1. INTRODUCTION
Numerous FE packages able to solve various kinds of practical problems have been well developed. It is known that optimal design problems need to be solved iteratively. How to use a FE package to solve optimal design problems is not a straightforward task. Most optimization literature is focused on verifying the validity of the optimization algorithms or theories and the illustrated examples are usually simple. For example, optimal design of truss and beam structures is frequently discussed. However, optimization of plate and shell structures is seldom investigated due to the complicated and time-consuming development of analysis codes. The objective of this paper is to develop a general-purpose, flexible, multi-functional and easy-to-maintain software system that integrates the optimization tools and FE packages. The developed system allows one to solve various kinds of advanced optimal design problems such as plate, shell, and nonlinear problems.

There are three different ways in integrating FE packages and optimization tools into an optimal design system, see Schneider et al. (2002). The first one is called the simulation-based approach. In this approach, the optimization algorithm and the definition of optimal design problems are implemented into the analysis package. For example, the Design Optimization module and Topology Optimization module in ANSYS (2005) provide users an interface to solve optimal design problems. In this way, the design system is not flexible since users cannot make any modification. Hence, this approach is not practical to develop a general-purpose optimal design system.

The second one is called the optimization-based approach, where the optimization algorithm is the core and the analysis by FE package is executed externally. For example, Zhang et al. (1999) integrated ABAQUS into optimal design system through two sets of text files. However, the system developed in this way has to be re-implemented when one wants to use other optimization tools. Therefore, this approach is limited when developing a flexible and extensible optimal design system.

The third one is called the module-based approach. Schneider et al. divided the system into four modules: model generation module, simulation module, error
calculation module and optimization module. In this approach, an optimization tool or a FE package can be added into the system without changing most parts of the system. In this paper, the module-based approach is used and the object-oriented (OO) methodology was adopted to develop a general-purpose and flexible optimal design system.

2. BACKGROUND OF OPTIMAL DESIGN

In this section, mathematical formulation of optimization problems is given first. Then, optimization methods used in this study are briefly reviewed.

2.1. Mathematical Formulation of Optimization Problems

An optimization problem can be defined as finding a set of design variables to minimize (or maximize) the objective function subject to some constraints. The above description can be summarized in Eqn 1, see Arora (1989):

\[ \text{Min. } f(x) = f(x_1, x_2, \ldots, x_n) \]  
\[ \text{Subject to: } h_i(x) = h_i(x_1, x_2, \ldots, x_n) = 0 \quad i = 1, \ldots, p \]  
\[ g_j(x) = g_j(x_1, x_2, \ldots, x_n) \leq 0 \quad j = 1, \ldots, q \]  
\[ x_k^l \leq x_k \leq x_k^u \quad k = 1, \ldots, n \]  

where \( x = \{x_1, x_2, \ldots, x_n\} \) is a set of design variables, \( f(x) \) is the objective function to be minimized, \( h_i(x) \) is the \( i \)th equality constraint, \( g_j(x) \) is the \( j \)th inequality constraint, and \( x_k^l \) and \( x_k^u \) are, respectively, the lower-bound and upper-bound of the \( k \)th design variable. In general, structural optimization problems are constrained and nonlinear.

2.2. Mathematical Programming

Among all the numerical methods for solving constrained nonlinear optimization problems, sequential quadratic programming (SQP) method is one of the best tools. The main concept of the SQP method is to transform the optimization problem into a quadratic programming problem during iteration. Detailed mathematical formulation of the SQP method can be found, for example in Arora (1989).

Two optimization tools: CFSQP (Lawrence et al., 1997) and IMSL C Numeric Library (2003), based on the SQP method, are integrated into the proposed optimal design system. CFSQP (C code for Feasible Sequential Quadratic Programming) is different from other SQP tools in that the code could detect whether the initial design is feasible and automatically generate a feasible initial design if the initial design is infeasible. One should implement four subroutines — obj(), cntr(), grob() and grcn() to define the objective function, constraints and their gradients when using CFSQP.

IMSL Library is developed by Visual Numerics and is composed of two main libraries (IMSL/ Math Library and IMSL/Stat Library). The “constrained_nlp()” function in IMSL/Math Library integrated into the system is also based on the SQP algorithm. One needs to implement fcn() and grad() functions to calculate the objective function and constraints when using the “constrained_nlp()” function.

2.3. Topology Optimization

In traditional structural optimal design problems, member properties or nodal coordinates are often chosen as design variables. However, in structural topology optimization problems, an initial design domain is given first and then, by changing the topology of the structure, the final optimal topology (or material distribution) is obtained while satisfying some specified constraints. In the past two decades, several topology optimization methods have been developed. For example, the homogenization method (Bendsoe and Kikuchi 1988) took the porous property of elements as design variables to obtain an optimal topology; the solid isotropic material with penalization (Bendsøe 1989; Mjelde 1992) chose the density of element as design variables to get an optimal solution; the Evolutionary Structural Optimization (ESO) (Xie and Steven 1997; Liang 2005) method found the final topology by alternating elemental properties or eliminating elements in the design domain.

The ESO method is chosen as a tool for topology optimization in the proposed system because it is convenient to be implemented with a commercial FE package. The procedure of this method is summarized as follows:

Step 1: Discretize the initial design domain into an appropriate number of finite elements.

Step 2: Execute a finite element analysis.

Step 3: According to the objective function of the topology optimization problem such as structural stiffness or modal frequency, calculate the element performance or sensitivity number of each element.

Step 4: Determine inefficient elements during iteration and gradually eliminate these elements by changing the thickness of the elements or directly remove the elements.

Step 5: Repeat Step 2 to Step 4 until an optimal topology is obtained.
3. SYSTEM FRAMEWORK ANALYSIS AND DESIGN

To design and implement a flexible and extensible optimal design system, modern software technologies including OO technology, design patterns and UML diagrams are used. First, the challenges faced are addressed before designing and developing the system. Then, the proposed optimal design system is discussed.

3.1 System Requirement Analysis

3.1.1. Execution of different FE packages

The process of optimal design requires iteration of objective function and constraints. In each design cycle, the physical problem needs to be analyzed by FE packages. There are various ways to execute the analysis package. For instance, ABAQUS (2004) can be executed by the command-line, API (Application Programming Interface), or GUI (Graphic User Interface) methods. To integrate a FE package into the optimal design system, one should provide an interface to let at least one of the execution methods work.

3.1.2. Extraction of the FE analysis results

After the execution of analysis using a FE package, the analysis results need to be obtained and processed in order to further calculate the objective function and constraints. Different FE packages have different kinds of analysis result formats. For example, analysis results can be reported in text-file format and binary-file format for ANSYS and in text-file format, binary-file format, and by the API method for ABAQUS. Thus, the optimal design system should offer a mechanism or an interface to automatically extract necessary FE analysis results from one of the result formats after each execution of the FE analysis.

3.1.3. Modification of the FE model

There are several types of optimal design problems including sizing, shape, and topology. Therefore, the FE model shall be modified according to the type of optimal design problems considered. To modify the FE model automatically during the design iteration, the system shall provide an interface to define the relationship between the FE model and the design variables considering various kinds of optimal design problems.

3.1.4. Linking of function evaluations with the FE analysis results

Optimal design procedure can be separated into two main phases. One is the optimization phase and the other is the analysis phase. Once the analysis phase is performed using a FE package, one could specify results to be output. The way to evaluate the objective function and constraints using these FE analysis results shall be established. Hence an interface must be provided for users to define the objective function and constraints, and their gradients will be calculated through the proposed system by using the forward finite difference method. Therefore, if the number of design variables is \( N \), then \((N+1)\) analyses need to be carried out. One analysis corresponds to the design and the other \( N \) analyses correspond to the perturbed designs. In each of the perturbed designs only one design variable is perturbed and the other ones being kept unchanged.

3.2. Design of the Optimal Design System

The proposed optimal design system is composed of five modules: OptDesign (Optimal Design) module, FEA (Finite Element Analysis) module, func_evaluation module and Model Modification module as depicted in Figure 1.

First, the OptDesign module is responsible for managing and controlling the whole optimal design procedure, including optimization analysis, execution of FE analysis, extraction of the FE analysis results, evaluation of the objective function and constraints, and modification of FE models. Object composition of the OO technology is mainly used to combine these five modules. The optimization tools implemented in the OptDesign module include two sequential quadratic programming algorithms; one is the subroutine constrained_nlp from the IMSL/Math Library and the other is CFSQP. In addition, the ESO is implemented for performing topology optimization.

Second, the FEA module is separated from the OptDesign module by adopting the Bridge pattern (Gamma et al. 1995) in order to let both FEA module and OptDesign be changed independently without affecting each other. The FEA module is responsible for executing the analysis of FE packages and checking the completeness of analysis. In addition, different ways of execution could be implemented in the subclass of the FEA class (module). Therefore, when a new FE package is to be integrated into the system, developers only have to add a new subclass of the FEA class and implement the ExecuteAnalysis() and WaitForCompletion() methods without changing other parts of the system. Currently, ABAQUS and ANSYS are implemented in the system.

Third, the Bridge pattern is used again to design the Data_Processor module and its implementation part, Data_Processor_Imp class. The abstraction of the Bridge pattern is the Data_Processor class and it manipulates both input data and output results of the FE package. The implementation part is responsible for the implementation of processing each FE package input
files and output results in detail. Also, both model information and analysis results required for further analysis is stored in the Model object and Result object, respectively in the system. When one wants to add a new FE package, he should add a subclass of the Data_Processor_Imp class and implement its ParseModelInput(), ParseResultOutput() and Create() methods, and hence the abstraction part and other modules of the system are not influenced by this modification.

Fourth, the Model_Modification module is designed to handle the challenge of modifying the FE model in different ways according to the type of optimal design problem considered before a new input file of FE package is created for next execution. The Strategy pattern (Gamma et al. 1995) is adopted in the design of this module. The context (the OptDesign class) and the strategy of modifying the FE model are separated. The Model_Modification module provides an interface to modify or changing the FE model in different ways such as changing the cross-sectional properties, modifying nodal coordinates, deleting elemental connectivity.

Finally, the func_evaluation module is designed to calculate the values of objective function, constraints and their gradients when these evaluations are needed in the optimization phase. In general, the objective function, constraints and their gradients are often implicit functions, i.e. the functions cannot be expressed explicitly in terms of design variables. The FE model, the analysis results and optimal design parameters should be linked with the calculation of these functions, and therefore different methods of function evaluations.
are isolated from the OptDesign module and each method is derived from the based class func_evaluation. Furthermore, when a new method of calculation is to be added, one only needs to add a new subclass of the func_evaluation class and implement its GetValue() method.

The communication between users and the proposed optimal design system is through a text-file approach. Take the definition of optimal design parameters as an example, design variables, objective function, constraints and methods of modifying the FE model are given in an input file by user. Thus, both the objective function and constraints are evaluated by the func_evaluation module by linking the analysis results, optimal design parameters and the FE model with the calculation of each function. Also, the FE model is updated according to the specified methods given by user before a new FE input file is created.

4. NUMERICAL EXAMPLES AND DISCUSSIONS

In this section, the proposed optimal design system is used to solve various kinds of optimal design problems including sizing, shape and topology optimization. Plate and shell structures with linear or nonlinear behavior are the focus. In addition, an optimal design problem on heat transfer is presented.

The details about how the optimal design system works and how the user carries out the optimal design will be demonstrated in the first optimal design problem, including specifying design variables, defining objective function and constraints, and linking design variables to the FE model. For other optimal design examples, only input files that define the optimization problems are provided.

4.1. Plate with Concentrated Load at Center

Consider a simply supported plate with the dimension of 6m × 6m. A concentrated load \( P = 4000N \) is applied at the center of the plate. Material constants are as follow: Young’s modulus \( E \) and Poisson’s ratio \( \nu \) are 10GPa and 0.3, respectively. The objective is to minimize the structural compliance (or maximize the structural stiffness) subjected to a constant volume of material. This is equivalent to minimizing the central deflection \( w_{\text{center}} \) as depicted in Eqn. 2:

\[
\text{Min. } f(x) = \text{Min. compliance } \rightarrow \text{Min. } w_{\text{center}} \tag{2}
\]

The plate is divided into 64 equal-size elements and the thickness of each element is chosen as the design variables. Due to symmetry, only 16 design variables are taken into account as shown in Figure 2. The upper bound, lower bound and initial values of the design variables are 0.2m, 0.001m and 0.1m respectively. The constant consumed volume of material has the expression of:

\[
\sum t_i = \text{constant} = 1.6 \text{ m} \tag{3}
\]

The input file for defining the optimal design problem is shown in Figure 3. There are five main parts of this input file: OptInfo (setting parameters such as number of design variables and number of constraints), XInfo (specifying upper-bound, lower-bound and initial values of design variables), OBJECTIVE (defining objective function), CONSTRAINT (defining constraints), and DESCRIPTION (relating design variables to the FE model). In OBJECTIVE and CONSTRAINT parts, users could use three different types of equation bound: abs, ub, and lb, which, respectively, correspond to absolute-value type, greater-and-equal type, and less-and-equal type. In DESCRIPTION part, users should specify the relationship between the design variables and the FE model.

This example is analyzed using SHELL93 shell element in ANSYS, and the optimal thickness distribution obtained by ANSYS in conjunction with the IMSL/Math library are shown in Figure 4(a) and Table 1. It can be found that the central deflection of the optimal design is only 24.6% that of the initial design. Note that the optimal design and the initial design have the same volume of consumed material.

Similarly, the optimal thickness distribution can be found for the case where all the four edges are clamped; the results are shown in Figure 4(b) and Table 1 again. Note that the central deflection of the optimal design is only 19.4% that of the initial design in this case.
Finally, let the dimension of the plate be changed to be 6m × 3m; the optimal thickness distribution is shown in Figure 4(c). Comparing Figure 4(a), 4(b), and 4(c) with the Von Mises stress distribution of the initial design (not shown here), one can find that the thickness distribution actually follows the pattern of the stress distribution. Namely, regions with thicker thickness correspond to regions with higher Von Mises stress.
4.2. Shell Structure with Geometric Nonlinearity

The shell structure (Bergan et al. 1978) shown in Figure 5 is simply-supported along the straight edges and free along the curved edges. The concentrated load $P$ is applied at the center and the dimension of the shell is as follows: the length of the straight edges $2L = 508$ mm, the radius of curvature $R = 2540$ mm and the angle considered $2\theta = 0.2$ rad. The Young’s modulus $E$ and Poisson’s ratio $\nu$ are $310275$ N/mm$^2$ and 0.3, respectively.

The shell is divided into 64 elements and the element thickness of each element is considered as the design variables. Again, due to symmetry, only 16 design variables are needed as illustrated in Figure 6. The upper bound, lower bound, and initial values of the design variable are 12.7 mm, 3.0 mm and 6.35 mm, respectively.

The objective is to maximize the limit load $P_{\text{limit}}$ for such a geometrically nonlinear problem. The constant volume constraint is written as

$$\sum t_i = \sum t_i^{\text{min}} = \text{constant} = 101.6 \text{ mm} \quad (4)$$

The input file for defining this optimal design problem is given in Figure 7. This example is analyzed using S8R5 shell element in ABAQUS, where the RIKS method is chosen as the geometrically nonlinear analysis scheme. This optimal design problem is solved by ABAQUS in conjunction with CFSQP. The obtained optimal thickness distribution is shown in Figure 8 and Table 2. The limit load $P_{\text{limit}}$ increases significantly from $0.585$ kN of the initial design to $1.018$ kN of the optimal design.

### Table 1. Optimal thickness results of plate structures

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Initial Design (simply-supported)</th>
<th>Optimal Design (simply-supported)</th>
<th>Initial Design (clamped)</th>
<th>Optimal Design (clamped)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_2$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_3$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_4$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_5$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_6$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_7$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_8$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_9$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{10}$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{11}$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{12}$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{13}$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{14}$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{15}$ (m)</td>
<td>0.1000</td>
<td>0.0010</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$t_{16}$ (m)</td>
<td>0.1000</td>
<td>0.2000</td>
<td>0.1000</td>
<td>0.0010</td>
</tr>
<tr>
<td>$w_{\text{center}}$ (m)</td>
<td>0.001852</td>
<td>0.000455</td>
<td>0.000881</td>
<td>0.000171</td>
</tr>
<tr>
<td>$w_{\text{inner}}$ (m)</td>
<td>0.001852</td>
<td>0.000455</td>
<td>0.000881</td>
<td>0.000171</td>
</tr>
</tbody>
</table>

Figure 5. Shell structure with geometric nonlinearity

Figure 6. Design variable arrangement in optimal design of shell structure
Figure 7. Input file for defining optimal design problem of shell structure

Max. \( f(x) = P_{\text{limit}} \)
\[
\sum t_i = \sum t_i \text{initial} = \text{constant} = 101.6 \text{ mm}
\]

<table>
<thead>
<tr>
<th>OBJECTIVE</th>
<th>LIMIT_LOAD</th>
<th>1</th>
<th>lb</th>
<th>41</th>
<th>2</th>
<th>0.0</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>CONSTRAINT</th>
<th>eq</th>
<th>XSUM</th>
<th>1</th>
<th>ub</th>
<th>101.6</th>
</tr>
</thead>
</table>

Figure 8. Optimal thickness distribution of shell with geometric nonlinearity
4.3. Optimal Shape Design of a Simply-Supported Bridge Structure

Figure 9 shows a simply-supported bridge structure (Wang et al. 2002) that is performed for shape optimization. To simulate the behavior of this bridge structure, beam element BEAM3 in ANSYS is used for optimization. To simulate the behavior of this bridge (et al. 2002), the vertical (Y) coordinates of the nodes of the upper chord are chosen as the design variables. To maintain symmetry of the structure, only five design variables (Y₁, Y₂, Y₃, Y₄ and Y₅) are needed. The objective is to minimize the weight of the structure while the vertical displacements of node 8 and node 10 are limited to 1 cm.

Note that the initial design shown in Figure 9 is infeasible since Y₃ = 3.15 cm and Y₂ = 3.14 cm. The input file for defining this optimal design problem is shown in Figure 10. The optimal design problem is solved by ANSYS in conjunction with the IMSL/Math library and the results are shown in Figure 11 and Table 3. As can be seen, the obtained results are in excellent agreement with those of Wang et al. 2002.

4.4. Three-Dimensional Cantilever Beam

Consider a three-dimensional cantilever beam subjected to a load P = 100 N at the free end as shown in Figure 12. Material constants are: Young’s modulus E = 200 GPa and Poisson’s ratio ν = 0.3. The design domain of Figure 12 is discretized into 1500 1m × 1m elements and the energy removal scheme in the ESO (Evolutionary Structural Optimization) algorithm (Xie and Steven 1997) is adopted to obtain an optimal topology. Elements with strain energy lower than two percent of the average value are considered inefficient during the optimization process and need to be removed. By gradually removing such inefficient elements, an optimal topology can be obtained.

The 20-node element SOLID95 in ANSYS is used for the FE analysis. Figure 13 shows the input file for defining this optimal topology design problem. Users should only specify the values of RRᵥ, ERR, JTᵥmax, and RVᵥmax. In order to compare the optimal result proposed by Jacobsen et al. (1998), the stop criterion used to determine the final topology is chosen as RVᵥmax = 60%. The topology is shown in Figure 14(a) when the removed volume ratio is 60% of the original volume. The obtained topology is in good agreement with that reported by Jacobsen et al. as shown in Figure 14(b).

Table 2. Optimal thickness results of shell with geometric nonlinearity

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Initial Design</th>
<th>Optimal Design (ABAQUS - CFSQP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>t₁ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₂ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₃ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₄ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₅ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₆ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₇ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₈ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₉ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₀ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₁ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₂ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₃ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₄ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₅ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₆ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
<tr>
<td>t₁₇ (mm)</td>
<td>3.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

4.5. Two-Dimensional Heat Sink Device

Figure 15 is a two-dimensional heat sink device with a triangular-distributed heat flux at the bottom (q₀ = 0.02 W/mm). Only heat conduction in the device and heat convection between the device and its surrounding media are considered in this heat transfer problem. The heat transfer properties are: heat conduction coefficient k = 200 × 10⁻³ W/mm°C, heat convection coefficient h = 45 × 10⁻⁶ W/mm²°C and surrounding temperature t₀ = 20°C. The design variables are the heights of the ten rectangular fins while the width keeps unchanged (B₁ = 1 mm). The upper bound, lower bound and initial value of the design variables H are 21.0 mm, 8.0 mm and 16.0 mm, respectively. In order to consider heat transfer efficiency of the heat sink device, the objective is to minimize the thermal resistance (or maximize the thermal performance), which is defined by Knight et al. (1991) and Kim (2004) as

\[
R_{\text{convection}} = \frac{1}{hA_1} \quad (5a)
\]

\[
A_i = A_{\text{total}} + \eta A_{\text{baseline}} \quad (5b)
\]

\[
\eta = \frac{q_{\text{total}}}{q_{\text{baseline}}} = \frac{\int (T_{\text{in}} - T_{\text{out}}) dA}{A_{\text{baseline}}(T_{\text{in}} - T_{\text{out}})} \quad (5c)
\]

where \( R_{\text{convection}} \) is the thermal resistance of the system, \( A_i \) is total surface area for convection and \( \eta \) is the fin efficiency. The convectional surface area is composed of two parts. One is the convection area \( A_{\text{baseline}} \) at the...
Optimal Design System Using Finite Element Package as the Analysis Engine

Figure 9. Initial design of the bridge structure

![Initial design of the bridge structure](image_url)

Table 3. Optimal design results of bridge structure

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Optimal Design (ANSYS-IMSL)</th>
<th>(Wang et al. 2002)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Y_1 = Y_{11}$ (m)</td>
<td>1.041</td>
<td>1.042</td>
</tr>
<tr>
<td>$Y_3 = Y_{13}$ (m)</td>
<td>1.646</td>
<td>1.646</td>
</tr>
<tr>
<td>$Y_7 = Y_{17}$ (m)</td>
<td>2.180</td>
<td>2.172</td>
</tr>
<tr>
<td>$Y_9 = Y_{15}$ (m)</td>
<td>2.420</td>
<td>2.449</td>
</tr>
<tr>
<td>$Y_{11}$ (m)</td>
<td>2.456</td>
<td>2.544</td>
</tr>
<tr>
<td>Total Weight (kg)</td>
<td>488.8</td>
<td>489.6</td>
</tr>
</tbody>
</table>

The bottom of the device, and the other is the effective convection area $\eta A_{fr}$ provided by the fins as measured by the ratio of actual heat convection to ideal heat convection. The heat sink device is simulated using the DC2D4 element in ABAQUS. The constraints considered in this example are the side constraints of the total fin area, i.e. $300\text{mm}^2 \leq A_{fr} \leq 330\text{mm}^2$. The input file for defining optimal design problem of this example is showed in Figure 16. The optimal design

Figure 10. Input file for defining optimal design problem of bridge structure

![Input file for defining optimal design problem of bridge structure](image_url)

Figure 11. Optimal design of the bridge structure

![Optimal design of the bridge structure](image_url)
figure 12. design domain of the 3-d cantilever beam under concentrated load

figure 13. input file for defining optimal design problem of 3-d cantilever beam

figure 14. optimal topology of 3-d cantilever beam

one could rapidly integrate different commercial fe packages and optimization tools into the system as it is very flexible. the developed system not only solved the complicated communication between the fe packages and optimization tools, but also had the applications of optimal design in different fields. in addition, a text-file-based interface for defining optimization problem is provided for users to easily define different types of optimal design problems such as sizing, shape or topology optimization. finally, the optimal design system has been verified by some benchmark problems and applied successfully to obtain the optimal solutions for complex optimal design problems, including plate and shell structures with linear or nonlinear behavior and heat transfer problem. one can apply the system to more complicated optimal design problems without implementing the analysis code. thus, the coding time of complex analysis such as analysis of nonlinear problems or shell structures is remarkably saved.

5. conclusion

an optimal design system developed using modern software technologies has been proposed in this paper.
Figure 15. Heat sink device with triangular distributed heat flux

Figure 16. Input file for defining optimal design problem of heat sink device

Table 4. Optimal fin height distribution of heat sink device

<table>
<thead>
<tr>
<th>Design</th>
<th>Initial Design (ABAQUS + IMSL)</th>
<th>Optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1 (mm)</td>
<td>16.00</td>
<td>16.19</td>
</tr>
<tr>
<td>H2 (mm)</td>
<td>16.00</td>
<td>16.41</td>
</tr>
<tr>
<td>H3 (mm)</td>
<td>16.00</td>
<td>17.11</td>
</tr>
<tr>
<td>H4 (mm)</td>
<td>16.00</td>
<td>18.48</td>
</tr>
<tr>
<td>H5 (mm)</td>
<td>16.00</td>
<td>18.80</td>
</tr>
<tr>
<td>H6 (mm)</td>
<td>16.00</td>
<td>18.80</td>
</tr>
<tr>
<td>H7 (mm)</td>
<td>16.00</td>
<td>18.48</td>
</tr>
<tr>
<td>H8 (mm)</td>
<td>16.00</td>
<td>17.11</td>
</tr>
<tr>
<td>H9 (mm)</td>
<td>16.00</td>
<td>16.41</td>
</tr>
<tr>
<td>H10 (mm)</td>
<td>16.00</td>
<td>16.19</td>
</tr>
<tr>
<td>R\text{convection} (°C/W)</td>
<td>7.815</td>
<td>7.542</td>
</tr>
</tbody>
</table>

* Hi is the height of the i\textsuperscript{th} fin counted from the left side.
ACKNOWLEDGMENT
This research was sponsored by the National Science Council of the Republic of China under Grant Numbers NSC 93-2211-E-002-043 and NSC 94-2211-E-002-050.

REFERENCES

APPENDIX: NOTATION

$A_{base}$ = Convection area at the bottom of the heat sink device
$A_s$ = Total surface area for convection
$B_i, H_i$ = Width and height of the rectangular fin
$E$ = Young’s modulus
$h$ = Heat convection coefficient
$k$ = Heat conduction coefficient
$L$ = Half of the length of the straight edge in shell structure
$P_{lim}$ = Limit load of shell structure
$q_0$ = Heat flux
$R$ = Radius of curvature of shell structure
$R_{convection}$ = Thermal resistance
$\Delta w_{center}$ = Central deflection of plate structure
$\eta$ = Fin efficiency
$\eta_{fin}$ = Effective convection area provided by the rectangular fins
$\nu$ = Poisson’s ratio
$\theta$ = Angle considered of the shell structure
$\rho$ = Density