

分散遺伝的アルゴリズムの性能におよぼす 交叉法とコーディング法の影響

The Effect of Crossover Operators and Bit Representation
for Distributed Genetic Algorithms

吉田 純一 (知的システムデザイン研究室)

Jun-ichi YOSHIDA (Intelligent Systems Design Laboratory)

Abstract The crossover is the primary means of search in genetic algorithms (GAs). However, the effect of crossover depends on the choice of crossover operators. In this paper, we investigate the effect of crossover operators and codings in distributed GAs (DGAs). From the experimental results, we found that 1- and 2-point crossovers with the Gray coding is the best choice for DGAs with mutation. This is the same result as the one in the single population GA (SGA). But, the effects of the crossover and the coding in DGAs are different from those in SGA.

1 はじめに

分散遺伝的アルゴリズム (Distributed Genetic Algorithms: DGA) では、母集団を複数のサブ母集団に分割し、そのサブ母集団ごとに遺伝的操作を行い、一定期間ごとに異なるサブ母集団間で移住を行う。分散 GA では単一母集団の GA と比較して高品質の解が得られると報告されている。これは、母集団の分割によって解の多様性が保持され、良好なスキーマの移住と交叉によって、よりよい解が生成されるためである [1]。

一般に GA では、個体間の情報交換の役割を持つ交叉の働きが重要になる。単一母集団の GA における交叉法やコーディング法に関する報告は多い [2] が、分散 GA に関するものは少ない。本研究では、3つの代表的なテスト関数を用い、分散 GA の解探索性能におよぼす交叉法とコーディング法の影響を検討した。

2 交叉法の影響

交叉の概念は、2つの個体間で染色体を組替えることによって、両親の持つ優れた部分形質を子に継承しようというものである。しかしながら実装する交叉法の違いによって、その働きは異なる。以下の議論は、本研究で用いた連続関数の最大化問題において、連続変数を符号化した場合のものである。

1点交叉 (1X) や2点交叉 (2X) では交叉点の数が少ないために親個体のスキーマの大半がそのまま子に伝わる。すなわち、1X ならば 10 設計変数のうちの少なくとも

も 9 個 (2X ならば 8 個) はそのまま子に引き継がれる。したがって、1X や 2X を用いた場合には初期母集団が持つ部分解の個体間での交換が主となる。このため、母集団サイズが小さく有効な部分解が少ない場合には局所解に陥りやすい。

一方、一様交叉 (UX) ではランダムに生成したマスクによって交叉点の数が決定され、平均すると染色体長 L に対し $\frac{L}{2}$ 個の交叉点を持つ。一般に、交叉点が多い交叉はスキーマを破壊する性格が強い。特に、母集団に多様性のある進化序盤でこの傾向が強い。しかしながら、これによって、大域的な探索可能になるという利点もある [2]。

3 コーディング法の影響

染色体に実数値を符号化する方法としては、代表的なものとしてバイナリコーディング (以下バイナリと称す) とグレイコーディング (以下グレイと称す) がある。バイナリでは隣り合う値に対応する符号のハミング距離が一定でないのに対し、グレイでは、隣り合う符号間のハミング距離が常に 1 となるよう符号化される。図 1 に示すように、バイナリではすべての変化が等確率で起こるのに対し、グレイでは、確率は変化量の絶対値に反比例し、大きな変化ほど起こりにくい。このため、一般にグレイの方が探索の効率が良なる。特に破壊的な性質の強いオペレータでその傾向は強まる [3]。

⁰本発表は、第 59 回情報処理学会全国大会の講演論文 (分冊 2 p.139-140) に加筆したものである。

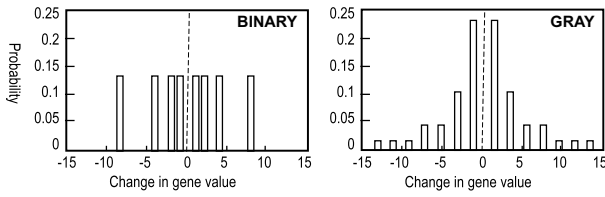


図 1: コーディング法と確率分布

4 実験

分散 GA における交叉法とコーディング法の影響を確認するために、10 次元の多峰性関数である 3 つのテスト関数 [4] を用いて実験を行った。

本研究で対象とした関数は以下の 3 つの 10 次元多峰性関数である (1) 式で示した Rastrigin 関数および (2) 式で示した Schwefel 関数は設計変数間に依存関係はない (3) 式で示した Griewank 関数は大域的には単峰関数のような形状を持ち、設計変数間に中程度の依存関係を有する。いずれも大域的最適解は 0 である。

$$f_{Rastrigin} = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) \quad (1)$$

定義域: $-5.12 < x_i \leq 5.12, n = 10$

$$f_{Schwefel} = \sum_{i=1}^n -x_i^2 \sin(\sqrt{|x_i|}) \quad (2)$$

定義域: $-512 < x_i \leq 512, n = 10$

$$f_{Griewank} = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \left(\cos\left(\frac{x_i}{\sqrt{i}}\right) \right) \quad (3)$$

定義域: $-512 < x_i \leq 512, n = 10$

分散 GA の性能に上で述べた交叉法やコーディング法がどのように影響するかを考える。ここでは 1X, 2X および UX の 3 種類の交叉法、およびグレイとバイナリの 2 種類のコーディング法の組み合わせからなる 6 通りの設定を考えた。すべての実験において、最大世代数を 1000、染色体のビット長 L は 100bit (1 設計変数 10bit)、母集団サイズは 450 または 900、交叉率は 0.6、突然変異率は 0 または $1/L$ とした。分散 GA の場合には、サブ母集団数を 9、移住間隔は 10 世代、移住率は 0.3 とした。また、選択オペレータはルーレット選択であり、エリート保存戦略を用いた。次節以降に示す実験結果は 20 試行の平均値である。

5 実験結果と考察

母集団サイズ 450 の 1000 世代での適合度を図 2 に示した。突然変異を行う場合と行わない場合とで結果に大きな差がある。

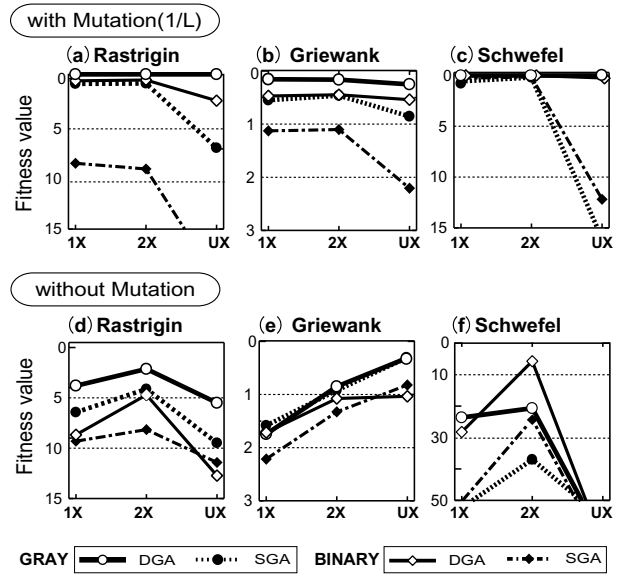


図 2: 1000 世代での適合度

5.1 突然変異を行う場合

適合度の観点からはバイナリよりもグレイ、単一母集団 GA よりも分散 GA の方がすべての関数においてもっとも良い成績を示した (図 2 上段)。このことから分散 GA においても、グレイの優位性は失われないことが分かる。また、全般的に分散 GA の方が成績が良いのは、母集団を複数に分割したことによって大域的な探索が可能になったためであると考えられる。グレイと分散 GA の組み合わせの中での交叉法の違いに注目するために、最適解が得られるのに要した世代数¹を表 1 に示した。

表 1: 最適解に到達した世代数

	1X	2X	UX
Rastrigin	471	520	923
Griewank	(-0.156)	(-0.164)	(-0.252)
Schwefel	510	455	793

表 1 が示すように、1X および 2X が良い成績を示している。2 節で述べたように、1X および 2X がスキーマを保存するのにに対し、UX はスキーマを破壊する。このため 1X・2X と比べて UX では前の世代の情報が有効に利用できず、探索の効率が悪い。分散 GA においては、移住してきた部分解を破壊してしまうため移住の効果は得られない。それにも拘わらず、分散 GA では UX の性能の劣化はあまり大きくなかった。これは、少ない個体数による多様性の減少が UX の破壊的な性格を弱めたこと、また、移住による部分解の交換ではなく複数の母集団での複数試行による解の高品質化の作用のみが現れたためと考えられる。

¹Griewank 関数ではいずれも最適解が得られなかったため、1000 世代での適合度を示した

5.2 突然変異を行わない場合

突然変異を行わない場合には各関数ごとに特有の傾向を示した(図2下段). Rastrigin 関数では, 突然変異を行う場合と似た傾向を示したが, Griewank 関数では, UX がよい結果を示した(図2(e)). Schwefel 関数では, グレイコーディングよりもバイナリコーディングの方が良い結果を示した(図2(f)). 突然変異を行わない場合は最適な交叉法・コーディング法は問題によって異なる. このことから, 交叉法やコーディングも突然変異的な役割を持つことが分かる.

一方, 突然変異を行わない場合の結果で特徴的なのは, バイナリの UX は分散化することで単一母集団 GA よりも性能が悪化したことである(図2(d,e)). 一般に分散 GA は単一母集団 GA よりも性能が優れているにも拘わらず, このような結果となった理由は次のように考えられる. バイナリと UX の組み合わせは, 大域的探索には優れているが, 情報の利用がなされないためにランダムサーチに近くなる. したがって, 分散による大域的な探索効果が有効に作用せず, 時には性能の悪化につながっていると考えられる.

6 おわりに

代表的なテスト関数を用いて, 分散 GA の性能におよぼす交叉法とコーディング法の影響について検討した. 分散 GA において最適な交叉法とコーディング法は, 突然変異を行う場合ではグレイコーディングと $1X \cdot 2X$ であることを示した. これは単一母集団 GA の場合と同様である. しかしながら, それぞれの交叉法とコーディング法がおよぼす影響は単一母集団の場合と分散の場合では異なることも確認された. 設計変数間に限定した交叉については今後の課題である.

参考文献

- [1] 三木, 廣安, 金子. 分散母集団 ga における解探索能力. 人工知能学会全国大会, 1999.
- [2] Kenneth A. De Jong and William M. Spears. A formal analysis of the role of multi-point crossover in genetic algorithms. *Annals of Mathematics and Artificial Intelligence*, 1992.
- [3] Harry Girelewski. The nature of mutation in genetic algorithms. *International Conference on Genetic Algorithms*, 1995.
- [4] S. Rana D. Whitley, K. Mathias and J. Dzubera. Building better test functions. *International Conference on Genetic Algorithms*, 1995.