

Parallel Simulated Annealing with Adaptive Neighborhood

Hiroki HIRAO^{*}, Yuichiro UEDA^{*}, Mitsunori MIKI^{**} and Tomoyuki HIROYASU^{***}

(Received January 20, 2009)

Simulated Annealing(SA) is one of the general heuristic methods to solve the optimization problems. In the case that SA is applied to continuous problems, the determination of the neighborhood is very important. However, the appropriate neighborhood range depends on target problems and their dimensions. Therefore it is not easy to find the appropriate neighborhood range. The solution to this problem is the introduction of an adaptive mechanism for changing the neighborhood range into SA method. In this paper, we propose the new method with multiple neighborhood ranges by parallelization, compare them, and it is found that the proposed method is very useful and effective.

Key words : optimization, simulated annealing, parallel, adaptive neighborhood

キーワード : 最適化, シミュレーテッドアニーリング, 並列, 適応的近傍

並列型近傍幅調節機能を持つシミュレーテッドアニーリング

平尾 洋樹・上田 祐一郎・三木 光範・廣安 知之

1. はじめに

シミュレーテッドアニーリング (Simulated Annealing:SA) は焼き鈍しと呼ばれる加熱炉内の個体の冷却過程をシミュレートした, 進化的最適化手法である¹⁾. SAは組合せ最適化問題に特に有効な手法であるが, 連続最適化問題においても対象とする問題の複雑度が高い場合には多く用いられる²⁾.

SAには, 温度および近傍と呼ばれるパラメータがある. 特に, 連続最適化問題にSAを適用する場合, 近傍は現状態から次状態へ摂動し得る最大の範囲であり, これを近傍幅と呼ぶ. 一般的に, 近傍幅が大きい場合には大域的最適解を得やすくなるが無駄な探索が

多くなり, 近傍幅が小さい場合には局所的な探索性能は向上するが局所最適解に収束しやすくなる. このため, 近傍幅の設計は探索性能に大きな影響を与える¹⁾.

近傍幅に関する研究は多くなされてきた³⁾. 一般的に, 探索序盤では大きな近傍幅で大域的探索を, 探索終盤では小さな近傍幅で局所的探索を行うことにより, 解精度が向上することがわかっている. しかし, 近傍幅を小さくする時期を誤ると局所解に陥る場合がある. 近傍幅が小さい状態で局所解に陥った場合は, その局所解を抜け出すことができないため, 近傍幅のスケジュールを調節することが重要となる⁴⁾.

これに対し, 先行研究により受理率に依存する適応的の近傍を持つSA (SA with Advanced Adaptive

^{*} Graduate Student, Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto
Telephone:+81-774-65-6921, Fax:+81-774-65-6716, E-mail:yueda@mikilab.doshisha.ac.jp

^{**} Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto
Telephone:+81-774-65-6930, Fax:+81-774-65-6716, E-mail:mmiki@mail.doshisha.ac.jp

^{***} Faculty of Life and Medical Sciences, Doshisha University, Kyoto
Telephone:+81-774-65-6932, Fax:+81-774-65-6019, E-mail:tomo@is.doshisha.ac.jp

Neighborhood:SA/AAN)⁵⁾ や近傍並列 SA (Neighborhood Parallel SA:NPSA)⁶⁾ などの手法が提案されている。SA/AAN は探索中の受率率を用いて近傍幅を自動調節する手法であり, NPSA は異なる近傍幅を用いて並列に探索を行う手法である。両手法とも近傍幅のチューニングを自動化した手法であるが, 前者では目標とする受率率を決定するための予備実験が必要になり, 後者では十分な近傍幅を用いるためには多くのプロセッサが必要となる。

本研究では, 近傍幅の自動調節と並列化に注目し, 各プロセスが適応的な近傍幅の調節を行う並列 SA として, 2 並列型近傍調節機能を持つ SA (PSA/AN(2N)) および 3 並列型近傍調節機能を持つ SA (PSA/AN(3N)) を提案する。提案手法は, 異なる近傍幅を持つ複数のプロセスで探索を行い, 一定の探索回数ごとに各プロセスの情報交換を行う。また, 各プロセスの探索における近傍幅は, 解の更新に基づいて適応的に調節される。これにより, 近傍幅のチューニングを自動化し, 少ない並列数での効率的な探索が期待できる。提案手法を代表的な数学関数最小化問題に適用し, その有効性を示す。

2. シミュレーテッドアニーリング (SA)

SA のアルゴリズムを Fig. 1 に示す。まず温度 T を初期設定した後, 与えられた初期状態 x_0 から出発して次状態 x' を生成し, その評価値 E' を計算する。なお, 次状態を生成し得る最大の範囲を近傍幅と呼ぶ。次に, 評価値の差分 $\Delta E (= E' - E)$ と温度 T_k に応じて受率判定を計算し, 受率の場合は次状態に推移する。受率判定には, 式 (1) に示すメトロポリス基準を用いる。この処理を繰り返し, 現在の温度 T_k で平衡状態に達したら, 冷却処理を行い次の温度 T_{k+1} を求め, 再び探索を進める。この冷却処理には, 式 (2) を用いる。十分温度が下がり終了条件に達したら探索を終了する。なお, 探索条件には温度や探索回数などが考えられ, 本研究では探索回数を終了条件としている。

$$A(E, E', T) = \begin{cases} 1 & \text{if } \Delta E < 0 \\ \exp\left(-\frac{\Delta E}{T}\right) & \text{otherwise} \end{cases} \quad (1)$$

$$T_{k+1} = \gamma T_k (0.8 \leq \gamma < 1) \quad (2)$$

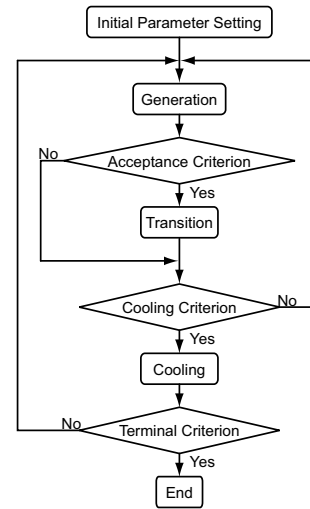


Fig. 1. Algorithm of Simulated Annealing.

3. 対象問題

本研究で対象とする連続関数最小化問題は次の2つの関数である。式 (3) に示す Rastrigin 関数⁷⁾ は, 局所解が格子状に存在する多峰性関数であり, 次元数が2の場合, 100 個の局所解を持つ。式 (4) に示す Griewank 関数⁷⁾ は, 設計変数間に依存関係を有する多峰性関数である。大域的には単峰性関数のような性質をもつが, 局所的には多数の局所最適解が存在する。

次元数を3とすると, Rastrigin 関数, Griewank 関数の準最適解の値はそれぞれ約 9.94×10^{-1} , 約 7.4×10^{-3} である。本論文では, これよりも低い評価値をとる設計変数の領域を最適解領域と定義する。

$$f_R(\mathbf{x}) = (N \times 10) + \sum_{i=1}^N (x_i^2 - 10 \cos(2\pi x_i))$$

定義域 : $-5.12 < x_i \leq 5.12$,
 最適解 : $\min(f_R(\mathbf{x})) = f_R(0, 0, \dots, 0)$,
 最適値 : $f_R(\mathbf{x}) = 0$ (3)

$$f_G(\mathbf{x}) = 1 + \sum_{i=1}^N \left(\frac{x_i^2}{4000} \right) - \prod_{i=1}^N \left(\cos \left(\frac{x_i}{\sqrt{i}} \right) \right)$$

定義域 : $-512 < x_i \leq 512$,
 最適解 : $\min(f_G(\mathbf{x})) = f_G(0, 0, \dots, 0)$,
 最適値 : $f_G(\mathbf{x}) = 0$ (4)

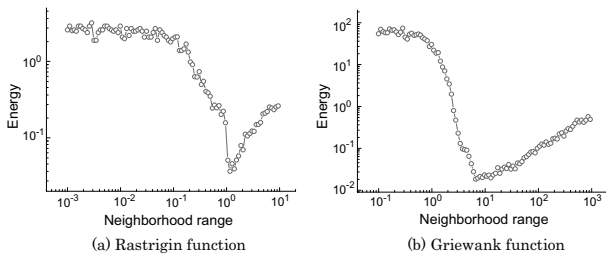


Fig. 2. Effect of the neighborhood range on the energy.

4. 連続最適化問題における SA

4.1 近傍

連続最適化問題に SA を適用する場合、近傍はユークリッド空間内での距離に相当し、自由に決めることができる。この範囲を近傍幅と呼ぶ。1 章で述べたように近傍幅の設計は探索性能に大きな影響を与える¹⁾。

ここで、近傍幅が探索に与える影響として、解精度と近傍幅の関係を Fig. 2 に示す。Fig. 2 は横軸に近傍幅、縦軸に評価値を示し、それぞれの近傍幅において 30 試行探索を行って得た評価値の中央値である。なお、次元数は 3 である。Fig. 2 より、3 次元の Rastrigin 関数では約 1.0、Griewank 関数では約 6.0 付近で評価値が良好である。なお、これらの値は各問題における局所解どうしの間隔と一致する。したがって、SA には探索の精度が良くなる近傍幅が存在し、この値は問題に依存すると言える。また、適切な近傍幅を設定しないと、探索の精度は悪化する。

4.2 近傍幅に対するアプローチ

前説で述べたように、SA における探索の精度は近傍幅に大きく依存する。また、各対象問題にはそれぞれ効率的な探索を行うことができる近傍幅が存在する。しかし、この近傍幅を決定するには膨大な予備実験が必要である。また、対象問題が複雑になると、探索状況によって適切な近傍幅が異なることも考えられる。このため、探索状況に応じた近傍幅を適応的に決定するアルゴリズムが重要と考えられる。

先行研究では、この課題に対して大きく 2 種類のアプローチが提案されている。1 つは逐次手法である「受理率に依存する適応的近傍を持つ SA(SA/AAN)⁵⁾」

であり、もう 1 つは並列手法である「近傍並列 SA(NPSA)⁶⁾」である。

SA/AAN では、探索中の受理率を用いて近傍幅を調節する。受理率とは、探索中に生成した一定数の次状態のうち受理した割合を示す。この受理率は近傍幅や探索状況に依存するため、この受理率に基づいた近傍幅調節を行うことで、探索状況に応じた近傍幅を自動設定することができる⁵⁾。一方 NPSA では、異なる近傍幅を用いた探索を複数のプロセスを用いて行い、一定間隔ごとに各プロセスで得た解を交換することで、探索状況に応じた近傍幅を得る手法である⁶⁾。

両手法とも近傍幅を自動調節できる手法である。しかし、SA/AAN では目標受理率という新たなパラメータを予備実験によって求める必要がある。また NPSA では、各プロセスで用いる近傍幅を経験的な値として与えるしかなく、この値を固定して探索を行うために、解交換をしたとしても、真に探索状況に適応しているとは言い難い。この問題は、プロセス数を増やし多種の近傍幅による探索を行えば解決できるが、そのためには多数のプロセスが必要となりコストがかかる。

本研究では、この近傍幅の自動調節と並列化に注目し、少ない並列数による探索手法として、「2 並列型近傍調節機能を持つ SA (PSA/AN(2N))」および「3 並列型近傍調節機能を持つ SA (PSA/AN(3N))」を提案する。これらの手法は、並列化によって異なる近傍幅を用いた SA による探索を行うと共に、各プロセスの探索においても近傍幅を適応的に調節するアルゴリズムを有する。単なる並列化だけでなく、各プロセスにおいても近傍幅を適応的に自動調節することで、多数のプロセスを用いずに探索状況に適応した近傍幅を自動的に調節することができる。以降、これら提案手法について詳しく述べる。

5. 2 並列型近傍幅調節機能を持つ SA

(PSA with Adaptive Neighborhood using Two Neighborhood ranges : PSA/AN(2N))

5.1 基本概念

一般的に、現在の探索点が最適解領域であれば近傍幅を小さくすることで局所的探索を、局所解付近であれば近傍幅を大きくすることで大域的探索を、それぞれ

行う必要がある。そこで本手法では、大小2つの異なる近傍幅を用いて並列探索を行う。そして、一定周期(同期間隔)ごとに各プロセスにおいて、探索状況に応じた近傍幅の調節を行う。これにより、少ない並列数で大域的探索と局所的探索のメカニズムを同時に有する探索が可能となり、探索状況に適した近傍幅の自動調節が期待できる。

5.2 アルゴリズム

本手法では、2つのプロセスに大小異なる近傍幅を設定する。ここで、それぞれのプロセスを大近傍、小近傍と呼ぶ。そして、これら2つの近傍幅を自動調節しながら探索を進める。

この自動調節のメカニズムの詳細を以下に述べる。なお、以下の式(5)から式(7)における N_l , N_s は大近傍、小近傍における近傍幅の値、 r は拡大率を示す。また、Fig. 3にこのメカニズムの概念図を示す。

(a-1) 大近傍で最良値が更新されたとき

$$\begin{cases} N_{l_{next}} = N_{l_{cur}} \\ N_{s_{next}} = N_{s_{cur}} \times r \end{cases} \quad (5)$$

(a-2) 小近傍で最良値が更新されたとき

$$\begin{cases} N_{l_{next}} = N_{l_{cur}} \times 1/r \\ N_{s_{next}} = N_{s_{cur}} \end{cases} \quad (6)$$

(b) 最良値の更新がない(拡大・縮小する)とき

$$\begin{cases} N_{l_{next}} = N_{l_{cur}} \times r \\ N_{s_{next}} = N_{s_{cur}} \times 1/r \end{cases} \quad (7)$$

本手法では、一定周期(同期間隔)ごとに各プロセスで最良値の更新が行われたかを判定する。式(5)、式(6)に示すように、大近傍が良好ならば拡大、小近傍が良好ならば縮小し、更新判定時に良好とされる近傍幅を残しつつ、大小2つの近傍幅の大きさを変化(近傍幅調節)する。一方、最良値の更新がないときは式(7)に示すように、大近傍を拡大し小近傍を縮小し、この操作を最良値の更新が行われるまで繰り返す。

ただし、近傍幅の上限を設計変数空間全体の大きさ、下限をその 10^{-3} 倍の大きさあらかじめ決めておき、その範囲を超えないようにする。また、下限において小近傍の最良値が微小に更新されても、最良値の更新があったとはみなさず、2つの近傍幅を変化させない。

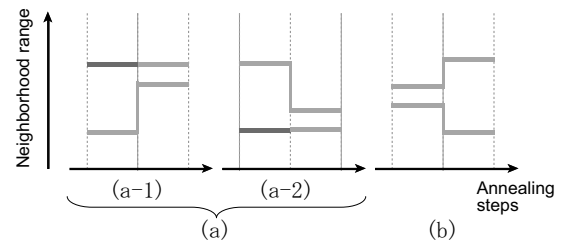


Fig. 3. Change in neighborhood of PSA/AN(2N).

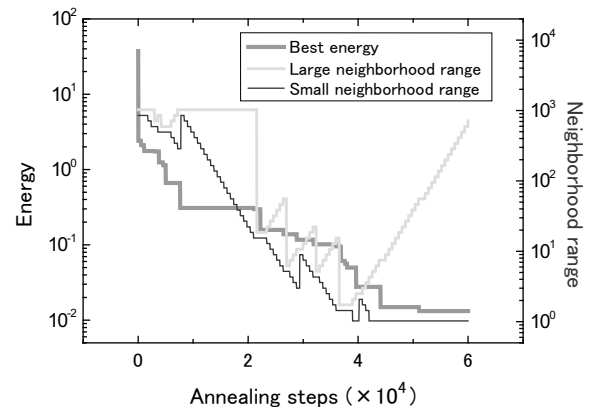


Fig. 4. History of Energy and Neighborhood ranges in PSA/AN(2N).

なお、同期後はこれまでの探索で最良値を記録した点から探索を再開する。これにより、最良値を記録した点が最適解領域の場合には、局所的探索により高い解精度を得られることが保証される。

ここで、本手法の最適解領域に到達した試行の探索序盤における最良値と近傍幅の履歴を Fig. 4に示す。

Fig. 4より、探索序盤で小近傍が拡大した際、最良値が更新されたときがある。これは大近傍の方が良好な結果を得ることで、局所解を抜け出したことを表している。これより、局所解に陥っても大近傍によって局所解に抜け出しやすい探索を行っていると言える。

6. 3 並列型近傍幅調節機能を持つ SA

(PSA with Adaptive Neighborhood using Three Neighborhood ranges : PSA/AN(3N))

6.1 基本概念

本手法では、大中小3つの異なる近傍幅を用いて並列探索を行う。そして、一定周期(同期間隔)ごとに各

プロセスにおいて、探索状況に応じた近傍幅の調節を行う。これにより PSA/AN(2N) と同様に、少ない並列数で大域的探索と局所的探索のメカニズムを同時に有する探索が可能となり、探索状況に適応した近傍幅の自動調節が期待できる。

6.2 アルゴリズム

本手法では、3つのプロセスに大中小異なる近傍幅を設定する。ここで、それぞれのプロセスを大近傍、中近傍、小近傍と呼ぶ。そして、これら3つの近傍幅を自動調節しながら探索を進める。

この自動調節のメカニズムの詳細を以下に述べる。なお、以下の式(8)から式(11)における N_l , N_m , N_s は大近傍、中近傍、小近傍における近傍幅の値、 r は拡大率を示す。また、Fig. 5にこのメカニズムの概念図を示す。

(a-1) 大近傍で最良値が更新されたとき

$$\begin{cases} N_{l_{next}} = N_{l_{cur}} \times r \\ N_{m_{next}} = N_{m_{cur}} \times r (= N_{l_{cur}}) \\ N_{s_{next}} = N_{s_{cur}} \times r (= N_{m_{cur}}) \end{cases} \quad (8)$$

(a-2) 中近傍で最良値が更新されたとき

$$\begin{cases} N_{l_{next}} = N_{m_{cur}} \times r \\ N_{m_{next}} = N_{m_{cur}} \\ N_{s_{next}} = N_{m_{cur}} \times 1/r \end{cases} \quad (9)$$

(a-3) 小近傍で最良値が更新されたとき

$$\begin{cases} N_{l_{next}} = N_{l_{cur}} \times 1/r (= N_{m_{cur}}) \\ N_{m_{next}} = N_{m_{cur}} \times 1/r (= N_{s_{cur}}) \\ N_{s_{next}} = N_{s_{cur}} \times 1/r \end{cases} \quad (10)$$

(b) 最良値の更新がない(拡大・縮小する)とき

$$\begin{cases} N_{l_{next}} = N_{l_{cur}} \times r \\ N_{m_{next}} = N_{m_{cur}} \\ N_{s_{next}} = N_{s_{cur}} \times 1/r \end{cases} \quad (11)$$

本手法でも PSA/AN(2N) と同様に、大中小3つの近傍幅を探索中の解の更新状況に応じて変化させる。最良値の更新があるとき、式(8)から式(10)に示すように、次周期においては、更新があった近傍を中近傍として探索を続ける。一方、最良値の更新がないとき、式(11)に示すように、中近傍をそのままにして、

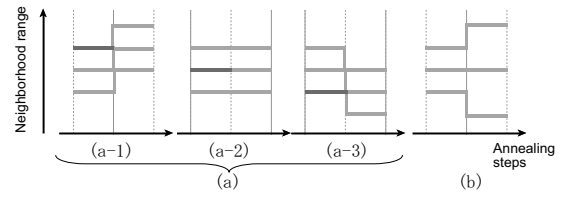


Fig. 5. Change in neighborhood of PSA/AN(3N).

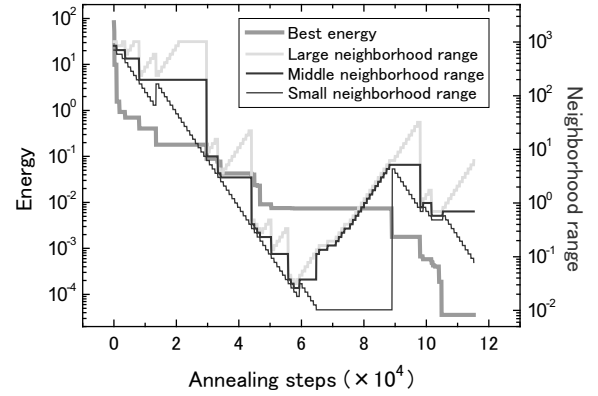


Fig. 6. History of Energy and Neighborhood ranges in PSA/AN(3N).

大近傍を拡大し、小近傍を縮小させる。それでも更新がないとき、再びこの操作を更新が行われるまで繰り返す。

また最良値の更新が行われなかったとき、小近傍が下限に達していれば、大近傍だけでなく中近傍も拡大し大域的探索を行う。これにより、局所解から抜け出しやすくなると考えられる。

ここで、本手法の最適解領域に到達したある1試行における評価値と近傍幅の履歴を、Fig. 6に示す。

Fig. 6より、探索中盤において近傍幅が縮小しているにも拘わらず最良値の更新がないことから、局所解に陥っていると考えられる。しかし、その後大近傍と中近傍を拡大していく中で最良値が更新されている。これは局所解から抜け出していると考えられる。さらに、探索終盤において小近傍を縮小していく中で最良値が更新されている。これは最適解領域において近傍幅を縮小することで局所的探索を行い、高い解精度が得られていることを表している。これより、近傍幅の拡大メカニズムが、有効に機能していることが確認された。

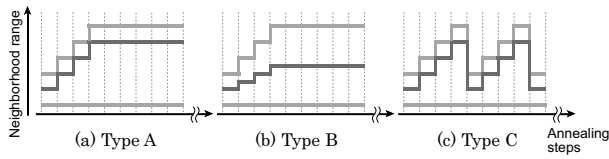


Fig. 7. Change in neighborhood of PSA/AN(3N).

Table 1. Ratio of solutions that reached the optimum solution area.

Algorithms	Type A	Type B	Type C
Rastrigin function	66%	86%	93%
Griewank function	54%	55%	59%

6.3 中近傍調節アルゴリズムの検討

本研究では、PSA/AN(3N)の中近傍調節アルゴリズムとして、Fig. 7に示す3種類について検討する。

- Type A : 大近傍とともに拡大する。
- Type B : 大近傍と小近傍の相乗平均とする。
- Type C : 大近傍とともに拡大し、

上限に達した場合は下限まで戻す。

これら3つの近傍調節アルゴリズムのうち、中近傍が最も有効に機能するものを検討するため、各アルゴリズムで100試行を行い、最適解領域への到達率をTable 1に示す。

Table 1よりいずれの関数においても、Type Cが最も良好な結果を示した。Type AやType Bでは最適な近傍幅を超えて拡大しすぎてしまうことが多い。また、相乗平均では最適な大きさに達しないことも考えられる。一方Type Cでは、中近傍が上限に達した場合に下限まで戻すため、最適な大きさを超えて拡大しても、この拡大操作を数回行うことで局所解を抜け出す可能性が向上していると考えられる。以上より、本手法ではType Cを用いることとする。

7. 数値実験

7.1 実験概要

前節で述べたように、PSA/AN(3N)の中近傍は上限まで拡大した場合に下限まで戻すことが良い。そこで、PSA/AN(2N)の大近傍も同様に、上限まで拡大

した場合は下限まで戻すことにする。

その上で、PSA/AN(2N)とPSA/AN(3N)の性能比較を行う。対象問題は3章に示す各問題である。各パラメータはTable 2とTable 3に示す予備実験により求めた最適な値を用いており、3並列型SAは2並列型SAに対して探索回数を2/3倍とし評価計算回数をそろえている。ここで、提案手法においては、一般的なSAに「同期間隔 (Synchronization interval)」、 「拡大率 (Increasing ratio)」という2つのパラメータを加えている。PSA/ANのアルゴリズムにおいて、クーリング周期中に、次周期の近傍幅を決定するために最良値の更新判定を数回行うが、その周期を同期間隔とする。また、次周期の近傍幅を決定するための倍率 r を拡大率とする。

7.2 結果と考察

各手法で100回試行を行い、各試行の最良値を昇順に並び替えた結果をFig. 8に示す。

各関数においてPSA/AN(2N)の方が高い解精度を得ている。これは、PSA/AN(3N)に対して1プロセスあたりの探索回数が多いため、最適解領域へ到達したとき局所的探索を長く行うことができるからである。

また、最適解領域への到達率は関数によって良好な手法が異なった。Rastrigin関数は設計変数空間の大きさ10.24に対して、最適な近傍幅の大きさが約1.0と近い値である。そのため、中近傍が有効に機能することが多かったと考えられる。一方、Griewank関数は設計変数空間の大きさ1024に対して、最適な近傍幅の大きさが約6.0と小さな値である。そのため、中近傍が有効に機能することが少なかったと考えられる。

以上より、PSA/AN(3N)よりPSA/AN(2N)の方が解精度が高いと言える。しかし、最適解領域への到達率については対象問題によって良好な手法が異なり、対象問題や目的に合わせて適切な手法を選択する必要があると言える。しかし、両手法とも少ないプロセス数でパラメータのチューニングを自動化できており、その有効性を示すことができた。

8. まとめ

本研究では、SAの重要なパラメータである近傍幅のチューニングを、少ないプロセス数を用いることで

Table 2. Parameters for Rastrigin function.

Methods	PSA/AN(2N)	PSA/AN(3N)
Total steps	57600	38400
Cooling number	32	32
Max. temperature	10.0	10.0
Min. temperature	0.01	0.01
Max. neighborhood range	10.24	10.24
Min. neighborhood range	10.24×10^{-3}	10.24×10^{-3}
Synchronization interval	450	400
Increasing ratio	1.5	2.0

Table 3. Parameters for Griewank function.

Methods	PSA/AN(2N)	PSA/AN(3N)
Total steps	115200	76800
Cooling number	32	32
Max. temperature	20.0	20.0
Min. temperature	0.001	0.001
Max. neighborhood range	1024	1024
Min. neighborhood range	1024×10^{-3}	1024×10^{-3}
Synchronization interval	180	150
Increasing ratio	2.0	2.0

自動化した, PSA/AN(2N) および PSA/AN(3N) を提案した. 数学関数最小化問題に適用しその性能比較を行った結果, 解精度については PSA/AN(2N) が高性能であったが, 最適解領域への到達率については対象問題によって良好な手法が異なった. したがって, 解精度や最適解への到達率など, 対象問題や目的に合わせて適切な手法を選択する必要がある. しかし, 両手法とも少ないプロセス数でパラメータの自動調節ができており, その有効性を示すことができた.

参 考 文 献

- 1) 喜多一, “シミュレーテッドアニーリング”, 日本ファジィ学会誌, Vol.9, No.6, pp.870-875 (1997).
- 2) L. Ingber, “Genetic Algorithms and Very Fast Simulated Reannealing: A Comparison”, Mathematical and Computer Modelling, Vol.16, No.11,

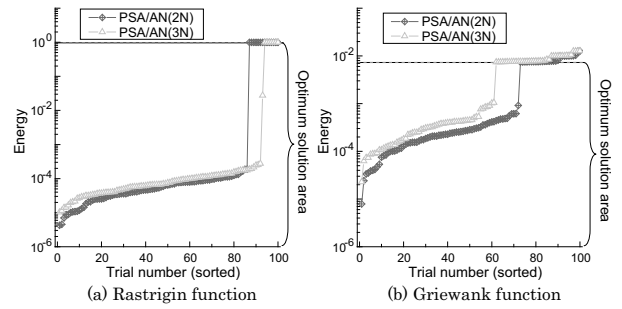


Fig. 8. Distribution of optimum solutions.

pp.87-100 (1992).

- 3) 上坂吉則, “シミュレーテッドアニーリングの摂動近傍と収束速度について”, 電子情報通信学会技術報告, (1990).
- 4) 三木光範, 廣安知之, 宮崎真, “2分木を用いた近傍幅調節機能を持つシミュレーテッドアニーリング”, 情報処理学会第67回全国大会講演論文集, Vol.1, pp.259-260 (2005).
- 5) 三木光範, 廣安知之, 小野景子, “最適な受理確率を目標とする適応的近傍を持つシミュレーテッドアニーリング”, 情報処理学会論文誌, Vol.44, No.1, pp.1-6 (2003).
- 6) 安藤景子, 三木光範, 廣安知之, 及川雅隆, “近傍並列シミュレーテッドアニーリング”, 計算工学会論文誌, No.20060026, (2006).
- 7) D. Whitley and K. Mathias and S. Rana and J. Dzubera, “Evaluating Evolutionary Algorithms”, Artificial Intelligence, Vol.85, pp.245-2761 (1996).