

目的関数の加法分解性および差分の符号独立性に基づく 実数値遺伝的アルゴリズムのリンケージ同定

手塚 大[†], 棟 朝 雅 晴^{††} 赤 間 清^{††}

最適化問題が、独立に最適化可能な複数の部分問題で表される場合、部分問題ごとに扱うことによって解の探索を効率化できる。遺伝的アルゴリズム(GA)では、この部分問題を構成する遺伝子座の集合をリンケージグループといい、リンケージグループを識別する手法をリンケージ同定という。本論文では、実数値GAのリンケージを明確に定義する。この定義に基づいてリンケージの識別を行う2つのリンケージ同定手法、LINC-RとLIDI-Rを提案する。LINC-Rは目的関数の加法分解性、LIDI-Rは差分の符号独立性に基づいてリンケージの有無を判定する。これらの手法は直接的にリンケージを識別するため、効率的にリンケージ同定ができる。

Linkage Identification for Real-coded Genetic Algorithms Based on Additive Decomposability and Difference Signature Independency of Objective Function

MASARU TEZUKA,[†] MASAHARU MUNETOMO^{††}
and KIYOSHI AKAMA^{††}

In the case that a problem is decomposable to a number of sub-problems which can be optimized independently, the problem is solved effectively by optimizing sub-problems separately. In optimization problems by means of genetic algorithms, a set of loci of which each sub-problem consists is called linkage group. Linkage identification is the method which recognizes linkage groups. In this paper, we define the linkage of Real-Coded GAs clearly. Then we propose two new linkage identification methods, LINC-R and LIDI-R, directly based on the definition. LINC-R is based on additive decomposability and LIDI-R is based on independency of the signature of difference of an objective function. These methods effectively identify linkages.

1. はじめに

最適化問題が、独立に最適化可能な複数の部分問題で表される場合、部分問題ごとに扱うことによって解の探索を効率化できる。

遺伝的アルゴリズム(GA)では、2つの遺伝子座上の遺伝子の値の依存関係をリンケージと呼び、リンケージのある遺伝子座の集合をリンケージグループという。このリンケージグループが構成する部分問題の解がビルディングブロックとなる。GAではビルディングブロックを交換することによって最適化が行われ

る。リンケージグループを適切に識別し、交叉などの操作でビルディングブロックが破壊されないようにすることでGAの探索性能が向上する。リンケージ同定とはリンケージグループを識別することである。

バイナリGAでは、環状の染色体に特殊な交叉を適用することで動的にリンケージを生成するLLGA(Linkage Learning GA)¹⁾、集団内の個体の分布をもとに間接的にリンケージを同定するBOA(Bayesian Optimization Algorithm)²⁾、遺伝子座間の非線形性、非単調性、エピスタシス尺度をもとに直接的にリンケージを同定するLINC(Linkage Identification by Non-linearity Check)³⁾、LIMD(Linkage Identification by non-Monotonicity Detection)⁴⁾、LIEM(Linkage Identification with Epistasis Measure)⁵⁾など多くの研究がされている。

実数値GAは実数のベクトルを染色体として持つGA⁶⁾で、様々な最適化問題への適用が研究されてい

[†] 北海道大学大学院工学研究科

Graduate School of Engineering, Hokkaido University

^{††} 北海道大学情報基盤センター

Information Initiative Center, Hokkaido University

現在、株式会社日立東日本ソリューションズ

Presently with Hitachi East Japan Solutions, Ltd.

る⁷⁾⁻⁹⁾。しかし、実数値 GA のリンケージ同定の研究は Tsutsui らによる PICI (Piecewise Interval Correlation by Iteration)¹⁰⁾⁻¹²⁾ が報告されているが、まだ数が少ない。大規模な実数最適化問題を GA で効率的に解くためには、実数値 GA 向けの性能の高いリンケージ同定手法が必要となる。

本論文では、2章で既存の実数値 GA のリンケージ同定手法である PICI について簡単に説明し、その課題を検討する。続いて、3章で実数値 GA のリンケージを新たに定義しなおし、この定義に基づき、直接的にリンケージ同定を行う方法を提案する。続いて数値実験を行い、提案手法のリンケージ同定性能を4章で、提案手法を用いた GA による最適化性能を5章で報告する。

2. 従来の実数値 GA リンケージ同定手法

ベクトル $\mathbf{x} \in \mathbb{R}^n$ とし、目的関数 $f: \mathbb{R}^n \mapsto \mathbb{R}$ の最適化問題を考える。実数値 GA ではこのベクトルが染色体となる。 \mathbf{x} の i 番目の要素 x_i が遺伝子の値であり、 i が GA の遺伝子座である。 $x_i \in X_i, i = 1, \dots, n$ とし、 $X_i \subseteq \mathbb{R}$ は定義域である。

Tsutsui¹¹⁾ らは、目的関数が

$$f(x_1, \dots, x_n) = g_1(x_1, \dots, x_k) + g_2(x_{k+1}, \dots, x_n) \quad (1)$$

のように、異なる遺伝子座を変数に持つ複数の部分関数に加法的に分解できる場合に、各部分関数の解をビルディングブロックととらえた。1つの部分関数に属する遺伝子座がリンケージグループである。これを本論文では「加法分解性の意味でのリンケージ」ということにする。Tsutsui らは、加法分解性の意味でのリンケージの同定を目的として PICI を提案した。

GA では最適化が進むにつれて、個体の分布が目的関数のランドスケープを反映するようになる。図1は実線が目的関数の等高線すなわちランドスケープを、点が GA の個体を示している。PIC I では個体の分布から、各遺伝子座の相関係数を計算し、相関が高い遺伝子座にリンケージがあると判定する。単に相関係数を求めると、二次以上の関係を求めることができない。そこで、実際には定義域を複数の小領域に分割し、小領域ごとに相関係数を求め、それらの重みつき和を Piecewise Interval 相関係数としている。図1の(a)では遺伝子座間に相関がないためリンケージがないと判定される。また、(b)のように各軸に対して斜めの谷がある場合に相関係数が大きくなり、リンケージがあると判定される。

PICI には2つのパラメータがある。1つは定義域

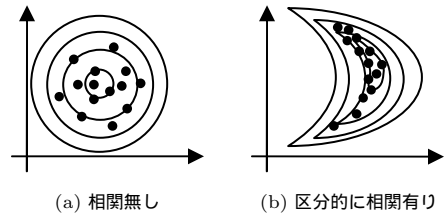


図1 目的関数のランドスケープと個体の分布

Fig. 1 Landscape of objective function and distribution of individuals.

を小領域に分割する際に用いられる N_{\min} で、小領域内の個体数がこの値以下にならないように、分割が行われる。もう1つは閾値 K_p で、Piecewise Interval 相関係数が K_p 以上の場合に、リンケージがあると判断される。

PICI には LISS (Linkage Identification with Single Stage evolution) と LIMS (Linkage Identification with Multi-Stage evolution) の2つの適用方法がある¹²⁾。LISS は各世代の集団の分布を用いてリンケージ同定を行う方法で、1世代ごとにリンケージ同定を行う。直前の世代で同定されたリンケージ情報を使い、リンケージグループごとにシンプレックス交叉を適用して次世代を生成する。LIMS は最適化を、初期ステージ、リンケージ学習ステージ、探索ステージの3段階に分ける。初期ステージ(第1世代から第 l_{begin} 世代)では通常の GA によって最適化を行いランドスケープの構築が行われる。続いてリンケージ学習ステージ(第 l_{begin} 世代から l_{end} 世代)でさらに通常の GA により最適化を行う。この学習ステージ内に生成されたすべての個体の分布から相関係数を求め、リンケージ同定を行う。同定されたリンケージ情報を用いて、探索ステージ(l_{end} 世代以降)でリンケージグループごとにシンプレックス交叉を適用し、最終的な最適解の探索が行われる。Tsutsui らによって LIMS のほうが最適化性能が高いことが報告されている。

PICI は個体の分布をもとに遺伝子座間の相関係数を求めてリンケージ同定を行うが、ランドスケープは複数の部分関数の影響を受けるため、分布にはノイズが多い。また、PIC I は目的関数を加法的に分解することを目的として考案されたが、遺伝子座間に相関があることと、目的関数が加法的に分解可能であることには直接的な関係はないため、リンケージ同定効率が低いと考えられる。

また、PIC I には2つのパラメータ N_{\min}, K_p および、LIMS を用いる場合にはさらにステージを分割

するパラメータである l_{begin} と l_{end} がある。これらの値は経験的にあるいは試行錯誤的に決定する必要がある。

3. 新しいリンケージの定義とリンケージ同定手法の提案

PICI の問題点を解決するために、実数値 GA のリンケージを新しく定義し直す。さらにこの定義をもとに直接的にリンケージ同定を行う手法を提案する。

3.1 リンケージの定義

目的関数 f の最大化を 2 つの遺伝子座に着目して考える。これ以外の遺伝子座は数式上では省略する。

ある遺伝子座 j の値を x_j とした場合に、 f が最大値をとるような i 番目の遺伝子座の値の集合 $X_i^*(x_j) \subseteq X_i$ は次式のように表される。

$$X_i^*(x_j) = \left\{ x \mid \forall y \in X_i, f(x, x_j) \geq f(y, x_j) \right\} \quad (2)$$

ただし、 $f(a, b)$ の引数 a は i 番目の遺伝子座の値、 b は j 番目の遺伝子座の値を取り、それ以外の遺伝子座の引数は省略してある。

この集合 $X_i^*(x_j)$ について、以下の式 (3) がつねに成り立つ。また、式 (4) と式 (5) のいずれかがつねに成り立つ。

$$\forall x_i^1, x_i^2 \in X_i^*(x_j), f(x_i^1, x_j) = f(x_i^2, x_j) \quad (3)$$

$$X_i \setminus X_i^*(x_j) = \emptyset \quad (4)$$

$$\forall x_i^* \in X_i^*(x_j), \forall x_i \in X_i \setminus X_i^*(x_j), f(x_i^*, x_j) > f(x_i, x_j) \quad (5)$$

式 (4) は j 番目の遺伝子座の値を x_j とした場合に、 i 番目の遺伝子座の定義域すべてで目的関数と同じ値を持つ場合に成り立ち、式 (5) はそれ以外の場合に成り立つ。

ここで、次式 (6) を満たす場合、遺伝子座 i と遺伝子座 j について別々に最大化し、その値を組み合わせることで f を最大化できる。

$$\begin{aligned} & [\forall x_j^1, x_j^2 \in X_j, X_i^*(x_j^1) = X_i^*(x_j^2)] \\ & \text{かつ} \\ & [\forall x_i^1, x_i^2 \in X_i, X_j^*(x_i^1) = X_j^*(x_i^2)] \end{aligned} \quad (6)$$

式 (6) が満たされない場合には、遺伝子座 i と j について同時に f を最大化する必要がある。このような、値の組合せを同時に最適化しないと解が得られない、すなわち部分問題に分解できない遺伝子座の組を、実数値 GA のリンケージと定義する。これを特に他の定義と区別する場合に、「独立最適化可能性の意味で

のリンケージ」ということにする。

3.2 LINC-R

独立最適化可能性を満たす最も簡単な状況は、加法分解性がある場合である。

加法分解性が成り立つ場合、ある遺伝子座についての目的関数の差分係数は、もう一方の遺伝子座の値によらない。つまり、 $\forall x_j^1, x_j^2 \in X_j$ について式 (7) が成り立つ。

$$\frac{\Delta f(x_i, x_j^1)}{\Delta x_i} = \frac{\Delta f(x_i, x_j^2)}{\Delta x_i} \quad (7)$$

この関係を用いてリンケージ同定を行う方法として LINC-R (Linkage Identification based on Nonlinearity Check for Real-Coded GAs) を提案する。

まず基準となる点 (x_i^1, x_j^1) 、 $x_i^1 \in X_i$ 、 $x_j^1 \in X_j$ を用意する。この点を変量 Δx_i と Δx_j で変位させ、これによってできる 4 点 (x_i^1, x_j^1) 、 (x_i^1, x_j^2) 、 (x_i^2, x_j^1) 、 (x_i^2, x_j^2) の目的関数値を求める。ここで $x_i^2 = x_i^1 + \Delta x_i$ 、 $x_j^2 = x_j^1 + \Delta x_j$ であり、また変量は、 $x_i^2 \in X_i$ 、 $x_j^2 \in X_j$ となるように選ぶ。

この 4 点の目的関数値を用いて、

$$\frac{\Delta f(x_i, x_j^1)}{\Delta x_i} = f(x_i^2, x_j^1) - f(x_i^1, x_j^1)$$

$$\frac{\Delta f(x_i, x_j^2)}{\Delta x_i} = f(x_i^2, x_j^2) - f(x_i^1, x_j^2)$$

を求め、2 つの差分が一致しない場合に遺伝子座 i と j にリンケージがあると判断する。実際には誤差を許して以下の条件を用いる。

$$\left| \left\{ f(x_i^2, x_j^1) - f(x_i^1, x_j^1) \right\} - \left\{ f(x_i^2, x_j^2) - f(x_i^1, x_j^2) \right\} \right| > \epsilon \quad (8)$$

ここで加法分解性の意味でリンケージがあることを示すには $\exists x_i^1, x_i^2 \in X_i, \exists x_j^1, x_j^2 \in X_j$ で条件 (8) が成立することを示せばよいが、ないことを示すには $\forall x_i^1, x_i^2 \in X_i, \forall x_j^1, x_j^2 \in X_j$ について条件 (8) が成立しないことを確認する必要がある。つまりリンケージ同定の探索空間の大きさは $|X_i|^2 \times |X_j|^2$ であり、本来の目的関数の探索空間の大きさ $|X_i| \times |X_j|$ の二乗となっている。

リンケージ同定の目的はリンケージを厳密に同定することではなく、可能な場合には部分問題に分解し、効率的に最適化を行うことである。つまり、リンケージの検出が困難な問題を対象とはせず、特徴的な解の構造を持ち、少ない探索回数でリンケージ同定できるような問題を対象とする。したがって、LINC-R では全探索空間を調べることはせず、複数の点を用いてリンケージ同定を行う。この点の数を、リンケージ同定

のための集団サイズと呼ぶ．ここで用いる点の数，すなわちリンケージ同定の集団サイズとリンケージ同定の性能の関係については後で述べる．

ところで，LINC-R では変数の定義域を $x_i \in X_i$, $i = 1, \dots, n$, $X_i \subseteq \mathcal{R}$ としている．また，基準点からの変量 Δx_i をランダムに選んでいる．ここで $X_i = \{0, 1\}$, $\Delta x_i = 1$ とし，基準点から変量による変位で新たな点を作る演算 $x_i + \Delta x_i$ をモジュロ 2 の演算とすると，バイナリ GA のリンケージ同定手法である LINC と同じものとなる．つまり，LINC-R は LINC を実数値に拡張し一般化した手法である．

3.3 LIDI-R

遺伝子座の間に加法分解性の意味でリンケージがある場合でも，各遺伝子座を独立に最適化できる場合がある．

遺伝子座 i の任意の 2 点間の目的関数の大小関係が，遺伝子座 j の値によらない場合， x_i は x_j によらず f を最大化できる．すなわち，

$$\begin{aligned} & \forall x_i^1, x_i^2 \in X_i, \forall x_j^1, x_j^2 \in X_j, \\ & \text{sgn}(f(x_i^1, x_j^1) - f(x_i^2, x_j^1)) \\ & = \text{sgn}(f(x_i^1, x_j^2) - f(x_i^2, x_j^2)) \\ \Rightarrow & \forall x_j^1, x_j^2 \in X_j, \quad X_i^*(x_j^1) = X_i^*(x_j^2) \end{aligned} \quad (9)$$

である．ここで，

$$\text{sgn}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \quad (10)$$

である．付録 A.1 に式 (9) の証明を付す．

この関係が遺伝子座 i と j の間に相互に成り立つ場合に式 (6) が成立し，各遺伝子座を個別に最適化できる．このような遺伝子座の関係を，目的関数の差分の符号独立性があるという．

この関係を用いてリンケージ同定を行う手法として LIDI-R (Linkage Identification by Difference signature Independency detection for Real-Coded GAs) を提案する．

LINC-R と同様に 4 点 (x_i^1, x_j^1) , (x_i^1, x_j^2) , (x_i^2, x_j^1) , (x_i^2, x_j^2) の目的関数値を求め，

$$\begin{aligned} & \left[\text{sgn}(f(x_i^1, x_j^1) - f(x_i^2, x_j^1)) \right. \\ & \quad \left. = \text{sgn}(f(x_i^1, x_j^2) - f(x_i^2, x_j^2)) \right] \\ \text{かつ} & \left[\text{sgn}(f(x_i^1, x_j^1) - f(x_i^1, x_j^2)) \right. \\ & \quad \left. = \text{sgn}(f(x_i^2, x_j^1) - f(x_i^2, x_j^2)) \right] \end{aligned} \quad (11)$$

が成り立たない場合にリンケージがあると判断する．

加法分解性が成り立つ場合には，差分の符号独立性も成り立つが，逆は成り立たない．式 (12) の f_{Sp^2} は加法分解性が成り立たないが差分の符号独立性がある関数の例である．

$$f_{\text{Sp}^2}(x_1, \dots, x_n) = \left(\sum_{i=1}^n (x_i - 1)^2 \right)^2 \quad (12)$$

f_{Sp^2} は式 (6) が成り立つため，各遺伝子座を別々に最適化可能であるが，加法分解性がないため，LINC-R では誤ってリンケージがあると判断する．LIDI-R を用いるとリンケージがないことを正しく判断し，部分問題に分解できる．

3.4 リンケージ同定能力の検討

LINC-R または LIDI-R では，リンケージの有無をすべての遺伝子座の組で調べ，リンケージのある遺伝子座の集合をリンケージグループとする．

各個体を基準となる点とし，リンケージ同定の対象となる遺伝子座のペアについて変量 Δx_i , Δx_j を用いて基準点を含めて 4 点を生成し，LINC-R または LIDI-R でリンケージの有無を検査する．これを集団中のすべての個体について行う．このステップを図 2 に示す．リンケージの同定での目的関数の評価回数は，

$$\left(\frac{3n(n-1)}{2} + 1 \right) m \quad (13)$$

となる． $m (= |P|)$ はリンケージ同定に用いる個体の数であり，その決定方法を検討する．

リンケージがある遺伝子座の組の集合を

$$\pi = \{(i, j) \mid \text{遺伝子座 } i \text{ と } j \text{ にリンケージあり}\} \quad (14)$$

とする．遺伝子座 i と j のリンケージ同定を 1 個体で行う場合に，同定に失敗する確率を $\rho_{(i,j)}$ とする．

1. 個体の集団 P をランダムに初期化．
2. for each $x \in P$
3. for $i = 1$ to $n - 1$
4. for $j = i + 1$ to n
5. 一様乱数 $\tilde{x}_i \in X_i$ の生成．
6. 一様乱数 $\tilde{x}_j \in X_j$ の生成．
7. $f_{11} = f(x_1, \dots, x_i, \dots, x_j, \dots, x_n)$
8. $f_{21} = f(x_1, \dots, \tilde{x}_i, \dots, x_j, \dots, x_n)$
9. $f_{12} = f(x_1, \dots, x_i, \dots, \tilde{x}_j, \dots, x_n)$
10. $f_{22} = f(x_1, \dots, \tilde{x}_i, \dots, \tilde{x}_j, \dots, x_n)$
11. LINC-R または LIDI-R による検査
12. end for
13. end for
14. end for

図 2 実数値 GA のリンケージ同定の手順

Fig. 2 Procedure of linkage identification for Real-Coded GA.

m 個体でリンケージ同定を行う場合に、この遺伝子座の組の同定に成功する確率は、 $1 - \rho_{(i,j)}^m$ であるから、 m 個体ですべてのリンケージの組の同定に成功する確率は、

$$Pr(m) = \prod_{(i,j) \in \pi} (1 - \rho_{(i,j)}^m) \quad (15)$$

となる。

したがって、リンケージのある遺伝子座の組の集合 π と、その同定に失敗する確率 $\rho_{(i,j)}$ が分かれば、式 (15) から、個体数 m とその場合のリンケージ同定成功率 $Pr(m)$ が分かる。

簡単のために、すべてのリンケージの組の同定失敗確率が等しく $\rho_{(i,j)} = \rho, \forall (i,j) \in \pi$ であるとすれば、確率 Pr 以上でリンケージ同定が成功する個体数は次式で求められる。

$$m \geq \frac{\ln(1 - Pr^{1/|\pi|})}{\ln \rho} \quad (16)$$

LIDI-R では、同定に失敗する確率 $\rho_{(i,j)}$ を求めるのは難しいが、LINC-R の場合には容易に求められる場合がある。これを以下で検討する。

次式のような $\mathbf{x} = (0, \dots, 0)$ で最大値 1.0 を持ち、 $\mathbf{x} = (1, \dots, 1)$ で準最大値 λ を持つ n 次元だまし問題を考える。

$$f_{\text{decep}}(x_1, \dots, x_n) = \begin{cases} 1 - \frac{1}{r} \left(\sum_{i=1}^n x_i \right), & \sum_{i=1}^n x_i \leq r \\ \frac{\lambda}{n-r} \left(\sum_{i=1}^n x_i - r \right), & \sum_{i=1}^n x_i > r \end{cases} \quad (17)$$

$n = 2, \lambda = 0.8, r = 0.8$ の例を図 3 に示す。

この問題は x_1, \dots, x_n を加法的に分解できない、すなわちリンケージがある。しかし、遺伝子座 i と j をリンケージ同定するとき、LINC-R で用いる 4 点が

すべて $x_i > r$ かつ $x_j > r$ の領域内となってしまう、あるいはすべて $x_i \leq r$ かつ $x_j \leq r$ の領域内となってしまうと、加法的分解性があると誤って判断してしまう。この誤って判断する確率が $\rho_{(i,j)}$ であり、これは定義域の大きさと、 r から求めることができる。

だまし問題の例では、定義域内に 2 つの加法的に分解可能な領域 ($\forall i, x_i > r$ と $\forall i, x_i \leq r$) があるが、より一般的に、加法的に分解可能な領域が k 個ある場合を考える。これらの領域を $A_u, u = 1, \dots, k$ とする。

さらに、加法的に分解できない領域もあるとする。加法分解できない領域も複数ありうるが、ここではそれらを区別して扱わないので、1 つの加法的に分解できない領域として扱い、これを B とする。

目的関数の定義域がこれらの $k+1$ 個の共通部分を持たない部分領域に分割できるとする。

$$\left(\bigcup_{u=1}^k A_u \right) \cup B = X \quad (18)$$

$$A_u \cap A_v = \phi, \quad u, v = 1, \dots, k, \quad u \neq v \quad (19)$$

$$A_u \cap B = \phi, \quad u = 1, \dots, k \quad (20)$$

LINC-R では $(x_i^1, x_j^1), (x_i^2, x_j^2), (x_i^1, x_j^2), (x_i^2, x_j^1)$ の 4 点を用いてリンケージの有無を調べる。この 4 点がすべて同一の加法的に分解可能な領域 A_u 内となってしまうとリンケージ同定ができない。

定義域にしめる領域 u の割合を $a_u = |A_u|/|X|$ とする。実際には 4 点が矩形の頂点となるようにランダムに生成するが、分析を簡単にするため、ここでは矩形の頂点という条件を無視して考えると、リンケージ同定に失敗する、すなわちランダムに与えられた 4 点がともに同一の領域に入る確率は $\rho = \sum_{u=1}^k a_u^4$ である。

これを式 (15), (16) に代入すると、 m 個体で加法的分解の意味でリンケージを検出可能な確率 $Pr(m)$ は次式で表される。

$$Pr(m) = \left(1 - \left(\sum_{u=1}^k a_u^4 \right)^m \right)^{|\pi|} \quad (21)$$

また、この式より、ある確率 Pr 以上で LINC-R によるリンケージ同定を成功させるために必要な個体の数は

$$m \geq \frac{\ln(1 - Pr^{1/|\pi|})}{\ln \left(\sum_{i=u}^k a_u^4 \right)} \quad (22)$$

となる。

LINC-R には許容誤差 ϵ とリンケージ同定で用いる

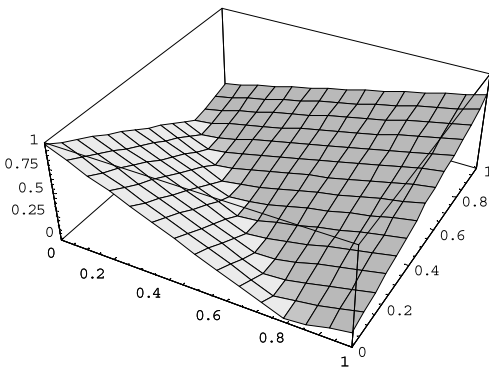


図 3 2 つの加法的分解可能な領域を持つ 2 次元だまし問題
Fig. 3 Two dimensional deceptive problem which has two additively decomposable areas.

個体数 m の 2 つのパラメータがある．目的関数にノイズがない場合には $\epsilon = 0$ とすることができる．また，個体数 m は式 (22) で求めることができる．ただし，加法的に分解可能な領域の割合が既知である必要がある．

LIDI-R のリンケージ同定能力は目的関数の形状に大きく依存するため，リンケージ同定可能な確率を求めることは困難である．しかし，1 個体でリンケージ同定に失敗する確率を求めることができれば，個体数を決定できる．この確率を求められない場合には，パラメータ m の値を試行錯誤的に設定する必要がある．

4. リンケージ同定能力の数値実験

4.1 個体数のリンケージ同定成功確率の確認

式 (17) のだまし関数で $n = 2, \lambda = 0.8$ とし，LINC-R と LIDI-R によるリンケージ同定が成功する確率を数値実験で確認した結果を図 4 に示す．一点鎖線が式 (21) による理論値，実線が LINC-R の実験値，点線が LIDI-R の実験値である．実験は 100 試行の平均値である．式 (21) は，LINC-R で用いる 4 点が矩形の頂点にあるという条件を無視して求めたものであるが，LINC-R の実験値に近い値となっている．この実験結果から LINC-R のパラメータ m を式 (22) で求められることが確認できる．なお，リンケージの組は 1 組なので $|\pi| = 1$ である．

4.2 PICI との比較

目的関数として Type I 関数， $R2n$ 関数を用いて PICI とのリンケージ同定能力の比較を行う．それぞれ次式で表される．これらは，Tsutsui らが用いた関数^{10),11)} である．

$$F_1(\mathbf{x}) = F_{Rsn}(x_1, \dots, x_T)$$

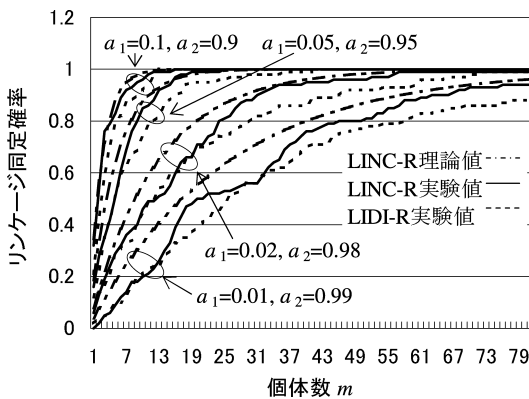


図 4 f_{decep} での個体数 m と LINC-R, LIDI-R 成功確率 $Pr(m)$

Fig. 4 LINC-R and LIDI-R success rate $Pr(m)$ with population size m on f_{decep} .

$$+ F_{Sp}(x_{T+1}, \dots, x_{T+L}) \tag{23}$$

$$F_{R2n}(\mathbf{x}) = \sum_{i=1}^n F_{Rsn}(x_{2i-1}, x_{2i}) \tag{24}$$

ここで， F_{Rsn} は式 (25) で表される T 次元 Rosenbrock 関数で，第 1 引数と他の引数との間にリンケージがある．また， F_{Sp} は式 (26) で表される L 次元 Sphere 関数で，引数間にリンケージはない．

$$f_{Rsn}(x_1, \dots, x_T) = \sum_{i=2}^T \left(100(x_1 - x_i^2)^2 + (x_i - 1)^2 \right), -2.048 \leq x_i < 2.048, i = 1, \dots, T \tag{25}$$

$$F_{Sp}(x_1, \dots, x_L) = \sum_{i=1}^L (x_i - 1)^2 \tag{26}$$

Type I 関数では $T = 2, L = 2$ および $T = 8, L = 20$ について， $R2n$ 関数では $n = 4$ について，すべてのリンケージが正しく同定されるまでの目的関数の評価回数を求めた．この結果を表 1 に示す．LINC-R, LIDI-R は，100 試行の平均値を示した．括弧内の数値は標準偏差である．

PICI の数値は，Tsutsui らの報告¹⁰⁾ の Fig. 8, 12, 17 (a) のグラフから読み取った数値である．これらのグラフは遺伝子座間の相関係数の変化の時系列グラフで，典型的な試行 (typical single run) での例として示されている．ここから，相関係数がリンケージの有無の判断の閾値である $K_p (= 0.25)$ 以上となった時点の目的関数の評価回数を読み取り，表に示した．

LINC-R と LIDI-R は，PICI と比較し，非常に少ない目的関数の評価回数ですべてのリンケージの同定に成功している．PICI の数値は 1 試行の値であるため直接の比較を行うことはできないが，LIDI-R での 100 試行で最も多くの評価回数が必要だった場合でも，それぞれ，190 回，31,309 回，1,033 回であった．このことから，PICI と比較して LINC-R, LIDI-R のリンケージ同定効率が高いことが分かる．

表 1 リンケージ同定に要する目的関数評価回数

Table 1 The number of function evaluation required to identify linkage.

	LINC-R	LIDI-R	PICI
Type I	19	52	5,000
$T = 2, L = 2$	(0)	(38)	
Type I	1,135	10,208	100,000
$T = 8, L = 20$	(0)	(5,322)	
$R2n$	85	442	12,000
$n = 4$	(0)	(209)	

数値実験で用いた目的関数は、目的関数を加法的に分解可能な領域が存在しない、すなわち、3.4 節での $k = 0$, $B = X$ である。このため、LINC-R では 1 個体で確実にリンケージ同定ができる。式 (13) より、1 個体でリンケージ同定する場合の目的関数の評価回数はそれぞれ、19 回、1,135 回、85 回でありこれは表 1 の数値に一致している。

以上をまとめると、LINC-R, LIDI-R が PICI と比較して効率的にリンケージ同定が可能である。また、LINC-R では目的関数を加法的に分解可能な領域の割合が得られれば、リンケージ同定に必要な個体数を理論的に求めることもできる。

5. 最適化能力の数値実験

5.1 実験システムの構成

実験で用いる交叉は、シンプレックス交叉¹³⁾を拡張した手法で、Tsutsui ら¹¹⁾によって考案されたものである。

シンプレックス交叉は複数の親ベクトルが張る単体内にランダムに子ベクトルを生成する。これを、リンケージグループごとにシンプレックス交叉を適用するように修正して用いる。つまり、親ベクトルの、リンケージグループ（部分空間）への射影ベクトルが張る単体内に子ベクトルを生成する。

世代交代には MGG (Minimal Generation Gap) モデル¹⁴⁾を、Tsutsui ら¹¹⁾と同様に多数の親個体から多数の子個体を生成するように修正して用いた。これは集団から M 個の親個体をランダムに非復元抽出で選択し、シンプレックス交叉により $C (= F_{MGG} \times M)$ 個の子個体を生成する。 F_{MGG} は子個体の生成数を決めるパラメータである。本論文では $F_{MGG} = 10$ とした。この親と子をあわせた $M + C$ 個の個体からトーナメント選択で M 個体を集団に戻す。

LINC-R と LIDI-R で用いる実数値 GA はリンケージ同定ステージと最適化ステージに分かれる。リンケージ同定ステージでは LINC-R または LIDI-R によってリンケージ同定を行う。

実験で用いる目的関数のリンケージは Rosenbrock 関数によるため、定義域全域で目的関数を加法的に分解できない、したがって、LINC-R によるリンケージ同定ステージの集団サイズは 1 とする。

LIDI-R ではリンケージ同定で用いる集団サイズを理論的に決定できない。そこで、目的関数の評価回数 50,000 回までをリンケージ同定に費やすこととし、式 (13) より、集団 P のサイズ m を逆算して設定した。したがって、解こうとする問題の次元数 n によって

集団サイズが異なる。 m は切り上げによって整数とする。また、目的関数の評価回数が 50,000 回となった時点で、リンケージ同定を終了する。なお、ここで 50,000 回とした理由は、比較で用いる PICI (LIMS) のリンケージ学習ステージが目的関数の評価 100,000 回までとしてあるが、LIDI-R はリンケージ同定効率が良いので、これよりも少ない回数として 50,000 を用いることとした。

最適化ステージでリンケージグループごとにシンプレックス交叉を適用し最適化を行う。最適化ステージでは、新たな個体の集団を作成し、この集団で最適化を行う。実数値 GA では理論的に最適化に適した集団サイズを決定する方法がないため、集団のサイズは、25, 50, 100, 150, 200, 250, 300 の 7 通りを用いた。実験結果には、最も良い結果の出た集団サイズでの値を示している。

比較で用いる PICI (LIMS) は集団の個体数が 500、リンケージ学習のパラメータ l_{begin} と l_{end} はそれぞれ目的関数の評価回数が 70,000 の時点と 100,000 回の時点である。

また、リンケージ情報なしの GA との比較も行うが、これはリンケージ同定を行わず、通常のシンプレックス交叉、すなわち染色体全体を対象とするシンプレックス交叉を行う GA である。

実験では、最適解に到達するか、目的関数の評価回数が 1,000,000 回になった時点で最適化を終了とした。最適解を (o_1, \dots, o_n) とし、 $\forall i, x_i \in [o_i - \Delta x/2, o_i + \Delta x/2]$ となったときに最適解に到達したと判断した。ここで $\Delta x = 0.001$ を用いた。この判断基準は Tsutsui ら^{11),12)} のものと同じである。

5.2 数値実験

目的関数として Type I, Type II, Type III 関数を用いた。Type I 関数は式 (23) である。Type II, Type III は次式で表され、関数値の最小化が目的である。

$$F_2(\mathbf{x}) = \sum_{i=1}^T F_{\text{Rsn}}(x_{2i-1}, x_{2i}) + F_{\text{Sp}}(x_{2T+1}, \dots, x_{2T+L}) \quad (27)$$

$$F_3(\mathbf{x}) = \sum_{i=1}^T F_{\text{Rsn}}(x_{2i-1}, x_{2i}) + \sum_{i=T+1}^{2T} F_{\text{Sp}^2}(x_{2i-1}, x_{2i}) \quad (28)$$

f_{Sp^2} は式 (12) で表される、加法的には分解できないが、差分の符号独立性がある関数である。

Type I, Type II は Tsutsui ら^{11),12)} が用いたも

表 2 Type I 関数の最適化結果

Table 2 Result of optimization of type I function.

		リンケージ情報なし			PICI (LIMS)		LINC-R					LIDI-R				
		100 試行			10 試行		100 試行					100 試行				
T	#Opt	MNE	STD	#Opt	MNE	#Opt	MNE	STD	#LI	pop	#Opt	MNE	STD	#LI	pop	
2	100	295,270	27,124	10	172,420	100	23,974	3,338	100	25	100	73,310	3,142	100	25	
3	100	395,887	45,369	10	204,837	100	47,127	9,057	100	50	100	96,940	9,189	100	25	
4	100	500,067	57,076	10	222,771	100	71,867	20,414	100	50	100	119,052	25,615	100	50	
5	100	602,737	60,968	10	246,794	100	118,641	71,098	100	100	100	170,849	47,593	100	100	
6	100	703,955	80,803	10	266,844	100	149,529	30,025	100	150	100	200,175	30,947	100	150	
7	94	780,851	64,968	10	287,307	100	195,037	35,801	100	250	100	243,288	26,217	100	250	
8	81	876,451	69,441	10	326,890	100	226,843	28,821	100	300	100	279,133	44,294	100	250	

表 3 Type II 関数の最適化結果

Table 3 Result of optimization of type II function.

		リンケージ情報なし			PICI (LIMS)		LINC-R					LIDI-R				
		100 試行			10 試行		100 試行					100 試行				
T	#Opt	MNE	STD	#Opt	MNE	#Opt	MNE	STD	#LI	pop	#Opt	MNE	STD	#LI	pop	
2	100	500,688	83,123	10	205,808	100	28,119	3,397	100	25	100	77,745	3,605	100	25	
3	95	753,311	105,269	10	252,105	100	33,415	5,300	100	25	100	82,672	4,627	100	25	
4	37	912,222	73,052	10	285,725	100	38,095	5,723	100	25	100	87,363	5,319	100	25	
5	1	996,285	-			100	44,171	6,144	100	25	100	93,258	6,281	100	25	
6	0	-	-			100	49,136	6,300	100	25	100	98,819	7,322	100	25	
7	0	-	-			100	55,227	10,088	100	25	100	105,063	9,976	100	25	
8	0	-	-			100	64,930	12,570	100	25	100	111,590	10,014	100	25	

表 4 Type III 関数の最適化結果

Table 4 Result of optimization of type III function.

		リンケージ情報なし			LINC-R					LIDI-R				
		100 試行			100 試行					100 試行				
T	#Opt	MNE	STD	#Opt	MNE	STD	#LI	pop	#Opt	MNE	STD	#LI	pop	
2	100	166,616	12,888	100	10,844	1,670	100	50	100	59,611	3,146	100	25	
3	100	511,416	26,750	99	75,918	63,692	100	250	100	65,512	4,258	100	25	
4	12	975,700	21,377	36	102,274	49,290	100	300	100	71,832	4,210	100	25	
5	0	-	-	5	102,772	51,227	100	250	100	78,829	5,447	100	25	
6	0	-	-	1	76,036	0	100	250	100	86,931	5,340	100	25	
7	0	-	-	0	-	-	100	-	100	96,684	7,548	100	25	
8	0	-	-	0	-	-	100	-	100	105,608	11,733	100	25	

のと同じ関数である。これらは $L = 20$ に固定し、 $T = 2, \dots, 8$ として最適化を行った。Type III 関数では、 $T = 2, \dots, 8$ として最適化を行った。実験結果を表 2、表 3、表 4 に示す。それぞれ 100 試行ずつ最適化を実行し、最適解に到達した試行の数を #Opt に示した。MNE (Mean Number of function Evaluations) と STD (Standard Deviation) は最適解に到達した試行での、最適解を得るまでの目的関数の評価回数、すなわち探索点の数の平均値および標準偏差である。評価回数にはリンケージ同定に要した評価回数も含んでいる。#LI は、LINC-R では加法分解性の意味で、LIDI-R では差分符号の独立性の意味で正確にリンケージ同定できた試行数である。pop は、LINC-R

と LIDI-R の最適化ステージで最良の結果を得た集団サイズである。

比較のために表 2、表 3、表 4 にはリンケージ情報を使わない GA の結果を、表 2、表 3 には PICI による結果も記載してある。PICI の値は Tsutsui¹²⁾ らの報告の値であり、全 10 試行のものとなっている。同報告では Type II 関数については $T \leq 4$ の結果のみが報告されている。なお 2 章で述べたように PICI には LISS と LIMS の 2 つの適用方法があるが、ここでは LIMS の値を掲載した。

表 2 と表 3 にみられるように LINC-R と LIDI-R を用いた GA ではすべてのリンケージを正確に同定し、全試行で最適解を得ている。また、従来手法であ

る PICI よりも少ない探索点の数 (MNE) で最適解を得ている。特に LINC-R の MNE は Type I 関数の $T = 2$ の場合で PICI の 14% 程度, $T = 8$ で 70% 程度となっている。これは, LINC-R のパラメータである個体数 m を理論的に求められること, また, 同定性能が高いため, この目的関数の場合, 1 個体でリンケージ同定できることが理由である。

LINC-R と LIDI-R を用いた GA では Type I 関数の T が大きくなると MNE が大きくなっている。Type I 関数では, 最大のビルディングブロック長が T で, 最短のビルディングブロック長は 1 である。 T が大きい場合について, 最適化の過程の個体の分布を観察したところ, 集団サイズが大きい場合には, 短いビルディングブロックが集団中に広まるのに時間がかかり, 最適解への到達が遅くなっていた。逆に集団サイズが小さい場合には, 長いビルディングブロックが十分探索される前に早熟収束し, 最適解を得られなくなっていた。また, Type I 関数で T が大きくなるにつれ, 最適化ステージに必要な集団サイズが大きくなっていくのも, ビルディングブロックが長いほど, 探索に大きな集団が必要となるからと考えられる。目的関数のビルディングブロック長の差が大きくなると最適化効率が低下すると思われる。

LIDI-R は目的関数の差分の符号独立性で, LINC-R は加法分解性でリンケージ同定を行う。したがって Type III 問題の F_{Sp2} のように, 差分の符号独立性はあるが加法的に分解できないような場合, 本来は個別に最適化可能な遺伝子座であるにもかかわらず, LINC-R ではリンケージありと判断されてしまう。表 4 が示すように, このような問題では LIDI-R を用いてリンケージ同定を行ってから最適化をするほうが, 最適化能力が高くなり, また少ない点の探索で最適解を得られる。

6. おわりに

本論文では, 実数値 GA におけるリンケージを, 遺伝子座間の独立最適化可能の意味でのリンケージとして明確に定義した。さらに, この定義をもとに直接的にリンケージ同定を行う LINC-R と LIDI-R を提案した。LINC-R は加法分解性をもとに, LIDI-R は目的関数の差分の符号独立性をもとにリンケージ同定を行う。

この 2 つの提案手法は従来手法である PICI と比べ, 試行錯誤的に設定する必要があるパラメータが少ないという長所を持つ。また, LINC-R についてはパラメータの 1 つであるリンケージ同定に必要な個体数

を, 加法的に分解可能な領域の割合をもとに求めることができる。

LIDI-R と比較すると LINC-R のほうが少ない点の探索でリンケージ同定が可能だが, 符号の差分独立性の意味でリンケージがない場合にも, リンケージがあると誤判定してしまう。そのような問題の場合には LIDI-R を用いるほうが良い。

数値実験により LINC-R と LIDI-R が高いリンケージ同定能力を持つことを示した。また従来手法よりも少ない点の探索で最適解を得られることを確認した。

NFL 定理¹⁵⁾ によって, 解探索アルゴリズムはどのようなアルゴリズムであっても, すべての問題に対する期待性能は等しいことが示されている。この定理は, すべての問題を効率的に解けるアルゴリズムが存在しないことを示すとともに, ある種類の問題群には高性能となるようにアルゴリズムを設計することが可能であることも示唆している。実用上は, 現実世界に広く存在するような問題群を効率的に解けることが重要である。本論文で報告した LINC-R は目的関数が加法的に分解できるような問題を, LIDI-R は目的関数の差分の符号独立性を持つような問題を効率的に解ける方法である。リンケージグループごとに探索することで, 探索空間を大幅に小さくすることが可能である。実問題にはこのように複数の部分問題に分解できる場合が多くあると考える。

参 考 文 献

- 1) Harik, G.R. and Goldberg, D.E.: Learning Linkage, *Foundations of Genetic Algorithms IV*, Belew, R.K. and Vose, M.D. (eds.), pp.247-262, Morgan Kaufmann, San Francisco, CA (1996).
- 2) Pelikan, M., Goldberg, D.E. and Cantú-Paz, E.: BOA: The Bayesian Optimization Algorithm, *Proc. 1999 Genetic and Evolutionary Computation Conference*, pp.525-532 (1999).
- 3) Munetomo, M. and Goldberg, D.E.: Designing a genetic algorithm using the linkage identification by nonlinearity check, IlliGAL Technical Report (1998).
- 4) Munetomo, M. and Goldberg, D.E.: Linkage Identification by Non-monotonicity Detection for Overlapping functions, *Evolutionary Computation*, Vol.7, No.4, pp.377-398 (1999).
- 5) 榎朝雅晴: エピスタシス尺度に基づくリンケージ同定手法の提案, 情報処理学会論文誌: 数理モデル化と応用, Vol.43, No.SIG10 (TOM7), pp.6-13 (2002).
- 6) Fogel, D.B.: Real-valued vectors, *Handbook*

of *Evolutionary Computation*, Bäck, T., Fogel, D.B. and Michalewicz, Z. (eds.), pp.C1.3:1-1, Institute of Physics Publishing and Oxford University Press (1997).

- 7) Wright, A.H.: Genetic Algorithms for Real Parameter Optimization, *Foundations of Genetic Algorithms*, Rawlins, G.J.E. (ed.), Morgan Kaufman (1991).
- 8) Michalewicz, Z. and Janikow, C.Z.: Handling Constraints in Genetic Algorithms, *Proc. 4th Intl. Conference on Genetic Algorithms*, pp.151-157 (1991).
- 9) Eshelman, L.J. and Schaffer, J.D.: Real-Coded Genetic Algorithms and Interval-Schemata, *Foundations of Genetic Algorithms 2*, Whitley, L.D. (ed.), Morgan Kaufman (1993).
- 10) Tsutsui, S., Goldberg, D.E. and Sastry, K.: Progress Toward Linkage Learning in Real-Coded GAs with Simplex Crossover, *IlligAL Report No.2000033* (2000).
- 11) Tsutsui, S., Goldberg, D.E. and Sastry, K.: Linkage Learning in Real-Coded GAs with Simplex Crossover, *Proc. 5th Intl. Conf. on Artificial Evolution*, pp.73-84 (2001).
- 12) Tsutsui, S. and Goldberg, D.E.: Simplex Crossover and Linkage Identification: Single-Stage Evolution VS. Multi-Stage Evolution, *Proc. 2002 Congress on Evolutionary Computation*, pp.974-979 (2002).
- 13) 樋口隆英, 筒井茂義, 山村雅幸: 実数値 GA におけるシンプレクス交叉の提案, *人工知能学会論文誌*, Vol.16, No.1, pp.147-155 (2001).
- 14) 佐藤 浩, 小野 功, 小林重信: 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, *人工知能学会誌*, Vol.12, No.5, pp.734-744 (1997).
- 15) Wolpert, D.H. and Macready, W.G.: No Free Lunch Theorems for Optimization, *IEEE Trans. Evolutionary Computation*, Vol.1, No.1, pp.67-82 (1997).

付 録

A.1 差分の符号独立性とリンケージ

遺伝子座 i の任意の 2 点間の目的関数の大小関係が, 遺伝子座 j の値によらない場合, x_i は x_j によらず f を最大化できること, すなわち

$$\begin{aligned} \forall x_i^1, x_i^2 \in X_i, \forall x_j^1, x_j^2 \in X_j, \\ \text{sgn}(f(x_i^1, x_j^1) - f(x_i^2, x_j^1)) \\ = \text{sgn}(f(x_i^1, x_j^2) - f(x_i^2, x_j^2)) \end{aligned} \quad (29)$$

が成立する場合に, 式 (2) で定義される $X_i^*(x_j)$ について,

$$\forall x_j^1, x_j^2 \in X_j, \quad X_i^*(x_j^1) = X_i^*(x_j^2) \quad (30)$$

となることを示す.

$\forall x_j^1, x_j^2 \in X_j$ について, $X_i^*(x_j^1) \neq X_i^*(x_j^2)$ と仮定する. $X_i^*(x_j^1) \neq X_i^*(x_j^2)$ となるのは, 以下の 3 つの場合である.

$$X_i^*(x_j^1) \subset X_i^*(x_j^2), \quad (31)$$

$$X_i^*(x_j^2) \subset X_i^*(x_j^1), \quad (32)$$

$$\begin{aligned} [X_i^*(x_j^1) \setminus X_i^*(x_j^2) \neq \phi] \\ \text{かつ } [X_i^*(x_j^2) \setminus X_i^*(x_j^1) \neq \phi] \end{aligned} \quad (33)$$

$X_i^*(x_j^1) \subset X_i^*(x_j^2)$ の場合,

$$\begin{aligned} \forall x_i^1 \in X_i^*(x_j^1) \\ \forall x_i^2 \in X_i^*(x_j^2) \setminus X_i^*(x_j^1) \end{aligned} \quad (34)$$

について, 式 (3), (5) より

$$\begin{aligned} \text{sgn}(f(x_i^1, x_j^1) - f(x_i^2, x_j^1)) = 1 \\ \text{sgn}(f(x_i^1, x_j^2) - f(x_i^2, x_j^2)) = 0 \end{aligned} \quad (35)$$

$X_i^*(x_j^2) \subset X_i^*(x_j^1)$ の場合,

$$\begin{aligned} \forall x_i^1 \in X_i^*(x_j^1) \setminus X_i^*(x_j^2) \\ \forall x_i^2 \in X_i^*(x_j^2) \end{aligned} \quad (36)$$

について, 式 (3), (5) より

$$\begin{aligned} \text{sgn}(f(x_i^1, x_j^1) - f(x_i^2, x_j^1)) = 0 \\ \text{sgn}(f(x_i^1, x_j^2) - f(x_i^2, x_j^2)) = -1 \end{aligned} \quad (37)$$

$[X_i^*(x_j^1) \setminus X_i^*(x_j^2) \neq \phi]$ かつ $[X_i^*(x_j^2) \setminus X_i^*(x_j^1) \neq \phi]$ の場合,

$$\begin{aligned} \forall x_i^1 \in X_i^*(x_j^1) \setminus X_i^*(x_j^2) \\ \forall x_i^2 \in X_i^*(x_j^2) \setminus X_i^*(x_j^1) \end{aligned} \quad (38)$$

について, 式 (5) より

$$\begin{aligned} \text{sgn}(f(x_i^1, x_j^1) - f(x_i^2, x_j^1)) = 1 \\ \text{sgn}(f(x_i^1, x_j^2) - f(x_i^2, x_j^2)) = -1 \end{aligned} \quad (39)$$

式 (35), (37), (39) より, $\forall x_j^1, x_j^2 \in X_j, X_i^*(x_j^1) \neq X_i^*(x_j^2)$ とすると式 (29) が成立しない.

したがって, 式 (29) が成立する場合は, $\forall x_j^1, x_j^2 \in X_j, X_i^*(x_j^1) = X_i^*(x_j^2)$ となる.

(平成 17 年 1 月 31 日受付)

(平成 17 年 3 月 30 日再受付)

(平成 17 年 6 月 6 日再々受付)

(平成 17 年 7 月 27 日再々々受付)

(平成 17 年 9 月 19 日採録)



手塚 大 (正会員)

1992年大阪大学基礎工学部生物工学科卒業。1994年同大学大学院基礎工学研究科物理系専攻博士前期課程修了。同年日立東北ソフトウェア(株)(現(株)日立東日本ソリューションズ)入社。2006年北海道大学大学院工学研究科システム情報工学専攻博士後期課程修了。博士(システム情報工学)。アプリケーションソフトウェアの研究、開発に従事。



棟朝 雅晴 (正会員)

1968年生。1996年北海道大学大学院工学研究科情報工学専攻博士後期課程修了。同年北海道大学大学院工学研究科システム情報工学専攻助手。1998~1999年イリノイ大学基礎工学科遺伝的アルゴリズム研究室客員研究員。1999年北海道大学情報メディア教育研究総合センター助教授。2003年同大学情報基盤センター助教授。博士(工学)。遺伝的アルゴリズム, ネットワークシステム, 分散処理システムに関する研究に従事。IEEE 会員。



赤間 清 (正会員)

1973年東京工業大学工学部制御工学科卒業。1975年同大学大学院修士課程修了。1979年同大学院博士課程単位修得退学。同年同大学助手。1981年北海道大学文学部講師。1989年同大学工学部助教授。1999年より同大学情報メディア教育研究総合センター教授。2003年同大学情報基盤センター教授。人工知能, 知識処理, 等価変換に基づく問題解決の研究に従事。工学博士。人工知能学会会員。