

環境の変化に対応可能な GA の性能の向上

Improved performance of GAs that can respond to changes
in the environment須藤 令郁[†]兪 明連[†]横山 孝典[†]

Rei Sudo

Yoo Myungryun

Takanori Yokoyama

1. はじめに

遺伝的アルゴリズム^[1] (GA) は、生物の進化や生存戦略の原理に基づいたアルゴリズムであり、遺伝子の交叉、突然変異、自然淘汰の操作を用いて、最適解もしくは最適解の近似解を探索するアルゴリズムである。GA は、経路の最適化、物流、形状設計の最適化などの問題に応用されてきた。しかし、実問題において従来の手法では、モノの価値などの環境や条件などの変化があるときに、もう一度学習を行う必要がある。

そこで、環境や条件の変化にも対応できるロバストな解を探索する Robustness-oriented compact Genetic Algorithm^[2] (RcGA) に基づいた研究が行われている。しかし、これらの手法には、メモリ使用量が多く、収束時間が長いというデメリットがある。

本研究では、これらのデメリットを改善するための新しい方法を提案する。

2. 関連研究

本章では、関連研究として RcGA、RcGA with base value^[3] について説明する。

2.1 Robustness-oriented compact Genetic Algorithm (RcGA)

2.1.1 概要

RcGA は、確率モデルを用いた、環境の変化に対応可能な GA である。特徴は、学習前に生成した個体とそれを複数の評価指標で評価した評価値の組み

合わせを保存したデータセットを用いて学習を行うことで、ロバストな解を探索していることである。RcGA の全体の流れを図 1 に示す。

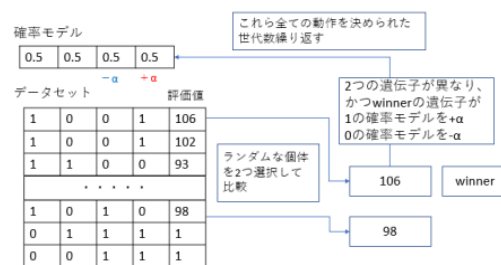


図 1. RcGA の全体の流れ

2.1.2 アルゴリズム

- 1) 遺伝子と評価値で構成されるデータセットから任意の個体を 2 つ選択する
- 2) 2 つの個体を比較し、評価値が高い個体を winner、低い個体を loser とする
- 3) 初期値が 0.5 である確率モデルに対して、2 つの遺伝子が異なり、かつ winner の遺伝子が 1 である部分に対応する確率モデルを α 増加、0 の部分に対応する確率モデルを α 減少させる。 α は確率モデルの更新率である
- 4) 1~3 の動作を予め定めた世代数繰り返し行い、収束されなかった場合には一般化する (確率モデルが 0.75 以上であれば 1、0.25 以下であれば 0、それ以外を#(ドントケア))

2.1.3 問題点

RcGA は、評価値が異なる同じ遺伝子を同時に用いるため、学習の効率が悪いというデメリットがある。また、従来の GA は、適応度が高い遺伝子が次

[†] 東京都市大学 Tokyo City University

世代まで行く可能性高く、世代が進化するが、RcGA では世代交代を行わず、学習の初期から永続的に膨大な数の遺伝子情報を持ったデータセットを用いて学習を行うため、メモリ使用量が多いというデメリットがある。

2.2 RcGA with base value

2.2.1 概要

RcGA with base value は、RcGA を基に作られた手法である。RcGA with base value は、データセットを用いずに世代を進むことで、メモリ使用量を大幅に削減している。また、2つの個体を直接比較せず、基準点との比較を行うことで両個体共に評価値が低いもしくは高い場合に正しく学習させることができ、学習の効率を上げている。図 2 に RcGA with base value の流れを示す。

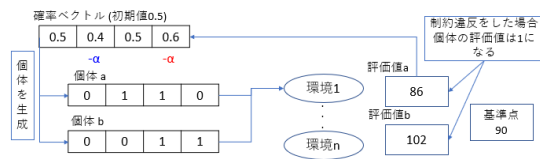


図 2. RcGA with base value の全体の流れ

2.2.2 アルゴリズム

- 1) 初期値 0.5 の確率ベクトルに従って 2 つの個体を生成する
- 2) 2つの個体を任意の環境(評価基準)で評価する。このとき、制約条件を違反した個体は、評価値が 1 になる
- 3) 2 つの個体の評価値をそれぞれ基準点と比較する
- 4) 基準点との比較に応じて確率ベクトル更新率 α に従って確率ベクトルを更新する
- 5) n 世代ごとに基準点を更新する
- 6) 全ての確率ベクトルが 0 か 1 に収束するまで 1 ~5 の動作を繰り返す

基準点と評価値を比較し、確率ベクトルを更新する際、3 パターンの更新方法が選ばれる。

(i) 2 個体の評価値が共に基準点を下回った場合、2 個体の中で同じ遺伝子が評価値を低くしている原因と考えられるため、両個体共に 0 の遺伝子に該当する確率ベクトルを確率ベクトル更新率 α 分加算し、両個体共に 1 の遺伝子に該当する確率ベクトルを α 分減少させる。

(ii) 2 個体の評価値が共に基準点を上回った場合 (i) と同様に、2 個体の中で同じ遺伝子が高い評価値を得た要因と考えられるため、両個体共に 0 である遺伝子に該当する確率ベクトルを α 分減算し、両個体共に 0 である遺伝子に該当する確率ベクトルを α 分加算させる。

(iii) 片方の個体のみ基準点を上回った場合、それぞれの遺伝子が異なる部分に対応する確率ベクトルを変動させる。2 個体の遺伝子が異なっている、かつ基準点を越えた個体の遺伝子が 1 である部分に該当する確率ベクトルに α 分加算し、基準点を越えなかった個体の遺伝子が 0 である部分に該当する確率ベクトルを α 分減少させる。

比較に用いる基準点は、n 世代ごとに更新される。

図 3 に n=10 の場合の例を示す。

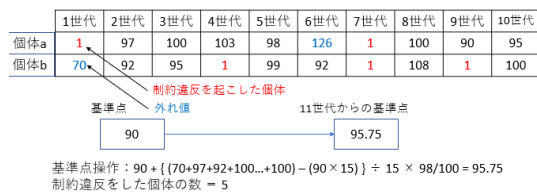


図 3. RcGA with base value の基準点更新の例

基準点更新は、n 世代ごとに、基準点に n 世代間の全ての個体の評価値とそれまでの基準点の差の平均を数パーセント下げたものを加算することで行われる。図 3 では 2 パーセント下げている。

2.2.3 問題点

RcGA with base value には、収束までに多くの世代が必要で、処理に時間がかかるという問題点がある。

3. 提案手法

3.1 概要

RcGA with base value の問題点を改善するために、確率ベクトル更新率を動的に変化させることで、収束の高速化を図る。

3.2 アルゴリズム

基本的な流れは、RcGA with base value と同様に行うが、確率ベクトル更新率に変更を加える。RcGA with base value では、常に一定の確率ベクトル更新率を用いて学習を行っていたが、提案手法では、確率ベクトル更新率を学習の過程で動的に変化させる。提案手法における確率ベクトル更新率の変化方法を図 4 に示す。

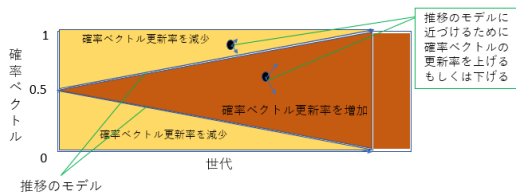


図 4. 確率ベクトル更新率の変化方法

提案手法では、確率ベクトルの理想の収束モデルを作成し、理想のモデルと実際の確率ベクトルとの距離に応じて確率ベクトル更新率を増減させる。図 4 の中心のエリアでは、確率ベクトル更新率を増加させ、上下のエリアでは確率ベクトル更新率を減少させる。理想のモデルは 0 に向かうモデルと 1 に向かうモデルがあり、それぞれ 10 世代ごとに初期の確率ベクトル更新率 α ずつ 0 と 1 に向かう。確率ベクトル更新率の動的変化の式を式 1、式 2 に示す。式 1 は、確率ベクトルが中心のエリアにある場合、式 2 は、確率ベクトルが上下のエリアにある場合である。 α' は確率ベクトル更新率、 m は

確率ベクトルの値がモデルから最も離れている (0.5 離れている) ときの確率ベクトル更新率の変化比率、 d は確率ベクトルの値から理想のモデルの値までの距離である。

$$\alpha' = \alpha \times \{1 + (m-1) \times d \div 0.5\} \quad (1)$$

$$\alpha' = \alpha \div \{1 + (m-1) \times d \div 0.5\} \quad (2)$$

4. 実験

4.1 変動型ナップサック問題^[2]

実験は、ナップサック問題を拡張させた、変動型ナップサック問題を用いて行う。ナップサック問題では、1つのアイテムセットを用いて学習を行うが、変動型ナップサック問題では、基準となるアイテムセットにそれぞれランダムな変化を与えた複数のアイテムセットを用いて学習を行う。よって、変動型ナップサック問題は、ある環境で高い評価であっても、他の環境では高い評価を得るとは限らず、高い評価を保ちつつも環境の変化に強いロバストな解の探索が求められる。変動型ナップサック問題のアイテムセット生成方法の例を図 5 に示す。

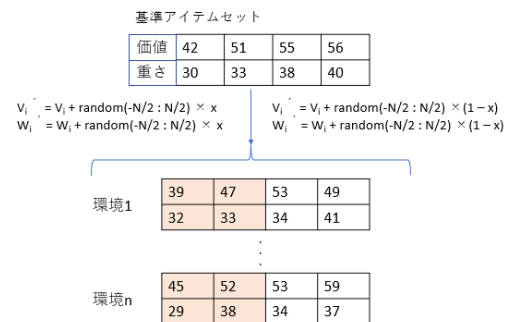


図 5. アイテムセット生成方法の例

図 5 における V_i は i 番目のアイテムの価値であり、 W_i は i 番目の重さ、 x は変動率、 N は変動幅を表している。基準アイテムの価値と重さに対して変動幅の範囲からランダムな値を選び、その値に対して半分のアイテムに変動率 x 、残りの半分のアイテム

ムに変動率 $1-x$ を乗算した値が変動後のアイテムセットとなる。変動率が 0 または 1 に近ければ近いほど数値の変動具合が大きく、0.5 に近づくほど数値の変動具合は等しくなる。

4.2 評価方法と使用パラメータ

提案手法と RcGA with base value を用いて変動型ナップサック問題を解き、収束までの世代数と解の精度を比較する。使用したパラメータを下記に示す。

- ・基準アイテムの価値と重さ : random(31 : 70)
- ・アイテムセットの数 : 5
- ・アイテムの変動幅 N : 60
- ・アイテムの変動率 x : 0.9
- ・アイテム数 : 30、50、100
- ・重量制限 : 1200、1500、3000
- ・確率ベクトル更新率 α : 0.01
- ・確率ベクトル変化比率 m : 5

4.3 実験結果

100 回の実行結果の平均値を表 1、表 2、表 3 に示す。

表 1. アイテム数 30、重量制限 1200 の実行結果

	RcGA with Base value		提案手法
解	1217.81	解	1131.656
世代数	11334.72	世代数	5572.99

表 2. アイテム数 50、重量制限 1500 の実行結果

	RcGA with Base value		提案手法
解	1716.682	解	1704.644
世代数	15102.49	世代数	6784.61

表 3. アイテム数 100、重量制限 3000 の実行結果

	RcGA with Base value		提案手法
解	3461.35	解	3389.046
世代数	17220.87	世代数	7019.21

アイテム数 50、重量制限 1500 の際のグラフを図 6 に示す。図 6 における赤い線が提案手法、青い線が RcGA with base value である。

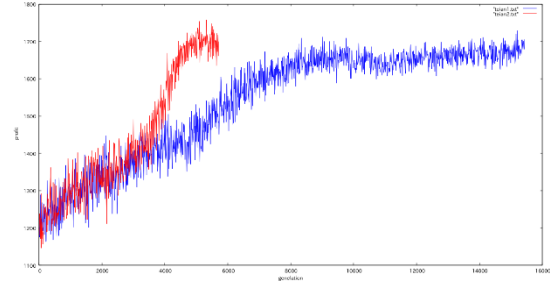


図 6. 実験結果のグラフ

5. 考察

表 1、表 2、表 3、図 6 を見ると収束までの世代数において全ての結果が RcGA with base value を倍近く上回っていることがわかる。これは、確率ベクトル更新率を動的に変化させることで、大幅に収束を早めることができたと考えられる。しかし、解の精度において、RcGA with base value を下回っている。これは、収束を早める過程で学習に影響を及ぼしていると考えられ、今回の実験よりも慎重に確率ベクトル更新率を変化させる必要があると考えられる。

謝辞

本研究は JSPS 科研費 20K11755 の助成を受けたものです。

参考文献

- [1] D. E. Goldberg : “Genetic Algorithms in Search, Optimization and Machine Learning”, Addison-Wesley (1989)
- [2] 田島友佑, 中田雅也, 佐藤寛之, 高玉圭樹 : “ナップサック問題における評価値変動に対応した遺伝的アルゴリズムの提案”, 情報処理学会研究報告, 2013-MPS-95, NO.6, PP.1-5 (2013)
- [3] 須藤 令郁 : “基準点操作を用いた評価指標の変化に対応可能な GA”, 東京都市大学卒業論文 (2022)