

分割と協調を導入した実数値 GA

Real-Coded GA Introducing Problem Decomposition and Cooperation

丹羽 貴敏 ^{*1}

Takatoshi Niwa

伊原 滉也 ^{*1*2}

Koya Ihara

加藤 昇平 ^{*1*2}

Shohei Kato

^{*1}名古屋工業大学 大学院工学研究科情報工学専攻

Dept. of Computer Science and Engineering Graduate School of Engineering, Nagoya Institute of Technology

^{*2}名古屋工業大学 情報科学フロンティア研究院

Frontier Research Institute for Information Science, Nagoya Institute of Technology

Genetic algorithms (GAs) are stochastic optimization methods that mimic the evolution of living organisms. Among them, Real-Coded GA (RCGA) is effective in continuous function optimization problems and is applied in various fields. Problem decomposition and cooperation is known as a method for improving search performance. In this research, we aim to improve efficiency of search by introducing problem decomposition and cooperation into RCGA. We focused on the group work which put into practice problem decomposition and cooperation in the real world. In group work, each of members collaborates with the others and is evaluated comprehensively from the outcome of the group and personal contribution to the group. We evaluated the search performance of the proposed method using six benchmark functions. As a result, we confirmed that the proposed method obtained the optimal solution with less evaluation number than the conventional methods.

1. はじめに

遺伝的アルゴリズム (GA) とは生物の進化を模倣した確率的最適化手法である。遺伝子型にビットストリングを用いる場合は離散的な表現になるため、連続性が全く考慮されない。一方遺伝子型に実数値をとる GA を実数値遺伝的アルゴリズム (RCGA) では連続値最適化問題において有効である。

RCGA の探索性能を上げる研究として世代交代モデルの提案がある [佐藤 97, 秋本 07]。既存の世代交代モデルとして Simple GA (SGA) や Minimal Generation Gap (MGG) [佐藤 97] が存在する。SGA では複製選択に適応度によるルーレット選択を用い、生存選択では親集団と子集団を入れ替える世代交代モデルである。複製選択とは子個体を生成する親個体の選択方法のことで、生存選択とは生成された子個体の中で次世代に残る子個体の選択方法のことである。SGA は探索序盤の選択圧が強いため初期収束が起きやすく、探索終盤では選択圧がかかりにくくなり進化が停滞してしまう。その問題点を解消するために [佐藤 97] は MGG を提案した。MGG の複製選択では集団からランダムに個体を取り出す。生存選択では親と子の中から適応度が最も高いもの 1 個体とルーレット選択により 1 個体を次世代に残す。MGG は SGA よりも初期収束や進化の停滞が起これにくく多様性の維持に優れている。しかし MGG は二親交叉を対象に設計されており、二親交叉は悪スケール性や変数間依存性に有効に働かない。一方多親交叉はそれらに対して有効に働くが MGG と組合せた場合十分な性能を発揮できない。そこで秋本らは Just Generation Gap (JGG) [秋本 07] を提案した。[秋本 10] により実数値 GA の利用では MGG と比較して JGG の方が探索効率が改善されたことが実証されたがいまだに最適解発見までに多くの適応度関数の計算が必要になる。そこで本稿では JGG に分割と協調を導入することで RCGA の探索効率を改善する。

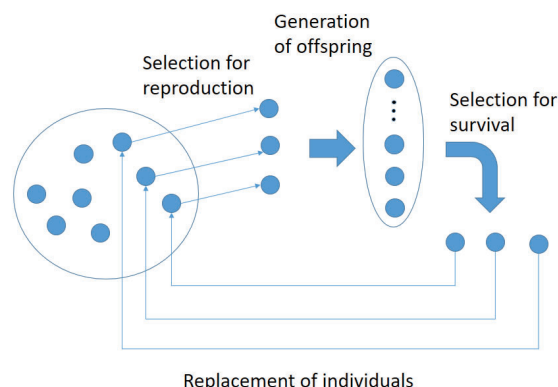


図 1: Just Generation Gap の流れ

2. 関連研究

多親交叉の潜在能力を引き出すための生存選択方法として JGG [秋本 07] が提案されている。JGG ではルーレット選択による収束速度の低下を防ぐため、最良の N_p 個体を生存選択する。また生存選択の対象が子個体群のため集団内の個体は一度だけ交叉に参加し、交代する。以下に JGG のアルゴリズムを示す。加えて図 1 に JGG の流れを示す。

- (1) 初期個体を N 個生成し集団を作成する。
- (2) 集団から N_p 個の個体をランダムに取り出し親個体群を生成する。
- (3) 親個体群に交叉を適用し N_c 個の子個体群を生成する。
- (4) 子個体群の評価を行う。

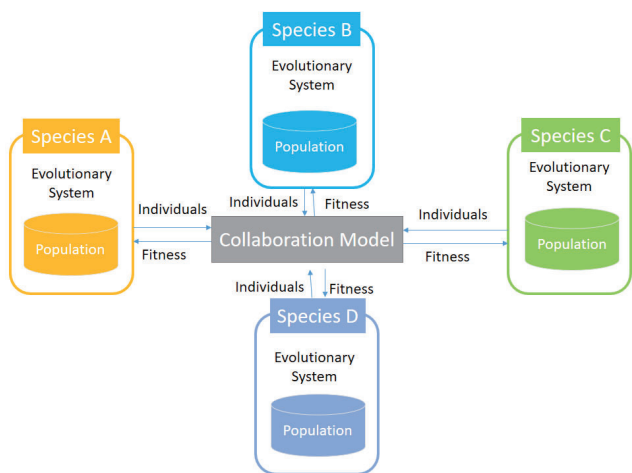


図 2: Cooperative Coevolution

- (5) 子個体群の中から適応度が高い上位 N_p 個体を選び集団に戻す。
- (6) 終了条件を満たすまで (2)~(5) を繰り返す。

3. 分割と協調

本稿では分割と協調としてそれぞれ決定変数ベクトルを複数の領域に分けることおよび異なる領域に属する個体同士を組合せて評価することを考える。

3.1 Cooperative Coevolutionary Genetic Algorithm

Evolutionary Algorithm における分割と協調の枠組みとして Cooperative Coevolution (CC) [Potter 94] がある。図 2 に CC の枠組みを示す。様々な Collaboration Model が提案されているが本稿では初期の CC の Collaboration Model を扱う。

図 2 の Evolutionary System に遺伝的アルゴリズムを使用したものを Cooperative Coevolutionary Genetic Algorithm (CCGA) と呼ぶ。CCGA では探索初期に決定変数ベクトルを分割し、各個体はランダムに他の領域の個体と組合せてその結果を適応度とする。以後は個体の適応度はそれぞれの領域における最良個体と組合せた決定変数ベクトルから得る。CCGA ではすべての個体の適応度を求めるのに $N_{field} \times N$ 回の適応度関数の計算が必要となる。以下に CCGA の手順を示す。

- (1) 決定変数ベクトルを N_{field} 個の領域に分割する。
- (2) 各領域ごとに初期個体を N 個生成する。
- (3) 個体をランダムに組合せて個体の適応度を求める。
- (4) 以下を各領域で実行する。
 - (a) 適応度に応じて個体を選ぶ
 - (b) 遺伝的操を行う。
 - (c) 個体の評価を行う。
- (5) 終了条件を満たすまで (4) を繰り返す。

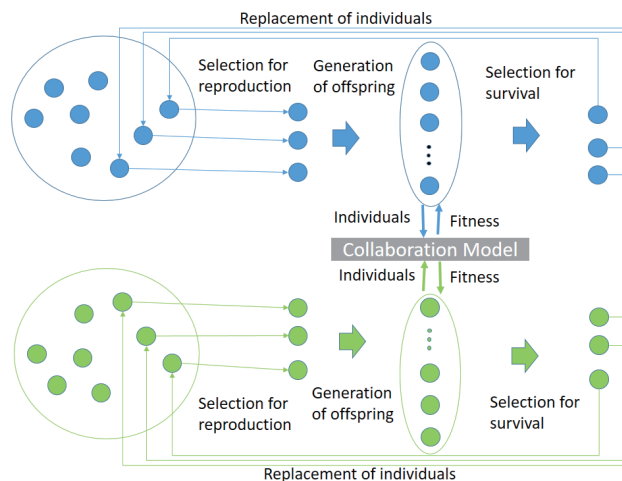


図 3: 提案手法の流れ

3.2 Group Work

実社会で分割と協調を実践している例としてグループワークがある。グループワークではグループに与えられた仕事を分割し、メンバーがそれぞれ分担して行う。メンバーは単独で評価されるのではなくグループの成果と個人の貢献度から総合的に決定される。本稿ではグループワークの特徴として以下の 2 点に注目する。

- (1) グループのメンバーで仕事を分担する。
- (2) グループでの仕事の成果は個人の評価に影響する。

これらを世代交代モデルに導入する。分割された解をメンバーと見立てるとメンバーの集まりであるグループが決定変数ベクトルを表す。したがってグループワークの個人の評価方法を個体評価に応用できるのではないかと考えた。グループを編成し、その評価をもとにグループに属した個体の適応度を算出する。すべての個体の適応度を求めるには分割数に関わらず N 回適応度を求めればよい。

4. 分割と協調を導入した世代交代モデル

CCGA は SGA の世代交代モデルに CC を導入したアルゴリズムに相当する。SGA と比較して、より多様性に優れており多親交叉との相性もよい JGG に分割と協調を導入した新しい世代交代モデルを提案する。以下に提案手法の流れを示す。加えて図 3 に分割数 2 の場合の例を示す。

- (1) 決定変数ベクトルを N_{field} 個の領域に分割する。
- (2) 各領域ごとに初期個体を N 個生成し集団を作成する。
- (3) 以下を各領域で実行する。
 - (a) 集団から N_p 個体をランダムに抽出し親個体群を生成する。
 - (b) 親個体群に交叉を適用し N_c 個の子個体群を生成する。
- (4) Collaboration Model により個体を評価する。
- (5) 各領域ごとに子個体群の中から適応度が高い上位 N_p 個体を選び集団に戻す。

(6) 終了条件を満たすまで (3)~(5) を繰り返す。

Collaboration Model として CC を用いたものを Cooperative Coevolutionary Just Generation Gap (CCJGG), グループワークを用いたものを Group Working Just Generation Gap (GWJGG) とする。個体の評価方法について説明する。CCJGG では CCGA と同様に探索序盤ではランダムに個体を組合せて適応度を求め、以後は個体の適応度はそれぞれの領域における最良個体と組合せた決定変数ベクトルから得る。GWJGG ではランダムにグループを編成する。このときすべての個体がいずれかのグループに属する。グループの適応度を求め、グループに属した個体の適応度としてそれを反映する。

5. 実験

5.1 ベンチマーク関数

実験に用いるベンチマークを以下に示す。

Sphere function

$$f_1(x) = \sum_{i=1}^n x_i^2 \quad (1)$$

$$x = [-5.12, 5.12], f_{best}(0) = 0$$

Rastrigin function

$$f_2(x) = A \times n + \sum_{i=1}^n (x_i^2 - A \cos(2\pi x_i)) \quad (2)$$

$$x = [-5.12, 5.12], f_{best}(0) = 0, A = 10$$

k-tablet function ($k = \frac{n}{4}$)

$$f_3(x) = \sum_{i=1}^k x_i^2 + \sum_{i=k+1}^n (100x_i)^2 \quad (3)$$

$$x = [-5.12, 5.12], f_{best}(0) = 0$$

Rosenbrock function

$$f_4(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2) \quad (4)$$

$$x = [-2.048, 2.048], f_{best}(1) = 0$$

Griewank function

$$f_5(x) = 1 + \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) \quad (5)$$

$$x = [-600, 600], f_{best}(0) = 0$$

以上の 5 関数に加えて Rastrigin function を回転させた Rotated Rastrigin function (f_6) も実験に用いる。 $f_1 \sim f_3$ は変数間に依存関係がない (分離可能な) 関数である。 f_3 は悪スケール性をもつ関数である。 $f_4 \sim f_6$ は変数間に依存関係がある (分離不可能な) 関数である。

5.2 実験設定

様々な分割方法が考えられるが、まずは適応度関数が 20 次元の場合、均等に分割した場合について性能を評価考察する。領域分割数は 2, 4, 5, 10, 20 の 5 通りである。各分割数で

の提案手法と JGG, CCGA との比較実験を行う。最適解との差が 1.0×10^{-7} 以内に収まることを最適解到達とする。最適解到達または関数評価回数が 1.0×10^7 以上になった場合に探索を終了する。各手法に対して 30 試行し、集団の大きさ N と子個体群 N_c のパラメータをグリッドサーチによる予備実験で決定した上で最良の性能を比較する。グリッド間隔を 50 としベンチマーク 6 関数で $50 \leq N \leq 500$, $50 \leq N_c \leq 500$ の範囲で探索を行う。また CCGA でも同様に個体数 N をグリッドサーチによる予備実験で決定した上で最良の性能を比較する。グリッド間隔を 50 としベンチマーク 6 関数で $50 \leq N \leq 500$ の範囲で探索を行う。交叉はシンプレックス交叉 [樋口 01] と UNDX-n [喜多 00] を用いる。各交叉においてパラメータ設定は推奨値を用いる。JGG における集団との交代個体数は親数と同数とすることが最もよいと [秋本 10] により実証されている。したがって JGG での交代個体数は親数と同数とする。提案手法では各領域において JGG の交代個体数と同数の個体を交代する。

5.3 結果

図 4 に結果を示す。(a)~(c) は分離可能な関数であるため、分割することによって効率的に探索をすることができ、分割を用いた手法が JGG よりも少ない評価回数で最適解に到達した。(b) の結果から分離可能であれば悪スケール性がある関数においても分割が有効に働くことを確認した。CCGA では分割数によっては最適化に到達できないものがあつた。一方 CCJGG や GWJGG ではどの分割数であっても最適解に到達することができた。JGG と組合せることで初期収束を避けることができたと考えられる。(d) では隣接する変数間に依存関係がある。GWJGG ではランダムに個体を組合せて適応度を求めるため、20 分割すると個体の適応度が正確に評価できなくなる。そのため最適解に到達することができなかった。しかしその他の分割数では JGG よりも少ない評価回数で最適解に到達した。(e) ではすべての手法において最適解に到達した。(f) では CCJGG や CCGA は最適解に到達することができなかった。また GWJGG では分割数によっては JGG よりも最適解到達までに多く評価回数がかかってしまったが最良の分割数では最適解到達までの評価回数の削減に成功した。(d)~(f) の結果から、分離不可能な関数においても GWJGG は有効に働き、JGG よりも少ない評価回数で最適解に到達することを確認した。

6. おわりに

本稿では JGG に分割と協調を導入した新たな世代交代モデルを提案した。ベンチマーク 6 関数を用いて性能評価を行い、JGG よりも GWJGG の方が良い解を獲得できることを確認した。本実験では最適な分割数が関数によって異なった。実問題では最適なパラメータを設定することができるのは困難である。そのため自動で最適な分割を行う手法を開発していく。個体の評価方法はグループの評価と同一のものとしたが、領域ごとに重みを追加するなどの個体評価も検討していく。最良の性能を得るためには変数間の依存関係を考慮し分割することが重要だと考える。したがって適応度関数の依存関係を分析する手法の開発に着手する。

参考文献

[Potter 94] Potter, M. A. and De Jong, K. A.: A Cooperative Coevolutionary Approach to Function Optimization,

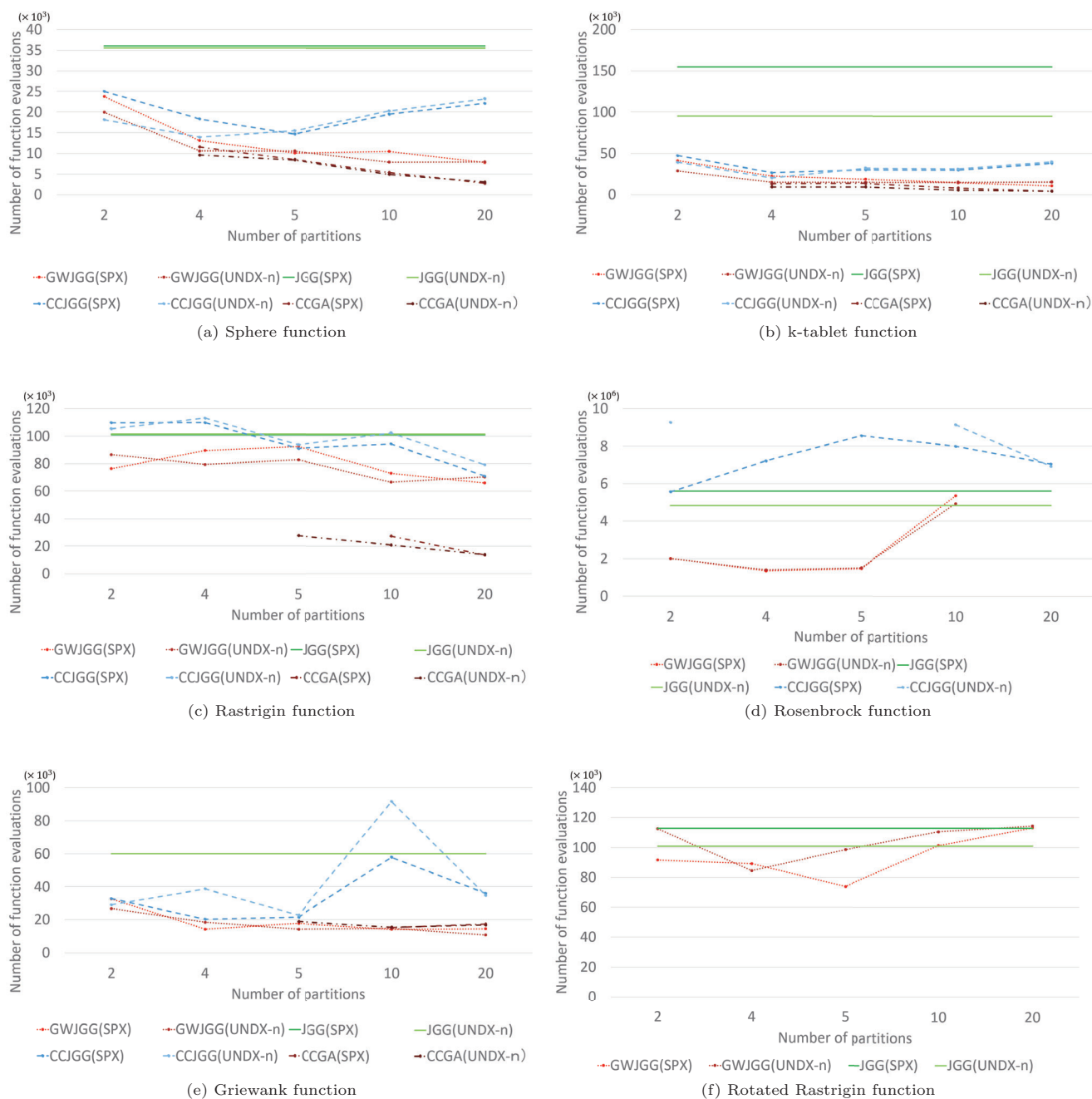


図 4: 既存手法との比較実験結果

in *International Conference on Parallel Problem Solving from Nature*, pp. 249–257 Springer (1994)

[喜多 00] 喜多一, 小野功, 小林重信: 実数値 GA のための正規分布交叉の多数の親を用いた拡張法の提案, 計測自動制御学会論文集, Vol. 36, No. 10, pp. 875–883 (2000)

[佐藤 97] 佐藤浩, 小野功, 小林重信 他: 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, 人工知能学会誌, Vol. 12, No. 5, pp. 734–744 (1997)

[秋本 07] 秋本洋平, 羽佐田理恵, 佐久間淳, 小野功, 小林重信: 多親を用いた実数値 GA のための世代交代モデル: Just

Generation Gap (JGG) の提案と評価, 自律分散システム・シンポジウム資料= SICE Symposium on Decentralized Autonomous Systems, 第 19 巻, pp. 341–346 (2007)

[秋本 10] 秋本洋平, 永田裕一, 佐久間淳, 小野功, 小林重信: 実数値 GA における生存選択モデルとしての MGG と JGG の挙動解析, 人工知能学会論文誌, Vol. 25, No. 2, pp. 281–289 (2010)

[樋口 01] 樋口隆英, 筒井茂義, 山村雅幸: 実数値 GA におけるシンプレクス交叉の提案, 人工知能学会論文誌, Vol. 16, No. 1, pp. 147–155 (2001)