

複数の確率モデルと独立成分分析を用いた分布推定アルゴリズムの提案

中尾 昌広[†], 廣安 知之^{††}, 三木 光範[‡], 横内 久猛^{††}, 吉見 真聡[‡]

同志社大学大学院[†], 同志社大学生命医科学部^{††}, 同志社大学理工学部[‡]

1 はじめに

設計変数が実数値である最適化問題（以下、連続問題と呼ぶ）に対して、効率良く最適解を求めるメタヒューリスティック手法として、分布推定アルゴリズム（EDA: Estimation of Distribution Algorithm）³⁾が近年注目されている。高い探索性能を持つEDAの1つにHIS（Hierarchical Importance Sampling）²⁾がある。HISは多様性が異なる複数の集団と確率モデルを用いることで、集団の収束と多様性の調節を行うという特徴を持つ。しかし、HISは離散問題に対しては有効性が確認されているが、連続問題に対しては用いる確率モデルによって最適解の発見が困難である場合がある¹⁾。また、EDAの1つにPBILc（Continuous Population-Based Incremental Learning）⁵⁾がある。PBILcはシンプルな確率モデルの更新則を持つが、高次元の連続問題においても高い性能を発揮する。しかし、各設計変数が独立に分布推定を行うため、依存関係を持つ連続問題に対して性能は低い。依存関係を持つ連続問題は、実問題においてよく出現する問題クラスである。

我々はHISにPBILcの更新則を用いた新しいEDAであるRMM（Real-coded EDA using Multiple Probabilistic Models）を開発している⁹⁾。本稿では、RMMの性能向上を目的として、RMMに独立成分分析機能を付加した新しいEDAを提案する。独立成分分析機能を用いることで、依存関係を持つ問題に対しても高い性能を発揮すると考えられる。本稿では、提案するEDAのアルゴ

リズムを説明し、依存関係を持つ連続問題に対してその有効性を調査する。

2 既存の分布推定アルゴリズム

2.1 Hierarchical Importance Sampling

HISは L 個の集団 X_1, \dots, X_L と L 個の確率モデル $p_1(x), \dots, p_L(x)$ を解探索に用いる。各集団と各確率モデルは1対1対応している。HISは、例えばほぼ収束している集団とほぼ一様分布に近い状態にある集団を同時に用いて解探索を行う。そのため、解探索終盤においても全探索点は収束することなく、良い探索点を発見する可能性を残すことができる。

HISは「サンプリング」と「推定」という2つの操作を用いて解探索を行う。「サンプリング」では、各確率モデルは独立に探索点を生成する（Fig. 1）。「推定」では、ある確率モデル $p_l(x)$ の分布推定を行うために、確率モデル $p_l(x)$ と対応関係にある集団 X_l と集団 X_{l+1} および X_{l-1} をマージした集団から良好な探索点を選択する（Fig. 2）。この工夫により、ある確率モデルが生成した良好な探索点を異なる確率モデルの構築に用いることができる。

HISでは探索点の分布の多様性を定義しており、解探索を行う前に、その定義を用いて各確率モデルの目標分布 $q_1(x), \dots, q_L(x)$ を設定する。目標分布 $q_1(x), \dots, q_L(x)$ は多様性が順に変化する値を設定する。「推定」において良好な探索点を選択する際、確率モデル $p_l(x)$ は対応する目標分布 $q_l(x)$ に近い値になるように探索点を選択する。具体的には、多様性は選択する探索点数によって変化するため、多様性を低くしたい確率モデルには探索点を少なく、多様性を高くしたい確率モデルには探索点を多く選択する。これらの工夫により、HISはすべての集合の多様性を組織化して調節できるため、有効な解探索を行うことができる。

Proposal of Real-coded Estimation of Distribution Algorithm Using Multiple Probabilistic Models and Independent Component Analysis

[†] Masahiro Nakao (mnakao@mikilab.doshisha.ac.jp)

^{††} Tomoyuki Hiroyasu (tomo@is.doshisha.ac.jp)

[‡] Miki Mitsunori (mmiki@mail.doshisha.ac.jp)

^{††} Hisatake Yokouchi (hyokouch@mail.doshisha.ac.jp)

[‡] Masato Yoshimi (myoshimi@mail.doshisha.ac.jp)

Graduate School of Engineering, Doshisha University (†)

Department of Life and Medical Sciences, Doshisha University (††)

Faculty of Science and Engineering, Doshisha University (‡)

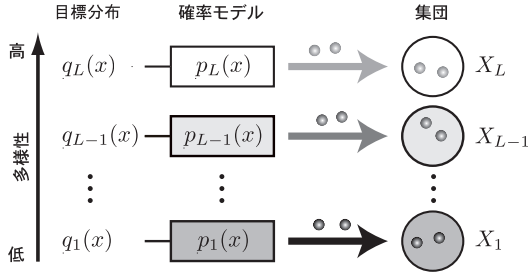


Fig. 1 HISの動作 (サンプリング)

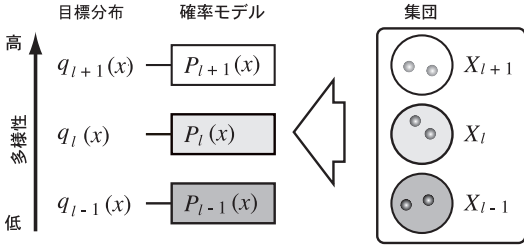


Fig. 2 HISの動作 (推定)

2.2 Continuous Population-Based Incremental Learning

PBILcでは設計変数毎に正規分布に基づく確率モデル $N(\mu_i, \sigma_i^2)$ を構築し、解探索を行う。 N は正規分布を表しており、 μ_i は平均、 σ_i^2 は分散である。

PBILcの確率モデルの構築方法について述べる。平均 μ_i は設計変数毎に式(1)に従い線形結合し、更新する。

$$\mu_i^{t+1} = (1-\alpha)\mu_i^t + \alpha(X^{best1} + X^{best2} - X^{worst}) \quad (1)$$

式(1)は集団の1番目と2番目に評価値の高い探索点の設計変数を足し、評価値が最も低い探索点の設計変数を引いた値をもとに、次世代の平均 μ_i を計算することを表している。式(1)の意図は、1番目と2番目に評価値の高い探索点を足すことで集団が1カ所に留まることを防ぎ、かつ評価値が最も低い探索点から離れる方向に平均 μ_i を移動させることで、良いと考えられる方向に確率モデルを移動させることである。 α ($0 < \alpha \leq 1$) は学習率である。学習率 α は小さいほど過去の確率モデルの影響を大きく受けるため、確率モデルの変化は小さくなる。

分散 σ_i^2 については、式(2)を用いた更新方法が推奨されている。

$$\sigma_i^{t+1} = (1-\alpha)\sigma_i^t + \alpha \sqrt{\frac{\sum_{j=1}^k (X^j - \bar{X})^2}{k}} \quad (2)$$

PBILcには探索点数 P 、学習率 α 、切捨て率 K ($0 < K \leq 1$) という3つのパラメータがある。

PBILcは切捨て率 K により、集団中の評価値の高い上位 $P \times K$ 個の探索点を選択する。この探索点数が式(2)で用いている k である。集団 X は評価値順にソート済みであるとする。 \bar{X} は選択された探索点の設計変数の平均値である。この式の意図は、集団中の上位 k 個の探索点の分散 σ_i^2 をもとに次世代の分散 σ_i^2 を決定することである。

PBILcの特徴は、学習率 α を用いることで急激な確率モデルの変化を防ぎ、集団の初期収束を防ぐことである。

3 提案アルゴリズム

我々が開発しているRMM⁹⁾は、複数の集団と確率モデルを用いる仕組みについてはHISを、集団から確率モデルを構築し、探索点を生成する方法についてはPBILcを参考にしている。本稿では、対象問題のランドスケープを考慮した解探索を行うために、独立成分分析を用いた探索点生成の機能をRMMに付加することを考える。

3.1 Real-coded EDA using Multiple Probabilistic Models

RMMではHISと同様に集団毎に多様性を変化させる。集団の多様性は、確率モデル構築に用いる正規分布のパラメータである分散 σ_i^2 に依存する。解探索が進むにつれ良好な評価値を持つ集団は収束していくため、分散 σ_i^2 の値は徐々に小さくなる。集団の多様性を調節する方法には、分散 σ_i^2 を直接設定することが考えられるが、設計変数によってスケールが異なる問題においては困難である。分散 σ_i^2 の小さくなる速度は学習率 α で調節が可能であるため、RMMでは各確率モデルで異なる学習率 β_i を用いることで、多様性の異なる集団を構成する。そして、多様性の異なる集団からPBILcの更新則を用いて分布推定を行うことで、効率的な解探索を行う。

3.2 独立成分分析

提案アルゴリズムでは、良好な探索点に対して独立成分分析を行う機能をRMMに付加する。独立成分分析の手法には、主成分分析 (PCA: Principal Component Analysis) を用いる^{6, 8)}。

提案アルゴリズムでは、良好な評価値を持つ探索点をアーカイブに保存し、そのアーカイブ内に保存された探索点群を用いて独立成分分析を行う。アーカイブを用いる理由は、少ない探索点数で解探索を行う場合であっても、十分なサン

プル数を用いて独立成分分析を行うためである。提案アルゴリズムにおける独立成分分析の手順を下記に示す。下記の操作は各確率モデルで独立で行う。

1. アーカイブの更新

集団をアーカイブに保存する。アーカイブに決められたサイズ以上の探索点が存在する場合、評価値が低い探索点を削除する。アーカイブに保存された探索点群を $E(t)$ とする。

2. 良好な集団を選択

集団から良好な集団 $X(t)$ を選択し、その平均ベクトル $T(t)$ を計算する。そして、 $X(t)' = X(t) - T(t)$ を行うことで、 $X(t)$ の平均を0にする。

3. $E(t)$ の共分散行列の計算

$E(t)$ の共分散行列から固有ベクトル v_1, \dots, v_n を求める。そして、座標軸変換のための行列 $V(t) = [v_1, v_2, \dots, v_n]$ を作成する。

4. $V(t)$ による $X(t)$ の無相関化

$X(t)'$ に対して右から $V(t)$ を掛けることで、 $X(t)'$ を無相関化する。無相関化した $X(t)'$ を $Y(t)$ とする。

5. 分布推定と探索点生成

$Y(t)$ に対しPBILcの更新則に従った分布推定を行う。そして、新しい探索点の生成を行う。新しい探索点群を $Y(t+1)$ とする。

6. 探索点の相関化

$Y(t+1)$ に右から $V(t)$ の逆行列を掛け、 $T(t)$ を足すことで、相関を元に戻す。

3.3 アルゴリズム

提案アルゴリズムの疑似コードをFig. 3に示す。提案アルゴリズムのパラメータは、確率モデルと集団の組の数 L 、1つの確率モデルが生成する探索点数 C 、切捨て率 K 、最大学習率 β_1 、最小学習率 β_L 、アーカイブサイズ A_{size} の6つである。

まず提案アルゴリズムは L 個の集団 X_1, \dots, X_L と L 個の正規分布に基づく確率モデル $p_1(x), \dots, p_L(x)$ を用意し、それぞれ初期化を行う。集団の初期化として、各集団に一様分布で生成した探索点を用いる。確率モデルの初期化として、平均 μ_i は各初期集団の最良探索点の設計変数値とした。分散 σ_i^2 は、提案アルゴリズムの探索序盤は探索

-
1. 設計空間領域に一様分布で探索点を生成し、初期集団 X_1, \dots, X_L とする
 2. 初期集団 X_1, \dots, X_L の評価値を計算
 3. 確率モデル p_1, \dots, p_L の初期化。 μ_i には各集団の最良探索点の設計変数を代入する。 σ_i^2 には各座標の定義域の大きさの $1/4$ の2乗を代入する
 4. 各確率モデルの学習率 β_1, \dots, β_L を計算
 5. 終了条件を満たすまで以下の手順を繰り返す
 - (1) $l = 1$
 - (2) 確率モデル p_l 毎に以下の手順を行う。
 - i. 集団 X_{l-1}, X_l, X_{l+1} から良好な探索点を選択
 - ii. 3.2節の独立成分分析に従い、次世代の集団 X_l を生成
 - iii. 集団 X_l の評価値を計算
 - iv. 集団 X_l を現世代の集団と置換
 - v. $l = l + 1$
-

Fig. 3 提案アルゴリズムの疑似コード

領域を広くしたいため、十分に大きな値である必要がある。今回は各座標の定義域の大きさの $1/4$ の2乗とした。分散 σ_i^2 は最適化対象に応じて自由に設定可能である。

次に、各集団で多様性を変化させるため、多様性を最も低くしたい確率モデル p_1 には最大学習率 β_1 を、多様性を最も高くしたい確率モデル p_L には最小学習率 β_L を設定する。その他の確率モデル p_2, \dots, p_{L-1} の学習率は値が順に小さくなるように設定する。今回は式(3)に示す通り等比的に値が変化するように設定した。

$$\beta_l = \beta_1 \cdot \left(\frac{\beta_L}{\beta_1} \right)^{\frac{l-1}{L-1}} \quad (3)$$

次に集団 X_{l-1}, X_l, X_{l+1} から $3 \times C \times K$ 個の探索点を評価値の高い順に選択する。ただし、 $l = 1$ の場合は X_1 と X_2 から、 $l = L$ の場合は X_{L-1} と X_L から $2 \times C \times K$ 個を選択する。そして、3.2節に従い、独立成分分析、確率モデルの更新、新しい探索点群の生成を順に行う。最後に、新しい探索点群と現在の集団との置換を行う。

4 数値実験

4.1 設定

提案アルゴリズムの性能を評価するために数値実験を行う。性能比較のために既存のEDAであるPBILcと分散確率モデル遺伝的アルゴリズム (DPMBGA: Distributed Probabilistic Model-Building Genetic Algorithm) ⁸⁾ を用いる。DPMBGAは提案アルゴリズムと同様に集団を分割して探索を行い、さらに独立成分分析を用い

Table 1 提案アルゴリズムのパラメータの組合せ

確率モデルと集団の組の数 L	4, 6, 8, 10
1つの確率モデルが生成する探索点数 C	5, 10, 15, 20, 25, 30
切捨て率 K	0.750, 0.667, 0.500 0.333, 0.250, 0.200
最大学習率 β_1	0.001, 0.002, \dots , 0.009 0.010, 0.020, \dots , 0.090 0.100, 0.150, \dots , 0.500
最小学習率 β_L	最大学習率 β_1 と同じ ただし, $\beta_1 > \beta_L$
アーカイブサイズ A_{size}	20, 40, 60, 80, 100

Table 2 PBILcのパラメータの組合せ

探索点数 P	20, 40, \dots , 300
切捨て率 K	Table 1の切捨て率 K と同じ
学習率 α	Table 1の最大学習率 β_1 と同じ

て解探索を行うという特徴を持つ。また、性能評価値の参考として、性能の高い実数値GAであるREX^{star7)}の結果も示す。

連続テスト関数としてはTable 4に示す関数を用いた。次元数は $n = 20$ である。評価計算回数が 2.0×10^6 回に達する前に、最良探索点の評価値が 1.0×10^{-7} 以下の値になったとき、最適解を発見したと判定する。

実験と同時に提案アルゴリズムとPBILcのパラメータチューニングを行う。提案アルゴリズムのパラメータの組合せをTable 1に示す。PBILcについては、可能な限り提案アルゴリズムと同等のパラメータになるようにTable 2の通りに設定した。DPMBGAのパラメータについては、参考文献⁸⁾に従いTable 3の通りに設定した。

実験では、全てのパラメータの組合せで各20試行を行い、20試行とも最適解を発見した組合せの中から、平均評価計算回数が最も少ないものを選出する。

4.2 結果

実験結果をTable 5に示す。REX^{star}の結果は参考文献⁷⁾からの引用である。参考文献⁷⁾で用いられていない連続テスト関数の結果については、(No Data)と記述した。また、“-”は最適解を発見できなかったことを示している。また、REX^{star}の結果は30試行の平均評価計算回数である。

Table 5の結果より、提案アルゴリズムは6つの連続テスト関数の内、5つの関数で一番良い性能を示している。また、提案アルゴリズムと

Table 3 DPMBGAのパラメータ

探索点数	512
切捨て率	0.250
エリート個体数	1
島数	32
移住率	1/16
移住間隔	5
アーカイブサイズ	100
拡大率	2.000
突然変異率	0.050

Table 5 結果

関数	提案手法	PBILc	DPMBGA	REX ^{star}
Ridg.	7.0×10^3	9.1×10^4	3.5×10^5	(No Data)
R _{Chain}	6.5×10^4	-	9.5×10^5	4.7×10^4
R _{Star}	2.5×10^4	-	4.5×10^5	5.5×10^4
Grie.	4.4×10^3	8.6×10^3	3.0×10^5	(No Data)
RoRa.	7.5×10^4	1.5×10^5	-	(No Data)
Scha.	7.4×10^4	9.4×10^4	-	7.7×10^4

PBILc, DPMBGAとを比較した場合、すべての関数において提案アルゴリズムの方が良い性能を示している。

5 考察

5.1 解探索履歴

提案アルゴリズムが独立成分分析を行うことで、ランドスケープを考慮した探索点が生成できることを確認するために、独立成分分析を行う場合と行わない場合における解探索履歴を比較する。

Fig. 4にRidge関数、Fig. 5にRosenbrock-Chain関数の解探索履歴を示す。ただし、次元数は $n = 2$ である。それぞれの図の背景は色が薄くなるほど評価値が高いことを示している。それぞれに用いたパラメータは、Table 5の結果を得たパラメータである。

Fig. 4とFig. 5より、提案アルゴリズムはランドスケープに沿った分布推定が行えていることがわかる。

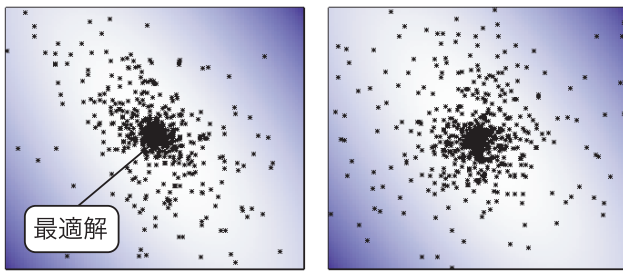
5.2 アーカイブサイズについて

アーカイブサイズと提案アルゴリズムの性能との関係を調べる。Table 1の組合せにおいて、アーカイブサイズ毎の最良の結果をTable 6に示す。Table 6中の(なし)は独立成分分析を行わないことを意味する。

Table 6より、単峰性関数ではアーカイブサイズは大きいほど性能は高い傾向にあるが、多峰性関数の場合は必ずしもそうでないことがわかった。

Table 4 連続テスト関数

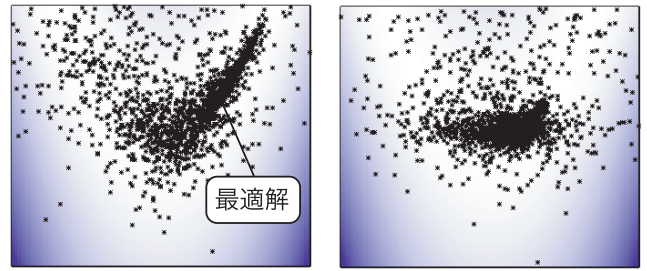
関数	数式 (n は次元数, S は設計変数空間)	性質	最適解
Ridge	$\sum_{i=1}^n \left(\sum_{j=1}^i x_j\right)^2$ $S = [-64, 64]^n$	直線の稜に沿った最適解領域がある単峰性関数	$x_i = 0.0$
Rosenbrock-Chain	$\sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1.0)^2)$ $S = [-2.048, 2.048]^n$	隣り合う変数間に強い依存関係がある単峰性関数	$x_i = 1.0$
Rosenbrock-Star	$\sum_{i=1}^{n-1} (100(x_1 - x_i^2)^2 + (x_i - 1.0)^2)$ $S = [-2.048, 2.048]^n$	x_1 と他変数の間に強い依存関係がある単峰性関数	$x_i = 1.0$
Griewank	$1.0 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \left(\cos\left(\frac{x_i}{\sqrt{i}}\right)\right)$ $S = [-512, 512]^n$	比較的弱い多峰性関数	$x_i = 0.0$
RotatedRastrigin-1.0	$10n + \sum_{i=1}^n ((x_i - 1.0)^2 - 10 \cos(2\pi(x_i - 1.0)))$ を60度回転 $S = [-5.12, 5.12]^n$	比較的強い多峰性関数	$x_i = 1.0$
Schaffer	$\sum_{i=1}^{n-1} (x_i^2 + x_{i+1}^2)^{0.25} \times (\sin^2(50(x_i^2 + x_{i+1}^2)^{0.1}) + 1.0)$ $S = [-100, 100]^n$	円状に準最適解がある多峰性関数	$x_i = 0.0$



独立成分分析あり

独立成分分析なし

Fig. 4 解探索履歴 (Ridge関数)



独立成分分析あり

独立成分分析なし

Fig. 5 解探索履歴 (Rosenbrock-Chain関数)

Table 7 提案アルゴリズムの最良パラメータ

関数	L	C	K	β_1	β_L	A_{size}
Ridg.	4	5	0.500	0.300	0.250	100
R _{Chain}	10	5	0.500	0.450	0.400	100
R _{Star}	8	5	0.500	0.450	0.400	100
Grie.	4	5	0.333	0.450	0.200	100
RoRa.	4	5	0.333	0.010	0.009	40
Scha.	4	5	0.667	0.070	0.060	60

単峰性関数でアーカイブサイズが大きいほど性能が高い理由は、アーカイブサイズは大きい方が関数のランドスケープをより精度良く推定できるためであると考えられる。

多峰性関数において、アーカイブサイズと性能に相関が見られない理由は、解探索途中ではアーカイブに保存されている探索点は局所解である可能性が高いため、大局的なランドスケープを推定できないためであると考えられる。

5.3 パラメータについて

本稿ではアーカイブサイズ以外のパラメータについて考察する。Table 5を得た最良のパラメータをTable 7に示す。

Table 7より、Griewank関数を除き、学習率 β_1

と β_L の値はTable 1において隣接する値であることがわかる。つまり、学習率同士の値の差は小さいと言える。学習率同士の差が小さい場合は、各集団は似たような集団になるため、差が大きい場合と比較して集団全体の多様性は小さくなる。予備実験より、単峰性関数における独立成分分析を用いない場合の最良パラメータの学習率の差は10倍程度であることを確認している。依存関係を持つ問題の場合、集団の収束を防ぐために多様性を維持する必要があるため、学習率の差を大きく設定する必要があるからである。しかし、独立成分分析を用いる場合は、学習率は近い値、もしくは同じ値に設定し、集団の収束を早めることで最適解により早く到達できると考えられる。

また、確率モデルと集団の組の数 L は、多峰性関数の場合は異なる箇所を探索する必要があるため、集団数を一定にした場合、 L の値は大きい方が良い解に到達することを確認している。しかし、今回の実験に用いた多峰性関数は、いずれも大局的には単峰性関数のランドスケープを持つため、少ない L の値でも最適解を発見できている。単峰性関数の場合は、 L の値を小さく設定することで世代交代を早め、かつ1つの確率モデ

Table 6 アーカイブサイズと性能の関係

関数	アーカイブサイズ					
	(なし)	20	40	60	80	100
Ridg.	5.4×10^4	4.2×10^4	1.7×10^4	9.0×10^3	7.3×10^3	7.0×10^3
R _{Chain}	1.3×10^5	1.0×10^5	1.2×10^5	1.4×10^5	1.2×10^5	6.5×10^4
R _{Star}	6.3×10^4	5.1×10^4	4.8×10^4	4.8×10^4	3.5×10^4	2.5×10^4
Grie.	5.8×10^4	1.2×10^4	1.1×10^4	4.7×10^3	5.5×10^3	4.4×10^3
RoRa.	1.1×10^5	1.2×10^5	7.5×10^4	1.7×10^5	1.5×10^5	1.2×10^5
Scha.	5.4×10^4	9.1×10^4	8.6×10^4	7.4×10^4	8.3×10^4	8.9×10^4

ルが生成する探索点数 C の値を大きく設定することで局所探索能力を高める方が良い。しかしTable 7において、 L の値が大きくても最適解を発見できている。この理由は、2つの学習率が共に大きい値であるため収束が早くなり、 L の値の影響が小さくなるからと考えられる。

6 まとめと今後の課題

本稿では、RMMをもとにして複数の確率モデルと独立成分分析を用いた新しいEDAを提案した。依存関係を持つ連続テスト関数を用いて、提案アルゴリズムを評価し、既存のEDAと比較して性能が高いことを示した。

しかしながら、多峰性関数においては、独立成分分析が有効に働かない場合があることもわかった。この理由は、アーカイブに保存された局所解が更新されないからであるため、参考文献⁸⁾にあるように、定期的にアーカイブを初期化することで、性能向上を図れると考えられる。

また、本稿で用いた連続テスト関数はすべて依存関係を持つものであったが、予備実験により、依存関係を持たない連続テスト関数では、独立成分分析を用いることで性能が悪化する場合があることも確認している。この対策として、参考文献⁴⁾で提案されている環境分散スキームを導入し、独立成分分析を行う確率モデルと行わない確率モデルの両方を用いることで、対象問題の依存関係の有無に関係なく、高い性能を発揮できると考えている。

参考文献

- 1) Takayuki Higo. *Research on the Importance Sampling Method for Evolutionary Algorithms Based on Probability Models*. PhD thesis, Tokyo Institute of Technology, 2008.
- 2) Takayuki Higo and Keiki Takadama. Maintaining multiple populations with different diver-

sities for evolutionary optimization based on probability models. *Information and Media Technologies*, Vol. 3, No. 2, pp. 362–374, 2008.

- 3) P. Larranaga and J. A. eds. Lozano. *Estimation of Distribution Algorithm*. Kluwer Academic Publishers, 2002.
- 4) Mitsunori Miki, Tomoyuki Hiroyasu, Mika Kaneko, and Kazuyuki Hatanaka. A parallel genetic algorithm with distributed environment scheme. *Man and Cybernetics Conference SMC' 99*, 1999.
- 5) M. Sebag and A. Ducoulombier. Extending population-based incremental learning to continuous search spaces. *Proc. of the Parallel Problem Solving from Nature - PPSN V*, pp. 418–427, 1998.
- 6) 高橋仁人, 喜多一. 実数値遺伝的アルゴリズムのための独立成分分析を用いた交叉の提案. 第13回自律分散システムシンポジウム資料, pp. 245–250, 2001.
- 7) 小林重信. 実数値GAのフロンティア. 人工知能学会論文誌 24巻 1号 SP-N, pp. 147–162, 2009.
- 8) 廣安知之, 三木光範, 下坂久司, 佐野正樹, 筒井茂義. 分散確率モデル遺伝的アルゴリズム. 情報処理学会論文誌「数理モデル化と応用」, Vol. 45, No. SIG2, pp. 56–65, 2004.
- 9) 中尾昌広, 廣安知之, 三木光範, 横内久猛. 複数の確率モデルを用いた実数値分布推定アルゴリズムの提案. 人工知能学会論文誌論文特集 - 進化計算のフロンティア, 2010. (投稿中).