GA の探索過程を把握し対象問題の特徴を捉えるシステムの構築

Classed Problems by Grasping a Search Process of Genetic Algorithms

赤塚 浩太（知的システムデザイン研究室）
Kouta AKATSUKA (Intelligent Systems Design Laboratory)

Abstract: In Genetic Algorithms (GAs), design variables of candidate solutions are encoded to binary strings. Therefore, the landscape where GAs are searching may be different from the landscape that we recognize. In this paper, we propose a method that help us to understand the landscape of the problems that GAs are searching. By applying the proposed methods to some test functions and designing truss structure problems, it was found that we can recognize the difficulty and the characteristics of the problems by this method.

1　はじめに

最適化問題を遺伝的アルゴリズム (Genetic Algorithms: GA) で探索する際，その問題の最適解を GA によって求めやすいか否かということは非常に重要な問題である．しかし，GA では通常設計変数の値をコード化して探索を行うため，我々が把握している対象問題の解空間とはまったく異なった空間を探索していると考えられる．それに対して，GA が探索している解空間を把握するための試みとして，内藤 [1] の研究などがあげられる．しかし，これらの研究は離散問題を主にその対象としているため，解の精度が大きく影響する連続問題には不向きである．そこで本研究では連続問題を対象とする手法を検討する．

2　ハミング距離，適合度，頻度によるランドスケープの表示

GA が現在探索を行っている空間を把握するため，ハミング距離，適合度，頻度によりランドスケープを表示する手法を提案する．また，対象問題全体のランドスケープでは精度が粗くなり探索状況の把握ができないため，その世代における GA のエリートと真の解の間の空間を表示するに個体を生成しその分布を観察する．ただし，コード化された空間を把握するため，エリートのビットを 1 ビットづつ真の解に近づける方法で，個体を生成する．

GA では，ハミング距離が現在のエリートと近い点から徐々に真の解と近い点に個体が生成されると考えられるが，この際適合度値が現在のエリートより悪い個体は生き残る可能性が少ない．そのため，現在のエリートと真の解の間に谷がある問題では，GA による探索では真の解に到達しにくいといえる．

3　ハミング距離，適合度，頻度による探索過程の把握

これまで，GA 探索中の個体情報を用いて探索の様子を把握する際には，対象問題の設計変数を 2 次元に固定し，コード化前の状態での設計変数値と関数値を元に，個体が関数上にどのように分布しているかを把握するものが大半であった．しかし，GA では探索の際にコード化を行うため，設計変数値と関数値では個体の散らばりを正確に把握することは困難であり，ハミング距離などを用いて多様性を測る研究も多く行われてきた．そこで，本研究では個体の分布をハミング距離，位相，関数値の 3 軸を用いて把握する事を試みる．

4　数値実験

4.1　対象問題

提案手法の性能を検証するため，数値実験を Rastrigin 関数，Rosenbrock 関数で行った．

$$f(x_1,...,x_n) = 10n + \sum_{i=1}^{n}[x_i^2 - 10cos(2\pi x_i)] \quad (1)$$

$$-5.12 \le x_i < 5.12$$

$$f(x_1,...,x_n) = 100\sum_{i=2}^{n}(x_{i-1} - x_i^2)^2 + (1 - x_i)^2 \quad (2)$$

$$-2.048 \le x_i < 2.048$$

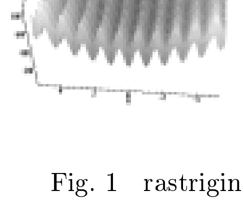


Fig. 1　rastrigin

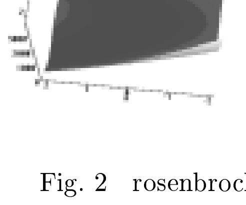


Fig. 2　rosenbrock

Rastrigin 関数は式 (1) で表される関数で，2 設計変数の場合の外観は Fig. 1 であり，GA 向きの関数とされ比較的簡単に解を求めることができるとされている．一方，Rosenbrock 関数は式 (2) で表される関数で，2 設計変数の場合の外観は Fig. 2 であり，GA で解くのは非常に困難とされている．

4.2 ランドスケープの表示

Fig. 3，Fig. 4 に GA で 100 世代探索を行った後に提案手法を適用した結果を示す．なお，グラフ中 x 軸に真の解からの Hamming 距離を，y 軸に適合度を，z 軸に頻度を用いている．

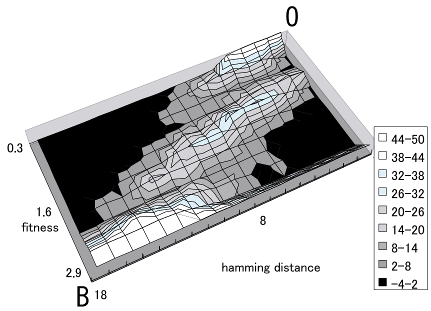


Fig. 3 rastrigin

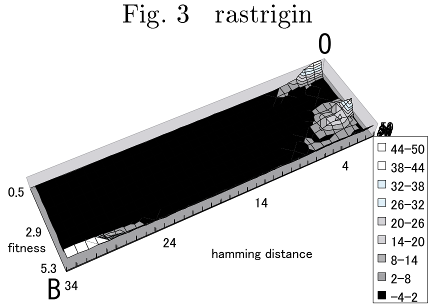
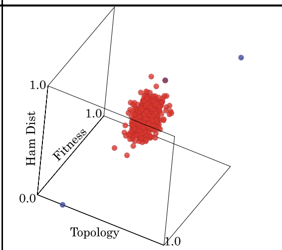
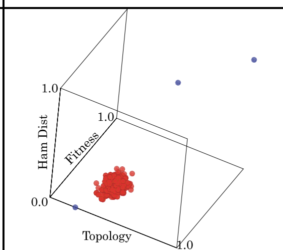
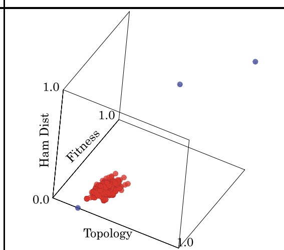
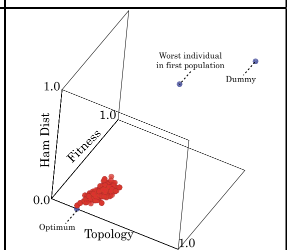
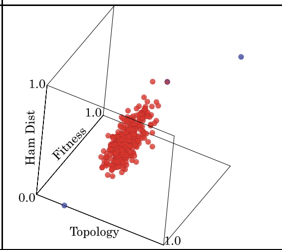
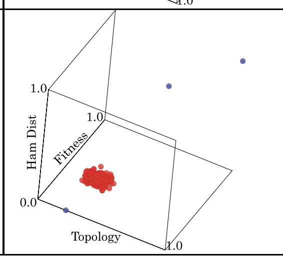
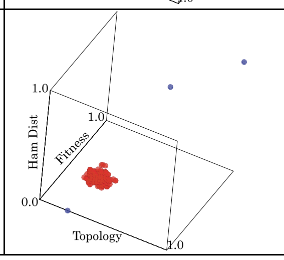
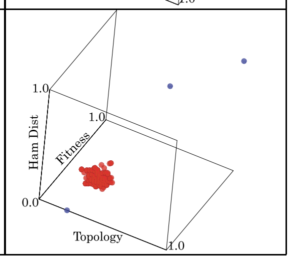


Fig. 4 rosenbrock

Table 1 Result of Rastrigin/Rosenbrock

Generation	0	100	250	400
Rastrigin				
Rosenbrock				

また，左手前 B 点が各世代でのエリート個体，右奥 O 点が真の解となる．Rastrigin 関数ではエリートから真の解まで個体が連続して分布しているのに対し，Rosenbrock 関数ではエリートと真の解の間に個体が分布していない領域がある．つまり，Rastrigin 関数は GA による探索が比較的容易であり，Rosenbrock 関数は探索が困難であると考えられる結果が得られた．

4.3 探索過程の把握

実験結果を Table 1 に示す．Rastrigin 関数では世代が進む毎に個体集団が最適解に近づいていくのに対し，Rosenbrock では 100 世代以降あまり変化が見られないことがわかる．すなわち，Rastrigin 関数と Rosenbrock 関数では個体集団の動きに大きな差があることがわかった．しかし，現在の方法では個体集団と真の解を比較しているため，GA 探索中に関数の性質を把握することはできない．今後は GA 探索中に関数の性質を把握できる手法を考案する必要がある．

5 おわりに

GA の探索空間を把握するための手法を提案し，Rastrigin 関数と Rosenbrock 関数で数値実験を行った．その結果，ランドスケープを表示する手法では，GA が得意な問題とされる Rastrigin 関数では探索が容易と考えられる結果が得られ，GA が不得意な問題とされる Rosenbrock 関数にでは探索が困難と考えられる結果が得られた．また，探索過程を把握する手法では，Rastrigin 関数と Rosenbrock 関数では探索の過程が異なる事がわかった．今後はこれらの手法を基に，対象問題に応じて最適な GA アルゴリズムやパラメータを決定するシステムを構築する．

参考文献

1) Ken NAITOU. Four-group equation of genetic algorithm. *JSME International Journal*, Vol. 38, No. 2, 1995.