



FLPTC :

$$\min. \sum_{i=1}^m \sum_{j=1}^n c_{ij} b_j x_{ij} + \sum_{i=1}^m \sum_{k=1}^{m_i} (f_i^k + g_i^k \sum_{j=1}^n b_j x_{ij}) y_i^k \quad \dots (1)$$

$$\text{s. t. } \sum_{j=1}^n b_j x_{ij} \leq \sum_{k=1}^{m_i} a_i^k y_i^k \quad i=1, \dots, m \quad \dots (2)$$

$$\sum_{i=1}^m x_{ij} = 1 \quad j=1, \dots, n \quad \dots (3)$$

$$\sum_{k=1}^{m_i} y_i^k \leq 1 \quad i=1, \dots, m \quad \dots (4)$$

$$x_{ij} \in \{0, 1\} \quad i=1, \dots, m ; j=1, \dots, n \quad \dots (5)$$

$$y_i^k \in \{0, 1\} \quad k=1, \dots, m_i ; i=1, \dots, m \quad \dots (6)$$

(1) 式は輸送費用, 建設費用, 操業費用の総和からなる目的関数の最小化. (2) 式は施設における供給能力の制約. (3) 式は輸送制約. (4) 式は多重選択制約. (5), (6) 式は決定変数を0または1に限る制約である.

### 3. 遺伝的アルゴリズムの適用

#### 3.1 遺伝的アルゴリズムの概要

まずFLPTCの解候補(実行可能性を問わない)を表す染色体を全く無作為に「複数個」用意する. それぞれの染色体には, 目的関数値を反映した値で, 次世代で生き残る染色体の期待値である適応度を与える. そしておのおのの染色体が「並列に」選択, 交叉, 突然変異によって集団的に探索を行い, それぞれに適応度のより高い点を目指そうとする. 探索の際, 先験的な知識や, 解領域特有の情報を用いず, 単に適応度のみを頼りに世代を進めていく. それに伴い, 適応度の高い染色体が集団の大多数を占めるようになり, 満足できる目的関数値を与える染色体を見いだす.

#### 3.2 染色体の決定

決定変数  $x, y$  を染色体として表す. 需要量によって降順ソートを行い, 上から需要地  $1, \dots, n$  と番号を付け直す. 輸送費もこれに対応して交換する. 染色体を構成する各遺伝子を  $A_1 \dots A_j \dots A_n$  ( $A_j \in \{1, \dots, m\}$ ) と表記することにより  $A_j$  の値は需要地  $j$  が供給を受ける建設候補地  $i$  に対応付けられ, 制約条件 (3), (5) を満たす. また, 輸送方法が決定されるため, 施設の供給量により

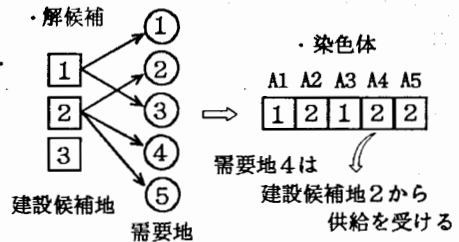


図2 解候補のコーディング

(4), (6) 式を満たす  $y_i^k$  を決定できる.

#### 3.3 遺伝的アルゴリズムの手順

STEP 0 : [初期集団の生成] 染色体を持つ個体を  $N_p$  (集団サイズ) 個発生させる. ここで  $N_p$  の2割の染色体について,  $A_j$  の値は需要地  $j$  から見て近い建設候補地  $i$  を選ばれ易くする. 残りの8割は, 全く無作為に  $A_j$  の値を発生させる.

STEP 1 : [適応度の計算] 次式で適応度を定義する. 個体  $u$  ( $u=1, \dots, N_p$ ) の実行可能性を問わない目的関数値を  $OBJ_u$ , 施設  $i$  での供給量を  $da_i$  とし, (2) 式を満たさない実行不可能解に対して正のペナルティ値  $\mu_1, \mu_2$  を与え, これを  $POBJ_u$  とすると,

$$POBJ_u = \begin{cases} OBJ_u & (\text{feasible}) & \dots (7) \\ OBJ_u + \sum_{i=1}^m \{ (da_i - a_i^{m_i}) * \mu_1 + \mu_2 \} & (\text{infeasible}) & \dots (8) \end{cases}$$

ここで、 $POBJ_u$  の最大値をMAX, プレ適応度を  $PFIT_u$  とすると,

$$PFIT_u = MAX * 2 - POBJ_u \quad u = 1, \dots, N_p \quad \dots (9)$$

すると、適応度  $FIT_u$  は,

$$FIT_u = N_p * PFIT_u / \sum_{i=1}^{N_p} PFIT_i \quad u = 1, \dots, N_p \quad \dots (10)$$

STEP 2: [ソート] 適応度の値によって、個体を降順ソートする。

STEP 3: [選択] 各染色体の適応度の違いに応じて選択確率が適切に配分されるように、適応度平均値 (FAVE) は変えず、適応度最大値 (FMAX) を FAVE の scal 倍 (scal は次世代に FMAX を持つ個体をどれだけ残そうとするかの期待値である) とする線形変換と、染色体の適応度の違いを顕著にさせるための指数変換を行う。集団内に同じ個体が複数ある場合、同じ個体の行き過ぎた復元抽出を抑制するため、重複個数を決められた割合 (選択圧力) に変換する。そして、 $N_p$  の 3 割は個々の個体の適応度の占める割合を、その個体を選択される確率として復元抽出する。 $N_p$  の 2 割は、ソートにより得られた上位の個体から選択する。選ばれた個体の中で適応度最大の個体は、突然変異の影響を受けずに残す。

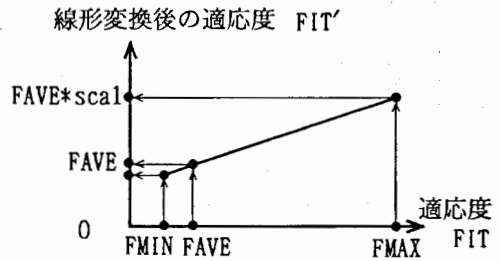


図3 線形変換

STEP 4: [交叉] 選択された個体を選択順に切断点を1箇所、または2箇所無作為に指定して、2つの個体間で染色体の組み替えを行い新しい個体を2つ生成する。これを6回繰り返すことで新しい12個の個体が生成され、この中で適応度の良い2つの個体と交叉前の2つの個体を残す。また、適応度最大の個体と2番、3番、4番目の個体に対しても同様の操作を行う。この時

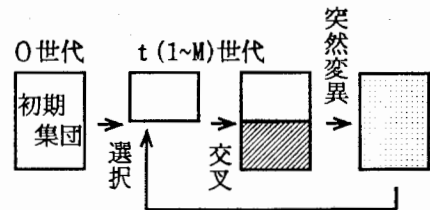


図4 GAの流れ

もし12個の個体の中の最大適応度が選択での適応度最大値よりも上回った場合、最大適応度を持つ個体は突然変異の影響を受けないとする。

STEP 5: [突然変異] 突然変異確率により遺伝子  $A_j$  の値を無作為に更新する。

STEP 6: [終了判定] 集団内での最良解が集団の過半数を占めるか、世代の上限  $M$  に達すると終了。さもなければSTEP 1へ。

#### 4. 数値実験

各建設候補地の選択可能数4, 建設候補地数10で、需要地数15である問題と、需要地数60である問題を、一様乱数によって10問ずつ作成した。各問に対してパラメータ  $M$  と  $N_p$  を変化させ、初期集団を5回ずつ替えて、数値実験を行った (表1, 表2)。表中の準最適解のバラツキは

$\frac{100(\text{optmax}-\text{optmin})}{\text{optmin}}$  であり、厳密解との差は  $\frac{100(\text{optave}-\text{optima})}{\text{optima}}$  であり、最良解との準最適解の評価は  $\frac{100(\text{optmin}-\text{minimum})}{\text{minimum}}$  である。ただし、 $\text{optmax}$ ,  $\text{optmin}$ ,  $\text{optave}$  はそれぞれ、各問の同じパラメ

一タごとの、準最適解の最大値、最小値、平均である。optima, minimumはそれぞれ、各問の厳密最適解、準最適解の最良値を表している。使用したプログラム言語はC言語、コンピュータはSPARC Classicである。表中の演算終了時間、最適解発見時間、厳密解到達率はそれぞれ10問5回、計50の平均、準最適解のバツキ、厳密解との差はそれぞれ10問の平均を表している。

表1 数値実験の結果(需要地数15)

世代数の上限と 集団サイズ	演算終了 時間(秒)	準最適解発見 時間(秒)	準最適解の バツキ(%)	厳密解との 差(%)	厳密解到達 率(%)
M=500, Np=100	38.1	13.0	1.924	0.681	64.0
M=500, Np=150	58.8	14.8	1.993	0.878	64.0
M=1000, Np=100	75.9	14.7	3.305	0.854	52.0
M=1000, Np=150	118.6	25.8	0.670	0.420	72.0

表2 数値実験の結果(需要地数60)

世代数の上限と 集団サイズ	演算終了 時間(秒)	準最適解発見 時間(秒)	準最適解の バツキ(%)	最良解との準最 適解の評価(%)
M=500, Np=300	352.0	233.7	3.360	0.879
M=1000, Np=100	222.6	148.5	4.184	1.321
M=1000, Np=150	341.1	226.5	2.935	1.326
M=3000, Np=100	653.7	439.5	2.610	0.863
M=3000, Np=150	1025.5	664.7	2.325	0.132

## 5. おわりに

本研究ではFLPTCに対してGAを適用することを提案した。GAで用いる染色体の構造を考案し、GAの適切な諸オペレータ、諸パラメータの設定を行った。FLPTCは、既往の研究では現実的な規模の問題を解くのは困難であったが、本研究では、現実的な規模の問題に対しても、実用可能な演算時間で解くことができ、準最適解であるが十分に有効な解が得られた。比較的規模の小さな問題に対しては、厳密最適解に達することができた。これは、3.2節で示した染色体の構造によって部分並列性が高くなり、交叉を有効に機能させることができたからである。また、初期集団の生成ルールも実行可能解を早く見つけることができることから有効であった。演算時間を早くするために集団サイズを小さくしてしまうと初期収束が起り、満足する目的関数値が得られないことがわかった。今後の課題としては、解候補のコーディング、有性生殖、並列処理などの検討を行い、準最適解の質の向上と演算時間をさらに短縮していくことが挙げられる。

### 【参考/引用文献】

- [1] Goldberg, D. E.: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.
- [2] 桧垣正浩, 西田直矩: 輸送制約が付加された問題, システム制御情報学会論文誌, Vol. 4, No. 4, pp. 155-162, 1991.
- [3] 小林重信: 遺伝的アルゴリズムの現状と課題, 計測と制御, Vol. 32, No. 1, pp. 2-9, 1993.