Journal of Marine Science and Application, Vol.6, No.3, September 2007, pp. 24-32

DOI: 10.1007/s11804-007-7012-6

Application of the optimal Latin hypercube design and radial basis function network to collaborative optimization

ZHAO Min [1] **and CUI Wei-cheng**[2]

1. State Key Laboratory of Ocean Engineering, Shanghai Jiao Tong University, Shanghai 200030, China
2. China Ship Scientific Research Center, P.O.Box 116, Wuxi 214082, China

Abstract: Improving the efficiency of ship optimization is crucial for modern ship design. Compared with traditional methods, multidisciplinary design optimization (MDO) is a more promising approach. For this reason, Collaborative Optimization (CO) is discussed and analyzed in this paper. As one of the most frequently applied MDO methods, CO promotes autonomy of disciplines while providing a coordinating mechanism guaranteeing progress toward an optimum and maintaining interdisciplinary compatibility. However, there are some difficulties in applying the conventional CO method, such as difficulties in choosing an initial point and tremendous computational requirements. For the purpose of overcoming these problems, optimal Latin hypercube design and Radial basis function network were applied to CO. Optimal Latin hypercube design is a modified Latin Hypercube design. Radial basis function network approximates the optimization model, and is updated during the optimization process to improve accuracy. It is shown by examples that the computing efficiency and robustness of this CO method are higher than with the conventional CO method.

Keywords: multidisciplinary design optimization (MDO); collaborative optimization (CO); optimal Latin hypercube design; radial basis function network; approximation

CLC number: U662 **Document code:** A **Article ID:** 1671-9433(2007)03-0024-09

1 Introduction

Modern engineering design problems are complex and involve multiple disciplines. We can benefit much from decomposition of the large complex problem into smaller tasks, which can be carried out by multiple teams in parallel. Therefore, more and more engineering designs are decomposed into different disciplinary tasks in order to exploit the computational benefits that usually arise from concurrent execution of analyses. However, during the development of this theory, it is found that smaller subtasks are hard to be solved independently since they are coupled. Subsequently, the multidisciplinary design optimization (MDO) [1] is brought into engineering design field to solve such large coupled systems.

MDO was originally developed in the design of aircraft. With the rapid growth of MDO over the past decade, it has been also discussed and used in the field of ship design. The design of a ship is a complex, multidisciplinary process which is characterized by thousands of design variables, multi-objectives and nonlinear constraints. A complete design requires analyses of hydrodynamics, structural mechanics, propulsion, performance, cost and others, some disciplinary are coupled during the process of ship design. Therefore, how to efficiently analyze and optimally design a ship is the key point. As MDO is developed for the system engineering governed by multiple coupled disciplines or made up of coupled components, it is a good choice for the ship design instead of traditional approach, which is a sequential order. However, after the past decade's development of MDO, MDO method is not totally mature and still developing, especially for Collaborative Optimization (CO), which is one of the most frequently applied multidisciplinary design optimization methods.

In this paper, the CO is analyzed and discussed, and the design of experiment (DOE) and global approximation are applied to improve CO. The methods of DOE and approximation are the optimal Latin hypercube design and the radial basis function network respectively. It is proved to be effective and robust through one mathematical example and one

Received date: 2007-04-02.

engineering example.

2 Implementation features of this method

For engineering optimization problems, the choice of initial point is very important for optimization. DOE can analyze a design space and provide a rough estimate of an optimal design, which can be used as a starting point for numerical optimization. More important, the optimal Latin hypercube design could cover the design space more evenly than other DOE methods, and generate more evenly distributed points. Therefore, in this paper, the optimal Latin hypercube design is adopted to find the initial point and create the database for approximation model. Besides, engineering optimization problems often need tremendous computation time for several programs running at the same time, such as Fluent and Nastran. We cannot afford to execute so large scale of exact MDO analyses to provide the evaluation of the objective function and constraints. The application of approximation is a necessity and in this paper, the radial basis function network is adopted due to its robustness.

2.1 Collaborative optimization

CO, one of the most frequently applied multidisciplinary design optimization methods, was thereafter developed to promote autonomy while providing a coordinating mechanism guaranteeing the progress toward an optimum and maintaining interdisciplinary compatibility (Braun, 1996; Kroo and Manning, 2000). It basically consists of a bi-level optimization structure, which is shown in Fig. 1. The task of the disciplinary-level teams to find a local design that satisfies local constraints and comes as close to that specified by system-level optimizer as possible. The task of the system is to provide a coordinating mechanism, which adjusts the target value to ensure the optimization progress towards the optimum and compatibility between the disciplinary-level designs. The significant advantage of this method is that its architecture is much like the plan of modern practical system engineering design, so CO has been widely studied and applied to practical engineering problems, such as launch vehicle design [2], undersea vehicles design [3] and conceptual ship design [4].

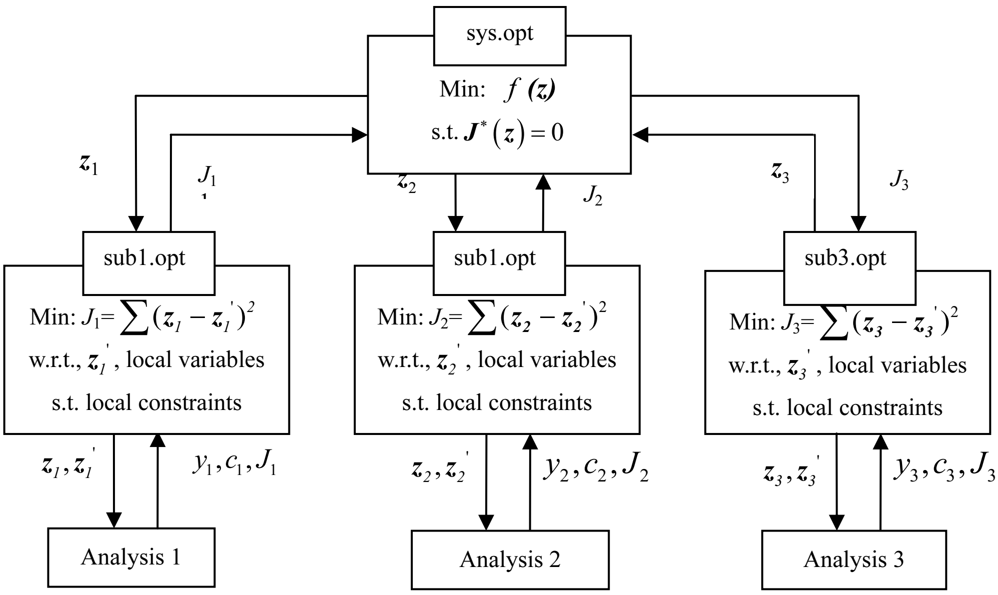
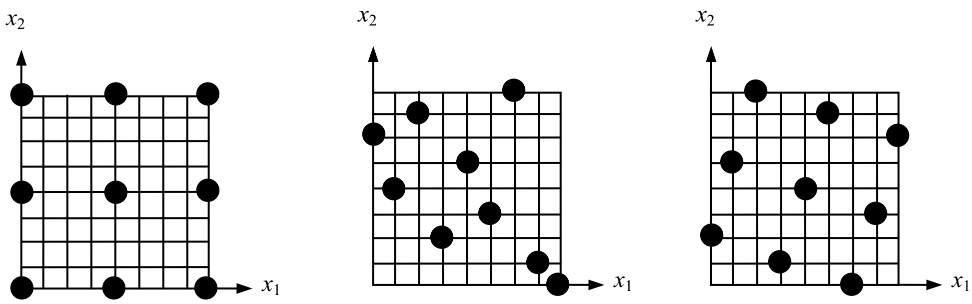


Fig. 1 Basic structure of collaborative optimization

However, there are still some problems in this immature method. First, because the subproblem optimization may involve substantial analysis, it will cause tremendous execution of disciplinary analyses. It makes the system optimizer endure heavy burden of large amount of computation. Secondly, Collaborative Optimization is sensitive to the initial point. When the initial point is not so close, it will lead to the slow rate of convergence and even non-convergence. Therefore, many researchers have focused on extension or modification to CO aiming at improving the overall efficiency. Sobieski et al. [5] proposed the use of response surface estimation in place of the disciplinary optimization in CO, and suggested two approaches to estimate the disciplinary optimal results. But this method adopts just a local approximation and is not efficient. To resolve the convergence problem of CO, Kroo and Manning [6] adopted the direct search method such as Hooke and Jeeves method, or the probabilistic search method such as genetic algorithm instead of the gradient-based method. DeMiguel and Murray[7] proposed a Modified Collaborative Optimization (MCO) at Stanford University in 1998. Han et al.[8] proposed an Improved Collaborative Optimization (ICO) at Beijing University of Aeronautics and Astronautics in 2006. But these methods are difficult to determine the penalty factor.

2.2 Introduction to optimal Latin hypercube design

Optimal Latin hypercube design [9], a modified Latin Hypercube design, is a kind of techniques in the design of experiment, in which the combination of factor levels for each factor is optimized, rather than randomly uniformly divided (the same number of divisions (n) for all factors). These levels are then randomly combined to generate a random Latin Hypercube as the initial DOE design matrix with n points (each level of a factor studies only once). An optimization process is then applied to this initial random latin hypercube design matrix. By swapping the order of two factor levels in a column of the matrix, a new matrix is generated and the new overall spacing of points is evaluated. The goal of this optimization process is to design a matrix, in which the points are spread as evenly as possible within the design space defined by the lower and upper level of each factor. The optimal Latin hypercube design concept is illustrated in Fig.2 for a configuration with two factors (x_1, x_2) and 9 design points. In Fig.2(a), a standard three level orthogonal array is shown. While this matrix has nine design points, there are only three levels for each factor. Consequently, a quadratic model could be fit to this data, but it is not possible to determine if the actual functional relationship between the response and these two factors is more nonlinear than quadratic. Fig.2(b) shows a random latin hypercube. This matrix also includes nine design points for the two factors, but there are nine levels for each factor as well, allowing higher order polynomial models to be fit to the data and greater assessment of nonlinearity. However, the design points in Fig.2(b) are not spread evenly within the design space. For example, there is little data in the upper right and lower left corners of the design space. An optimal Latin hypercube design matrix is displayed in Fig.2(c). With this matrix, the nine design points cover nine levels of each factor and are spread evenly within the design space. For example, there is little data in the upper right and lower left corners of the design space. An optimal Latin hypercube design matrix is displayed in Fig.2(c). With this matrix, the nine design points cover nine levels of each factor and are spread evenly within the design space. For cases where one purpose of executing the design of experiment is to fit an approximation model to the resulting data, the optimal Latin hypercube design gives the best opportunity to model the true function or true behavior of the response across the range of the factors.



(a) 3-level orthogonal array (b) Random Latin hypercube (c) Optimal Latin hypercube

Fig. 2 Optimal Latin hypercube design configuration for two factors, with nine points

Optimal Latin hypercube design ensures the sample points distributing more evenly than other DOE techniques. By making use of this characteristic, the more evenly distributed sample points are gained.

2.3 Introduction to radial basis function network

One of the most general uses of the approximation functions is the artificial neural network. Usually, we suppose the existence of a relation between several input variables and one output variable. After the learning of artificial neural networks, an approximator between these inputs and this output can be built to represent this unknown relation. Radial basis function (RBF) network is a type of neural network layer of linear units, and is characterized by reasonably fastening the training and reasonably compacting networks. A traditional radial basis function network is shown in Fig.3. It contains three layers, including the inputs, the hidden layer and the output node(s). Each component of the input units x feeds forward to m hidden units, whose outputs are linearly combined with weights $\{w_j\}_{j-1}^{m}$ into the network output units y, where each hidden unit represents a single radial basis function ϕ. A set of RBFs can serve as a basis for representing a wide class of functions that are expressible as linear combinations of the chosen RBFs:

$$y(\boldsymbol{x}) = \sum_{j=1}^{M} \boldsymbol{w}_j \bullet \boldsymbol{\phi}\left(\left\|\boldsymbol{x} - \mu_j\right\|\right). \tag{1}$$

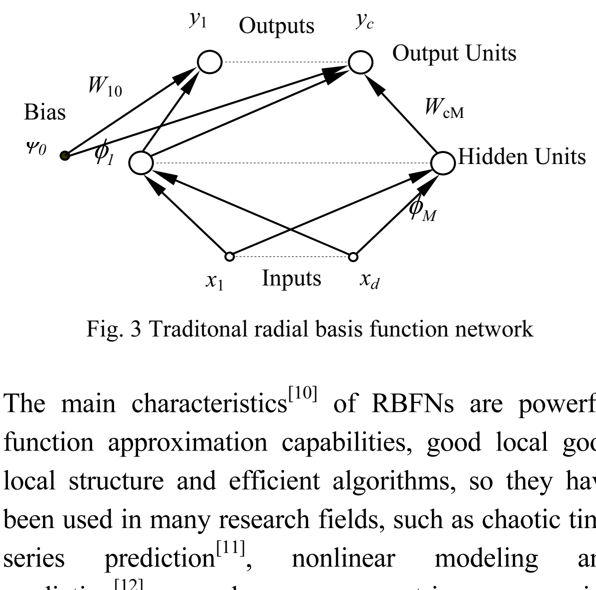


Fig. 3 Traditonal radial basis function network

The main characteristics[10] of RBFNs are powerful function approximation capabilities, good local good local structure and efficient algorithms, so they have been used in many research fields, such as chaotic time series prediction[11], nonlinear modeling and prediction[12] and nonparametric regression estimation[13].

In this paper, the Hardy[14] method is adopted as described by Kansa[15]:

Let $x_1, \ldots, x_N \in \Omega \subset \mathbf{R}^n$ be a given set of nodes. Let

$$g_j(\boldsymbol{x}) \equiv \boldsymbol{g}\left(\left\|\boldsymbol{x} - x_j\right\|\right) \in \mathbf{R}, j = 1, \ldots, N, \tag{2}$$

be a set of any RBF basis functions. Here $\left\|\boldsymbol{x} - x_j\right\|$ is the Euclidean distance. Given interpolation data values $y_1, \ldots, y_N \in \mathbf{R}$ at locations $x_1, \ldots, x_N \in \Omega \subset \mathbf{R}^n$, the RBF interpolate

$$F(\boldsymbol{x}) = \sum_{j=1}^{N} \alpha_j g_j(\boldsymbol{x}) + \alpha_{N+1} \tag{3}$$

is obtained by solving the system of $N+1$ linear equations

$$\begin{aligned} \sum_{j=1}^{N} \alpha_j g_j(\boldsymbol{x}) + \alpha_{N+1} &= y_i,\ i = 1, \ldots, N, \\ \sum_{j=1}^{N} \alpha_j &= 0, \end{aligned} \tag{4}$$

for $N+1$ unknown expansion coefficients α_j. Hardy[14] adds a constant to the expansion and constrains the sum of the expansion coefficients to be zero. Introducing the notation

$$\boldsymbol{p} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \in \mathbf{R}^{\mathrm{N}},\ \boldsymbol{H} = \begin{bmatrix} \boldsymbol{G} & \boldsymbol{p} \\ \boldsymbol{p}^{\mathrm{T}} & \boldsymbol{0} \end{bmatrix} \in \mathbf{R}^{(\mathrm{N}+1)\times(\mathrm{N}+1)},$$

$$\boldsymbol{G} = \begin{bmatrix} g_1(x_1) & \cdots & g_N(x_1) \\ \vdots & \vdots & \vdots \\ g_1(x_N) & \cdots & g_N(x_N) \end{bmatrix} \in \mathbf{R}^{\mathrm{N}\times\mathrm{N}}$$

$$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{N+1})^{\mathrm{T}},\ \boldsymbol{y} = (y_1, \ldots, y_N, 0)^{\mathrm{T}} \in \mathbf{R}^{\mathrm{N}+1},$$

the system (4) can rewritten in the matrix form as

$$\boldsymbol{H} \bullet \boldsymbol{\alpha} = \boldsymbol{y}. \tag{5}$$

Then the interpolation expansion coefficients are given by

$$\boldsymbol{\alpha} = \boldsymbol{H}^{-1} \bullet \boldsymbol{y}. \tag{6}$$

Therefore, either the value of interpolate or the derivatives at the nodes x_i can be easily found, e.g.

$$F'(x_i) = \sum_{j=1}^{N} \alpha_j g_j'(x_i),\ i = 1, \ldots, N, \tag{7}$$

$$F''(x_i) = \sum_{j=1}^{N} \alpha_j g_j''(x_i),\ i = 1, \ldots, N. \tag{8}$$

The radial basis function used in this paper is

$$\left\|\boldsymbol{x} - x_i\right\| c, \tag{9}$$

where c is a shape function variable between $0.2 < c < 3$, the reason for choosing this basis function is its ability to model extreme functions within a narrow range of values of c.

2.4 Procedure of optimal Latin hypercube design and radial basis function network applied in CO

Remaining the advantage, i.e. the disciplinary-level autonomous optimization in CO, the initial point is created with the optimal Latin hypercube design and the approximations of compatibility constraints and system-level objective are modeled with the radial basis function network.

In order to achieve more accurate approximation, the adaptive approximation is used in single optimization (AASO) [16]. In this strategy, the approximation model is improved and updated during the process of optimization until convergence. The main procedure is specified as follows:

1) Use optimal Latin hypercube design to create a dispersion of inputs vectors $\{z\}_k$ to a disciplinary to provide a wide and unbiased coverage of the design space defined by the system design variables.

2) Perform disciplinary-level optimizations at each sample point from the DOE. Minimize the difference between the target vector $\{z\}$ and local values of $\{\boldsymbol{x}\}$, which is the compatibility constraint $\boldsymbol{J}$, and compute the system-level objective f.

3) Find the best point among these sample points as the initial point z_0 of optimization.

4) Create a surrogate approximation model for each output item with radial basis function network.

5) Perform one complete optimization with approximation model, and calculate the penalty function of optimal design point z^* in the approximation model.

$$F_{\text{approximation}} = f_{\text{approximation}}(\boldsymbol{x}) + \sum\left(g_i(\boldsymbol{x}) - lb\right)^2, \quad (10)$$

where $g_i(\boldsymbol{x})$ is the constraint value, lb is the boundary value.

6) Update the approximation model with the optimal design point z^*.

7) Perform exact analysis of the optimal design point, and also calculate the penalty function of the optimal design point z^* in exact model.

$$F_{\text{exact}} = f(\boldsymbol{x}) + \sum\left(g_i(x) - lb\right)^2. \quad (11)$$

8) Compare $F_{\text{approximation}}$ with F_{exact}, if $F_{\text{exact}} \le F_{\text{approximation}}$, set initial point $z_{0\text{next}}$ with z^* in the next iteration, otherwise, set initial point $z_{0\text{next}}$ with z_0 in the next iteration.

9) Check if the design history has converged. If the convergence has been achieved, terminate the process. Otherwise, continue steps 5) to 8) until convergence.

3 Examples

In order to verify the ability of the application of optimal Latin hypercube design and radial basis function network to Collaborative Optimization, two examples are adopted, one is a typical nonlinear inequality constraint optimization problem, the other is an MDO benchmark optimization problem of NASA. All the optimizations in this paper are performed with iSIGHT (Engineous Software Inc. 2004).

3.1 Mathematical example

This example was first used by Robert [17] to test the MDO method. It is a nonlinear inequality constraint optimization problem, which is specified as follows:

$$\begin{cases} \min\ f(x) = x_1^2 + x_2^2, \\ \text{s.t. } x_1 + 0.1x_2 < 4, \\ \quad 0.1x_1 + x_2 > 2. \end{cases} \quad (12)$$

The conventional CO model of this mathematical example is shown below:

The system-level optimization problem is

$$\begin{cases} \min\ f(x) = z_1^2 + z_2^2, \\ \text{s.t. } J_1 = (z_1 - x_1\text{sub1})^2 + (z_2 - x_2\text{sub1})^2 < \varepsilon, \\ J_2 = (z_1 - x_1\text{sub2})^2 + (z_2 - x_2\text{sub2})^2 < \varepsilon, \end{cases} \quad (13)$$

where ε is the compatibility constraint, here we set $\varepsilon = 0.0001$.

The subsystem1 optimization problem is

$$\begin{cases} \min\ J_1 = (z_1 - x_1\text{sub1})^2 + (z_2 - x_2\text{sub1})^2, \\ \text{s.t. } x_1\text{sub1} + 0.1x_2\text{sub1} < 4. \end{cases} \quad (14)$$

The subsystem2 optimization problem is

$$\begin{cases} \min\ J_2 = (z_1 - x_1\text{sub2})^2 + (z_2 - x_2\text{sub2})^2, \\ \text{s.t. } 0.1x_1\text{sub2} + x_2\text{sub2} > 2. \end{cases} \quad (15)$$

The optimal point of this problem is (0.198, 1.980)[19], the value of objective function is $f^* = 3.998$. This paper performs optimization with four different initial points. The method used in system-level and disciplinary-level optimizations is NLPQL (Sequential Quadratic Programming). Table 1 shows the optimization results with the conventional CO model.

Table 1 Results of example1 with standard CO model

Initial point	Optimal point	Value of objective	System-level iterations
(0,0)	(0.177 1, 1.972 3)	3.921 2	65
(1,1)	(0.205 8, 1.969 7)	3.922 0	67
(-5,5)	(-5.000 0, 4.995 0)	49.950 0	78
(-3,3)	(-3.000 0, 2.997 0)	17.982 0	36

It can be seen that the conventional CO model is sensitive to the initial point, but it is difficult for us to determine an initial point for practical engineering. Then optimal Latin hypercube design and radial basis function network are applied to collaborative optimization. The method used in system-level and disciplinary-level optimizations is NLPQL. Set 50 sample points in optimal Latin hypercube design, and construct approximation between system design variables and system-level outputs with radial basis function network. The result and cycle procedure are shown in Table 2 and Fig. 4 respectively.

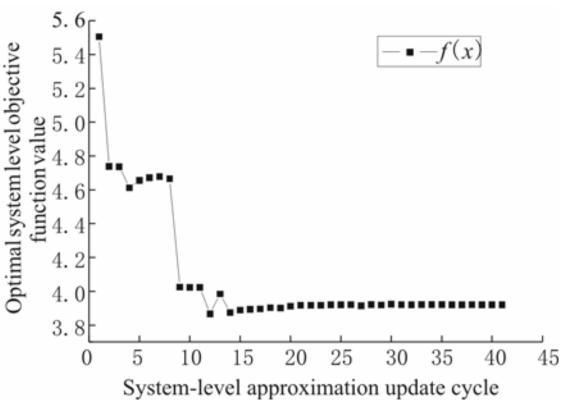


Fig.4 Optimal system-level objective function value vs system-level cycle

Fig.4 demonstrates that the optimization problem is close to convergence after 22 cycles and converges at the 41st cycle. The optimization result is near to the optimal value. Comparing with the result of the conventional CO, this method finds the optimal point with relatively less computation, and does not need to choose initial point. Therefore, it is more efficient and more robust than the conventional CO.

Table 2 Results of example1 with application of optimal Latin hypercube design and radial basis function network

Optimal point	Value of objective	System-level cycle
(0.195 4, 1.970 4)	3.920 8	41

3.2 Golinski's Speed Reducer Optimization

Golinski's Speed Reducer Optimization [18] is one of the ten standard examples which is used by NASA to provide the MDO researchers with a set of problems for the development of new optimization methodologies, to establish a "standard" set of problems for comparing relative advantages of MDO approaches and formulations and to provide the applied mathematics community with MDO problems' representative of various engineering areas. This problem represents the design of a simple gear box and is posed as an artificial multidisciplinary design problem comprising the coupling between gear design and shaft design disciplines. The objective is to minimize the speed reducer weight while satisfying a number of constraints imposed by gear and shaft design practices. Seven design variables are available to the optimizer, and each has an upper and lower limit imposed. The Speed Reducer model is shown in Fig. 5.

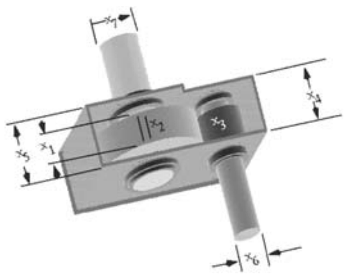


Fig. 5 The Speed Reducer model

Mathematically, the original definition of its optimization is specified as follows:

$$\min\ f(x) = 0.785\,4 x_1 x_2^2 \left(3.333\,3 x_3^2 + 14.933\,4 x_3 - 43.093\,4\right) - 1.508 x_1 \left(x_6^2 + x_7^2\right) + 7.477\left(x_6^3 + x_7^3\right) + 0.785\,4\left(x_4 x_6^2 + x_5 x_7^2\right) \quad (16)$$

s.t. $g_1 = 27/(x_1 x_2^2 x_3) - 10 \le 0$,

$g_2 = 397.5/(x_1 x_2^2 x_3^2) - 1 \le 0$,

$g_3 = 1.93/(x_2 x_3 x_6^4) - 1 \le 0$,

$g_4 = 1.93/(x_2 x_3 x_7^4) - 1 \le 0$,

$g_5 = A_1 / B_1 - 1100 \le 0$,

$g_6 = A_2 / B_2 - 850 \le 0$,

$g_7 = x_2 x_3 - 40 \le 0$,

$g_8 = x_1 / x_2 - 12 \le 0$,

$g_9 = -x_1 / x_2 + 5 \le 0$,

$g_{10} = (1.5x_6 + 1.9)/x_4 - 1 \le 0$,

$g_{11} = (1.1x_7 + 1.9)/x_5 - 1 \le 0$,

where,

$$\begin{cases} A_1 = \left[\left(\dfrac{745x_4}{x_2x_3}\right)^2 + 16.9\times10^6\right]^{0.5}, \quad B_1 = 0.1x_6^3 \\ A_2 = \left[\left(\dfrac{745x_5}{x_2x_3}\right)^2 + 157.5\times10^6\right]^{0.5}, \quad B_2 = 0.1x_7^3 \end{cases}$$

$2.6 \le x_1 \le 3.6$ width of the gear face, cm;

$0.7 \le x_2 \le 0.8$ teeth module, cm;

$17 \le x_3 \le 28$ number of pinion teeth;

$7.3 \le x_4 \le 8.3$ shaft 1's length between bearings, cm;

$7.3 \le x_5 \le 8.3$ shaft 2's length between bearings, cm;

$2.9 \le x_6 \le 3.9$ diameter of shaft 1, cm;

$5 \le x_7 \le 5.5$ diameter of shaft 2, cm.

The optimal point of this problem is (3.5, 0.7, 17, 0.3, 7.71, 3.35, 5.29) [19]. The value of objective function is $f^* = 2\,996.1701$. This paper performs optimization with four different initial points. The method used in system-level and disciplinary-level optimizations is NLPQL (Sequential Quadratic Programming). Table 3 shows the optimization results with conventional CO model.

The same as the first example, it is also found that conventional CO model is sensitive to the initial point, which is difficult for us to determine for practical engineering problem. Therefore, in this paper, optimal Latin hypercube design and radial basis function network are applied to collaborative optimization. The method used in system-level and disciplinary-level optimizations is NLPQL. Set 50 sample points in optimal Latin hypercube design, and construct approximation between system design variables and system-level outputs with radial basis function network. The result and the cycle procedure are shown in Table 4 and Fig. 6 respectively.

Table 3 Results of example 2 with conventional CO model

Initial point	Optimal point	Values of objective	System-level iterations
(3.500, 0.700, 20.00, 7.300, 7.714, 3.350, 5.286)	(3.499, 0.699, 17.00, 7.300, 7.723, 3.350, 5.291)	2 997.111 7	14
(3.000, 0.800, 20.00, 7.300, 7.300, 3.350, 5.100)	(3.499, 0.699, 17.00, 7.300, 7.786, 3.350, 5.305)	3 007.050 6	49
(3.500, 0.800, 17.00, 7.500, 7.600, 3.000, 5.500)	(3.498, 0.709, 17.00, 7.485, 7.762, 3.351, 5.329)	3 024.437 6	43
(3.500, 0.800, 20.00, 7.300, 7.300, 3.500, 5.300)	(3.499, 0.699, 17.00, 7.300, 7.723, 3.500, 5.292)	3 037.263 4	50

Table 4 Results of example 2 with application of optimal Latin hypercube design and radial basis function network

Optimal point	Value of objective	System-level iterations
(3.493, 0.701, 17.00, 7.953, 7.715, 3.355, 5.287)	3 000.754 76	22

Fig.6 demonstrates that the optimization problem is close to convergence after 7 cycles and converges at the 22nd cycle. The optimization result is much close to the optimal value. Therefore, with the application of optimal Latin hypercube design and radial basis function network to CO, there is no need for researchers to choose the initial point, and more accurate results can be found while reducing the heavy burden of computation. Comparing with the result of conventional CO, this method is more efficient and more robust.

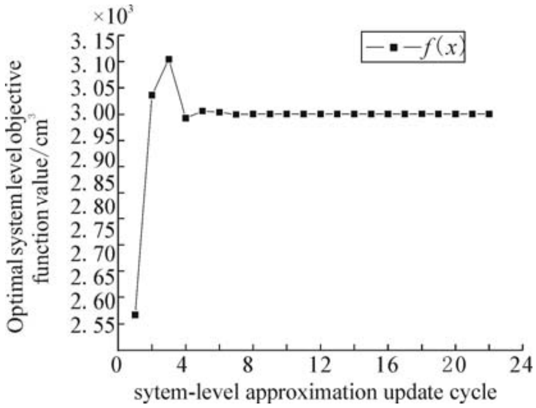


Fig.6 Optimal system-level objective function value vs system-level cycle

4 Conclusions

Collaborative optimization is a widely used multidisciplinary design optimization method. However, conventional CO has suffered from some difficulties such as the choice of initial point and tremendous computation. By applying the optimal Latin hypercube design and the radial basis function network to conventional CO, it has been shown in this paper that these deficiencies can be overcome. optimal Latin hypercube design can generate more evenly distributed sample points, and provide a rough estimate of an optimal design which can be used as an initial point for optimization. Since the radial basis function network has powerful function approximation capabilities, it is adopted in this paper to approximate the disciplinary-level optimal objective function and the system-level objective function. The approximation model is updated during the optimization. Comparing with the conventional CO, the advantages of this method are as follows: 1) avoiding the choice of initial point; 2) relieving the heavy burden of computation while satisfaction of accuracy; 3) improving the efficiency of convergence and the robustness. The example results have proved that the above performances are better than conventional CO.

The ship optimization design also belongs to MDO, which contains many disciplines, such as structure mechanics, propulsion, resistance, machinery and cost. There is much interactive coupling among these disciplines. How to coordinate these disciplines in the optimization design is very important. CO is a feasible and effective way to coordinate all disciplines simultaneously. However, some disciplinary analyses need lots of time to operate, such as Fluent for hydrodynamics and Nastran for structural analysis. The computation will be tremendous if these disciplinary analyses are integrated into the overall design optimization directly. The application of optimal Latin hypercube design and radial basis function network to CO is an effective way to relieve the burden of computation while satisfying the accuracy, and provides a promising method for the ship optimization design. So the future research is to apply this method to the ship design optimization.

References

[1] SOBIESCZANSKI-SOBIESKI J, HAFTKA R T. Multidisciplinary aerospace design optimization: survey of recent developments[R]. AIAA-1996-0711, Reno: AIAA, 1996.

[2] BRAUN R D, MOORE A A, KROO I M. Collaborative architecture for launch vehicle design[J]. Journal of Spacecraft and Rockets, 1997, 34(4): 478-486.

[3] MCALLISTER C D, SIMPSON T W, KURTZ P H, et al. Multidisciplinary design optimization test based on autonomous underwater vehicle design[R]. AIAA-2002-5630, Atlanta: 2002.

[4] LEE K Y, ROH M I. A hybrid optimization method for multidisciplinary ship design[J]. Journal of Ship Technology Research, 2000, 47(4): 181-185.

[5] SOBIESKI I P, MANNING V M, KROO I M. Response surface estimation and refinement in collaborative optimization[R]. AIAA-1998-4753, St. Louis, 1998.

[6] KROO I M, MANNING V M. Collaborative optimization: status and directions[R]. AIAA-2000-4721, Long Beach, 2000.

[7] de MIGUEL A V, MURRAY W. An Analysis of Collaborative Optimization Methods [R]. AIAA-2000-4720, Long Beach, 2000.

[8] HAN Minghong, DENG Jiati. Improvement of collaborative optimization[J]. Chinese Journal of Mechanical Engineering, 2006, 42(11): 34-38 (in Chinese).

[9] JIN R, CHEN W, SUDJIANTO A. An efficient algorithm for constructing optimal design of computer experiments [J]. Journal of Statistical Planning and Inference, 2003, 134(1): 268-287.

[10] GHOSH J, NAG A. An overview of radial basis function networks [A]. In: Howlett R J and Jain L C. Radial Basis Function Neural Network Theory and Applications[M]. Heidelberg: Physica-Verlag, 2000.

[11] ERFANIAN, A, Chaotic radial basis function network with application to dynamic modeling of chaotic time series[A]. Chang H k and Zhang Y T. Proceedings of the 20th Annual International Conference of the IEEE[C]. Hong Kong, 1998.

[12] HE Xiangdong, LAPEDES A. Successive approximation radial basis function networks for nonlinear modeling and prediction[A]. Proceedings of 1993 International Joint Conference on Neural Networks[C]. Nagoya, 1993.

[13] KRZYZAK A, SCHAFER D. Nonparametric regression estimation by normalized radial basis function networks[J]. IEEE transactions on information theory, 2005, 51(3): 1003-1010.

[14] HARDY R L. Multiquadratic equations of topography and other irregular surfaces[J]. Journal of Geophysics, 1971, 76(8): 1905-1915.

[15] KANSA E J. Motivation for using radial basis functions to solve PDEs[EB/OL]. 1999-08-24, http://rbf-pde.uah.edu/kansaweb.html/.

[16] JANG B S, YANG Y S, JUNG H S, et al. Managing approximation models in collaborative optimization[J]. Structural and Multidisciplinary Optimization, 30(1): 11-26.

[17] BRAUN R, GAGE P, KROO I, et al. Implementation and performance issues in collaborative optimization[R]. AIAA-1996-4017, Washington: AIAA, 1996.

[18] PADULA S L, ALEXANDROV N M, GREEN L L. MDO test suite at NASA lLangley research center [R]. AIAA-1996-4028, Washington: AIAA, 1996.

[19] AZARM S, LI W C. Optimality and constrained derivatives in two-Level design optimization [J]. Journal of Mechanical Design, 1990, 112 (12): 563-568.

ZHAO Min was born in 1981. He is a PhD candidate for design and manufacture of naval and ocean structures at school of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University. His current research interest is Multidisciplinary Design Optimization of submersible design.

CUI Weicheng was born in 1963. He is a professor and PhD tutorr of China Ship Scientific Research Center. His recent research interests include Marine Structural Reliability, Systematic Investigation of Ultimate Strength Analysis Methods, Hydroelastic Responses of Very Large Floating Structures and Multidisciplinary Design Optimization of submersible design.