

局所探索を導入した確率モデル構築型遺伝的アルゴリズムの 計算コストについての検討

佐竹 佑太[†] 棟朝 雅晴^{††} 赤間 清^{††}

確率モデル構築型遺伝的アルゴリズム (Probabilistic Model-Building Genetic Algorithms, PMBGAs) は集団の分布を表した確率モデルを構築し、構築されたモデルを基に新たな個体を生成するアルゴリズムである。PMBGAs は少ない適応度評価回数で広範囲の最適化問題を解くことができるが、モデルを構築するために非常に大きな計算コストを要する。いっぽう、局所探索を導入した PMBGAs では適応度評価回数は多くなるが、モデル構築にかかるコストを削減できる。本論文では、局所探索を導入した PMBGAs における適応度評価回数とモデル構築にかかるコストの関係について検討する。

Considering Computational Cost of Probabilistic Model-Building Genetic Algorithms which Introduce Local Search

YUTA SATAKE,[†] MASAHARU MUNETOMO^{††} and KIYOSHI AKAMA^{††}

Probabilistic Model-Building Genetic Algorithms (PMBGAs) build their probabilistic models which represent distribution of promising solutions and generate new solutions based on the models. Although PMBGAs can solve a wide spectrum of optimization problems effectively, their probabilistic model-building processes need extensive computational cost. On the other hand, in PMBGAs which introduce local search, although the number of evaluations is high, they can reduce the cost to build models. In this paper, we consider the relation of the number of evaluations and the cost to build models in PMBGAs which introduce local search.

1. はじめに

近年、広範囲の最適化問題を解くことが可能なアルゴリズムとして、確率モデル構築型遺伝的アルゴリズム (Probabilistic Model Building Genetic Algorithms, PMBGAs)²⁾ が注目されている。PMBGAs は分布推定アルゴリズム (Estimation of Distribution Algorithms, EDAs) とも呼ばれる。PMBGAs は集団の分布を表した確率モデルを構築し、構築されたモデルを基に新たな個体を生成する。これにより、PMBGAs は少ない評価回数で広範囲の問題を解くことが可能であるが、モデル構築に要する計算コストが非常に大きいという欠点をもつ。クラシカル GA と同様、PMBGAs にも局所探索を組み込む手法がある。この

局所探索を導入した PMBGAs は、大域的な探索と局所的な探索の両方を有するため強力なアルゴリズムとなる。局所探索を導入した PMBGAs では評価回数が多くなるが、モデル構築に要する計算コストを削減できる。局所探索を導入した PMBGAs においては、従来コストとされていた評価回数だけではなく、モデル構築に要するコストにも着目する必要がある。

従来研究では、局所探索を導入した PMBGAs の探索能力のみに着目し、PMBGAs と局所探索を導入した PMBGAs とを比較していなかった。また、評価回数とモデル構築に要するコストの関係についての考察がなされていない。そこで本論文では実験を通して PMBGAs と局所探索を導入した PMBGAs の比較をおこない、評価回数とモデル構築に要するコストの関係について検討する。

まず最初に、本論文で扱う PMBGAs である Extended Compact GA (ECGA)¹⁾ について簡単に説明する。次いで、局所探索を導入した PMBGAs に局所探索を組み合わせた手法とその目的について述べる。最後に、実験結果を示し考察を加える。

[†] 北海道大学大学院 情報科学研究科
Division of Systems and Information Engineering,
Graduate School of Information Science and Technol-
ogy, Hokkaido University.

^{††} 北海道大学 情報基盤センター
Division of Large-Scale Computational Systems, Infor-
mation Initiative Center. Hokkaido University.

- (1) 個体集団 (個体数 N) をランダムに初期化する
- (2) 適応度の評価をおこなう
- (3) トーナメント選択法により選択をおこなう
- (4) 貪欲 MPM 探索により集団の確率モデルを構築する
- (5) モデルに基づいて新たな集団を生成する
- (6) 終了条件が満たされた場合は終了. それ以外の場合は 2. へ戻る

図 1 Extended Compact Genetic Algorithm
Fig. 1 Extended Compact Genetic Algorithm

2. Extended Compact Genetic Algorithm

Extended Compact GA (ECGA)¹⁾ は Harik によって提案された確率モデル構築型遺伝的アルゴリズムである。ECGA は複数の変数間の依存関係を検出できるため、適用範囲が広い。ECGA の手順を図 1 に示す。

ECGA では 2 つ以上の遺伝子の同時確率分布を表すことのできる MPM (Marginal Product Model) という確率モデルを構築する。この MPM を用いて MDL (Minimum Description Length) 原理に基づいたリンケージ学習を行う。つまり、集団に存在する個体の分布をできる限り短い記述長で記述できるようなリンケージを探索する。この記述長を表すために、次に示す 2 つの複雑度が定義される：

$$C_m = \log N \sum_i 2^{S[i]} \quad (1)$$

$$C_p = N \sum_i E(M[i]) \quad (2)$$

ここで、 N は集団サイズ、 $S[i]$ は i 番目のリンケージの長さ、 $M[i]$ は i 番目のリンケージに関する周辺分布、 $E(M[i])$ は $M[i]$ エントロピーである。

貪欲 MPM 探索は、全ての遺伝子が独立である状態 (個体長が l の場合、 $[0][1] \dots [l-1]$) から始める。2 つのリンケージを選んで 1 つのリンケージにした場合の複雑度 C をすべてのリンケージの組み合わせについて計算する。その中でもっとも小さな複雑度が現在の複雑度よりも小さな場合は、そのリンケージの組み合わせを採用する。以上のリンケージ学習を組み合わせるリンケージのペアがなくなるまで繰り返す。このようにして得られたリンケージを基に、新たな個体集団を生成する。

3. 局所探索を導入した確率モデル構築型遺伝的アルゴリズム

GA と局所探索とを組み合わせる手法は古くからおこなわれてきた手法である⁴⁾。このハイブリッドアルゴリズムは、GA の大域的な探索と局所探索による局所的な探索によって強力なアルゴリズムとなる。

PMBGAs または EDAs に局所探索を組み合わせた手法についてもすでに研究がおこなわれており、EDA/L³⁾ などが提案されている。局所探索を組み合わせる目的は 2 つある：(1) クラシカルな GA と同様に、GA の大域的な探索と局所探索による局所的な探索によって強力なアルゴリズムとする目的と、(2) 局所探索によって局所解に到達した個体を基に確率モデルを構築することで、構築されるモデルの品質を高める目的である。ここでは (2) に注目する。単なる PMBGAs では選択された適応度の高い個体を用いて確率モデルを構築する。これではそれほど適応度の高くない個体を用いてモデルを構築することになる。いっぽう、局所探索を導入した PMBGAs では、選択後に局所探索をおこなうことでより適応度の高い個体にする。これによって局所解に到達した個体を用いて確率モデルを構築することができる。

本論文では、局所探索としてランダム局所探索を用いる。以後、局所探索を導入した ECGA を ECGA+LS と呼ぶものとする。

4. 実験

4.1 テスト関数

テスト関数として、5 ビットの 2 進数からなる 5 ビットトラップ関数 $trap_5$ を用いた。対象問題は 5 ビットトラップ関数 $trap_5$ の和で定義する：

$$F_{trap_5}(x) = \sum_{i=1}^n trap_5(u_i) \quad (3)$$

$$trap_5(u) = \begin{cases} 4 - u, & 0 \leq u \leq 4 \\ 5, & u = 5 \end{cases} \quad (4)$$

u は 0, 1 からなる長さ 5 の個体中の 1 の数、 n は部分関数の個数である。個体長 l は $l = 5n$ であり、実験では $l = 50, 75, 100, 125, 150$ ($n = 10, 15, 20, 25, 30$) とした。各 $trap_5$ 関数はだまし性をもつ。5 ビット中に存在する 1 の数は平均的に 2.5 であるため、局所的な探索をおこなうと局所最適解 00000 に収束する可能性が高い。したがって、リンケージ学習によって各変数間の依存関係を検出するなどして、ビルディング

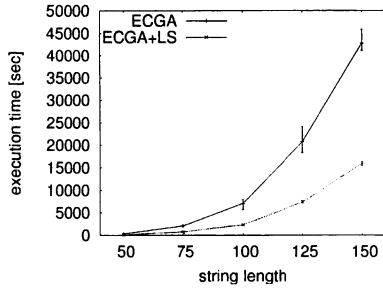


図 2 5 ビットトラップ関数における全実行時間
Fig. 2 Overall execution time for the 5-bit trap test function

グブロックを破壊することなしに組み合わせる必要がある。

4.2 実験結果

ECGA と ECGA+LS の比較をおこなう。実験は SGI Onyx 300 (MIPS R14000 600MHz × 32 CPU, 16GB 共有メモリ) 上でおこなった。比較は、適応度評価およびモデル構築などのすべてのコストを考慮に入れた全実行時間でおこなう。ここで示す実験結果は、異なる乱数シードを用いて 10 回実行した結果の平均である。なお、パラメータは小さな値から徐々に大きくしていき、最適なものを選んだ。局所探索をおこなう回数は各世代において 1/2 回とした。

実験結果を図 2 に示す。この図からわかるように、ECGA+LS の方が ECGA よりも高速に最適解を得ることが可能である。これは、全実行時間中でほとんどの時間を占めるモデル構築にかかる時間を短縮できたためである (表 1)。5 ビットトラップ関数は前述の通り、局所的な探索をおこなうと局所最適解に到達する可能性が高い。しかし、このようなだまし性をもつような関数であっても、局所解に到達した個体を用いることでモデル構築にかかる時間を短縮できることがわかった。

4.3 考察

局所探索を導入することにより、モデル構築にかかる時間を短縮できた。この理由について考察する。

まず、ECGA と ECGA+LS におけるリンケージ同定正解率についてみてみる (図 3)。対象問題は 5 ビットトラップ関数の和である。したがって、各部分関数を構成する 5 ビットを 1 つのリンケージとして検出できた場合に正解とした。図 3 からわかるように、ECGA+LS は ECGA よりも小さい集団サイズでリンケージ同定が可能である。正しいリンケージが

表 1 5 ビットトラップ関数 ($l = 150$) における各実行時間
Table 1 Execution time for the 5-bit trap function ($l = 150$)

	ECGA	ECGA+LS
集団サイズ	7800	3200
選択時間 (秒)	0.515	0.162
局所探索時間 (秒)	—	8.360
モデル構築時間 (秒)	42758.592	15951.612
交叉時間 (秒)	27.093	8.443
適応度評価回数	167700	1925120
リンケージ同定正解率 (%)	100	100
平均世代数	20.5	15.6
全実行時間 (秒)	42786.2	15981.2

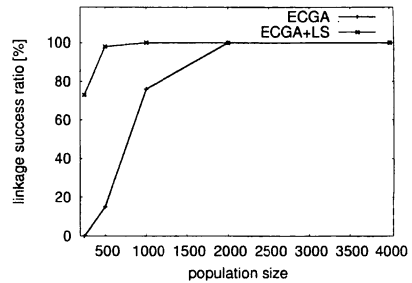


図 3 ECGA と ECGA+LS におけるリンケージ同定正解率
Fig. 3 Success rates of linkage identifications for ECGA and ECGA+LS

同定されれば、ビルディングブロックを破壊することなしに効率よく組み合わせることができるため、最適解を得ることが可能となる。さらに、モデル構築にかかる時間を調べた。結果を図 4 に示す。この図は、個体長を 50 に固定した場合の集団サイズに対するモデル構築時間を示している。モデル構築時間は集団サイズにほぼ比例していることがわかる。このことから、集団サイズが小さくなることによりモデル構築時間が短縮できたと考えることができる。

局所探索を導入した PMBGAs では、PMBGAs に比べ適応度評価回数が多い。適応度評価に時間がかかる問題に対しては、局所探索を導入した PMBGAs が有効でない場合がある。局所探索を導入した PMBGAs が有効である条件について考える。以下で 1 世代毎の各オペレータの時間を定義する。1 つの個体を選択するために要する時間を T_s 、1 つの個体を評価するために要する時間を T_f 、1 つの個体を交叉によって生成するために要する時間を T_c 、トーナメントサイズを s 、局所探索をおこなう回数を n 、集団サイズを $popsiz$ e としてモデル構築時間を $T_M(popsiz$ e, l)、PMBGAs における集団サイズを N 、局所探索を導入

表 2 5 ビットトラップ関数における各実行時間
Table 2 Execution time for the 5-bit trap function

l	50	75	100	125	150
N	1800	2800	4000	6000	7800
N'	800	1300	1600	2600	3200
N'/N	0.44	0.46	0.40	0.43	0.41
gen	11.1	15.0	17.1	20.3	20.5
gen'	8.8	11.2	13.0	14.3	15.6
gen'/gen	0.79	0.75	0.76	0.70	0.76
T_s (秒)	0.003	0.004	0.007	0.012	0.018
$T_M(N, l)$ (秒)	33.922	139.638	397.176	1019.310	2085.785
$T'_M(N, l)$ (秒)	13.992	70.219	181.634	514.482	1022.539
T_c (秒)	0.023	0.107	0.235	0.531	0.931
T_f (秒)	1.0×10^{-6}	2.5×10^{-6}	3.0×10^{-6}	3.8×10^{-6}	4.8×10^{-6}
θ (秒)	0.007	0.021	0.033	0.054	0.097

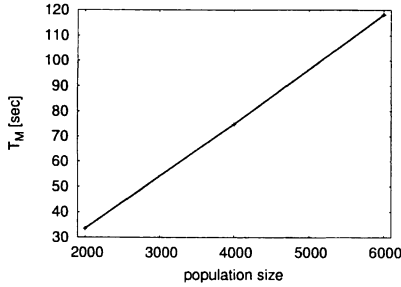


図 4 モデル構築にかかる実行時間 ($l = 50$)
Fig. 4 Execution time for model-building process ($l = 50$)

入した PMBGAs における集団サイズを N' とする。ただし、簡単のため N , N' はともに s で割り切れるものとする。PMBGAs における全実行時間は次のように表すことができる：

$$Time(N, gen) = (NT_f + \frac{N}{s}T_s + T_M(N, l) + NT_c) \cdot gen \quad (5)$$

同様に、局所探索を導入した PMBGAs における全実行時間は次のように表すことができる：

$$Time(N', gen') = (N'T_f + \frac{N'}{s}T_s + \frac{N'}{s}nT_f + T_M(N', l) + N'T_c) \cdot gen' \quad (6)$$

$Time(N, gen) > Time(N', gen')$ となるとき、つまり、

$$T_f < \{(N - N'gen'/gen)(T_s/s + T_c) + T_M(N, l) - T_M(N', l)gen'/gen\} / \{N'(1 + n/s)gen'/gen - N\} \quad (7)$$

となるときに局所探索を導入した PMBGAs が有効となる。このときの右辺を θ とする。ECGA および

ECGA+LS を 5 ビットトラップ関数に適用した場合の各オペレータの実行時間を表 2 に示す。ここで示す T_s , T_c , T_c , および T_f は、ECGA と ECGA+LS の全ての実行における平均値である。 $l = 50, 75, 100, 125, 150$ いずれの場合も、 $T_f < \theta$ となっているため、ECGA よりも ECGA+LS が有効であることが確かめられる。

5. おわりに

ECGA に局所探索を導入することで、モデル構築にかかる時間が短縮できることを示した。また、モデル構築にかかる時間が短縮できる理由について示した。さらに、適応度評価回数およびモデル構築時間に注目し、局所探索を導入した PMBGAs が PMBGAs よりも有効である条件について考察した。

参考文献

- 1) Georges Harik: Linkage Learning via Probabilistic Modeling in the ECGA (IlligAL Report No. 99010), 1999.
- 2) Martin Pelikan, David E. Goldberg, and Fernando Lobo: A Survey of Optimization by Building and Using Probabilistic Models (IlligAL Report No. 99018), *Computational Optimization and Applications*, Vol. 21, No. 1, pp. 5-20 (2002).
- 3) Qingfu Zhang and Jianyong Sun and Edward Tsang and John Ford: Hybrid Estimation of Distribution Algorithm for Global Optimization, *Engineering Computations*, Vol. 21, No. 1, pp. 91-107 (2004).
- 4) Patrick Mills, Edward Tsang, Qingfu Zhang and Jhon Ford: A Survey of AI-based Metaheuristics for Dealing with Local Optima in Local Search, *Department of Computer Science, University of Essex*, No. CSM-416 (2004).