876 *KSME International Journal, Vol. 15, No. 7, pp. 876~888, 2001*

Efficient Approximation Method for Constructing Quadratic Response Surface Model

Kyung-Jin Hong
Graduate Research Assistant, Department of Mechanical Design and Production Engineering, Hanyang University, Seoul 133-791, Korea
Min-Soo Kim
Contract Professor of the BK21 Division for Research and Education in Mechanical Engineering, Hanyang University, Seoul 133-791, Korea
Dong-Hoon Choi*
Director, Center of Innovative Design Optimization Technology, Hanyang University, Seoul 133-791, Korea

For a large scaled optimization based on response surface methods, an efficient quadratic approximation method is presented in the context of the trust region model management strategy. If the number of design variables is n, the proposed method requires only $2n+1$ design points for one approximation, which are a center point and two additional axial points within a systematically adjusted trust region. These design points are used to uniquely determine the main effect terms such as the linear and quadratic regression coefficients. A quasi-Newton formula then uses these linear and quadratic coefficients to progressively update the two-factor interaction effect terms as the sequential approximate optimization progresses. In order to show the numerical performance of the proposed method, a typical unconstrained optimization problem and two dynamic response optimization problems with multiple objective are solved. Finally, their optimization results compared with those of the central composite designs (CCD) or the over-determined D-optimality criterion show that the proposed method gives more efficient results than others.

Key Words : Efficient Quadratic Approximation, Response Surface, Trust Region, Approximate Optimization

1. Introduction

Recently, response surface approximations coupled with numerical optimizers were widely studied to optimize multidisciplinary systems. This approach may be effective for optimizing some high fidelity computational models such as the computational fluid dynamics (CFD), nonlinear finite element analysis (FEA), and fatigue analysis codes that do not support design sensitivity information.

In many of these studies, the early works used fixed or adaptive move limit strategies to insure that design decisions were made based on lower fidelity information (Wujek, et al., 1996; Bloebaum, et al., 1994). However, they did not guarantee the converged designs, even though these move limit strategies led to improved designs. Thus, recent works fundamentally employ the trust region model management strategy (Dennis and Torczon, 1996; Alexandrov, 1996) because it algorithmically provides the

* Corresponding Author,
E-mail : dhchoi@emal.hanyang.ac.kr
TEL : +82-2-2290-0478 ; **FAX** : +82-2-2290-1639
Director, Center of Innovative Design Optimization Technology, Hanyang University, Seoul 133-791, Korea. (Manuscript **Received** October, 4, 2000; **Revised** March 28, 2001)

convergence of the approximate optimization strategies. The trust region model management strategies adaptively restrict design moves within trust regions, where the lower fidelity approximate model produces information that agrees with the high fidelity analysis model within an acceptable tolerance to error. It still however requires a large number of design points to construct a response surface for a function that involves n variables, even though the trust region model management strategy provides a mathematical foundation on using the approximate optimization strategy.

In order to overcome these difficulties, this study proposes an efficient quadratic response surface approximation method. The proposed method has the following two merits. Firstly, it requires only $2n+1$ points for determining the regression coefficients of linear and quadratic terms in each approximation. However, it can mathematically determine the regression coefficients of two-factor interaction terms using the normalized quasi-Newton formula. Also, this progressive quadratic approximation algorithmically converges from the global quadratic approximation to the local quadratic approximation in the context of the trust region model management strategy. Secondly, it does not require the additional CPU time to explicitly construct a quadratic approximate model. This is because it uniquely determines all the regression coefficients for linear and quadratic terms and algorithmically updates the remained regression coefficients for the two-factor interaction terms using the uniquely determined terms.

Section 2 reviews the trust region algorithm and the sequential approximate optimizations with the trust region model management strategy. Section 3 fully describes the basic concept and mathematical foundation of the progressive quadratic response surface approximation (PQRSA) method. Section 4 explains the computational procedure of the sequential approximate optimization (SAO) with the PQRSA in the context of the trust region model management strategy. Section 5 shows the numerical performance by solving typical unconstrained optimization problems and comparing the results with those of other trust region methods that are combined with CCD or over-determined D-optimality designs (Carpenter, 1993). Section 6 presents the concluding remarks of this study.

2. Review of Trust Region Based Approximate Optimization

We first review the trust region algorithm that will be used to algorithmically manage the approximate model in the sequential approximate optimization process described in this study.

2.1 Basic concept of trust region algorithm

The trust region algorithm (Fletcher, 1987) approximates the quadratic function $q(\boldsymbol{\delta}_k)$, obtained by truncating the Taylor series for $f(\mathbf{x}_k+\boldsymbol{\delta}_k)$ which does not have a unique minimum and does not define Newton's method. In order to avoid this difficulty, it is assumed that some neighborhood $\Gamma^k=\{\boldsymbol{\delta}_k : \|\boldsymbol{\delta}_k\|\leq h^k\}$ of $\mathbf{x}_k$ is defined where $q(\boldsymbol{\delta}_k)$ agrees with $f(\mathbf{x}_k+\boldsymbol{\delta}_k)$ in some sense. Then it would be appropriate to choose $\mathbf{x}_{+1}=\mathbf{x}_k+\boldsymbol{\delta}_k$, where the correction $\boldsymbol{\delta}_k$

$$\text{minimizes } q(\boldsymbol{\delta}_k)=\nabla f(\mathbf{x}_k)^T\boldsymbol{\delta}_k+\frac{1}{2}\boldsymbol{\delta}_k^T\mathbf{G}_k\boldsymbol{\delta}_k, \quad (1)$$

$$\text{subject to } \|\boldsymbol{\delta}_k\|\leq h_k. \quad (2)$$

This method may be called a *Trust Region Algorithm*. In order to prevent undue restriction of the step, h^k should be as large as possible to a certain measure of agreement existing between $q(\boldsymbol{\delta}_k)$ and $f(\mathbf{x}_k+\boldsymbol{\delta}_k)$. This can be quantified by defining the actual reduction in f on the k^{th} step as

$$\Delta f^k=f(\mathbf{x}_k)-f(\mathbf{x}_k+\boldsymbol{\delta}_k) \quad (3)$$

and the corresponding predicted reduction as

$$\Delta q^k=q(\mathbf{0})-q(\boldsymbol{\delta}_k)=f(\mathbf{x}_k)-q(\boldsymbol{\delta}_k). \quad (4)$$

Then the ratio

$$r_k=\Delta f^k/\Delta q^k \quad (5)$$

measures the accuracy to which $q(\boldsymbol{\delta}_k)$ approximates $f(\mathbf{x}_k+\boldsymbol{\delta}_k)$, in the sense that the closer r_k is to unity, the better the agreement is. Most of the typical trust region algorithms adaptively reduce

or increase h^k to maintain a certain degree of agreement as measured by r_k. This is one reason that makes the trust region algorithm to be one of the best model management strategies in approximate optimization.

Although the Hessian $\mathbf{G}_k$ of Eq. (1) is replaced by an approximate Hessian $\mathbf{B}_k$, the global convergence of the trust region algorithm if $\mathbf{B}_k$ is bounded was proven by Fletcher (1972) and Powell (1975). We believe that this is another reason for the trust region concept to be newly employed for managing the approximate model in the SAO.

2.2 Sequential approximate optimization with trust region model management strategy

Since Dennis and Torczon (1996) introduced the trust region concept to manage the approximation model in optimization, most of the recent works in this area fundamentally use the trust region model management strategy (Alexandrov, 1996; Rodriguez, et al., 1988; Nelson and Papalambros, 1999). It provides a mathematical foundation upon which the use of approximate optimization strategies can continue to be developed, because it guarantees the converged design in the sequential approximate optimization (Dennis and Torczon, 1996). Also, it can easily be expanded to manage the approximate models in the constrained optimization process (Rodriguez, et al., 1988; Celis, et al., 1984), as the trust region method can be fundamentally used as a sub-optimizer for the augmented Lagrange multiplier method (Rodriguez, et al., 1988) and the sequential quadratic programming method (Celis, et al., 1984).

Although the trust region model management strategy provides a mathematical foundation on using the approximate optimization strategy, it is noted that the following two difficulties still remain when using response surface models for a large scaled system design. Firstly, it requires a large number of design points to construct a response surface for a function that involves n variables. Secondly, the use of response surface models cannot reduce the overall computational expense in the approximate optimization process. This is because the computational cost is incurred in the creation of the response surface models rather than during the optimization process (Haim, et al., 1999).

2.3 Review of experimental designs

In order to explain the relations between the number of design points and the response surface model, we consider constructing a second-order approximate model. It requires at least 3 design points for each factor so that the regression coefficients in the model can be estimated. This samples the corner points as well as the midpoints of the line connecting the corners of a hypercube. Thus 3^n full factorial experiments may be necessary. Although this design works well for a small value of n such as 2 or 3, it is difficult to use the *full factorial designs* (FFD) for the design considering a large number of design parameters.

Box and Wilson first introduced the *central composite designs* (CCD) that enabled the information surface, described by the sampled designs, to be rotatable (Box and Draper, 1987). This consists of a two-level factorial, the corner points of a hypercube, plus the center point and star points arranged along the design variable axes and symmetrically positioned with respect to the factorial hypercube. This criterion uses a total of 2^n+2n+1 points. Although the CCD needs a lower number of points than the FFD, the number of points for the CCD becomes unacceptable even for a problem with 10 design variables.

In order to reduce the number of experimental design points, the *D-optimality criterion* has been used (Box and Draper, 1971), as this could use the experimental design points equal to the number of regression coefficients in the approximate model fitted. However, it is reported that this saturated D-optimal design made poor converge of the region of interest (Carpenter, 1993). Thus the 20 % to 50 % over-determined D-optimal designs for an approximation model building is widely used because it leaves a good choice for response surface model building for the deterministic computational experiments (Unal, et al., 1998). Although this *over-determined D-optimality criterion* can use a lower number of

points than the CCD, it still requires 1.2~1.5 times the number of regression coefficients. Considering only a second-order model, the number of regression coefficients is $1+2n+n(n-1)/2$.

3. Progressive Quadratic Response Surface Approximations

The progressive quadratic response surface approximations (PQRSA) fundamentally use a quasi-Newton formula. However, it is completely different from a typical trust region algorithm with the quasi-Newton Hessian updated formula. In other words, the conventional quasi-Newton Hessian updated formula uses only local information between consecutive iterations throughout the optimization process.

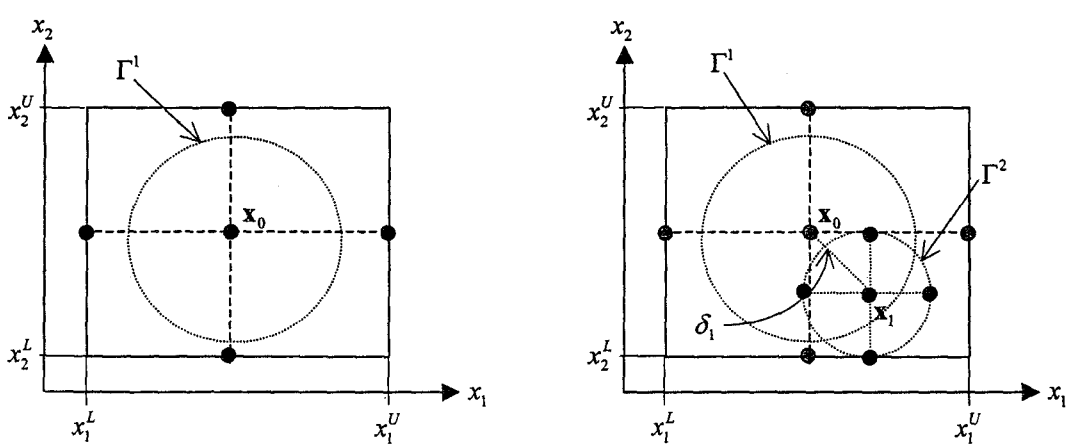
The PQRSA method is composed of the following two processes. The first one is the initial design sampling process, because a quasi-Newton formula basically needs the gradient and design change information between two design points. The second one is the quadratic approximation process using the normalized quasi-Newton formula.

3.1 Design sampling process during the first two iterations

Before we explain the proposed PQRSA, we first describe the sampling design points in the first two steps. At the initial step, the two points are sampled along the axes of each design variable and the center point is added as shown in Fig. 1 (a). Then the exact function values are evaluated for the selected $2n+1$ points. A response surface model can be uniquely developed using these $2n+1$ points because the quadratic terms for the approximate model are only diagonal components. This approximate model is optimized within Γ^1. Let the new design be $\mathbf{x}_1$ as shown in Fig. 1 (b). Now evaluate the exact function value for $\mathbf{x}_1$ and check the approximation accuracy using Eq. (5) and adjust the new trust region Γ^2 for the 2nd iteration. At the 2nd iteration, $2n$ points are newly sampled along the axes of each design variable centered at $\mathbf{x}_1$. Fig. 1 (b) shows this sampling. Then the quadratic model is constructed similar to the 1st step. It is noted that this simplified quadratic model is not directly used in the 2nd optimization iteration. That is, the PQRSA is applied from the 2nd iteration. This is explained in the next section.

3.2 Progressive quadratic response surface approximation using normalized quasi-newton formula

Let $\mathbf{g}_k$ and $\mathbf{D}_k$ be the first-order and the second-order terms of the k^{th} quadratic approximation model approximated by $2n+1$ points. Also, Let $\delta_k=\mathbf{x}_k-\mathbf{x}_{k-1}$ and $\mathbf{y}_k=\mathbf{g}_k-\mathbf{g}_{k-1}$. The approximate Hessian can then be constructed as Eq. (6) by a quasi-Newton formula (the well-known



(a) At the 1st iteration (b) At the 2nd iteration

Fig. 1 Design points sampled at the first two iterations

BFGS formula is used in this study)

$$\mathbf{B}_k=\mathbf{B}_{k-1}-\frac{(\mathbf{B}_{k-1}\boldsymbol{\delta}_k)\,(\mathbf{B}_{k-1}\boldsymbol{\delta}_k)^T}{\boldsymbol{\delta}_k^T\mathbf{B}_{k-1}\boldsymbol{\delta}_k}+\frac{\mathbf{y}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\boldsymbol{\delta}_k} \quad (6)$$

where $\mathbf{B}_k$ denotes the k^{th} approximate Hessian matrix and $\mathbf{B}_0=\mathbf{D}_0$.

Although the original second-order term ($\mathbf{D}_k$) has no off-diagonal coefficients at $\mathbf{x}_k$, it is noted that a quasi-Newton formula of Eq. (6) mathematically gives the off-diagonal terms. However, $\mathbf{D}_k$ gives more accurate information than the diagonal terms of $\mathbf{B}_k$, especially for small Γ^k. We believe that this characteristic will be gradually dominated as the optimization process is progressed. Thus, in order to directly use $\mathbf{D}_k$ as the diagonal terms of $\mathbf{B}_k$, the updated Hessian $\mathbf{B}_k$ is normalized as

$$\tilde{\mathbf{G}}_k=\mathbf{S}_k^T\mathbf{B}_k\mathbf{S}_k, \quad (7)$$

where the scaling matrix $\mathbf{S}_k$ has only the diagonal terms. The i^{th} component of $\mathbf{S}_k$, $\mathbf{S}_k^{ii}$ is conceptually $\sqrt{|D_k^{ii}|}/\sqrt{|B_k^{ii}|}$. Also, in the normalized Hessian $\tilde{\mathbf{G}}_k$, the signs of the diagonal terms are reset as those of $\mathbf{D}_k$. Unlike mathematical problems, the practical problems can be basically non-convex functional. In other words, some components of $\mathbf{D}_k$ have positive values while others have negative values making the convergence of the approximate optimization difficult. Thus, if some components of $\mathbf{D}_k$ have opposite signed values, the corresponding off-diagonal terms of $\tilde{\mathbf{G}}_k$ are reset as zero values. This represents that saddle points exist between the corresponding design variables. In other words, the approximated function can be separable between them. It is noted that this treatment enables the proposed approximation method to reduce the error for the approximating non-convex functional, although only quadratic terms are used in approximation.

Suppose that this approximation is combined with one of the trust region model management strategies. The adaptively adjusted trust regions increase the accuracy of matrix $\mathbf{D}_k$, since $\mathbf{D}_k$ and $\mathbf{g}_k$ can be equal to those of central difference approximation in small or well-established trust regions. This represents that the normalized matrix $\tilde{\mathbf{G}}_k$ can be nearly equal to the exact Hessian matrix for $\mathbf{x}_k$. Finally, the normalized Hessian matrix $\tilde{\mathbf{G}}_k$ modify Eq. (6) as

$$\mathbf{B}_k=\tilde{\mathbf{G}}_{k-1}-\frac{(\tilde{\mathbf{G}}_{k-1}\boldsymbol{\delta}_k)\,(\tilde{\mathbf{G}}_{k-1}\boldsymbol{\delta}_k)^T}{\boldsymbol{\delta}_k^T\tilde{\mathbf{G}}_{k-1}\boldsymbol{\delta}_k}+\frac{\mathbf{y}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\boldsymbol{\delta}_k} \quad (8)$$

We call Eq. (8) a *normalized quasi-Newton formula.* In section 4, we will fully describe the computational procedure of a sequential approximate optimization method combined with the PQRSA.

3.3 Hyper-cubic typed trust region for effective treating side constraints

As the trust region algorithm is fundamentally developed only to solve unconstrained optimization problems without side constraints, the original hyper-spherical type trust region, shown in Fig. 1, may not be appropriate for engineering optimization problems. Thus we recommend that one uses a hyper-cubic type trust region, shown Fig. 2, since most of the engineering optimization problems have hyper-cubic typed side constraints.

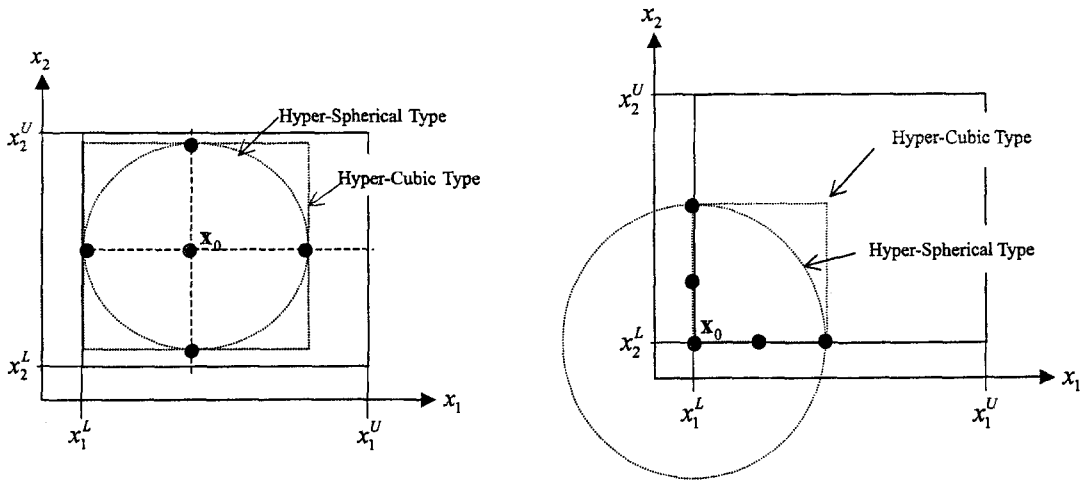
Now we turn to explain the effectiveness of the hyper-cubic type trust region. If an iterative design (including the given initial design) is inactive to all the side constraints shown in Fig. 2 (a), both trust regions can successfully represent a sub-region of design space. However, if an iterative design is active to some side constraints, shown in Fig. 2 (b), the hyper-cubic type trust region easily represents a sub-region, including the iterative design, within side constraints but the hyper-spherical type trust region does not.

Also, suppose that the optimum design is active to some side constraints. Then the hyper-spherical type trust region should be infinitely small in order to include the optimum point within it. However, the hyper-cubic type trust region can easily include the optimum point without any difficulties.

4. Computational Procedure of SAO Combined with PQRSA

4.1 Computational procedure

This section presents the computational procedure of a Sequential Approximate Optimization



(a) In case that a nominal design is inactive to all side constraints

(b) In case that a nominal design is active to some side constraints

Fig. 2 Design point sampling strategy according to the activeness of side constraints

(SAO) method, which uses a trust region model management strategy for the effective managing of the progressively approximated quadratic models. The basic computational procedure is as follows:

Step 0. The initial trust region is assumed 30% ~50% for design space that includes the initial design. Set $k=1$. If one does not give the initial design, select the initial $2n+1$ sampling points shown in Fig. 1. Otherwise, select the initial designs shown in Fig. 2.

Step 1. Construct a quadratic model using a quadratic polynomial approximation along each design variable axes. Let their coefficients be $\mathbf{g}_k$ and $\mathbf{D}_k$ for linear and quadratic terms. If $k=1$ Then go to *Step 3*. Otherwise, set $\mathbf{y}_k=\mathbf{g}_k-\mathbf{g}_{k-1}$.

Step 2. Evaluate the coefficient matrix of the second-order term ($\mathbf{B}_k$) using Eq. (8) and construct $\tilde{\mathbf{G}}_k$ by normalizing $\mathbf{S}_k^T\mathbf{B}_k\mathbf{S}_k$.

Step 3. Minimize $q(\boldsymbol{\delta}_k)=\mathbf{g}_k^T\boldsymbol{\delta}_k+\frac{1}{2}\boldsymbol{\delta}_k^T\tilde{\mathbf{G}}_k\boldsymbol{\delta}_k$ subject to $\|\boldsymbol{\delta}_k\|_\infty\leq h_k$. Let $\boldsymbol{\delta}_k^*$ be the approximate optimum. For effective solving of this sub-problem, more detailed discussion is presented in section 4.2.

Step 4. Evaluate the actual and the predicted reductions such as Δf^k and Δq^k using Eq. (3) and Eq. (4). Then check the reduction ratio r_k. If $r_k<0.25$ set $h_k=\|\boldsymbol{\delta}_k\|_\infty/4$, if $r_k>0.75$ and $\|\boldsymbol{\delta}_k\|_\infty=h_k$, then set $h_k=2h_k$, otherwise set $h_k=h_k$. If $r_k>0$, then update $\mathbf{x}_k=\mathbf{x}_k+\boldsymbol{\delta}_k$ and go to *Step 5* with $k=k+1$. Otherwise, go to *Step 3* with the reduced trust region.

Step 5. If $\Delta f^k\leq\varepsilon_1$, $\Delta f^k/f^k\leq\varepsilon_1$ or $\|\mathbf{g}_k\|\leq\varepsilon_2$ then stop. Otherwise, select new $2n$ design points around $\mathbf{x}_{k+1}$, within the new trust region Γ^{k+1} and go to *Step 1*.

The constants 0.25, 0.75, etc. have been used in many trust region algorithms. They are also known to be quite insensitive to their change. Thus the values are not changed in this study.

4.2 Numerical considerations

Although the basic procedure for the proposed approximate optimization is described in section 4.1, the following four guidelines are needed to use the above computational procedure in solving the approximate optimization problems.

First, in order to solve the approximate optimization problem of *Step 3*, at least two optimization algorithms are recommended. For $\tilde{\mathbf{G}}_k$ being positive definite, a Newton's method (Fletcher, 1987) is used in this study. However, for $\tilde{\mathbf{G}}_k$ being non-positive definite, a global optimization algorithm is used. A Simulated Annealing (SA) method (Corana, et al., 1987) is employed for the latter.

Second, for multi-criteria optimization, it is recommended that each objective function be

respectively approximated. Then, in order to transform the multi-objective function into a scalar function, the max-value typed preference function (Osyzka, 1984) is used as

$$\tilde{P}(\boldsymbol{\delta}_k)=\max_{i=1,\cdots,m}\left\{w_i\frac{\tilde{f}_i(\boldsymbol{\delta}_k)-f_i^*}{f_i^*}\right\} \text{ or}$$

$$\max_{i=1,\cdots,m}\{w_i\tilde{f}_i(\boldsymbol{\delta}_k)\} \text{ for } \sum_{i=1}^{m}w_i=1 \quad (9)$$

where w_i is the i^{th} weighting coefficient, and f_i^* the ideal value for the i^{th} objective. Also, the i^{th} approximated objective function is defined as

$$\tilde{f}_i(\boldsymbol{\delta}_k)=f_i(\mathbf{x}_0)+(\mathbf{g}_k^i)^T\boldsymbol{\delta}_k+\frac{1}{2}\boldsymbol{\delta}_k^T\tilde{\mathbf{G}}_k^i\boldsymbol{\delta}_k \quad (10)$$

Then, Eq. (9) is used as $q(\boldsymbol{\delta}_k)$ at *Step 3* in the algorithm above. In addition, the actual and predicted reductions are evaluated by the proposed preference function.

Third, for considering constrained and approximate optimization problems, it is recommended that the objective and constraint functions be respectively approximated. Then, any constrained optimization algorithm is used to solve this approximate optimization problem. However, this study uses the L_1 exact penalty function (Fletcher, 1972) to evaluate the reduction ratio for the constrained optimization problem at *Step 4* in the above computational procedure. This is because this functional is exact in the sense that local minimums of the functional are equivalent to local minimums of the original problem to a large extent. The L_1 exact penalty function is represented as:

$$\tilde{L}_1(\boldsymbol{\delta}_k)=\tilde{f}(\boldsymbol{\delta}_k)+r\sum_{j=1}^{Nieq}\max\{0,\ \tilde{g}_j(\boldsymbol{\delta}_k)\} \quad (11)$$

where $\tilde{f}$ and $\tilde{g}_j$ denote the objective function and the j^{th} inequality constraint function that are approximated. This approximated L_1 exact penalty function of Eq. (11) is directly solved by the SA algorithm. This is in order to solve the constrained approximate optimization problems.

Finally, for evaluating the reduction ratio r_k of *Step 4* in the above computational procedure, using the following modified formula is recommended in order to avoid additionally introducing the heuristic limitation of $\|\boldsymbol{\delta}_k\|_\infty$ for $r_k>1$.

$$r_k=\min\left\{\frac{\Delta f^k}{\Delta q^k},\ \frac{\Delta q^k}{\Delta f^k}\right\} \quad (12)$$

Suppose that $\Delta f^k/\Delta q^k=10$, Eq. (5) gives $r_k=10$ but Eq. (12) gives $r_k=\min\{10, 0.1\}=0.1$. In other words, according to *Step 4* in the computational procedure of Sec. 4.1, the original formula may increase or maintain the size of trust region because $r_k>0.75$. However, the modified formula reduces the trust region because $r_k<0.25$. We believe that this modified formula is a better measure to the approximation accuracy because the closer r_k being to unity represents a better agreement between f and q.

5. Numerical Studies

In order to show the numerical performance of the proposed approximation method, a sequential approximate optimizer (SAO) is developed based on the computational procedures described in Sect. 4. This fundamentally employs a trust region model management strategy and has the option of three approximation methods such as the PQRSA, the CCD and the over-determined D-optimality criterion. The employed convergence criteria are that: 1) the relative or absolute deviation of cost functions between consecutive SAO iterations, and 2) the maximum violation amount of constraints, should be less than ε_1 and ε_2, respectively.

In this section, two typed sample problems are studied. One is an unconstrained mathematical optimization problem. Another is a multi-objective dynamic response optimization problem from the reference (Haug and Arora, 1979).

5.1 Unconstrained mathematical optimization problems

This sample problem is selected to compare the numerical performance of the three approximation methods according to the increasing number of design variables. The problem is to minimize

$$f(\mathbf{x})=\sum_{k=1}^{n-1}(x_{k+1}-x_k^2)+(1-x_k)^2,$$

where the initial design is given as $x_k=-1.2$, $x_{k+1}=1.0$ for $k=1, 3, 5, \cdots, n-1$. We solve these sample problems for $n=2$, 6, 10, 15, 30 and 50. Table 1 lists the number of design points required

Table 1 The number of design points required for the three approximation methods to construct a quadratic response surface model

n	PQRSA	D-Optimality*	CCD
2	5	9	9
6	13	42	77
10	21	99	1045
15	31	204	32799
30	61	744	-
50	101	2837	-

* The 50% Over-determined D-Optimality Criterion is used.

for the three approximation methods. As can be seen, it seems that the CCD are not recommended for a larger number ($n \geq 6$), and the over-determined D-Optimality criterion not for $n \geq 15$ as well, since they require too many numbers of design points. The convergence tolerance ε_1 is specified as 1×10^{-3} because it is an unconstrained problem. The optimization results are listed in Table 2. In Table 2, Iteration represents the number of approximations, and the Total NFE denotes the cumulative number of function evaluations during SAO. In this study, Total NFE represents total number of sampling points plus additional points to check convergence during the optimization process. The Total NFE is counted by *Iteration**$(2n+1)$ for the PQRSA, *Iteration**$(1+1.5*(1+2n+n(n-1)/2))$ for the 50 % over determined D-optimality criterion, and *Iteration**$(1+2^n+2n+1)$ for the CCD. It is noted that the D-optimality criterion and the CCD require one additional evaluation, because the approximate optimum may not be included in the next design set.

Table 2 lists four iteration histories of the cost value for the three approximation methods. The CCD and the 50 % over determined D-optimality criterion are not applied for $n > 6$ and for $n > 15$, respectively, because it requires too many design points. All the tested results show that the PQRSA method is more efficient than the other two approximation methods, although the number of SAO is greater than the other methods. Fig. 3 graphically shows the Total NFE of three approximation methods according to the number of design variables. It is especially noted that the

Table 2 Iteration history for the unconstrained mathematical optimization problem

(a) In case of having 2 design variables

Iteration	PQRSA	D-optimality	CCD
1	3.7055	2.8901	2.8901
2	5.1145*	1.0000	1.4416
3	2.1043	0.2876	0.5489
4	1.2254	0.6759*	0.1086
5	0.1609	0.0704	0.0088
6	0.1274	0.0211	0.0010
7	0.0654	0.0179	
8	0.0138	0.0039	
9	0.0032	0.0015	
10	0.0012		
Total NFE	51	91	61

* The SAO is failed to improve design. Thus, this design is automatically rejected and a new approximation is used within the reduced trust region.

(b) In case of having 6 design variables

Iteration	PQRSA	D-optimailty	CCD
1	9.4930	11.2904	11.1763
2	13.8118*	2.9348	5.5606
3	4.8713	21.9377*	2.3966
4	2.7191	0.7227	0.3721
5	0.9933	0.2044	0.1130
6	0.6142	0.0825	0.0541
7	0.1459	0.0242	0.0269
8	0.0610	0.0064	0.0096
9	0.0492	0.0016	0.0012
10	0.0181		
11	0.0047		
12	0.0012		
Total NFE	167	388	703

* The SAO is failed to improve design. Thus, this design is automatically rejected and a new approximation is used within the reduced trust region.

(c) In case of having 10 and 15 design variables

Iteration	10 design variable		15 design variables	
	PQRSA	D-optimality	PQRSA	D-optimality
1	14.9794	27.4430	19.3918	6.6761
2	18.1220*	2.2691	25.8837*	6.6761
3	7.7918	3.4559*	5.8421	5.7377*
4	3.9710	0.9480	5.8315	0.4853
5	1.3988	0.1376	4.2636	0.1323
6	1.2832	0.7361*	2.7152	0.0535
7	0.9660	0.0518	1.6224	0.0108
8	0.7224	0.0453	1.0172	0.0075
9	0.5104	0.0028	0.3544	0.0007
10	0.3425	0.0023	0.1409	
11	0.2198		0.0461	
12	0.1011		0.0181	
13	0.0313		0.0087	
14	0.0121		0.0040	
15	0.0038		0.0017	
16	0.0012			
Total NFE	337	1001	466	1846

* The SAO is failed to improve design. Thus, this design is automatically rejected and a new approximation is used within the reduced trust region.

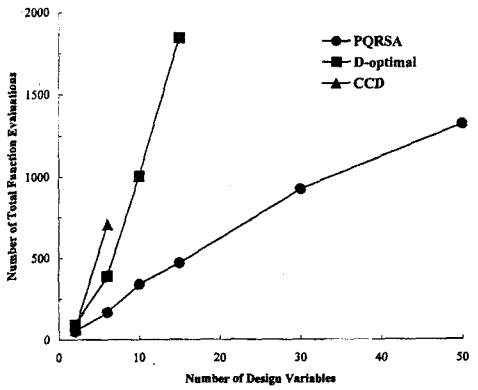


Fig. 3 Comparison of the total function evaluations for three approximation methods according to the number of design variables

Total NFE of the PQRSA for $n=50$ requires much less function evaluations than those of the 50 % over determined D-optimality criterion for $n=15$.

5.2 Two multi-objective dynamic response optimization problems

These problems fundamentally have time dependent constraints and multi-objectives that consist of several max-value functions over time intervals. Thus, we remove their time dependencies by replacing all the time dependent constraints with max-value functions respectively (Grandhi, et al., 1986). For direct comparison of the optimization results with those of Haug and Arora, the preference function is used as $\max_{i=1,\cdots,m}\{\tilde{f}_i(\delta_k)\}$. The convergence tolerances ε_1 and ε_2 are specified as 1×10^{-2} and 1×10^{-4}, respectively.

5.2.1 A linear dynamic absorber design with five objective functions

The objective is to find the damping and spring constants of a dynamic absorber shown in Fig. 4. This minimizes the peak transient dynamic responses of the main mass for a given set of excitation frequency ratios, subject to constraints on transient and steady state responses to a given five forcing functions. The starting values of design variables are (1.0, 0.2) and the corresponding initial cost values are (8.65, 8.07, 3.20, 5.21, 9.41). Haug and Arora (1979) obtained the optimum designs (0.921, 0.154) and corresponding maximum cost values (4.291) after 22 iterations. They evaluated function and gradient evaluation per iteration. The iteration histories of the cost value are listed side by side for the three approximation methods in Table 3.

Table 3 Optimization results of a two DOF linear dynamic absorber design with five objective functions

Iteration	PQRSA	D-optimality	CCD
1	5.5695	4.5904	5.2053
2	5.7665*	4.9940*	4.3858
3	4.3321	4.2953	4.8727*
4	4.2993	4.2949	4.2948
5	4.2995*		4.2947
6	4.2980		
Total NFE	31	41	51

* The SAO is failed to improve design. Thus, this design is automatically rejected and a new approximation is used within the reduced trust region.

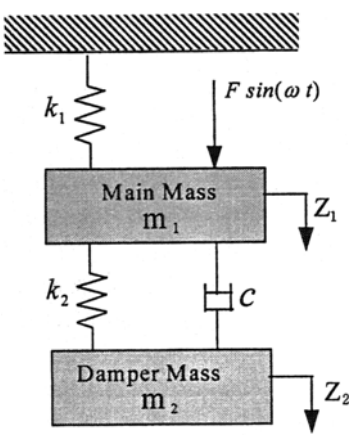


Fig. 4 Linear vibration isolator

As shown in Table 3, all the approximation methods successfully converged. In this problem, the PQRSA require more iterations than the other two methods like those of the unconstrained optimization problem. However, in the PQRSA, more iteration does not simply represent increasing additional CPU time. This is because it does not require additional CPU time to construct the approximate model, unlike the CCD or the D-optimality criterion. Figure 5 shows the convergence history of the PQRSA method, which does not include the rejected design during SAO.

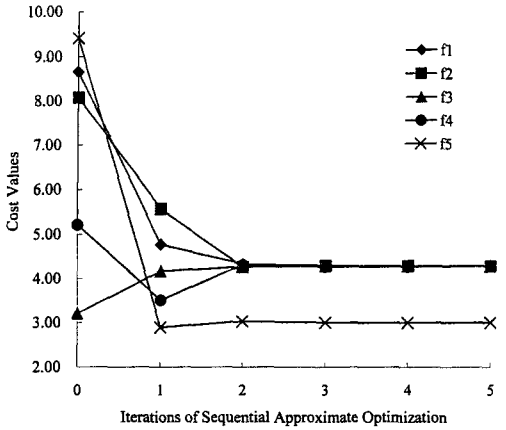


Fig. 5 Convergence history of the PQRSA for the linear vibration isolator excited by five forcing functions

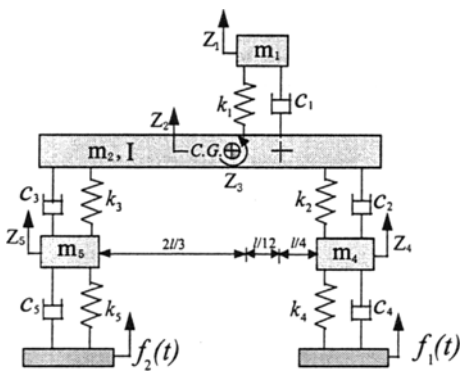


Fig. 6 Five-degree-of-freedom Vehicle system

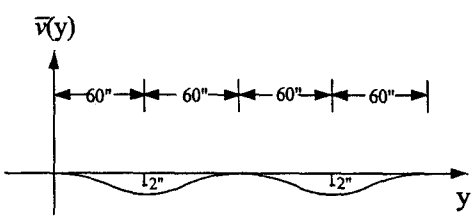
5.2.2 A vehicle suspension system design with two objective functions

The five degree-of-freedom vehicle suspension system of Fig. 6 is designed to minimize the extreme acceleration of the driver's seat for a constant vehicle speed and the two road conditions. The motion of the vehicle is constrained so that the relative displacements between the chassis and the driver's seat, the chassis and the front and rear wheels, and the road surface and front and rear wheels, are within given limits. Spring constants and damping coefficients in the system are chosen as design parameters. The profiles 2 and 3 of road conditions of Fig. 7 are used for determining the displacement functions. It is a combination of two sinusoidal curves with constant half-wave lengths. The starting values of design variables are (100, 300, 300, 10, 25, 25) and the corresponding initial cost values are (198.6, 142.4). Haug and Arora (1979) obtained the optimum designs (50, 200, 200, 8.93, 45.92, 37.81) and corresponding maximum cost value (125.5) after 40 iterations. The iteration histories of the cost value are listed side by side for the three approximation methods in Table 4.

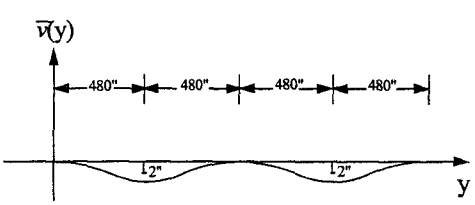
Table 4 Optimization results of a vice DOF vehicle suspension system with two objectives

Iteration	PQRSA	D-optimality	CCD
1	214.7961	137.4375	142.4214
2	130.3887	144.9298*	142.7022*
3	134.5938*	131.2935	128.6760
4	127.4947	129.5541	130.0800*
5	130.5330*	129.0609	129.5109
6	126.7498	127.6085	127.8777
7	157.2262*	126.9647	127.9362*
8	126.3163	126.5974	127.7061
9	126.2330		
Total NFE	118	345	625

* The SAO is failed to improve design. Thus, this design is automatically rejected and a new approximation is used within the reduced trust region.



(a) Profile 2



(b) Profile 3

Fig. 7 Two Road surface profiles

Table 4 shows that all the three methods are successfully converged. As can be seen, the PQRSA gives more efficient result than the other two methods. Particularly for the small design, such as the above linear dynamic absorber design

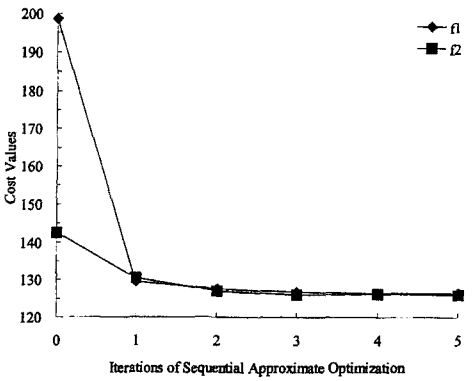


Fig. 8 Convergence history of the PQRSA for the vehicle suspension system excited by profiles 2 and 3

problem, the Total NFE seems to be competitive in the tested three methods. However, for the medium scaled design such as this example, the Total NFE of the PQRSA is found to be only 33 % of those of the D-optimality criterion and 19 % of those with the CCD. These two iteration histories for constrained multi-objective optimization problems show the similar trend to those shown in Fig. 3 for unconstrained optimization problems. That is, the PQRSA gives more efficient results than the other two approximation methods for larger design problems shown in Fig. 3.

Figure 8 shows the convergence history of the PQRSA, which does not include designs rejected during SAO. Two cost functions are consistently minimized and have the same magnitudes at the final design.

6. Concluding Remarks

An effective quadratic response surface approximation method has been proposed in this study. The proposed method does not require the additional CPU time to explicitly construct an approximate model. This approximation method used only $2n+1$ design points for one approximate optimization, and uniquely determined the main effect terms such as linear and quadratic regression coefficients. The two-interaction terms were then progressively updated using main effect terms between consecutive iterations. To do this, the study employed a normalized quasi-Newton update formula. This whole procedure was called a progressive quadratic response surface approximation (PQRSA) method. To implement computational procedures of the PQRSA in the context of the trust region model management strategy, a SAO program having an option of three approximation methods, such as the PQRSA, the 50 % over determined D-optimality criterion and the CCD, was developed and applied to a typical mathematical unconstrained optimization problem consisting of five design cases and two multi-objective dynamic response optimization problems.

For all design cases, the proposed PQRSA gave more efficient results than the other two approximation methods. In the unconstrained mathematical optimization problems, it was especially noted that the Total NFE of PQRSA for the case having $n=50$ was much less than those of the 50 % over -determined D-optimality criterion for $n=15$. Also, for multi-objective dynamic response optimization problems, the proposed PQRSA gave more efficient results than the other two approximation methods. Particularly for the medium scaled design such as this example, the Total NFE of the PQRSA is found to be only 33 % of those of the D-optimality criterion and 19 % of those with the CCD. These results showed that the PQRSA can be effective for optimizing large-scale design problems combined with the CFD, nonlinear FEA, and fatigue analysis codes that do not support design sensitivity information.

The performance of the proposed PQRSA was evaluated by comparing it's optimization results for three example problems with those of two approximation methods combined with two typical experimental designs. The comparison clearly shows the great superiority of the proposed approximation method over those of conventional approaches, which verifies that the PQRSA is an effective and efficient quadratic response surface modeling.

Acknowledgement

This work was supported by the BK21 projects,

division for research and education in mechanical engineering of the Korean ministry of education.

References

Alexandrov, N., 1996, "Robustness Properties of a Trust Region Frame Work for Managing Approximations in Engineering Optimization," *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium*, AIAA 96-4102-CP, Bellevue, Washington, September 7-9, pp. 1056~1059.

Bloebaum, C. L., Hong, W. and Peck, A., 1994, "Improved Move Limit Strategy for Approximate Optimization," *Proceedings of the 5th AIAA/USAF/NASA/ISSMO Symposium*, AIAA 94-4337-CP, Panama City, Florida, September 7-9, pp. 843~850.

Box, G. E. and Draper, N. R., 1987, *Empirical Model Building and Response Surfaces*, John Wiley, New York.

Box, M. J. and Draper, N. R., 1971, "Factorial Designs, the $|X'X|$ Criterion, and Some Related Matters," *Technometrics*, Vol. 13, No. 4, pp. 731~742.

Carpenter, W. C., 1993, "Effect of Design Selection on Response Surface Performance," *Contractor Report 4520*, NASA, June.

Celis, M. R., Dennis, J. E. and Tapia, R. A., 1985, "A Trust Region Strategy for Nonlinear Equality Constrained Optimization," *Numerical optimization 1984* (Boggs PT, Byrd RH and Schnabel RB. Eds), SIAM, Philadelpia, pp. 71~88.

Corana, et al., 1987, "Minimizing Multimodal Functions of Continuous Variables with the Simulated Annealing Algorithm," *ACM Transaction on Mathematical Software*, Vol. 13, No. 3, pp. 262~280.

Dennis, J. E. and Torczon, T., 1996, "Approximation Model Management for Optimization," *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium*, AIAA 96-4046, Bellevue, Washington, September 7-9, pp. 1044~1046.

Fletcher, R., 1972, "An Algorithm for Solving Linearly Constrained Optimization Problems," *Math. Prog.*, Vol. 2, pp. 133~165.

Fletcher, R., 1987, *Practical Method of Optimization*, John Wiley & Sons: Chichester.

Grandhi, R. V., Haftka, R. T. and Watson, L. T., 1986, "Design-Oriented identification of Critical Times in Transient Response," *AIAA Journal*, Vol. 24, No. 4, pp. 649~656.

Haim, D., Giunta, A. A., Holzwarth, M. M., Mason, W. H., Watson, L. T. and Haftka, R. T., 1999, "Comparison of optimization Software Packages for an Aircraft Multidisciplinary Design Optimization problem," *Design Optimization: international Journal for product & Process Improvement*, Vol. 1, No. 1, pp. 9~23.

Haug, E. J. and Arora, J. S., 1979, *Applied optimal Design*, Wiley-Interscience, New York, pp. 341~352.

Nelson, II. S. A. and Papalambros, P. Y., 1999, "The Use of Trust Region Algorithms to Exploit Discrepancies in Function Computation Time Within Optimization Models," *ASME Journal of Mechanical Design*, Vol. 121, pp. 552~556.

Osyzka, A., 1984, *Multicriterion optimization in Engineering with Fortran programs*. Ellis Horwood: Chichester, pp. 31~39.

Powell, M. J. D., 1975, "Convergence Properties of a Class of Minimization Algorithms," *Nonlinear Programming 2* (Mangasarian OL, Meyer RR and Robinson SM Eds), Academic Press, New York.

Powell, M. J. D., 1978, "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations," *Numerical Analysis Proceedings Dundee 1977* (Watson GA. Eds), Springer-Verlag: Berlin, pp. 144~157.

Rodriguez, J. F., Renaud, J. E. and Watson, L. T., 1988, "Trust Region Augmented Lagrangian methods for Sequential Response Surface Approximation and Optimization," *ASME Journal of Mechanical Design*, Vol. 120, pp. 58~66.

Unal, R., Lepsch, R. A. and McMilin, M. L., 1988, "Response Surface Model Building and Multidisciplinary Optimization Using D-Optimal Designs," *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, AIAA-98-4759, St. Louis, Missouri, September 2-4, pp. 405~411.

Wujek, B., Renaud, J. E., Batill, S. M. and Brockman, J. B., 1996, "Concurrent Subspace Optimization Using Design Variable Sharing in a Distributed Computing Environment," *Concurrent Engineering: Research and Applications* (*CERA*), Technomic Publishing Company Inc: December.