Multi-point Simulated Annealing with Adaptive Neighborhood

Keiko ANDO, Mitsunori MIKI, and Tomoyuki HIROYASU

Abstract—**When SA is applied to continuous optimization problems, the design of the neighborhood used in SA becomes important. A lot of experiments are necessary to determine an appropriate neighborhood range in each problem, because the neighborhood range corresponds to the distance in the Euclid space and is decided arbitrarily. We proposed a Multi-point Simulated Annealing with Adaptive Neighborhood (MSA/AN) for continuous optimization problems, which determines the appropriate neighborhood range automatically. The proposed method provides the neighborhood range from the distance and the design valuables of two search points, and generates candidate solutions using a probability distribution based on this distance in neighborhood, and selects the next solutions from them based on the energy. In addition, a new acceptance judgment is proposed for multi-point SA based on the Metropolis criterion. The proposed method shows a good performance in solving typical test problems.**

I. INTRODUCTION

Recently, there are many tasks wherein optimization such as resource and energy conservation is considered necessary. In conjunction with this, optimization for system has become important. Optimization problems are problems that find the optimum solution for maximizing or minimizing evaluation function within the given limitation conditions, and consistently crop up in real world scenarios.

Optimization problems can be divided into the following two categories: continuous optimization problems wherein the design variables are continuous values and combinatorial optimization problems wherein the design variables are discrete values. The former category of problems usually find the optimum solution by changing the design valuables with a continuous value based on objective function gradient information. Regarding the latter category of problems, on the other hand, because finding the optimum solution becomes difficult due to solution space increasing explosively as the problem becomes larger, methods such as Genetic Algorithm (GA) and Simulated Annealing (SA) which are heuristic methods are used[1].

SA was presented by Kirkpatrick et al, and which algorithm imitates physical annealing. It is an algorithm that is effective for optimization problems, and especially for combinatorial optimization problems. In SA, objective functions that should be minimized are called energy, and it is very useful for several types of functions which are non-liner functions[2], [3], [4].

Though SA is an effective method for combinatorial optimization problems, it is also used in complex continuous optimization problems. And recently SA is being applied to problem solving from the viewpoint of energy minimization of the 3-dimensional structure of protein[5]. Even though optimization problem is continuous as described above, SA can be said to also be effective for complex continuous optimization problems with objective functions of many local optima as well.

The important parameters in SA are the neighborhood and the temperature. In combinatorial optimization problems, two adjoining elements are switched, etc., and a neighborhood is generated. Because the neighborhood structure can be determined uniquely, adjustment of the temperature schedule is extremely important.

Alternatively, when SA is applied to a continuous optimization problem, the neighborhood is related to the distance within the Euclid space, and it is possible to determine freely. Generally, when the neighborhood range is too small it tends to fall into local minima, and when the neighborhood is too large local search is insufficient and the solution accuracy becomes deficient. Due to this, when SA is applied to a continuous optimization problem, it is difficult to determine the appropriate neighborhood range.

In response to this, the research witch adjusts the neighborhood rance according to the landscape of the objective function adaptively has been done. Corana automated the neighborhood range adjustment according to a landscape using the acceptance ratio might be set to 0.5[6]. Also, the authors conceived a new neighborhood range design that can provide arbitrary acceptance ratios and proposed a simulated annealing with advanced adaptive neighborhood range (SA/AAN)[7], [8]. Because these methods adjust neighborhood range automatically, the tuning of these methods for every problem is unnecessary. However, these methods do require acceptance rate setting. In most problems a positive result with a value of 0.2 is obtained, but whether or not this value is constant for all problems is not clear.

Thus, even when using an adaptive method, a minimal parameter is necessary for automatic adjustment of the neighborhood range. The cause of this lies in the conventional method generates neighborhood range which determines the range within a certain distance from the current state, and creating the next state within that range interior. Due to this, when using the conventional method generates neighborhood range, the necessity of adjusting the neighborhood range occurs in every problem.

This research proposes the adaptive neighborhood range Multi-point Simulated Annealing with Adaptive Neighbor-

Keiko ANDO is with the Graduated School of Knowledge Engineering and Computer Science, Doshisha Univ. Kyoto, Japan ando@mikilab.doshisha.ac.jp

Mitsunori MIKI, and Tomoyuki HIROYASU are with the Department of Knowledge Engineering and Computer Science, Doshisha Univ, Kyoto, Japan mmiki@mail.doshisha.ac.jp, tomo@is.doshisha.ac.jp

hood (MSA/AN) as a method that does not require parameters for neighborhood generation. The proposed method does not adjust neighborhood range via parameters, but instead uses multiple search points and finds the range that automatically generates the neighborhood range by using that search point information.

Also, the proposed method is applied to typical mathematics functions minimizing problems, and the effectiveness is verified.

II. Multi- Point Stimulated Annealing with Adaptive Neighborhood (MSA/AN)

A. *Multi- Point Stimulated Annealing with Adaptive Neighborhood (MSA/AN) concept*

The neighborhood range definition is shown in Fig. 1. The contour in Fig. 1 shows the objective function contour and shows the generation of the next state from the current state within design space interior. The left figure (a) shows the conventional definition method, and the right figure (b) shows the proposed definition method.

Because the neighborhood range is related to the distance within Euclid space, in conventional SA the appropriate distance range interior centering on the current search point is determined as the neighborhood range, and the next state is generated within that neighborhood range. Due to this, the neighborhood range parameter is necessary. Also, the appropriate neighborhood range is dependant on the problem, and many preliminary experiments are necessary for finding this. In contrast, the preliminary experiment load is reduced when using an adaptive neighborhood mechanism, but new parameters become necessary[8].

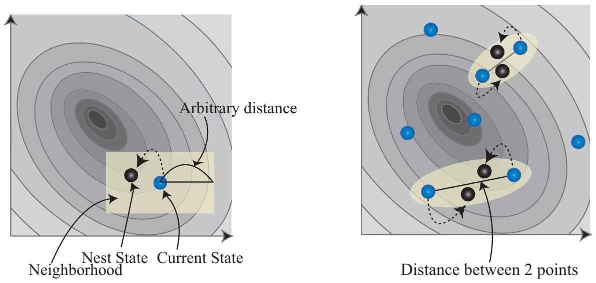
In order to overcome this problem, we propose herein a new method using a multi-point search SA as a process for making parameters unnecessary. We call this method Multi-point Simulated Annealing with Adaptive Neighborhood: MSA/AN.

The basic concept of this method is as follows. In summary, the method uses multiple search points, and then creates a probability distribution and generates the neighborhood range structure based on the location information of those search points. Then, following the generated neighborhood construction, the next state N point is generated. This eliminates the necessity for the neighborhood range parameter, and the appropriate neighborhood range can be considered automatically obtainable.

Also, in this method, when search points are scattered within the design space interior, the next state is generated in a wide range. And when the search points are clustered together, the next state is generated in a range near the search points.

B. *Multi- Point Stimulated Annealing with Adaptive Neighborhood (MSA/AN) algorithm*

By generating the neighborhood range based on search point location information, MSA/AN generates and adjusts the neighborhood range automatically. The MSA/AN algorithm is shown in Fig. 2. The process of MSA/AN is stated below. However, the selection points are considered to be 2 or 3 points.



(a) Conventional Neighborhood Range

(b) Proposal Method Neighborhood Range

Fig. 1. Neighborhood Range Definition

1) Multiple starting points are generated.
2) From the search points, 2 or 3 points are selected randomly.
3) Based on the 2 or 3 search points selected, a probability distribution neighborhood structure is generated.
4) Next state candidates are generated following the neighborhood structure.
   (Next state candidates are generated N points.)
5) Next state is selected from the next state candidates.
   (Based on a probability distribution, the next state is created within the neighborhood range, but in order to consider the landscape of objective function, multiple next state candidates are created and 2 states is selected. One is best state and another is a randomly selected state.)
6) Acceptance judging
7) Transition
8) Cooling on a fixed cycle

C. *About Acceptance Judging*

Whether or not a transition to the next state is accepted based on the difference ΔE (= E'- E) for energy E' of the next state x' and energy E of the current state x, as well as the temperature parameter T. In the standard SA, Metropolis standard[4] of (1) is applied. Temperature T is a parameter that exerts a crucial influence on the probability of transition to the direction of energy increase. When the temperature is high, the probability of transition to the direction of energy increase is high. Conversely, when the temperature is low, there is a tendency of a probability for transition towards the direction of energy decrease. However, regardless of the temperature, the probability of transition to the direction of energy increase never becomes 0.

$$\begin{cases} 1 & if\ \Delta E < 0 \\ \exp\left(-\frac{\Delta E}{T}\right) & otherwise \end{cases} \tag{1}$$

In the case of MSA/AN, because 2 search points are generated as the next state from 2 search points, there are

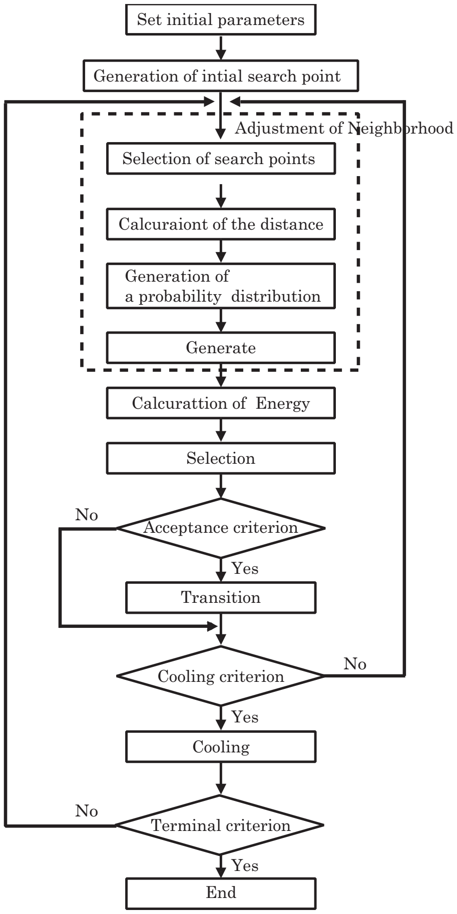


Fig. 2. MSA/AN algorithm

2 kinds of energy that should be compared to the next state energy. This results in the necessity of a new acceptance criterion from the conventional Metropolis criterion. Figure 3 shows a conceptual diagram of the proposed acceptance criterion. The vertical axis of Fig. 3 is the energy value, and the horizontal axis is pattern. The vertical axis E1 and E2 is the energy value of the current 2 search points. The higher energy value is E1. The 2 search points generated as a next state execute acceptance judging one by one via the following acceptance criterion. The energy value of the search points which execute the judging is E, and the difference between E1E2 and E is $\Delta E1$ and $\Delta E2$.

Acceptance criteria are classified into the following 3 patterns as shown in Fig. 3.

1) The energy value E has a lower energy value than E1 and E2.
2) E has an energy value that is lower than E1 but higher than E2.
3) E has an energy value that is higher than both E1 and E2.

In the case of pattern 1, it can be considered that new search point has been generated in an improvement direction more than the current 2 search points. This search point is accepted at acceptance probability 1.

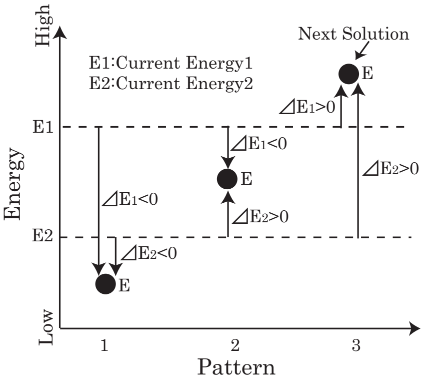


Fig. 3. MSA/AN Acceptance Judging

In the case of pattern 2, because in thinking from E1 new search point is generated in an improvement direction, though transfer at acceptance probability 1 is valid, in thinking from the energy E2 of the other current search point new search point is generated in an uphill transition. Due to this, the algebra average of the improvement direction acceptance probability 1 and the uphill transition acceptance probability found via Metropolis criteria is made the acceptance probability.

In the case of pattern 3, because new search point is generated in uphill transitions more than the current search points, the algebra average of the uphill transition acceptance probabilities of E1 and E2 is made the acceptance probability.

Equation (2) shows the probability equation used in acceptance judging that sums up pattern 1 to pattern 3.

$$\begin{cases} 1 & if\ \Delta E_1 < 0, \Delta E_2 < 0 \\ \left\{1+\exp\left(-\frac{\Delta E_2}{T}\right)\right\}/2 & if\ \Delta E_1 < 0, \Delta E_2 > 0 \\ \left\{\exp\left(-\frac{\Delta E_1}{T}\right)+\exp\left(-\frac{\Delta E_2}{T}\right)\right\}/2 & otherewise \\ (\Delta E_1 = E - E1) & \\ (\Delta E_2 = E - E2) & \end{cases} \qquad (2)$$

III. Proposal Method Effectiveness Verification

A. Optimization problems

To examine the search performance of the proposed method, 4 standard test functions are used. Those are the Rastrigin function[12] shown in (3), the Sphere function[13] shown in (4), the Rosenbrock function[12] shown in (5) and the Rotated Rastrigin function is obtained by rotating the Rastrigin function about the origin by 45 degrees.

The optimum solutions are located at the origin for the Rastrigin, Shere and Rotated Rastrigin functions, and the function values are 0, and the optimum solution is located at(1,...,1) for the Rosenbrock function, and its function value is also 0.

$$\begin{aligned} f_{\mathrm{Rast}}(\vec{x}) &= (N \times 10) + \left[\sum_{i=1}^{N} \left(x_i^2 - 10\cos(2\pi x_i)\right)\right] \\ domain &: -5.12 < x_i \leq 5.12, \\ optimum\ solution &: (x_1, x_N) = (0, \ldots, 0), \\ optimum\ value &: f = 0 \end{aligned} \tag{3}$$

$$\begin{aligned} f_{\mathrm{Sphere}}(\vec{x}) &= \sum_{i=1}^{N} x_i^2 \\ domain &: -5.12 < x_i \leq 5.12, \\ optimum\ solution &: (x_1, x_N) = (0, \ldots, 0), \\ optimum\ value &: f = 0 \end{aligned} \tag{4}$$

$$\begin{aligned} f_{\mathrm{Rosen}}(\vec{x}) &= \sum_{i=1}^{N} \left[100(x_1 - x_i^2)^2 + (x_i - 1)^2\right] \\ domain &: -2.048 < x_i \leq 2.048, \\ optimum\ solution &: (x_1, x_N) = (1, \ldots, 1), \\ optimum\ value &: f = 0 \end{aligned} \tag{5}$$

B. Parameter Setting

In the numerical experiment, uniform distribution and normal distribution are used as probability distributions for MSA/AN neighborhood range structure. In the case of uniform distribution, the method called BLX-α[11], used in Genetic Algorithm (GA) research, exists as a method for generating 2 next state candidates from 2 search points. That method is used herein.

Conversely, in the case wherein normal distribution is used for neighborhood range structure, the method called UNDX[10], which is a Genetic Algorithm intersection method, exists as a method for generating 2 next state candidates from 3 search points. Though there are many methods proposed for creating a normal distribution, because UNDX is a method of high performance, it is used herein.

It is thought that UNDX is effective in problems wherein dependency exists between design variables, and BLX-α is effective in problems wherein no dependency exists between design variables.

In order to indicate the effectiveness of the MSA/AN proposed in this paper, as a comparison method, PSA (Parallel SA), UNDX and BLX-α are used. PSA is method of SA, which is the most common method and its performance is high. UNDX and BLX-α are methods of GA, which is a heuristic method just as SA is. these mehods are typical, and show extremely high performance.

2-dimensions, 5-dimensions, 10-dimensions, and 20 dimensions of the Rastrigin function, Sphere function, Rosenbrock function, and Rotated Rastrigin function are used for test functions.

The patterns used for 4 test functions are indicated in Table I, II, and III.

The number of times of cooling for MSA/AN was set to 32, and the PSA parallel process number was also set at 32. Regarding PSA, 32 parallel processors are used, and the total annealing step was set to the same as MSA/AN and GA (UNDX and BLX-α) by setting the cooling cycle to 1/32 times.

Refer to reference text[8] for the detailed parameter setting method pertaining to MSA/AN and PSA, and to reference text[10], [11] for the detailed parameter setting method pertaining to GA (UNDX and BLX-α). Rand48 was used for random numbers. These random numbers are created by a 48 bit linear pseudorandom number generation function created by Martin Birgmeier et al. Refer to reference text[14] for the details of these random numbers.

TABLE I
PARAMETERS(MSA/AN)

Function	Rastrigin RotatedRastrigin	Sphere	Rosenbrock
Max temp.	200	1	1
Min temp.	50	0.00001	0.00001
Markov Length	2500	1000	2500
Dimension	2, 5, 10, 20		

TABLE II
PARAMETERS(PSA)

Function	Rastrigin RotatedRastrigin	Sphere	Rosenbrock
Max temp.	200	1	1
Min temp.	50	0.00001	0.00001
Markov Length	2500/32	1000/32	2500/32
Dimension	2, 5, 10, 20		
Neighborhood range	1	1	0.5
Number of processors	32		

TABLE III
PARAMETERS(GA(UNDX), GA(BLX-α))

Function	Rastrigin RotatedRastrigin	Sphere	Rosenbrock
Generation	2500	1000	2500
Number of Crossover	300	50	50
Dimension	2, 5, 10, 20		

IV. Results and Discussion

A. MSA/AN Performance

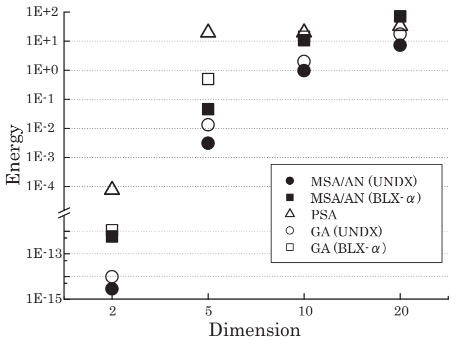


Fig. 4. Performance Comparison by Method (Rastrigin)

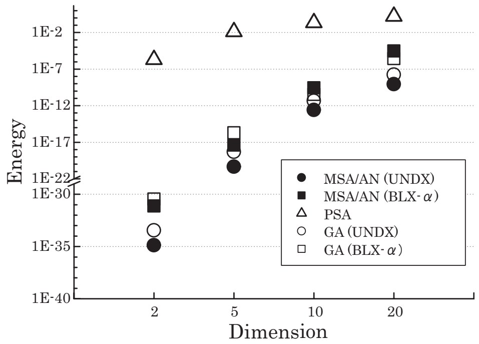


Fig. 5. Performance Comparison by Method (Sphere)

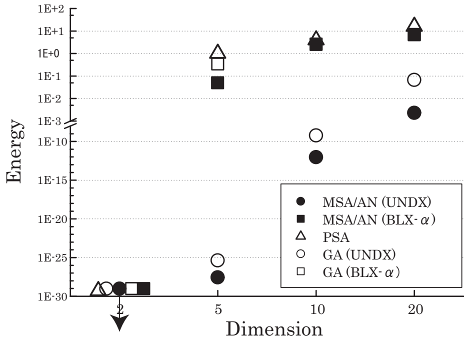


Fig. 6. Performance Comparison by Method (Rosenbrock)

Minimum energy value obtained using the Rastrigin function is indicated in Fig. 4, the result from using the Sphere function is indicated in Fig. 5, the result from using the Rosenbrock function is indicated in Fig. 6, and the minimum energy value obtained using the Rotated Rastrigin function is indicated in Fig. 7. These results are the median values of 30 trials. The reason for using median values is that multiple local minimums exist, and when the difference of their function values is great, comparing to a median value is sturdier as a parameter of location estimator than comparing to an average value. Energy value is indicated on the vertical axis and dimension number is indicated on the horizontal axis.

Also, the probability distribution type used in the proposed MSA/AN method is shown in parenthesis. By comparing PSA and MSA/AN, it is clear from Fig. 4, 5, 6 and 7 that the proposed method(MSA/AN) is shows an extremely high performance from every dimension.

Next, the case wherein UNDX distribution is used in MSA/AN is compared to the result of UNDX as a GA method. The results show that the Rastrigin function, Sphere function, Rosenbrock function, and Rotated Rastrigin function, along with MSA/AN(UNDX) have better results than GA(UNDX).

Also, comparing the case in which BLX-α distribution is used in MSA/AN (MSA/AN(BLX-α)), and the results of BLX-α as a GA method, it is clear that in all functions MSA/AN(BLX-α) shows results that are the same as or better than GA(BLX-α).

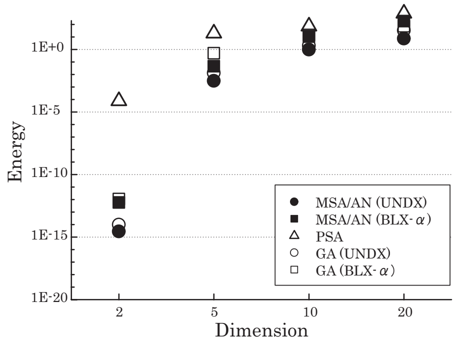


Fig. 7. Performance Comparison by Method (Rotated Rastrigin)

These results signify that MSA/AN shows results that are better than the typical high performance PSA. And because the results are better than GA using UNDX or BLX-α, it can be thought that this is an effective method.

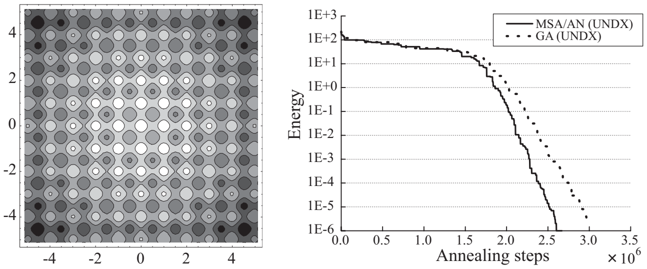
B. MSA/AN Effectiveness

Herein we shall consider the effectiveness of MSA/AN(UNDX) by comparing the energy history (search history) of the MSA/AN(UNDX) that displayed good results on Fig. 4, 5, 6 and 7 and that of GA(UNDX). A 20 dimensional Rastrigin function, Sphere function, Rosenbrock function, and Rotated Rastrigin function are used in the objective problem. The energy history obtained by applying the Rastrigin function is shown on Fig. 8(b), the results from applying the Sphere function are shown on Fig. 9(b), the results from applying the Rosenbrock function are shown on Fig. 10(b), and the results from applying the Rotated Rastrigin are shown on Fig. 11(b). Also, the respective contours are shown on Fig. 8(a), 9(a), 10(a), and 11(a). These results are the histories for 1 trial. The horizontal axis shows the annealing step number, and the vertical axis displays the energy value.

In Fig. 8(b), 9(b), 10(b), and 11(b), no difference is seen in the methods in the first stage. However, in the middle stage difference begins to occur between the methods, and it is clear that MSA/AN(UNDX) is obtaining the optimum solution faster than GA(UNDX). The reason for this can be thought to be the temperature that is SA characteristic, is involved.

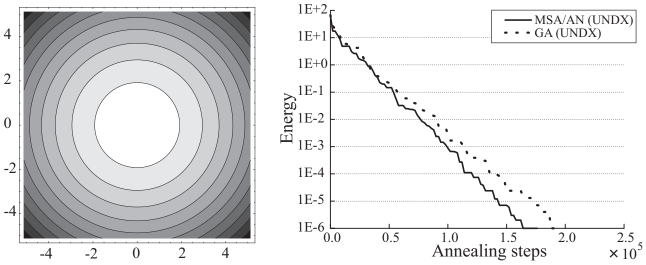
GA(UNDX) generates child individuals and executes roulette selection based on rank from the generated child individuals. Then, it proceeds with the search by returning the selected child individual to the population. With this method, when the selection method is determined, the probability that the energy of the selected child individual will become worse than the parental individual (probability that allows uphill) is fixed.

On the other hand, MSA/AN(UNDX) generates a next state, executes acceptance judging based on temperature, and proceeds with the search while allowing uphill transition. Because a high temperature exists in the first stage, the



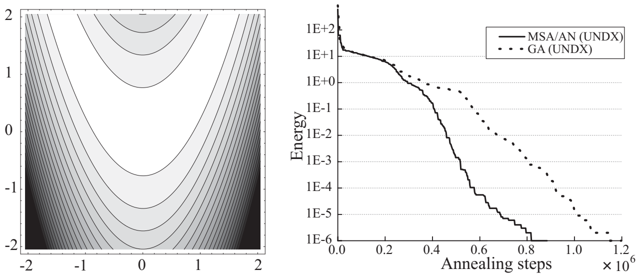
(a) Contour image (b) Search history

Fig. 8. Rastrigin function convergence curve



(a) Contour image (b) Search history

Fig. 9. Sphere function convergence curve

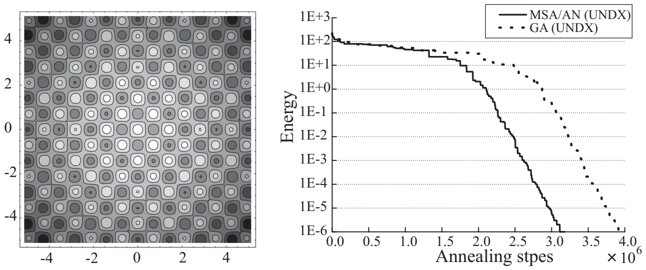


(a) Contour image (b) Search history

Fig. 10. Rosenbrock function convergence curve

probability of uphill transition is high. And at the end of the stage the low temperature results in a low probability of uphill transition. So the uphill transition fluctuates along with the search process.

Due to this, when a local minimum drop occurs in the first half of the stage, because MSA/AN(UNDX) allows uphill transition much more than GA(UNDX), it can be thought that escape from the local minimum is easily executed. On the other hand, regarding the last half of the stage, because MSA/AN(UNDX) does not easily allow an uphill transition, unnecessary uphill does not occur, and as a result it can be considered that an adequate local search is facilitated and accuracy improves.



(a) Contour image (b) Search history

Fig. 11. Rotated Rastrigin function convergence curve

V. Conclusion

When simulated annealing is applied to a continuous optimized problem, neighborhood range adjustment is necessary and indispensable. This research proposed a Multipoint Simulated Annealing with Adaptive Neighborhood (MSA/AN) possessing a new neighborhood range generation mechanism that makes unnecessary parameters which are necessary for neighborhood adjustment within conventional adaptive neighborhood range mechanisms. The proposed method is found to be very effective for solving continuous optimization problems by SA.

References

[1] Reeves, C.R. : Modern Heuristic, THE NIKKAN KOGYO SHIMBUN(1997).

[2] Hajime Kita. Simulated Annealing. Japan Society of Fuzzy Theory and Intelligent Informatics, (1997).

[3] Aarts, E. and Korst, J.: Simulated Annealing and Boltzmann Machines, john Wiley & Sons, (1989).

[4] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. and Teller, E.: Equation of State Calculation by Fast Computing Machines, Journ. of Chemical Physics, Vol. 21, pp. 1087-1092 (1953).

[5] Minoru Kanehisa : Invitation to genome information, KYORITSU SHUPPAN, (1996).

[6] Corana, A., Marchesi, M., Martini, C. and Ridella, S.: Minimizing Multimodal Functions of Continuous Variables with the ”Simulated Annealing” Algorithm, ACM Trans. on Mathematical Software, Vol. 13, No. 3, pp. 262-280 (1987).

[7] Mitsunori Miki, Tomoyasu Hiroyasu, Keiko Ono : Simulated Annealing with Advanced Adaptive Neighborhood, Information Processing Society of Japan, Vol.44, No.1, pp1-6 (2003).

[8] Mitsunori Miki, Tomoyasu Hiroyasu, Masayuki Kasai, Keiko Ono : Temperature Parallel Simulated Annealing with Adaptive Neighborhood, Information Processing Society of Japan, Vol.42, No.4, pp745-753(2003)

[9] Bruce E. Rosen, Ryohei Nakano : Simulated Annealing, Vol.9, NO.3, pp365-371 (1994).

[10] Ono, I. and Kobayashi, S. : A Real-coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distribution Crossover, The Japanese Society for Artificial Intelligence, Vol14-6,pp1146-1155 (1999)

[11] L. Eshelman. : The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination , Foundations of Genetic Algorithms, pp. 265-283, Morgan Kaufmann (1991)

[12] Whitley, D., Mathias, K., Rana, S. and Dzubera, J.: Evaluating Evolutionary Algorithms, Artificial Intelligence, Vol. 85, pp. 245-276 (1996).

[13] http://www.ft.utb.cz/people/zelinka/soma/func.html

[14] http://www.ics.uci.edu/eppstein/projects/pairs