

## ノイズを用いた局所探索法

田中秀俊\*, 白石将\*, 川上かおり\*, 青山功+, 佐藤裕幸\*  
\*三菱電機 情報技術総合研究所 +三菱電機 鎌倉製作所

直交計画法で探索方向を決定する局所探索法 ODLS(Orthogonal Design Local Search)は、同一の変数値組でも異なる値を返すようなランダムノイズ要素を持つブラックボックスの局所的最適化で高性能を発揮する。この ODLS にシミュレーテッドアニーリング(SA)の機能を導入することで、局所解多数を含むような関数最適化問題を高速に解く試みを行っている。局所解脱出のために、SA の可変ステップ幅を導入して、ODLS の探索方向決定に用いる観測点を現在点からランダムな距離にとり、SA の解悪化許容機能を導入して、線形探索時の関数値から正の一様整数乱数を引くようにしたところ、関数評価回数限定でのベンチマーク問題による比較では、SA と同等ないし上回る性能を示した。

### Local Search Method using Noises and Randomness

Hidetoshi TANAKA\*, Masashi SHIRAIISHI\*, Kaori KAWAKAMI\*, Isao AOYAMA+, Hiroyuki SATO\*  
\*Information Technology R&D Center, +Kamakura Works, Mitsubishi Electric Corp.

Orthogonal Design Local Search (ODLS) estimates gradient vector of noisy functions. The neighborhood set by orthogonal design of random distance enables ODLS go out of local minima. Performances of Simulated Annealing (SA) and ODLS are compared through the famous benchmark problems: Rastrigin Function, Griewank Function, and Schwefel Function. They clarified that ODLS outperforms SA in average when the functions have the large number of variables.

## 1. はじめに

非線形関数最適化問題で目的関数の導関数が入手困難な場合、局所探索法(Local Search, LS)が有効である。LS は初期値を与えて探索を開始し、探索中は、ある点(現在点)の周囲の点(近傍点)の中から良い点を選んで次の現在点とする。導関数が入手困難な場合は、ランダムに近傍点を選択して現在点と比較する方法がよく採用され、遺伝的アルゴリズム、シミュレーテッドアニーリング(Simulated Annealing, SA)、タブーサーチなどはこれに該当する。一方、直交計画法局所探索法(Orthogonal Design Local Search, ODLS)【1~6】のように、直交計画法で近傍点選択と次の現在点の合成とを行うことにより、勾配方向を推定して探索方向を決定する方法もある。これは近傍点集合に対して変数毎に評価値の平均を比較するため、変数が多いほど有効であり、特に同一の変数値組でも異なる値を返すようなランダムノイズ要素を持つ関数の局所的最適化では、高い性能を発揮する。また、近傍点集合の関数評価は並列に行えるため、並列処理による高速化が容易である。

勾配方向へ探索を進める基本 ODLS は、局所解に陥った場合にそこから脱出する機能を持たなかった。今回示す改良 ODLS には、SA の特徴であるランダムなステップ幅と、ランダムな関数値悪化許容幅を導入して、局所解脱出機能を付加した。第2章でその導入方式を示し、第3章でSAとの数値実験結果を示し、第4章で導入効果と一般性について検討する。

## 2. ODLS

### 2.1. 基本 ODLS

一次導関数形が入手可能な無制約多変数非線形最適化問題では、最急降下法がよく用いられる。これは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時点でまた導関数値を入力する方法である。LS で現在点の一次導関数値が入手困難な場合、よく用いられるのは近傍点からランダムに次の現在点を選ぶような方法だが、近傍点から導関数値を推定して最急降下法を適用することもできる。導関数値の簡単な推定法としては、変数間の相関が少ないと仮定して、各変数で独立に導関数値を推定し、それを合成して勾配方向を求める方法がある。この時、変数を現在点から一定幅の増加(+ )もしくは減少(- )の2値に単純化すると、2水準直交計画によって勾配方向を推定できる。基本 ODLS では、以下のような手順で探索を行う。

1. 近傍点を選択する。近傍点数  $m$ 、および近傍点  $x_i$  の変数  $j$  の値  $X_{ij}$  は、変数の数を  $n$ 、現在点と近傍点の距離を決める正の数を  $w$ 、 $i$  を  $q$  桁のグレーコードで表したうちの  $s$  桁目を  $g_s$ 、 $0 \sim m-1$  のランダムな並び替えを  $C()$ 、 $k=C(j)$  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  $\text{mod}_2$  とすると、式(1)で表される。

$$X_{ij} = 2w \lfloor L_{ij} - 0.5 \rfloor; \quad L_{ij} = \text{mod}_2 \sum_{s=0}^{q-1} g_s b_{q-s-1}; \quad m = 2^q; \quad 2^{q-1} \leq n < 2^q; \quad q \in \text{整数} \quad (1)$$

2. 各近傍点  $x_i$  について関数値  $f(x_i)$  を求める。
3. 各変数の勾配方向を求める。現在点と次の現在点との距離を決める正の数  $d$  と、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  $j$  の勾配成分  $e_j$  は式(2)で表される。

$$e_j = \begin{cases} -d & (\mu_j^- + B < \mu_j^+) \\ +d & (\mu_j^+ + B < \mu_j^-) \\ 0 & (\text{otherwise}) \end{cases} \quad \mu_j^+ = \frac{2}{m} \sum_{i=0; L_{ij}=1}^n f(x_i) \quad \mu_j^- = \frac{2}{m} \sum_{i=0; L_{ij}=0}^n f(x_i) \quad (2)$$

他の局所探索法が近傍点集合を 1 点ないしはたかだか数十点とすることが多いのに対し、ODLS は例えば 1000 変数の場合 1024 点を要する。この多量の関数評価回数に加え、512 点ずつの + と -、計 2000 通りの平均を算出して 1000 組の比較をするという大きなオーバーヘッドを抱える。しかし、1000 次元の格子点を考えたときの近傍点全体  $3^{1000} \approx 10^{478}$  もの本来の近傍点の中からランダムにたかだか数十点選択する方法に比べ、着実によい勾配方向が得られると期待できる。

## 2.2. SA(Simulated Annealing)

SA は探索空間において現在値の近傍を探索する際、評価関数値が高くなる方向への現在値更新を許す点に特徴があり、これによって局所解からの脱出が可能となる。評価関数値が高くなる方向への現在値更新の頻度を制御するために、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、現在値近傍の範囲が大きくなり、評価関数値が高くなる現在値への更新確率が増える【7】。

SA は *melting* と *cooling* の 2 段階の処理からなる。*melting* 過程では、低温から出発し近傍全点が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。現在値近傍にはガウシアン分布を用いる方法や *cauchy* 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【8,9】。本稿ではこの急速冷却を行うものを SA と呼んでいる。

## 2.3. 改良 ODLS

ODLS は、勾配推定をベースとするため、局所解脱出の機能がないという欠点があった。また比較実験【2~5】によると、1 回の改善幅に対する前述のオーバーヘッドが大きいため、収束速度が遅いという欠点があり、総合的には SA に劣っていた。そこで、局所解脱出機能を持たせることと、高速化を図ることの 2 点に関して、改良を試みた。まず、上述の現在点と近傍点の距離係数  $w$  と、現在点と次の現在点との距離係数  $d$  については、各反復で  $w = d$  としていたが、高速化のために、勾配方向への線形探索を用いて最適な  $d$  を求めるようにした。また、局所解脱出のために、 $w$  を所与の幅の一樣乱数で決定して、さらに線形探索時には関数値に所与の幅の一樣乱数をノイズとして付加するようにした。

### 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(3)~(5)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。ノイズ  $N$  は適当な幅の正の一様整数乱数で、 $N=0$  のときに各関数の本来の値になる。 $R(x)$ ,  $G(x)$ の最小値は0、 $S(x)$ の最小値はおよそ  $-419 \times n$  である。

$$R(x) = 10n + \sum_{i=1}^n \left[ \left( \frac{x_i}{100} \right)^2 - 10 \cos \left( \frac{2\pi x_i}{100} \right) - N \right] \quad (3)$$

$$G(x) = 1 + \sum_{i=1}^n \left[ \frac{x_i^2}{4000} - N \right] - \prod_{i=1}^n \cos \left( \frac{x_i}{\sqrt{i}} \right) \quad (4)$$

$$S(x) = \begin{cases} \sum_{i=1}^n \left[ -\xi_i \sin(\sqrt{|\xi_i|}) - N \right] & (\xi_i = 512 - \text{mod}_{512} x_i; \quad x_i \geq 512) \\ \sum_{i=1}^n \left[ -x_i \sin(\sqrt{|x_i|}) - N \right] & (-512 \leq x_i < 512) \\ \sum_{i=1}^n \left[ -\eta_i \sin(\sqrt{|\eta_i|}) - N \right] & (\eta_i = -512 + \text{mod}_{512} |x_i|; \quad x_i < -512) \end{cases} \quad (5)$$

SA はフリーソフト【9】を用い、各関数で簡単なパラメータサーベイを実施して得た設定【5】と同じ設定を用いた。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5万回に限定して、アルゴリズムの一定時間内での性能を比較した。今回の実験では共通して式(1)の  $w$  を  $1 \sim 250$  の一様整数乱数とし、式(2)の  $B$  は0とした。

### 4. 実験結果

各関数にノイズを加えない状態で比較を行った。図1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つあたりの悪さを図1左および中中で示している。 $S(x)$ では真の最小値がおよそ  $-419 \times$  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つあたりの悪さを図1右で示している。 $R10, R100, R1000, R2000$  は  $R(x)$ 、 $G10, G100, G1000, G2000$  は  $G(x)$ 、 $S10, S100, S1000, S2000$  は  $S(x)$ の、それぞれ変数の数  $n$  を  $10, 100, 1000, 2000$  としたもので、探索開始点を  $-512 \sim +511$  の間の一様乱数を成分とする点としている。図中の各6値は、探索開始点を変えて10回ずつ試行した結果で、順に ODLS の最小、平均値、最大、SA の最小、平均、最大を示している。 $G(x)$ だけは縦軸を対数にとっている。

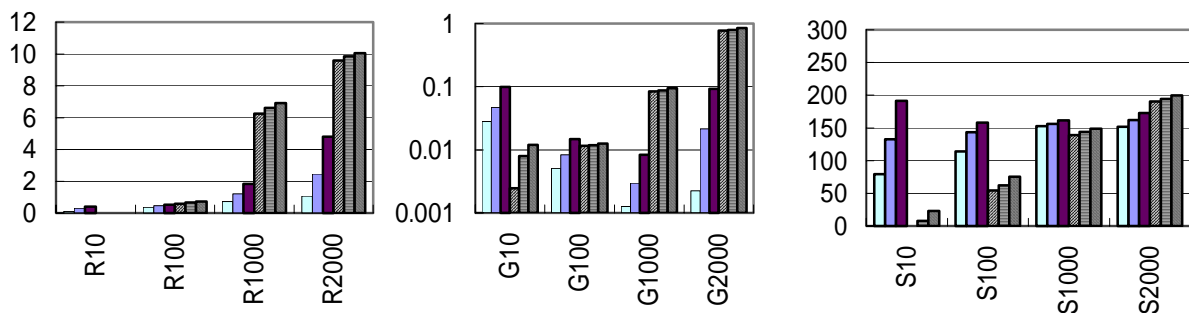


図1 変数の数に対する解の近似度の比較 (評価回数5万回、各10試行)

次に、 $n=1000$  として各関数の式(3)~式(5)の  $N$  を  $0 \sim 100$  まで各10試行して、ODLS 内で性能を比較した結果を図2に示す。

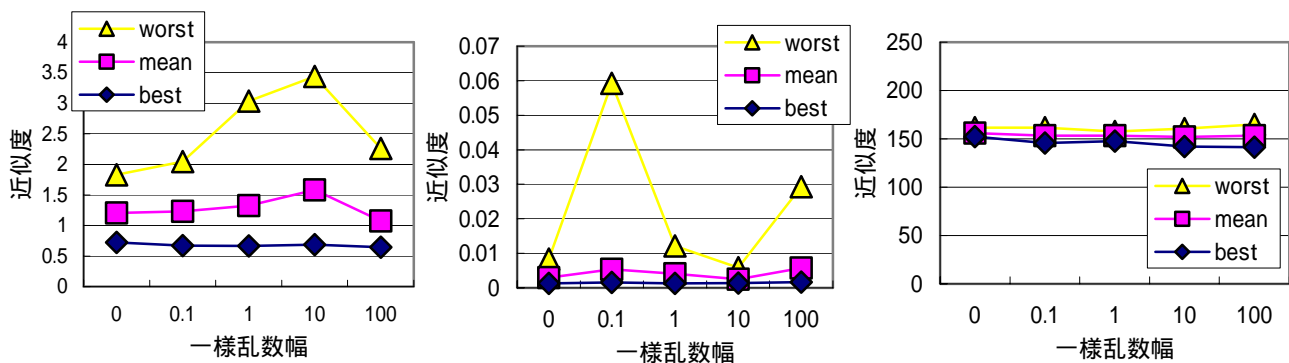


図2 線形探索時の乱数幅に対する解の近似度の比較 (評価回数 5 万回、各 10 試行)

## 5. 考察

図1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、 $n=1000$  および  $2000$  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、 $S(x)$  は  $n=100$  でもまだ SA が良く、 $n=1000$  および  $n=2000$  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への勾配が得られにくく、勾配推定をベースとしている ODLS では性能が発揮できなかったためと考えられる。

悪化を許容するために導入したノイズの効果は、図2によれば、 $R(x)$ 、 $G(x)$ 、 $S(x)$  いずれに対しても、良い影響も悪い影響も与えていない。

## 6. まとめ

直交計画を用いた勾配推定による関数最適化手法に、シミュレーテッドアニーリングの特徴である可変ステップ幅、線形探索、乱数化を導入した。その結果、可変ステップ幅と線形探索の導入により、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、シミュレーテッドアニーリングよりも平均的によい近似解が得られることが分かった。

### 【参考文献】

- [1] 田中, 他: “直交計画法によるノイズつき多次元関数の勾配推定”, 第 61 回情処全国大会, 1D-06, 2000-9.
- [2] 田中, 他: “並列局所探索法の比較評価”, 第 62 回情処全国大会, 2R-04, 2001-3.
- [3] 田中, 他: “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, 2001-3.
- [4] 白石, 他: “タンパク質立体構造予測問題に関する最適化アルゴリズムの比較評価”, 第 63 回情処全国大会, 2P-01, 2001-9.
- [5] 青山, 他: “多峰性関数に対する最適化アルゴリズムの探索性能比較”, 第 63 回情処全国大会, 2P-04, 2001-9.
- [6] 田中, 他: “直交計画法を用いた局所探索法の改良”, 第 64 回情処全国大会, 3D-05, 2002-3.
- [7] Frost, R. and Hieneman, P. Simulated Annealing: A Heuristic for Parallel Stochastic Optimization. PDPTA '97 (1997)
- [8] Ingber, L. Simulated Annealing; Practice versus theory. Lester Ingber Research (1993).
- [9] SA C++ package, <http://www.taygeta.com/>