

Sequential Approximate Optimization using Radial Basis Function network for engineering optimization

メタデータ	言語: eng 出版者: 公開日: 2017-10-03 キーワード (Ja): キーワード (En): 作成者: メールアドレス: 所属:
URL	https://doi.org/10.24517/00008486

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 International License.



Sequential Approximate Optimization Using Radial Basis Function Network for Engineering Optimization

Satoshi Kitayama¹, Masao Arakawa², Koetsu Yamazaki³

¹ *Kanazawa University, Kakuma-machi, Kanazawa, 920-1192, Japan*

+81-76-234-4758

kitagon@t.kanazawa-u.ac.jp (Corresponding Author)

² *Kagawa University, Hayashi-chi, Takamatsu, Kagawa, 761-0396, Japan*

+81-87-864-2223

arakawa@eng.kagawa-u.ac.jp

³ *Kanazawa University, Kakuma-machi, Kanazawa, 920-1192, Japan*

+81-76-234-4666

yamazaki@t.kanazawa-u.ac.jp

Abstract

This paper presents a Sequential Approximate Optimization (SAO) procedure that uses the Radial Basis Function (RBF) network. If the objective and constraints are not known explicitly but can be evaluated through a computationally intensive numerical simulation, the response surface, which is often called meta-modeling, is an attractive method for finding an approximate global minimum with a small number of function evaluations. An RBF network is used to construct the response surface. The Gaussian function is employed as the basis function in this paper. In order to obtain the response surface with good approximation, the width of this Gaussian function should be adjusted. Therefore, we first examine the width. Through this examination, some sufficient conditions are introduced. Then, a simple method to determine the width of the Gaussian function is proposed. In addition, a new technique called the adaptive scaling technique is also proposed. The sufficient conditions for the width are satisfied by introducing this scaling technique. Second, the SAO algorithm is developed. The optimum of the response surface is taken as a new sampling point for local approximation. In addition, it is necessary to add new sampling points in the sparse region for global approximation. Thus, an important issue for SAO is to determine the sparse region among the sampling points. To achieve this, a new function called the density function is constructed using the RBF network. The global minimum of the density function is taken as the new sampling point. Through the sampling strategy proposed in this paper, the approximate global minimum can be found with a small number of function evaluations. Through numerical examples, the validities of the width and sampling strategy are examined in this paper.

Keywords: Response Surface, Sequential Approximate Optimization, RBF network, Density Function, Engineering Optimization

1. Introduction

In recent years, many commercial software programs for design optimization have been widely utilized in a variety of industries. Recently, some population-based global optimization techniques, such as the Genetic Algorithm (GA), the Particle Swarm Optimization (PSO), and the Differential Evolution (DE), have been developed in comparison with classical mathematical programming. These global optimization techniques have been applied to practical design optimization. In addition, these methods are applicable to the multi-objective optimization problems. In general, the population-based optimization techniques require a large number of function evaluations to find the global minimum and a set of pareto-optimal solutions. This makes the direct application of these optimization techniques to practical design optimization problems difficult in some cases due to the time-consuming. Since classical mathematical programming requires the sensitivity of the objective and constraints, it is not applicable to non-differentiable problems. In addition, function evaluations for calculating the sensitivity and determining the step-size are required. Nowadays, the time made available to develop new products is continuously being shortened, making it preferable to reduce the computing-time required for optimization. This implies that one of the most important aspects is reducing the function evaluations in practical design optimization. It is important to find the global minimum with high accuracy using global optimization techniques, and these global optimization techniques generally require a large number of function evaluations. However, it is also important to find an approximate global minimum for a design problem with a small number of function evaluations even when the objective and constraints are not known explicitly.

If the objective and constraints are not known explicitly but can be evaluated through computationally intensive numerical simulation, the response surface, which is called meta-modeling, is an attractive method for finding an approximate global minimum with a small number of function evaluations [1]. The Design of Experiment (DOE) is one of the most popular response surface methods [2]. The general and classical response surface procedure is briefly summarized as follows:

(STEP1) First, numerous sampling points in the design variable space are set. The orthogonal array or Latin Hypercube Design (LHD) is often used to determine these sampling points.

(STEP2) The objective and constraints are evaluated at these sampling points. Thus, the number of sampling points is equal to the function evaluations.

(STEP3) Then, a response surface which approximates the objective and constraints is constructed. Quadratic polynomials, the Kriging, and the Radial Basis Function (RBF) network are used to construct the response surface.

(STEP4) Finally, the approximate optimum can be obtained by optimizing the response surface. The optimum of the response surface is taken as the approximate optimum of the original design optimization problem

It is understood that the response surface is one of the approximation techniques. It is clear from the general flow described above that the number of function evaluations is drastically reduced by using the response surface approach. Using the quadratic polynomials as the response surface, it is possible to approximate the original function globally. It is possible to use the Kriging [3,4,5] and the RBF Network [6,7,8,9,10,11] to approximate the original function locally and globally, because these two methods utilize the Gaussian function as the basis function. Thus, the response surface using the Kriging and RBF Network is expressed by the linear combination of the weight and Gaussian function. In addition, the global and local approximations by the Kriging and the RBF Network imply that the response surface by these methods will be a multi-modal function. However, an appropriate parameter should be adjusted in order to approximate the original function locally and globally. Therefore, one key for a good approximation is to determine the parameter appropriately. Adjusting this parameter appropriately will allow the global minimum to be found with high accuracy. This parameter is the width of the Gaussian function [9,12]. If this width is small, the response surface will become peaky. Otherwise, the response surface will become smooth. The effect of the width is shown in Figs.1 (a) and (b). In Fig.1, the black dots represent the sampling points, the dashed line represents the Gaussian function, and the bold line denotes the response surface. The following weights are assigned to the sampling points: $w_1 = 0.5$ at $x = 1$, $w_2 = 1.7$ at $x = 3$, and $w_3 = 1.3$ at $x = 5$. The difference between Fig.1(a) and Fig.1 (b) is the value of the basis function width. The widths in Figs.1(a) and (b) are set to 0.5

and 1.0, respectively. It is clear from Fig.1 that the determination of the width plays an important role.

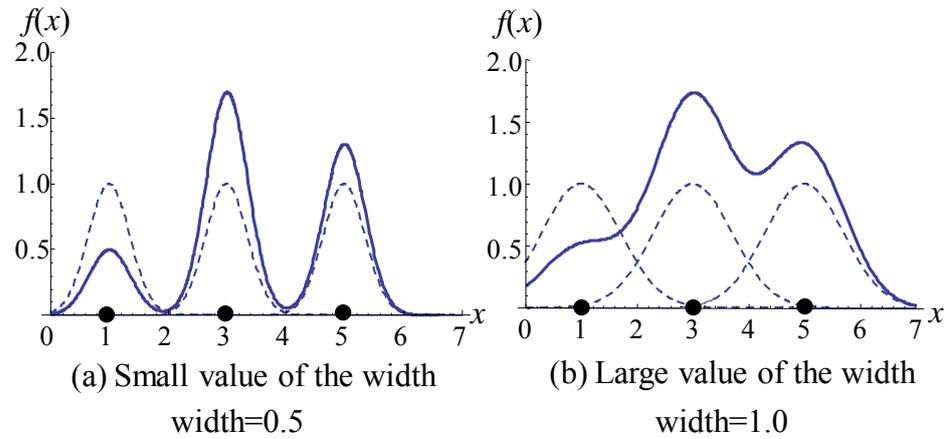


Fig.1 Effect of width in the Gaussian function

In recent years, the Sequential Approximate Optimization (SAO) has been widely studied [13,14,15,16,17,18,19], compared with the classical response surface approach described above. The general procedure for SAO is shown in Fig.2.

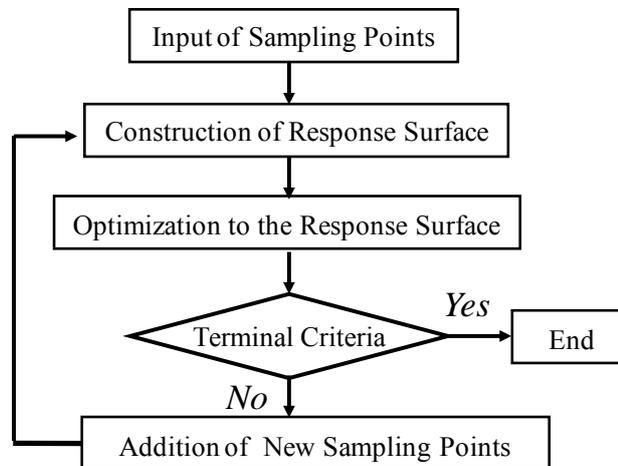


Fig.2 General procedure of SAO

In SAO, the response surface is constructed repeatedly by adding new sampling points, until the terminal criterion determined by the decision-maker is satisfied. In comparison with the classical response surface approach described above, it is expected that an approximate global minimum with high accuracy can be obtained through the addition of the new sampling points. In order to obtain an approximate global minimum with high accuracy, it has been reported that the most important requirement is simultaneously adding the new sampling points

around (1) the optimum of the response surface and (2) the sparse region in the design variable space [3,9,13,14,15,16,17,18,19,20,21,22]. Let us consider the first objective, which is to add the optimum of the response surface as the new sampling point. This will lead to a local approximation with high accuracy. The zooming method belongs to this category [23]. However, only the successive additions of the optimum of the response surface may result in finding the local minimum. Then, the second objective, which is to add a sampling point in the sparse region, plays an important role. The addition of a new sampling point in the sparse region will lead to the global approximation. By this addition, it is possible to avoid falling into the local minimum. Thus, global and local approximations will be achieved simultaneously through the above sequential sampling strategy. In this sequential sampling strategy, it is important to find the sparse region in the design variable space. This paper will roughly belong to Ref.[3,20,21]. In these references, the expected improvement (EI) algorithm is employed to find the sparse region. In the EI algorithm, the region with high uncertainty corresponds to the sparse region. By adding the new sampling points to the regions with high uncertainty, a global approximation can be achieved. However, the Gaussian function is also employed in the EI algorithm. In order to find the sparse region with the EI algorithm, the parameter in the Gaussian function should be adjusted. Therefore, the common subject is the determination of the width in the Gaussian function with a simple manner.

In this paper, we use the RBF network to construct the response surface, in which the Gaussian function is employed as the basis function. In particular, we will consider the determination of the width and the exploration of the sparse region in the design variable space. First, we discuss the width of the Gaussian function, which affects the accuracy of the response surface. Two equations for determining the width have been proposed [9,12]. Among these, the equation proposed by Nakayama [9] is effective through the author's numerical experiences in the case of one or two design variables. Thus, a good approximation can be achieved by using the equation proposed by Nakayama in the case of one or two design variables. However, it may be impossible to find an approximate global minimum with high accuracy in a case of involving more than three design variables. Therefore, a new equation for determining the width is necessary. By examining the equation proposed by Nakayama, some sufficient

conditions are introduced. Then, a new equation for determining the width is proposed. In addition, a new technique called the adaptive scaling technique is also proposed in this paper. Second, an effective method for determining the sparse region in the design variable space is considered. In Ref.[9], a simple method for determining the sparse region was proposed. However, this method basically depends on the randomness. Thus, a deterministic method for finding the sparse region is preferable from the viewpoint of efficiency.

The remainder of this paper is organized as follows: In section 2, the RBF network is described briefly. In this section, the width proposed by Nakayama is also analyzed, and some sufficient conditions for a good approximation are determined. Then, a new equation for the width is proposed. In addition, the adaptive scaling technique is also described. In section 3, the new function to find the sparse region in the design variable space, which is called the density function, is introduced, and the details of an SAO algorithm that uses the density function are shown in section 4. The density function utilizes the RBF network, making its construction easy. In section 5, some benchmark problems are discussed in order to examine the proposed SAO algorithm.

2. Radial Basis Function Network

2.1 Learning of RBF network

An RBF network is a three-layer feed-forward network. The output of the network $f_a(\mathbf{x})$, which corresponds to the response surface, is given by

$$f_a(\mathbf{x}) = \sum_{i=1}^m w_i h_i(\mathbf{x}) \quad (1)$$

where m represents the number of sampling points, $h_i(\mathbf{x})$ is the i -th basis function, and w_i denotes the weight of the i -th basis function. In this paper, the following Gaussian function is used as the basis function.

$$h_i(\mathbf{x}) = \exp\left(-\frac{(\mathbf{x} - \mathbf{x}_i)^T (\mathbf{x} - \mathbf{x}_i)}{r_i^2}\right) \quad (2)$$

In Eq(2), \mathbf{x}_i represents the i -th sampling point, and r_i is the width of the i -th basis function. The response y_i is calculated at sampling point \mathbf{x}_i . The learning of the RBF network is usually accomplished by solving

$$E = \sum_{i=1}^m (y_i - f_a(\mathbf{x}_i))^2 + \sum_{i=1}^m \lambda_i w_i^2 \rightarrow \min \quad (3)$$

where the second term is introduced for the purpose of regularization. It is recommended that λ_i in Eq.(3) have a sufficiently small value (e.g. $\lambda_i = 1.0 \times 10^{-3}$). Thus, the learning of the RBF network is equivalent to finding the weight vector \mathbf{w} [24]. The necessary condition of Eq.(3) leads to the following equation:

$$\mathbf{w} = (\mathbf{H}^T \mathbf{H} + \mathbf{A})^{-1} \mathbf{H}^T \mathbf{y} \quad (4)$$

where \mathbf{H} , \mathbf{A} , and \mathbf{y} are given as follows:

$$\mathbf{H} = \begin{bmatrix} h_1(\mathbf{x}_1) & h_2(\mathbf{x}_1) & \cdots & h_m(\mathbf{x}_1) \\ h_1(\mathbf{x}_2) & h_2(\mathbf{x}_2) & \cdots & h_m(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ h_1(\mathbf{x}_m) & h_2(\mathbf{x}_m) & \cdots & h_m(\mathbf{x}_m) \end{bmatrix} \quad (5)$$

$$\mathbf{A} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \lambda_m \end{bmatrix} \quad (6)$$

$$\mathbf{y} = (y_1, y_2, \dots, y_m)^T \quad (7)$$

It is clear from Eq.(6) that the learning of the RBF network is equivalent to the matrix inversion $(\mathbf{H}^T \mathbf{H} + \mathbf{A})^{-1}$. In the SAO, the new sampling points are added. Using the RBF network, it is easy to calculate the weight vector \mathbf{w} , because the additional learning is reduced to the incremental calculation of the matrix inversion. The detailed procedure is found in Ref.[24].

2.2 Width of Basis Function

Determining the width of the basis function is the key factor for good approximation. The optimization with respect to width may be valid. However, the increment of the sampling points will cause some difficulties in optimizing the width, such as the local minimum. Thus, it is preferable to determine the width with a simple method. To determine the width easily, the following equation was proposed by Nakayama [9]:

$$r = \frac{d_{\max}}{\sqrt[n]{nm}} \quad (8)$$

where d_{\max} denotes the maximum distance among the sampling points. n denotes the number of design variables, and m is the number of sampling points. Eq.(8) is

applied to all basis functions. Thus, $r_1 = r_2 = \dots = r_m = r$. Eq.(8) is considered to be the generalization of the equation proposed by Haykin [13].

Suppose that all of the design variables are equally scaled. This scaling technique, which is called the adaptive scaling technique, will be described in section 2.3. Let us consider the K -level full factorial design, in which the regular interval is given by Δd . In this case, d_{\max} is given by

$$d_{\max} = \sqrt{n}(K-1)\Delta d \quad (9)$$

The black dots in Fig.3 show the sampling points with two design variables.

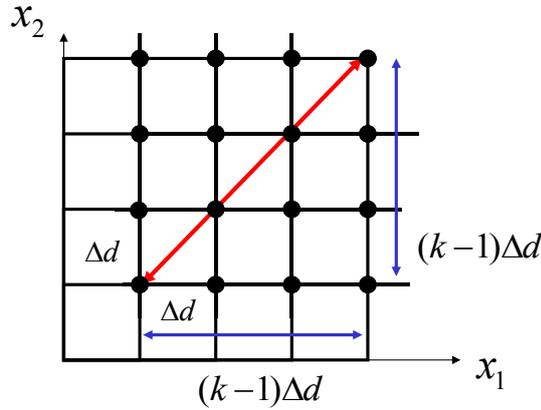


Fig.3 Sampling points with two design variables

In the case of n design variables, the number of sampling points, m , is simply calculated as follows:

$$m = K^n \quad (10)$$

Eqs.(9) and (10) are substituted into Eq.(8). We solve Eq.(8) with respect to $r/\Delta d$, and then we can finally obtain the following equation:

$$\frac{r}{\Delta d} = n \frac{n-2}{2^n} \left(1 - \frac{1}{K}\right) \quad (11)$$

In Eq.(11), $K \rightarrow \infty$ is considered. This implies an ideal distribution of the sampling points in the design variable space. Table 1 shows the convergence at $K \rightarrow \infty$.

Table 1 Convergence of $r/\Delta d$ at $K \rightarrow \infty$

The number of design variables n	$r/\Delta d$
1	1.000
2	1.000
3	1.201
4	1.414
5	1.621
6	1.817
7	2.003
8	2.181

It is clear from Table 1 that the uniform convergence of $r/\Delta d$ can be achieved in the cases of $n=1$ and $n=2$. However, $r/\Delta d$ does not converge uniformly in the case of $n \geq 3$. Therefore, $r/\Delta d \rightarrow 1$ cannot be achieved at $K \rightarrow \infty$. It is assumed that the key factor for a good approximation is the uniform convergence, which is $r/\Delta d \rightarrow 1$ at $K \rightarrow \infty$. Then, on the basis of Eq.(8), some sufficient conditions for the width for a good approximation by the RBF network are summarized as follows:

(W1) It is preferable to consider the number of design variables, n .

(W2) It is also preferable to consider the number of sampling points, m .

(W3) It is preferable to consider the maximum distance among the sampling points, d_{\max} .

(W4) It is preferable to consider the uniform convergence of $r/\Delta d$ through the increment of the number of design variables. ($r/\Delta d \rightarrow 1$ at $K \rightarrow \infty$)

In order to satisfy the above sufficient conditions, the following equation for the width may be valid:

$$r_1 = r_2 = \dots = r_m = \frac{d_{\max}}{\sqrt{n} \sqrt[m]{m}} \quad (12)$$

Since Eq.(12) satisfies the above sufficient conditions at $K \rightarrow \infty$, a good approximation can be expected. However, Eq.(12) does not consider the sparseness and density of the sampling points. In addition, it is clear from Eq.(10) that numerous sampling points are required for a good approximation, using Eq.(12). Then, the following equation considering the sparseness and density of the sampling points is proposed in this paper.

$$r_i = \frac{d_{i,\max}}{\sqrt{n} \sqrt[m-1]{m-1}} \quad i = 1, 2, \dots, m \quad (13)$$

where $d_{i,\max}$ denotes the maximum distance from the i -th sampling point. Eq.(13) is applied to each basis function individually, unlike Eqs.(8) and (12).

2.3 Adaptive Scaling Technique

As already described, all of the design variables should be scaled equally in the development of Eq.(13). A simple scaling technique, called the adaptive scaling technique, is introduced in this section. The following equation is used to scale all of the design variables:

$$X_I = \frac{x_I - x_I^L}{x_I^U - x_I^L} \times s \quad I = 1, 2, \dots, n \quad (14)$$

where x_I is the I -th design variable. x_I^U and x_I^L denote the upper and lower bounds of the I -th design variable, respectively. s (>0) in Eq.(14) denotes the scaling coefficient. Using Eq.(14), all of the design variables are scaled between 0 and s . The scaling coefficient s plays an important. If the scaling coefficient s is fixed, (W4) described above may not be satisfied. Thus, scaling coefficient s should be adjusted adaptively. Then, we develop the adaptive scaling technique to satisfy (W4). The algorithm for this technique is summarized as follows:

(STEP1) Initial scaling coefficient s (>0) is set up.

(STEP2) All of the design variables are scaled by Eq.(14).

(STEP3) The width given by Eq.(13) is calculated in the scaled space.

(STEP4) The minimum width r_{\min} is found.

$$r_{\min} = \min_{1 \leq i \leq m} \{r_i\} \quad (15)$$

(STEP5) If $r_{\min} \leq 1$, then scaling coefficient s is updated as follows:

$$s = \alpha \times s \quad (\alpha > 1) \quad (16)$$

Otherwise, the adaptive scaling algorithm will be terminated. On the basis of the author's numerical experiences, $\alpha = 1.2$ is recommended.

The characteristics of this scaling technique are as follows: (1) it can be used to calculate the width in the scaled space and (2) it can be used to examine (W4), which is one of the sufficient conditions. Therefore, the sufficient conditions for a good approximation are always verified.

3. Density Function Using RBF network

The objective of the density function is to discover a sparse region in the design variable space. It is expected that the addition of new sampling points in the sparse region will lead to the global approximation. An important issue is the construction of a density function using the RBF network. In the SAO, the approximate global minimum can be found through the addition of new sampling points. From the point of view of computer-programming code, it is not preferable to construct numerous subroutines. For simpler computer-programming code, it is preferable to use one or two subroutines multiple times. Thus, the density function using the RBF network is developed.

The basic concept of the density function is very simple. The local maxima are generated at the sampling points. To achieve this objective, every output \mathbf{y} of the RBF network is replaced with +1. Suppose that the number of sampling points is m , and $D(\mathbf{x})$ denotes the density function. Eq.(13) with the adaptive scaling technique is also used for the density function. The detailed procedure to construct the density function is summarized as follows:

(D-STEP1) The following vector \mathbf{y}^D is prepared at the sampling points.

$$\mathbf{y}^D = (1, 1, \dots, 1)_{m \times 1}^T \quad (17)$$

(D-STEP2) The weight vector \mathbf{w}^D of the density function $D(\mathbf{x})$ is calculated as follows:

$$\mathbf{w}^D = (\mathbf{H}^T \mathbf{H} + \mathbf{A}) \mathbf{H}^T \mathbf{y}^D \quad (18)$$

(D-STEP3) The density function $D(\mathbf{x})$ is minimized to determine the sparse region in the design variable space.

$$D(\mathbf{x}) = \sum_{i=1}^m w_i^D h_i(\mathbf{x}) \rightarrow \min \quad (19)$$

(D-STEP4) The point at which the density function $D(\mathbf{x})$ is minimized is taken as the new sampling point.

Fig. 4 shows an illustrative example in one dimension. The black dots denote the sampling points.

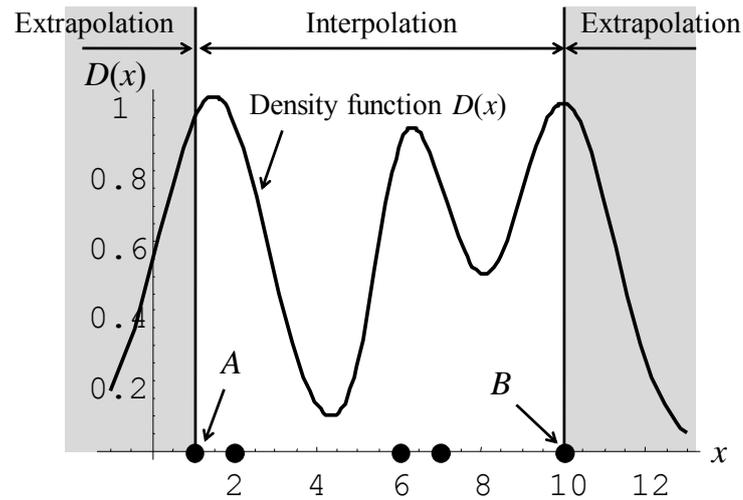


Fig.4 Illustrative example of density function in one dimension

It is clear from Fig.4 that local minima are generated in the sparse region of the sampling points and that local maxima are also generated at the sampling points. The RBF network is basically the interpolation between sampling points: therefore, points A and B in Fig.4 are the lower and upper bounds of the density function.

4. Algorithm for SAO Using RBF network

Fig.5 shows the detailed algorithm for SAO using the RBF network. In this paper, the terminal criterion of SAO is determined by the maximum number of sampling points, m_{\max} .

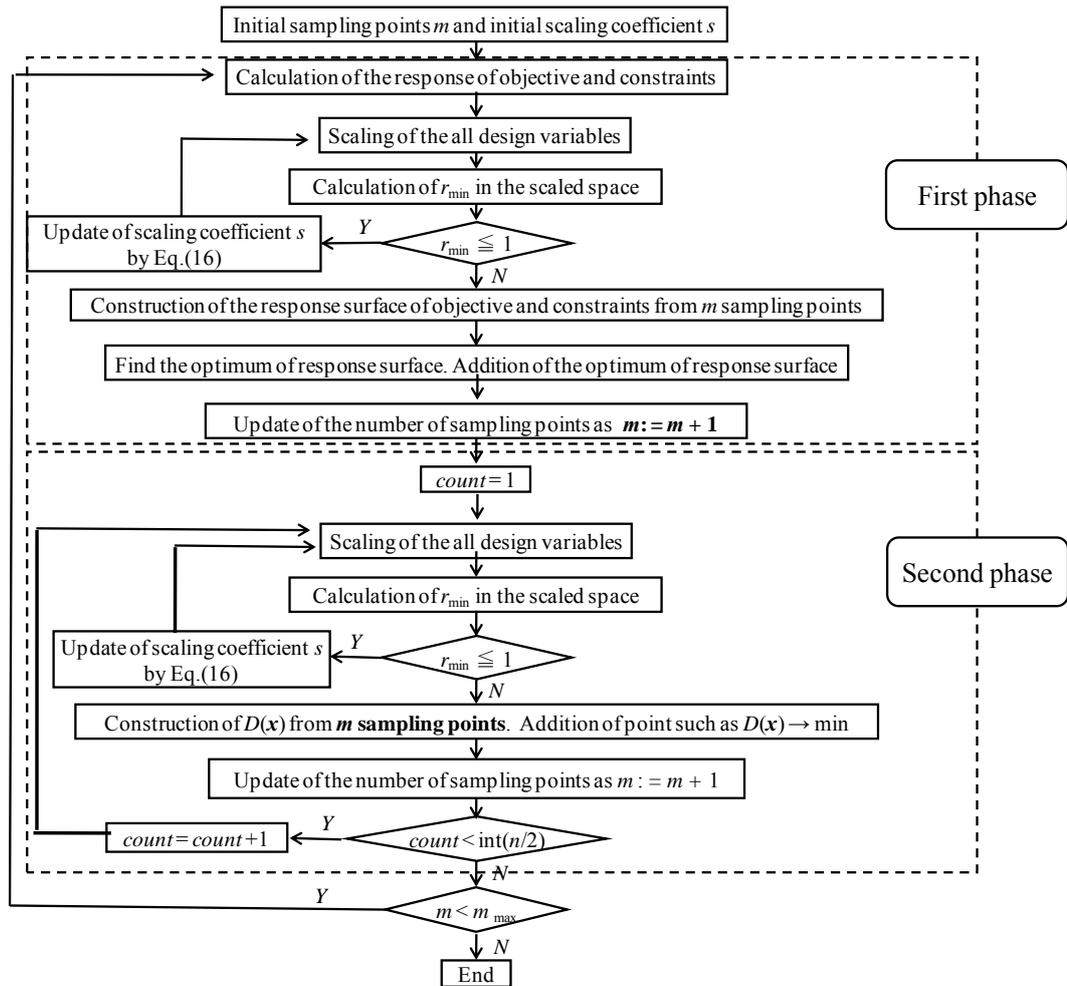


Fig.5 Proposed SAO algorithm

The proposed SAO algorithm is roughly divided into two phases. The first phase is used to construct the response surface and add the optimum of response surface as a new sampling point. Thus, in the first phase, the number of new sampling points is one that is the optimum of the response surface. The second phase is used to construct the density function and add the optimum of the density function as a new sampling point. It should be noted that the density function is constructed until the terminal criterion, which is described later, is satisfied. As a result, many new sampling points will be added, according to the number of design variables, n .

Let us consider the first phase. First, the initial sampling points are determined by using the orthogonal array, the LHD, and so on. The number of sampling points is m . The initial scaling coefficient is also set up. The objective and constraints are calculated at the sampling points. Then, the adaptive scaling technique is applied. After the scaling coefficient is determined with the adaptive

scaling technique, the response surface of the objective and constraints is constructed from m sampling points, by RBF network. The global optimization technique is applied to the response surface, in order to find the optimum of the response surface. The optimum of the response surface is directly taken as the new sampling point. In this phase, the number of sampling points is updated as $m = m + 1$.

Then, the second phase, in which the density function is constructed, is considered. In order to construct the density function by the RBF network, the adaptive scaling technique is also employed. Thus, the scaling coefficient and width are determined and these values are used to construct the density function. The point at which the density function is minimized is then found. The optimum of the density function is taken as the new sampling point, and the number of sampling points is updated as shown in Fig.5. In Fig.5, the parameter *count* is introduced. This parameter controls the number of sampling points that can be obtained by the density function. Thus, in the proposed algorithm, the number of sampling points by the density function varies according to the number of design variables. If the parameter *count* is less than $\text{int}(n/2)$, this parameter is increased as $\text{count} = \text{count} + 1$, and the adaptive scaling techniques is also employed as shown in Fig.5. The terminal criterion in the second phase is given by $\text{int}(n/2)$, where $\text{int}()$ represents the rounding-off. If the terminal criterion is satisfied, the number of sampling point m is compared with m_{\max} . If the m is less than m_{\max} , the objective function and constraints are calculated as shown in Fig.5. Otherwise, the algorithm is terminated. In the SAO, several optima can be obtained because the response surface is constructed repeatedly through the addition of the new sampling points. In this paper, the optimum of the response surface at m_{\max} is taken as the final optimum.

5. Numerical Examples

The validity of the proposed SAO algorithm will now be examined through some typical numerical examples. The objective and constraints are approximated separately by the RBF network. These response surfaces and the density function become a multi-modal function: Therefore, the global optimization technique is required to find the global minimum of the response surface. Then, the Particle Swarm Optimization (PSO) is used as the global

optimizer. In all of the numerical examples, the following PSO parameters are used: (1) the number of particles is 30 and (2) the maximum search iteration is set to 500. The initial scaling coefficient is set to 1 in all of the numerical examples.

5.1 Illustrative Example

Let us consider the following optimization problem.

$$f(\mathbf{x}) = -(x_1 - 1)^2 - (x_2 - 0.5)^2 \rightarrow \min \quad (20)$$

$$g_1(\mathbf{x}) = \frac{[(x_1 - 3)^2 + (x_2 + 2)^2] \exp(-x_2^7)}{12} - 1 \leq 0 \quad (21)$$

$$g_2(\mathbf{x}) = (10x_1 + x_2)/7 - 1 \leq 0 \quad (22)$$

$$g_3(\mathbf{x}) = \frac{(x_1 - 0.5)^2 + (x_2 - 0.5)^2}{0.2} - 1 \leq 0 \quad (23)$$

$$0 \leq \mathbf{x} \leq 1 \quad (24)$$

In Fig.6, the local minimum \mathbf{x}^L and global minimum \mathbf{x}^G are shown by the squares.

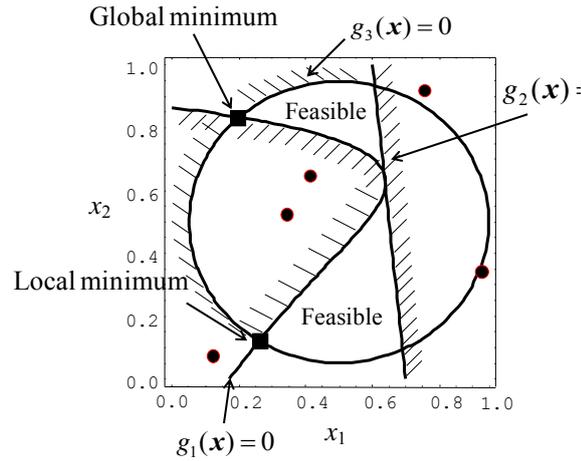


Fig.6 Feasible region, and local and global minima

The objective functions at \mathbf{x}^L and \mathbf{x}^G are given as follows:

$$f(\mathbf{x}_L) = -0.6867 \quad \text{at} \quad \mathbf{x}_L = (0.2623, 0.1223)^T \quad (25)$$

$$f(\mathbf{x}_G) = -0.7484 \quad \text{at} \quad \mathbf{x}_G = (0.2016, 0.8332)^T \quad (26)$$

Two constraints $g_1(\mathbf{x})$ and $g_3(\mathbf{x})$ are active at the local and global minima. It is clear from Fig.6 that there are two separate feasible regions in this problem and that the response surface approach is valid.

The five initial sampling points represented by the dots • in Fig.6 are determined by the LHD, and the maximum number of sampling points m_{\max} is set to 50. The objective and constraints are approximated separately, and the

presented SAO algorithm is applied. The distribution of the sampling points at m_{\max} is shown in Fig.7, and the objective and constraints at the optimum of the response surface through the successive addition of the sampling points are shown in Table 2.

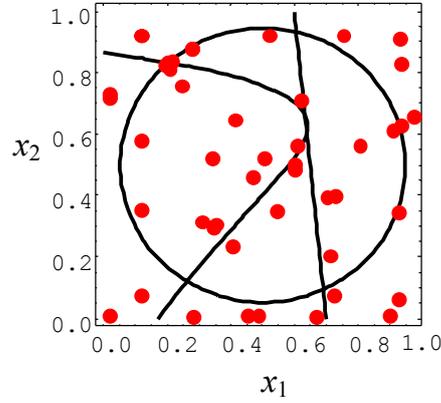


Fig.7 Distribution of sampling points at m_{\max}

Table 2 Objective and constraints at the optimum of the response surface

Number of sampling points	$f(\tilde{\mathbf{x}}_G)$	$g_1(\tilde{\mathbf{x}}_G)$	$g_2(\tilde{\mathbf{x}}_G)$	$g_3(\tilde{\mathbf{x}}_G)$
5	-0.7818	0.2189	-0.7468	-0.2441
11	-0.6321	0.0966	-0.5380	-0.3510
17	-0.7245	0.0414	-0.5886	-0.0920
23	-0.7257	-0.0159	-0.5686	-0.0309
29	-0.7510	0.0158	-0.6006	-0.0100
35	-0.7468	0.0000	-0.5919	0.0000
41	-0.7468	0.0000	-0.5919	0.0000
47	-0.7468	0.0000	-0.5919	0.0000

The sampling points are distributed around the global minimum, and are also distributed in the design variable space. The approximate global minimum is $\tilde{\mathbf{x}}_G = (0.2024, 0.8327)^T$, and the objective at this point is $f(\tilde{\mathbf{x}}_G) = -0.7468$.

5.2 Application to several benchmark problems

The validity of the proposed SAO algorithm is examined through five typical benchmark problems. These problems are listed in Table 3. The initial sampling points are determined by the LHD. In each of these problems, the initial number of sampling points is set to five. Twenty trials are performed with different random seeds. In the constrained problems, the objective and constraints are approximated separately. The results are shown in Table 4.

Table 3 Benchmark problems considered in this paper

No.	Objective and constraints	$f(x_G)$
1	$f(\mathbf{x}) = \sum_{i=1}^5 i \cos[(i+1)x_1 + i] \rightarrow \min$ $0 \leq x_1 \leq 7.5$	$f(x_G) = -12.871$
2	$f(\mathbf{x}) = 2 + 0.01(x_2 - x_1^2)^2 + (1 - x_1)^2 + 2(2 - x_2)^2 + 7 \sin(0.5x_1) \sin(0.7x_1x_2) \rightarrow \min$ $0 \leq \mathbf{x} \leq 5$	$f(x_G) = -1.4565$
3	$f(\mathbf{x}) = \sum_{i=1}^2 x_i \sin x_i + 0.1x_i \rightarrow \min$ $-10 \leq \mathbf{x} \leq 10$	$f(x_G) = 0$
4	$f(\mathbf{x}) = x_1^2 + x_2^2 \rightarrow \min$ $g_1(\mathbf{x}) = -(x_1 + 4)^2/3 - (x_2 - 0.1)^2 + 20 \leq 0$ $-6 \leq x_1 \leq 4 \quad -4 \leq x_2 \leq 6$	$f(x_G) = 11.4371$
5	$f(\mathbf{x}) = -(x_1 - 1)^2 - (x_2 - 0.5)^2 \rightarrow \min$ $g_1(\mathbf{x}) = [(x_1 - 3)^2 + (x_2 + 2)^2] \exp(-x_2^2) - 12 \leq 0$ $g_2(\mathbf{x}) = 10x_1 + x_2 - 7 \leq 0$ $g_3(\mathbf{x}) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 - 0.2 \leq 0$ $0 \leq \mathbf{x} \leq 1$	$f(x_G) = -0.7483$

Table 4 Results of benchmark problems

	Test1	Test2	Test3	Test4	Test5
m_{max}	15	50	50	50	50
Minimum of objective	-12.8708	-1.4557	4.3601E-04	11.4426	-0.7486
Maximum of objective	-12.3941	-1.3107	9.3849E-03	11.9480	-0.7431
Average of objective	-12.7723	-1.4061	3.5725E-03	11.6164	-0.7467
Standard deviation of objective	1.4444E-01	4.8181E-02	3.1839E-03	1.7970E-01	1.7961E-03

It is clear from Table 4 that the proposed SAO algorithm is valid for the benchmark problems considered here.

5.3 Comparison with other SAO algorithms

It is difficult to examine and evaluate all of the SAO algorithms because they employ the specific parameters in the algorithm. In addition, various sequential sampling algorithms are developed. One of the important aspects in the SAO is to reduce the function evaluations. Thus, it may be possible to examine the validity of the proposed algorithm from the view point of function evaluations. In this section, the proposed SAO algorithm is compared with other SAO algorithms through benchmark problems. The benchmark problems are taken

from Refs. [7] and [14], in which the function evaluations are clearly described. The proposed SAO algorithm is compared through six benchmark problems. In the proposed algorithm, initial sampling points are determined by the LHD, and initial number of sampling points is set to 10. The comparisons of function evaluations and the objective at the approximate global minimum are listed in Table 5.

Table 5 Comparison of other method based on Ref.[7] and [14]

Ref.	Objective and constraints	$f(x_G)$	Objective at approximate global minimum	Function evaluations	Proposed algorithm	Function evaluations by the proposed algorithm
14	$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{x_1^6}{3} + x_1x_2 - 4x_2^2 + 4x_2^4 \rightarrow \min$ $-2 \leq x \leq 2$	-1.032	-1.029	44	-1.016	28
14	$f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2) \rightarrow \min$ $-1 \leq x \leq 1$	-2.0	-1.854	60	-1.990	28
14	$f(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2$ $+ 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10 \rightarrow \min$ $-5 \leq x_1 \leq 10 \quad 0 \leq x_2 \leq 15$	0.398	0.398	36	0.398	34
14	$f(x) = [1 + (x_1 + x_2 + 1)^2 \times (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)]$ $\times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)] \rightarrow \min$ $-2 \leq x \leq 2$	3.000	3.000	77	3.050	60
7	$f(x) = x_1 \sin(x_2) + x_2 \sin(x_1) \rightarrow \min$ $-2\pi \leq x \leq 2\pi$	-9.629	N/A	80	-9.628	40
7	$f(x) = x_1 \sin(x_1) + x_2 \sin(x_2) \rightarrow \min$ $-2\pi \leq x \leq 2\pi$	-9.629	N/A	60	-9.623	30

5.4 Examination of width

The validity of the width proposed in this paper is examined through the following problem.

$$f(x) = \frac{1}{2} \sum_{i=1}^n (x_i^4 - 16x_i^2 + 5x_i) \rightarrow \min \quad (27)$$

$$-5 \leq x \leq 5 \quad (28)$$

The global minimum x_G is $x_G = (-2.9035, -2.9035, \dots, -2.9035)^T$. The number of design variables n is set to 10. In this case, the objective function at x_G is $f(x_G) = -391.661$. The PSO is applied to this problem directly. The number of particles is set to 20, and the maximum search iteration is set to 500. Therefore, 10000 function evaluations are required to find the global minimum. The PSO results are shown in Table 6 through 10 trials.

Table 6 Results of direct search by the PSO

Trial	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	obj.
1	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-391.6600
2	-2.9036	-2.9036	-2.9036	-2.9035	-2.9035	-2.9035	-2.9035	2.7468	-2.9035	-2.9035	-377.5200
3	-2.9046	-2.9027	-2.9021	-2.9038	-2.9032	-2.9047	-2.9015	-2.9043	-2.9029	-2.9049	-391.6600
4	-2.9036	-2.9035	-2.9036	-2.9036	-2.9035	-2.9035	-2.9036	-2.9035	-2.9036	-2.9036	-391.6600
5	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-391.6600
6	-2.9035	-2.9035	-2.9035	-2.9035	-2.9036	-2.9035	2.7468	-2.9036	-2.9035	-2.9035	-377.5200
7	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-391.6600
8	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-391.6600
9	-2.9036	-2.9034	-2.9036	-2.9036	-2.9034	-2.9036	-2.9035	-2.9035	-2.9036	-2.9036	-391.6600
10	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-2.9035	-391.6600

To reduce the function evaluations, we try to find the global minimum with 500 function evaluations. Thus, the maximum number of sampling points is set to 500. First, 30 sampling points are distributed at random in the design variable space. Ten trials are performed using Eqs.(8) and (13). The results are shown in Table 7 and Table 8.

Table 7 The result by using Eq.(8)

Trial	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	obj.
1	-2.6590	-2.7695	-2.1054	3.0079	-3.2740	-2.7108	-2.6121	-2.3852	2.7003	-0.9410	-314.1416
2	-2.0759	-2.7692	2.2905	-1.9245	2.7989	-2.6619	1.9850	-2.0531	2.4101	-1.4912	-273.9235
3	-2.9410	-2.8063	-3.0479	-2.6390	-2.9142	-2.8250	-2.8031	0.9497	-2.7844	0.7700	-318.2305
4	-2.8926	-2.9712	-2.0732	-3.2507	-2.8878	-2.7514	-3.3558	-1.2210	-2.2069	-1.5235	-324.6039
5	-2.9462	2.5448	-2.9124	-2.7911	-3.0628	2.3251	0.6797	-2.9464	-2.9779	2.7018	-308.3509
6	2.4241	-0.3086	3.1126	-3.6446	-1.6238	-2.6086	-3.4843	1.1949	-0.0281	1.9153	-192.1104
7	-2.8263	-2.9896	2.7270	-3.1419	-2.6376	-2.6945	-2.4389	-3.2174	-1.6055	-3.3261	-347.9492
8	-2.9634	-2.7063	2.5490	-2.6656	-2.5071	-2.9957	-2.9850	2.8242	-3.0073	2.2608	-341.3516
9	-2.6664	2.4891	-2.7121	-2.2098	-2.7480	-0.1432	1.5402	-2.7654	-3.2011	-3.0773	-300.2067
10	-2.3563	-2.7148	-2.7854	-2.6837	2.3436	-1.2683	2.4122	-2.6995	-2.7142	-2.1761	-321.3327

Table 8 The result by using Eq.(13)

Trial	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	obj.
1	-2.8096	-2.8428	-2.8138	-2.8923	-2.9249	-2.8322	-2.7792	-2.9189	-2.9251	-2.8174	-390.8271
2	-2.8796	-2.7992	-2.9779	-2.9068	-2.8315	-2.9377	-2.9344	-2.9574	-2.9395	-2.8551	-391.1336
3	-2.9828	-2.8559	-2.8442	-2.7923	-2.8739	-2.9228	-2.7560	-2.9362	-2.9070	-2.8233	-390.7393
4	-2.8815	-2.8551	-2.9466	-2.8876	-2.8993	-2.8246	-2.8708	-2.8602	-2.8473	-2.8967	-391.3663
5	-2.9453	-2.9166	-2.7878	-2.8491	-2.8713	-2.7786	-2.9730	-2.7636	-2.9755	-2.9006	-390.5783
6	-2.9192	-2.8345	-2.9351	-2.8462	-2.9019	-2.8998	-2.8975	-2.9197	2.6884	-2.8685	-377.2915
7	-2.9376	2.6801	-2.8858	-2.8898	-2.8615	-2.8626	-2.9049	-2.8773	-2.8294	-2.8921	-377.2669
8	2.6569	-2.9470	-2.9554	-2.8557	-2.8986	-2.9416	-2.8419	-2.8627	-2.9532	-2.8738	-377.1143
9	-2.9211	-2.8614	2.7425	-2.9349	-2.8388	-2.8856	-2.8809	-2.8981	-2.9182	-2.8922	-377.3801
10	-2.8901	-2.8468	-2.8828	-2.9049	-2.9252	2.6898	-2.8235	-2.9735	-2.8813	-2.8938	-377.2004

By comparing Table 7 with Table 8, it is clear that better results can be obtained, by using Eq.(13). Through five trials (Trial No.1 – Trial No.5), the

approximate global minimum could be obtained by Eq.(13). In addition, the other trials (Trial No.6 – Trial No.10) yield quasi-optimums. However, an approximate global minimum cannot be obtained with Eq.(8). These results imply that it is preferable to apply a different width to each basis function.

5.5 Optimum Design of Tension/Compression Spring

One of the most popular test problems proposed by Arora can be considered [25]. Many researchers have used this as benchmark problem in the structural optimization [26,27,28]. The design variables are (1) the diameter d ($=x_1$), (2) mean coil diameter D ($=x_2$), and (3) number of active coils N ($=x_3$). The problem can be formulated as follows:

$$f(\mathbf{x}) = (2 + x_3)x_1^2x_2 \rightarrow \min \quad (29)$$

$$g_1(\mathbf{x}) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0 \quad (30)$$

$$g_2(\mathbf{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \quad (31)$$

$$g_3(\mathbf{x}) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0 \quad (32)$$

$$g_4(\mathbf{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0 \quad (33)$$

$$0.05 \leq x_1 \leq 2.00 \quad (34)$$

$$0.25 \leq x_2 \leq 1.30 \quad (35)$$

$$2.00 \leq x_3 \leq 15.0 \quad (36)$$

The orthogonal array $L9$, which is shown in Table 9, is used to determine the initial sampling points.

Table 9 Orthogonal array $L9$

	x_1	x_2	x_3
No.1	0.05	0.25	2
No.2	0.05	0.775	8.5
No.3	0.05	1.3	15
No.4	1.025	0.25	8.5
No.5	1.025	0.075	15
No.6	1.025	1.3	2
No.7	2	0.25	15
No.8	2	0.075	2
No.9	2	1.3	8.5

m_{\max} is set to 150, and 11 trials are performed to compare the past researches. The results obtained by applying the proposed SAO algorithm are shown in Table 10. It is clear from Table 10 that the function evaluations are drastically reduced in comparison with those in the past researches.

Table 10 Comparison of results on the optimum design of tension/compression spring

Design variables	Best solutions found				
	Arora ⁽²⁵⁾	Coello ⁽²⁶⁾	Ray ⁽²⁷⁾	Hu ⁽²⁸⁾	This research
x_1	0.053396	0.051480	0.050417	0.051466	0.050000
x_2	0.399180	0.351661	0.321532	0.351384	0.314777
x_3	9.185400	11.632201	13.979915	11.608659	14.650042
$g_1(\mathbf{x})$	0.000019	-0.002080	-0.001926	-0.003336	-0.018820
$g_2(\mathbf{x})$	-0.000018	-0.000110	-0.012944	-0.000110	-0.006566
$g_3(\mathbf{x})$	-4.123832	-4.026318	-3.899430	-4.026318	-3.837790
$g_4(\mathbf{x})$	-0.698283	-0.731239	-0.752034	-0.731324	-0.756815
$f(\mathbf{x})$	0.012730	0.012705	0.013060	0.012667	0.013103
Function Call	N/A	900000	1291	N/A	66
Average of $f(\mathbf{x})$	N/A	0.012769	0.013436	0.012719	0.013273
Worst of $f(\mathbf{x})$	N/A	0.012822	0.013580	N/A	0.013643

5. Conclusions

In this paper, the Sequential Approximate Optimization (SAO) algorithm using the RBF network has been proposed. The Gaussian function is employed as the basis function. We have examined the width of the Gaussian function, which

affects the accuracy of the response surface. By examining the width equation proposed by Nakayama, some sufficient conditions for a good approximation are introduced. Then, a new equation to determine the width has been proposed. In addition, a simple scaling technique called the adaptive scaling technique has also been proposed. In this technique, the sufficient conditions for a good approximation are always verified. Clearly, it is better to optimize the width in these kinds of meta-modeling techniques. However, optimization of the width is very time-consuming task. Therefore, it is preferable to determine the width with a simple manner. Many meta-modeling techniques, such as Kriging, RBF network, and Support Vector Regression (SVR), have been proposed. The Gaussian function, which is sometimes called the Gaussian kernel, is commonly employed in all these methods. The equivalence between ordinary Kriging and SVR has been reported under the assumption that the covariance function is used as the kernel function [29]. The equivalence between SVM and the regularization neural network has been also reported [30]. This equivalence can be extended to RBF network, considering the suggestions of Ref. [31]. Thus, it is considered that the equivalence between SVM and RBF network can be established. In the Gaussian kernel, the width plays an important role. Therefore, one of the important issues is the determination of the width with a simple manner. It is expected that the proposed width with the adaptive scaling technique is applicable to Kriging, SVR, and so on, in which the Gaussian function is employed.

Second, the sampling strategy has been examined. In the SAO, the optimum of the response surface is taken as the new sampling point in order to improve the local accuracy. In addition, new sampling points in the sparse region are required for a global approximation. To determine the sparse region, the density function constructed by the RBF network has been developed. This density function generates local minima in the sparse region, so that the minimum of this function can be taken as a new sampling point. In the proposed SAO algorithm, the density function is constructed repeatedly until the terminal criterion is satisfied. As the result, many new sampling points can be obtained. Through typical mathematical and engineering optimization problems, the validity of the proposed SAO algorithm has been examined.

Acknowledgements.

The authors have developed Eq.(13), through in-depth discussions with Prof. Nakayama, H.(Konan University). The reviewers also gave us the useful and constructive suggestions to improve the quality of this paper. We would like to thank to them for their useful and constructive comments.

References:

1. Wang GG, Shan S (2007) Review of Metamodeling Techniques in Support of Engineering Design Optimization, *Trans. of ASME, J. of Mechanical Design*, 129: 370-380.
2. Myers RH, Montgomery DH (1995) *Response Surface Methodology*, Wiley, New York.
3. Donald J, Schonlau M, Welch WJ (1998) Efficient Global Optimization of Expensive Black-Box Functions, *J. of Global Optimization*, 13: 455-492.
4. Simpson TW, Mauery TM, Korte JJ, Mistree F (2001) Kriging Models for Global Approximation in Simulation-Based Multidisciplinary Design Optimization, *AIAA Journal*, 39(12): 2233-2241.
5. Martin JD, Simpson TW (2005) Use of Kriging Models to Approximate Deterministic Computer Models, *AIAA Journal*, 43(4): 853-863.
6. Muller AA, Messac A (2005) Extended Radial Basis Functions: More Flexible and Effective Metamodeling, *AIAA Journal*, 43(6): 1306-1315.
7. Hussain MF, Barton RR, Joshi SB (2002) Metamodeling: Radial Basis Functions, versus Polynomials, *European Journal of Operational Research*, 138: 142-154.
8. Jin R, Chen W, Simpson TW (2001) Comparative Studies of Metamodelling Techniques under Multiple Modelling Criteria, *Structural and Multidisciplinary Optimization*, 23: 1-13.
9. Nakayama H, Arakawa M, Sasaki R (2002) Simulation-Based Optimization Using Computational Intelligence, *Optimization and Engineering*, 3: 201-214.
10. Fang H, Horstemeyer MF (2006) Global Response Approximation with Radial Basis Function, *Engineering Optimization*, 38(4): 407-424.
11. McDonald DB, Grantham WJ, Tabor WL, Murphy MJ (2007) Global and Local Optimization Using Radial Basis Function Response Surface Models, *Applied Mathematical Modelling* 31: 2095-2110.
12. Haykin S (1994) *Neural Networks: A Comprehensive Foundation*, Macmillan College Publishing Company: 236-284.
13. Simpson TW, Peplinski JD, Kock PN, Allen JK (2001) Metamodels for Computer-based Engineering Design: Survey and Recommendations, *Engineering with Computers*, 17: 129-150.
14. Wang GG (2003) Adaptive Response Surface Method Using Inherited Latin Hypercube Design Points, *Trans. of ASME/ J. of Mechanical Design*, 125: 210-220.
15. Rodriguez JF, Perez VM, Padmanabhan D, Renaud JE (2001) Sequential Approximate Optimization Using Variable Fidelity Response Surface Approximations, *Structural and Multidisciplinary Optimization*, 22: 24-34.

16. Perez VM, Renaud JE, Watson LT (2002) Adaptive Experimental Design for Construction of Response Surface Approximations, *AIAA Journal*, 40(12): 2495-2503.
17. Sobester A, Leary SJ, Keane AJ (2005) On the Design of Optimization Strategies Based on Global Response Surface Approximation Models, *J. of Global Optimization*, 33: 31-59.
18. Huang D, Allen TT, Notz WI, Zeng N (2006) Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models, *J. of Global Optimization*, 34: 441-466.
19. Simpson TW, Mistree F (2001) Kriging Models for Global Approximation in Simulation-Based Multidisciplinary Design Optimization, *AIAA Journal*, 30(12): 2233-2241.
20. Sasena MJ, Papalambros PY, Goovaerts P (2002) Exploration of metamodeling sampling criteria for constrained global optimization, *Engineering Optimization*, 34(3): 263-278.
21. Sasena MJ, Papalambros PY, Goovaerts P (2002) Global Optimization of Problems with Disconnected Feasible Regions via Surrogate Modeling, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA2002-5573.
22. Sharif B, Wang GG, ElMekkawy TY (2008) Mode Pursuing Sampling Method for Discrete Variable Optimization on Expensive Black-Box Functions, *Trans. of ASME, J. of Mechanical Design*, 130: 021402-1-021402-11.
23. Kurtaran H, Eskandarian A, Marzougui D, Bedewi NE (2002) Crashworthiness Design Optimization Using Successive Response Surface Approximations, *Computational Mechanics*, 29: 409-421.
24. Orr MJL, Introduction to Radial Basis Function Networks, <http://www.anc.ed.ac.uk/rbf/rbf.html> (On-line available)
25. Arora JS (1989) Introduction to Optimum Design, McGraw-Hill, New York.
26. Coello Coello CA (2000) Use of a Self-Adaptive Penalty Approach for Engineering Optimization Problems, *Computers in Industry*, 41: 113-127.
27. Ray T, Saini P (2001) Engineering Design Optimization Using Swarm with an Intelligent Information Sharing among Individuals, *Engineering Optimization*, 33: 735-748.
28. Hu XH, Eberhart RC, Shi YH (2003) Engineering Optimization with Particle Swarm, *IEEE Swarm Intelligence Symposium*: 53-57.
29. An W, Sun Y (2006) An Equivalence between SILF-SVR and Ordinary Kriging, *Neural Processing Letters*, 23: 133-141
30. Andras P (2002) The Equivalence of Support Vector Machine and Regularization Neural Networks, *Neural Processing Letters*, 65: 97-104.
31. Poggio T, Girosi F (1990) Networks for Approximation and Learning, *Proc. of the IEEE*, 78(9): 1481-1497