Simulated Annealing using an Adaptive Search Vector

Mitsunori Miki*, Satoru Hiwa†, Tomoyuki Hiroyasu*
*Department of Engineering, Doshisha University, Kyoto, Japan
Email: mmiki@mail.doshisha.ac.jp, tomo@is.doshisha.ac.jp
†Graduate School of Engineering, Doshisha University, Kyoto, Japan
Email: shiwa@mikilab.doshisha.ac.jp

***Abstract*— It is reported that simulated annealing (SA), which changes one design variable at a time, is effective when applied to high-dimensional continuous optimization problems. However, if a correlation exists among the design variables, it is not efficient to search each dimension. In this paper, we propose SA with a mechanism to determine an appropriate search direction according to the landscape of the given problems. Its effectiveness is verified for high-dimensional problems with correlation among design variables.**

***Keywords*—simulated annealing, adaptive search vector, correlation among design variables.**

I. Introduction

Simulated annealing (SA)[2] is an optimization method that simulates the physical process of annealing and is very useful for solving optimization problems.[3][4] Although SA is particularly effective for solving combinatorial optimization problems it can also be applied to continuous optimization problems when the complexity of a given problem is very high.

When SA is applied to continuous optimization problems, the appropriate adjustment of the neighborhood range according to the landscape of the given problem is very important. Therefore, SA using an adaptive neighborhood mechanism that adaptively adjusts the neighborhood range has been proposed.[5][6]

However, the performance of SA decreases significantly with an increase in the dimensions of the problems. For high-dimensional problems, it is reported that SA, which changes one design variable at a time, is effective;[7] however, its effectiveness is lost if there exists a correlation among the design variables of a problem.

In high-dimensional variable space, the space within which the search point moves increases significantly and the number of search directions increases likewise; thus, it is difficult to find an appropriate direction. In this paper, we propose a new SA using an adaptive search vector (SA/ASV) that possesses a mechanism for determining the appropriate search directions. Further, its effectiveness is verified for high-dimensional continuous optimization problems whose design variables are correlated.

II. Simulated annealing

By simulating annealing — a process employed to obtain a perfect crystal by the gradual cooling of a melted solid — SA obtains the minimum value of energy. This energy is equivalent to the objective function value in conventional optimization problems.

The SA algorithm comprises three operations — generation, acceptance, and cooling. The generation operation changes the current solution x and generates the next solution x' by using a probability distribution. The acceptance operation decides whether the change is acceptable. This acceptance is determined by a difference $\triangle E(= E' - E)$ of the current energy $E = f(x)$ and energy of the next solution $E' = f(x')$ as well as the temperature parameter T. Metropolis et al.[8] introduced a simple algorithm (shown in (1)) to provide an efficient simulation.

$$P_{AC} = \begin{cases} 1, & \text{if } \triangle E < 0 \\ \exp\left(-\dfrac{\triangle E}{T}\right), & \text{otherwise} \end{cases} \tag{1}$$

That is, if $\triangle E < 0$, the change is accepted. Otherwise, the modification is accepted at a certain probability.

The cooling operation generates the temperature of the next state from the temperature of the current state. If the temperature parameter T is large, the probability of accepting the solution whose energy is larger than that of the previous solution increases, while the probability of accepting the solution with smaller energy decreases if T is low. Therefore, at the beginning of simulation, both the temperature and the acceptance levels need to be high. As the simulation proceeds and temperature decreases, a search point attains the global optimum solution.

III. SA for high-dimensional problems

A. The effects of the neighborhood range for high-dimensional problems

When SA is applied to optimization problems, the appropriate adjustment of the temperature schedule and the design of the neighborhood structure are very important. For continuous optimization problems, in particular, the neighborhood range determines the range within which the search point can move; therefore, the neighborhood range in SA significantly effects the accuracy of the solutions. For example, a large neighborhood range enables a global search, while a small neighborhood range enables a local search. Therefore, it is possible to improve the performance of SA by appropriately adjusting the neighborhood range according to the landscape of the given problems.

However, the performance of SA decreases significantly with an increase in the dimension of the problems. Figure 1 shows the effects of the neighborhood range on the energy of the optimum solutions for the rotated Rastrigin function. These effects are typically observed among mathematical test functions for continuous optimization problems. The dimensions of the problem are 2, 3, 5, and 10. These plots represent the median value of 30 trials. The rotated Rastrigin function is obtained by rotating the design variables by $\pi/12$ in the standard Rastrigin function shown below. The rotated Rastrigin function is a multipeak function with correlation among its design variables. Figures 2 and 3 show the landscape and contour plots of the rotated Rastrigin function.

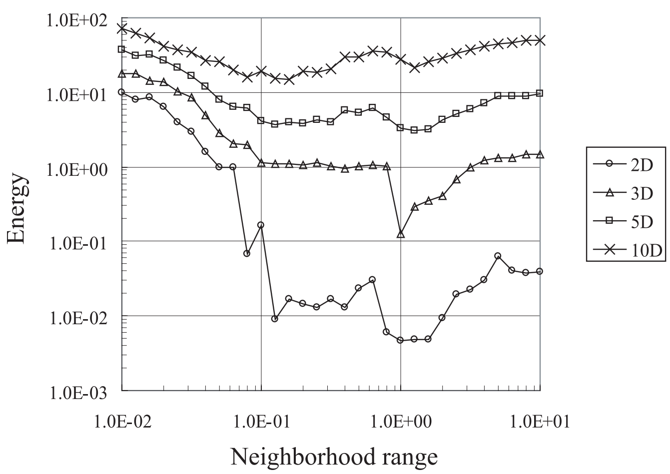

Fig. 1. Effects of neighborhood range on the energy of optimum solutions

$$F_{Rastrigin}(x) = 10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i)\right) \qquad (2)$$

$$(-5.12 \leq x_i < 5.12)$$

$$min(F_{Rastrigin}(x)) = F(0, 0, \ldots, 0) = 0$$

The result indicates that the neighborhood range significantly affects the accuracy of the solutions obtained in two- or three-dimensional variable spaces. However, this effect is not prominent in five- or ten-dimensional variable spaces.

The result is based on the size of the neighborhood space. Figure 4 illustrates the search directions in one- and two-dimensional variable spaces. In a one-dimensional variable space, there are two search directions (positive or negative), and either of them is appropriate. If the probability of transition to an appropriate direction is 0.5, the probability in a two-dimensional variable space becomes 0.5^2 because it is necessary to transit with probabilities of 0.5 to an appropriate direction at each dimension. Therefore, in an N-dimensional variable space, the probability of transition to an appropriate direction is 0.5^N. In this case, the probability in a ten-dimensional variable space becomes $0.5^{10} = 0.0009765$, and is considerably lower than that in two- or three-dimensional variable spaces.

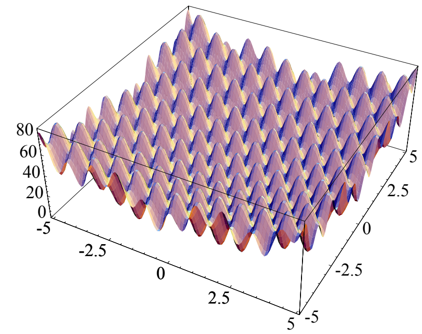

Fig. 2. Landscape of the rotated Rastrigin function

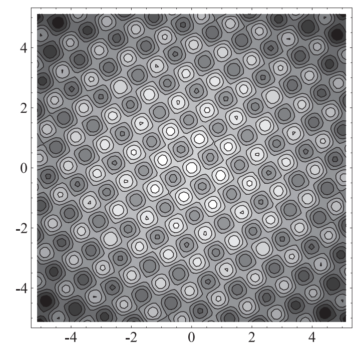

Fig. 3. Contour plot of rotated Rastrigin function

From the abovementioned discussion, the probability of transition to an appropriate direction decreases significantly in high-dimensional problems. Therefore, it should be noted that improving the accuracy of solutions by adjusting the neighborhood range is difficult in high-dimensional problems.

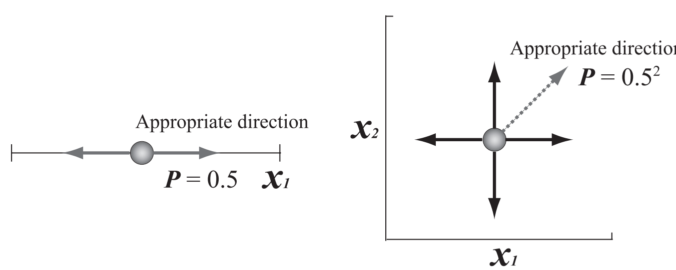

Fig. 4. Appropriate search directions and the probabilities of transition

B. SA that changes one design variable at a time

For high-dimensional problems, SA that changes one design variable at a time (it is referred to as dimensional-split SA in this paper) is effective.[7] The dimensional-split SA (DSA) exhibits very high accuracy for solutions of problems without correlation among their design variables. The global optimal solution for such problems can be obtained by combining the best solution in each dimension.

On the other hand, Fig. 5 represents the low efficiency of DSA for a problem with correlation among its design variables. For such problems, it is not effective to change one design variable at a time. Therefore, for high-dimensional problems, which have some correlation among their design variables, it is difficult to improve the performance of SA by DSA.

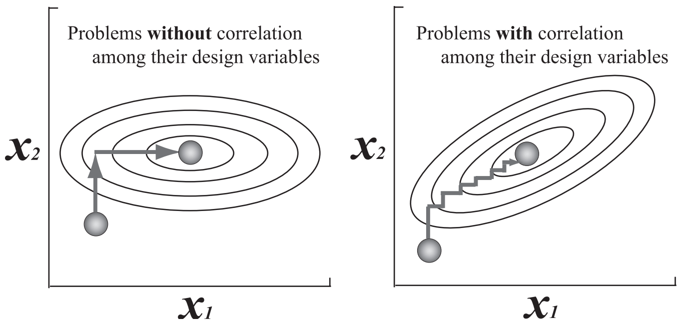

Fig. 5. Search using DSA

IV. SA using an Adaptive Search Vector

A. Appropriate search directions

For effective search in high-dimensional problems with correlation among their design variables (illustrated in Fig. 6), it is necessary to determine the appropriate search directions at each search point. Therefore, we consider that an effective search can be performed by adaptively determining the appropriate search directions in SA. In this paper, we propose SA that possesses a mechanism for determining the search directions; we refer to this as SA using an adaptive search vector (SA/ASV).

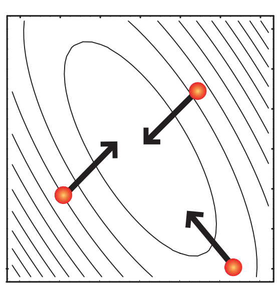

Fig. 6. Appropriate search directions

B. SA/ASV algorithm

The SA/ASV algorithm is based on Powell's method. In SA/ASV, the appropriate search direction is based on the solution obtained by DSA. Figure 7 illustrates the principle of SA/ASV search. The detailed procedure of SA/ASV is shown below.

1) Generate initial point $\vec{x}_s = (x_{s_1}, x_{s_2}, \cdots, x_{s_d})$, where d is the dimension of a given problem

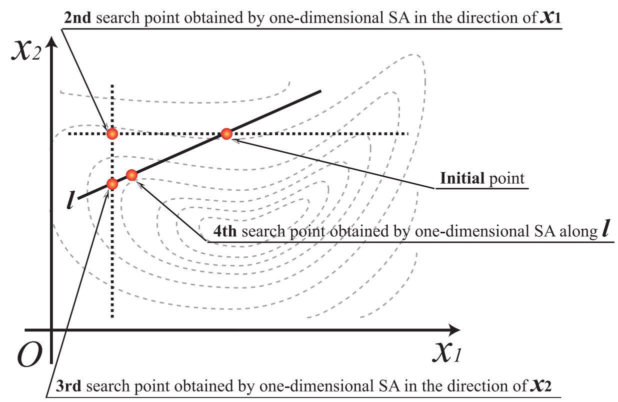

Fig. 7. Principle of SA/ASV

2) Execute K iterations of one-dimensional SA in the direction of x_i
3) Move the search point to the best solution in search at step 2
4) Execute step 2 and 3 in each dimension.
5) Set $\vec{u} = \vec{x}_e - \vec{x}_s$, where $\vec{x}_e$ is the best solution of search in the previous dimension
6) If $|\vec{u}| < \epsilon$, execute K iterations of one-dimensional SA along the direction determined by $\vec{u}$, otherwise execute K iterations of standard SA
7) Set $\vec{x}_s$ to the best solution in search at step 6
8) Cooling
9) Repeat steps 2 through 8

The search along the direction determined by $\vec{u}$ is executed based on (3):

$$x^{(k+1)} = x^{(k)} + t \cdot \vec{u} \tag{3}$$

where t is a random number in the range $[-D, D]$ and D is the neighborhood range. We defined $\epsilon = 10^{-2}$ in the rule in step 6. In step 2, the selection of the dimension to be searched is in a random order.

It is important to note at this point that the proposed algorithm differs significantly from gradient descent methods such as quasi-Newtonian methods. If the gradient descent methods are applied to line-search, the proposed algorithm will be trapped to local optimums in multipeak functions.

V. Numerical Experiments

In order to verify the effectiveness of the proposed method — SA/ASV — SA and DSA are applied to solve two mathematical test functions for continuous optimization problems — the rotated Rastrigin function and Griewank function. These functions are multipeak functions with correlation among their design variables. These functions are both five- and ten-dimensional.

Figures 8 and 9 show the landscape and contour plot of Griewank function. Although this function appears to have only one global peak, it has many local peaks.

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

$$(-512 \le x_i < 512)$$

$$min(F_{Griewank}(x)) = F(0, 0, \ldots, 0) = 0$$

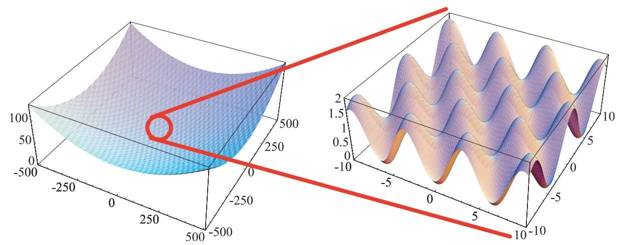

Fig. 8. Landscape of the Griewank function

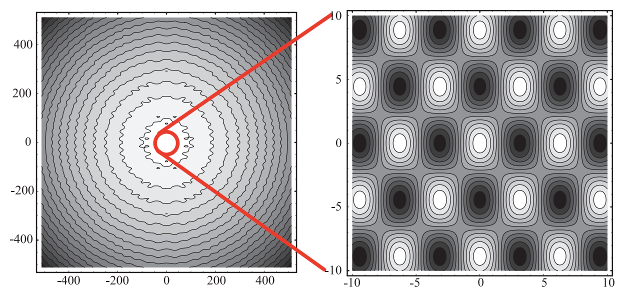

Fig. 9. Contour plot of the Griewank function

The SA/ASV parameters are shown in Tables I and II. The parameters in standard SA and DSA are almost the same as in SA/ASV, but the neighborhood range in standard SA and DSA for the Griewank function is different from that for SA/ASV. The neighborhood range in standard SA and DSA for the Griewank function is listed in Table III. The parameters shown are obtained through preliminary experiments.

The distribution of optimum solutions for 50 trials for the five- and ten-dimensional rotated Rastrigin function are shown in Figs. 10 and 11. The trial number is plotted along the horizontal axis and the converged energy in each trial is plotted in the ascending order.

From the result, it is observed that the proposed method — SA/ASV — provides good solutions as compared to the conventional method.

Futhermore, Figs. 12 and 13 show the result for the Griewank function. From these figures, SA/ASV produces the best result as compared to the other methods.

TABLE I

SA/ASV parameters for the rotated Rastrigin function

Total steps	192000(5D), 352000(10D)
Neighborhood range	1.0
Max. temperature	10.0
Min. temperature	0.01
K (one-dimensional search)	1000
Cooling steps	32

TABLE II

SA/ASV parameters for the Griewank function

Total steps	192000(5D), 352000(10D)
Neighborhood range	8.0
Max. temperature	20.0
Min. temperature	0.001
K (one-dimensional search)	1000
Cooling steps	32

TABLE III

Neighborhood ranges for standard SA and DSA for the Griewank function

	Standard SA	DSA
5D	8.0 (same as in SA/ASV)	16.0
10D	1.0	50.0

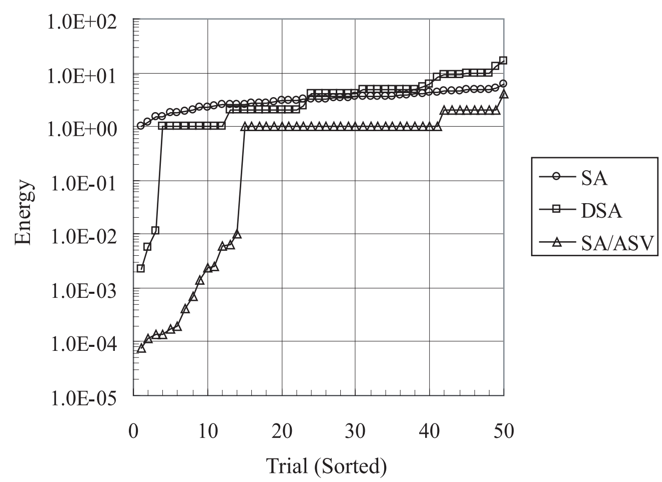

Fig. 10. Distribution of optimum solutions for the five-dimensional rotated Rastrigin function

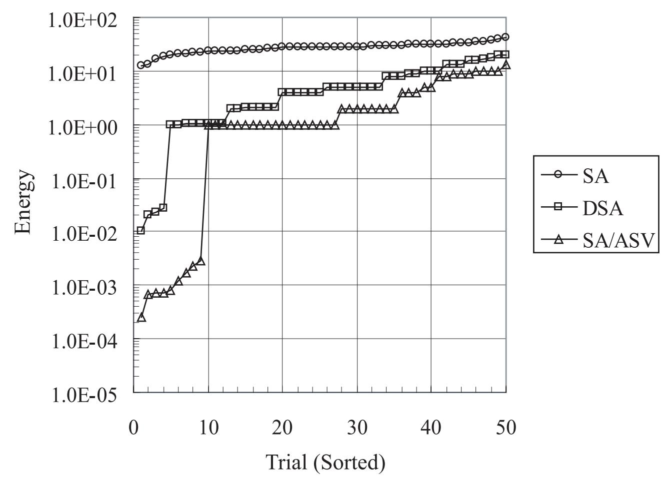

Fig. 11. Distribution of optimum solutions for the ten-dimensional rotated Rastrigin function

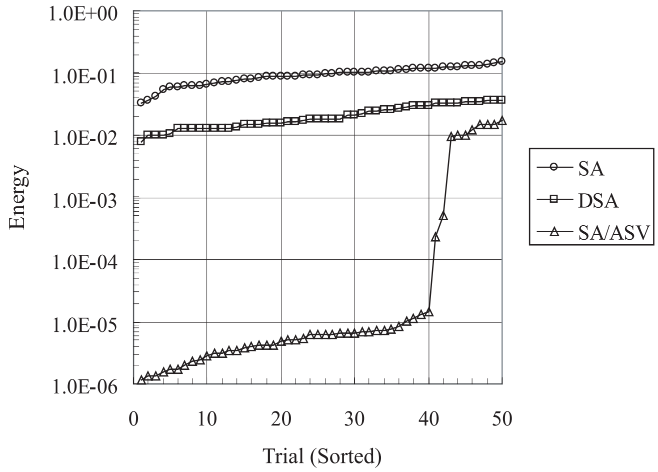

Fig. 12. Distribution of optimum solutions for the five-dimensional Griewank function

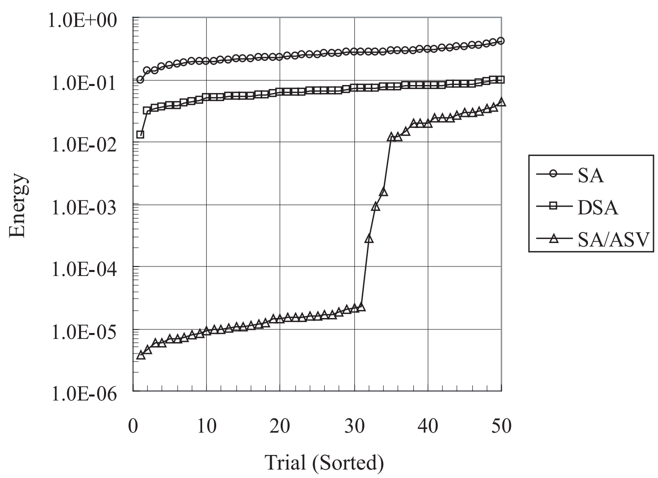

Fig. 13. Distribution of optimum solutions for the ten-dimensional Griewank function

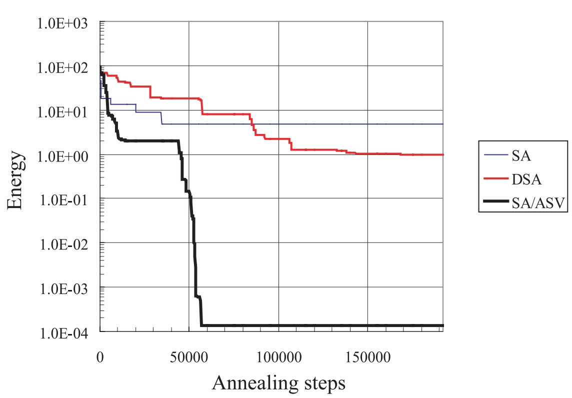

Fig. 14. Typical example of the history of the best energy for the five-dimensional rotated Rastrigin function

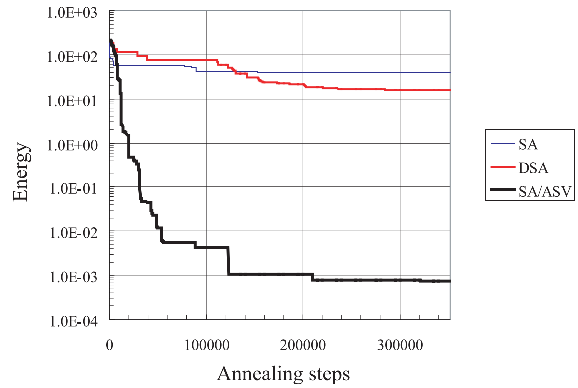

Fig. 15. Typical example of the history of the best energy for the ten-dimensional rotated Rastrigin function

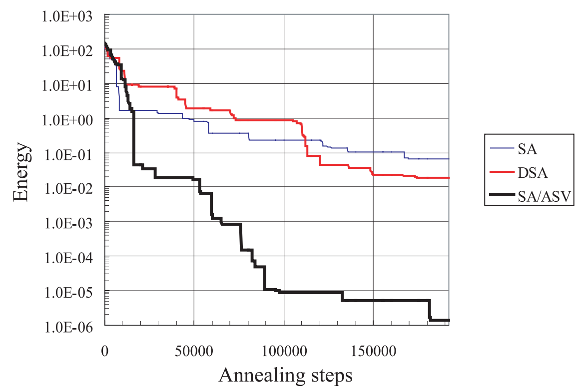

Fig. 16. Typical example of the history of the best energy for the five-dimensional Griewank function

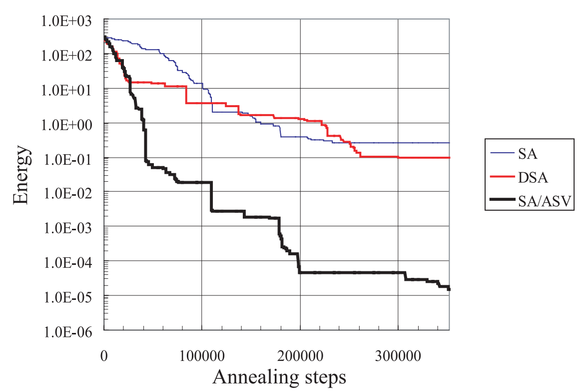

Fig. 17. Typical example of the history of the best energy for the ten-dimensional Griewank function

Furthermore, a typical example of the history of the best energy in each method is shown in Figs. 14 and 15 for the rotated Rastrigin function and Figs. 16 and 17 for the Griewank function.

From the figures, for all the functions, it is observed that SA/ASV can obtain a good solution at an early stage of a search as compared with standard SA and DSA.

Therefore, the proposed method — SA/ASV — is very effective for high-dimensional multipeak problems with correlation among their design variables.

VI. Conclusion

Conventional SA exhibits a poor performance for high-dimensional optimization problems with correlation among their design variables. In this paper, simulated annealing using an adaptive search vector (SA/ASV) is proposed. The proposed method can determine the appropriate search direction according to the landscape of the given problem. SA/ASV is applied to the rotated Rastrigin function and the Griewank function, which are multipeak functions that have correlation among their design variables. Further, it is found that the proposed method is very effective as compared to conventional methods.

References

[1] Kirkpatrick, S., Gelett Jr. C. D., and Vecchi, M. P. Optimization by Simulated Annealing, *Science*, Vol.220, No.4598, pp 671-680, 1983.

[2] Ingber, L. Genetic algorithms and very fast simulated reannealing, *A Comparison, Mathematical and Computer Modeling*, Vol.16, No.11, pp 87-100, 1992.

[3] Corana, A., Marchesi, M., Martini, C., 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.

[4] Mitsunori Miki, Tomoyuki Hiroyasu, Keiko Ono. Simulated Annealing with Advanced Adaptive Neighborhood, *Journal of Information Processing Society of Japan*, Vol.44, No.1, pp 1-6, 2003.

[5] Mitsunori Miki, Tomoyuki Hiroyasu, Toshihiko Fushimi. Parallel Simulated Annealing with Adaptive Neighborhood Determined by GA, *Proceeding of the 2003 IEEE International Conference on Systems, Man & Cybernetics*, pp 26-31, 2003.

[6] Tomoyuki Hiroyasu, Mitsunori Miki, Maki Ogura, and Yuko Okamoto. Examination of Parallel Simulated Annealing using Genetic Crossover, *Journal of Information Processing Society of Japan: Transactions on Mathematical Modeling and its Applications*, Vol.43, No. SIG7(TOM6), pp 70-79, 2002.

[7] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of State Calculations by Fast Computing Machines, *The Jurnal of Chemical Physics*, Vol.21, No.6, pp 1087-1092, 1953.

[8] M.J.D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives, *Computer Journal*, Vol.7, pp 155-162, 1964.