

タブーリストを用いたベイジアン最適化アルゴリズムによる多峰性関数最適化

勝 又 勇 治[†] 倉 橋 節 也[†] 寺 野 隆 雄[†]

本論文では、我々がすでに提案している多峰性関数最適化問題に対するハイブリッド型遺伝的アルゴリズム Tabu-BOA (Tabu-Bayesian Optimization Algorithm) の性能を実験的に評価した結果について報告する。Tabu-BOA では、解候補を直接複数のタブーリストに蓄積し解の多様性を維持するとともに、遺伝的操作にベイジアンネットワークを利用する。Tabu-BOA の特長は、多峰性関数の最適解を複数個同時に、しかも効率良く求められることにある。本論文の主要な成果は、これをさまざまな GA 困難な問題について実証的に示したことにある。本論文では、まず Tabu-BOA の原理を述べ、性能評価を行い、さらに、既存の手法との比較検討を行う。

Multimodal Function Optimization by Bayesian Optimization Algorithm with Tabu Lists

YUJI KATSUMATA,[†] SETSUYA KURAHASHI[†] and TAKAO TERANO[†]

This paper experimentally analyzes the performance of Tabu-BOA (Tabu-Bayesian Optimization Algorithm; a new version of hybrid Genetic Algorithms (GAs)), which we have already proposed. In Tabu-BOA, solution candidates are directly stored to multiple tabu-lists. Bayesian network learning is used as genetic operations to generate offspring. The very unique features of Tabu-BOA are that (1) we are able to have multiple optima or semi-optima simultaneously via Tabu-BOA and (2) the convergence speed is higher than the conventional hybrid GAs. This paper describes the basic principles of the Tabu-BOA, the performance analyses, and the comparisons with conventional 'competing' genetic algorithms.

1. はじめに

遺伝的アルゴリズム (Genetic Algorithm: GA) は、解候補の選択機構、および選択された解候補の遺伝的機構による並列探索機構である^{1)~4)}。GA の成功は、これら 2 つの機構の結合によってもたらされたものである。しかし、現実の複雑な多峰性問題に GA を適用する場合は、(a) 遺伝的浮動 (Genetic Drift) による局所解への早期収束現象；(b) 最適解が複数存在する問題；ならびに、(c) 非常に複雑な景観を持つ関数に対する問題の 3 点を解決しなければならない。いわゆる最近の Competing GA の研究はその方向にある。その中でも特に多峰性関数最適化アルゴリズム⁵⁾ の開発は重要な研究課題である。

我々は、この文脈に従った新しいハイブリッド GA として、これまで Tabu-GA⁶⁾ を、続いて Tabu-

BOA^{7)~9)} を提案した。BOA は、遺伝的操作に全 (あるいは部分) ポピュレーションの利用による Bayesian Network を応用したものであり、GA-hard 問題に対して非常に高い能力を示すことが報告されている^{10)~12)}。Tabu-GA は、多峰性かつ多目的問題を対象として、選択機構を担うタブーリスト、および遺伝的機構を担う Simple GA とのハイブリッド化を行ったものである。Tabu-BOA は、Tabu-GA とベイジアン最適化アルゴリズム (Bayesian Optimization Algorithm; BOA)^{10)~12)} を統合したものである。Tabu-BOA の基本的な構想は非常に単純である。Tabu-GA 同様に、解候補を直接複数のタブーリストに保存し選択操作の制御を行い、遺伝的機構を BOA によって構成する。本論文の主要な成果は、Tabu-BOA が上述した 3 つの問題に対する解を与えることを示すとともに、既存の手法と比較しても十分な性能を持つことを実証したことである。

本論文の構成を次に示す。2 章では、背景と基本的な技術について概要を述べ、あわせて Tabu-GA およ

[†] 筑波大学大学院ビジネス科学研究科
Graduate School of Business Science, University of
Tsukuba

び Bayesian 最適化アルゴリズムについて紹介する．3 章では，Tabu-BOA の原理と概要を説明し，そして 4 章で他の類似アルゴリズムとの比較について述べる．5 章では，Tabu-BOA の有効性実証のための種々の実験，およびその結果について述べる．実験に用いた評価関数は，そのすべてが多数のピークと複雑な形状を持っている変形 One-Max 関数，Sine 関数，Rastrigin 関数および階層型トラップ関数等である．最後に，6 章で結論と課題等について論じる．

2. 背景と基本技術

GA と，シミュレーテッドアニーリング，タブーサーチ^{13),14)}，およびニューラルネットワーク等とのハイブリッド化は，難しい問題に対する探索能力の改善を目的としたものである．しかし，先行研究の大部分が，GA による全体的な探索と，他の方法による局所探索に焦点を合わせたものであり，この方式では局所探索の調整は困難であった．GA は，多峰性関数の最適化に関する唯一の方法であり，これに関する種々の研究が行われてきた．

母集団の多様性保持に関する従来の提案方法は次の方式によるものである．(1) シェアリング関数によって，類似個体への収束を避ける方法．(2) 新個体のリプレース対象を制限する crowding 法．(3) 交叉操作を制限する方法．しかし，調整すべき多くのパラメータがあるため，これらの方法は実務的な問題に適用することが困難であった．

2.1 タブー遺伝的アルゴリズム (Tabu-GA)

このような従来の方法と異なり，Tabu-GA は直接個体を複数のタブーリストの中に格納する新しい手法である (図 1 参照)．Tabu-GA のタブーリストの役割は以下のとおりである．

- (1) 評価値上位個体のリストへの格納．
- (2) エリート個体の再利用．
- (3) 母集団の多様性維持．
- (4) 個体の局所解への収束回避．

Tabu-GA は従来のハイブリッド手法に比べ，扱いが容易であり，また頑健かつ強力である．Tabu-GA は，多峰性，多目的関数最適化を GA ベースの新しい手法で開発することを目的としている．

Tabu-GA では，ただ選択プロセスの改善を目標に設計を行っており，遺伝的操作機構としては従来の Simple GA を使用している．

以下に，Tabu-GA の処理プロセスを示す．

STEP(1)：履歴 H を空とする．

STEP(2)： $N(H, x^{now})$ を選択する．ただし x^{now}

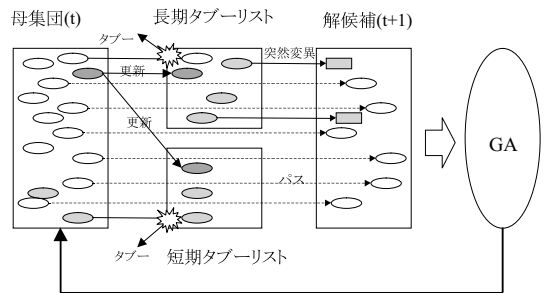


図 1 Tabu-GA の基本的な原理

Fig. 1 Fundamental principle of Tabu-GA.

X .

STEP(3)： $fitness_selection(c(N(H, x^{now})))$ を選択する．

STEP(4)： $x^{next} = crossover(x^{next})$ と $x^{next} = mutation(x^{next})$ を実行する．

STEP(5)： $x^{best} = best_fitness(x^{next})$ を評価する．

STEP(6)：設定された繰返し条件を満たせば終了．

STEP(7)： x^{best} が履歴 H 中の x^{old} よりも適応度が高く，かつ x^{old} と似ていなければ， x^{best} を x^{old} と交換する．

STEP(8)： $X \leftarrow mutation(x^{best})$ を追加する (短期はなし)．

STEP(9)：STEP(2) へ戻る．

ここで， H ：長期タブーリスト登録内容， X ：母集団， $N(H, x^{now})$ ： X^{now} の中で履歴 H を除いた集合， x^{now} ：現世代の個体， x^{old} ：過去の世代の個体， x^{next} ：次世代の個体， x^{best} ：適合度が最良の個体， $fitness_selection()$ ：GA 選択操作を表す (評価値の高い個体を選択)， $c()$ ：目的関数の値， $crossover()$ ：交叉， $mutation()$ ：突然変異， $best_fitness()$ ：最大評価値，である．

次世代候補解とタブーリストの比較，およびタブーリストの更新は，式 (1) の Hamming 距離 d_H で評価する． $d_H > p_d$ であれば更新する (p_d はパラメータ)．

$$d_H(a, b) = \sum_{i=1}^n |a_i - b_i| \quad (1)$$

ここで， a, b ：個体 a, b を示す， $d_H(a, b)$ ：個体 a と個体 b とのハミング距離， n ：個体のビット数 (ストリングサイズ)， i ：個体の各ビットの番号， a_i ：個体 a の i ビット目の値 (0 または 1) である．

2.2 Bayesian 最適化アルゴリズムの基本原則

遺伝的アルゴリズムの問題点の 1 つとして，ビルディングブロック (スキーマ) の破壊があげられる．

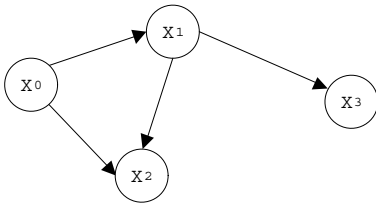


図2 ベイジアンネットワーク
Fig.2 Bayesian Network.

これに対する新しい技術として、解候補の確率的分布推定に基づく研究が注目されている。ベイジアンネットワーク (Bayesian Network) による確率的分布推定方式の遺伝的アルゴリズム (BOA) は Pelikan らによって提案された方法である。BOA は従来の交叉によらない遺伝的処理のため、指定オーダまでビルディングブロックが複製され、非常に効率的に成長し、きわめて高い収束性を示す。

2.2.1 Bayesian Network

BOA では、Bayesian Network が新しい解候補生成のための確率分布推定のために使われる。Bayesian Network は、図 2 に示すように、各独立変数をノードで表し、かつ以下の 2 項目で定義される確率モデルである^{10),15),16)}。

- (1) 全ノードの依存関係を示す非循環有向グラフ。
- (2) 子ノードでの局所的な依存関係を定量的に表した条件付き確率。

図 2 のネットワークの各ノードはそれぞれ変数を表す。 $X_i : (i = 0, \dots, 3)$ は変数およびこの変数に対応するノードを示している。各変数は、解を表すストリングにおける 1 ビットに対応している。2 つの変数間の関係は、2 つのノード間のエッジ (ネットワーク) によって示される。数学的には、有向なエッジからなる非循環 Bayesian Network は式 (2) に示す同時結合確率分布 (joint probability distribution) を表す。

$$p(X) = \prod_{i=0}^{n-1} p(X_i | \pi_{X_i}) \tag{2}$$

ここで、 $X = (X_0, \dots, X_{n-1})$ は変数のベクトル、 π_{X_i} は X_i の親ノード (ネットワークでは π_{X_i} から X_i へのエッジが存在する) である。たとえば、図 2 の Bayesian Network の場合、式 (3) で示される。

$$p(X) = p(X_0)p(X_1|X_0)p(X_2|X_0, X_1)p(X_3|X_1) \tag{3}$$

2.2.2 学習によるネットワークの構築

ネットワーク構築アルゴリズムには評価基準と、探索手法の 2 つの基本要素が存在する。評価基準は、ネッ

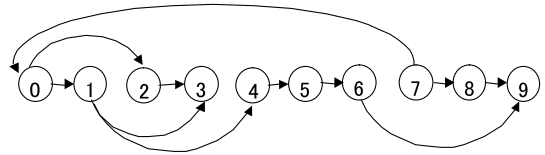


図3 BOAにおける Bayesian Network
Fig.3 Bayesian Network in BOA.

トワークがいかに良好にデータをモデル化しているかの指標である。探索手法は、評価基準の高いネットワーク発見のため用いられる。

(1) 評価基準 (スコアリングメトリック)

BOA では、ある世代の母集団より、適合度の高い個体の集合を抽出する。これをもとにすべてのノード間の Bayesian Network のメトリックを計算する。メトリックはネットワークの良さを示す。

(2) 探索手法 (サーチ処理)

メトリックの高い順にネットワークを確定していく。この際、探索空間は制約オペレータにより縮小可能である。制約は、循環グラフとならないこと、各ノードに入力されるエッジを K 個以内に限定するというものである。 K は設定可能なパラメータである。

2.2.3 Bayesian Dirichlet Metric

ネットワークの品質の指標として、Bayesian Dirichlet (BD) metric を用いる^{10),15)}。これは、対象問題に関する事前知識と、与えられたデータセットの統計的データを統合するものである。BOA において、Bayesian Network は式 (4) に示された Bayesian Dirichlet メトリックに基づいて構築される。より高い評価値の個体群が母集団から抽出され、個体群の統計値に基づいて図 3 のように Bayesian Network が構築される。これにより、新しい個体群 (子孫) が生成される。

$$p(D, B|\xi) = p(B) \prod_{i=0}^{n-1} \prod_{\pi_{X_i}} \frac{m'(\pi_{X_i})!}{(m'(\pi_{X_i}) + m(\pi_{X_i}))!} \prod_{X_i} \frac{(m'(x_i, \pi_{X_i}) + m(x_i, \pi_{X_i}))!}{m'(\pi_{X_i})!} \tag{4}$$

ここで、 D : データセット、 B : ネットワーク、 ξ : 背景インフォメーション、 n : ノード数、 i : ノード番号、 X_i : ノード i の変数ベクトル、 $p(B|\xi)$: ネットワーク B の事前確率、 x_i : X_i のすべてのインスタンス、 π_{X_i} : x_i の親のすべてのインスタンス、 $m(\pi_{X_i})$: X_i の親をともなうデータセット D におけるインスタンスの数を示す。なお、インスタンスは親ノードと子ノードの実例データを示したものであり、データセッ

トはインスタンスの集合体である．また， $m(\pi_{X_i})$ と $m(x_i, \pi_{X_i})$ との関連を式 (5) に示す．

$$m(\pi_{X_i}) = \sum_{x_i} m(x_i, \pi_{X_i}) \quad (5)$$

$m'(x_i, \pi_{X_i})$: x_i にセットした可変的な X_i を持っているインスタンスの数についての事前情報を表す．そして変数 π_{X_i} のセットは π_{X_i} にインスタンスを作る．式 (5) は， m 同様， m' にも適用できる．

2.2.4 BOA の処理

図 3 は，10ビットストリングの例である．10個の各ノード (0~9) は，ストリングの各ビットに対応する．異なるビット間の依存関係をネットワークで表現している．ノード 7 (以下，node(7) と記す) が始点となり，node(3) および node(9) が終点となる．node(1) に着目すると，その親ノードは node(0) であり，さらに node(0) の親ノードは node(7) である．node(1) の生起確率は node(0) の結果に依存する．node(3) の場合，node(1) と node(2) の結果に依存する．

以降，BOA の処理プロセスについて示す．

STEP(1) : 初期個体母集団生成

初期母集団 $P(0)$ をランダムに作成する ($t=0$) .

STEP(2) : 候補個体選択

母集団 $P(t)$ から，候補個体 $S(t)$ を選択する．

STEP(3) : Bayesian Network 構築

$S(t)$ の各個体の統計量をもとに Bayesian Network を構築する．

STEP(4) : 新個体 (子孫) 生成

Bayesian Network により作成された結合確率分布に従い，新しい個体 (子孫) を作成する．

STEP(5) : 母集団への置き換え

母集団 $P(t)$ における評価値の低い個体と新個体を置き換え，新しい母集団 $P(t+1)$ を作成する．

STEP(6) : 終了条件判定

もし，最終基準でない場合は，STEP(2) に行く．

3. Tabu-BOA

本章では，Tabu-BOA の詳細について解説を行う．Tabu-GA と比較した Tabu-BOA の特徴を次に示す．(1) 遺伝的処理に BOA を使用する．(2) 良いビルディングブロックを継承させるため，突然変異は用いない．

3.1 タブーリストの構成と使用法

Tabu-BOA における長期および短期タブーリストの基本構成は Tabu-GA と同等である．Tabu-BOA のアルゴリズム基本的構想は，各世代の最良解を短期および長期タブーリストに格納することである．最良解が複数ある場合は複数個の格納が可能である．個体間

の Hamming 距離 (式 (1) 参照) を測定し，その距離が d 以下の場合はタブーとして扱う．選択時のタブー制約によって，局所解に陥る危険性を低減することができる．これは，多様解を得るために，新しい解空間を探索することに寄与する．

3.1.1 長期タブーリスト

子孫は，タブーリストに記録されたすべてのデータと Hamming 距離により比較され，評価値が他のものより高い場合にのみ長期タブーリストに格納される．つまり，リスト内で最も新しいデータが一番高い評価値であることを意味する．また，長期タブーリストのサイズは任意に設定可能である．リストが充満した際，古いデータから順に削除される．これは，いくつもの解を得ることができることを意味する．たとえば，もし長期タブーリストの大きさを 2 に設定した場合，タブーリストの中に解が存在すれば，異なる 2 つの解を得ることができる．これにより，多峰性関数であれば複数の (準) 最適解が同時に得られる．

長期タブーリストの役割は次のとおりである．

- 個体群の多様性を維持．
- 遺伝子個体は重複しない．
- 過去の最適個体 (多様解) を保持．

3.1.2 短期タブーリスト

短期タブーリストは個体が n 回以上選択されるのを妨げる．これは，過去 n 世代分の世代最適解を格納し， n 世代間のみタブーとなる．長期タブーリストのみで処理した場合，ある世代において，今までに得られた最良解の少し下付近を振動し進化が停滞する傾向にある．短期タブーリストによりこれを改善できる．短期タブーリストの役割は次のとおりである．

- 過去 n 世代分の世代最適解．
- n 世代間のみタブーとなる．
- 局所解への停留禁止．

3.2 Tabu-BOA の処理

Tabu-BOA の処理概要を図 4 に示す．また，Tabu-BOA の処理プロセスを以下に示す．

STEP(1) : 初期個体母集団生成

初期母集団 $P(0)$ をランダムに作成する ($t=0$) .

STEP(2) : 長期・短期タブーリスト登録

STEP(3) : 候補個体の集合を選択

母集団 $P(t)$ から，候補個体 $S(t)$ を選択する．

STEP(4) : Bayesian Network 構築

$S(t)$ の各個体の統計量をもとに Bayesian Network を構築する．

STEP(5) : 新個体 (子孫) 生成

Bayesian Network により作成された結合確率分

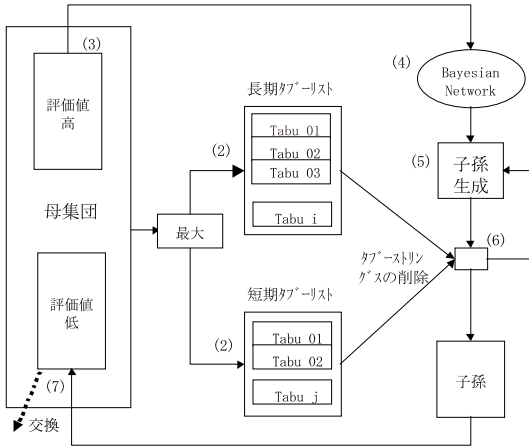


図4 Tabu-BOA の処理概要
Fig. 4 Treatment outline of Tabu-BOA.

布に従い、新しい個体 (子孫) を作成する。

STEP(6) : Tabu 個体か否かの判断

短期および長期タブーリストの内容と比較し、Tabu 個体の場合は再度個体を作り直す。

STEP(7) : 母集団への置き換え

母集団 P(t) における評価値の低い個体と新しい個体とを置き換え、新母集団 P(t+1) を作成する。

STEP(8) : 終了条件判定

もし、最終基準でない場合は、STEP(2) に行く。

4. 他の類似アルゴリズムとの比較

4.1 Sharing-BOA

シェアリングは、多くの点が集中している部分での関数値には比較的小さい重みをかけるのに対して、孤立している点の関数値には相対的に大きい重みをかけることにより、関数値の分布の相対的な均一化を図る操作である^{1),17)}。具体的には、ある個体から他の個体間の距離 (本論文では、Hamming 距離) の総和を求め、その大きさに応じた重みをその個体にかける方法である。これにより、(1) 最適化問題における初期収束の防止、(2) 分類システムにおける多様性の維持、(3) 局所解の分布の検出、などの効果が期待できる。ここでは、Tabu-GA および Tabu-BOA と同様、選択機構を担う Sharing、および遺伝的機構を担う BOA とのハイブリッド化による Sharing-BOA を定義し、Tabu-BOA との性能比較を行う。コード化された文字列 s_i, s_j 間の距離 $d_{ij} = d(s_i, s_j)$ を考慮し、かつ式 (6), (7), (8) を満たす関数を sharing 関数という。

$$0 \leq sh(d) \leq 1, \forall d \in [0, \infty) \quad (6)$$

$$sh(0) = 1 \quad (7)$$

$$\lim_{d \rightarrow \infty} sh(d) = 0 \quad (8)$$

sharing 関数については、式 (9), (10) の提案がある。

$$sh(d) = 1 - \left(\frac{d}{\sigma_{share}}\right)^\alpha, \dots d < \sigma_{share} \text{ のとき} \quad (9)$$

$$sh(d) = 0 \dots \text{その他} \quad (10)$$

ここで、 σ_{share} 、および α は定数である。いま、個体 s_i の適合度関数を $f(s_i)$ 、個体群サイズを N として、シェアリングにより式 (11), (12) に従って適合度を修正する。

$$f(s_i) = \frac{f(s_i)}{m_i} \quad (11)$$

$$m_i = \sum_{j=1}^N sh(d(s_i, s_j)) \quad (12)$$

$d(s_i, s_j)$ コード化された文字列 s_i, s_j 間の距離。

4.2 階層型 BOA

イリノイ大学の Pelikan らは、BOA と Niching テクニックを結合した階層型 BOA¹²⁾を提案した。階層型 BOA は、(1) Bayesian Network、(2) ビルディングブロックのローカル構成、および、(3) 限定トーナメント選択をベースにしている。我々の提案した Tabu-BOA は、限定トーナメント選択の代わりにタブーリストを用いたものである。Pelikan らは、階層型トラップ問題を作成し、階層型 BOA で解いている。本論文では、Tabu-BOA により階層型トラップ問題を解き、階層型 BOA との性能比較を行う。

5. 実験

Tabu-BOA の有効性を実証するために、多峰性のテスト関数による実験を行い、最適解の獲得性能を評価した。定数関数、One-Max-minus-One 関数、Sine 関数、Rastrigin 関数および階層型トラップ問題を扱う。定数関数は、第 1 の問題、Tabu-BOA が遺伝的浮動現象を避けることができるかどうかの確認に相当である。One-Max-minus-One 関数と Sine 関数は、第 2 の問題、複数の最適解を持つ関数に対するアルゴリズムの評価に相当である。Rastrigin 関数と階層型トラップ問題は、第 3 の問題に対する回答を与えるとともに、あわせて Tabu-GA との複雑な景観を持つ関数に対する評価に相当である。

なおすべての実験につき、乱数初期値を変化させて 10 回試行したうち、最良値を結果として各グラフに表現した。最良値と平均値とは大きな相違はなかった。

また、各関数は実数値関数であるが、実験には各個体のバイナリ表現を利用している。この実験条件はオ

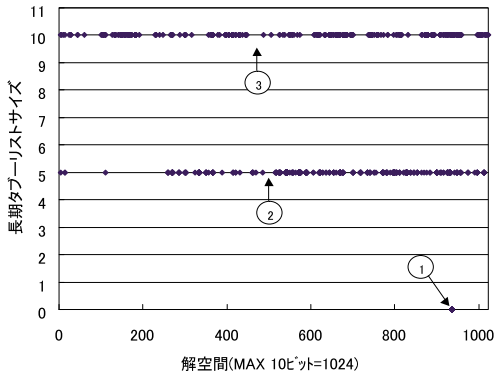


図 5 定数関数 (全個体の分布)

Fig. 5 Constant function (the distribution of all the individuals).

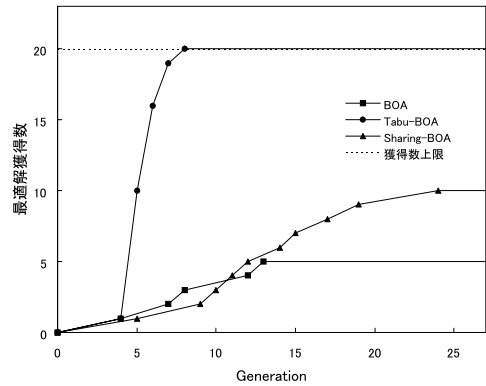


図 6 1次元の実験結果

Fig. 6 Result of 1 dimension.

リジナル BOA と同じである。

5.1 遺伝的浮動の回避

BOA は、世代が進むごとに、解候補の 1 点に向けて全個体が収束する特徴があり、多様性が維持できないという欠点がある。Tabu-BOA では、タブーリストにより多様性の維持を図るものである。定数関数 $y=100$ を用いて実験を行い、多様性を確認する。

- Problem サイズ=10 .
- Population サイズ=512 .
- Generation=568 (BOA が収束した世代数) .
- 評価関数 $y=100$.

図 5 の ① は BOA において全個体が 1 点に収束 (568 世代目) した状態を示す。② は 568 世代目における長期タブーリストサイズ 5 の Tabu-BOA の場合、③ は 568 世代目における長期タブーリストサイズ 10 の Tabu-BOA の場合を示す。BOA が 1 点に収束した世代においても Tabu-BOA の場合の個体は、広域に分布している。また、長期タブーリストサイズが大きい方がより均等に拡散分布していることが観察され、解の多様性確保が確認できる。

5.2 複数最適解を持つ関数の最適化 (1)

One-Max-Minus-One 関数は、多数のピークを用いた問題設定によく使われる One-Max 関数の変形である。ストリングが n ビットから形成される場合、One-Max-Minus-One 関数は、次のように定義される。

- $F(x) = k : 0 \leq k < n$ のとき、かつストリング内の 1 の数が k の場合。
 - $F(x) = 0$: ストリングス内の 1 の数が n の場合。
- たとえば、16 ビット個体の場合、下記のように 16 個の解が存在する。

```
0111 1111 1111 1111
1011 1111 1111 1111
```

....

```
1111 1111 1111 1101
1111 1111 1111 1110
```

これを、C プログラムで表現したものを下記に示す。

```
for ( i=0; i<n; i++)
    f += x[i];
if(f==n)
    f=0;
return f;
```

ここでは 1 次元、2 次元、そして 5 次元について One-Max-Minus-One 関数の実験を行う。One-Max-Minus-One 関数の 2 次元化については、1 次元の配列を 2 つ設定し、個別にオペレーションを行う。評価値は、両配列の和 $[f(x) = f(x1) + f(x2)]$ をとることとする。 $f(x)$ のサイズが 8 ビットの場合、ピーク数は 64 個となる。5 次元の場合も同様であり、ピーク数 32,768 個の多峰性関数となる。

5.2.1 1次元の実験結果 (図 6):

- Problem サイズ=20 .
- Population サイズ=512 .
- 全最適解数=20 .
- Sharing の $\alpha=1.0$.
- Sharing の $\alpha_{share}=100$.

BOA は、13 世代目に 5 個の解を得て、その後の世代ではそれ以上の解を得ることが不可能となっており、すべての 20 個の最適解を得ることができない。Sharing-BOA は、24 世代目に 10 個の最適解を獲得し、BOA よりは優れているもののすべての最適解を得ることはできない。それに対し、Tabu-BOA は、8 世代目においてすべての最適解 20 個を得ることに成功している。

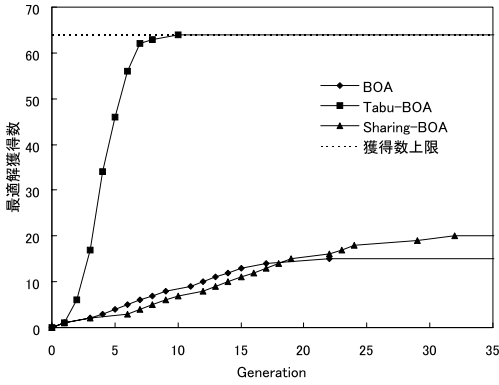


図7 2次元の実験結果
Fig. 7 Result of 2 dimensions.

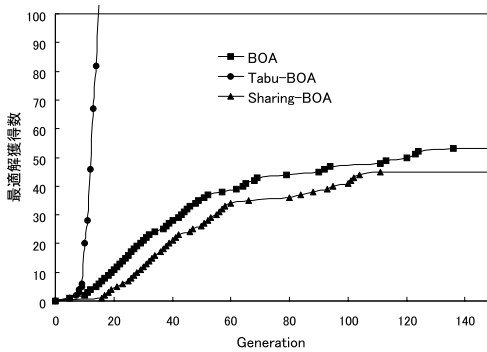


図8 5次元の実験結果
Fig. 8 Result of 5 dimensions.

5.2.2 2次元の実験結果(図7):

- Problem サイズ=8 × 2 = 16 .
- Population サイズ=1024 .
- 全最適解数=64 .
- Sharing の $\alpha=1.0$.
- Sharing の $\alpha_{share}=100$.

Tabu-BOA は、10 世代目にすべての最適解 64 個を得ることに成功している . しかし、BOA と Sharing-BOA は、20 個以下の獲得数となっている .

5.2.3 5次元の実験結果(図8):

- Problem サイズ=8 × 5 = 40 .
- Population サイズ=2048 .
- 全最適解数=32,768 .
- Sharing の $\alpha=1.0$.
- Sharing の $\alpha_{share}=100$.

全最適解数の 32,768 の実験である . Tabu-BOA は、20 世代以下で 100 個以上の最適解を獲得している . それに対し、BOA と Sharing-BOA は 140 世代にて 60 個以下の獲得数にとどまっている .

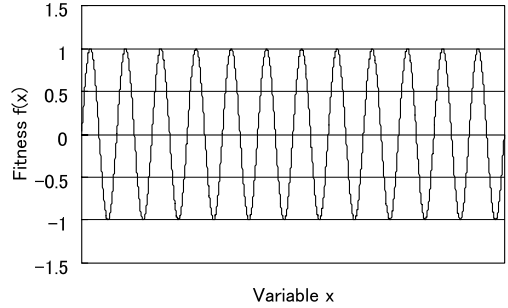


図9 評価関数(一般Sine関数)
Fig.9 Fitness function (normal Sine).

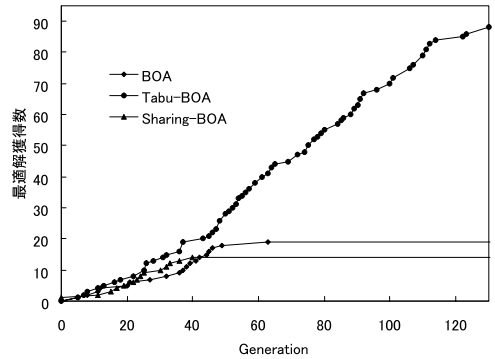


図10 一般Sine関数の実験結果
Fig.10 Results of normal Sine function.

5.3 複数最適解を持つ関数の最適化(2)

典型的な多峰性関数として、式(13)に示す単純な Sine 関数(図9)を使用して、アルゴリズムをテストした . 結果を図10に示す .

- Problem サイズ=20 .
- Population サイズ=1024 .
- Sharing の $\alpha=1.0$.
- Sharing の $\alpha_{share}=100$.
- 全最適解数=524 .

$$y(x) = \sin(0.001 \cdot x), \tag{13}$$

where $0 < x < 1048575$ (20ビット)

BOA と Sharing-BOA は、最適解の獲得数は 20 個以下にとどまっているが、Tabu-BOA の場合は直線状に増加し、120 世代で 85 個の最適解を獲得している .

5.4 複雑な景観を持つ関数の最適化(1)

Tabu-GA と Tabu-BOA の性能比較を行った . 図11に示す 2次元 Rastrigin 関数において、最適解到達までの性能を比較する . 結果を表1に示す . 関数評価回数は、Tabu-GA が 5,400、Tabu-BOA が 2,560 となり、Tabu-BOA の方が優れていることが分かる . これは、BOA の解探索性能の良さに起因するものである .

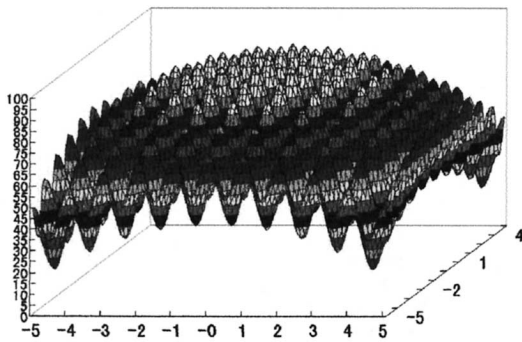


図 11 Rastrigin 関数 (2次元)

Fig. 11 Rastrigin function (2 dimensions).

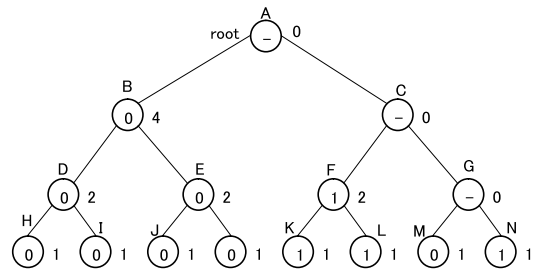


図 13 HIFF 関数

Fig. 13 HIFF function.

表 1 Tabu-GA と Tabu-BOA の比較 (2次元 Rastrigin 関数による実験)

Table 1 Comparison of Tabu-GA and Tabu-BOA (2 dimensions Rastrigin function).

	tabu-GA	Tabu-BOA
母集団サイズ (a)	20	256
最適解到達までの世代数 (b)	270	10
(a) × (b)	5,400	2,560

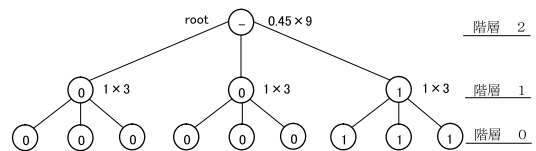


図 14 trap3 関数

Fig. 14 Trap3 function.

- Population size=1024 .
- Sharing の $\alpha=1.0$.
- Sharing の $\alpha_{share}=100$.

Tabu-BOA と BOA を比較すると、2次元と5次元の実験において、Tabu-BOA の方が勝っている。Sharing-BOA は、BOA よりむしろ劣っており、5次元の実験では最適解を得ることができなかった。

5.5 複雑な景観を持つ関数の最適化 (2)

本節では、Pelikan らが提案した階層的トラップ関数¹²⁾について、Tabu-BOA による実験結果を示す。

5.5.1 HIFF (Hierarchical If-and-Only-If) 関数

HIFF は、図 13 に示す 2 分木により構成される。最下層のリーフノード (H) ~ (N) は、入力ストリング (00001101) を示す。ノードの丸の中にノード値 (0, 1) を示し、ノードの右側に貢献値を示す。ノード値は 2 つの子ノードの値により決定される。ノードの貢献値はノード値とリーフノードからの高さ (1, 2, ...) の 2 乗を掛け合わせるにより得られる。全体の評価値は各ノードの貢献値の総和であり、きわめて複雑な多峰性関数となる。最適解は 2 個である。

5.5.2 階層型トラップ関数 (HTF: Hierarchical Trap Function)

階層型トラップ関数は、図 14 で示される 3 分木により構成される。3 つの子ノード値から式 (14), (15) および図 15 によりトラップ値を求め、さらにこれにリーフノードからの高さの 3 乗を掛け合わせるにより得られる。HIFF 関数と同様きわめて複雑な多峰

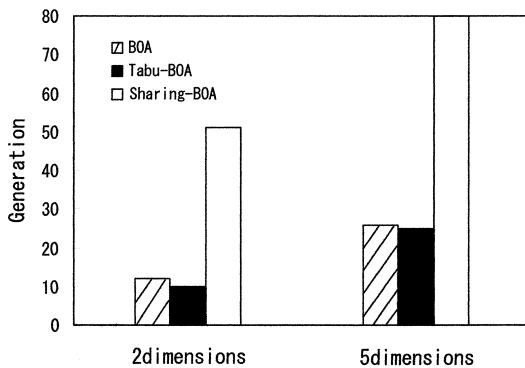


図 12 Rastrigin 関数 (2次元, 5次元) 結果

Fig. 12 Results of Rastrigin function (2 and 5 dimensions).

さらに、BOA, Tabu-BOA および Sharing-BOA について 2次元と5次元の Rastrigin 関数に適用して実験を行った。図 12 に 2次元と5次元の両ケースの結果を示す。これは、最適解を得るのに必要とされる世代数を示す。

(1) 2次元のケース :

- Problem size=16 .
- Population size=512 .
- Sharing の $\alpha=1.0$.
- Sharing の $\alpha_{share}=100$.

(2) 5次元のケース :

- Problem size=40 .

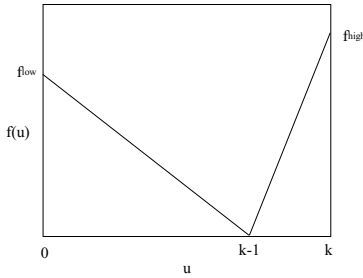


図 15 trap 関数

Fig. 15 Trap function.

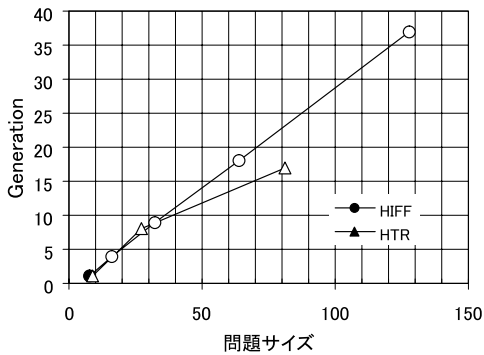


図 16 階層問題の実験結果

Fig. 16 Results of hierarchical problem.

性関数であり、最適解は 1 つである。

(1) $u = k$ の場合

$$f(u) = f_{high} \quad (14)$$

(2) $u \neq k$ の場合

$$f(u) = f_{low} - u \frac{f_{low}}{k-1} \quad (15)$$

各変数は、 $f(u)$: トラップ関数、 u : 変数、 k : u のとりうる最大値、 f_{low} : $u = 0$ のとき $f(u)$ の値、 f_{high} : $u = k$ のときの $f(u)$ の値をあらわす。

5.5.3 実験結果

両階層問題についての母集団数 1,280 による実験結果を図 16 に示す。問題サイズを変化させ、最適解到達世代数をプロットしたものである。Pelikan らの階層型 BOA による実験結果とほぼ同等の性能を示した。

5.6 実験の考察

Tabu-BOA は、多様解の発見能力および収束の速さにおいて、BOA および Sharing-BOA をしのぐ。Sharing-BOA は、一時的な局所解への収束回避に寄与するものの、その効力は長期間持続せずまた同じ所を探索する傾向がある。よって、多峰性関数の最適解あるいは多様解の獲得については、Tabu リストと BOA とのハイブリッド化による Tabu-BOA が最良であるとの結論に至った。

これはタブーリストの効果が大きいといえる。Sharing-BOA は、BOA と比べ顕著に優れているとはいい難く、むしろ劣っているケースの実験結果もあった。Sharing はパラメータの設定が困難であり、良い性能を示すパラメータの組合せを見つける手法の研究が必要と考える。また Sharing-BOA は 1 世代の処理に Tabu-BOA の数倍の時間を要するケースもあった。さらに Tabu-BOA は、階層型トラップ問題に対しても高い性能を示すことを確認した。

6. おわりに

本論文では、多峰性関数最適化手法として Tabu リストとベイジアン最適化アルゴリズムのハイブリッド化による Tabu-BOA の性能を検証した。各種の多峰性関数の実験を通して、Tabu-BOA が、(a) 遺伝的浮動、(b) 複数最適解を持つ関数、(c) 複雑な景観を持つ関数について優れた性能を持つことを実証した。

Tabu-BOA は実問題の解決にも強力な手段である。我々は Tabu-BOA を「発電所電気設備最適設計」問題に適用してその解決に成功した⁹⁾。

また、複雑な人工社会シミュレーションの最適化^{18),19)}にも同様な考え方を適用している。これらの成果は提案した手法の頑健性を示すものである。

今後の課題として、さらなるパフォーマンス向上のためのチューニング手法の確立がある。また、実数値 GA への発展、さらなる実問題への応用があげられる。

謝辞 本研究の基盤である BOA を研究開発し、論文およびソースコードを公開し提供いただいた Illinois 大学の Pelikan, Goldberg 氏に感謝の意を表します。

参考文献

- 1) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, MA: Addison Wesley (1989).
- 2) 北野宏明(編): *遺伝的アルゴリズム*, 産業図書 (1993).
- 3) 寺野隆雄: GA を使いこなすには、*情報処理*, Vol.40, No.6, pp.628-631 (1999).
- 4) Mitchell, M.: *An Introduction to Genetic Algorithm*, Massachusetts of Technology (1996).
- 5) *Proc. Genetic and Evolutionary Computation Conference (GECCO 2001)*, Morgan-Kaufmann (2001).
- 6) Kurahashi, S. and Terano, T.: A Genetic Algorithm with Tabu Search for Multimodal and Multiobjective Function Optimization, *Proc. Genetic and Evolutionary Computation Conference (GECCO 2000)*, pp.291-298 (2000).

- 7) Katsumata, Y., Kurahashi, S. and Terano, T.: Hybridizing Bayesian Optimization and Tabu Search for Multimodal Functions, *Rate breaking paper of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, pp.227-233 (2001).
- 8) 勝又勇治, 倉橋節也, 寺野隆雄: ベイジアンネットワークとタブーリストを利用したハイブリッド GA による多峰性関数最適化, 第 15 回人工知能学会全国大会 2C3-01, 松江 (2001).
- 9) 勝又勇治, 寺野隆雄: 発電所電気設備設計に対する進化計算の適用—タブー探索とベイジアン最適化による接近, 電気学会論文誌 C, Vol.122-C, No.3, pp.417-424 (2002).
- 10) Pelikan, M., Goldberg, D.E. and Cantu-Paz, E.: Linkage Problem, Distribution Estimation, and Bayesian Networks, University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory Technical Reports No.98013 (1998). <http://www-illigal.ge.uiuc.edu/>
- 11) Pelikan, M., Goldberg, D.E. and Cantu-Paz, E.: Bayesian Optimization Algorithm, Population Sizing, and Time to Convergence, University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory Technical Reports No.2000001 (2000). <http://www-illigal.ge.uiuc.edu/>
- 12) Pelikan, M. and Goldberg, D.E.: Escaping Hierarchical Traps with Competent Genetic Algorithms, *Proc. Genetic and Evolutionary Computation Conference (GECCO 2001)*, pp.511-518 (2001).
- 13) Reeves, R.C., 横山隆一ほか(訳): *モダンヒューリスティックス—組合せ最適化の先端手法*. 日刊工業新聞社 (1997).
- 14) Glover, F. and Laguna, M.: *TABU SEARCH*, Kluwer Academic Publishers (1997).
- 15) Heckerman, D.: *Learning Bayesian Networks: The Combination of Knowledge and Statistical Data*, Microsoft Research Advanced Technology Division Technical Report MSR-TR-94-09 (1994).
- 16) 本村陽一: 不確実モデリングのための情報表現: ベイジアンネットワーク, 2001 年ベイジアンネットワークチュートリアル (2001).
- 17) 坂和正敏, 田中雅博: *遺伝的アルゴリズム*, 日本ファジイ学会, 朝倉書店 (1995).
- 18) 倉橋節也, 南 潮, 寺野隆雄: *逆シミュレーション*

ン手法による人工社会モデルの分析, 計測自動制御学会論文集, Vol.35, No.11, pp.1454-1461 (1999).

- 19) 倉橋節也, 寺野隆雄: エージェントシミュレーションによる共同分配規範モデル, 電子情報通信学会論文誌 D-I, Vol.J84-D-I, No.8, pp.1160-1168 (2001).

(平成 14 年 2 月 4 日受付)

(平成 14 年 4 月 16 日再受付)

(平成 14 年 5 月 28 日採録)



勝又 勇治 (学生会員)

1979 年明治大学工学部電気工学科卒業。同年電源開発(株)。1997 年筑波大学大学院政策科学研究科修士課程(経営システム科学専攻)修了。1999 年筑波大学大学院博士課程ビジネス科学研究科入学, 現在に至る。人工知能学会, 電気学会, 計測自動制御学会各会員。



倉橋 節也

1995 年放送大学教養学部(産業と技術専攻)卒業。1998 年筑波大学大学院政策科学研究科修士課程(経営システム科学専攻)修了。2002 年同大学院博士課程(企業科学専攻)修了。博士(システムズ・マネジメント)。東京電機産業(株)に勤務。人工知能学会, 計測自動制御学会, 日本オペレーションズリサーチ学会各会員。



寺野 隆雄 (正会員)

1978 年東京大学大学院情報工学専攻修士課程修了。1978~1989 年(財)電力中央研究所。1990 年~現在, 筑波大学社会工学系(大学院企業科学専攻・経営システム科学専攻)。1996 年同教授。1991 年工学博士。1996 年イリノイ大学, スタンフォード大学客員研究員。人工知能学会, 計測自動制御学会, 経営情報学会, 日本オペレーションズリサーチ学会, 電気学会, スケジューリング学会, 社会情報学会各会員。