

実数値 GA のフロンティア

The Frontiers of Real-coded Genetic Algorithms

小林 重信
Shigenobu Kobayashi

東京工業大学 大学院総合理工学研究科 知能システム科学専攻
Department of Intelligent Systems Science, Interdisciplinary Graduate School of Science and Engineering,
Tokyo Institute of Technology
kobayashi@dis.titech.ac.jp

keywords: Real Parameter Optimization, Real-coded Genetic Algorithms, Generation Alternation Model, Multi-parental Crossover, Global Descent Direction, JGG, $\text{REX}(\varphi, n+k)$, REX^{star}

Summary

Real-coded genetic algorithms (RCGA) are expected to solve efficiently real parameter optimization problems of multimodality, parameter dependency, and ill-scale. Multi-parental crossovers such as the simplex crossover (SPX) and the UNDX- m as extensions of the unimodal normal distribution crossover (UNDX) show relatively good performance for RCGA. The minimal generation gap (MGG) is used widely as a generation alternation model for RCGA. However, the MGG is not suited for multi-parental crossovers. Both the SPX and the UNDX- m have their own drawbacks respectively. Therefore, RCGA composed of them cannot be applied to highly dimensional problems, because their hidden faults appear.

This paper presents a new and robust framework for RCGA. First, we propose a generation alternation model called JGG (just generation gap) suited for multi-parental crossovers. The JGG replaces parents with children completely every generation. To solve the asymmetry and bias of children distribution generated by the SPX and the UNDX- m , an enhanced SPX (e -SPX) and an enhanced UNDX (e -UNDX) are proposed. Moreover, we propose a crossover called $\text{REX}(\varphi, n+k)$ as a generalization of the e -UNDX, where φ and $n+k$ denote some probability distribution and the number of parents respectively. A concept of the globally descent direction (GDD) is introduced to handle the situations where the population does not cover any optimum. The GDD can be used under the big valley structure. Then, we propose REX^{star} as an extension of the $\text{REX}(\varphi, n+k)$ that can generate children to the GDD efficiently. Several experiments show excellent performance and robustness of the REX^{star} . Finally, the future work is discussed.

1. はじめに

パラメータが連続的な値を取る実数値関数の最適化を関数最適化(function optimization)^{*1}という。関数最適化問題は、システムの解析・同定・制御・計画・設計・運用など科学技術や産業応用の様々な分野、様々な現場で日常的に直面するきわめて普遍的な問題クラスである。

関数最適化問題は、一般に、次式のように定式化される。

$$\min_x f(x), x \in S \subseteq R^n \quad (1)$$

実行可能領域 S に含まれる解 x を実行可能解という。

$$f(x^*) \leq f(x), \forall x \in S \quad (2)$$

を満たす実行可能な x^* を最適解または大域的最適解という。実行可能な x^* の ε 近傍内に $f(x') < f(x^*)$ となるような $x' \in S$ が存在しないとき、 x^* を局所最適解という。

非線形計画では、実行可能領域 S は制約関数を用いて

$$S = \{x | g_i(x) \leq 0, i = 1, \dots, m\} \quad (3)$$

のように明示的に表される場合が多い。さらに、目的関数 f と制約関数 $g_i, i = 1, \dots, m$ の連続微分可能性や凸性が仮定される。その仮定のもとに、非線形最適化手法が今日までに大きな発展を遂げている [今野 78, 小島 01]。

一方、GA(Genetic Algorithms) に代表される進化計算(evolutionary computation) では、目的関数の微分可能性や凸性を仮定しない。制約関数についても同様であり、これを明示的に与える必要はなく、解の実行可能性を判定することができればよいとの柔軟な立場を取る。

§1 関数最適化問題の特徴

目的関数の幾何学的な形状を適応度景観 (fitness landscape) または景観という。微視的に見ると小さな谷が多数存在するが巨視的に見ると大きな谷がひとつしか存在しない景観を大谷構造(big valley structure)[Boese 95] または大域的単峰 (globally unimodal) という。さらに、複数の大谷構造を持つ景観を大域的多峰(globally multimodal) という。大域的多峰のもとで GA の集団全体が有力な局所最適解の谷に引き込まれて最適解の発見が困難になるとき、そのような景観は騙し構造(deceptive structure) ま

^{*1} 連続変数最適化 (continuous parameter optimization) または実変数最適化 (real parameter optimization) ともいう

たはUV 構造[池田 02]*2が存在するという。

関数最適化において考慮すべき重要な性質として変数間依存性と悪スケール性がある。変数間に非線形な依存関係が存在するために、複数の変数を同時に変化させなければ目的関数の改善を効率的に行えない性質を変数間依存性(parameter dependency)、座標系のスケールの取り方が異なるために、目的関数に対する感度が変数によって大きく異なる性質を悪スケール性(ill-scale)という。

伝統的なビットストリング GA[Goldberg 89]では、実数値ベクトルをビットストリングに変換した後、多点交叉や一様交叉で子個体を生成し、実数値ベクトルに逆変換する。この過程において変数間依存性や悪スケール性などの情報が失われるため、突然変異などの補助的手段を用いても、関数最適化における性能は低い水準に留まる。

§2 実数値 GA の枠組み

実数値ベクトルを遺伝子型として扱う GA を実数値 GA (Real-coded Genetic Algorithms) という。実数値 GA では、交叉によって実数値ベクトルを直接操作するので、親個体群の近傍に子個体を精確に生成できる。

実数値 GA の一般的な手順を図 1 に示す。現代の集団に対し、複製選択・交叉・生存選択の操作を順に適用して、次世代の集団を得る。この手続きを、終了条件が満たされるまで、繰り返す。実数値 GA では、通常、突然変異(mutation)のような補助的手段を必要としない。

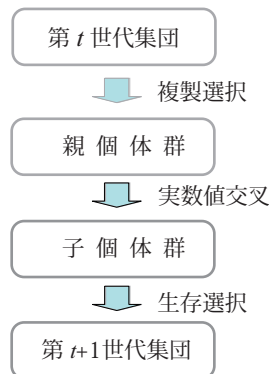


図 1 実数値 GA の一般的な手順

集団に含まれる個体数 n_{pop} を集団サイズ(population size) という。 n_{pop} は多様性(diversity)を制御する重要なパラメータであるが、通常、世代を通じて一定とされる。交叉に用いる親個体を集団から選ぶことを複製選択(reproduction)、親個体の近傍に子個体を生成する操作を交叉(crossover)という。親個体数 n_p は交叉によって異なる。ひと組の親個体に対し、交叉を繰り返し適用して子個体を n_c 個生成する。 n_c は、通常、次元数 n の数倍から 10 倍程度の大きさに設定される。親個体群と子個体群の中から次世代に残す個体を選ぶことを生存選択

(survival selection)、複製選択と生存選択を合わせて世代交代モデル(generation alternation model)という。

実数値ベクトルを遺伝子型として用いる同様の枠組みとして、EDA[Larranaga 00]CMA-ES[Hansen 04]がある。EDA や CMA-ES は、分散共分散行列を明示的に扱うため、 $O(n^3)$ の計算量を必要とする。実数値 GA では分散共分散行列は非明示的に扱われるので、計算量が $O(n^2)$ に抑えられことは大きな利点のひとつといえる。

§3 従来の取り組み

実数値ベクトルを遺伝子型とする実数値 GA への取り組みは 90 年頃に始まり、様々な交叉が提案された [Davis 90, Wright 91, Mühlenbein 93, Eshelman 93]。その中で、BLX- [Eshelman 93] は突然変異などの補助的手段を用いないで比較的良好な性能を示す。BLX- は両親によって定まる各変数の区間を 1 倍拡張してつくられる超直方体内に子個体を一様に生成する。BLX- は悪スケール性の影響を受けにくい。しかし、Rosenbrock 関数のように変数間依存性が強い関数には対処できない。

UNDX[Ono 97, 小野 99] は、変数間依存性へ対処することを目標に設計された交叉であり、両親を結び主軸を主探索成分、主軸に直交する超平面を担う副探索成分として正規分布によって子個体を生成する。UNDX は主探索成分による直線探索によって変数間依存性が比較強い関数に対して良好な収束性能を示すが、 k -tablet 関数のように悪スケール性が強い関数には対処できない。

探索途中で集団が縮退してしまう初期収束(premature convergence) や局所最適解に陥ることを回避するためには、集団の多様性維持(maintenance of diversity)が必要である。集団サイズを大きく取れば、多様性を確保できる。しかし、集団サイズを増やすと収束速度はほぼ比例して遅くなるので、集団サイズにできるだけ頼らないで多様性維持を図ることが世代交代モデルの役割とされる。

多様性維持を考慮した世代交代モデルとして IGS[Ackley 87], SS[Syswerda 90], CHC[Eshelman 90], ER[Thirrens 94], TDGA[森 96], MGG[佐藤 97] などが提案された。その中で、MGG[佐藤 97] は多様性維持に比較優れており、UNDX との相性が良く、レンズ系設計での成功例 [Ono 00] もあって、MGG はこれまで広く使われてきた。

UNDX/MGG の成功例が契機となり、[山村 98] は、GA による理想的な探索のシナリオとして、交叉は親個体が覆う局所的領域の補間的探索に徹し、探索領域の絞りこみは世代交代モデルに委ねるのがよいとの機能分担仮説を提唱した。[喜多 99a, 喜多 99b] は、機能分担仮説に基づく実数値 GA の設計指針として、交叉は親個体群の平均値ベクトルと分散共分散行列を保存するのがよいとの統計量遺伝(preservation of statistics)を提唱した。

統計量遺伝を設計指針として、UNDX- m [喜多 00], SPX[Tsutsui 99, 樋口 01], ENDX [木村 00] などの多親交叉が相次いで提案された。SPX は BLX- の変数間依存性を改善することを意図したもので、 $n+1$ 個の親個

*2 通常、実数値 GA は大谷構造を仮定する。本論文でも大谷構造を仮定するが、UV 構造への対応は 5.4.4 節で議論する

体を頂点とする n 次元単体 (simplex) を相似的に ε 倍拡張した領域内部に子個体を一様に生成する。拡張率は $\varepsilon = \sqrt{n+2}$ のとき、統計量遺伝を満たす [樋口 01]。

UNDX- m [喜多 00] は UNDX のスケール依存性の改善を意図したもので、 $m+2$ 個の親を用いて、 m 次元部分空間を担う主探索成分と $n-m$ 次元部分空間を担う副探索成分により子個体を生成する。UNDX- m は、主探索成分のパラメータは統計量を保存するように、副探索成分は UNDX との整合性を考慮したうえで次元数 n への依存性をなくすように定めている。数値実験の結果などに基いて、 $m=4\sim 5$ を推奨値^{*3}としている [喜多 00]。

実数値 GA の基礎研究は、設計指針としての機能分担仮説や統計量遺伝が提唱され、多親交叉が相次いで提案された頃がひとつのピークであった [小林 03] が、その後は、研究の主力が他分野の開拓などに向かったこともあり、新しい展開があまりないまま、現在に至っている。

§4 論文の目的と構成

本論文は、実数値 GA の深化と新たな展開を目標に、従来の実数値 GA の枠組みの問題点を明らかにすること、実数値 GA の在るべき姿を設計方針として示すこと、設計方針に基づいて実数値 GA の新しい枠組みを提案し、その妥当性と有効性を実験で検証することを目的とする。

2 章「問題提起および設計方針、評価方法」では、世代交代モデル MGG、多親交叉の SPX と UNDX- m 、機能分担仮説と統計量遺伝について、それぞれの問題点を指摘し、本論文の取り組み方を設計方針として述べる。

3 章「世代交代モデル JGG の提案」では、多親交叉向けの世代交代モデルとして、親個体群と子個体群が完全に入れ替わる Just Generation Gap (JGG) を提案する。

4 章「交叉 SPX、UNDX- n の改良」では、交叉 SPX の子個体生成分布の非対称性を指摘し、鏡映個体を用いた中心対称多面体内に子個体を一様に生成する enhanced SPX (e-SPX) を提案する。ついで、UNDX- n における主親の選び方によるバイアスを指摘し、すべての親個体を同等に扱う enhanced UNDX (e-UNDX) を提案する。

5 章「多親交叉の一般的枠組み $\text{REX}(\varphi, n+k)$ 」では、交叉 e-UNDX を一般化した枠組みとして、子個体生成に用いる確率分布 φ と親個体数 n_p を任意に選択できる $\text{REX}(\varphi, n+k)$ を提案し、 φ および n_p が探索性能にどのような効果を与えるかを実験によって明らかにする。

6 章「大域的降下方向を用いた REX^{star} の提案」では、集団や親個体群が最適解を覆っていない状況に対処するために、大域的降下方向という新しい概念を導入し、子個体生成分布の中心を親個体の重心から大域的降下方向に移動して子個体を生成する REX^{star} を提案する。 REX^{star} は、有望な領域に子個体を生成するので、従来の実数値 GA に比べて、安定収束に必要な集団サイズと生成子個体

数を大幅に引き下げ、収束速度の改善が著しいこと、さらに初期化領域の取り方に対して頑健であることを示す。

7 章では、本論文の研究成果を取りまとめたうえで、 REX^{star} の改良に向けて検討すべき課題と方針を述べる。

2. 問題提起および設計方針、評価方法

2.1 従来の取り組みの問題点

§1 世代交代モデル MGG の問題点

MGG の世代交代はつぎのように行われる [佐藤 97]。複製選択では、集団から交叉に用いる n_p 個体をランダムに非復元抽出して、親個体群とする。親個体群に対し交叉を繰り返し適用して、 n_c 個の子個体を生成する。生存選択では、まず親個体群から n_p-2 個体をランダムに選び、集団に戻す。つぎに、残りの親 2 個体と n_c 個の子個体を合わせた n_c+2 個体を家族と呼び、家族の中から評価値が最良の 1 個体とランキングに基づくルーレット選択で選ばれた 1 個体を加えて、次世代の集団とする。

MGG は評価値が劣る個体にも生存の機会を与えることにより多様性維持を図っている。しかし、ルーレット選択は収束速度を遅らせる原因となり、またエリート保存は局所最適解からの脱却を遅らせる原因となりうる。

さらに、MGG は 2 親を用いた交叉を前提に設計されていることから、多親を用いた交叉に MGG を用いる場合には n_p-2 個の親個体は無条件で次世代に残す措置が取られる。このため、次元数 n の増加とともに世代交代率が低下することになり、収束が遅れる原因となりうる。

§2 多親交叉 SPX、UNDX- m の問題点

$n+1$ 個の親個体を交叉に用いる SPX は、変数間依存性や悪スケール性に対処でき [Tsutsui 99, 樋口 01] 点で優れた交叉である。しかし、図 2 に示すように、SPX の子個体生成分布は親個体群の中心に対して対称ではない。子個体生成分布の非対称性は変数間依存性が強い関数において SPX の性能を低下させることが懸念される。

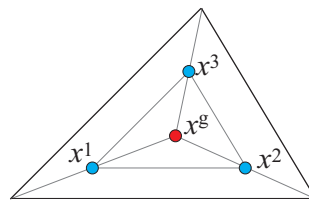


図 2 SPX の子個体生成分布の非対称性

UNDX- m は、推奨値 $m=4\sim 5$ のもとでは、変数間依存性、とくに悪スケール性の対処が十分ではない。変数間依存性や悪スケール性へ対処するためには、 $n+1$ 個の親個体を用いることが望ましい。 $n+1$ 個の親個体を用いる場合の UNDX- m をとくに UNDX- n と表記する。UNDX- n は次式によって子個体を生成する [喜多 00]。

*3 世代交代モデルとして 2 親交叉を前提とした MGG を用いた場合の推奨値であって、本論文で提案する多親交叉に適した世代交代モデル JGG を用いた場合の推奨値は $m=n$ である

$$x^c = x^g + \sum_{i=1}^n \xi^i (x^i - x^g), \xi^i \sim \mathcal{N}(0, \sigma_\xi^2) \quad (4)$$

ここで, x^g は親個体群 x^1, \dots, x^{n+1} の重心を表し, 子個体生成分布の中心となる. ξ^i は正規乱数 $\mathcal{N}(0, \sigma_\xi^2)$ に従うパラメータで, $\sigma_\xi^2 = 1/n$ のとき統計量遺伝を満たす.

ところで, UNDX- n の子個体生成を表す式 (4) の第 2 項の計算には n 個の親個体しか使われていない. すなわち, n 個体は主親扱いだが, 残りの 1 個体は重心 x^g を決めるために使われる副親扱いとなっている. 図 3 に示すように, UNDX- n では主親の選び方によって子個体生成分布にバイアスが生じる. このようなバイアスは UNDX- n の探索性能に何らかの悪影響を及ぼすことが懸念される.

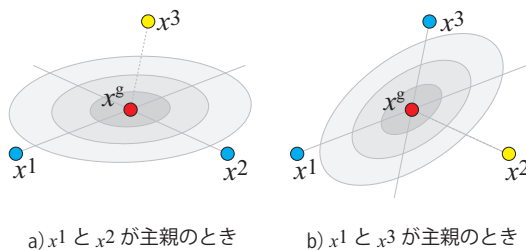


図 3 UNDX- n における子個体生成分布のバイアス

§3 機能分担仮説と統計量遺伝の問題点

親個体群の分布を変化させないように子個体を生成する交叉が望ましいとする機能分担仮説 [山村 98], その言い換えとして, 親個体群の平均値ベクトルと分散共分散行列を保存する交叉が望ましいとする統計量遺伝 [喜多 99a] は, 親個体群が最適解の存在するとみられる有望な領域を覆っている^{*4}ことを暗黙のうちに仮定している.

しかし, たとえば実数値 GA によって Rosenbrock 関数を探索する過程を観察してみればわかるように, 変数間依存性が高い関数最適化では, 集団全体が最初是最適解を覆っていても, やがて最適解を覆えなくなる状況に至る. また, 集団全体が最適解を覆っていたとしても, 交叉のためにランダムに選ばれた親個体群が最初から最適解を覆っていない状況は十分ありうる. さらに, 実問題では実行可能解を生成することは容易ではなく, 初期化領域の中に最適解が含まれていない状況は十分ありうる.

このような状況のもとでは, 親個体群の重心を中心に子個体を生成することにこだわると, 収束速度を遅らせるだけでなく, 初期収束の原因となることが懸念される.

一方, 親個体群の分散共分散行列は, 変数間の依存関係だけでなく, 分布の広がりに関する情報も持つので, 機能分担仮説の観点からも極めて重要である. しかし, 上述したような状況に対処するためには, 分散共分散行列を保存することを拘束条件と考えるのではなく, これを緩和して柔軟に利用することも考慮すべきと考えられる.

^{*4} ここでは, 集団または親個体群によって定まる凸包の内部に最適解が存在するという意味で「最適解を覆っている」という.

2.2 実数値 GA の設計方針

2.1 節で指摘した従来の実数値 GA の問題点を踏まえて, これらを克服する新しい枠組みを提案するための基本的な考え方を設計方針として述べる. さらに, 性能評価実験で用いるテスト関数と評価の方法についても述べる.

§1 世代交代モデルの設計方針

- (1) 複製選択 評価値の優劣にかかわらず, 個体が交叉に参加する機会はすべて平等とするという MGG の考えは多様性維持の観点からも妥当であり, 交叉に用いる n_p 個体を集団からランダムに非復元抽出して, 親個体群とする MGG と同じ考えを採用する.
- (2) 生存選択 交叉に参加した親個体群は, 評価値の優劣にかかわらずすべて淘汰する. すなわち, エリート保存戦略は採用しない. 親子間での完全世代交代により, 集団全体の世代交代率は n_p/n_{pop} となり, 探索過程を通じて一定の比率が維持される. また, ルーレット選択は, 不平等になりがちであるとの考えから採用しない. すなわち, 子個体群からの選択は評価値の高い順が望ましいとの考えを採用する.

§2 交叉の設計方針

- (1) 親個体数 変数間依存性や悪スケール性に適切に対処するためには, 親個体群の周辺の局所的景観を正確に把握することが必要であり, 少なくとも $n+1$ 個の親個体を用いるのが望ましいとの立場を取る.
- (2) 子個体生成分布のかたち 子個体生成分布は中心からみて対称であること, すなわち歪度 (skewness) が 0 であることが望ましく, また, 主親の選び方により子個体生成分布にバイアスが生じるような交叉は望ましくないとの立場を取る. 正規分布は生成子個体のエントロピーを最大にするという意味で合理的な分布である. しかし, 子個体数 n_c が限られているとき正規分布を用いることが妥当であるかどうかは疑問であり, 分布のかたちについては自由に選べるような枠組みであることが望ましいとの立場を取る.
- (3) 親個体群の分散共分散行列の保存 2.1.3 節で議論したように, 親個体群の分散共分散行列は, 変数間の依存関係だけでなく, 親個体群の広がりに関する情報を持つ. とくに, 親個体群の広がりには探索範囲を限定する上で有用な情報であり, 子個体群の広がりも親個体群と同程度にするという考え方は合理的である. しかし, 親個体群の分散共分散行列の保存を拘束条件と考えずに, 柔軟に利用する立場を取る.
- (4) 親個体群の平均値ベクトルの保存 2.1.3 節で議論したように, 親個体群が最適解をつねに覆っているという仮定はあまり当てにならない. 親個体群の外側に最適解が存在する可能性が高いならば, 親個体群の平均値ベクトルを保存することが合理的であるとは言えない. 親個体群の平均値ベクトルを保存することは妥当であるとはいえないとの立場を取る.
- (5) 大域的降下方向の推定と利用 親個体群が最適解の

存在する領域を覆っていない状況に対処するためには、最適解が存在する方向を同定し、その方向へ子個体生成分布の中心をシフトすればよい。しかし、親個体群の中心における勾配に基づく局所的な降下方向は多峰性関数では局所最適解に陥るリスクを高めるので利用することはできない。大谷構造を有する多峰性のもとで、大域的にみて降下方向とみなせる有望な方向を推定し、その方向に子個体生成分布の中心をシフトするのが望ましいとの立場を取る。

2.3 テスト関数および手法の評価

3章から6章では、提案手法の有効性を確認するために、既存手法との比較実験をしばしば行う。比較実験で用いるテスト関数および評価の方法を予め述べておく。

§1 テスト関数

1章で述べた関数最適化問題の特徴を典型的に示すテスト関数が数多く整備されている。悪スケール性・変数間依存性・多峰性のバランスや強さを考慮して、以下の関数をテスト関数として性能比較の実験で用いる。

Sphere 関数：

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

$$S = [-5.12, 5.12]^n, x^* = (0, \dots, 0)$$

性能評価の基本となる関数。

Ellipsoid 関数：

$$f(x) = \sum_{i=1}^n (1000^{i-1/n-1} x_i)^2$$

$$S = [-5.12, 5.12]^n, x^* = (0, \dots, 0)$$

比較的弱い悪スケール性。

k-tablet 関数 ($k = n/4$)：

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

$$S = [-5.12, 5.12]^n, x^* = (0, \dots, 0)$$

強い悪スケール性。

Rosenbrock 関数 (star 型)

$$f(x) = \sum_{i=2}^n (100(x_1 - x_i^2)^2 + (1 - x_i)^2)$$

$$S = [-2.048, 2.048]^n, x^* = (1, \dots, 1)$$

変数 x_1 と他変数の間に強い変数間依存性。

Rosenbrock 関数 (chain 型)*5

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

$$S = [-2.048, 2.048]^n, x^* = (1, \dots, 1)$$

隣り合う変数間に強い変数間依存性。

Bohachevsky 関数

$$f(x) = \sum_{i=1}^{n-1} (x_i^2 + 2x_{i+1}^2 - 0.3 \cos(3\pi x_i) - 0.4 \cos(4\pi x_{i+1}) + 0.7)$$

$$S = [-5.12, 5.12]^n, x^* = (0, \dots, 0)$$

比較的弱い多峰性。

Ackley 関数

$$f(x) = 20 - 20 \exp(-0.2 \sqrt{1/n \sum_{i=1}^n x_i^2}) + e - \exp(1/n \sum_{i=1}^n \cos(2\pi x_i))$$

*5 3章から5章の実験では、比較手法はいずれもこの関数の最適解を発見できないので、これらの章ではテスト関数から除外する。Rosenbrock 関数 (chain 型) は6章の実験でのみ用いる

$$S = [-32.768, 32.768]^n, x^* = (0, \dots, 0)$$

比較的弱い多峰性。

Schaffer 関数

$$f(x) = \sum_{i=1}^{n-1} (x_i^2 + x_{i+1}^2)^{0.25} \times (\sin^2(50(x_i^2 + x_{i+1}^2)^{0.1}) + 1.0)$$

$$S = [-100, 100]^n, x^* = (0, \dots, 0)$$

強い多峰性。

Rastrigin 関数*6

$$f(x) = 200 + \sum_{i=1}^n ((x_i - 1)^2 - 10 \cos(2\pi(x_i - 1)))$$

$$S = [-5.12, 5.12]^n, x^* = (1, \dots, 1)$$

強い多峰性。

§2 手法の評価方法

実験は世代交代モデル間の性能比較または交叉オペレータ間の性能比較を対象とする。とくに断らない限り、次元数 $n = 20$ 、初期化領域は定義域全体をデフォルト設定とする。最良個体の評価値が 10^{-7} 以下の値になったとき、最適解到達と判定する。本論文では、各手法に対して30試行試みてすべての試行が最適解に収束するとき安定収束(stable convergence)するという。安定収束する範囲内で、平均評価回数をできるだけ少なくなるように集団サイズ n_{pop} や生成子個体数 n_c などのパラメータを準備実験により決定したうえで、手法の性能比較を行う。

安定収束するときの平均評価回数*7によって2つの手法を比較する。すべてのテスト関数について、手法Aの平均評価回数が手法Bのそれより少なければ、手法Aは手法Bより優れていると判断できるので、 $A \succ B$ で表す。テスト関数が異なると手法の優劣が入れ替わる場合には、優劣はつけられないと判断されるので、 $A \sim B$ で表す。

3. 世代交代モデルJGGの提案

3.1 親子間完全世代交代モデルJGG

2.2.1節で述べた世代交代モデルの設計方針(1)と(2)に基づいて、多親交叉に適した世代交代モデルJGG(Just Generation Gap)を提案する[秋本07, 小林07]。

世代交代モデルJGG

複製選択 集団から n_p 個の個体をランダムに非復元抽出して、交叉に用いる親個体群とする
子個体の生成 親個体群に対し交叉を繰り返し適用して、 n_c 個の子個体を生成する
生存選択 子個体群から評価値が上位の n_p 個の個体を選び、集団に加える

JGGでは生存選択の対象は子個体に限定し、交叉に参加した親個体はすべて淘汰する。すなわち、親子間で完

*6 Rastrigin 関数は、初期化領域の真ん中に最適解が存在することが交叉によっては有利に働く場合がある。そこで、その影響を排除するために最適解の位置を $(1, \dots, 1)$ にシフトしている

*7 標準偏差は、手法によって多少の差異はあるが、いずれの手法も安定収束のもとでは平均評価回数に対し数%程度と十分小さいことが確認されたので、表に記載することは省略する

全世代交代を行う．生存選択の対象を子個体に限定しても，交叉により親の分布を適切に反映した子個体が十分生成されるならば評価値は改善できると見込まれる．

3.2 JGG の性能評価

実験 1 JGG と MGG の性能を比較する実験を行った．次元数 $n = 20$ とし，交叉は SPX を用いた．安定収束する集団サイズ n_{pop} ，生成子個体数 n_c ，平均評価回数 n_{eval} を表 1 に示す．「%」表示のある列の数字は平均評価回数について JGG の MGG に対する百分率^{*8}を表す．

表 1 JGG と MGG の比較 (20 次元/SPX)

	JGG	MGG	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	10n 6n 3.73×10^4	10n 2n 1.40×10^5	27
Ellip	12n 6n 5.50×10^4	12n 2n 2.10×10^5	26
k-tab	25n 6n 1.53×10^5	20n 2n 4.62×10^5	33
R _{star}	40n 8n 4.98×10^5	40n 2n 8.59×10^5	58
Boha	11n 6n 5.02×10^4	12n 2n 1.99×10^5	25
Ackl	12n 6n 8.70×10^4	12n 2n 3.32×10^5	26
Scha	17n 8n 3.31×10^5	25n 2n 1.54×10^6	22
Rast	50n 8n 3.60×10^5	50n 2n 1.34×10^6	27

表 1 から，JGG>MGG がいえる．JGG は MGG に比べて安定収束に必要な評価回数がかなり少ない．JGG の世代交代率が n_p/n_{pop} であるのに対し，MGG は高々 $2/n_{pop}$ に過ぎない．20 次元問題では，SPX を用いた場合，集団サイズが同じであれば世代交代率に 10 倍以上の差が生じる．したがって，世代交代率の違いが JGG と MGG の性能差に影響を与えた可能性は否定できない．

実験 2 世代交代率の違いが JGG と MGG の性能差に与える影響を取り除くために，JGG の生存選択に変更を加える．すなわち，MGG と同様に親個体群からランダムに選ばれた $n_p - 2$ 個体は集団に戻し，子個体群の中で評価値が上位の 2 個体だけを次世代集団に加えるモデル BBC (Best and 2nd-Best in Children) を考える．BBC に対して同様の実験を行った．結果を表 2 に示す．

表 2 から BBC>MGG がいえる．BBC は MGG に比べて 40% 以上少ない評価回数で最適解に到達している．Rastrigin 関数では MGG の性能がさらに劣るのは，エリート保存が局所最適解からの脱却を遅らせているためと考えられ，実際，収束曲線からも局所最適解からの脱却に手間取っていることが観察される．BBC はすべての関数について MGG より小さい集団サイズで安定収束していることから多様性維持に優れたモデルといえる．

以上の結果と考察から，親子完全入れ替えで多親交叉に対応し，多様性維持に優れた世代交代モデルとして JGG が推奨される．4 章以降の実験では，JGG を採用する．

*8 表 2 以下でも「%」表示のある列の数字の見方は同様である

表 2 BBC と MGG の比較 (20 次元/SPX)

	BBC	MGG	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	8n 2n 8.12×10^4	10n 2n 1.40×10^5	58
Ellip	10n 2n 1.19×10^5	12n 2n 2.10×10^5	57
k-tab	17n 2n 2.63×10^5	20n 2n 4.62×10^5	57
R _{star}	30n 2n 4.85×10^5	40n 2n 8.59×10^5	56
Boha	10n 2n 1.12×10^4	12n 2n 1.99×10^5	56
Ackl	10n 2n 1.89×10^5	12n 2n 3.32×10^5	57
Scha	20n 2n 8.38×10^5	25n 2n 1.54×10^6	54
Rast	30n 2n 4.43×10^5	50n 2n 1.34×10^6	33

4. 交叉 SPX，UNDX- n の改良

2.1.2 節で指摘したように，交叉 SPX には子個体生成分布の非対称性が，UNDX- n には主親の選び方に依存する子個体生成分布のバイアスが，それぞれ存在する．本章では，これらの問題点を解消するためのアイデアを述べ，改良された交叉により性能が向上することを示す．

4.1 e-SPX：SPX の改良版

$n+1$ 個の親個体 x^1, \dots, x^{n+1} が与えられたとき，親個体群の重心 x^g に対する各親個体の鏡映点(reflection point) $\underline{x}^1, \dots, \underline{x}^{n+1}$ を生成し，鏡映個体とよぶ．鏡映個体群は親個体群の分散共分散行列を保存する．親個体群と鏡映個体群を頂点とする多面体 P を構成する． P は中心対称(centrally symmetric)である．多面体 P を，図 4 に示すように， x^g を中心に相似的に拡張し，その内部に子個体を一様に生成する交叉を考える．これは子個体生成分布の非対称性を解消している点で SPX の改良版になっているので，e-SPX (enhanced SPX)^{*9} とよぶ．

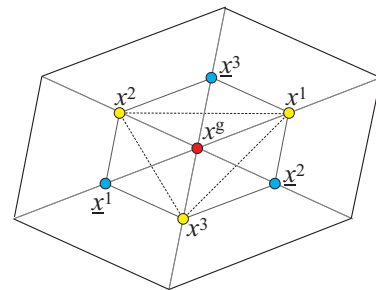


図 4 e-SPX の子個体生成分布

e-SPX の子個体生成分布が親個体群の分散共分散行列を保存するように拡張率 ε を決めることは自明なことではない．しかし，数値実験の結果などから SPX の拡張率 $\varepsilon = \sqrt{n+2}$ をそのまま用いて差し支えないことがわかっている．e-SPX は，子個体生成分布の非対称性を解消し

*9 実装上は，多面体 P を $n+2$ 個の単体に分解し，単体ごとに SPX のアルゴリズム [樋口 01] を適用し，合成すればよい

ているため，SPX に比べて性能の改善が期待できる．

実験 3 20 次元問題を対象に，JGG のもとで， e -SPX と SPX の性能比較実験を行った．子個体数 n_c は関数の難易度に応じて $6n$ または $8n$ とした．結果を表 3 に示す．

表 3 e -SPX と SPX の比較 (20 次元/JGG)

	e -SPX	SPX	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$7n \ 6n \ 3.30 \times 10^4$	$10n \ 6n \ 3.73 \times 10^4$	88
Ellip	$7n \ 6n \ 4.08 \times 10^4$	$12n \ 6n \ 5.50 \times 10^4$	74
k -tab	$9n \ 6n \ 6.52 \times 10^4$	$25n \ 6n \ 1.53 \times 10^5$	43
R_{star}	$15n \ 8n \ 1.79 \times 10^5$	$40n \ 8n \ 4.98 \times 10^5$	36
Boha	$7n \ 6n \ 4.05 \times 10^4$	$11n \ 6n \ 5.02 \times 10^4$	81
Ackl	$8n \ 6n \ 7.27 \times 10^4$	$12n \ 6n \ 8.70 \times 10^4$	84
Scha	$12n \ 8n \ 2.98 \times 10^5$	$17n \ 8n \ 3.31 \times 10^5$	90
Rast	$17n \ 8n \ 2.71 \times 10^5$	$50n \ 8n \ 3.60 \times 10^5$	75

表 3 から， e -SPX \succ SPX がいえる． e -SPX の性能は SPX をかなり上回っている． k -tablet 関数と Rosenbrock 関数 (star 型) における性能改善が著しい．また， e -SPX は SPX に比べて安定収束に必要な集団サイズが小さい．これらより， e -SPX は SPX における子個体生成分布の非対称性を解消することにより性能が改善されたといえる．

4.2 e -UNDX : UNDX- n の改良版

2.1.2 節で指摘した UNDX- n における主親の選び方に依存する子個体生成分布のバイアスを取り除くために，UNDX- n の子個体生成式 (4) をつぎのように変更する．

$$\mathbf{x}^c = \mathbf{x}^g + \sum_{i=1}^{n+1} \xi^i (\mathbf{x}^i - \mathbf{x}^g), \xi^i \sim \mathcal{N}(0, \sigma_\xi^2) \quad (5)$$

式 (5) による子個体生成分布が統計量遺伝を満たすように， $\mathcal{N}(0, \sigma_\xi^2)$ の分散を $\sigma_\xi^2 = 1/(n+1)$ に変更する．式 (5) による交叉を e -UNDX (enhanced UNDX- n) とよぶ．

実験 4 20 次元問題を対象に，JGG のもとで， e -UNDX と UNDX- n の比較実験を行った．子個体数 n_c は関数の難易度に応じて $6n$ または $8n$ とした．結果を表 4 に示す．

表 4 e -UNDX と UNDX- n の比較 (20 次元/JGG)

	e -UNDX	UNDX- n	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$7n \ 6n \ 3.03 \times 10^4$	$7n \ 6n \ 3.33 \times 10^4$	91
Ellip	$8n \ 6n \ 4.20 \times 10^4$	$8n \ 6n \ 4.51 \times 10^4$	93
k -tab	$11n \ 6n \ 7.10 \times 10^4$	$14n \ 6n \ 9.20 \times 10^4$	77
R_{star}	$17n \ 8n \ 2.10 \times 10^5$	$23n \ 8n \ 2.55 \times 10^5$	82
Boha	$7n \ 6n \ 3.77 \times 10^4$	$8n \ 6n \ 4.17 \times 10^4$	90
Ackl	$8n \ 6n \ 6.88 \times 10^4$	$9n \ 6n \ 7.46 \times 10^4$	92
Scha	$15n \ 8n \ 3.35 \times 10^5$	$16n \ 8n \ 3.50 \times 10^5$	96
Rast	$37n \ 8n \ 3.07 \times 10^5$	$39n \ 8n \ 3.15 \times 10^5$	98

表 4 から， e -UNDX \succ UNDX- n がいえる．顕著な差異はないが， k -tablet 関数と Rosenbrock 関数 (star 型) における性能改善が目立つ．また， e -UNDX は UNDX- n に比べて安定収束に必要な集団サイズが小さい場合が多い．これらより， e -UNDX は主親の選び方に依存するバイアスを解消することにより性能が改善されたといえる．

4.3 e -SPX と e -UNDX の比較

e -SPX と e -UNDX の性能比較を表 5 に示す．表 5 から e -SPX \sim e -UNDX である． e -UNDX が e -SPX の性能を上回っているのは，Sphere 関数，弱い多峰性の Bohachevsky, Ackley 関数だけで，これらの最適解は定義域の真ん中に存在することで共通する．その他の関数では e -SPX の性能が上回っており，集団サイズも小さい．

表 5 e -SPX と e -UNDX の比較 (20 次元/JGG)

	e -SPX	e -UNDX	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$7n \ 6n \ 3.30 \times 10^4$	$7n \ 6n \ 3.03 \times 10^4$	109
Ellip	$7n \ 6n \ 4.08 \times 10^4$	$8n \ 6n \ 4.20 \times 10^4$	97
k -tab	$9n \ 6n \ 6.52 \times 10^4$	$11n \ 6n \ 7.10 \times 10^4$	92
R_{star}	$15n \ 8n \ 1.79 \times 10^5$	$17n \ 8n \ 2.10 \times 10^5$	85
Boha	$7n \ 6n \ 4.05 \times 10^4$	$7n \ 6n \ 3.77 \times 10^4$	107
Ackl	$8n \ 6n \ 7.27 \times 10^4$	$8n \ 6n \ 6.88 \times 10^4$	106
Scha	$12n \ 8n \ 2.98 \times 10^5$	$15n \ 8n \ 3.35 \times 10^5$	89
Rast	$17n \ 8n \ 2.71 \times 10^5$	$37n \ 8n \ 3.07 \times 10^5$	88

e -SPX は一様分布を用いて， e -UNDX は正規分布を用いて，それぞれ子個体を生成する．しかし，その使いかたが異なるので，表 5 だけからは，子個体生成にどのような確率分布を用いるのが望ましいのかという結論を導くことはできない．確率分布の違いの影響を調べるためには，使いかたが同じ枠組みで比較する必要がある．

5. 多親交叉の一般的枠組み REX($\varphi, n+k$)

本章では，平均値ベクトルと分散共分散行列を保存する拘束条件のもとで，4.2 節で提案した e -UNDX を一般化した交叉の枠組みとして REX($\varphi, n+k$) を提案し，確率分布のかたちや親個体数の効果を調べ，考察を加える．

5.1 REX($\varphi, n+k$) の枠組み

2.2.2 節の設計方針 (1) に従い，交叉には次元数 n より大きい $n+k$ 個の親個体を用いる．親個体 $\mathbf{x}^1, \dots, \mathbf{x}^{n+k}$ の重心を \mathbf{x}^g で表し，次式にしたがって子個体を生成する．

$$\mathbf{x}^c = \mathbf{x}^g + \sum_{i=1}^{n+k} \xi^i (\mathbf{x}^i - \mathbf{x}^g), \xi^i \sim \varphi(0, \sigma_\xi^2) \quad (6)$$

ここで， $\varphi(0, \sigma_\xi^2)$ は，平均が 0，分散が σ_ξ^2 の任意の対称な確率分布を表し， σ_ξ^2 にはつぎのような条件を課す．

$$\sigma_{\xi}^2 = \frac{1}{n+k} \quad (7)$$

式 (6), 式 (7), に基づく交叉の枠組みを **REX**($\varphi, n+k$) とよぶ^{*10}. **REX**($\varphi, n+k$) は統計量遺伝を満たす^{*11}.

REX($\varphi, n+k$) では, 確率分布 φ を用いるとき, 式 (7) を満たすようにパラメータを設定する必要がある. たとえば, φ に区間 $[-a, a]$ の一様分布を用いるときは $a = \sqrt{3/(n+k)}$ に, 区間 $[-a, a]$ の両端の値が $1/a$ で, $x=0$ の値が 0 の V 字形分布を用いるときは $a = \sqrt{2/(n+k)}$ にそれぞれ設定すればよい. 正規分布, 一様分布, V 字形分布を, それぞれ記号 \mathcal{N} , \mathcal{U} , \mathcal{V} で表す. 4.2 節で提案した e-UNDX は, **REX**($\mathcal{N}, n+1$) にほかならない.

5.2 **REX**($\varphi, n+k$) における φ の効果

実験 5 **REX**($\varphi, n+1$) において確率分布 φ のかたちの違いが性能に与える効果を調べるため, 20 次元問題を対象に, JGG のもとで, $\varphi = \mathcal{N}, \mathcal{U}, \mathcal{V}$ の性能を比較する実験を行った. 子個体数 n_c は関数の難易度に応じて $6n$ または $8n$ とした. 結果を表 6 と表 7 に分けて示す.

表 6 **REX**($\mathcal{U}, n+1$) と **REX**($\mathcal{N}, n+1$) の比較 (20 次元/JGG)

	REX ($\mathcal{U}, n+1$)	REX ($\mathcal{N}, n+1$)	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$6n \ 6n \ 2.72 \times 10^4$	$7n \ 6n \ 3.03 \times 10^4$	90
Ellip	$6n \ 6n \ 3.38 \times 10^4$	$8n \ 6n \ 4.20 \times 10^4$	80
k -tab	$7n \ 6n \ 4.80 \times 10^4$	$11n \ 6n \ 7.10 \times 10^4$	68
R_{star}	$15n \ 8n \ 1.57 \times 10^5$	$17n \ 8n \ 2.10 \times 10^5$	75
Boha	$6n \ 6n \ 3.38 \times 10^4$	$7n \ 6n \ 3.77 \times 10^4$	90
Ackl	$6n \ 6n \ 5.54 \times 10^4$	$8n \ 6n \ 6.88 \times 10^4$	81
Scha	$9n \ 8n \ 2.29 \times 10^5$	$15n \ 8n \ 3.35 \times 10^5$	68
Rast	$22n \ 8n \ 2.20 \times 10^5$	$37n \ 8n \ 3.07 \times 10^5$	72

表 7 **REX**($\mathcal{V}, n+1$) と **REX**($\mathcal{U}, n+1$) の比較 (20 次元/JGG)

	REX ($\mathcal{V}, n+1$)	REX ($\mathcal{U}, n+1$)	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$5n \ 6n \ 2.42 \times 10^4$	$6n \ 6n \ 2.72 \times 10^4$	89
Ellip	$5n \ 6n \ 2.99 \times 10^4$	$6n \ 6n \ 3.38 \times 10^4$	88
k -tab	$6n \ 6n \ 4.17 \times 10^4$	$7n \ 6n \ 4.80 \times 10^4$	87
R_{star}	$12n \ 8n \ 1.48 \times 10^5$	$15n \ 8n \ 1.57 \times 10^5$	94
Boha	$5n \ 6n \ 3.09 \times 10^4$	$6n \ 6n \ 3.38 \times 10^4$	91
Ackl	$5n \ 6n \ 4.89 \times 10^4$	$6n \ 6n \ 5.54 \times 10^4$	88
Scha	$8n \ 8n \ 2.22 \times 10^5$	$9n \ 8n \ 2.29 \times 10^5$	97
Rast	$21n \ 8n \ 2.33 \times 10^5$	$22n \ 8n \ 2.20 \times 10^5$	106

表 6 から **REX**($\mathcal{U}, n+1$) \succ **REX**($\mathcal{N}, n+1$) がいえる. 悪スケール性の強い k -table 関数, 変数間依存性の強い

*10 **REX** は, Real-coded Ensemble Crossover の略称である

*11 **REX**($\varphi, n+k$) が統計量遺伝を満たす証明は付録 A に示す

Rosenbrock 関数, 多峰性の強い Schaffer, Rastrigin 関数では \mathcal{U} は \mathcal{N} に比べて 20~30% ほど評価回数が少ない.

REX($\mathcal{U}, n+1$) は親個体群の重心から各方向に三角分布で子個体を生成する. \mathcal{N} の性能が劣るのは, 子個体生成分布の裾が少し広がる反面, 親個体群の重心付近が厚目にサンプリングされる確率が高いためと考えられる.

表 7 から, Rastrigin 関数以外では, **REX**($\mathcal{V}, n+1$) \succ **REX**($\mathcal{U}, n+1$) がいえる. \mathcal{V} のほうが \mathcal{U} より若干性能が優れているのは, 式 (6) において V 字形分布で重みづけることが各親個体およびその鏡映個体周辺のサンプリング確率を高める効果があり, \mathcal{U} に比べて重心付近のサンプリングバイアスが少し緩和されるためと考えられる.

表 6 と表 7 から, **REX**($\varphi, n+1$) に用いる確率分布としては, おおむね, $\mathcal{V} \succ \mathcal{U} \succ \mathcal{N}$ の順に優れているといえる.

5.3 **REX**($\varphi, n+k$) における k の効果

REX($\varphi, n+k$) における k の効果を検証する. k を大き目に設定することにより親個体群が覆う領域が広くなり, かつ景観をより精確に捉えられるので, 景観が複雑な関数ほど k を大きくする効果が現れると期待される.

実験 6 $\varphi = \mathcal{U}$ の場合について, 20 次元問題を対象に, JGG のもとで, $k=1$ と $k=0.5n$ の比較実験を行った. この実験では集団サイズ n_{pop} は同じ値に, 子個体数 n_c は安定収束する最小の値に設定した. 結果を表 8 に示す.

表 8 **REX**($\mathcal{U}, 1.5n$) と **REX**($\mathcal{U}, n+1$) の比較 (20 次元/JGG)

	REX ($\mathcal{U}, 1.5n$)	REX ($\mathcal{U}, n+1$)	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$6n \ 6n \ 2.29 \times 10^4$	$6n \ 5n \ 2.50 \times 10^4$	92
Ellip	$6n \ 7n \ 3.08 \times 10^4$	$6n \ 5n \ 3.08 \times 10^4$	100
k -tab	$7n \ 7n \ 4.27 \times 10^4$	$7n \ 6n \ 4.80 \times 10^4$	89
R_{star}	$13n \ 10n \ 1.54 \times 10^5$	$13n \ 8n \ 1.68 \times 10^5$	92
Boha	$6n \ 6n \ 2.97 \times 10^4$	$6n \ 5n \ 3.16 \times 10^4$	94
Ackl	$6n \ 8n \ 5.31 \times 10^4$	$6n \ 6n \ 5.54 \times 10^4$	96
Scha	$9n \ 8n \ 1.88 \times 10^5$	$9n \ 8n \ 2.29 \times 10^5$	82
Rast	$22n \ 8n \ 1.83 \times 10^5$	$22n \ 8n \ 2.20 \times 10^5$	83

表 8 から **REX**($\mathcal{U}, 1.5n$) \succ **REX**($\mathcal{U}, n+1$) といえる. k -table, Rosenbrock(star 型), Schaffer, Rastrigin のように, 景観を精確に捉える必要がある関数において, **REX**($\mathcal{U}, 1.5n$) は **REX**($\mathcal{U}, n+1$) に比べて, 10% から 20% ほど評価回数が少ないので, k を大きくする効果はある.

5.4 考 察

REX($\varphi, n+k$) のパラメータの φ と k について考察する. さらに, **REX**($\varphi, n+k$) の UV 構造もとの最適化への適用および多目的最適化への適用について考察する.

§1 確率分布 φ のかたち

4 次の統計量である尖度を比べると, **REX**($\mathcal{N}, n+1$) がもっとも大きく, **REX**($\mathcal{V}, n+1$) がもっとも小さい. 5.2

節の実験結果は、尖度の小さい確率分布を用いたほうが安定収束に必要な集団サイズが小さく、評価回数が少ないことを示唆する．とくに、 k -tablet, Rosenbrock(star), Schaffer, Rastrigin 関数のように難しい関数において、 N が他に比べてかなり大きな集団サイズを必要とするのは、親個体群の中心付近を重点的に探索する傾向が強く、多様性を失いやすいためと考えられる． V は U に比べて集団サイズを $1n$ 下げられる分だけ評価回数が少ない．しかし、 V の集団サイズを U に合わせて同じにすると、両者の評価回数の差はほとんどなくなり、やや V のほうが多くなる．したがって、実用上は $\text{REX}(U, n+1)$ と $\text{REX}(V, n+1)$ のどちらを用いてもよいように思われる．

§2 親個体数 $n+k$

$\text{REX}(\varphi, n+k)$ では、 k は $1 \leq k \leq n_{pop} - n$ の範囲で任意に設定できる． k を大きくすると、親個体群が覆う領域が広くなり、また局所的景観をより精確に捉えることができるようになるので、有望な子個体の発見率は向上すると期待できる．しかし、 k を大きくし過ぎると、世代交代率が高くなるので、多様性が維持されにくくなる．すなわち、親個体数増加に伴う有望な子個体の発見率逓増と多様性逓減の間にはトレードオフの関係が存在する．

実験 7 k を極端に大きくすると、 $\text{REX}(\varphi, n+k)$ がどのような挙動を示すのかを調べるために、20 次元問題を対象に、JGG のもとで、集団の個体全部を親個体群とする $\text{REX}(U, n_{pop})$ と $\text{REX}(U, n+1)$ の比較実験を行った． $\text{REX}(U, n_{pop})$ では、関数の難易度に応じて、子個体数 n_c を予備実験により調節した．結果を表 9 に示す．

表 9 $\text{REX}(U, n_{pop})$ と $\text{REX}(U, n+1)$ の比較 (20 次元/JGG)

	$\text{REX}(U, n_{pop})$	$\text{REX}(U, n+1)$	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$6n \ 15n \ 1.94 \times 10^4$	$6n \ 5n \ 2.50 \times 10^4$	78
Ellip	$7n \ 19n \ 2.73 \times 10^4$	$6n \ 5n \ 3.08 \times 10^4$	89
k -tab	$8n \ 40n \ 4.64 \times 10^4$	$7n \ 6n \ 4.80 \times 10^4$	97
R_{star}	—	$15n \ 8n \ 1.57 \times 10^5$	—
Boha	$6n \ 21n \ 2.60 \times 10^4$	$6n \ 5n \ 3.16 \times 10^4$	82
Ackl	$6n \ 24n \ 4.38 \times 10^4$	$6n \ 6n \ 5.54 \times 10^4$	79
Scha	$9n \ 36n \ 1.69 \times 10^5$	$9n \ 8n \ 2.29 \times 10^5$	74
Rast	$15n \ 120n \ 2.01 \times 10^5$	$22n \ 8n \ 2.20 \times 10^5$	91

表 9 から $\text{REX}(U, n_{pop})$ は、Rosenbrock(star 型) 以外の関数では、 $\text{REX}(U, n+1)$ の性能を上回っている．表 8 と表 9 から、Sphere, Ellipsoid, Bohachevsky, Ackley, Schaffer 関数では、 $\text{REX}(U, n_{pop})$ が $\text{REX}(U, 1.5n)$ の性能を上回っている．これらの関数は、最適解が初期化領域の中央に存在し、悪スケール性や変数間依存性があったとしても弱い点で共通している．こういう特徴が予めわかっている関数に対しては、 $\text{REX}(U, n_{pop})$ は有効といえる．

しかし、 $\text{REX}(U, n_{pop})$ では、集団のメンバーが毎世代すべて入れ替わるので、集団が縮小する速度が早く、その

分多様性が失われやすい． $\text{REX}(U, n_{pop})$ が Rosenbrock 関数 (star 型) の探索に失敗するのは、稜上を探索する途中で多様性を失い、初期収束を起こすためと考えられる．

パラメータ k の経験的推奨値は $k = 1 \sim 0.5n$ であるが、利用する状況に応じて柔軟に設定することが望ましい．

§3 UV 構造もとの関数最適化への適用

UV 構造を持つ多峰性関数最適化問題に対する有力な接近法のひとつに適応的近傍探索ANS(Adaptive Neighboring Search)[高橋 01] がある．ANS は近傍限定による複製選択、交叉的突然変異XLM(Crossover-Like Mutation)、および局所決定的生存選択から構成される．ANS では各個体が、近傍に存在する他個体を参照しながら、それぞれ自律分散的かつ局所適応的に探索を進めることにより、複数の最適解や有力局所解を 1 試行で発見できる．

XLM は UNDX- m [喜多 00] をもとに設計されたオペレータであり、 m が次元数 n に等しいとき、主親 x^p を中心に副親 x^1, \dots, x^n を用いて、次式で子個体を生成する．

$$x^c = x^p + \sum_{i=1}^n \xi^i (x^i - x^p), \quad \xi^i \sim \mathcal{N}(0, \sigma_\xi^2) \quad (8)$$

式(8)は式(4)における親個体群の重心 x^g を主親 x^p で置き換えているため、親個体群の平均値ベクトルと分散共分散行列のどちらも保存されない．強い変数間依存性に対処するには、分散共分散行列は保存したほうがよいとの考えから、式(8)の第 2 項をつぎのように修正する．

$$x^c = x^p + \sum_{i=1}^{n+k} \xi_i (x^i - x^g), \quad \xi_i \sim \mathcal{N}(0, \sigma_\xi^2) \quad (9)$$

ここで、 x^p は主親、 x^g は $n+k$ 個の副親 x^1, \dots, x^{n+k} の重心であり、 $\sigma_\xi^2 = 1/(n+k)$ のとき、式(9)は副親個体群の分散共分散行列を保存する．式(9)で表される交叉的突然変異を拡張 XLM とよぶ．拡張 XLM は $\text{REX}(\mathcal{N}, n+k)$ における子個体生成の中心を重心 x^g から主親 x^p にシフトたものにほかならない．拡張 XLM を用いることにより、ANS の性能は大幅に改善されることが期待できる．

§4 多目的関数最適化への適用

多目的関数最適化問題は、一般に、次式で表される．

$$\min_x f(x) = (f_1(x), \dots, f_M(x))^T, \quad x \in S \quad (10)$$

解 $x_1, x_2 \in S$ に対し、 $\forall i \in \{1, \dots, M\}$, $f_i(x_1) \leq f_i(x_2)$ かつ $\exists i \in \{1, \dots, M\}$, $f_i(x_1) < f_i(x_2)$ が成り立つとき、 x_1 は x_2 より優れているといい、 $x_1 \succ x_2$ と表す．実行可能領域 S 内に $x' \succ x$ なる解 x' が存在しないとき、解 x をパレート解 (Pareto solution) という．一般に、パレート解は集合をなすので、これをパレート解集合とよぶ． x の実行可能な近傍に $x' \succ x$ なる解 x' が存在しないとき、 x を局所パレート解 (Pareto solution) という．

通常、 $n > M$ であり、このときパレート解および局所パレート解は局所的に $M-1$ 次元多様体をなすことが知られている [Hillermeier 01]．したがって、解析的に求ま

らない限り、パレート解をすべて求めることはできない。そこで、パレート解集合をよく近似する解集合を効率良く求めることが多目的最適化手法の目的とされている。

多目的最適化手法の性能は、求まる解集合のパレート解集合への近さと多様性で評価されることが多い [Deb 01, Knowles 04]。近さはつぎの要素からなる [原田 06]。局所パレート解乗り越え 解集合が、局所パレート解近傍に留まらないで、パレート解近傍に到達すること 高精度化 近似パレート解集合として精度が十分高いこと

多様性はつぎの要素からなる [Deb 01, Knowles 04]。広がり 解がパレート解集合の端まで広がっていること 均一さ 解集合が偏ることなく均一に分布していること 多峰性のもとでは、パレート解集合は一般に、複数の $M-1$ 次元多様体上に分散して存在する。したがって、多目的最適化手法には、パレート解が属する多様体をできるだけ多く見出すこと、そして多様体ごとにできるだけ高精度かつ均質な解集合を見出すことが求められる。

パレート解が存在する多様体をできるだけ多く見出すためには大域的探索(Global Search, 以下GS という)が必要であり、高精度で均質な解集合を見出すためには局所的探索(Local Search, 以下LS という)が必要である。

多峰性のもとでの多目的関数最適化の枠組みとして、[原田 06] はGS then LS の枠組みが有用であることを主張し、GS に UNDX, LS にパレート降下法[原田 06b]を用いた枠組みの有効性を既存手法との比較実験で示している。しかし、UNDX は 1.0.3 節で指摘したように悪スケール性に対処できない。パレート降下法は勾配法であるために、目的関数が不連続な場合には利用できない。

以上の考察および予備の実験の結果を踏まえて、本論文では GS の探索オペレータとして $\text{REX}(\mathcal{U}, n+1)$ を用いること、LS の探索オペレータとして $\text{REX}(\mathcal{U}, M)$ を用いることを推奨する。LS では M 個の親個体しか用いないのは、上述したように、パレート解集合は局所的には $M-1$ 次元多様体として存在するからである。GS における複製選択は、JGG に準拠して、集団からランダムに非復元抽出された個体を親個体群とするのが適当と思われる。一方、LS における複製選択は、集団からある個体 x をランダムに選び、残りの $M-1$ 個体を x の k -NN 近傍から選び、親個体群とするのが適当と考えられる。

6. 大域的降下方向を用いた REX^{star} の提案

本章では、2.1.3 節で指摘した「集団または親個体群あるいは初期化領域が最適解を覆っていない状況」に対処するために、2.2.2 節で述べた交叉の設計方針 (4)~(5) に基づいて、新しい交叉の枠組みを提案する。まず、大谷構造のもとで最適解が存在する方向を推定する大域的降下方向という概念を導入する。ついで、 $\text{REX}(\mathcal{U}, n+1)$ をベースに分布の中心を大域的降下方向にシフトして子個体を生成する REX^{star} を提案し、その性能を評価する。

6.1 大域的降下方向

大谷構造のもとでマクロにみて降下方向とみなせる方向を推定するために、4.1 節で SPX の拡張に用いた中心対称多面体^{*12}を利用する。 $n+1$ 個の点 x^1, \dots, x^{n+1} が与えられたとき、その重心 x^g に対し、 x^1, \dots, x^{n+1} の x^g の鏡映点を生成し、それぞれ $\underline{x}^1, \dots, \underline{x}^{n+1}$ で表す。元の点と鏡映点を合わせた計 $2(n+1)$ 個の点の中で、評価値が上位半分^{*}の点の重心を x^b 、評価値が下位半分^{*}の点の重心を x^w で表す。方向 $x^b - x^g$ を x^g における大域的降下方向(globally descent direction)とよぶ。 $x^g - x^w$ や $x^b - x^w$ を大域的降下方向とよんでも同じである。

図 5 に大域的降下方向の例を示す。 $x^1, \underline{x}^2, \underline{x}^3$ が評価値上位の点であり、その重心 x^b は $x^g x^1$ 上に存在する^{*13}ので、方向 $x^1 - x^g$ と大域的降下方向 $x^b - x^g$ は一致する。

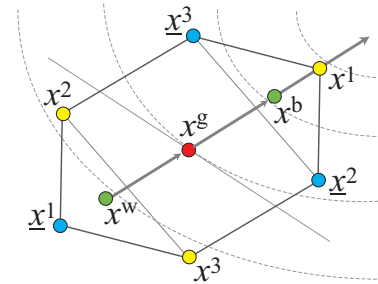


図 5 大域的降下方向の概念図

中心対称多面体の性質より、重心 x^g を通る任意の超平面は $2(n+1)$ 個の頂点集合を $n+1$ 個ずつの頂点集合に分離する。重心 x^g を通る目的関数 f の等高面の曲率が十分大きければ、図 5 がその例であるが、 x^g を通る超切平面によって $n+1$ 個の評価値上位の頂点群と $n+1$ 個の評価値下位の頂点群は分離される。したがって、子個体生成分布の中心を大域的降下方向にシフトすることにより、改善子個体の生成確率が高まると期待できる。

6.2 REX^{star} の枠組み

子個体生成分布の中心を、親個体群の重心ではなく、大域的降下方向へシフトする交叉を提案する。すなわち、 $\text{REX}(\mathcal{U}, n+1)$ を用いて、次式により子個体を生成する。

$$x^c = x^g + \text{diag}(\xi_1^t, \dots, \xi_n^t)(x^b - x^g) + \sum_{i=1}^{n+1} \xi_i^t (x^i - x^g) \quad (11)$$

$$\xi_j^t \sim \mathcal{U}(0, t); j = 1, \dots, n \quad (12)$$

$$\xi_i^t \sim \mathcal{U}(-\sqrt{3/n+1}, \sqrt{3/n+1}); i = 1, \dots, n+1 \quad (13)$$

*12 中心対称多面体の性質をまとめたものを付録 B に示す

*13 付録 B の「中心対称多面体」の性質 4 を参照されたい

ここで、パラメータ t は子個体生成分布中心の大域的降下方向への移動量を定めるものでステップサイズとよぶ。

式 (11) の第 1 項と第 2 項は、 x^g と $x^g + t(x^b - x^g)$ を対角頂点とする超直方体内に一様分布にしたがって子個体生成の中心を定める。第 2 項の付加により、親個体群の平均値ベクトルと分散共分散行列はいずれも保存されなくなり、統計量遺伝から乖離する。しかし、最適解が存在するとみられる有望な領域に子個体を生成するのが望ましいと思われるので、子個体生成分布の中心を大域的降下方向にシフトすることは合理的といえる。式 (11)~(13) により子個体を生成する交叉を REX^{star} とよぶ^{*14}。

6.3 REX^{star} の性能評価

実験 8 20 次元問題を対象に、JGG のもとで、 REX^{star} と $\text{REX}(\mathcal{U}, n+1)$ の性能比較実験を行った。 REX^{star} の子個体数 n_c は、関数の難易度に応じて、 $2n$ または $3n$ に設定した。ステップサイズ t と集団サイズ n_{pop} の間にはトレードオフの関係が存在するので、 t と n_{pop} は予備実験により表 11 に示す値に設定した。結果を表 10 に示す。

表 10 REX^{star} と $\text{REX}(\mathcal{U}, n+1)$ の比較 (20 次元/JGG)

	REX^{star}	$\text{REX}(\mathcal{U}, n+1)$	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$2n, 2n, 6.89 \times 10^3$	$6n, 5n, 2.50 \times 10^4$	28
Ellip	$2n, 2n, 8.46 \times 10^3$	$6n, 5n, 3.08 \times 10^4$	27
k -tab	$2n, 2n, 1.05 \times 10^4$	$7n, 6n, 4.80 \times 10^4$	22
R_{star}	$5n, 3n, 5.45 \times 10^4$	$15n, 8n, 1.57 \times 10^5$	35
R_{chai}	$2n, 3n, 4.72 \times 10^4$	—	—
Boha	$4n, 2n, 1.54 \times 10^4$	$6n, 5n, 3.16 \times 10^4$	49
Ackl	$2n, 3n, 1.44 \times 10^4$	$6n, 6n, 5.54 \times 10^4$	26
Scha	$5n, 3n, 7.70 \times 10^4$	$9n, 8n, 2.29 \times 10^5$	34
Rast	$20n, 3n, 1.23 \times 10^5$	$22n, 8n, 2.20 \times 10^5$	56

表 10 から $\text{REX}^{star} > \text{REX}(\mathcal{U}, n+1)$ である。 REX^{star} は、 $\text{REX}(\mathcal{U}, n+1)$ に比べて、安定収束する平均評価回数が非常に少ない。従来の実数値 GA では困難であった Rosenbrock 関数 (chain 型) の最適解を求めることに成功している。 REX^{star} の集団サイズと生成子個体数は、 $\text{REX}(\mathcal{U}, n+1)$ に比べて、単峰性関数では約 1/3 である。多峰性関数でも集団サイズは小さく、生成子個体数は 1/2 以下である。 REX^{star} が生成子個体数を 1/3 ~ 1/2 程度に削減できるのは、それだけ改善子個体数の生成確率が高いためと考えられる。また、 REX^{star} が集団サイズをすべての関数に対し削減できるのは、式 (11) の第 2 項の付加が多様性の拡大に寄与しているためと考えられる。

以上から、大域的降下方向を利用する REX^{star} が非常に優れた収束性能を有することが確認されたといえる。

*14 大域的降下方向を推定するために生成される $n+1$ 個の鏡映個体は、 REX^{star} によって生成される子個体と同じ扱いとして、評価回数にカウントしたうえで、生存選択の対象とする

6.4 高次元問題における REX^{star} の性能

実験 9 REX^{star} の高次元問題における性能を調べるために、20 次元、40 次元、80 次元、160 次元問題を対象に性能評価実験を行った。各ケースで用いた集団サイズ n_{pop} 、生成子個体数 n_c 、ステップサイズ t を表 11 に示す。生成子個体数 n_c は、Sphere, Ellipsoid, k -tablet, Bohachevsky 関数は $2n$ 、その他は $3n$ に設定した。結果を単峰性関数と多峰性関数に分けて図 6, 図 7 に示す。

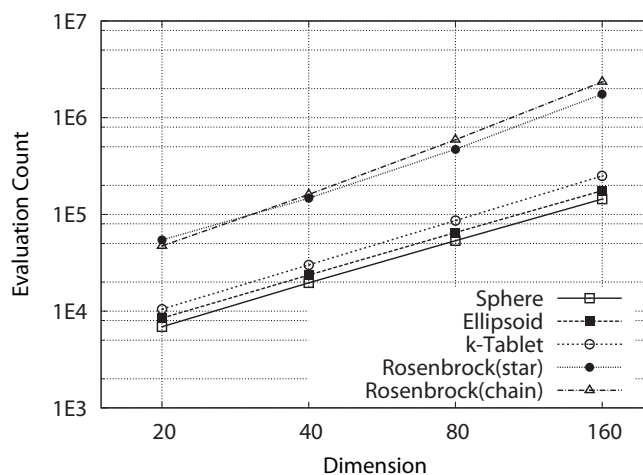


図 6 単峰性関数に対する REX^{star} の性能曲線

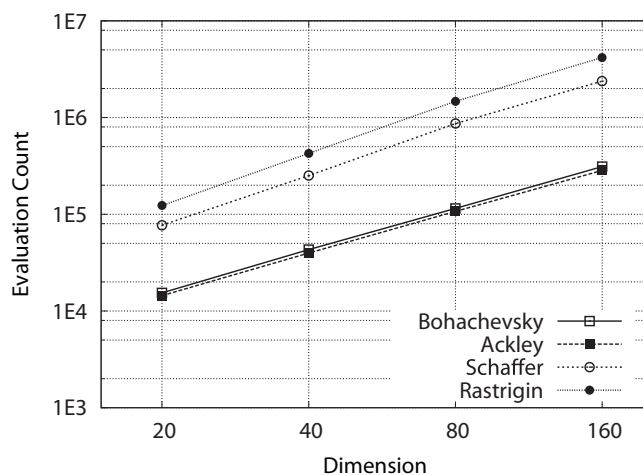


図 7 多峰性関数に対する REX^{star} の性能曲線

図 6, 図 7 から関数ごとの REX^{star} の性能曲線は両対数グラフ上でほぼ直線をなす。この傾きより、次元数が 2 倍に増えると、平均評価回数は Sphere, Ellipsoid, Bohachevsky, Ackley 関数では約 2.8 倍、 k -tablet 関数では約 2.9 倍、Schaffer 関数では約 3.2 倍、Rosenbrock 関数 (star 型) では約 3.3 倍、Rastrigin 関数では約 3.4 倍、Rosenbrock 関数 (chain 型) では約 3.7 倍増加する。

これらから、200 次元問題での評価回数を予測すると、Sphere, Ellipsoid, k -tablet, Bohachevsky, Ackley 関数

表 11 高次元問題における REX^{star} のパラメータ (JGG)

	$n = 20$	$n = 40$	$n = 80$	$n = 160$
関数	n_{pop}, t	n_{pop}, t	n_{pop}, t	n_{pop}, t
Sphe	$2n, 6$	$2n, 8$	$2n, 8$	$2n, 9$
Ellip	$2n, 7$	$2n, 7$	$2n, 8$	$2n, 12$
k -tab	$2n, 7$	$2n, 8$	$2n, 10$	$2n, 12$
R_{star}	$5n, 4$	$7n, 4$	$10n, 4$	$15n, 4$
R_{chai}	$2n, 7$	$2n, 7$	$2n, 7$	$2n, 7$
Boha	$4n, 6$	$4n, 7$	$4n, 7$	$4n, 9$
Ackl	$2n, 7$	$2n, 8$	$2n, 9$	$2n, 10$
Scha	$5n, 5$	$5n, 8$	$5n, 12$	$5n, 15$
Rast	$20n, 2.5$	$20n, 2.5$	$20n, 2.5$	$20n, 2.5$

では $2 \sim 5 \times 10^5$ 程度の評価回数で, Rosenbrock(star 型, chain 型), Schaffer, Rastrigin 関数では $3 \sim 6 \times 10^6$ 程度の評価回数で最適解に到達することが見込まれる.

6.5 初期化領域に対する頑健性の評価

性能評価のために用いているテスト関数の多くは超立方体で定義された実行可能領域の中央付近に最適解が存在する. しかし, 2.1.3 節で指摘したように, 実問題では実行可能領域全体を覆うように初期化領域を設定することは難しい. 初期化領域が実行可能領域の一部分で, その中に最適解が含まれていない状況は十分想定される. 初期化領域に対する頑健性は, 実数値 GA の評価においてこれまでほとんど考慮されてこなかった項目である.

初期化領域を狭く設定すると探索性能にどのような影響が生じるかを実験で調べる. 初期化領域の設定は 2 通り用意する. 設定 1 では, 実行可能領域全体を初期化領域とする. 設定 2 では, 次元ごとに定義域の左端 $1/4$ の大きさの区間に初期化領域を設定する. たとえば, 実行可能領域が $[-5.12, 5.12]^n$ の場合, 設定 2 では初期化領域を $[-5.12, -2.56]^n$ に設定する. この設定の場合, すべてのテスト関数の最適解は初期化領域の外に存在する.

実験 10 初期化領域に対する頑健性を評価するために, 20 次元問題を対象に, JGG のもとで REX^{star} と $REX(\mathcal{U}, n+1)$ の比較実験を行った. 結果を表 12 に示す. 表中, 関数名が表示されている行は設定 1 の場合, つぎの行は設定 2 の場合で, n_{pop}, n_c, n_{eval} を表す「倍率」が表示されている行は初期化領域を設定 1 から設定 2 に変化させたとき, n_{eval} が増加する割合を倍率で表す.

表 12 から, $REX(\mathcal{U}, n+1)$ は設定 1 から設定 2 に変化させると, もっとも簡単な Sphere 関数でも平均評価回数は 1.79 倍増加し, その他の関数では増加の程度が著しく, Rastrigin 関数では最適解の発見に失敗している. $REX(\mathcal{U}, n+1)$ は, 設定 1 の場合に比べて, 多様性維持のために集団サイズをかなり大きく取ることが必要であり, そのことが収束速度を遅らせる結果となっている.

一方, REX^{star} は, 設定 2 に変化させると, 多峰性の

表 12 初期化領域に対するロバスト性の評価 (20 次元/JGG)

	REX^{star}	$REX(\mathcal{U}, n+1)$	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
Sphe	$2n \ 2n \ 6.89 \times 10^3$	$6n \ 5n \ 2.50 \times 10^4$	28
初期 2	$2n \ 2n \ 7.83 \times 10^3$	$9n \ 5n \ 4.47 \times 10^4$	18
倍率	1.24	1.79	
Ellip	$2n \ 2n \ 8.46 \times 10^3$	$6n \ 5n \ 3.08 \times 10^4$	27
初期 2	$2n \ 2n \ 9.96 \times 10^3$	$15n \ 5n \ 1.07 \times 10^5$	9
倍率	1.18	3.47	
k -tab	$2n \ 2n \ 1.05 \times 10^4$	$7n \ 6n \ 4.80 \times 10^4$	22
初期 2	$2n \ 2n \ 1.17 \times 10^4$	$16n \ 6n \ 1.77 \times 10^5$	7
倍率	1.11	3.69	
R_{star}	$5n \ 3n \ 5.45 \times 10^4$	$15n \ 8n \ 1.57 \times 10^5$	35
初期 2	$6n \ 3n \ 6.52 \times 10^4$	$60n \ 8n \ 6.31 \times 10^5$	10
倍率	1.29	2.98	
R_{chai}	$2n \ 3n \ 4.72 \times 10^4$	—	—
初期 2	$3n \ 4n \ 6.90 \times 10^4$	—	—
倍率	1.46	—	
Boha	$4n \ 2n \ 1.54 \times 10^4$	$6n \ 5n \ 3.16 \times 10^4$	48
初期 2	$4n \ 3n \ 1.68 \times 10^4$	$60n \ 8n \ 3.96 \times 10^5$	4
倍率	1.09	12.53	
Ackl	$2n \ 3n \ 1.44 \times 10^4$	$6n \ 6n \ 5.54 \times 10^4$	26
初期 2	$2n \ 3n \ 1.59 \times 10^4$	$14n \ 6n \ 1.45 \times 10^5$	11
倍率	1.10	2.62	
Scha	$5n \ 3n \ 7.70 \times 10^4$	$9n \ 8n \ 2.29 \times 10^5$	34
初期 2	$10n \ 3n \ 1.62 \times 10^5$	$80n \ 8n \ 1.73 \times 10^6$	9
倍率	2.10	7.55	
Rast	$20n \ 3n \ 1.23 \times 10^5$	$22n \ 8n \ 2.20 \times 10^5$	56
初期 2	$80n \ 3n \ 5.26 \times 10^5$	—	—
倍率	2.39	—	

強い Schaffer, Rastrigin 関数では, 設定 1 に比べて平均評価回数は 2 倍余り増加しているが, その他の関数では 1.5 倍以下の増加率で最適解が求まっている. これらのことから, REX^{star} は, $REX(\mathcal{U}, n+1)$ に比べて, 初期化領域が狭い場合でもかなり頑健な性質を持つといえる.

初期化領域からかなり離れた所に最適解が存在する場合には, 集団全体が最適解の方向に移動する必要があり, 大域的降下方向へ子個体を選択的に生成する REX^{star} は合理的といえる. しかし, 集団全体の重心を大きく変化させなければならない状況では, 生存選択によって選ばれる子個体群は親個体群に比べて分布の幅がかなり狭まる. とくに Schaffer や Rastrigin 関数のように強い多峰性関数では初期収束が生じやすくなる. 初期収束を回避するために, REX^{star} でも集団サイズを大きく取る必要があり, そのことが評価回数の増加につながっている.

集団サイズを変えることなく重心移動が必要な状況に対処するためには, 子個体生成分布の中心を大域的降下

方向ヘシフトするだけでなく、子個体生成分布の広がりを意図的に拡大するメカニズムを組み込む必要がある。

6.6 REX^{star} に関する考察

§1 ステップサイズ t の効果

大域的降下方向を求めるために生成される鏡映個体群は親個体群の分散共分散行列を保存する。したがって、ステップサイズ t が小さければ、REX^{star} の子個体群はおおむね分散共分散行列を保存する。 t が大きくなるにつれて、子個体群と親個体群の分散共分散行列の乖離は大きくなるが、子個体生成分布を大域的降下方向にシフトする効果が生じて改善個体の生成確率が高まるとみられる。

実験 11 20 次元 Sphere 関数を対象に、JGGのもとで、 t を変化させたとき、安定収束するために必要な集団サイズ n_{pop} 、生成子個体数 n_c 、平均評価回数 n_{eval} を調べる実験を行った。結果を表 13 に示す。第 4 列は REX($\mathcal{U}, n+1$) に対する評価回数の百分率表示である。

表 13 REX^{star} のステップサイズ t の効果 (20 次元 Sphere/JGG)

REX ^{star}	n_{pop}, n_c	n_{eval}	%
$t=0.0$	$6n, 4n$	2.38×10^4	95
$t=0.5$	$6n, 4n$	2.16×10^4	86
$t=1.0$	$5n, 4n$	1.90×10^4	76
$t=2.0$	$4n, 3n$	1.41×10^4	56
$t=4.0$	$2n, 3n$	9.09×10^3	36
$t=6.0$	$2n, 2n$	6.89×10^3	28
$t=8.0$	$2n, 2n$	7.44×10^3	30
REX($\mathcal{U}, n+1$)	$6n, 5n$	2.50×10^4	100

$t=0$ のとき $n+1$ 個生成される鏡映個体は子個体として利用されるだけで、子個体生成分布の大域的降下方向への移動はないが、表 13 から REX^{star} は REX($\mathcal{U}, n+1$) に比べて n_c を $1n$ 分、平均評価回数を 5% 程度削減する効果がある。 t を大きくするにつれて、 n_{pop} と n_c の両方を削減できる効果が生じることがわかる。 t の増加につれて n_c を半分に以下に削減できるのは、上述したことであるが、改善個体の生成確率が高まるためと考えられる。

§2 ステップサイズ t と集団サイズ n_{pop} のトレードオフ

ステップサイズ t は大域的降下方向への外挿的探索の強さと集団の多様性の拡大を制御するパラメータと考えることができる。したがって、ステップサイズ t と集団サイズ n_{pop} の間にはトレードオフの関係が存在する。このトレードオフ関係は、表 11 から明らかなように、関数と次元数 n にかかなり依存する。次元数 n に対する t と n_{pop} の関係は、表 11 からつぎのパターンに分類できる。

1. n_{pop} を一定にしたまま、 t を大きくする

Sphere, Ellipsoid, k -tablet, Bohachevsky, Ackley, Schaffer 関数が該当する。これらの関数では、次元数 n の増加に対して、 n_{pop} を一定にして、 t を大きくすることによる外挿的探索の効果がプラスに働くと考えられる。

2. t を一定にしたまま、 n_{pop} を大きくする

Rosenbrock 関数 (star 型) が該当する。この関数には変数 x_1 と他の変数の間に依存関係が存在するため、変数 x_1 の動きに合わせて、他の変数がこれに追従する形で最適化が行われる。 t を大きくし過ぎると、稜構造の谷を踏み外しやすくなるので、次元数 n の増加に対しては n_{pop} を大きくして多様性を維持したほうが有利と考えられる。

3. n_{pop} と t は一定にしたままとする

Rosenbrock(chain 型) 関数と Rastrigin 関数が該当する。Rosenbrock(chain 型) 関数には隣り合う変数間に依存関係が存在する。このため、最適化は 1 変数ずつ順に行われるので、次元数の増加に対して集団サイズもステップサイズも変化させないほうがよいと考えられる。Rastrigin 関数は有力な局所最適解が多数存在するため、ステップサイズ t を大きく設定しないほうがよいと考えられる。

§3 子個体生成分布の中心の決め方

REX^{star} では、図 8 に示すように、子個体生成分布の中心を x^g と $x^t = x^g + t(x^b - x^g)$ を対角頂点とする超直方体内に一様分布によって決めている。すなわち、拡張率 $\alpha = 1$ の交叉 BLX- α によって中心を決めている。

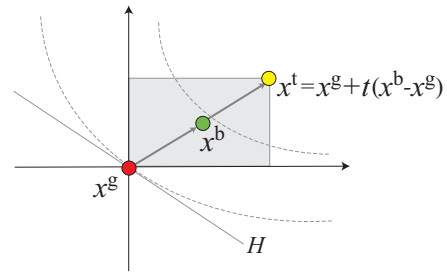


図 8 REX^{star} の子個体生成分布の中心がおかれる領域

x^g を原点にみれば、 x^g を通る超切平面 H によって定まる降下方向の広がりが $1/2$ の大きさであるのに対し、大域的降下方向が存在する象限の広がりは $(1/2)^n$ と狭く、それが改善子個体の生成確率を高めると考えられる。

しかし、図 8 から明らかなように、子個体生成分布の中心を式 (11) により決めることは、座標軸の回転に対する依存性の問題を惹起する。Rosenbrock 関数 (star 型) の探索で親個体群が稜構造上に存在するとき、大域的降下方向と稜線の方法は高精度で一致するため、大域的降下方向に沿って探索を進めたほうがよいと思われる。

式 (11) の変形として、次式による子個体生成を考える。

$$x^c = x^g + \xi^t(x^b - x^g) + \sum_{i=1}^{n+1} \xi^i(x^i - x^g) \quad (14)$$

式 (14) は、図 8 の線分 $x^g x^t$ 上に子個体生成分布の中心を生成する。この子個体生成法を REX_{line} とよぶ。

実験 12 20 次元 Rosenbrock 関数 (star 型) を対象に、JGGのもとで、REX_{line} と REX^{star} の性能を比較する実験を行った。結果を表 14 に示す。予想通り、REX_{line} の

表 14 REX_{line} と REX^{star} の比較 (20 次元/JGG)

	REX _{line}	REX ^{star}	
関数	n_{pop}, n_c, n_{eval}	n_{pop}, n_c, n_{eval}	%
R _{star}	7n 4n 4.20×10 ⁴	5n 3n 5.45×10 ⁴	77

ほうが平均評価回数が少ない。しかし、その他の関数では、REX^{star} のほうが REX_{line} の性能をかなり上回る。全体的にみて、REX^{star} のほうが性能が優れているのは、子個体生成分布を大域的降下方向だけでなく、その周辺にも拡大することが効果的に働くためと考えられる。

7. お わ り に

研究成果を要約する。実数値 GA の従来の取り組みを概観した後、世代交代モデル MGG、多親交叉 SPX と UNDX- n 、および多親交叉の設計指針である機能分担仮説と統計量遺伝の問題点を指摘した。MGG についてはエリート保存とルーレット選択が問題であることを指摘し、親子間で完全な世代交代を行う JGG を提案した。SPX については子個体生成分布の非対称性が問題であることを指摘し、鏡映個体を用いた中心対称多面体を用いて子個体を一様に生成する e -SPX を提案した。UNDX- n については主親の選び方によるバイアスが問題であることを指摘し、すべての親個体を同等に扱う e -UNDX を提案した。 e -UNDX の一般化の枠組みとして、任意の確率分布 φ と任意個の親個体を用いる REX($\varphi, n+k$) を提案した。 e -SPX, e -UNDX, REX($\varphi, n+k$) はすべて統計量遺伝にしたがう交叉である。しかし、これらの交叉では集団や親個体群が最適解を覆っていない状況に対処するためには不十分であることを指摘し、子個体生成分布の中心を親個体の重心から大域的降下方向に移動して子個体を生成する REX^{star} を提案した。REX^{star} は集団サイズと生成子個体数を大幅に引き下げることができるため性能が著しく向上すること、そして初期化領域の取り方に対して頑健であることを実験により示した。

REX^{star} の最高性能を引き出すためには、ステップサイズ、集団サイズ、生成子個体数を調節する必要があるが、その作業は容易とはいえない。REX^{star} の使い勝手を良くするために、パラメータ n_{pop}, t などを適応的に調節するメカニズムを開発することが今後の課題である。

Rosenbrock 関数における稜構造上の探索や最適解が初期化領域の外に存在するときの探索では、最適解の方向に集団の重心を移動させる必要がある。子個体生成分布を大域的降下方向にシフトする REX^{star} は、集団の重心移動時への対処法のひとつと解釈することができる。集団の重心移動時への別の対処法として、CMA-ES における SSA (Step Size Adaptation)[Hansen 04] のように、集団の重心移動時に子個体生成分布を拡大する手法がある。実数値 GA は分散共分散行列を非明示的に扱うことにより、計算量を $\mathcal{O}(n^2)$ に抑えている。この利点を損なわず

に、REX^{star} に SSA 的な機構を組み込むことができれば、性能の更なる向上が期待できると思われる。

謝 辞

本論文の執筆に際して、ご協力いただいた秋本洋平、羽佐田理恵、原田健、佐久間淳、小野功の各氏に感謝する。また、本論文に対して、有益なコメントをお寄せいただいた筒井茂義、大林茂、廣安知之の各氏に感謝する。

付 録 A

REX($\varphi, n+k$) におけるパラメータの導出

REX($\varphi, n+k$) で用いる確率分布 φ のパラメータが平均 $\mu = 0$ 、分散 $\sigma^2 = 1/(n+k)$ のとき、統計量遺伝を満たすことを以下に示す。

REX($\varphi, n+k$) は式 (15) ~ (17) で子個体を生成する。

$$\mathbf{x}^c = \mathbf{x}^g + \sum_{m=1}^{n+k} \xi^m (\mathbf{x}^m - \mathbf{x}^g) \quad (15)$$

$$\mathbf{x}^g = \frac{1}{n+k} \sum_{m=1}^{n+k} \mathbf{x}^m \quad (16)$$

$$\xi^m \sim \varphi(0, \sigma^2); m = 1, \dots, n+k \quad (17)$$

親個体群と子個体群の平均を、それぞれ $\bar{\mathbf{x}}^p, \bar{\mathbf{x}}^c$ で表す。子個体の期待値 $\langle \mathbf{x}^c \rangle$ は、

$$\langle \mathbf{x}^c \rangle = \langle \mathbf{x}^g \rangle + \sum_{m=1}^{n+k} \langle \xi^m \rangle \langle \mathbf{x}^m - \mathbf{x}^g \rangle = \bar{\mathbf{x}}^p$$

であり、親個体の平均と等しいので、 $\bar{\mathbf{x}}^c = \bar{\mathbf{x}}^p$ 。したがって、子個体の分散共分散行列の要素 γ_{ij}^c は、

$$\gamma_{ij}^c = \langle (x_i^c - \bar{x}_i^p)(x_j^c - \bar{x}_j^p) \rangle \quad (18)$$

で表される。式 (15) に式 (16) を代入、整理すると

$$\mathbf{x}^c = \sum_{m=1}^{n+k} \left\{ \frac{1}{n+k} \left(1 - \sum_{l=1}^{n+k} \xi^l \right) + \xi^m \right\} \mathbf{x}^m \quad (19)$$

ここで、

$$t_m = \frac{1}{n+k} \left(1 - \sum_{l=1}^{n+k} \xi^l \right) + \xi^m \quad (20)$$

とおくと、式 (19) は、

$$\mathbf{x}^c = \sum_{m=1}^{n+k} t_m \mathbf{x}^m \quad (21)$$

となり、式 (20) より、次式が成り立つ。

$$\sum_{m=1}^{n+k} t_m = 1 \quad (22)$$

式 (18) の $\langle \rangle$ 内の $(x_i^c - \bar{x}_i^p)$ は式 (21), (22) を用いると,

$$\begin{aligned} x_i^c - \bar{x}_i^p &= \left(\sum_{m=1}^{n+k} t_m x_i^m \right) - \left(\sum_{m=1}^{n+k} t_m \bar{x}_i^p \right) \\ &= \sum_{m=1}^{n+k} t_m (x_i^m - \bar{x}_i^p) \end{aligned}$$

と変形できるので, 式 (18) は,

$$\gamma_{ij}^c = \left\langle \left(\sum_{m=1}^{n+k} t_m (x_i^m - \bar{x}_i^p) \right) \left(\sum_{m=1}^{n+k} t_m (x_j^m - \bar{x}_j^p) \right) \right\rangle \quad (23)$$

と書き換えられる. 親の選び方は独立なので, $a \neq b$ のとき $\langle (x_i^a - \bar{x}_i^p)(x_i^b - \bar{x}_i^p) \rangle = 0$ であるから, 式 (23) は

$$\gamma_{ij}^c = \left\langle \sum_{m=1}^{n+k} t_m^2 (x_i^m - \bar{x}_i^p)(x_j^m - \bar{x}_j^p) \right\rangle \quad (24)$$

となる. 親の選び方と乱数が独立であるから, 式 (24) は

$$\gamma_{ij}^c = \left\langle \sum_{m=1}^{n+k} t_m^2 \right\rangle \langle (x_i^m - \bar{x}_i^p)(x_j^m - \bar{x}_j^p) \rangle = \left\langle \sum_{m=1}^{n+k} t_m^2 \right\rangle \gamma_{ij}^p$$

ここで, γ_{ij}^p は親個体の分散共分散行列の要素を表す. したがって, $\langle \sum_{m=1}^{n+k} t_m^2 \rangle = 1$ のとき, $\gamma_{ij}^c = \gamma_{ij}^p$ となる.

$$\begin{aligned} \left\langle \sum_{m=1}^{n+k} t_m^2 \right\rangle &= \left\langle \sum_{m=1}^{n+k} \left\{ \frac{1}{n+k} \left(1 - \sum_{l=1}^{n+k} \xi^l \right) + \xi^m \right\}^2 \right\rangle \\ &= \left\langle \sum_{m=1}^{n+k} \frac{1}{(n+k)^2} \left(1 - \sum_{l=1}^{n+k} \xi^l \right)^2 \right\rangle \\ &\quad + \left\langle \sum_{m=1}^{n+k} \frac{2}{(n+k)} \left(1 - \sum_{l=1}^{n+k} \xi^l \right) \xi^m \right\rangle \\ &\quad + \left\langle \sum_{m=1}^{n+k} (\xi^m)^2 \right\rangle \\ &= \frac{1}{n+k} (1 + (n+k)\sigma^2) - 2\sigma_\xi^2 + (n+k)\sigma^2 \end{aligned}$$

であるから, $\sigma^2 = 1/(n+k)$ のとき, $\gamma_{ij}^c = \gamma_{ij}^p$ となり, 統計量遺伝が満たされる.

付 録 B

中心対称多面体

n 次元空間の点 x^1, \dots, x^{n+1} を頂点とする単体 σ を考える. σ の重心を x^g , σ の頂点 x^i の x^g に対する鏡映点を \underline{x}^i で表す. $\underline{x}^1, \dots, \underline{x}^{n+1}$ を頂点とする単体を $\underline{\sigma}$ で表す. x^i と $\underline{\sigma}$ に属する \underline{x}^i 以外の頂点からなる単体を σ_{x^i} で表す. σ_{x^i} の重心を $x_{\sigma_{x^i}}^g$ で表す. $n+2$ 個の単体 $\sigma, \sigma_{x^1}, \dots, \sigma_{x^{n+1}}$ の和集合で定義される多面体を P で表す. P はつぎの性質を持つ.

[性質 1] P は中心対称である

[性質 2] x^g を通り, σ_{x^i} と $\sigma_{\underline{x}^i}$ を分離する超平面 H が存在する

[性質 3] x^g を通る任意の超平面 H によって分離される σ_{x^i} と $\sigma_{\underline{x}^i}$ が存在する

[性質 4] $x_{\sigma_{x^i}}^g$ は直線 $x^i x^g$ 上に存在する

[性質 5] $\|x_{\sigma_{x^i}}^g - x^g\| / \|x^i - x^g\| = 2/(n+1)$ である

[性質 6] P, σ の体積をそれぞれ $V(P), V(\sigma)$ とするとき, $V(P) = nV(\sigma)$ である

図 9 に 2 次元の場合の中心対称多面体の例を示す. 図 9 の点 x^b を単体 $\sigma_{x^1} = x^1 x^2 x^3$ の重心 $x_{\sigma_{x^1}}^g$ におけば, 性質 5 から $\|x^b - x^g\| / \|x^1 - x^g\| = 2/(n+1)$ である. したがって, $x^b - x^g$ が大域的降下方向であれば, $t = (n+1)/2$ のとき $x^t = x^g + t(x^b - x^g)$ は x^1 に一致する. たとえば, $n = 20$ であれば, $t = 10.5$ である. 20 次元問題において大域的降下方向へのステップサイズを $t = 5 \sim 6$ に設定するとき, REX^{star} における子個体生成分布の中心は線分 $x^g x^1$ の中点辺りにおかれる.

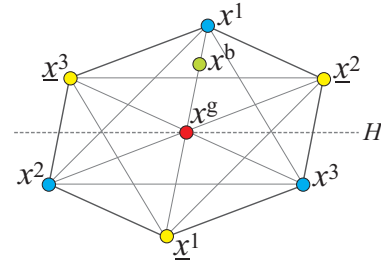


図 9 中心対称多面体の例

◇ 参 考 文 献 ◇

- [Ackley 87] Ackley, D. H.: An Empirical Study of Bit Vector Function Optimization, *Genetic Algorithms and Simulated Annealing*, pp. 170-204 (1987)
- [Boese 95] Boese, K.: Cost versus Distance in the traveling Salesman Problem, *Tech. Rep. TR-950018*, UCLA CS Department (1995)
- [Davis 90] Davis, L.: *The Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York (1990)
- [Deb 01] Deb, K.: *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Chichester, UK (2001)
- [Eshelman 90] Eshelman, L. J.: The CHC Adaptive Search Algorithm: How to have Safe search When Engaging in Non-traditional Genetic Recombination, *Foundations of Genetic Algorithms*, pp. 265-283 (1990)
- [Eshelman 93] Eshelman, L. J. and Schaffer, J. D.: Real-coded Genetic Algorithms and Interval- Schemata, *Foundations of Genetic Algorithms 2*, pp. 187-202 (1993)
- [Goldberg 89] Goldberg, D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Company Inc., (1989)

- [Hansen 04] Hansen, N. and Kern, S.: Evaluating the CMA Evolution Strategy on Multimodal Test Functions, *Proc. of 8th Int. Conf. on Parallel Problem Solving from Nature (PPSN VIII)*, pp. 282-291 (2004)
- [Hillermeier 01] Hillermeier, C.: Nonlinear Multiobjective Optimization: A Generalized Homotopy Approach, *Vol. 25 of Int. Series Numerical Mathematics*, Birkhäuser Verlag (2001)
- [Knowles 04] Knowles, J. D. and Corne, D. W.: On Memetic Algorithms for Multiobjective Optimization: Issues, Methods and Prospects. In *Recent Advances in Memetic Algorithms*, pp. 313-352, Springer (2004)
- [Larranaga 00] Larranaga, P., Etxeberria, R., Lozano, J. A., and Pena, J. M.: Optimization in Continuous Domains by Learning and Simulation of Gaussian Networks, *Proc. of Genetic and Evolutionary Computation Conf. (GECCO 2000)*, pp. 201-204 (2000)
- [Mühlenbein 93] Mühlenbein, H. and Schlierkamp-Voosen, D.: Predictive Models for the Breeder Genetic Algorithm I; Continuous Parameter Optimization, *Evolutionary Computation*, Vol. 1, pp. 25-49 (1993)
- [Ono 97] Ono, I. and Kobayashi, S.: A Real-coded Genetic Algorithms for Function Optimization using Unimodal Normal Distribution Crossover, *Proc. of 7th Int. Conf. on Genetic Algorithms*, pp. 246-253 (1997)
- [Ono 00] Ono, I., Kobayashi, S., and Yoshida, K.: Optimal Lens Design by Real-coded Genetic Algorithms using UNDX, *Computer Methods in Applied Mechanics and Engineering*, No. 186, pp. 1146-1155 (2000)
- [Syswerda 90] Syswerda, G.: A Study of Reproduction in Generational and Steady-State Genetic Algorithms, *Foundations of genetic Algorithms*, pp. 94-101 (1990)
- [Tsutsui 99] Tsutsui, S., Yamamura, M. and Higuchi, T.: Multi-parent Recombination with Simplex Crossover in Real Coded Genetic Algorithms, *Proc. of Genetic and Evolutionary Computation Conf. (GECCO 1999)*, pp. 657-664 (1999)
- [Thieryns 94] Thieryns, D. and Goldberg, D. E.: Elitist Recombination: An Integrated Selection Recombination GA, *Proc. of IEEE Conf. on Evolutionary Computation (CEC 1999)*, pp. 508-512 (1994)
- [Wright 91] Wright, A.: Genetic Algorithms for Real Parameter Optimization, *Foundations of Genetic Algorithms*, pp. 205-218 (1991)
- [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 ~ , SICE 第 19 回自律分散システムシンポジウム資料, pp. 341-346 (2007)
- [池田 02] 池田心, 小林重信: GA の探索における UV 現象と UV 構造仮説, 人工知能学会論文誌, Vol. 17, No. 3, pp. 239-246 (2002)
- [小野 99] 小野功, 佐藤浩, 小林重信: 単峰性正規分布交叉 UNDX を用いた実数値 GA による関数最適化, 人工知能学会誌, Vol. 14, No. 6, pp. 1146-1155 (1999)
- [喜多 99a] 喜多一, 小野功, 小林重信: 実数値 GA のための正規分布交叉に関する理論的考察, 計測自動制御学会論文誌, Vol. 35, No. 11, pp. 1333-1339 (1999)
- [喜多 99b] 喜多一, 山村雅幸: 機能分担仮説に基づく実数値 GA の設計指針, 計測と制御, Vol. 38, No. 10, pp. 612-617 (1999)
- [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誌, Vol. 36, No. 10, pp. 875-883 (2000)
- [木村 00] 木村周平, 小野功, 喜多一, 小林重信: 交叉の設計指針に基づく UNDX の拡張: ENDX の提案と評価, 計測自動制御学会論文誌, Vol. 36, No. 10, pp. 875-883 (2000)
- [小島 01] 小島政和, 土谷隆, 水野真治, 矢部博: 内点法, 朝倉書店 (2001)
- [小林 03] 小林重信: GA の研究に取り組んで十余年, 人工知能学会誌, Vol. 18, No. 4, pp. 439-451 (2003)
- [小林 07] 小林重信: 実数値 GA のブレークスルーに向けて, 進化計算シンポジウム 2007 講演論文集, pp. 1-10 (2007)
- [今野 78] 今野浩, 山下浩: 非線形計画法, 日科技連 (1978)
- [佐藤 97] 佐藤浩, 小野功, 小林重信: 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)
- [高橋 01] 高橋治, 木村周平, 小林重信: 交叉的突然変異による適応的近傍探索, 人工知能学会論文誌, Vol. 16, No. 2, pp. 175-184 (2001)
- [原田 06] 原田健, 佐久間淳, 小野功, 小林重信: 多目的関数最適化における GA と局所探索の組み合わせ: GA then LS の推奨, 人工知能学会論文誌, Vol. 21, No. 6, pp. 482-492 (2006)
- [原田 06b] 原田健, 佐久間淳, 小野功, 小林重信: 多目的関数最適化のための局所探索: パレート降下法, 人工知能学会論文誌, Vol. 21, No. 4, pp. 350-360 (2006)
- [原田 07] 原田健, 佐久間淳, 小野功, 小林重信: 関数最適化のための制約対処法: パレート降下修正オペレータ, 人工知能学会論文誌, Vol. 22, No. 4, pp. 364-374 (2007)
- [樋口 01] 樋口隆英, 筒井茂義, 山村雅幸: 実数値 GA におけるシンプレックス交叉の提案, 人工知能学会論文誌, Vol. 16, No. 1, pp. 147-155 (2001)
- [森 96] 森直樹, 吉田潤二, 喜多一, 西川禎一: 遺伝的アルゴリズムにおける熱力学的選択ルールの提案, システム制御情報学会論文誌, Vol. 9, No. 2, pp. 734-744 (1997)
- [山村 98] 山村雅幸: モンテカルロ法による遺伝的オペレータの解析, SICE システムシンポジウム/情報合同シンポジウム, pp. 157-162 (1998)

〔担当委員: 筒井 茂義〕

2008 年 10 月 15 日 受理

著者紹介



小林 重信 (正会員)

1974 年 3 月東京工業大学理工学研究科経営工学専攻博士課程修了。工学博士。同年 4 月同大学助手。同大学助教授を経て, 1990 年 8 月教授となり現在に至る。創発システム論, 進化計算, 強化学習などの研究に従事。