

SGA における GA パラメータの効果比較と 自動チューニング方法に関する考察

Sangheon Han*

1 はじめに

遺伝的アルゴリズム (Genetic Algorithms : GA) においては、交叉率 (Crossover rate)、突然変異率 (Mutation rate) など各種パラメータのチューニングはもちろん、選択や交叉、突然変異のアルゴリズムそのものも重要な研究対象となる。そこで、今回は交叉と突然変異の効果に着目し、Simple Genetic Algorithm (SGA) における複数のパラメータの値をテストモデルにそれぞれ適用し、それらのパラメータの有効性を検証した。シミュレーションに対する適切なパラメータの決め方については、実験計画方法¹を用いるなど様々な研究が行われた。しかしながら、問題によってはそのパラメータの設定だけに膨大な時間やコストが必要となり、実用的なレベルでは多少困難な状況である。本研究では、GA の効率的なパラメータの決め方に対して、自動的かつ柔軟にパラメータを変える新しいパラメータ・チューニングメソッドについて考察をおこなう。

2 遺伝的アルゴリズム (Genetic Algorithm)

遺伝的アルゴリズム (Genetic Algorithms : GA) は、生物が環境に適応して進化していく過程を科学的に模倣した学習型アルゴリズムの一つである。GA の研究は 1960 年代後半から 1970 年のはじめに Michigan 大学の J.H.Holland^[4]らによって始められ、その研究成果は 1975 年に「Adaptation in Natural and Artificial System」という題で出版されている。また、1989 年に出版された Goldberg の「Genetic Algorithms in Search, Optimization, and Machine Learning」以降、日本でも数多くの研究実績があり、現在では生物学を含めて、工学、経営学、経済学など様々な分野に応用されている。自然界における生物の進化過程においては、ある世代を形成している個体の集合、すなわち母集団の中で、環境により適応した個体がより高い確率で生き残り、次の世代に優秀な因子を残す。このメカニズムをモデル化し、環境に対して最もよく適応した個体 (Individual)、すなわち目的関数に対して最適値を与えるような解を計算機上で求めようというのが GA の仕組みである。GA では、個体は設計数の値がコーディングされた染色体 (Chromosome) と呼ばれる文字列として表現され、この染色体をデコーディングすることにより設計変数を読み出し、目的関数の値を計算する。このとき、染色体の構造のことを遺伝子型 (GenoType)、これによって定まる個体の形質を表現型 (PhenoType) と呼ぶ。また、個体の集団のことを母集団 (Population)

* Faculty of Management Information Science, Nagoya University of Commerce & Business, Komenoki, Nishin, Aichi 560-0043, Japan

¹ 効率のよい実験方法をデザインし、結果を適切に解析することを目的とする統計学の応用手法である。R.A. フィッシャーが 1920 年代に農学試験から着想して発展させた。実験計画法を用いた GA のパラメータ分析については、HAN (2001) を参照されたい。

と呼ぶ。GA はこの母集団に対して選択 (Selection)、交叉 (Crossover)、突然変異 (Mutation) などの遺伝的操作を繰り返し行うことによって解を求めて行く。

Genetic Algorithm

- Step1. 初期集団の生成
- Step2. 適用度による選択／淘汰 (評価)
- Step3. 交叉
- Step4. 突然変異
- Step5. 次世代集団生成 (Loop to step2)

図 2-1 GA の処理手順

- **Initialization (母集団の初期化)**

この操作は、あらかじめ設定された数だけランダムに個体を生成するものである。生成した個体の数のことを母集団サイズ (Population Size) や単に個体数と呼び、ここで生成した個体の集団を初期母集団とする。

- **Evaluation (評価)**

この操作は、各個体の持つ染色体を問題空間にデコードして解を求めるものである。ここで求めた解のことを評価値 (Evaluation Value) という。

- **Selection (選択)**

この操作は、生物の適者生存を模倣したものである。この操作では、まず各個体の評価値から次世代への生き残りやすさを求め、これに基づいて次世代の母集団を形成する。この次世代への生き残りやすさのことを適合度 (Fitness) と呼ぶ。

- **Crossover (交叉)**

この操作は、生物の有性生殖を模倣したものである。この操作により、個体間で染色体情報が交換される。最適解を表す個体の一部分を持った個体同士が交叉すれば、より最適解に近い個体が得られる可能性が高くなる。個体集団のうち何割の個体が交叉するかを交叉率 (Crossover Rate) と呼ばれるパラメータによって定める。

- **Mutation (突然変異)**

染色体は遺伝子 (Gene) を格納する複数の遺伝子座 (Locus) から構成され、遺伝子座に入りうる遺伝子のことを対立遺伝子という。突然変異とは、染色体上の遺伝子座の遺伝子を別の対立遺伝子に置き換える操作のことであり、自然界における DNA 複写の際に起こるコピーミスにあたる。染色体のうち何割の遺伝子座が変異するかを突然変異率 (Mutation Rate) と呼ばれるパラメータによって定める。

- **Terminate Condition (終了判定)**

この操作は、あらかじめ定められた終了条件に基づいて GA を終了させるためのものである。

3 Simple GA における各種パラメータ

3-1. 母集団サイズ (Populations Size)

GA は多点探索であり、その探索点の数のことを母集団サイズ (Population Size) と呼ぶ。探索点の数が多いと広い範囲を探索することが可能となるために最適解を発見しやすくなる。ただし、多すぎると収束が遅くなるために余計な評価計算回数が増える。逆に、探索点が少ないと収束が速くなるために局所探索性能が向上する。しかし、少なすぎると広い範囲を探索することができずに最適解ではない局所解に落ち込む可能性が高くなる。つまり、個体数は対象問題の性質や選択手法との関係が深いパラメータであると考えられる。また、母集団のサイズは、大きいほど精度が高いが計算の速度など効率が悪くなるため、適切なサイズが望ましい。また、様々な研究からその効果について検証が行われている。

3-2. 評価 (Evaluation)

評価とは、個体 (Individual) の持つ染色体 (Chromosome) をデコーディングして設計変数を読み出し、目的関数の値を計算する操作である。GA におけるコーディングの方法として、本研究では、問題空間の状態を 0 または 1 で表現された遺伝子によって表すコーディングを採用する。このコーディングには、バイナリコーディングとグレイコーディングの2つがある。バイナリコーディングは単に整数を2進法によって表現したものであり、グレイコーディングは、連続する2つの符号間のハミング距離が常に1であるという特徴を持っているものである。これによりグレイコーディングはバイナリコーディングに比べて、特に終盤での解探索の効率が良くなるという報告がなされている^[6]。しかし、本研究ではテスト関数の特徴上バイナリコーディング法を採用した。

3-3. 選択手法 (Selection Method)

- **Roulette 選択**

この手法は、個体群の中の各個体の適合度とその総計を求めて、適合度の総計に占める各個体の適合度の割合を選択確率として個体を選択するというものである。

- **エリート戦略**

個体群の中で最も適応度の高い個体を、無条件でそのまま次世代に残す手法である。個体群の中で最も適応度の良い個体を、次世代にそのまま残すエリート戦略が考え出された。エリート戦略を採用すると、その時点で最良の個体は交叉や突然変異によって破壊されることはないという利点がある。その反面、エリート個体の遺伝子が個体群の中に急速に広がる可能性が高いので、局所的な解に陥る危険性も含んでいる。

- **Tournament 選択**

この手法は、個体群の中からあらかじめ定められた定数 (トーナメントサイズ : Tournament Size) 分の個数の個体をランダムに選び出し、その中で最も適合度の高い個体を次世代に残すという手続きを次世代に残したい数の個体を選択されるまで繰り返すというものの^[9]であり、トーナメントサイズを大きくすれば適合度の低い個体が淘汰される確率が高くなる。トーナメントに選び出す個体の重複を許さない場合、トーナメントサイズを2に設定した Tournament 選択は、ranking Roulette 選択に近似できる^[9]。

• Roulette Tournament 選択

この手法は、上記の Roulette 選択と Tournament 選択を組み合わせたものである。具体的には、トーナメントサイズ分の個体の抽出方法としてルーレットを用いるものである。

選択の方法に対しては、解の精度はそれほど大きく変わらず、問題の特徴やコーディングの形にあわせて選ぶことが多い。

3-4. 交叉

• 一点交叉 (One-Point Crossover)

この手法は、ランダムに1つの交叉点 (Crossover Point) を定め、その点を境目に染色体を交換するものである。

• 二点交叉 (Two-Point Crossover)

この手法は、ランダムに2つの交叉点を定め、その間にある染色体を交換するものである。

• 一様交叉 (Uniform Crossover)

この手法は1989年にG.Syswerdaによって提案された手法で、一点交叉、二点交叉と異なり、交叉操作の度に交叉点の数が増える。交叉の度にランダムにマスクパターンを生成し、それに従って交叉点を作成するというものである。

3-4-1. 交叉率

母集団のうち、どのぐらいの割合の個体が交叉に参加するかを定めるのが交叉率 (Crossover Rate) である。母集団内に存在する個体が偶数の場合、交叉率1ですべての個体が交叉に参加することになる。

3-5. 突然変異

GAにおける突然変異は、染色体上のある遺伝子座 (Locus) の遺伝子 (Gene) を対立遺伝子 (Allele) に置き換えることによって起こる。通常、単に突然変異と呼ばれるものである。この手法では、あらかじめ定めた突然変異率に従ってすべての遺伝子座について突然変異するか否かの判定を行い、突然変異する遺伝子座の遺伝子を他の対立遺伝子に置き換える。

3-5-1. 突然変異率

染色体長 (Chromosome Length) に占める突然変異する遺伝子座の割合を定めるパラメータが突然変異率 (Mutation Rate) である。前述の突然変異手法の内、通常の突然変異においては、このパラメータを比較検討する必要がある。

SGAのパラメータは多く、すべてのパラメータの組み合わせを調べることは現実的ではない。さらに、各パラメータを独立に変化させる方法ではパラメータ間の依存関係を把握することが困難となる。このため本研究では、SGAが交叉・突然変異の2つの遺伝的操作からなることに注目してパラメータを以下のように分類し、各分類に関してパラメータの比較を行った。

- ・交叉：交叉手法, 交叉率 (0.0 ~ 1.0)
- ・突然変異：突然変異手法, 突然変異率 (0.0 ~ 1.0)

4 数値実験

4-1. テストモデル (VRP Model)

パラメータの設定に関する実験には、組み合わせが、以下のような組み合わせ最適化問題の典型的な問題である TSP (Traveling Salesman Problem) に、様々な現実的な制約を考慮した VRP (Vehicle Routing Problem) を採用した。VRP は、配送先の需要や配送手段であるトラックのキャパシティなどを考慮しながら最適な巡回のルートを求める問題で、巡回先の数が増えると非常に難解となり、NP 完全問題に属するものである。

$$\begin{aligned}
 &\text{Minimize} && \sum_{(i,j) \in A} c_{ij} x_{ij} \\
 &\text{subject to} && \sum_{j \in V} x_{ij} = 1 \quad \forall i \in N \\
 &&& \sum_{j \in V} x_{ji} = 1 \quad \forall i \in N \\
 &&& x_{ij} = 1 \Rightarrow y_i - q_i = y_j \quad \forall (i,j) \in I \\
 &&& q_i \leq y_i \leq Q \quad \forall i \in V \\
 &&& x_{ij} \in \{0,1\} \quad \forall (i,j) \in A
 \end{aligned}$$

4-2. GA パラメータの効果

パラメータの変動による効果を観測するため、crossover rate と mutation rate をそれぞれ 0.0 から 1.0 まで 0.1 の間隔で模擬実験を行った。二つのパラメータの相関性は無視できないため、単独ではなく組み合わせで実験を行った。

- Population size : 101
- Selection : エリート戦略
- Crossover method : one-point crossover
- Mutation method : bit inverse
- Generation : 2000
- Crossover & Mutation rate : 0.0 ~ 1.0

実験は、Windows XP, Pentium 4 2GHz, 1G-RAM の PC で行い、開発ツールとしては、Borland C++ Builder を使用した。

² VRP モデルについて、詳しくは Han(2002) と Golden(1998)を参照されたい。

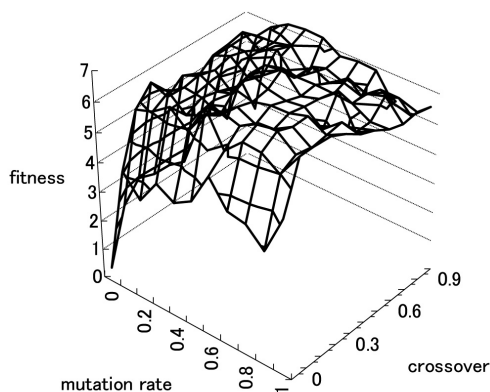


図 4-1 交叉と突然変異における GA パラメータの効果

図 4-1 は、交叉と突然変異のパラメータの変化による Fitness Value（評価値）を表している。その二つのパラメータの変動による評価値の変動が激しく、解の精度にも影響があることが観測できた。さらに、GA の重要パラメータである、交叉と突然変異の影響力も確かめることができた。

5 柔軟的なパラメータを持つ GA の提案

本研究では、問題ごとに適切なパラメータの設定が難しいことに着目し、変動型のパラメータ設定を持つ GA を提案する。ここでは、交叉と突然変異二つのパラメータを考慮し、0.0 から 1.0 まで 0.1 の間隔で変動するよう設定した。

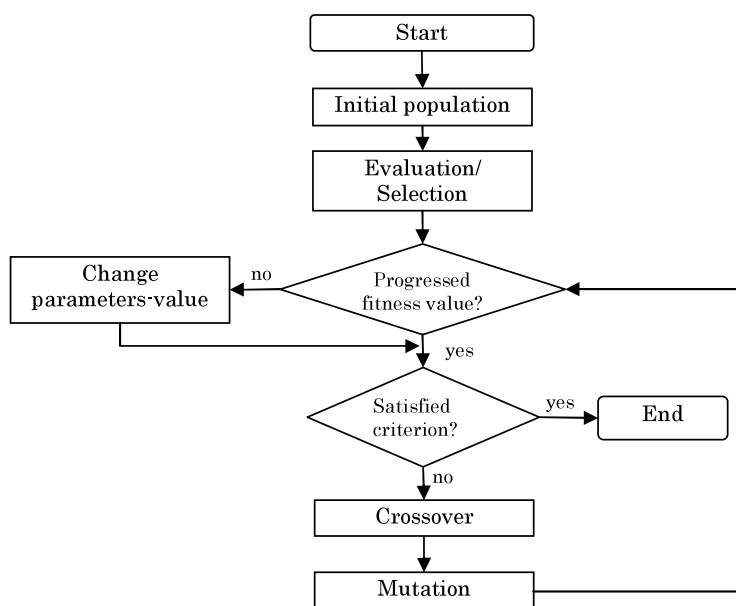


図 5-1 変動型パラメータを持つ GA

図 5-1 が示すように、一定の間隔で適度関数の値を測り、その値の変化が見られない場合に限って交叉率と突然変異率を変動させる。パラメータは、初期値から徐々に大きくして、長い世代に渡っても改善が見られない場合には、逆に値を下げていく仕組みである。

6 数値実験と結果

以下、テスト問題に対する数値実験の結果をまとめる。解の精度を同時に観測するため、世帯数を 3000 回に増やして模擬実験を行った。

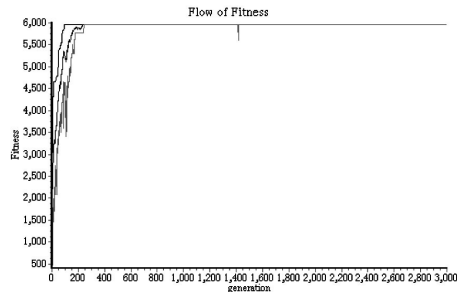


図 6-1 悪いパラメータの設定

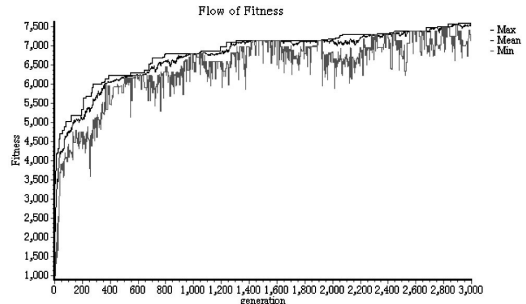


図 6-2 一般的なパラメータの設定

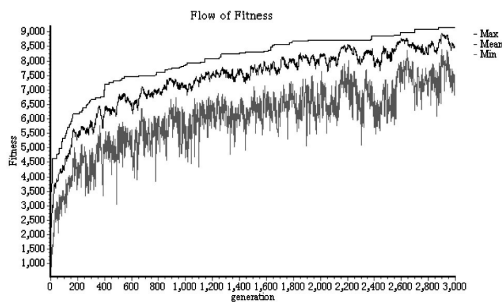


図 6-3 変動型のパラメータの適用

図 6-1 は、数多くの実験の中でパラメータの設定が悪いケースの結果を表している。グラフから見ると、解が早く収束してしまい精度が非常に悪くなっている。図 6-2 では、一般的なパラメータの設定で行った結果の一例である。解の初期収束性はあまり見られなく良い動きを見せている。しかしながら、図 6-3 と比べると、同じ世代であっても適度度が低いことが分かる。今回の模擬実験で、変動型のパラメータを適用して得られたデータから、本来の方法より平均 12% の改善が得られた。

7 結論および今後の課題

今回の模擬実験では、予測より良い結果を得られたが、まだ残っている課題は多い。まず、世代数（繰り返し）が 5000 回、10000 回、50000 回のように増えていく場合、どの時点の解がチューニングによる効果があったかを判断するのは難しい。ただし、実用的なレベルから考えると、決められた時間内での結果としては十分意味があると考えられる。今回の研究の成果と

して挙げた，自動化したパラメータの効果については，このテスト問題に限って言えば十分意味があると思われるが，様々なケースに対してアルゴリズム開発や模擬実験を行う必要がある．また，パラメータを決める方法自体を，GAのようなメータ・ヒューリスティック手法を使った学習型で解決することを今後の課題として考えたい．

参考文献

- [1] Goldberg DE (1989) Genetic Algorithm in Search, Optimization, and Machine Learning, Addison-Wesley.
- [2] Han SH (2001) Design of experiments to identify optimal parameters of genetic algorithm, Journal of Scientiae Mathematica Japonica, Vol.55(3), pp. 539-545
- [3] Heinz M (1997) Genetic algorithms. In Local Search in Combinatorial Optimization, John Wiley & Sons, Chichester, 137-172.
- [4] Holland JH (1975) Adaption in Natural and Artificial Systems. University of Michigan Press, Ann Arbor.
- [5] Ph.Vincke, Multicriteria Decision-Aid, John Wiley and Sons, Chichester, New York, 1989.
- [6] Thangiah Sam (1995) Vehicle routing with time windows using genetic algorithms. In Application Handbook of Genetic Algorithms, New Frontiers, CRC Press, 2, 253-277.
- [7] B. L. Golden, A. A. Assad (Eds.): "Vehicle Routing: Methods and Studies, Studies in Management Science and Systems", Vol. 16, Elsevier Science Publ., Amsterdam, 1988, 7-45.