探索法

	構築法	改善法	
単一解		GRASP	
解集団	Ant Colony	散布探索法	GA

1

用語解説1

べき集合

ある有限集合 S の部分集合全体を表す集合. 2^S のように表す

ex) $S = \{a, b\}$ で表されるとき, $2^S = \{\phi, \{a\}, \{b\}, \{a, b\}\}$

半順序関係

以下の3つの条件が成立するような関係 \prec を半順序関係という

$$x \prec x$$

$$x \prec y \text{ かつ } y \prec x \text{ なら } x = y$$

$$x \prec y \text{ かつ } y \prec z \text{ なら } x \prec z$$

半順序関係の例

・集合の包括関係 \subseteq, \supseteq

このうち, 以下の条件が成り立つような関係を全順序関係という

$$x \prec y \text{ or } y \prec x \quad \forall x, y$$

全順序関係の例

・実数の不等式 \leq, \geq

2 用語解説2

半順序集合

半順序関係が定義されている集合

$$S_1 = \{\phi, \{1\}, \{2\}\} \quad \leftarrow \text{全順序集合}$$

$$S_2 = \{\phi, \{a\}, \{b\}, \{a, b\}\} \quad \leftarrow \text{半順序集合だが全順序集合でない}$$

※べき集合は半順序集合

Hasse図

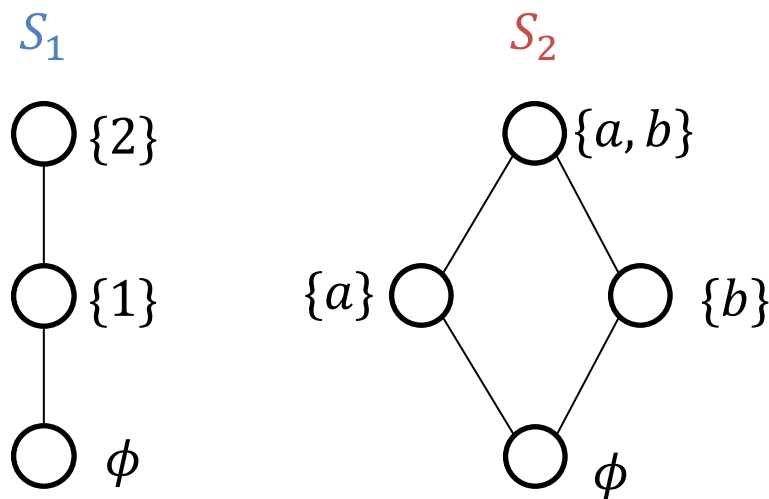
半順序集合を図示したもの



が成り立ち、かつ

$$x < z < y$$

となる z が存在しない時、枝 (x, y) を作る



3

用語解説3

独立システム

一般的な組み合わせ最適のフレームワーク

空でない有限集合 S に対して, S のべき集合 2^S の部分集合 F が

$$x \in F, y \subseteq x \Rightarrow y \in F$$

を満たすような F を独立システムという

最適問題が独立システム上にあるとは,

- ・解集合が, ある有限集合のべき集合の部分集合で表される
(べき集合を表すHasse図の一部で表される)
- ・1つの解の部分集合が必ず解集合に含まれる
(Hasse図の1つの解の下側全てが解に含まれる)

ナップサック問題は, 上の条件を満たすので独立システムで表現可能

4 例題1(ナップサック問題)

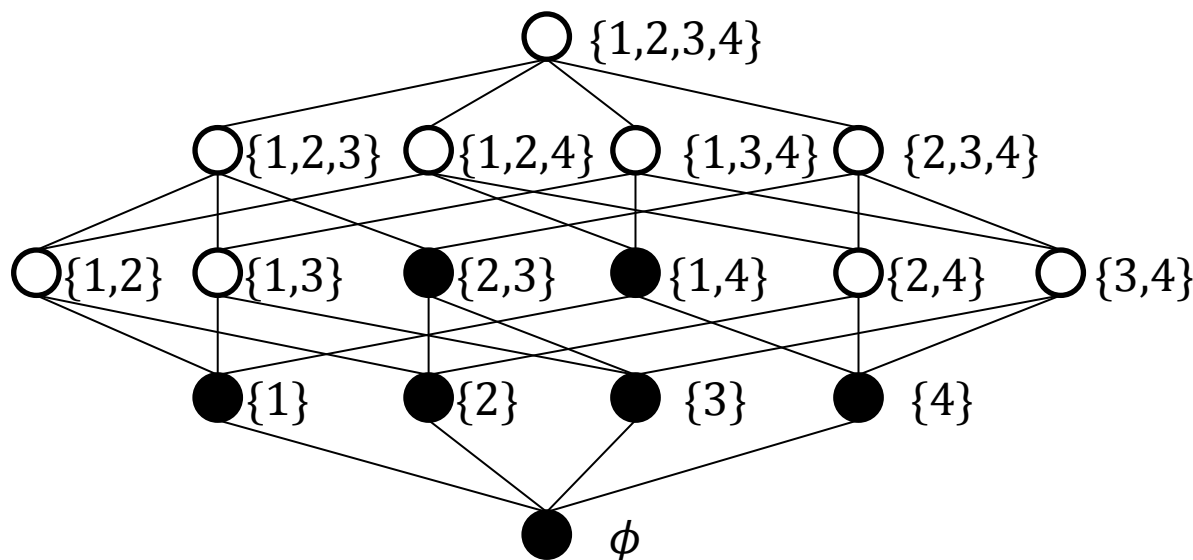
お祭りに来ました. 1000円で何をする?

		満足度	値段(円)
1.綿アメ		12	600
2.たこ焼き		9	500
3.金魚すくい		9	500
4.かき氷		5	400

この時, 予算内で一番満足
ができる組み合わせは,



満足度18, 値段1000円



解集合は左図のような
Hasse図の一部(独立シ
ステム)として表現可能

5

貪欲法(欲張り法)

貪欲法

問題をいくつかの部分問題に分割し、独立に評価する。そして評価の高いものから順に取り込んでいく手法。

		満足度	値段(円)
1.綿アメ		12	600
2.たこ焼き		9	500
3.金魚すくい		9	500
4.かき氷		5	400

評価値=満足度/値段

1

一番高いものを加える

0.9

次に高いが、1000円を超えてしまうのでス

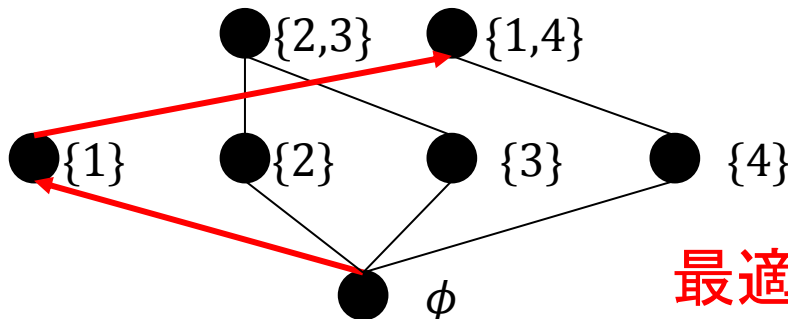
0.9

キップ

0.8

次に高いものを加える

貪欲法により得られた解



満足度17, 値段1000円

最適解ではない!

(初期値の選び方によっては最適解になる)

6

貪欲ランダム構築法

貪欲法では、初期値が同じなら1つの(最適でないかもしれない)解しか得ることができない

→加える解の候補にランダム性をもたせる

STEP1:解に加える候補の集合を生成

[生成ルール]

- ・評価値が最大値の $\alpha\%$ 以上のもの
- ・評価値が上から数えて β 番以内のもの

STEP2:候補から解に加える要素を選択

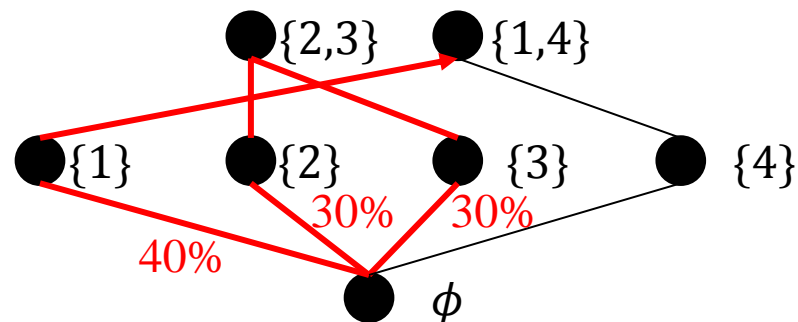
[選択ルール]

- ・順位ごとに決められた割合で選択
- ・評価値に比例した選択確率

STEP3:選択された候補を加える

加えたときに制約条件を満たすならそのまま加える
満たさないなら加えない

→ 選択された候補を候補集合から取り除き、再び選択を行う



7

GRASP(貪欲ランダム適応型探索法)

貪欲ランダム構築法を繰り返し適用することで良質な解が探索できる
→局所探索法と組み合わせることでさらに良い解を探索

STEP1:初期値の生成

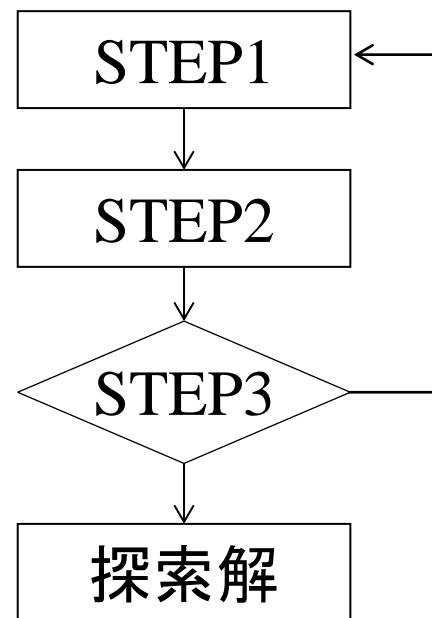
貪欲ランダム構築法により局所探索のための初期値を生成

STEP2:局所探索

STEP1で得られた解を初期値として
局所探索法で解を探索

STEP3:最適解の更新

STEP1～STEP2を終了条件を満たすまで繰り返し、
より良い解に更新していく

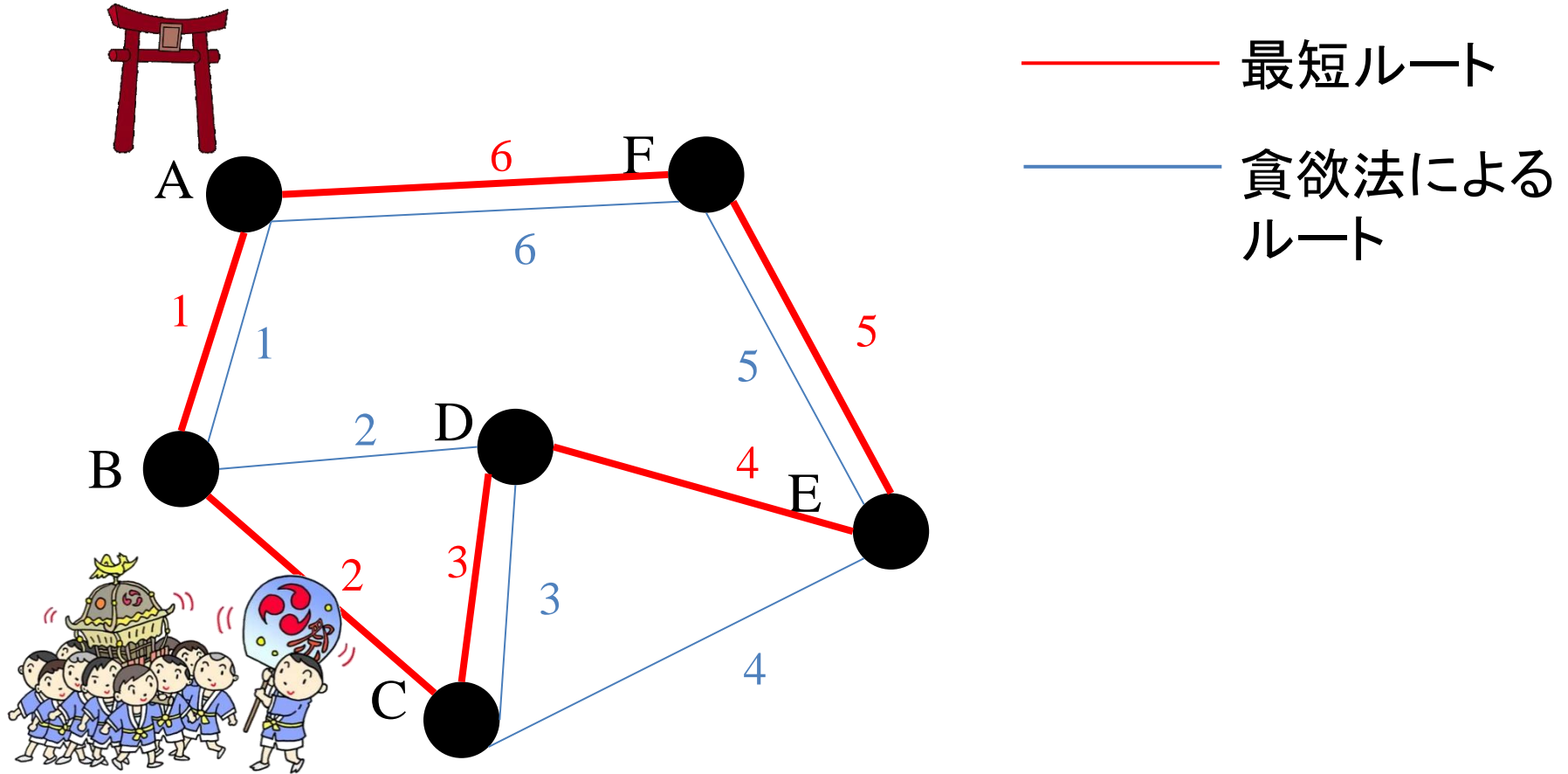


通常の貪欲ランダム構築法よりも収束が早くなる

8

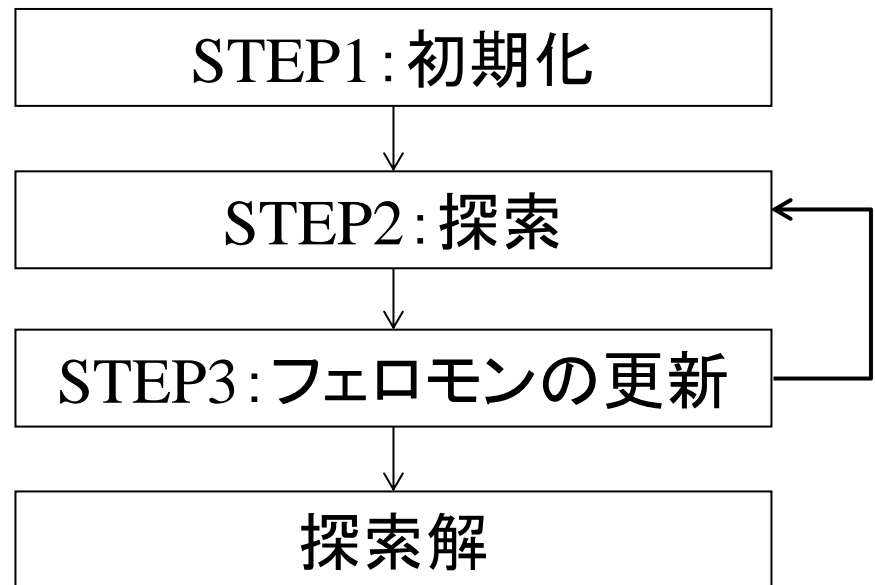
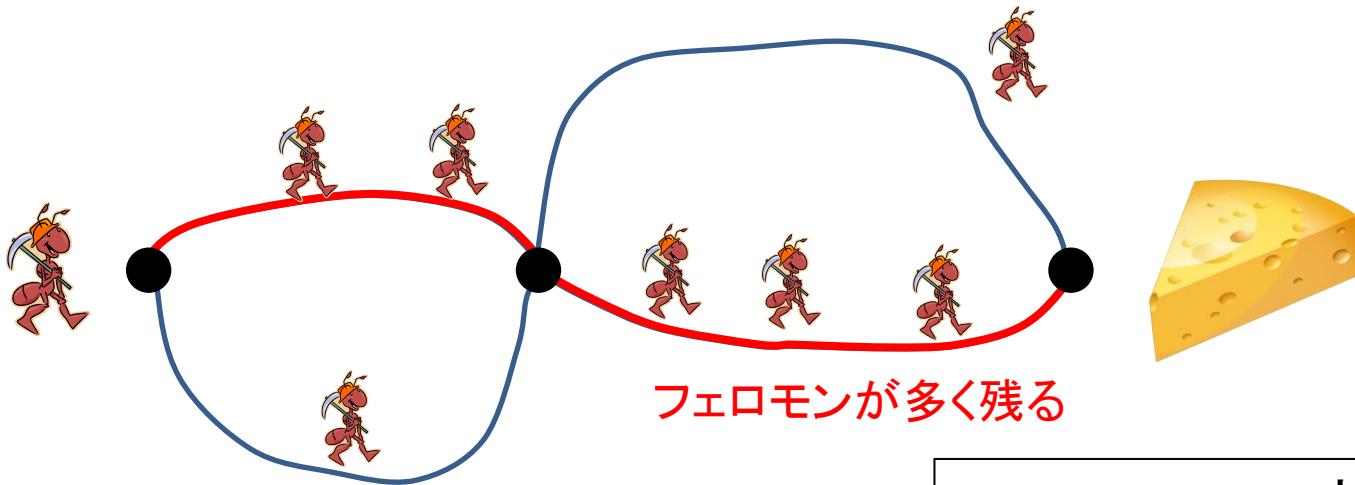
例題2(巡回セールスマン問題)

各ポイントを全部通る神輿の通る最短ルートは？



9 Ant Colony法(蟻群生法)

蟻が食糧を探索し，巣に戻るときにフェロモンをつけて戻る習性を模倣したアルゴリズム(構築型). 巡回セールスマン問題の探索に使われる



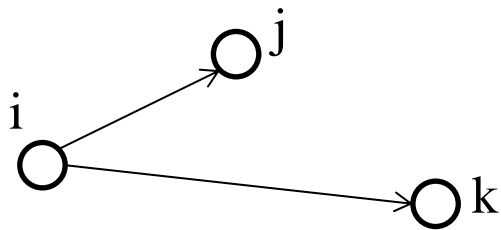
10 Ant Colony法(蟻群生法)

STEP1:初期化

初期集団のスタート地点をランダムに選択する

STEP2:探索

次に移動する地点を確率的に選択する



経路の効用 $V_{ij} = \frac{f_{ij}}{d_{ij}^\beta}$

経路の選択確率 $p_{ij} = \frac{V_{ij}}{V_{ij} + V_{ik}}$

d_{ij} : ij 間の距離 f_{ij} : ij に残ったフェロモン $1 \leq \beta$

STEP3:フェロモンの更新

前の期までのフェロモン → 割引率 ρ をかける

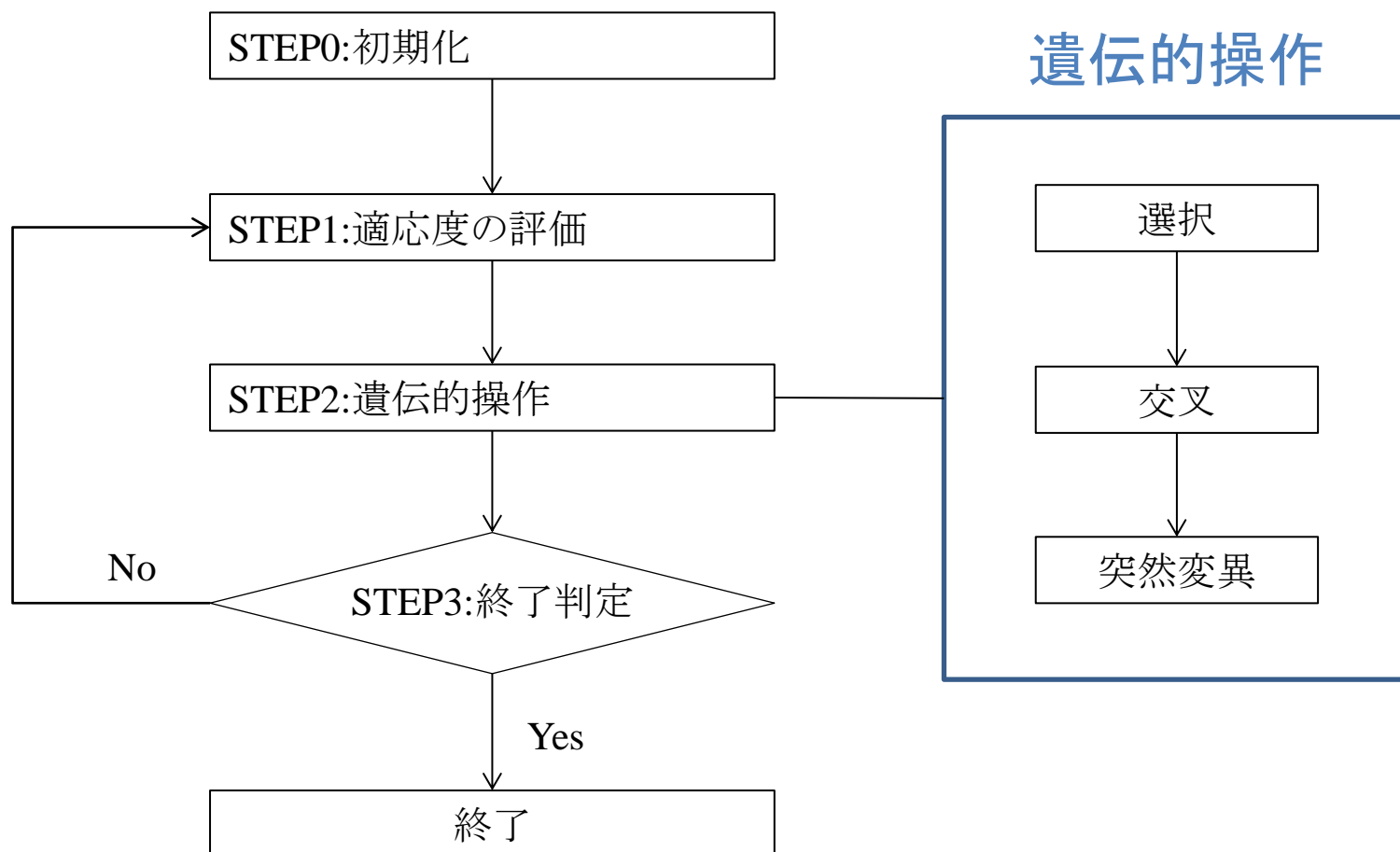
新しいフェロモン → 総経路長 L に応じて算出



足して新しい期のフェロモンを作成

11 遺伝的アルゴリズム(GA)

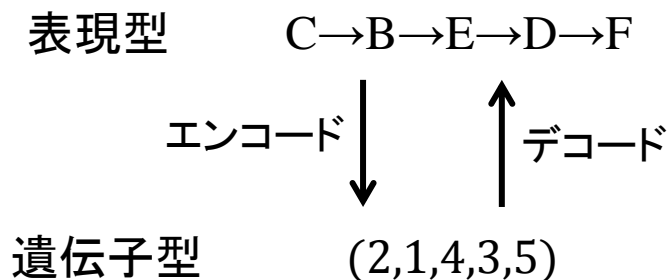
解候補の集団が、生物の進化のように変化し
最適解を探索するアルゴリズム(改善型)



12 遺伝的アルゴリズム(GA)

STEP0:初期化

初期集団を生成する. この時, 各個体を遺伝的操作を行えるような形(遺伝子型)にエンコードする



STEP1:適応度の評価

各個体の適応度を計算する.

最大化問題の場合 → 目的関数

最小化問題の場合 → -(目的関数), $1/(\text{目的関数})$

STEP2-1:選択

集団内で交叉に用いる(親になる)個体を集団から選択する

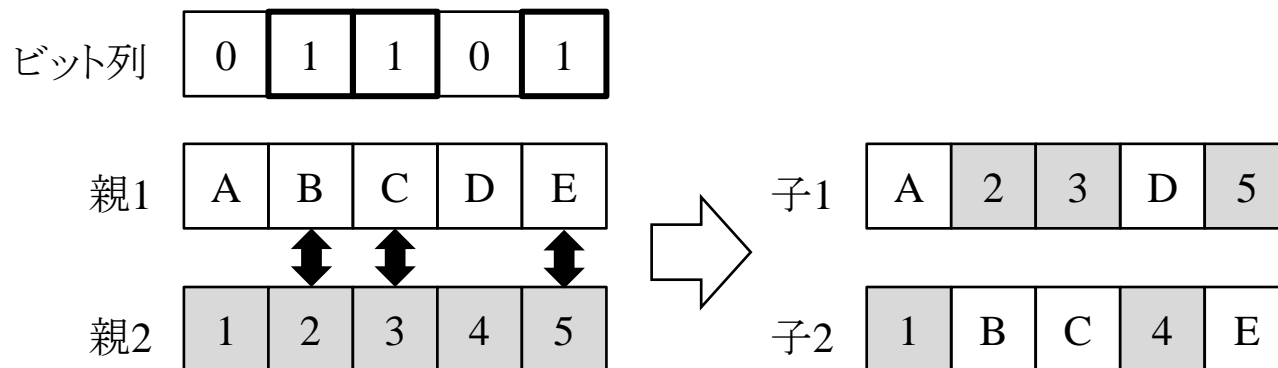
- ・ルーレット選択: 適応度に応じて個体の選択確率を比例配分する
- ・ランキング選択: 1位50%, 2位20%というように順位ごとに選択確率を決める
- ・**エリート選択**: 上位の個体をそのまま次の世代に残す. 残りの集団から他の選択方法で選択し交叉させる

13 遺伝的アルゴリズム(GA)

STEP2-2: 交叉

選択した個体を親として交叉させ、新しい個体を作る

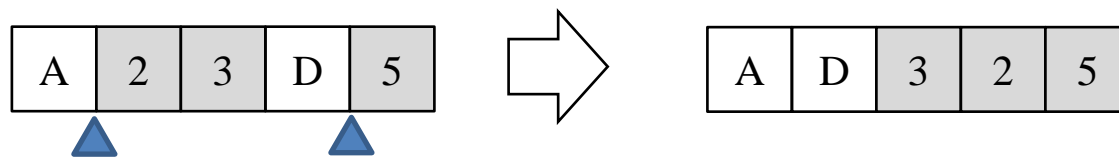
- ・一点交叉: 1つ場所を選んでその前後の遺伝子列を入れ替える
- ・二点交叉: 2つ場所を選んで2点間の遺伝子列を入れ替える
- ・**一様交叉**: 個体と同じ長さのビット列を作成し、1なら入れ替える



STEP2-3: 突然変異 ← 局所最適に陥らないための工夫

交叉で作成した個体をランダムに突然変異させる

- ・単一遺伝子突然変異: 個体の遺伝子を一つ、乱数と置き換える
- ・**転座**: ランダムに2点選択し、その間の遺伝子の順序を逆にする

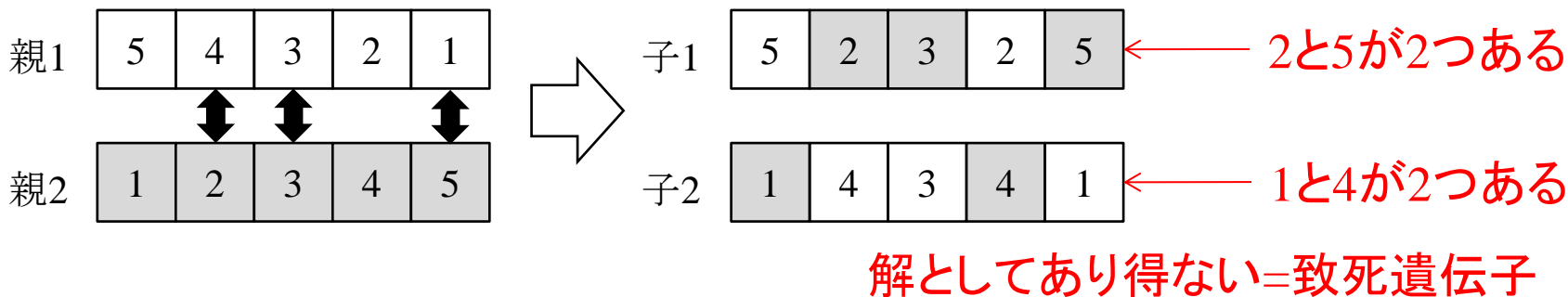


14 遺伝的アルゴリズム(GA)

制約条件は解の表現に含まれている

交叉, 突然変異によって制約条件を満たさない個体(致死遺伝子)ができる場合がある

例)順序を表す遺伝子型の場合



致死遺伝子ができないための工夫

- ・表現法・交叉の方法を工夫する
- ・致死遺伝子を実行可能解にするための操作を行う
- ・実行不可能性を最小化する問題として目的関数を追加する
- ・実行不可能性をペナルティとして目的関数に加える

15 散布探索法

タブーサーチの生みの親であるGloverが提案した手法.
多様性の保持に注目した手法

STEP1:初期化

初期集団を生成する.

※ランダムにではなく、長期メモリのような仕組みで多様な初期解を生成する

STEP2:近傍探索1

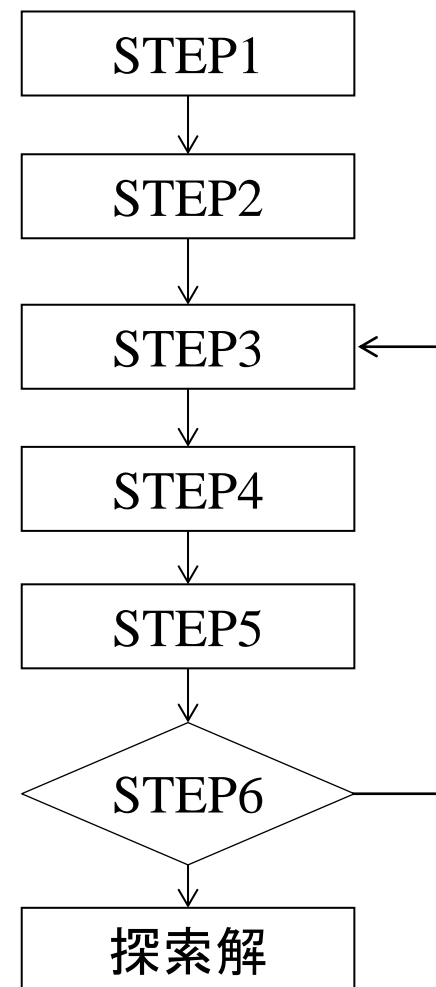
初期値の付近で最適解を探索する

STEP3:解集合の更新

はじめに適応度の高い個体をいくつか選んで選択集合に加える. その後, その集団からもっとも「遠い」個体を選んで加える

STEP4:選択

構造化合成のための親を解集合より選択する. 親は2よりも多い場合がある.

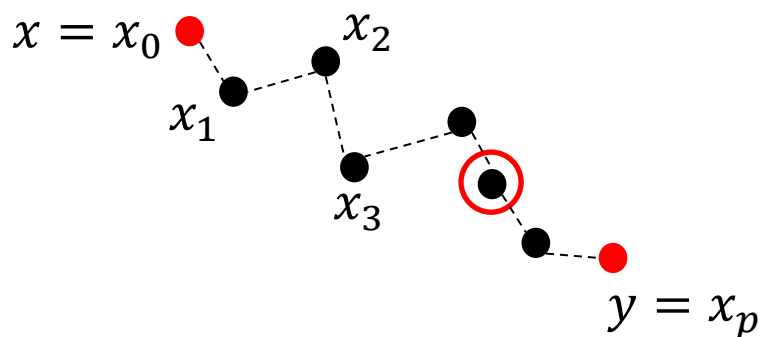


16 散布探索法

STEP5:構造化合成

遺伝的アルゴリズムでは致死遺伝子が発生する場合がある
→構造化合成では, 近傍を利用し, 実現可能性を保持する

構造化合成の例(パス結合法):



解 x と y の近傍パスの中から, 適応度が最大のものを新しい子とする

STEP6:近傍探索2

STEP5にて得られた子に対して近傍探索を行い, 解を改善する

→終了条件を満たしていれば終了, 満たしていなければSTEP3~5を繰り返す