Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 193 (2004) 385–404
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

An efficient algorithm for Kriging approximation and optimization with large-scale sampling data

S. Sakata [a,*], F. Ashida [a], M. Zako [b]

[a] *Department of Electronic Control Systems Engineering, Interdisciplinary Faculty of Science and Engineering, Shimane University, 1060, Nishikawatsu-cho, Matsue City 690-8504, Japan*
[b] *Department of Manufacturing Science, Graduate School of Engineering, Osaka University, 2-1, Yamada-Oka, Suita City 565-0871, Japan*

Received 29 January 2003; received in revised form 18 September 2003; accepted 14 October 2003

Abstract

This paper describes an algorithm to improve a computational cost for estimation using the Kriging method with a large number of sampling data. An improved formula to compute the weighting coefficient for Kriging estimation is proposed. The Sherman–Morrison–Woodbury formula is applied to solving an approximated simultaneous equation to determine a weighting coefficient. A profile of the matrix is reduced by sorting of given data.

Applying the proposal formula to several examples indicates its characteristics. As a numerical example, layout optimization of a beam structure for eigenfrequency maximization is solved. The results show an applicability and effectiveness of the proposed method.

Keywords: Kriging estimation; Sherman–Morrison–Woodbury formula; Computational cost; Structural optimization

1. Introduction

An approximate optimization method is available for industrial design problems, and several methods have been studied. It seems that those methods can be classified in three categories such as, the response surface method (RSM) with optimization of coefficients for a base function, the neural network approximation (NN) and an estimation method with using observed values at sampling locations to compute an estimated value at an optional location in a solution space. Although these all can be used practically in industry, each method has different features to be applied to approximation. Several comparisons among those methods have been reported [1–4].

The RSM is one of the very effective approaches for an optimization problem with small numbers of design variable and its solution space is not so complex. Many researchers have been reported its

* Corresponding author. Tel./fax: +81-852-32-6840.
E-mail address: sakata@ecs.shimane-u.ac.jp (S. Sakata).

0045-7825/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2003.10.006

effectiveness of the RSM against optimization problems in engineering [5–8]. Barthelemy and Haftka [9] or Haftka and Scott [10] reported on their survey of optimization using the RSM. It seems that parameter optimization to determine coefficients of an approximate function is not so difficult. However, the RSM, which is based on experimental programming, normally requires the assumption of the order of the approximated base function because the approximation process is performed using the least-square method for the coefficients of the function. Therefore, the designer must evaluate the schematic shape of the objective function over an entire solution space. This will sometimes be difficult because it requires an understanding of the qualitative tendency of the entire design space. This is because it would be difficult to determine an order of the base function to minimize the approximation error without any knowledge of the solution space. As the another problem in using the RSM, Shi et al. [11] pointed out the difficulty of applying RSM based on experimental programming to a design problem having many design variables.

NN has been used for an approximate optimization to solve difficult optimization problems [12–14]. NN generally minimizes the sum of the approximation errors at sampling locations, so that the accuracy of the approximated value at a sampling location is relatively high. As the other merit of using NN, Carpenter and Barthelemy [1] reported that NN offers more flexibility to allow fitting than RSM.

NN, however, presents some practical difficulties. One is the computational cost incurred for learning. A learning process will be same as optimization for a large number of design variables, and it will involve high computational cost. The other problem is, for example, the need for the operator to be skilled or experienced in using NN [1].

The Kriging method, which is one of the spatial estimation methods with using the sample data, has been noticed recently. Several researches on an approximate optimization using Kriging estimation were reported [15–18]. Simpson et al. [19] reported a comparison between RSM and the Kriging method. Sakata et al. [20] reported a comparison between NN and the Kriging method.

To use Kriging estimation for structural optimization, more sample points in a solution space will be required for more precise estimation. Especially, using a large number of sampling (training) data will enable NN or the Kriging method to estimate a complex function, a valid approximated surface for a multi-peaked solution space can be produced. However, increase the number of sampling data generally causes a higher computational cost.

Computational cost of the Kriging method to determine the estimation model is not so high, however, that to estimate a function value at each location will be higher than NN or RSM. The reason for high computational cost for Kriging estimation is that large-scale simultaneous equations must be solved to determine a weighting coefficient for each location where that is to be estimated. A large number of sample points are required for the more precise estimation, while the number of equations increases in the number of sample data, therefore, increase of the total number of sample points causes high computational cost for estimation. In case of using a large number of sampling data, reducing a computational cost to solve a simultaneous equation to determine the weighting coefficient is very important to apply the Kriging method to optimization of a complex problem such as an approximate optimization of multi-peaked solution space.

In this paper, to reduce a computational cost for Kriging estimation, a new formula to calculate a weighting coefficient is proposed. Some numerical examples illustrate an application of proposed method. As an example of structural optimization by using the proposed method, layout optimization of a beam structure is attempted using the proposed method.

2. Kriging estimation

The Kriging method [21,22] is a method of spatial prediction that is based on minimizing the mean error of the weighting sum of the sampling values. A linear predictor of $\widehat{Z}(\boldsymbol{s}_0)$ can be obtained from Eq. (1) using a weighting coefficient $\boldsymbol{w} = \{w_1, w_2, \ldots, w_n\}^{\mathrm{T}}$.

$$\widehat{Z}(\boldsymbol{s}_0) = \sum_{i=1}^{n} w_i Z(\boldsymbol{s}_i), \tag{1}$$

where $Z(\boldsymbol{s}_1), Z(\boldsymbol{s}_2), \ldots, Z(\boldsymbol{s}_n)$ observed values which are obtained at the nth known locations $\boldsymbol{s}_1, \boldsymbol{s}_2 \ldots, \boldsymbol{s}_n$ in a solution space, $\widehat{Z}(\boldsymbol{s}_0)$, which shows an estimated value of $Z(\boldsymbol{s}_0)$ at $\boldsymbol{s}_0 \in \boldsymbol{S}$, which is the point where we want to estimate the value of the function, is obtained as follows.

$\widehat{Z}(\boldsymbol{s}_0)$ will be determined to minimize the mean-squared predictor error as

$$\sigma^2(\boldsymbol{s}_0) = E\{|\widehat{Z}(\boldsymbol{s}_0) - Z(\boldsymbol{s}_0)|^2\} = -\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} + 2\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\gamma}^{**}. \tag{2}$$

A conditioned extreme value problem for Eq. (2) with an unbiased condition for $\widehat{Z}(\boldsymbol{s}_0)$, the following Lagrange function can be determined.

$$\phi(\boldsymbol{w}, \lambda) = -\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} + 2\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\gamma}^{**} - 2\lambda(\boldsymbol{w}^{\mathrm{T}}1 - 1), \tag{3}$$

where $\boldsymbol{\Gamma}$ and $\boldsymbol{\gamma}^{**}$ are a coefficient function matrix and vector, which are expressed as

$$\boldsymbol{\Gamma} = \{\gamma(\boldsymbol{s}_i - \boldsymbol{s}_j)\}_{ij}, \tag{4}$$

$$\boldsymbol{\gamma}^{**} = \{\gamma(\boldsymbol{s}_1 - \boldsymbol{s}_0), \ldots, \gamma(\boldsymbol{s}_n - \boldsymbol{s}_0)\}, \tag{5}$$

where γ is a correlation function that is described as a semivariogram model. A semivariogram is a variance function in a probabilistic field, which is used to express the dispersion of the data. In this study, the Gaussian-type semivariogram model was adopted since estimated surface using the Gaussian-type semivariogram model semivariogram will be smooth and continuous, making it suitable for use in an optimizing design. The Gaussian-type semivariogram model is expressed by the following form,

$$\gamma(\boldsymbol{h}; \boldsymbol{\theta}) = \theta_0 + \theta_1\left[1 - \exp\left(-\left\{\frac{|\boldsymbol{h}|}{\theta_2}\right\}^2\right)\right], \tag{6}$$

where θ_0, $\theta_1 \geqslant 0$, $\theta_2 > 0$ are the model parameters. Typically, the parameter $\boldsymbol{\theta} = \{\theta_0, \theta_1, \theta_2\}$ in Eq. (6) is determined, for example, using the least-square method. To determine the parameter $\boldsymbol{\theta}$, Cressie's criterion [23], which is a robust efficient estimator to a change in the scale of data, is used in this paper.

By applying stationary condition $\delta\phi = 0$, the following standard equation is obtained.

$$\begin{bmatrix} \boldsymbol{\Gamma} & \mathbf{1} \\ \mathbf{1}^{\mathrm{T}} & 0 \end{bmatrix}\begin{Bmatrix} \boldsymbol{w} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{\gamma}^{**} \\ 1 \end{Bmatrix}. \tag{7}$$

An estimated value can be calculated by Eq. (1) using a solution of Eq. (7) for each $\boldsymbol{s}_0$. To determine a weighting coefficient $\boldsymbol{w}$, a simultaneous equation Eq. (7) must be solved. Since a dimension of a coefficient matrix $\boldsymbol{\Gamma}$ is equal to the number of sampling data, it will be large when a large number of sampling data are used for estimation.

3. Fast Kriging algorithm

The weighting coefficient $\boldsymbol{w}$ can be calculated by solving Eq. (7). Then we can obtain the following form as a solution.

$$w_i = \Gamma_{ij}^{-1}\gamma_j^{**} + \left(\frac{1 - 1_i\Gamma_{ij}^{-1}\gamma_j^{**}}{1_i\Gamma_{ij}^{-1}1_j}\right)\Gamma_{ij}^{-1}1_j, \tag{8}$$

where Γ_{ij}^{-1} is an inverse matrix of Γ_{ij} and $1_i = (1, 1, \ldots, 1)$.

From Eq. (8), it is recognized that Γ_{ij}^{-1} must be calculated at a first step in estimation process, and $\Gamma_{ij}^{-1}\gamma_j^{**}$ must be calculated for each estimated location to determine $\boldsymbol{w}$. A computational cost for Kriging estimation is mainly affected by the cost for solving a simultaneous equation as

$$\Gamma_{ij}x_j = \gamma_i^{**}, \tag{9}$$

where x_j is an unknown variable vector. Generally, an effective algorithm such as the Gaussian elimination with LU factorization is used to solve a linear simultaneous equation with symmetric coefficient matrix for several different right-side vectors. However, a coefficient matrix Γ_{ij} will be generally a full matrix, high computational cost will be involved for estimation with using a large number of sampling data even if a LU factorization is used. To reduce a computational cost for Kriging estimation, therefore, reducing a computational cost for solving Eq. (9) should be endeavored.

Now we assume that the Gaussian-type model is used as a semivariogram model, a component of the coefficient matrix Γ_{ij} can be expressed by

$$\Gamma_{ij} = \gamma(l_{ij};\boldsymbol{\theta}) = \theta_0 + \theta_1\left[1 - \exp\left(-\left\{\frac{l_{ij}}{\theta_2}\right\}^2\right)\right], \tag{10}$$

where $l_{ij} = |\boldsymbol{s}_i - \boldsymbol{s}_j|$ shows a distance between two locations.

A semivariogram model parameter vector $\boldsymbol{\theta}$ is determined for once generation of estimation model. Then a semivariogram matrix that is expressed by Eq. (10) can be rewritten as follows by difference of two matrices such as

$$\begin{aligned}\Gamma_{ij} &= \theta_0 + \theta_1\left(1 - \exp\left(-\left(\frac{l_{ij}}{\theta_2}\right)^2\right)\right) = \theta_0 + \theta_1 - \theta_1\exp\left(-\left(\frac{l_{ij}}{\theta_2}\right)^2\right)\\ &= (\theta_0 + \theta_1)1_i \otimes 1_j - \theta_1\exp\left(-\left(\frac{l_{ij}}{\theta_2}\right)^2\right).\end{aligned} \tag{11}$$

The inverse of Γ_{ij} can be, therefore, calculated by using Sherman–Morrison–Woodbury formula [24] as

$$\begin{aligned}\Gamma_{ij}^{-1} &= \left((\theta_0 + \theta_1)1_i \otimes 1_j - \theta_1\exp\left(-\left(\frac{l_{ij}}{\theta_2}\right)^2\right)\right)^{-1}\\ &= A_{ij} - \frac{A_{ij}1_j \otimes 1_i A_{ij}}{\dfrac{1}{(\theta_0 + \theta_1)} + l_i A_{ij} 1_j} = A_{ij} - \frac{\sum_j A_{ij} \otimes \sum_i A_{ij}}{\dfrac{1}{(\theta_0 + \theta_1)} + \displaystyle\sum_i \sum_j A_{ij}},\end{aligned} \tag{12}$$

where

$$A_{ij} = \left(-\theta_1\exp\left(-\left(\frac{l_{ij}}{\theta_2}\right)^2\right)\right)^{-1}. \tag{13}$$

Thus $\Gamma_{ij}\gamma_j^{**}$ can be calculated as

$$\Gamma_{ij}^{-1}\gamma_j^{**} = A_{ij}\gamma_j^{**} - \frac{\sum_j A_{ij} \otimes \sum_i A_{ij}}{\dfrac{1}{(\theta_0 + \theta_1)} + \displaystyle\sum_i \sum_j A_{ij}}\gamma_j^{**}. \tag{14}$$

Here, we assume that a component of Γ_{ij} can be regarded as zero when l_{ij} is enough large. In general cases, zero components will appear randomly in $\overline{\Gamma}_{ij}$ which is an approximate matrix of Γ_{ij}, such as

$$\Gamma_{ij} \approx \overline{\Gamma}_{ij} = \begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} & \cdots & \theta & \gamma_{1,i+1} & \cdots & \gamma_{1,m} \\ & \gamma_{2,2} & \cdots & \gamma_{2,i} & \theta & \cdots & \theta \\ & & \ddots & \vdots & \vdots & & \vdots \\ & & & \gamma_{i,i} & \gamma_{i,i+1} & & \gamma_{i,m} \\ & & & & \gamma_{i+1,i+1} & & \theta \\ & \text{sym.} & & & & \ddots & \vdots \\ & & & & & & \gamma_{m,m} \end{bmatrix}, \tag{15}$$

where θ is a constant.

Generally, a computational cost of Eq. (14) will be same degree or more than direct calculation of $\Gamma_{ij}^{-1}\gamma_j^{**}$ even if an approximated coefficient matrix $\overline{\Gamma}_{ij}$ where many components are constant. However, if a bandwidth of non-constant components of $\overline{\Gamma}_{ij}$ is enough narrow, a calculation cost for the first term of right side of Eq. (14) can be clearly reduced comparing with that for a full matrix by using an effective algorithm such as the skyline method. Therefore, minimization of a profile is applied to an approximated coefficient matrix of A_{ij}. A profile β can be determined by Eq. (16)

$$\beta = \sum_{i=1}^{n} \beta_i, \tag{16}$$

where β_i is the number of components from a minimum line that has a non-zero component to ith diagonal component for each ith column.

Fig. 1 illustrates a scheme of transformation of $\overline{\Gamma}_{ij}$ into a banded matrix Γ_{ij}^{*} to minimize a profile of a coefficient matrix A_{ij}. In minimizing the profile, constant components in $\overline{\Gamma}_{ij}$ can be regarded as zero. Only the non-constant components in a skyline, which is shown in Fig. 1, will be used to compute $A_{ij}\gamma_j^{**}$ in Eq. (14). If a bandwidth can be enough reduced, it is considered that a computational cost of $A_{ij}\gamma_j^{**}$ will be also reduced.

For practicality, it is considered that the components of Γ_{ij} had better to be arranged to reduce a bandwidth of Γ_{ij}^{*}. Since the components of $\overline{\Gamma}_{ij}$ can be rewritten as

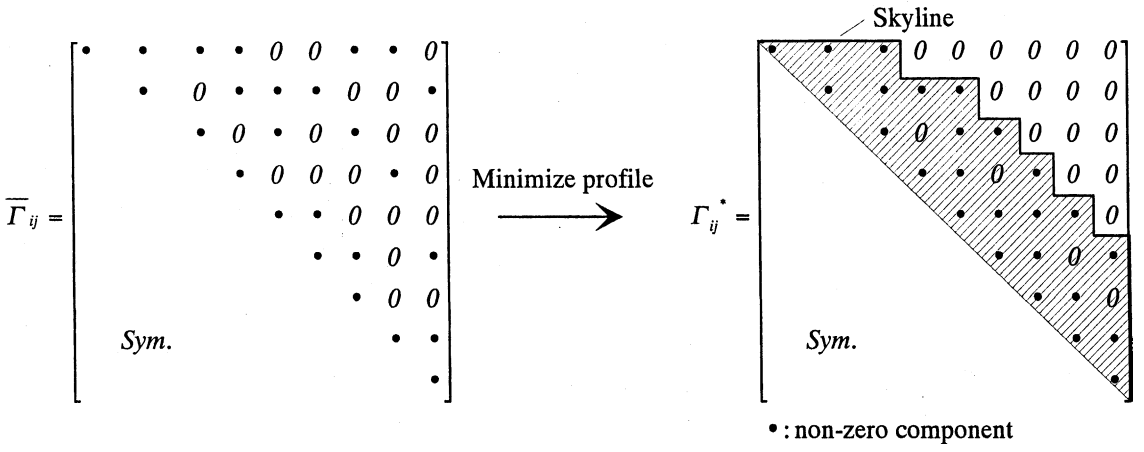

Fig. 1. Transformation of $\overline{\Gamma}_{ij}$ into a banded matrix Γ_{ij}^{*}.

$$\overline{\Gamma}_{ij} = \begin{cases} \Gamma_{ij} : \Gamma_{ij}/\Gamma_{ii} \geqslant th, \\ \theta : \Gamma_{ij}/\Gamma_{ii} < th, \end{cases} \tag{17}$$

the bandwidth can be reduced by arranging the order of the γ^{**} according to each distance between $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$. As the simplest approach, the order may be arranged by the distance between $\boldsymbol{s}_1$ and $\boldsymbol{s}_i$ for one-dimensional problem. For a higher dimensional problem, a general algorithm to minimize bandwidth of a coefficient matrix will be used.

If we can reduce a profile of $\overline{\Gamma}_{ij}$, then a banded symmetric matrix Γ^*_{ij} can be obtained. In this case, a coefficient matrix A_{ij} expressed by Eq. (13) is also to be a banded matrix A^*_{ij}. Therefore, an approximated form of Eq. (14) can be expressed as the following equation.

$$\Gamma^{-1}_{ij}\gamma^{**}_j \approx A^*_{ij}\gamma^{**}_j - \frac{\sum_j A^*_{ij} \otimes \sum_i A^*_{ij}}{\dfrac{1}{(\theta_0+\theta_1)} + \displaystyle\sum_i \sum_j A^*_{ij}}\gamma^{**}_j = A^*_{ij}\gamma^{**}_j - \frac{1}{c}a_i\bar{\boldsymbol{a}}_j\gamma^{**}_j, \tag{18}$$

where

$$a_i = \sum_j A^*_{ij}, \tag{19}$$

$$\bar{\boldsymbol{a}}_j = \sum_i A^*_{ij}, \tag{20}$$

$$c = \frac{1}{(\theta_0+\theta_1)} + \sum_i \sum_j A^*_{ij} = \frac{1}{(\theta_0+\theta_1)} + \sum_j \bar{\boldsymbol{a}}_j. \tag{21}$$

Substitution of Eq. (18) into Eq. (8) yields an approximation form of the weighting coefficient as

$$w_i \approx w^*_i = A^*_{ij}\gamma^{**}_j - \frac{1}{c}\bar{\boldsymbol{a}}_j\gamma^{**}_j a_i + \frac{1 - 1_i\left(A^*_{ij}\gamma^{**}_j - \dfrac{1}{c}\bar{\boldsymbol{a}}_j\gamma^{**}_j a_i\right)}{1_i\left(A^*_{ij}1_j - \dfrac{1}{c}\bar{\boldsymbol{a}}_j 1_j a_i\right)}\left(A^*_{ij}1_j - \frac{1}{c}\bar{\boldsymbol{a}}_j 1_j a_i\right) = \bar{\boldsymbol{w}}_i + \frac{1 - \sum_i \bar{\boldsymbol{w}}_i}{\sum_i a_i}a_i, \tag{22}$$

where

$$\bar{\boldsymbol{w}}_i = A^*_{ij}\gamma^{**}_j - \frac{1}{c}\bar{\boldsymbol{a}}_j\gamma^{**}_j a_i. \tag{23}$$

Since the second term of Eq. (23) can be easily rewritten as

$$\frac{1}{c}\bar{\boldsymbol{a}}_j\gamma^{**}_j a_i = \frac{1}{c}\sum_i (A^*_{ij}\gamma^{**}_j)a_i, \tag{24}$$

an additional calculation cost for the second term of Eq. (23) in an iterating process involves only nth summation. Since a calculation cost for inverse of a banded matrix Γ^*_{ij} is clearly less than that of Γ_{ij}, if a calculation cost for $A_{ij}\gamma^{**}_j$ is enough reduced, total calculation cost for w^*_i will be also reduced.

4. Discussions about correlation between a threshold and estimation error

Reducing components of a coefficient matrix may cause increase of an estimation error. In the following section, therefore, an effect of reduction of components of a coefficient matrix on estimation error is investigated.

4.1. One-dimensional problem

As one of the simplest examples, estimation of the following equation is attempted by using the proposed method. This function is multi-peaked, continuous and smooth in a considered region.

$$f(x) = \sin(0.02x) \times \cos(0.2(x + 25)) \times \cos(0.1(x + 50)), \quad 0.0 \leqslant x \leqslant 100.0. \tag{25}$$

Sample values used for estimation are calculated at several points, which are generated at regular intervals as sampling point. Since the function is multi-peaked, it is considered that many sample points should be involved for precise estimation. In this case, 101st sampling points are generated.

Fig. 2 shows an exact surface of Eq. (26) and estimated surface produced by using the Kriging method without using the proposed algorithm. From Fig. 2, it is considered that good estimation can be obtained for such multi-peaked function. The RMS error between original and estimated surface is 0.0273. This error is calculated by using about one thousand exact function values and estimated values. In this case, the parameters of semivariogram for estimation are as follows. These parameters are determined by using the Cressie's criteria [23] and the Burnell's Algorithm [25].

$$\{\beta_0, \beta_1, \beta_2\} = \{1.00 \times 10^{-5}, 1.62 \times 10^{-1}, 5.76 \times 10^{0}\}. \tag{26}$$

Now we attempt to apply the proposed formula to estimation of this function. To reduce a computational cost for Eq. (8), a threshold *th* in Eq. (17) must be determined. A computational cost for Eq. (8) will be more reduced when *th* is larger, an estimation errors, however, will increase. To determine an appropriate threshold, therefore, a relationship between *th* and estimation errors must be investigated.

Fig. 3 shows an example of a relationship between a threshold and distance from an optional location. In this case, its relationship at $x = 0.0$ is illustrated. This figure indicates, for example, estimation at location $x = 0.0$ uses the sample data that ranges between $x = 0.0$ and $x = 26.0$ when $th = 10^{-4}$. From this figure, it is clearly found that a weight of the data for estimation will goes small exponentially as it is far from a location where is estimated, and it is considered that an observed value at a location far from an location that is attempted to estimate has few effect on a result of estimation.

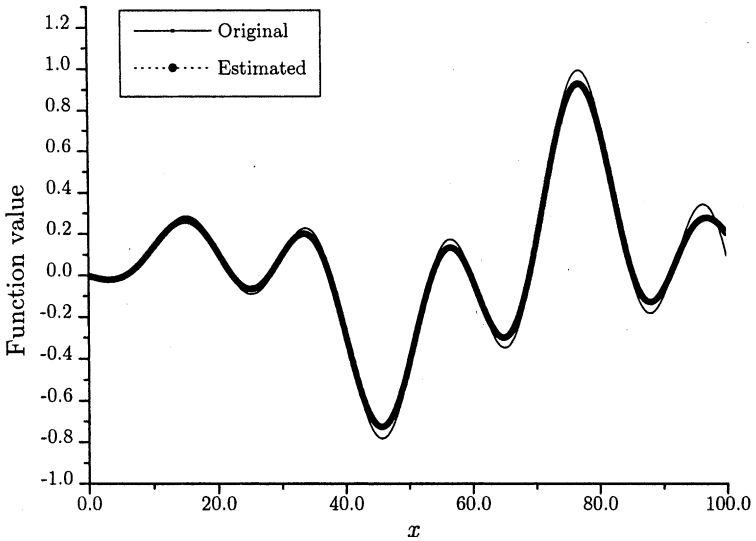

Fig. 2. Test function and its estimation.

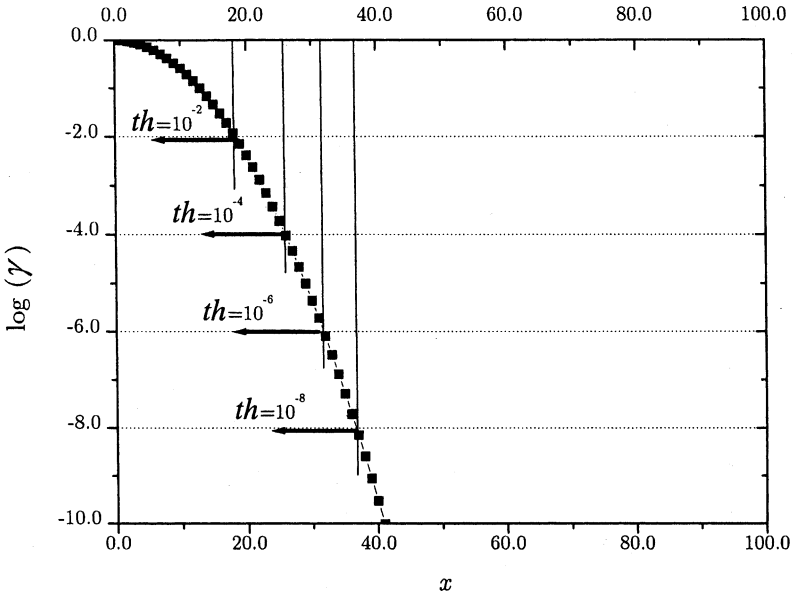

Fig. 3. Semivariogram function value at each location for $x = 0.0$.

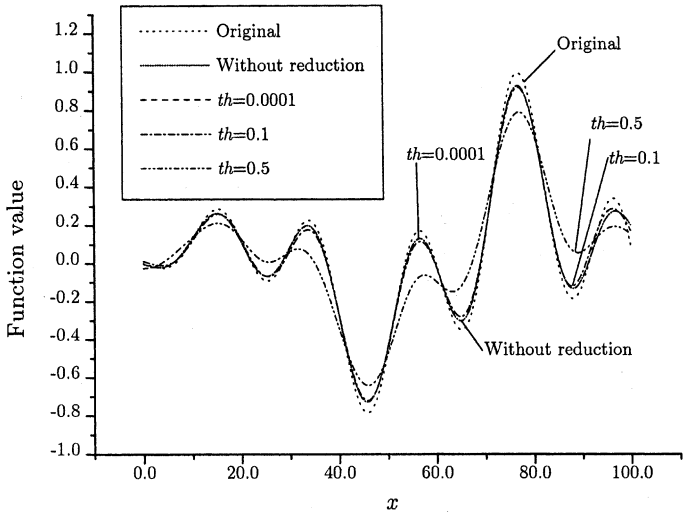

Fig. 4. Estimated surface using each threshold.

To evaluate an effect of a threshold on accuracy of estimation, change of the estimated function by each threshold is illustrated in Fig. 4. From Fig. 4, it is found that an estimated surface becomes to be different from an exact surface of a considered function as a threshold is larger. This fact shows that accuracy of estimation decreases with reduction of the number of observed data that are used for estimation at each location. Fig. 5 shows an effect of a threshold on estimation error. It can be recognized that an estimation

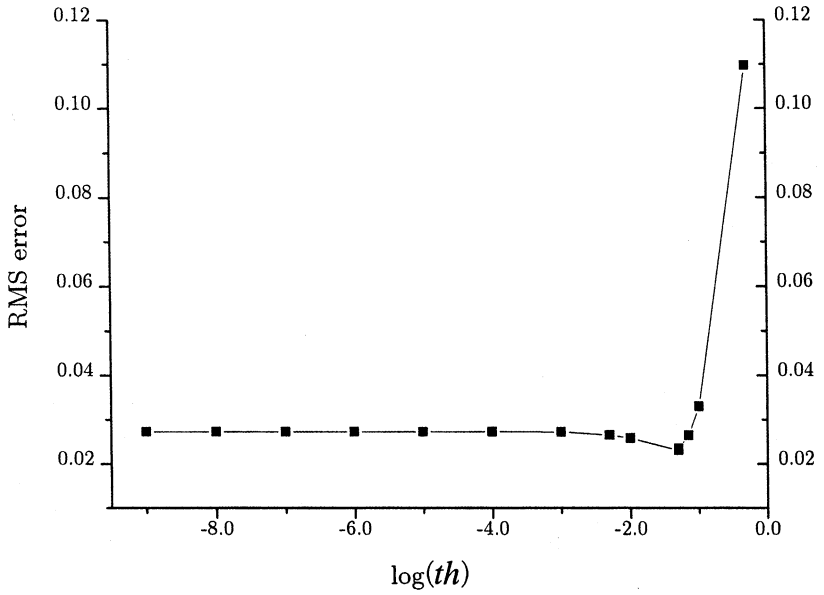

Fig. 5. Relationship between RMS error and a threshold.

error hardly increases when th is smaller than 10^{-3}, and estimation error increases dramatically when th is larger than 10^{-1} in this case.

4.2. Two-dimensional problem

As a more general problem, two-dimensional function, which is expressed as the following equation, is approximated using the proposed formula.

$$f(x_1, x_2) = \sin(0.4x_1) + \cos(0.2x_2 + 0.2), \quad 0.0 \leqslant x_1,\ x_2 \leqslant 10.0. \tag{27}$$

A surface of the original function is shown in Fig. 6. Although a surface is not so complex, sometimes a large number of sampling points are required. For example, high dense sampling points will be used for precise estimation. In this case, 2601 sampling data are prepared to estimate this surface. Each 51 points are generated as sampling points at regular interval for each axis.

For this function, effects of a threshold on estimation are investigated. Fig. 7 shows a reduction of computational cost for estimation at different ten thousands points, which are used to draw an estimated surface by each different threshold. Normalized value of total numbers of profiles, computational time to execute LU factorization of a coefficient matrix Γ^*_{ij}, total computational time to estimate ten thousands values are plotted in Fig. 7. All components are effectively improved by raising a threshold, especially, computational time to execute LU factorization is greatly improved.

To determine a threshold, change of estimated surface by difference of thresholds is investigated. Estimated surfaces for thresholds $th = 10^{-8}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$ are shown in Figs. 8–12. From these figures, it is found that the surface is well estimated when a threshold is smaller than 10^{-3}, and the surface becomes to be fluctuated when a threshold is larger than 10^{-2}. From these results, the larger threshold causes invalid estimation.

For detail evaluation, change in estimation error by thresholds is also investigated. Fig. 13 shows an estimation error for each threshold. From Fig. 13, effect on estimation error can be neglected when a

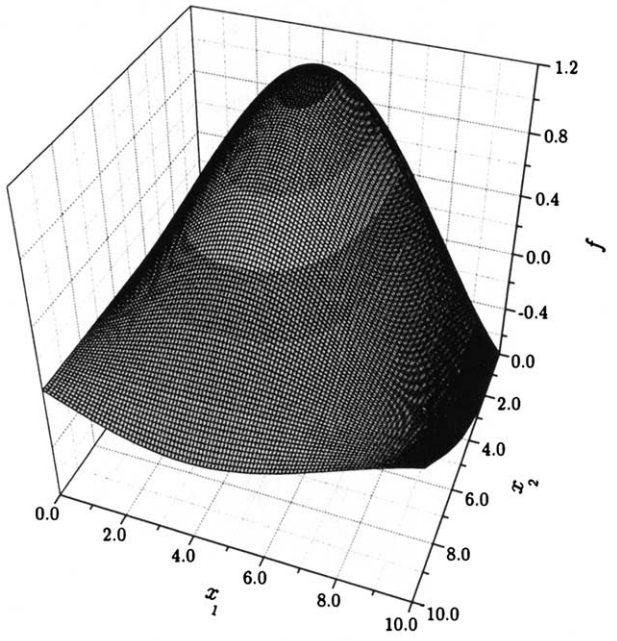

Fig. 6. Surface of the original function.

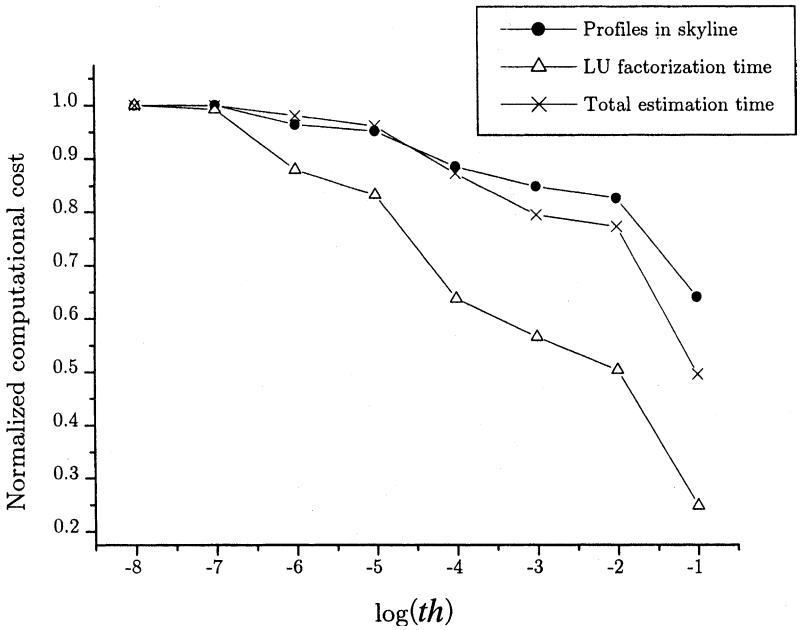

Fig. 7. Improvement of computational cost with change of a threshold.

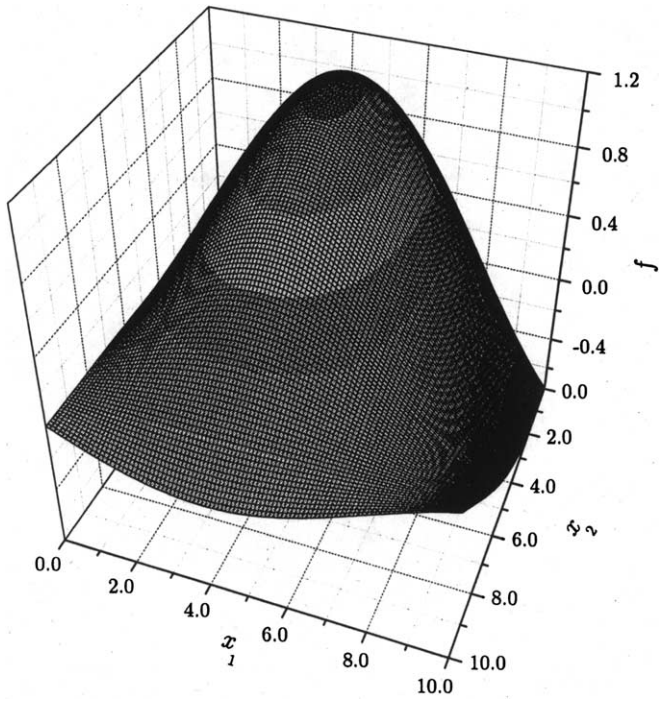

Fig. 8. Estimated surface for $th = 10^{-8}$.

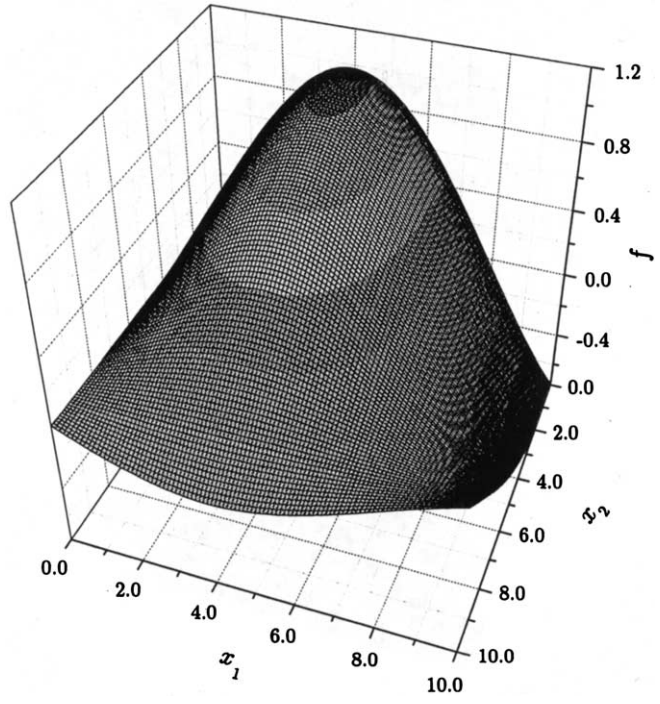

Fig. 9. Estimated surface for $th = 10^{-4}$.

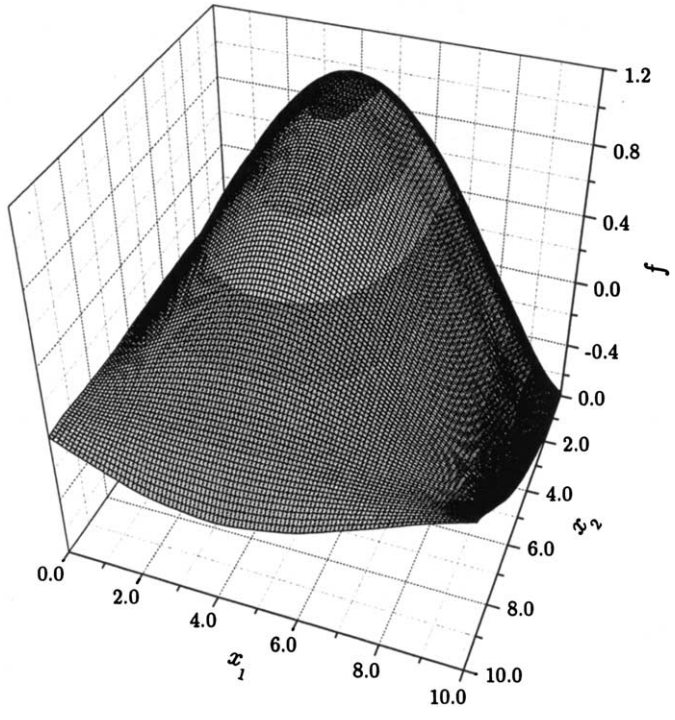

Fig. 10. Estimated surface for $th = 10^{-3}$.

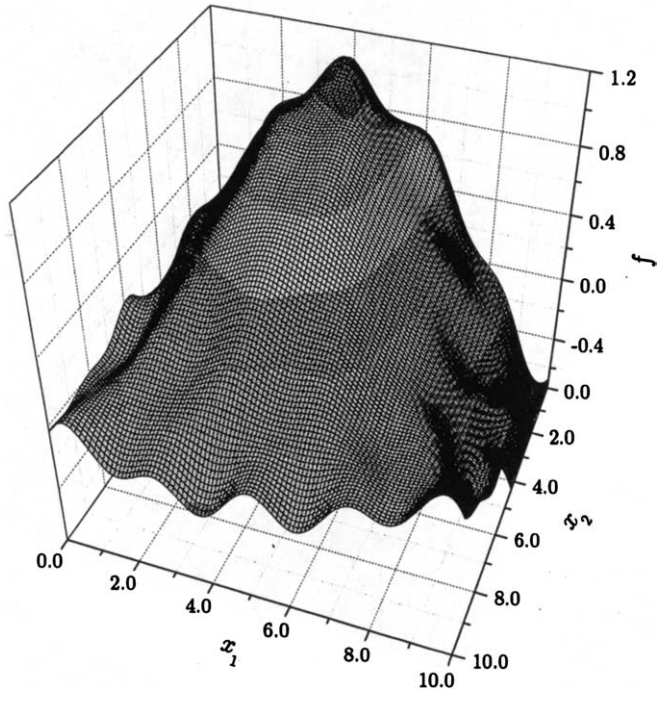

Fig. 11. Estimated surface for $th = 10^{-2}$.

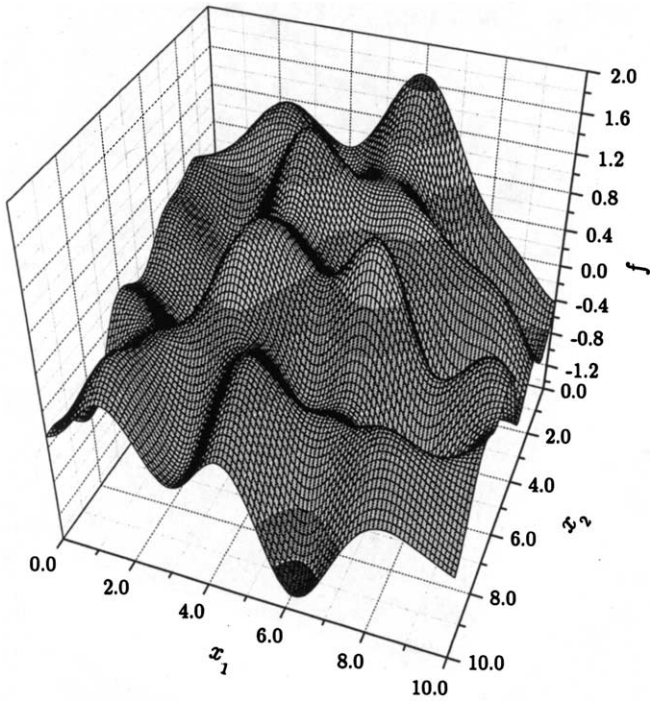

Fig. 12. Estimated surface for $th = 10^{-1}$.

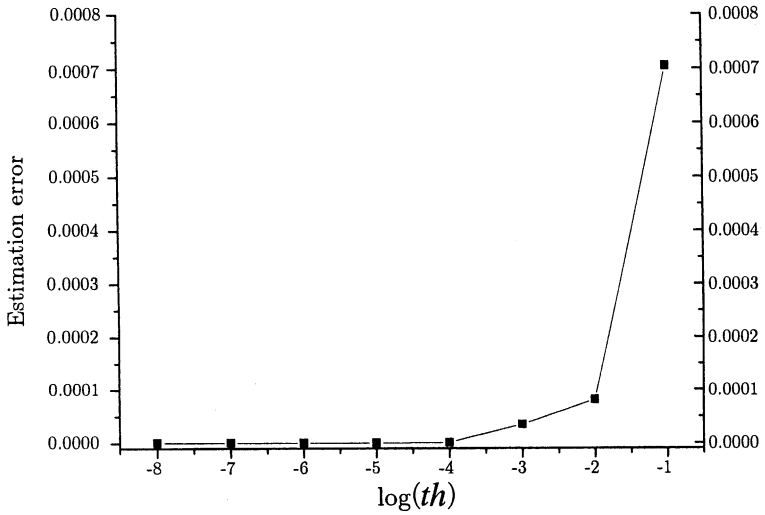

Fig. 13. Mean estimation error for each threshold.

threshold is smaller than 10^{-4}. From Figs. 8–13, it can be considered that a threshold can be allowed to be less than 10^{-3}. Although the larger threshold can be adopted and a computational cost can be reduced if any estimation errors are allowed, generally, it can be recommended that a threshold is smaller than 10^{-3} in

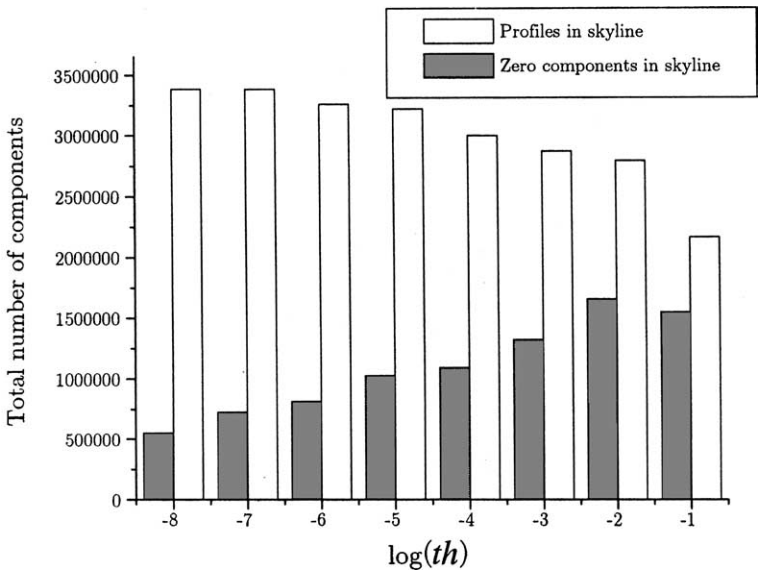

Fig. 14. Total number of zero components in Γ_{ij}^{*} for each threshold.

the case of two-dimensional problem. In this case, about 20% of computational cost can be reduced, the effectiveness of the proposed method was illustrated.

As the other viewpoint, effect of a bandwidth reduction procedure on a computational cost is discussed. Reduction efficiency of computational cost is greatly affected by efficiency of a bandwidth reduction algorithm. We have no approximation algorithm for bandwidth minimization, thus the Cuthill–Mckee [26] and Gibbs–Poole–Stockmeyer [27] algorithms are used in this case. By using these algorithms, total profiles of a coefficient matrix can be effectively reduced, however, a large numbers of zero components are still remained in an arranged coefficient matrix. Fig. 14 shows a total profile β and the total number of zero components in the skyline of an arranged coefficient matrix. Total profiles are effectively reduced when a threshold becomes large, however, the total number of zero components in the skyline are increased. Since only non-zero components are used to compute an estimated value, a computational cost for estimation can be reduced increasingly if a more effective algorithm to reduce the profiles is used.

5. Structural optimization using the proposed method

5.1. Layout optimization of a beam structure

As an application of the proposed method to a structural optimization, a layout optimization problem of a beam structure for eigenfrequency maximization is solved. Eigenfrequency problem of a beam structure is solved by using the finite element method. To determine a location of an additional element to maximize eigenfrequency of a structure, effect of additional element on eigenfrequency for each different inserted location must be investigated. Although a surface of solution space will be almost continuous and smooth, the effect of insertion can be evaluated at discrete locations in the case of using FEM analysis for evaluation, and thus an approximation optimization method will be effective to solve the optimization problem.

In this paper, a layout optimization of additional member to a two-dimensional beam structure is solved. Geometry of a beam structure is illustrated in Fig. 15. Total numbers of finite element nodes are 201.

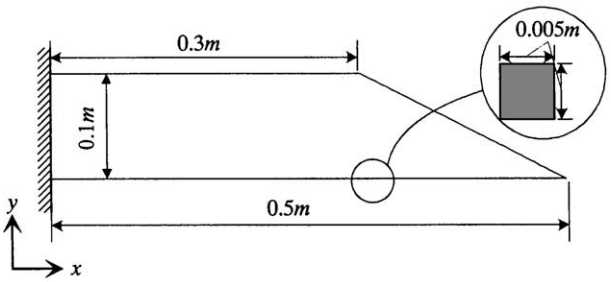

Fig. 15. Geometry of a beam structure.

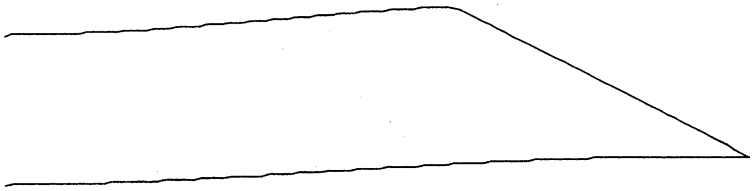

Fig. 16. First-order eigenmode of the beam structure.

Fig. 16 shows a first-order eigenmode of the structure. The optimization problem can be formulated as follows.

$$\left.\begin{array}{ll} \text{find} & \boldsymbol{l} = \{l_1, l_2\} \\ \text{to maximize} & \omega \end{array}\right\}, \tag{28}$$

where l_1 shows a location of one edge of an inserted member on a lower element of a structure, which is illustrated in Fig. 15, l_2 shows a location of the another edge on an upper element, and ω is a first-order eigenfrequency of a structure. The optimization problem is, therefore, to find a layout of an inserted member to maximize a first-order eigenfrequency of a structure.

Since a transformation of eigenmode or multi-peaked optimization should be considered when a layout optimization of a beam structure for eigenfrequency problem is solved, a large number of sampling points should be prepared for valid approximate optimization.

In this case, 2500 sampling points are used to estimate a solution space. Each 50 sampling points are generated at regular interval for each axis. By using the proposed method, a surface of solution space is estimated. A threshold th it set at 10^{-3} in this case. Fig. 17 shows an estimated surface. Estimated values at ten thousands of different locations are computed. Fig. 18 shows an original surface of a solution space, which is plotted using solutions for all combinations of nodes. Comparing Figs. 17 and 18, it can be recognized that a good estimated surface can be obtained. A total estimation time was about 1562 seconds by using Pentium4-2 GHz based PC, while about 2728 seconds was taken if bandwidth is not reduced. From this result, it is recognized that about 42% of a computational cost for estimation can be reduced.

By using the proposed algorithm with a threshold $th = 10^{-3}$, the approximated optimum solution is searched. Since this optimization problem is not a convex optimization problem, a global optimization method is used to find an optimum solution. In this case, the CSSL method [28], which was proposed by authors, is used to solve the optimization problem.

An estimated optimum solution and computational cost are shown in Table 1. l_1 in Table 1 expresses a location of one edge of an inserted member on a lower element of a beam structure which is illustrated in

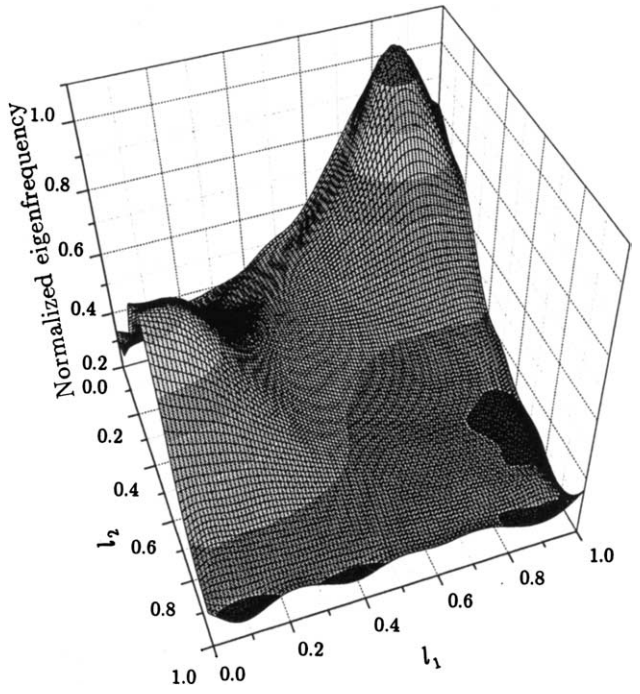

Fig. 17. Estimated solution space for eigenfrequency optimization of a beam structure.

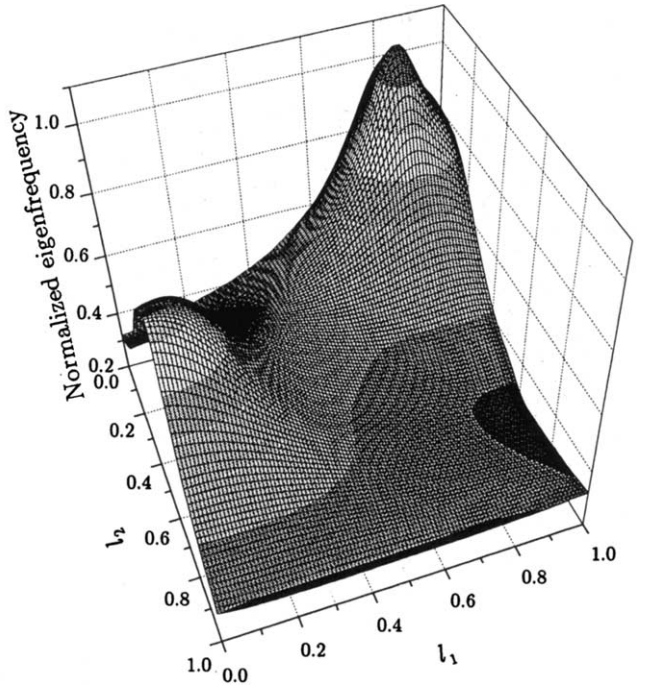

Fig. 18. Solution space for eigenfrequency optimization of a beam structure.

Table 1
Estimated optimum solution and computational time

	Estimated optimum solution	Computational time (s)
Original Kriging optimization	$(l_1, l_2) = (0.829, 0.089)$	402.6
Proposed method	$(l_1, l_2) = (0.829, 0.089)$	278.4
Optimum solution	$(l_1, l_2) = (0.83, 0.09)$	–

Fig. 15, l_2 in Table 1 expresses a location of another edge on an upper element. Variables l_1 and l_2 is normalized by 0.5, if l_1 equals 0.5, for example, it shows that one edge is located at $x = 0.25$ m on a lower element of the beam structure. "Optimum solution" in Table 1 expresses an exact solution that is obtained by fully FEM analyses.

An estimated optimum solution, which is obtained by using the proposed method, accords with an optimum solution by using the original method, which uses not-reduced coefficient matrix. On the other hand, a computational cost using the proposed method is reduced by about 30% of the original method. This result shows that the proposed method enables to reduce a computational cost with similar precision of approximate optimization, and it shows effectiveness of the proposed algorithm.

Fig. 19 shows an optimum layout created by an estimated optimum solution. This layout almost accords with the optimum solution, which is selected by all candidates evaluated by exact FEM analysis. This result shows validity of the proposed approximate optimization procedure.

5.2. Eigenfrequency optimization of a wing structure

As the other problem, thickness optimization of a stiffened hollow wing structure to control of an eigenfrequency of a structure is solved by using the proposed method. Fig. 20 shows geometry and design variables of the winged structure. t_i in Fig. 20 shows thickness at each substructure of the wing structure. t_4 and t_5 show thickness of each lib. As shown in Fig. 20, thickness at five substructures is optimized. Fig. 21 shows a finite element model of the structure. A 4-node isoparametric shell element is used. Total number of elements is 858, total number of nodes is 845. In this case, a difference between first and second eigenfrequency is minimized. Therefore, the optimization problem can be formulated as follows:

$$\left.\begin{array}{ll} \text{find} & \boldsymbol{t} = \{t_1, t_2, t_3, t_4, t_5\} \\ \text{to minimize} & \omega_2 - \omega_1 \\ \text{such as} & 0.01 \leqslant t_i \leqslant 0.51 \end{array}\right\}, \tag{29}$$

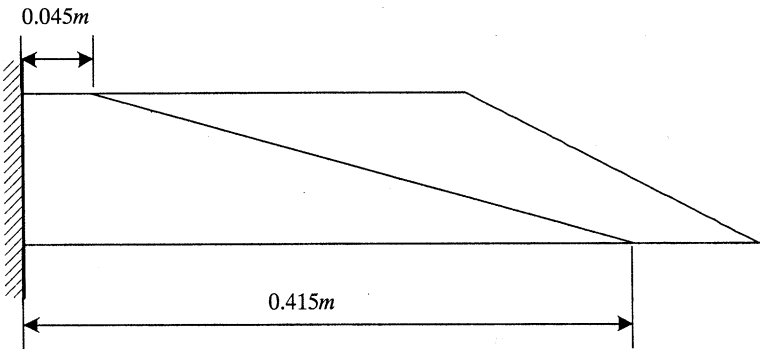

Fig. 19. Estimated optimum layout.

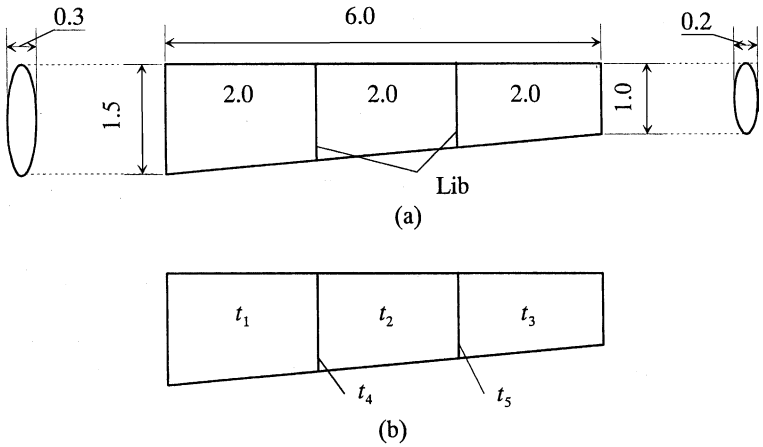

Fig. 20. Scheme of a wing structure (unit: m). (a) Geometry and (b) design variables.

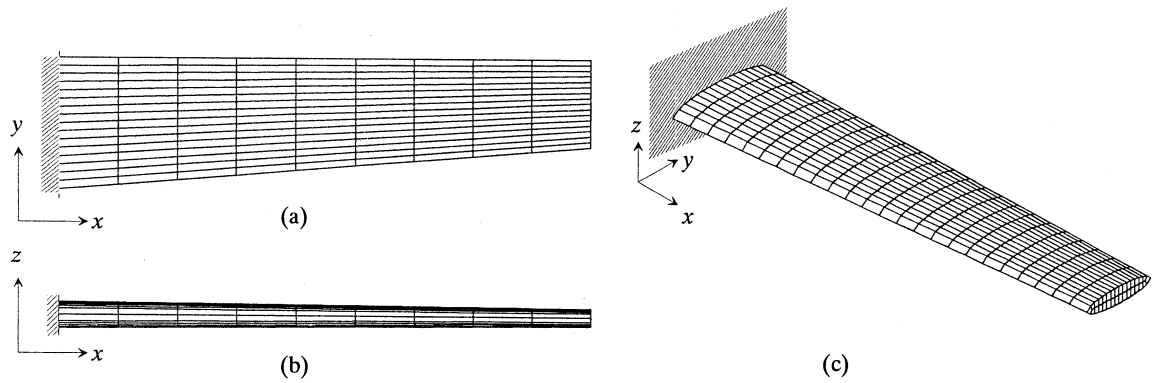

Fig. 21. Finite element model of a wing structure. (a) Top view, (b) side view and (c) bird view.

where $\boldsymbol{t}$ is thickness, ω_1 and ω_2 are first- and second-order eigenfrequencies. Namely, five-dimensional optimization problem is solved by using the proposed method.

To apply the proposed method, sampling values are evaluated in the solution space. In this case, five sampling points are generated at regular intervals for each design variable axis, therefore 3125 sampling point are totally generated. A total computational time for sampling evaluations was about 4130 minutes by using 2.4 GHz-based Windows PC, therefore an average computational time for a single analysis was about 1.3 minutes.

According to the previous discussion, we evaluate computational times and estimated optimum solutions for each threshold $th = 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$. Table 2 shows estimated optimum solutions and computational times for each threshold. Similarly to the previous example, the CSSL method is applied to minimizing the estimated objective function. From Table 2, it can be recognized that a similar solution to the optimum solution, which is obtained with no approximation ($th = 0$) of the coefficient matrix, can be obtained when the threshold is less than 10^{-3}. In the case of $th = 10^{-3}$, computational time can be reduced at about 48% of the conventional approach, therefore effectiveness of the proposed method can be also shown. This numerical result shows effectiveness of the proposed method for high dimensional optimization problem.

Table 2
Estimated optimum thickness and computational time

Threshold	t_1 (m)	t_2 (m)	t_3 (m)	t_4 (m)	t_5 (m)	Time (s)
0 (Original Kriging)	0.127	0.376	0.01	0.183	0.374	1759
10^{-5}	0.127	0.376	0.01	0.183	0.374	1069
10^{-4}	0.127	0.376	0.01	0.183	0.374	1041
10^{-3}	0.127	0.376	0.01	0.183	0.374	840
10^{-2}	0.126	0.376	0.01	0.182	0.375	576
10^{-1}	0.117	0.364	0.01	0.213	0.071	117

6. Conclusion

This paper has described an algorithm to improve a computational cost for Kriging estimation. The Kriging method involves solving a simultaneous equation to determine weighting coefficients for estimation. For a large numbers of sampling points or a large-scale system, a computational cost for estimation is mainly affected by solving a simultaneous equation. Especially, a computational cost for solving a simultaneous equation must be reduced when the Kriging method is used for an approximate optimization, since an estimated value or a gradient component of an estimated surface must be computed at several locations in a solution space.

To reduce a computational cost for Kriging estimation, The Sherman–Morrison–Woodbury formula has been introduced to compute an inverse of a coefficient matrix. And by introducing a threshold into a coefficient matrix, an approximated coefficient matrix can be constructed. By sorting a component of coefficient matrix by using Cuthill–Mckee and Gibbs–Poole Stockmeyer algorithm, a computational cost for Kriging estimation can be reduced by about 20%.

A structural optimization, which is to determine the optimum layout of a beam structure to maximize a first-order eigenfrequency, has been solved by using the proposed method. From the numerical results, the effectiveness of the proposed method has been illustrated. In this case, the proposed method saved about 30% of a computational cost comparing with the original method, which does not have a bandwidth reduction process, and gave a good estimated optimum solution. These results show validity and effectiveness of the proposed method.

As a higher dimensional optimization problem, thickness optimization of a wing structure has been solved. From the numerical result, a computational cost can be also reduced at about 48% of the original method in the case of the threshold is 10^{-3}, effectiveness of the proposed method on a higher dimensional problem is also shown.

As an additional discussion, possibility of an improvement in computational cost by using the proposed method has been pointed out. If the more effective algorithm for minimization of bandwidth of a coefficient matrix can be used, a computational cost for Kriging estimation will be more reduced.

References

[1] W.C. Carpenter, J.F.M. Barthelemy, A Comparison of polynomial Approximations and Artificial Neural Nets as Response Surfaces, A collection of technical papers: the 33rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA, Dallas, TX, April 13–15, 1992, pp. 2474–2482.

[2] E. Nikolaidis, L. Long, Q. Ling, Neural networks and response surface polynomials for design of vehicle joints, AIAA-98-1777, 1998, pp. 653–662.

[3] N. Papila, W. Shyy, N. Fitz-Coy, R.T. Haftka, Assessment of neural net and polynomial-based techniques for aerodynamic applications, in: AIAA 17th Applied Aerodynamics Conference, AIAA-99-3167, 1999.

[4] R. Jin, W. Chen, T.W. Simpson, Comparative studies of metamodeling techniques under multiple modeling criteria, in: 8th AIAA/NASA/USAF/ISSMO Symposium on Multi-disciplinary Analysis and Optimization, Long Beach, CA, AIAA, 2000, AIAA-2000-4801.
[5] G.E.P. Box, K.B. Wilson, On the experimental attainment of optimum conditions, J. Roy. Statist. Soc. Ser. B 13 (1951) 1–45.
[6] T. Kashiwamura, M. Shiratori, Q. Yu, Statistical optimization method, Comput. Aided Optimum Des. Struct. V (1997) 213–227.
[7] I.P. Sobieski, V.M. Manning, I.M. Kroo, Response surface estimation and refinement in collaborative optimization, AIAA-98-4753, 1998, pp. 4753–4770.
[8] S. Hosder, L.T. Watson, B. Grossman, W.H. Mason, H. Kim, R.T. Haftka, S.E. Cox, Polynomial response surface approximations for the multidisciplinary design optimization of a high speed civil transport, Technical Reports of NCSTRL at Virginia Tech CS (TR-01-03), 2001.
[9] J.F.M. Barthelemy, R.T. Haftka, Approximation concepts for optimum structural design—a review, Struct. Optim. 5 (1993) 129–144.
[10] R.T. Haftka, E.P. Scott, Optimization and Experiments—A Survey, Theoretical and Applied Mechanics 1996, Elsevier Science, 1997, pp. 303–321.
[11] Q. Shi, I. Hagiwara, F. Takashima, The most probable optimal design method for global optimization, in: Proceedings of the 1999 ASME Design Engineering Technical Conferences DETC/DAC-8635, Las Vegas, Nevada, September, 1999, pp. 12–15.
[12] L. Berke, P. Hajela, Application of artificial neural nets in structural mechanics, Struct. Optim. 4 (1992) 90–98.
[13] I. Hagiwara, Q. Shi, W. Kozukue, Development of non-linear dynamic optimization method by using the holographic neural-network, 1st Report: Basic examination with a mass–spring model, Trans. JSME Ser. A 63 (616) (1997) 2510–2517 (in Japanese).
[14] I. Hagiwara, Q. Shi, F. Takashima, Function approximation method for crash optimization using neural network, 2nd Report: Application to vehicle component, Trans. JSME Ser. A 64 (626) (1998) 7–13 (in Japanese).
[15] J. Sacks, W.J. Welch, T.J. Mitchell, H.P. Wynn, Design and analysis of computer experiments, Statist. Sci. 4 (4) (1989) 409–435.
[16] A. Giunta, L.T. Watson, J. Koehler, A Comparison of approximation modeling techniques: polynomial versus interpolating models, in: 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization, St. Louis, MI, September 2–4, AIAA-98-4758-CP, 1998.
[17] D.J. Cappelleri, M.I. Frecker, T.W. Simpson, Design of a PZT bimorph actuator using a metamodel-based approach, J. Mech. Des. Trans. ASME 124 (2002) 354–357.
[18] S. Sakata, F. Ashida, M. Zako, Optimization of laminated composites for eigenfrequency problem using K-CSSL method, in: F.-K. Chang (Ed.), Proceedings of the Tenth US–Japan Conference on Composite Materials, DEStech Publishers, 2002, pp. 560–569.
[19] T.W. Simpson, T.M. Mauery, J.J. Korte, F. Mistree, Comparison of Response Surface and Kriging Models for Multidisciplinary Design Optimization, 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization, St. Louis, MI, September 2–4, AIAA-98-4755, 1998.
[20] S. Sakata, F. Ashida, M. Zako, Structural optimization using Kriging approximation, Comput. Methods Appl. Mech. Engrg. 192 (7–8) (2003) 923–939.
[21] N. Cressie, The origins of kriging, Math. Geol 22 (3) (1990) 239–252.
[22] S. Mase, J. Takeda, Spatial Data Modeling, Kyoritsu Shuppan Co. Ltd., 2001 (in Japanese).
[23] N. Cressie, Fitting variogram models by weighted least squares, Math. Geol. 17 (5) (1985) 563–586.
[24] G.H. Golub, C.F. Van loan, Matrix Computations, third ed., The Johns Hopkins University Press, 1996.
[25] R.M. Brunell, B.M. Squibb, An automatic variogram fitting procedure using Cressie's criterion, in: ASA Proceedings of the Section on Statistics and Environment, 1993, pp. 135–137.
[26] E. Cuthill, J. Mckee, Reducing the bandwidth of sparse symmetric matrices, in: Proceedings of ACM National Conference, Association for Computing Machinery, 1969, pp. 157–172.
[27] N.E. Gibbs, W.G. Poole, P.k. Stockmeyer, An algorithm for reducing the bandwidth and profile of a sparse matrix, SIAM J. Numer. Anal. 13 (2) (1976) 236–250.
[28] S. Sakata, F. Ashida, M. Zako, Applications of CSSL method to multi-peaked structural optimization problem, Theor. Appl. Mech. 51 (2002) 335–341.