

ランダムなサイズの直交計画近傍を用いた大局的最適解探索法

田 中 秀 俊[†]

直交計画に基づいて近傍点集合を作成し、その近傍点集合から各変数の改善方向を推定してその方向に探索を行う、反復改善型のもっとも適解探索法 Orthogonal Design Local Search (ODLS) に、大局的最適化法の代表であるシミュレーテッドアニーリングの特徴を導入して、大局的に最適解を探索できるようにした。近傍点集合までの距離をランダムにとること、その距離を基準にして勾配方向への二分探索を行うこと、二分探索時には関数値を乱数化して解の悪化を許容することの3点について、それぞれの効果を数値実験を通じて検証した。その結果、ランダムな距離と二分探索とを導入した場合の有効性を示すことができた。シミュレーテッドアニーリングと比較したところ、変数の数が多く、変数間の相互作用が小さい問題について、同一評価回数という条件下で、平均的に良い近似解が得られることが分かった。

Global Optimization Using Orthogonal Design Neighborhood of Random Size

HIDETOSHI TANAKA[†]

Orthogonal Design Local Search (ODLS) solves optimization problems by iteratively estimating improving direction per variable from neighborhood points which are selected according to the orthogonal design of experiment. Features of Simulated Annealing (SA) are introduced to ODLS to solve multi-optima problems: the set of neighborhood points at random distances, binary search, and artificial random perturbation of the objective function. Numerical experiments clarified that the random distances plus the binary search are so effective that ODLS outperforms SA in average when the functions have the large number of variables.

1. はじめに

1.1 問題設定と従来手法

数値を受けつける入力を多数と、数値を返す出力を1つ持つ「箱」を考える。この箱は同じ入力値組を与えても同じ出力値を返すとは限らないが、同じ入力値組で得られる出力値は正規分布に従うとする。この箱から一番小さい値が返るような、入力値の組を捜したい。これが本研究で対象としている問題である。ゲノム情報の分野でいまだに大きな問題として残っているものの1つ、タンパク質立体構造予測問題には、いくつかのアプローチがあるが¹⁾、この問題と見なすこともできる。タンパク質のアミノ酸鎖を構成する原子の位置、あるいは原子間結合距離・結合角・二面角といった要素を入力変数とし、分子全体のポテンシャルエネルギー、あるいは既知分子の構造の統計に照らして得られる尤度を出力変数とすると、エネルギー最小化、あるいは尤度最大化の問題としてとらえることが可能

である。これらエネルギーも尤度も、入力変数で記述しきれていない要素、たとえば水分子の存在、あるいは既知構造の増加などによって、同じ入力変数を与えても同じ出力が返るとは限らない。これをおおよそ正規分布で値が返ると仮定すると、上記の問題に帰着される²⁾。

このような問題に対しては、適当な現在点をまず決めて出力変数値を観察し、現在点の改善を反復することによって最適な点を探索するという方法がとられる。その改善法には、以下の2つの方向性がある。

1つは、現在点の周囲の点からランダムに選択した集合(近傍点集合)を作って各点における出力値を調査し、良い点を次の現在点とするという方法である。近傍点集合の少々悪い点でも、ある程度の確率で「良い点」として選ぶシミュレーテッドアニーリング(Simulated Annealing, SA)³⁾、現在点を集合として持ち、良い点を現在点集合と近傍点集合から作る遺伝的アルゴリズム(Genetic Algorithm, GA)⁴⁾などは、こちらの方法に属するといえる。良い点が見つかりやすいかどうかは、現在点の周囲における出力値の形状(ランドスケープ)に依存する。近傍点集合は良い点

[†] 三菱電機株式会社情報技術総合研究所
Information Technology R&D Center, Mitsubishi Electric Corporation

が得られるまで増えることになるので、集合に属する点の数もこのランドスケープに依存する。

もう1つは、システマティックに選択した近傍点集合を作って各点での出力値を調査し、良い方向を算出してその方向に集中的に探索を進め、発見された最良点を次の現在点とするという方法である。こちらの方法では普通、近傍点集合は現在点付近のランドスケープとは独立に決定される。よって点の数も固定的となる。現在点の導関数値を変数ごとに中心差分法(1変数の値だけ所定幅増減させた近傍点の評価関数値から、現在点の導関数値を見積もる方法)によって見積もる最急降下法(Steepest Descend, SD)⁵⁾や、現在点を集合として持ち、現在点集合から近傍点集合を作成する方法が決まっている多方向探索法(Multi-Directional Search, MDS)⁶⁾、近傍点集合を直交計画に従って選択し、近傍のランドスケープを二次関数などで近似する応答曲面法(Response Surface Method, RSM)⁷⁾を多段階的に用いた場合、あるいは統計的設計支援システム(Statistical Design Support System, SDSS)⁸⁾における多段階法などがこれに相当する。

1.2 直交計画を用いる最適化諸手法

直交計画局所探索法(Orthogonal Design Local Search, ODLS)⁹⁾は、システマティックに近傍を選択する方法に属し、タンパク質立体構造予測のような数千を超える変数の問題を少しでも効率良く探索するための手法として提案された。

ランダムに近傍を探索する方法では、変数の数が増えると、変数全部が揃って最適点の方向へ改善される確率は、急激に下がっていく。ここで悪化した何個かの変数がすべて元に改善されるまでには、相応の余分な反復が費やされる。変数が多いほど、全変数について悪くない方向を選ぶだけで、探索範囲を大幅に小さくできる。

ODLSは、2水準直交表によって近傍点集合を選び、各変数で現在点より+した方がよいか、-した方がよいかだけを、近傍点の評価値の平均を比較して判断し、総合して改善方向を得るものとした。この2水準直交表の性質から、ODLSは変数間が独立な場合に良い性能を発揮する。また、直交表の各因子への変数の割当てをランダム化し、改善方向として選んだ点は、評価値が改善されない場合には次の現在点としない。すなわち、ランダムに近傍点集合を選ぶ場合と同様、近傍点集合の点数は増加する。したがって、交互作用がある場合にも、ランダムに近傍点集合を選ぶ諸手法と同程度には、改善方向を見つける能力があると考えられる。

ODLSは、直交計画を用いる諸手法、田口メソッド¹⁰⁾、応答曲面法、SDSSの多段階適用法の一類型と見なせる。ただし、設計の問題でなくタンパク質立体構造予測という自然界の問題を念頭においているため、いくつかの前提に差異がある。

まず、問題の性質上、多段階適用する場合の反復回数が本来の想定よりかなり多くなる点があげられる。タンパク質立体構造予測問題の変数としては、二面角が主要な役割を担う。二面角は0~360度をとる実数変数で、精度を5度と粗くにとって離散化しても72種類になり、値の種類が多い。多段階適用法では通常、変数の変域の絞りこみや移動を数回程度に抑えるが、この離散化を施した場合でも5度刻みで最適点を探しまわることになり、相当に反復回数が多くなる。

また、本来変数の数を少なく抑える田口メソッドはもとより、大きな直交表を駆使して多くの変数を扱うSDSSと比べても、変数の数は多くなる。変数が数千ともなれば、応答曲面法に必要な直交表が巨大になり、交互作用を考慮して直交表へ変数を割り当てる作業の負荷が大きくなる。

そこで、応答曲面法の特徴である関数の二次曲面近似による最適点見積りや平面近似による最急降下方向見積りは行わず、各変数の+方向と-方向の優劣だけを判定し総合して改善方向としている。この改善方向は、変数間の交互作用などが原因で誤った方向になるおそれもあるので、改善方向には改めて探索を行い、改善を確認している。

1.3 大局的最適解の探索

ODLSは、近傍点の評価値の平均という操作を通じて相殺できる程度の細かい局所的最適解であれば、それらにとらわれず改善方向を見積もることができる。しかし、基本的には局所的最適解にとらわれてしまう手法だった。これは現在点と近傍点との距離を固定して作った近傍点集合によって改善方向を算出し、その改善方向へ単純な線形探索をしていたためである。本稿ではこの対策について、ランダムな近傍を用いる方法の一部を採用することについて議論する。

システマティックな近傍を用いて、大局的最適解への到達を可能とするために採用できる工夫が1つある。局所的最適解は距離的に比較的短い周期での評価値の増減によって発生するものと考えて、適切な遠方を観測することによって、大局的な最適解を目指す方法である。適切な距離は局所的なランドスケープに依存するので、距離を適応的に、あるいはランダムに決めることで対処する。たとえばSAでは本来からランダムに決めるアプローチを採用しており、近傍点と現在点

の距離分布をコーシー分布とするなどによって、適切な遠方を観測する手段を用意できている．また適応的に決める研究もなされている¹¹⁾．この類の工夫は、ランダムな近傍を用いる諸手法には比較的自然的に組み入れられている概念だが、システムティックな近傍を用いる諸手法において効果が確認された例は、筆者の知る限り少ないようである．たとえば、MDS は適応的に決めるアプローチを採用している．しかし、ベンチマーク関数およびタンパク質折り畳み問題での実験によれば、SA は局所的最適解を回避できるが、MDS は局所的最適解にとらわれてしまうという結果が得られた¹²⁾．

本稿では、SA の特徴を ODLS に組み入れ、局所的最適解の回避機能を持たせた点について報告する．導入した機能は 3 つある．1 つは、近傍点集合までの距離をランダムに決めるアプローチの採用で、適当な幅までの一様乱数を用い、適切な遠方を観測できるようにした．また、その距離を基準として、得られた方向への二分探索を実施することにより、不適切に小さい距離を選んでしまった場合に、すぐに別の距離の近傍点集合を選択し直せるようにした．さらに、二分探索中の評価関数値から適当な幅の一様乱数を差し引くことにより、解の悪化を許容するようにした．

本稿の構成は以下のとおりである．2 章で基本 ODLS、線形探索 ODLS ならびにこの改良版、ランダムステップ ODLS、そして本稿で比較対象としている SA について示し、3 章で SA とランダムステップ ODLS との数値実験比較結果を示し、4 章で導入効果と今後の課題について検討する．

2. ODLS と SA

ODLS は、中心差分法によって一次導関数値を見積もる最急降下法をベースとして、局所的最適解への到達速度を保ちながら、大局的最適解への到達可能性を高める方法として提案された．基本 ODLS では、中心差分法の代わりに直交計画を、一次導関数値の代わりに $\pm w$ (w は定数) と 0 の 3 値からなる大まかな改善方向を用いる．線形探索 ODLS では、その改善方向への線形探索を行って現在点を更新する．今回提案するランダムステップ ODLS では、近傍点集合までの距離をランダムにとり、得られた改善方向へ二分探索を行い、探索中の評価関数値からランダムな値を差し引く．本章では最急降下法、中心差分法、直交計画法について簡単に述べ、基本 ODLS、線形探索 ODLS、今回の改良版であるランダムステップ ODLS について順に述べる．さらに、次章で比較実験に用いた SA

についても簡単に述べる．

2.1 準備

最急降下法は、最適化問題で一次導関数値が得られる場合の一手法である．適当な初期点を現在点として探索を開始し、現在点の導関数値の示す方向に線形に探索を進めて、解の改善ができなくなった点を次の現在点とする．一次導関数値は、導関数形が得られていない場合、近傍点集合から中心差分法や前進差分法で見積もることになる．どちらも各変数で独立に導関数値を推定し、それを単純に集めて勾配方向とする．中心差分法は、近傍点を現在点の各変数のどれか 1 つに所与の値を加えた点および差し引いた点とし、それらの関数値から、その変数の一次導関数値を求める．変数の数を n とすると、中心差分法の近傍点数は $2n$ である．前進差分法は、近傍点を現在点の各変数のどれか 1 つに所与の値を加えた点とし、それらの関数値と現在点の関数値とから、その変数の一次導関数値を求める．前進差分法の近傍点数は n になる．

2.2 直交計画

組合せ最適化の用語を使って簡単に直交計画について述べる．2 つの値、ここでは -1 か $+1$ のいずれかをとる 2 値変数 n 個からなる最適化問題と、その空間内の点の集合を考える．適当な 2 変数に着目したとき、4 種類の値組合せ ($--/-+/-++/++$) がそれぞれ同じ回数出現する場合、集合のその変数間には「直交している」という．そして表 1 のように、どの 2 変数を見ても直交している集合を「直交計画」と呼ぶ．ある大きさの直交計画を作るには、その大きさのガロア体を作ればよい．表 1 は 8 点からなる 7 変数の直交計画で、どの 2 変数も $--/-+/-++/++$ が 2 回ずつ出現している．各点について評価値を求め、各変数について $-$ となっている点と $+$ となっている点の平均を比較すると、傾向として $-$ と $+$ のどちらにした方がよいかの判断ができる．たとえば、変数 2 は点 1, 2, 7, 8 が $-$ 、残りが $+$ なので、点 1, 2, 7, 8 の評価値平均 25.435 と、点 3~6 の評価値平均 2.715

表 1 直交計画
Table 1 Orthogonal design of experiment.

	変数 1	変数 2	変数 3	変数 4	変数 5	変数 6	変数 7	評価値
点 1	-	-	-	-	-	-	-	49.00
点 2	-	-	-	+	+	+	+	30.25
点 3	-	+	+	+	+	-	-	3.61
点 4	-	+	+	-	-	+	+	1.00
点 5	+	+	-	-	+	+	-	2.25
点 6	+	+	-	+	-	-	+	4.00
点 7	+	-	+	+	-	+	-	10.24
点 8	+	-	+	-	+	-	+	12.25

を比較する．明らかに後者が小さいので，他の変数の値にかかわらず，変数 2 は + にした方が小さい評価値が得られるだろう，ということが分かる．また，変数 7 のように + と - の評価値平均があまり差がない場合には，判断を保留することになる．

最急降下法で，一次導関数値の代わりに，現在点から $\pm w$ (w は定数) の値および現在点と等しい値 (0) の 3 値に単純化した改善方向を用いることを考える．この改善方向は，判断保留の変数について 0 を割り当てることにすると，上記の直交計画を近傍点集合として用いることにより，求めることができる．この場合の近傍点集合の点数 m は，変数の数を n とすると，「 n より大きい 2 の整数乗」で， $n < m \leq 2n$ となる．

2.3 基本 ODLS

基本 ODLS では，以下の (1) ~ (4) の手順を繰り返す．

(1) 近傍点を直交計画に従って選択する．

直交計画に従った近傍点設計法を以下に述べる．考え方の概略を列挙示す．

- 変数の数に応じた因子数の 2 水準直交表を作成．
- 直交表の因子からランダムに選んで近傍点の変数へ割り当てる．
- 直交表の各実験を近傍点に対応させる．
- 直交表の各因子の 2 水準を，各変数の + 側近傍と - 側近傍とに対応させる．
- 現在点から所定の距離に近傍点を置く．

以上を簡潔に示したのが式 (1) である．

近傍点の数を m ，変数の数を n とする．第 i 近傍点を x_i とし， x_i の第 j 変数の値を X_{ij} とする．現在点を x^{current} ， x^{current} の第 j 変数の値を x_j^{current} とする．現在点と近傍点の距離を決める正の数を w とする． i を q 桁のグレイコードで表したうちの s 桁目を g_s ， $1 \sim m-1$ からランダムに n 個選び， $0 \sim n-1$ とランダムに対応させる関数を $C(\cdot)$ とする． $k = C(j)$ を q 桁の 2 進数で表したうちの t 桁目を b_t ，2 の剰余を mod_2 とする．近傍点の位置，すなわち近傍点 x_i の第 j 変数の値 X_{ij} は式 (1) で表される．

$$X_{ij} = x_j^{\text{current}} + 2w[L_{ij} - 0.5]; \quad (1)$$

$$L_{ij} = \text{mod}_2 \sum_{s=0}^{q-1} g_s b_{q-s-1};$$

$$m = 2^q; 2^{q-1} \leq n < 2^q; q \in \text{整数}$$

(2) 各近傍点 x_i について関数値 $f(x_i)$ を求める．

(3) 各変数の改善方向を求める．

判断マージンとして正の数 B を用意すると，最小化問題の場合，現在点における変数 j の改善方向 e_j

は式 (2) で表される．

$$e_j = \begin{cases} -1 & (\mu_j^+ + B < \mu_j^-) \\ +1 & (\mu_j^+ + B < \mu_j^-) \\ 0 & (\text{otherwise}) \end{cases} \quad (2)$$

$$\mu_j^+ = \frac{2}{m} \sum_{i=0; L_{ij}=1}^{m-1} f(x_i)$$

$$\mu_j^- = \frac{2}{m} \sum_{i=0; L_{ij}=0}^{m-1} f(x_i)$$

(4) 現在点を更新する．

現在点と次の現在点との距離を決める正の数 C を用意し，現在点 x^{current} を方向 e へ幅 C だけ改善し，次の現在点 x^{new} とする (式 (3))．

$$x_j^{\text{new}} = x_j^{\text{current}} + C e_j \quad (3)$$

ODLS の近傍点集合は，たとえば 1,000 変数の場合 1,024 点となる．この多量の評価回数に加え，512 点ずつの + と -，計 2,000 通りの平均を算出して 1,000 組の比較をするという大きなオーバーヘッドを，ODLS はかかえている．しかし，直交計画をランダムに削減して近傍点数を減らす実験⁹⁾により，直交計画の近傍点数が最適であり，これ以上少ない場合には改善方向が悪化することが示されている．直交計画によらず，ランダムな近傍点集合を用いて改善方向を見積もる方法も考えられるが，この実験結果から，近傍点数はの場合でも直交計画と同じ数程度必要になると思われる．

2.4 線形探索 ODLS

基本 ODLS は総合的には SA に劣っていた．これは 1 回の改善幅に対する前述のオーバーヘッドが大きく，収束速度が遅いという欠点のためと考えられた．その高速化を試みたのが線形探索 ODLS²⁾である．上述の式 (1) における現在点と近傍点の距離係数 w と，式 (2) における現在点と次の現在点との距離係数 C について，基本 ODLS では各反復で $C=w$ としていたが， $C=k w$ ，ただし k は正の整数として， $k=1$ から線形探索して最適な C を求めるようにした．

2.5 ランダムステップ ODLS

線形探索 ODLS には局所的最適解からの脱出機能がなかった．そこで局所解脱出のために，3 つの機能を追加した．

(1) 近傍点と現在点の距離，すなわち式 (1) の w を $1 \sim W$ の整数一様乱数とする． W は事前に与える．

(2) 現在点からの改善方向への探索を，二分探索と

する．すなわち， $f(x^{\text{current}} + Ce_j)$ について，式 (3) の C の値を $C = 0 \sim 2w$ ，ただし C が整数の範囲で，二分探索により求める（二分探索の幅が 1 になるまで実施する）．

- (3) 現在点からの改善方向への探索において，評価値の悪化を一定確率で許容する仕組みを加える．すなわち， $f(x^{\text{current}} + Ce_j)$ の代わりに，正の一樣乱数 D を用意し， $f(x^{\text{current}} + Ce_j) - D$ について，式 (3) の C の値を $C = 0 \sim 2w$ ，ただし C が整数の範囲で二分探索により求める．

本稿ではこの 3 機能の有効性の有無について，数値実験を通じて示す．

機能 (1) は，SA のランダムなステップ距離を模して導入した．式 (1) の w は，現在点での改善方向見積りのために，どれだけ遠方を観測して判断するかを示すものである．この大小は，ランドスケープをどこまで細かく見るかに対応する． w が小さい場合には小さい幅の増減を観測でき， w が大きい場合には大きい幅の増減しか観測しない．現在点周辺が w よりも十分大きい領域で局所的な解を持つ場合を考えると，現在点から w の距離の近傍点を使って改善方向を見積もったとき，その局所解以外が考慮されることはないと考えられる．その局所解以外の大域的解を考慮に入れるには，十分に大きい w をとることが不可欠である．しかし，そのような w は問題のランドスケープに依存し，事前の決定は困難である．そこで，SA にならない， w をランダムに定めることとした．

SA では，ランダムなステップ距離だけ改善した点が，直接に次の現在点の候補となるが，ODLS では単に改善方向の見積りに使い，次の現在点は，改めて改善方向への探索の結果決定される．したがって， w の大小にかかわらず，次の現在点との距離は探索の結果得られる式 (3) の C で決まる．

機能 (2) は改善方向への二分探索である．線形探索 ODLS のように，機能 (1) で求められた改善方向へ，幅 w で線形探索する方法も試みたが，線形探索で評価回数を何百回も費やすケースがしばしば起きることが分かった．これは w が不適当に小さい場合に起きる．近傍点の評価が並列性のある処理なのに対し，線形探索は逐次的な処理なので，この評価回数を少なくした方が速くなる．そこで，探索範囲を w の倍までとする二分探索を採用した．探索の結果，現在点より良い点が得られなかった場合，すなわち $C = 0$ のときには， w を選び直して近傍点集合を再設定する．

機能 (3) は，SA の解悪化許容の仕組みを模して導入を試みた．SA では解が改善した場合には採用し，悪化した場合には悪化許容確率分布に従って許容する．一樣分布を差し引くことは，許容確率分布の確率密度関数として，原点を通り一樣分布 D の上限を定義域の上限とする線形関数を用いていることに相当する．

次章で示す SA との比較実験では，まずランダムステップ機能と二分探索機能の有効性を確認し，次に解悪化許容機能を含めた場合の効果を測定した．整数一樣乱数 D の適切な上限は，事前には分からないため，実験によって確定を試みた．

2.6 SA

本稿の実験で ODLS の比較対象としたアルゴリズム，実数値を探索空間とする SA について簡単に述べる．SA は現在点から適切なステップ距離をとった近傍点集合を生成し，適切な悪化許容確率に従って，評価関数値が高くなる方向の近傍点でも次の現在点とする．この両者によって局所解からの脱出が可能となる．ステップ距離と悪化許容確率を適切となるよう制御すべく，SA では「温度」と呼ばれるパラメータを用いる．温度が高い場合には，ステップ距離が大きくなり，悪化許容確率が増える．

SA は melting と cooling の 2 段階の処理からなる．melting 過程では，低温から出発して，とりうる近傍点集合全部が悪化許容確率の範囲で選択可能になるまで温度を高めていく．melting により十分高温になると，次に cooling 過程では所与の冷却スケジュールによって温度を徐々に下げていく．ステップ距離にはガウシアン分布を用いる方法やコーシー分布を用いる方法などが提案されており，それぞれについて，最適解に到達する確率が 1 となるような冷却スケジュールが存在する．それより急速な冷却は，確率 1 での最適解到達が保証されないが，実行時間の都合上用いられることが多く，経験的には良い解が得られる¹³⁾．本稿ではこの急速冷却を行うものを実験に利用している．

本稿の実験においては taygeta.com から提供されている SA¹⁴⁾ をベースとして用いた．melting は省略して開始温度 T_0 を事前に定めた．冷却スケジュールについては，開始温度 T_0 ，終了温度 T_E と総反復回数 N から第 k ステップの温度 T_k を式 (4) で求めることとした．悪化許容確率には，温度 T_k とエネルギー差 ΔE ，悪化許容係数 A から求まる確率 $a(\Delta E, T_k)$ (式 (5)) を採用した．ステップ距離 Δx については，温度 T_k における各変数での Δx_i が一次元コーシー分布 g_i (式 (6)) に従うような確率密度分布 $g(\Delta x, k)$ を採用した．各式における開始温度 T_0 ， N 回反復後

表 2 SA の諸パラメータ
Table 2 Parameters for Simulated Annealing.

	T_0	T_E	A
Rastrigin	25	0.05	0.9
Griewank	10	0.1	0.5
Schwefel	1	0.1	0.9

の終了温度 T_E , 悪化許容係数 A は, 各関数で簡単なパラメータサーチを実施して表 2 のように決めた.

$$T_k = \frac{NT_0T_E}{(T_0 - T_E)k + NT_E} \quad (4)$$

$$a(\Delta E, T_k) = (1 + \exp[\frac{\Delta E}{AT_k}])^{-1} \quad (5)$$

$$g_i(\Delta x_i, k) = \frac{T_k}{\Delta x_i^2 + T_k^2} \quad (6)$$

3. 数値実験対象および比較条件

本稿では比較実験の対象問題として, ベンチマーク用として有名な, Rastrigin 関数 $R(x)$, Griewank 関数 $G(x)$, Schwefel 関数 $S(x)$ の最小化問題を用いた. $R(x)$, $G(x)$, $S(x)$ を式 (7) ~ (9) に示す. ただし, 元の $S(x)$ は範囲を限定しないと最小値が決まらないので, $-512 \sim +511$ の範囲を反復するように多少の変更を加えている. $R(x)$, $G(x)$ の最小値は 0, $S(x)$ の最小値は変数の数を n とするとおおよそ $-419n$ である. このベンチマーク問題で, $n=1,000 \sim 2,000$ にのぼる場合の, 各アルゴリズムの比較検証を実施する. これは, 本来の目標としているタンパク質立体構造予測問題において, 実用的なサイズのタンパク質を扱う場合の検証を意図している.

関数の評価に時間がかかる応用を想定し, 各関数の評価回数を 5 万回に限定 ($N=50,000$) して, アルゴリズムの一定時間内での性能を比較した. 式 (1) の w は正の整数 ($1 \sim W$) をとる一様乱数を採用し, その最大幅 W については簡単な実験 (図 1) を行った. 図 1 は, $n=100$ の各関数を $N=50,000$ の範囲で最小化する実験を 50 回ずつ, $W=50, 100, 150, 200, 250$ について実施した結果を示している. 見つかった最小の $G(x)$, $R(x)$, $S(x)$ の平均値 $\bar{G}$, $\bar{R}$, $\bar{S}$ について, 視覚的な確認の目的で変域を合わせ, $100\bar{G}/n$, $\bar{R}/n$, $(\bar{S}/n + 419)/100$ を縦軸にして示した. この結果から, $W \geq 100$ であればよいと思われる. 以後, $\bar{S}$ の良かった $W=200$ を用いる. また, 式 (2) の B は 0 とした.

$$R(x) = 10n + \sum_{i=1}^n ((\frac{x_i}{100})^2 - 10 \cos(\frac{2\pi x_i}{100})) \quad (7)$$

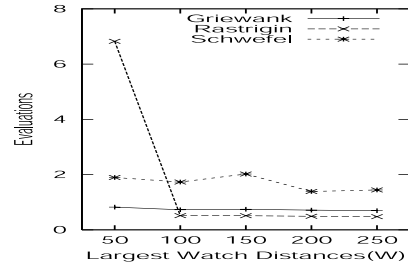


図 1 最大幅 W に関する性能比較
Fig. 1 Optimization of largest watch distance.

$$G(x) = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) \quad (8)$$

$$S(x) = \begin{cases} \sum_{i=1}^n -\xi_i \sin(\sqrt{|\xi_i|}) & (\xi = 512 - \text{mod}_{512} x_i; x_i \geq 512) \\ \sum_{i=1}^n -x_i \sin(\sqrt{|x_i|}) & (-512 \leq x_i < 512) \\ \sum_{i=1}^n -\eta_i \sin(\sqrt{|\eta_i|}) & (\eta_i = -512 + \text{mod}_{512} |x_i|; x_i < -512) \end{cases} \quad (9)$$

4. 実験結果

4.1 ランダムステップの有効性

ステップ距離のランダム化の効果を確認するために, 機能 (1) のランダムステップ化と機能 (2) の二分探索を用いて比較を行った. 図 2, 3, 4 は $R(x)$, $G(x)$, $S(x)$ の最小化問題について, SA および ODLS で得られた最小値の比較である. $R(x)$ および $G(x)$ では真の最小値が 0 なので, 変数の数で割った値によって, 解の変数 1 つあたりの悪さ (近似度) を図 2, 図 3 で示している. $S(x)$ では真の最小値がおおよそ $-419 \times$ 変数の数となるため, 得られた解と最小値との差を変数の数で割った値によって, おなじく解の変数 1 つあたりの悪さ (近似度) を図 4 で示している. 横軸には変数の数 n をとり, 探索開始点は $-512 \sim +511$ の間の一様乱数を成分とする点としている. 各図では ODLS, SA それぞれの各 n について, 探索開始点を変えて 10 回ずつ試行した平均値を線で結び, 最小と最大をエラーバーで示している. $G(x)$ だけは縦軸を対数にとっている.

4.2 二分探索中のノイズの効果

次に, 機能 (1) のランダムステップ化と, 機能 (3) の一様乱数を付加した二分探索とを用いて, ODLS 内で比較実験を行った. 変数の数を $n=10, 100, 1,000, 2,000$ とし, 機能 (3) の一様乱数 D の幅を変数 1 つ

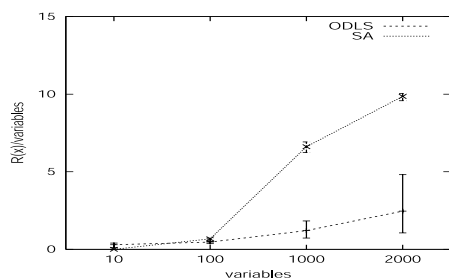


図 2 Rastrigin 関数における性能比較

Fig. 2 Optimization of Rastrigin by SA and ODLS.

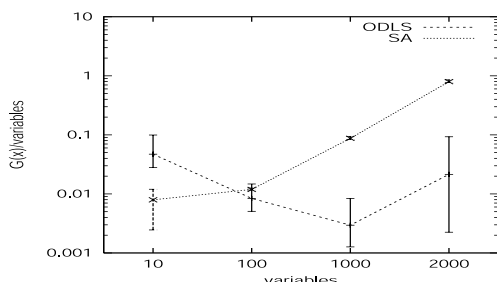


図 3 Griewank 関数における性能比較

Fig. 3 Optimization of Griewank by SA and ODLS.

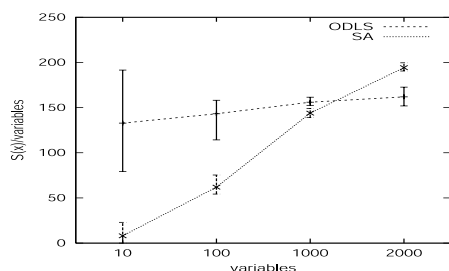


図 4 Schwefel 関数における性能比較

Fig. 4 Optimization of Schwefel by SA and ODLS.

あたり 0.1, 1.0, 10, 100 とし, 各 50 試行して平均性能を比較した。その結果を図 5, 6, 7 に示す。いずれも変数の数が同じものを線で結んだグラフであり, 横軸は乱数幅 D , 縦軸の値は, Rastrigin および Schwefel は図 2, 図 4 と同一, Griewank は対数をとらない元の値である。

5. 考察および課題

図 2 ~ 図 4 によれば, 局所解の多い $R(x)$, $G(x)$ とともに, 5 万回限定の比較で ODLS は SA を平均的に上回っている。 $R(x)$ と $G(x)$ では $n=10$ で SA, $n=100$ で同等, $n=1,000$ および 2,000 で ODLS が良い結果を出している。ODLS に持たせた局所解回避および高速化は, ノイズを付加しない状態で十分な効果が得られていると考えられる。一方, $S(x)$

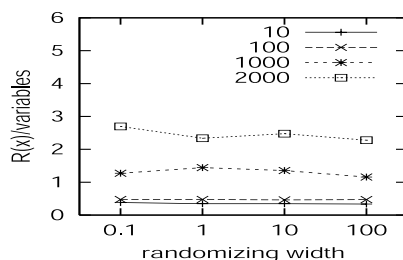


図 5 Rastrigin 関数の一様乱数幅の効果

Fig. 5 Optimization of Rastrigin minus uniform random.

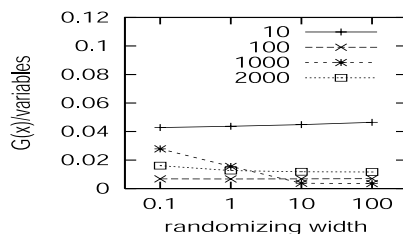


図 6 Griewank 関数の一様乱数幅の効果

Fig. 6 Optimization of Griewank minus uniform random.

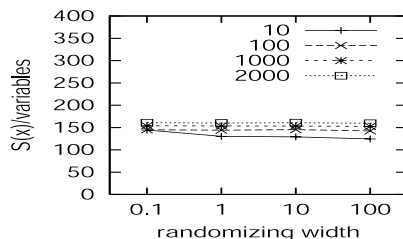


図 7 Schwefel 関数の一様乱数幅の効果

Fig. 7 Optimization of Schwefel minus uniform random.

は $n=100$ でもまだ SA が良く, $n=1,000$ および $n=2,000$ ではほぼ同等となっている。 $S(x)$ で良い結果が得られていない理由は, $S(x)$ が振幅の変化する周期関数で, w を大きくしても最小値方向が局所的状況からは得られにくく, ODLS の性能が発揮できなかったためと考えられる。

また, 悪化を許容するために導入したノイズの効果は, 図 5 ~ 図 7 によれば, $n=1,000$ の場合の $G(x)$ を除いて, ほぼ平坦, つまりほぼ影響なしとの結果が得られている。変数の数が多く乱数の幅が大きいほど平均的に良くなるという傾向がわずかに見られるが, 今後の検証上の課題としたい。

以上の結果から, ランダムステップ ODLS + 二分探索は, 変数の多い場合に有効であり, ランダムステップ ODLS に一様乱数付加の二分探索を施しても大きな効果は得られない, という結論が導ける。

Schwefel 関数のように, 振幅が大きくなるが平均が変わらないようなケースでは, 長い周期を観察しても

正しい改善方向が得られにくいと考えられる．このような問題への対策は，今後の課題である．

本稿の実験で用いたベンチマーク関数は，みなほぼ変数から関数値への影響が独立だった．一方，ODLSでは直交表の中からランダムに列を選んで変数に割り当てている．たとえば現在の方法では，1,000 変数に対して 1,023 列の中から 1,000 選ぶので，交互作用のない問題は ODLS に有利な設定といえる．一方，本来の目標としているタンパク質立体構造予測問題は，変数の数の多さとともに，原子間の相対位置から構造上の評価値が決まるため，変数間の交互作用が密であるという特徴も持つ．今後の課題として，変数間が独立でない問題，たとえばこれらベンチマーク関数を適当な軸で回転させた関数を対象として，その性質を検証することを予定している．タンパク質立体構造についても，既存のポテンシャルエネルギーを用いた最適化実験^{12),15)}をランダムステップ ODLS で改めて検証することを予定している．

6. ま と め

直交計画に基づいて近傍点集合を作成し，その近傍点集合から改善方向を推定してその方向に探索を行う最適化手法 ODLS に，SA の特徴を導入して，大局的に最適解を探索できるようにした．近傍点集合までの距離をランダムにとること，その距離を基準にして勾配方向への二分探索を行うこと，二分探索時には関数値を乱数化して解の悪化を許容することの 3 点について，それぞれの効果を数値実験を通じて検証した．その結果，ランダムな距離と二分探索とを導入した場合の有効性を示すことができた．SA と比較したところ，ベンチマーク問題である Rastrigin 関数，Griewank 関数の変数の数が多い場合に，同一評価回数という条件下で，平均的に良い近似解が得られることが分かった．

謝辞 本研究は経済産業省プロジェクト「次世代情報処理基盤技術開発」にて行われた．関係諸氏に謝意を表す．本稿は投稿後に査読委員からの数度にわたる有益な助言によってかなりの修正を行った．ここに記して謝意を表す．

参 考 文 献

- 1) 中村，有坂（編）：タンパク質のかたちと物性，共立出版（1997）．
- 2) 並列応用三菱研究室：タンパク質立体構造予測～物理・統計融合型シミュレーション，*RWC Final Exhibition & Symposia*（2001）．

- 3) Kirkpatrick, S., et al.: Optimization by Simulated Annealing, *Science*, Vol.220, pp.671-680 (1983).
- 4) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley (1989).
- 5) Nemhauser, G., et al. (編)：最適化ハンドブック，朝倉書店（1995）．
- 6) Torcson, V.J.: Multi-Dimensional Search: A Direct Search Algorithm for Parallel Machines, Ph.D. Thesis, Rice U. (1989).
- 7) Montgomery, D.C.: *Design and Analysis of Experiments*, 4th Edition, John Wiley & Sons, Inc. (1997).
- 8) 樫村，白鳥，于：実験計画法による非線形問題の最適化，朝倉書店（1998）．
- 9) Tanaka, H.: Local Search Using Orthogonal Design of Experiment, *Proc. Intl. Conf. on Parallel and Distributed Processing Techniques and Applications*, pp.651-657 (2000).
- 10) 田口：実験計画法，第 3 版上下，丸善（1976）．
- 11) 三木ほか：適応的近傍を持つ温度並列シミュレートドアニーリング，情報処理学会誌，Vol.42, No.4, pp.745-753 (2001)．
- 12) Shiraishi, M., et al.: Evaluating Global Optimization Algorithms on Molecular Potential Energy Function, *Proc. 2nd Intl. Conf. on Software Engineering, Artificial Intelligence, Network and Parallel/Distributed Computing*, pp.71-77 (2001).
- 13) Ingber, L.: *Simulated Annealing; Practice versus theory*, Lester Ingber Research (1993).
- 14) SA C++ package. <http://www.taygeta.com/>
- 15) 田中ほか：分子ポテンシャル最小化問題に関する並列局所探索法の比較評価，数理モデル化と問題解決研究会 MPS-33-12 (2001)．

（平成 14 年 4 月 11 日受付）

（平成 14 年 5 月 29 日再受付）

（平成 14 年 7 月 9 日再々受付）

（平成 14 年 8 月 27 日再々々受付）

（平成 14 年 9 月 20 日採録）

田中 秀俊（正会員）



1963 年生．1986 年東京大学工学部計数工学科卒業．同年三菱電機（株）入社．1989 年～1995 年新世代コンピュータ技術開発機構へ出向．1997 年～2001 年次世代情報処理基盤技術開発へ参画．IEEE Computer Society，ISCB，日本バイオインフォマティクス学会各会員．