

遺伝的アルゴリズムによる 最適値探索とその高速化

吉原 郁夫[†] 仙石 浩明[†] 江端 智一[†] 坪井 潤[‡]
[†](株)日立製作所 システム開発研究所 [‡]日立中部ソフトウェア(株)

遺伝的アルゴリズムは、生物の進化過程を模擬した探索アルゴリズムであり、最適化問題の求解法としても有望視されている。遺伝的操作のうち交差および突然変異を解の逐次改善に積極的に利用し、解探索を速めることを試みる。また、大域的探索能力を高めるため、類似な染色体をもつ個体の数に制限を設ける方法を提案する。組合せ最適化問題、多峰性関数の最小値探索問題を題材に実験結果を報告する。

Fast Optimization Method using Genetic Algorithm

Ikuo YOSHIHARA[†] Hiroaki SENGOKU[†]
 Tomoichi EBATA[†] Jun TSUBOI[‡]

[†] Systems Development Laboratory, Hitachi, Ltd.,
[‡] Hitachi Chubu Software, Ltd.

The genetic algorithm, that models on natural evolution process, is expected one of the most powerful tools for solving optimization problems. The authors propose active crossover and mutation methods for fast convergence. They also propose how to limit the number of similar individuals for effective search of the global minimum. The methods are applied to two types of optimization problems.

1. はじめに

生物の進化過程を模擬した遺伝的アルゴリズム (Genetic Algorithm: GA) は、生物進化の研究、最適値探索の研究、免疫システムの研究など各方面からの注目を集めている。われわれの関心は最適化手法としての GA にある[6][7][8]。

GA が従来解法に比べ特徴的なことは、関数の勾配情報がなくても解けること、及び局所最適解から抜け出せることである。しかし、このような特徴を出そうとすれば必然的に検索が非効率になり時間がかかるという副作用が生じる。求解を速めるためには、アルゴリズムの改善、プログラム設計の改善などいろいろな面からの工夫が必要であるが、本報告では、前者の面からの試みを行なう。

別報[6][8]で取り上げた巡回セールスマン問題、波形照合の問題を題材に、交差、突然変異などの遺伝的操作を意図的に行う、高速化法を提案する。

2. 遺伝的アルゴリズム

はじめに GA の概要を簡単に説明しておこう。

まず、取り扱う問題に適した遺伝子表現を考えだす必要がある。これは通常のモデル化に相当する。解を特徴付ける記号列を考え、この記号列を生物の染色体(遺伝子列)とし、逆に染色体で特徴付けられる一つの解(実は解の候補)を一匹の生物とする。記号列は文字通り記号であってもよいし、数値やベクトルであってもよい。

<初期化>

GA では、はじめに上記のように定義された染色体を多数用意し、それをこの生物の祖先とする。初期の染色体は乱数を用いて適当に設定する。

<世代更新>

親から子を作り、さらにその子(孫)を作る……という操作を繰り返し進化、発展させる。一世代は次の4ステップから成る。

step 1 (交差)

個体群の中からランダムに2個体ずつ選びそれぞれを親個体とする。両親の染色体の部分列をとり、新たな染色体を生成し子個体を作る。

step 2 (適応度評価)

各個体の染色体情報をもとに個体の評価値を求め、適応度を算出する。適応度は、各個体の集団に於ける生き残り易さを表わす指標である。

step 3 (淘汰・増殖)

step 2 で求めた適応度をもとに、適応度の低い個体を淘汰し、高い個体を増殖させ、優れた個体の多い集団に改善する。

step 4 (突然変異)

ランダムに選んだ個体に対し、染色体の一部を強制的に変更する。こうすること

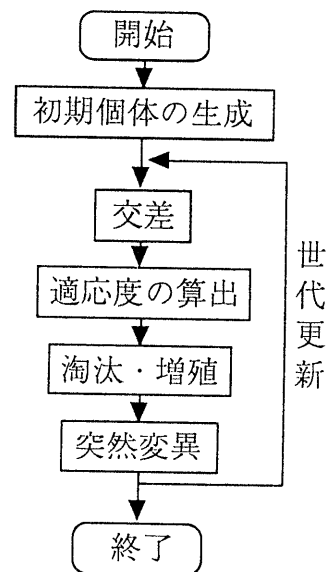


図1 GAの手順

により、親個体の持っていなかった形質を発現させることができる。

交差、淘汰、増殖、突然変異などの遺伝的操作の具体的方法は多種多様であり、また、これら操作の処理順序もさまざまである。詳細は文献[1][3]等を参照されたい。

3. 巡回セールスマン問題

組合せ問題の例として、巡回セールスマン問題(Traveling Salesman Problem: TSP)を取り上げる。

3.1 従来法の問題点と改善の方針

GAによるTSPの解法としては、'85年 Grefenstetteの報告が有名である[2]。同氏の遺伝子コーディング方法 "ordinal representation"(順序表現)は、交差(巡回経路の組替えに相当する)しても致死遺伝子が決して生じない点で優れている。しかし、この方法は交差箇所より後方の遺伝子(染色体の後半部)は、親個体の形質のごく一部しか継承しないという欠点がある[4][5]。

我々の改良のポイントは二つある。一つは、親個体の形質を継承しやすい交差を行うことである。この点に関しては、いくつかの報告もあるが、本報ではさらに交差を行うさい計算負荷が少くなるよう工夫する。他の一つは、突然変異の積極的利用である。突然変異には、局所最適解からの脱出と、解の改良という機能がある。われわれは特に後者の機能に着目し、突然変異が解の改良に有効であった場合のみ採用するという戦略をとることにより、探索を加速する。

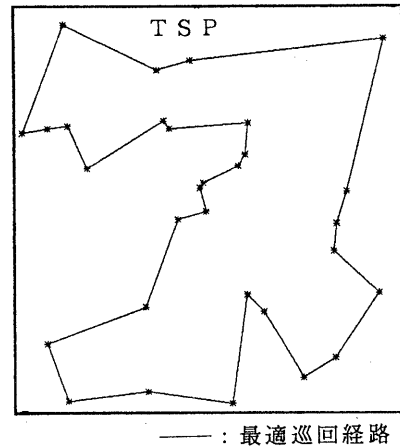


図2 30都市の配置

3.2 提案アルゴリズム

問題とする都市の集合が $C = \{c_1, c_2, \dots, c_n\}$ 、2都市間のコストが $d_{ij} = d(c_i, c_j)$ と与えられている。都市の巡回経路をそのまま染色体(遺伝子列)とする。これを経路表現と呼ぶ。ある巡回経路 $P = c_{i_1} c_{i_2} \dots c_{i_n}$ に対応する個体 X の染色体を $g_x = (x_1, x_2, \dots, x_n)$ 、ただし $x_k = c_{i_k}$ (for all k) とする。

3.2.1 交差の戦略

二つの親個体 a 、 b の染色体を $g_a = (a_1, a_2, \dots, a_n)$ 、 $g_b = (b_1, b_2, \dots, b_n)$ とする。まず、都市を一つランダムに選び、これを c とする。 a 、 b それぞれの中で都市 c の現れる位置を i 番目、 j 番目とする。都市 c を起点とし、隣接する都市を重複なく選べる限り、都市名を接続して行く。このアルゴリズムをまとめると以下の通りである。

```

<初期化>
  Find i, j;
  k ← i, ℓ ← j;
  Sa ← true, Sb ← true;
  g = φ;

<反復>
  do
    if(Sa and ak ∉ g) then g ← {ak · g}, k ← k-1;
                        else Sa ← false;
    if(Sb and bℓ ∉ g) then g ← {g · bℓ}, ℓ ← ℓ+1;
                        else Sb ← false;
  until (¬Sa and ¬Sb)

```

ここで、 S_a 、 S_b は都市を選べるか否かを示す状態変数である。「 \cdot 」は接続演算子であり、 $g \leftarrow \{a_k \cdot g\}$ は a_k を g の前に繋ぐことを表す。また、 $k = k(\text{mod } n)$ 等とする。

このようにして作られた染色体 g の長さ(選ばれた都市数)は、一般に親個体のそれよりも短い。 g の長さが足りない部分は、残された都市をランダムに選んで埋める。 g の長さがある基準値以上のときは、この交差を採用し、未たない場合採用しない戦略をとる。

提案法の交差と、従来法(Grefenstette流)の交差を比較評価する。実験に用いるTSPの都市配置を図2に示す。同図の巡回経路は最適解である。都市数は30あり、コストは都市間の距離とする。全体フローやシミュレーションパラメータの詳細は、文献[6]を参照されたい。

或る親個体から、提案法の交差で作った子個体の評価値(コスト)と、従来法(Grefenstette流)によるそれを求め、それぞれを図3のグラフ上の点(x, y)に対応させプロットする。親個体は、初期状態(0~10世代)及び十分進化した世代(900~910世代)から選んだが、ほとんどの場合、点は直線 $y = x$ より上側にあり、提案法による子個体の評価値の方が良い(コストが低い)ことがわかる。

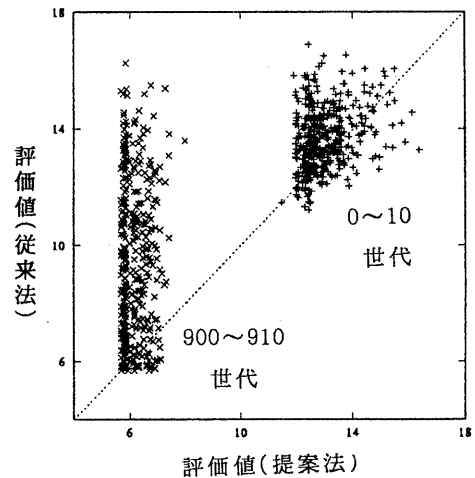


図3 交差方法の性能比較

3.2.2 突然変異の戦略

突然変異は染色体のある区間を反転する方法をとる。この実際的意味は、「ランダムに2都市を選び両都市間に含まれる都市の巡回経路を逆転させる」ことである。例えば、選ばれた都市が c_{i_s} 、 c_{i_t} であれば巡回経路 $P = c_{i_1} c_{i_2} \dots c_{i_n}$ は $P' = \dots c_{i_{s-1}} \underline{c_{i_t} c_{i_s}} c_{i_{t+1}} \dots c_{i_n}$ に変わる。下線部が逆転させた区間である。

この逆転によりコストが下がるか否かを次の判別式で評価する。

$$D(s,t) = d(c_{i_{s-1}}, c_i) + d(c_{i_s}, c_{i_{t+1}}) - (d(c_{i_{s-1}}, c_s) + d(c_{i_t}, c_{i_{t+1}}))$$

$D < 0$ はコスト減を意味する。 $D < 0$ ならばこの突然変異を採用し、そうでなければ採用しない。この方法は突然変異の直後に淘汰を行うことに相当し、以下これを「瞬時淘汰」と呼ぶ。

こうすることにより、個体評価(コスト計算)を省略できるだけでなく、全個体にわたる適応度評価も省略でき、計算負荷が軽減できる。さらに、同一個体に対し突然変異を何度もくり返し施す。このやり方は、ヒューリスティックな反復改良法を導入したのと等価である。

瞬時淘汰の有効性を示すために、突然変異を無条件に採用する従来法(普通のGA)との比較実験を行う。突然変異率を0%から50%まで変化させて、ある水準の解(最適解の3割増し)が得られるまでに要した世代数を測る。20回実験を繰り返して所要世代数の平均をとったグラフを図4に示す。突然変異率を上げていくと、従来法では、求解に要する世代数がいったん下がったあと再び増加に転じるのに対し、提案法では所要世代数が単調に減少することがわかる。

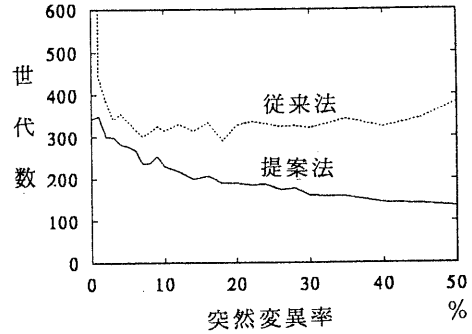


図4 瞬時淘汰の効果

3.2.3 大域的探索の戦略

GAでは、淘汰・増殖の結果、ほとんど同じ遺伝子をもつ個体が集団の大半を占め、これが局所最適解からの脱出を妨げることがある。上記のような交差、突然変異は効率的である反面、局所最適解へ落ち込みやすくしている恐れがある。これを回避するため、染色体の大部分が同じ個体(類似個体)が複数ある場合は一個体のみ残すという戦略をとる。

30都市問題の一つを固定し、1000世代打ち切りとし従来法(Grefenstette流)と提案方法で、各20回ずつ解き比較する[6]。真の最適解は4.756であるが、従来法では最良解8.219、平均8.677を得、提案法では最良解4.756、平均4.757を得た。提案法では20回中18回真の最適解を発見できたが、従来法では0回であった。

4. 時系列データに含まれる波形の照合

多峰性関数の最適値探索問題の一例として、波形照合を取り上げる[8]。ここで云う波形照合とは、例えば図5のように、システムの状態監視時に時々刻々入ってくる観測データ $x(t)$ の中に、予め与えられたある波形(基準パターン $s(t)$) が現われるか否かを判別することである。以下、図5に示すテスト問題を例に説明する。

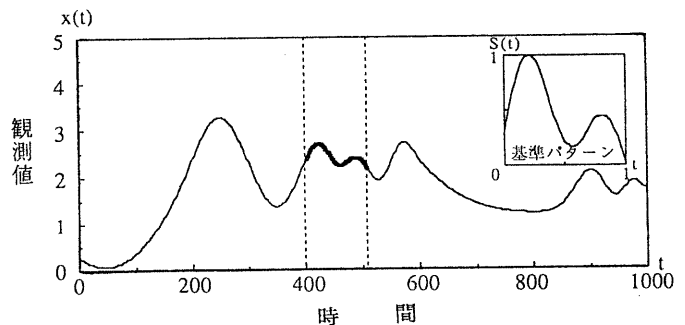


図5 波形照合のテストデータ

4.1 問題の定式化

基準パターン $s(t)$ は、単位正方形に納まるように規格化しておく。また、観測データ $x(t)$ は等間隔の時系列データとして与えられるとする。波形の一致・不一致の判別には、観測データの切り出し部分をいろいろ変え、基準パターンとのずれの大きさが最小となる区間を見出す必要がある。 $x(t)$ のうち区間 $T = [T_1, T_2]$ に含まれる部分と $s(t)$ との一致の度合は、 $x(t)$ を単位正方形に納まるよう変換し、 $s(t)$ に重ね合わせたときのずれ面積の大きさと考え、それを自乗誤差の型で評価する。

$$F(T_1, T_2) = \left[\int_0^1 |x'(t) - s(t)|^2 dt \right]^{1/2} \quad (2)$$

$$\text{ただし, } x'(t) = a(x((T_2 - T_1)t + T_1) - b) \quad (3)$$

$$a = \frac{1}{\frac{\max\{x(t)\} - \min\{x(t)\}}{t \in T}} \quad (4)$$

$$b = \min\{x(t)\} \quad (5)$$

$x'(t)$ は、 $x(t)$ がちょうど単位正方形に納まるよう縦横異なる倍率で変換した関数である。以下では、特に断らない限り「ずれ面積の大きさ」と言えば式(2)で定義される F を指すものとする。 F は、 T_1, T_2 に依存する関数であり、一般には図6に示すように、多峰性(多谷性?)である。ただし、図6は切り出し区間幅に制限 ($\alpha \leq T_2 - T_1 \leq \beta$) を設け、かつ F の形状を見やすくするため上下反転して表示している。

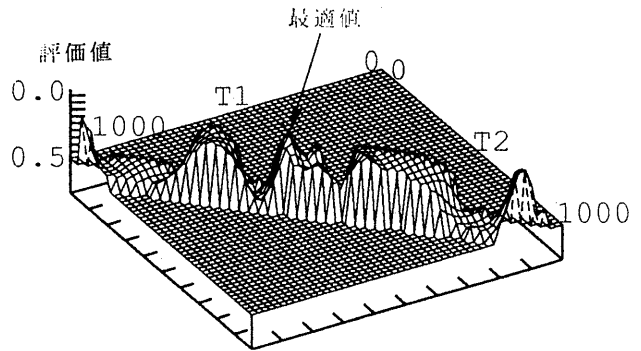


図6 評価関数 F の形状

F の最小値探索に GA を用いる。切り出し区間の始点 T_1 、終点 T_2 の座標をそれぞれ 2 進表現のビット列に対応させ、それらを染色体とする。即ち、一つの個体は二本の染色体をもつ。

4.2 提案アルゴリズム

求解手順は、図1に示したものと基本的には同じである[8]。また、交差、突然変異に第3章と同様の考えで工夫を施し、高速化を図る。

4.2.1 交差の戦略

ランダムに2個体を選び、 T_1 を表す染色体同士、または T_2 を表す染色体同士を組合せ、

一点交差を行う。どちらの染色体にするかはランダムに決める。片方の染色体のみ操作する理由は、両方を同時に操作するよりも致死遺伝子が生じにくいと考えるからである。その検証実験を以下に示す。

二つの染色体のうち一方だけ交差させる方式(一染色体交差)と、両方交差させる方式(二染色体交差)との差異を調べてみる。毎世代の個体の中から30個体を選びそれを元に870組(=30×29)の親を作り、一染色体交差、二染色体交差を行ない、生存率(致死しない割合)を調べた。図7に示すように、世代が進むといくぶん差は縮まるが、一染色体交差の方が生存率は高い。また、いずれの方式の方が評価値の低い子(優れた子個体)が生まれやすいかを別途調べたが、ほとんど差はない。一染色体交差の方が処理が簡単という実利もあり、こちらが有利である。

4.2.2 突然変異の戦略

ランダムに個体を選び、いずれかの染色体に強制的に変更を加える。片方の染色体のみ操作する理由は、交差の場合と同じである。

ふつうの考えでは、突然変異はランダムな位置に生じるものであり、本問題の場合一様乱数の値を加えることと等価である。しかし、ここでは現在値に頂点を持ち、 T_1, T_2 の変域を底辺とする三角形の分布に従う乱数を加える。こうする理由は突然変異が微調整の役割を果たす確率を高めるため、及び致死による無駄を省くためである。

通常の突然変異と提案方式の突然変異の比較実験を行う。ただし、通常の突然変異を行い致死した場合は、初期化と同じ方法でランダムに一個体を発生させ、個体数を維持するものとする。いくつかの試行の中から、典型例を3つ上げて説明する。図8の上側は、通常の突然変異を行った場合であり、図の横軸は世代数、縦軸はその世代における最優良個体の評価値である。評価値(ずれ面積)が0.05に達した場合は収束とみなして打ち切っている。3ケースとも大差はない。

同じ初期個体群から出発し、提案方式の突然変異を施した結果を同図下側に示す。case1、case2は遥かに収束が速くなっている。これは、初期個体の中に最適解に比較的近い所にいるものが含まれていたからであり、提案方式は、解の微調整に向いていることが窺える。しかし、case3のようにかえって悪くなる場合も時々ある。これは、提案方式がランダムサーチ的機能をいくぶん犠牲にしているためと考えられるが、その対策については次項で説明する。

4.2.3 大域的探索の戦略

TSPの場合と同様、上記のような、交差、突然変異戦略をとると局所最適解へ落ち込みやすくなる。ここでも、類似個体の数に制限を設ける対策が有効である。

個体の切り出し区間が互いに90%以上重複している場合、それらをお互いに類似個体と

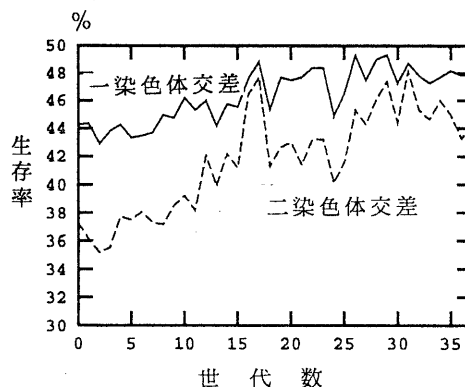


図7 子個体の生存率比較

考える。最優良個体の近傍の個体数を全個体の50%までしか許さないとし、許容値を越えた分は抹殺し、同数ランダムに発生させる。何もしない場合は、50回の試行のうち37回しか解が求まらないが、提案方式では50回すべて解が求まっており導入効果は顕著である。

5. おわりに

遺伝的アルゴリズムで、最適化問題を解く際の収束を速める方法を検討した。

交差及び突然変異による解の逐次改善効果に着目し、これらを意図的に操作し、交差、突然変異の直後に「瞬時淘汰」を行なうことにより、解の探索効率が高まることを実験的に示した。また、これらの工夫の副作用として局所解への落ち込みが危惧されるが、それを避ける方法として、類似の染色体をもつ個体数に制限を設ける方法が有効であることを示した。

数値計算の観点からは、さらにプログラミングに密着した部分での改良や並列処理も興味ある課題である。

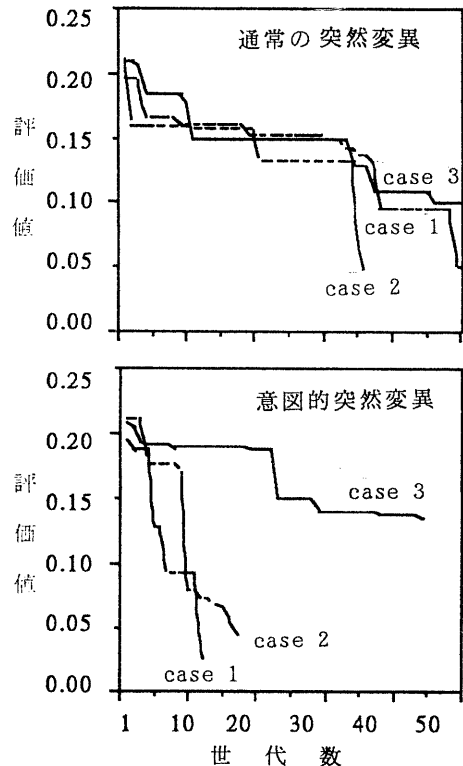


図8 意図的突然変異の効果

<参考文献>

- [1] Goldberg D.E.: Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, (1989)
- [2] Grefenstette, J. J., et al.: "Genetic Algorithms for the Traveling Salesman Problem", Proc. 1st ICGA(1985)
- [3] 星野: 「遺伝的アルゴリズムの基礎と課題」、遺伝的アルゴリズムに関するシンポジウム、(1992.11)
- [4] 山村、小野、小林: 「形質の遺伝を重視した遺伝的アルゴリズムに基づく巡回セールスマン問題の解法」、人工知能学会誌 Vol.7 No.6(1992)
- [5] Goldberg, D.E. and Lingle, R., Jr: "Alleles, Loci and the Traveling Salesman Problem", Proc. 1st ICGA(1985)
- [6] 仙石、吉原: 「遺伝的アルゴリズムによるTSPの高速解法」、情処第46回全大2-305(1993)
- [7] 江端、吉原: 「遺伝的アルゴリズムによる個体評価法に関する一提案」、情処第46回全大2-281(1993)
- [8] 吉原、江端: 「遺伝的アルゴリズムによる波形照合」、情処第46回全大2-283(1993)