

Applying Virtual Reality Techniques to Sensitivity-Based Structural Shape Design

T. P. Yeh

Engineering Animation, Inc.
ISU Research Park
2321 North Loop Drive
Ames, IA 50010,
tyeh@eai.com

J. M. Vance

Iowa Center for Emerging
Manufacturing Technology,
Department of Mechanical Engineering,
Iowa State University, Ames, IA 50011,
jmvance@iastate.edu

Virtual reality (VR) provides a design space consisting of three-dimensional computer images where participants can interact with these images using natural human motions in real time. In the field of engineering design, prototyping and design verification have provided the initial application areas for VR. The research presented in this paper takes the scenario one step further by incorporating free-form deformation techniques and sensitivity analysis into the virtual world such that the designer can easily implement analysis-based shape design of a structural system where stress considerations are important. NURBS-based free-form deformation (NFFD) methods and direct manipulation techniques are used as the interface between the VR interaction and the finite element model. Sensitivity analysis is used to allow the designer to change the design model and immediately view the effects without performing a re-analysis. An engine connecting rod is analyzed to demonstrate how virtual reality techniques can be applied to structural shape design.

Introduction

Virtual reality techniques can be used to remove the barriers of the traditional computer interface and allow the user to experience a strong sense of presence in the three-dimensional computer environment. Immersion is a key component in virtual reality and serves to distinguish VR from traditional three-dimensional simulations (Aukstakalnis and Blatner, 1992). Immersion refers to the user's ability to feel present in the computer environment. It is supported in VR by the use of a viewer-centered frame of reference and natural human motions to interact with computer models in real time. Much of engineering today is focused around building and analyzing computer models. Since virtual reality offers an innovative way to interact with computer models, researchers have started to investigate the use of VR for applications in engineering design and manufacturing (Gupta, 1996; Vance, 1996).

This paper presents an application of virtual reality techniques to the specific challenge of structural shape design. Our goal was to develop an application where the user can change the shape of a virtual object and examine the effect the shape change has on the structure deformation and stress distribution throughout the object. Using such a tool, a designer can easily relate shape changes to structural responses and quickly find a suitable shape that will meet deformation or stress constraints.

Most current methods of structural shape design are based on parametric or variational geometry-based CAD packages such as Pro/Engineer (Chen and Tortorelli, 1996; Hardee et al., 1996). It is our belief that using VR will enhance the designer's ability in visualization and manipulation for structural shape design in the following ways:

1. VR allows the designer to use an easy and intuitive way to navigate among the computer models. When using traditional computer interfaces, a mouse or joystick controls the viewpoint. In a virtual environment, head movement controls the viewpoint, just as in real life when the user views a physical model. This capability provides the user with a better spatial understanding of the three-dimensional geometry of the computer models that is not

provided by traditional computer interfaces. The mapping of physical motion to virtual motion can be properly scaled to adjust the viewing range in the virtual environment. For some virtual spaces which are too large to be navigated by physical motion due to the limitation of the tracking device, a wand or other such device can be used to allow the user to "fly" through the virtual world which further extends the user's ability to navigate.

2. Natural hand motion can be used to manipulate virtual objects in VR. By using an input device, such as an instrumented glove, a three-dimensional mouse or even audio input, the designer is able to design and manipulate computer models as if they existed as real objects. The dimensions and the free-form shape of the virtual object can be changed by intuitively grabbing or pressing the object by using simple hand motions and gestures.
3. VR offers a stereo view of the computer model which further extend the user's understanding of the three-dimensional model. The stereo view is created by using offset images to simulate the binocular depth cues of convergence and stereopsis of the user's eyes. The depth cues provided through the stereo image are very helpful in conveying the spatial concept of the computer-generated images.

VR provides a real time environment for viewing and manipulating computer model. In order to implement structural shape design in VR, structural responses related to the model's shape are also required for real time determination. Re-solving the analysis in real time is not feasible, therefore, a linear approximation based on sensitivity analysis is used to approximate the deformation and stress changes. A mechanism to relate the structural shape design variables and the finite element model should be well established such that it not only provides an easy way to manipulate the structural shape but also maintains a meaningful solution to the finite element based sensitivity analysis. The combination of providing an immersive environment together with the ability to use natural hand motions to alter computer models provides a unique interface for structural shape design. The image shown in Fig. 1 illustrates the concept of the structural shape design in the virtual environment. It is our intent to couple the analysis results with shape design such that a designer can intuitively alter the shape of the image as if a physical model is placed in front of the designer, and exam-

Contributed by the Design for Manufacturability Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Jun. 1997; revised Aug. 1998. Associate Technical Editor: K. Ishii.

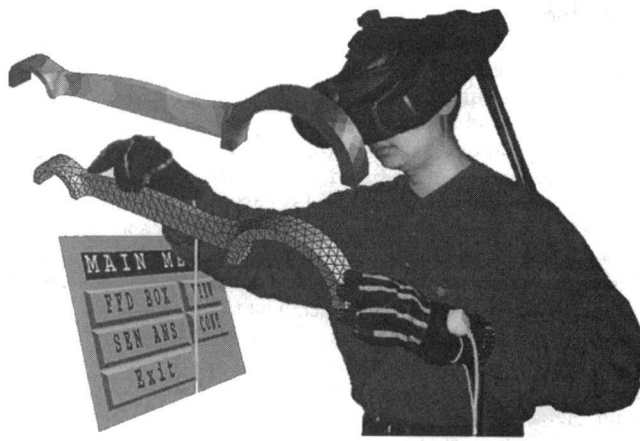


Fig. 1 Structural shape design in a virtual environment

ine the resulting changes to the deformation and stresses visually displayed on the virtual object.

A general purpose structural shape design program is developed in this project to take full advantage of these VR characteristics. The design variables are limited to the data points on the object or the shape control points of the object such that when the designer grabs a data point or a control point by using an instrumented glove, the design variable is automatically assigned. The linear relationships between changes in the design variable and the object's shape change are maintained in order to utilize linear sensitivity methods to update the analysis results (Choi and Chang, 1994). NURBS-based free-form deformation (NFFD) and direct manipulation techniques are the methods implemented which allow the designer to interactively change the shape of the object. They are also used to determine the change in the position of the finite element nodal coordinates as a function of the shape change. Deformation and stress sensitivities obtained from the finite element model can then be used to approximate the corresponding structural responses.

Nurbs-Based Free-Form Deformation and Direct Manipulation

NURBS-based free-form deformation (NFFD), proposed by Lamousin and Waggenspack (Lamousin and Waggenspack, 1994), is a technique that allows a designer to modify the shape of a three-dimensional solid model. The procedure consists of placing the solid model within a non-uniform rational B-spline (NURBS) volume defined by a parallelepiped control point lattice. As the user deforms the NURBS volume, the underlying solid model is deformed. Any point inside the NURBS volume can be evaluated based on the newly deformed lattice. This method allows shape changes which are independent of the type of geometric model description of the original embedded model. Since the finite element model can be thought of as a secondary representation of a solid model, in this project, we are using NFFD to manipulate finite element models which provide the finite element nodal coordinates and connectivity information only. Changes in the location of the finite element nodes are then used to calculate stress and deformation changes.

Free-form deformation (FFD) was first presented by Sederberg and Parry (Sederberg and Parry, 1986). In their work, the 3D volume is defined as a trivariate tensor product Bernstein polynomial. Griessmair and Pughthofer (Griessmair and Pughthofer, 1989) then used trivariate B-splines as blending functions, and Lasser (Lasser, 1995) presented rational Bézier volumes for the FFD basis. Although the basis functions to define the 3D volume can be different, the processes of the free-form deformation are essentially the same. The designer first constructs the control point lattice embedding the geometric model.

The control point lattice is used to construct a 3D volume that provides a unique mapping from the parametric space of an unit cube to the homogeneous coordinate of the volume that embeds the model. For the NURBS-based free-form deformation, a data point S in a 3D NURBS volume with parametric space coordinate (u, v, w) is defined as

$$S(u, v, w) = \frac{\sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^o N_{i,p}(u) N_{j,q}(v) N_{k,r}(w) W_{i,j,k} P_{i,j,k}}{\sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^o N_{i,p}(u) N_{j,q}(v) N_{k,r}(w) W_{i,j,k}} \quad (1)$$

where p, q and r are the degrees, and $(n + 1)$, $(m + 1)$, and $(o + 1)$ are the number of control points $P_{i,j,k}$ in the three parametric directions respectively. $\{N_{i,p}(u)\}$, $\{N_{j,q}(v)\}$ and $\{N_{k,r}(w)\}$ are the B-spline basis functions defined over the nonperiodic, nonuniform knot vectors in u, v and w parametric coordinates. $W_{i,j,k}$ are the weighting factors. The knot vectors U, V and W , with $n + p + 2$, $m + q + 2$ and $o + r + 2$ nondecreasing sequence of knots respectively, are defined as follow.

$$U = \{ \underbrace{0, \dots, 0}_{p+1}, u_{p+1}, \dots, u_n, \underbrace{1, \dots, 1}_{p+1} \}$$

$$V = \{ \underbrace{0, \dots, 0}_{q+1}, v_{q+1}, \dots, v_m, \underbrace{1, \dots, 1}_{q+1} \}$$

$$W = \{ \underbrace{0, \dots, 0}_{r+1}, w_{r+1}, \dots, w_o, \underbrace{1, \dots, 1}_{r+1} \}$$

Note that the index number of the knot starts from 0. The parametric coordinate of each data point on the geometry should be determined first in order to relate the geometric model to the 3D NURBS volume. Determining the parametric coordinates of a NURBS data point is subject to numerical search methods. For a specified data point \hat{S} , the parametric value $(\hat{u}, \hat{v}, \hat{w})$ is determined by finding the optimal solution of the following form

$$\begin{aligned} &\text{Minimize } \|S(\hat{u}, \hat{v}, \hat{w}) - \hat{S}\| \\ &\text{Subject to } 0 \leq \hat{u}, \hat{v}, \hat{w} \leq 1 \end{aligned} \quad (2)$$

The code used to solve the above optimization problem is CFSQP (Tits et al., 1996), which is a C code for the feasible sequential quadratic programming method. The initial searching point of $(\hat{u}, \hat{v}, \hat{w})$ is set to $(0.5, 0.5, 0.5)$. Once the data points of the geometric model are parametrized using Eq. (2), the isoparametric mapping method is used to update the model by Eq. (1) when the control point lattice is modified. The isoparametric mapping method simply uses the same parametric values derived by Eq. (2) to evaluate the corresponding data point with a new control point setting by using Eq. (1).

Using NURBS as the basis functions for FFD allows the user to modify the solid model in many ways by either changing the control point coordinates, the weighting factors or even the knot vectors of the control point lattice. In addition, using NURBS basis functions maintains all of the advantages of using NURBS for modeling (Lamousin and Waggenspack, 1994), such as:

1. the evaluation of the NURBS basis functions are computationally stable when the degree increases,
2. the continuities across the NURBS lattice points are automatically maintained for local modification, and
3. the control point lattice can be unevenly divided such that the divided sections can be coarser in areas requiring little deformation and finer in areas requiring higher degrees of deformation.

Direct manipulation, presented by Hsu, Hughes, and Kaufman (Hsu et al., 1992) is another powerful tool for geometric shape deformation. This method uses the pseudoinverse to calculate the corresponding control point movement with respect to the movement of selected data points on the object. It allows the designer to directly deform the geometric object, resulting in more intuitive and more precise shape changes than modifying the control point lattice.

To perform the direct manipulation, the incremental equation modified from Eq. (1) is used to establish the relation of the single data point movement ΔS and the control point lattice changes $\Delta \mathbf{P}$.

$$\Delta S = \frac{\mathbf{N}}{\text{Sum}} \Delta \mathbf{P}^T \quad (3)$$

where \mathbf{N} and $\Delta \mathbf{P}$ are both row vectors with the dimension of the number of total control points. Assume the control point lattice has $(n + 1)$, $(m + 1)$, and $(o + 1)$ number of control points in the three parametric directions with degrees of p , q , and r respectively. The vectors in Eq. (3) are

$$\mathbf{N} = [N_{0,p}N_{0,q}N_{0,r}W_{0,0,0}, N_{0,p}N_{0,q}N_{1,r}W_{0,0,1}, \dots, N_{n,p}N_{m,q}N_{o,r}W_{n,m,o}]$$

$$\mathbf{P} = [P_{0,0,0}, P_{0,0,1}, \dots, P_{n,m,o}]$$

and

$$\text{Sum} = \sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^o N_{i,p}(u)N_{j,q}(v)N_{k,r}(w)W_{i,j,k}$$

which is a scalar. Since the data point movement ΔS is given, the control point lattice movements can be found by the pseudoinverse of \mathbf{N}/Sum that satisfies the following equation

$$\Delta \mathbf{P}^T = \left(\frac{\mathbf{N}}{\text{Sum}} \right)^+ \Delta S \quad (4)$$

In this project, we are only aware of single data point movement, thus the pseudoinverse of $(\mathbf{N}/\text{Sum})^+$ can be determined by

$$\left(\frac{\mathbf{N}}{\text{Sum}} \right)^+ = \frac{\text{Sum} \cdot \mathbf{N}^T}{\|\mathbf{N}\|^2} \quad (5)$$

The NURBS-based free-form deformation and the direct manipulation method have been used as a free-form geometry shape change method. In this project, these methods are applied to manipulate the finite element model such that the designer can specify either the control point or a nodal point of a finite element model as the design variable in order to change the shape and also perform the analysis. Since these two methods provide a unique mathematical relationship between the design variable and the shape changes of the finite element model. Once changes to the nodal coordinates of the finite element model are determined, shape design sensitivity analysis can be used to calculate deformation and stress changes.

Shape Design Sensitivity Analysis

Shape design sensitivities are essential to quickly determine the structural responses when the shape of the model is changed. Shape sensitivities are obtained in two steps. First, changes in the design variables and the resultant movement of the finite element nodal coordinates are calculated resulting in the design velocity field. Next, the sensitivities of the structural responses can be determined using either discrete methods (Arora and Haug, 1979) or domain-continuum methods (Belegundu and Rajan, 1988; Haug et al., 1986).

Design velocity field computation is an important step in determining shape design sensitivity coefficients and updating the finite element mesh for the structural shape “what-if” studies and shape optimization. Improper choice of the design velocity field may lead to erroneous sensitivity results in the second step. Kodiyalam et al. (Kodiyalam et al., 1992) proposed a finite difference method based on automatic mesh generation techniques. Although this method is easy to implement in common CAD package, the topology of the mesh and the linear dependency to the design variables are not easy to maintain. Yao and Choi (Yao and Choi, 1989) solved an auxiliary model with prescribed boundary displacement to determine the design velocity field. Mikaili and Bernard (Mikaili and Bernard, 1992) used Non-Zero Section (NZS) elements on the boundary of a structure to determine the boundary velocity field. Additional finite element solutions are required in all of the above methods. To improve the efficiency of the sensitivity calculation, Choi and Chang (Choi and Chang, 1994) presented a hybrid method that combines the isoparametric mapping method for the boundary velocity field and boundary enforced displacement method to determine the interior or domain velocity field. Chen and Tortorelli (Chen and Tortorelli, 1996) also used isoparametric mapping to determine the boundary velocity field, but the domain velocity field is determined by Laplacian smoothing techniques.

The isoparametric mapping method is an easy and efficient way to determine the design velocity field (Choi and Chang, 1994). This method doesn't require additional finite element calculations or automatic mesh generators but only results in several matrix multiplications. Most CAD packages only support boundary representation, such as IGES file format, for the solid model, therefore the isoparametric mapping method is only useful for determining the boundary velocity field. The method presented in this paper uses the free-form deformation as the basis for the isoparametric mapping technique to determine both boundary and domain velocity fields. It requires less spatial decomposition and easier parametrization than the surface nodal isoparametric mapping method.

In the following, the design velocity field within the NFFD bounding box is derived in order to perform the shape design sensitivity analysis of a finite element model. In this project, the design variable can be either the control points of the NFFD or the nodal points of the finite element model. For the case where the control point is the design variable, the design velocity field can be determined by taking the derivative of Eq. (1) with respect to the control point

$$\frac{\partial S(u, v, w)}{\partial P_{x,y,z}} = \frac{N_{x,p}(u)N_{y,q}(v)N_{z,r}(w)W_{x,y,z}}{\text{Sum}} \quad (6)$$

For the case of direct manipulation, the finite element nodal point $\hat{S}(\hat{u}, \hat{v}, \hat{w})$ is the design variable, and the design velocity field can be obtained from the derivative of Eq. (3) as

$$\frac{\partial S(u, v, w)}{\partial \hat{S}(\hat{u}, \hat{v}, \hat{w})} = \hat{N} \cdot \left(\frac{\hat{N}}{\|\hat{N}\|^2} \right)^T \quad (7)$$

where $\hat{N} = \{N_{i,p}N_{j,q}N_{k,r}W_{i,j,k}\}$ is a row vector of non-vanished basis functions with a dimension of $(p + 1) \times (q + 1) \times (r + 1)$. The index numbers i , j , and k of each component of the vector \hat{N} depend on the span index number of the selected data points' parametric values (Piegl and Tiller, 1995). Let $uspan$, $vspan$, and $wspan$ be the span index numbers for the parametric values (u, v, w) of the selected data point respectively, then $i \in [uspan - p, upsan]$, $j \in [vspan - q, vpsan]$, and $k \in [wspan - r, wpsan]$ due to the fact that the basis functions with the index number inside these regions are non-zero.

Note from Eqs. (6) and (7), since the basis functions will not change during the modification process, the sensitivities of the data point movement with respect to the design variable are

Table 1 Virtual reality peripheral devices

DEVICE	DESCRIPTION	FUNCTION
Fakespace BOOM3C	display device with integrated motion tracking and navigation ability	viewing and navigation
n-Vision VGA Head Mounted Display equipped with an Ascension Flock of Birds tracker	display device with motion tracking	viewing
Ascension Flock of Birds	motion trackers	motion tracking
FakeSpace Pinch Gloves	picking/navigation device	picking/navigation
Virtual Technologies Cyber-glove	picking/navigation device	picking/navigation
StereoGraphics CrystalEyes	display device for stereo images	viewing

constant for both control point or nodal point manipulation methods, thus the design velocity field calculation satisfies the linearity and regularity requirements for the isoparametric mapping method (Choi and Chang, 1994) to update the finite element mesh.

In the virtual environment developed in this project, the designer specifies the design variable by using a picking device, such as the PinchGlove or CyberGlove, to reach into the virtual world and pick a control point or a nodal point of the finite element model. The program then automatically generates the input file for MSC/NASTRAN (User's Manual 1991) design sensitivity analysis solution sequence (SOL 200). The DVGRID entries in the input data file are the boundary design velocity field values which can be obtained by Eqs. (6) and (7) depending on the type of design variable selected.

The sensitivities obtained from NASTRAN are used in a Taylor series to approximate the structural responses of stresses and displacements at each finite element node:

$$\sigma(x + \Delta x) = \sigma(x) + S_{\sigma}(\Delta x) \quad (8)$$

$$D(x + \Delta x) = D(x) + S_D(\Delta x) \quad (9)$$

where

$\sigma(x + \Delta x)$, $D(x + \Delta x)$ = approximate stress, σ , and displacement, D , at the design variable $x + \Delta x$

$\sigma(x)$, $D(x)$ = stress and displacement at the original design variable x obtained from finite element solution

S_{σ} , S_D = stress and displacement sensitivities

Δx = change in the design variable

Since the update Eqs. (8) and (9) are easy and efficient to implement, the designer will be able to obtain the approximated analysis results in real time without performing finite element reanalysis.

Integration of Free-Form Deformation and Shape Sensitivities Into the Virtual Environment

The software used for building the virtual environment is EAI/Sense8's WorldToolKit (WTK) (User's Manual 1996). It consists of a set of C functions that provide drivers for managing the peripherals used for interacting with the virtual world. The hardware peripherals for VR can be categorized into three areas: display devices, picking/navigating devices and trackers. A motion tracker is needed in VR applications in order to pick, navigate or view. Some display devices like the BOOM3C have integrated trackers. Most picking devices such as the Cyber-Glove require separate motion trackers. There are two modes of interaction needed in this application: picking (selecting and moving the geometry) and navigation (viewing the geometry from various viewpoints). This application can use any combination of the devices listed in Table 1 as VR interfaces.

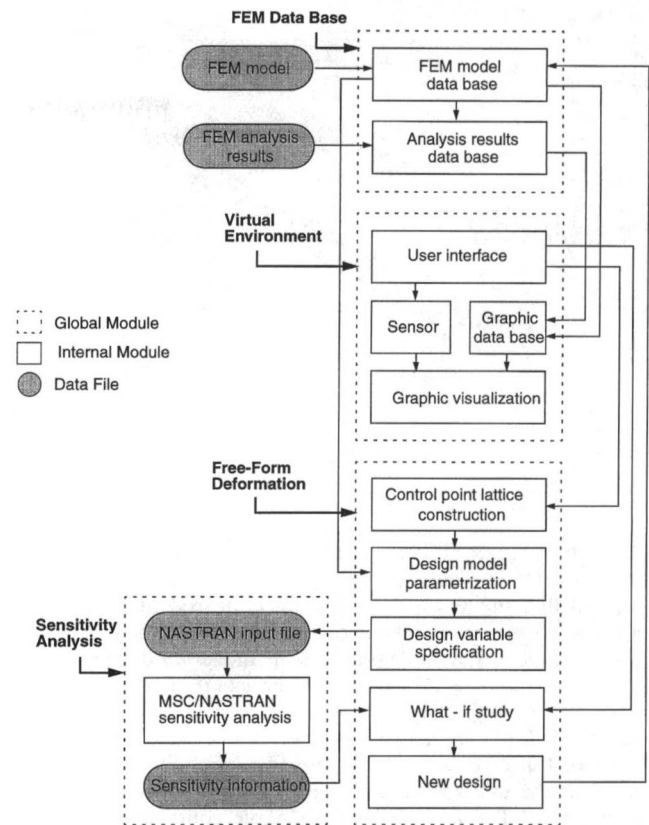


Fig. 2 Program architecture

The architecture of the program includes several global modules: *FEM Data Base*, *Virtual Environment*, *Free-Form Deformation* and *Sensitivity Analysis*. From an object oriented data structure point of view, the global module is essentially a C++ class or hierarchic classes. Each global module contains several local modules which perform independent tasks but share the same private data of the global module. The framework of these modules is illustrated in Fig. 2. The shape design of an engine connecting rod is also presented to show the usage of each module.

The connecting rod shown in Fig. 3 is a quarter model in order to take advantage of the rod's geometric symmetry. To simplify the problem, the boundary condition consists of a clamped condition in the crankshaft end and a uniform pressure of 100 MPa at the pin end toward the crankshaft end to simulate the firing force. Young's modulus and Poisson's ratio are 2.07×10^5 MPa and 0.3 respectively. The purpose of this connecting rod design is to explore a variety of different shapes and find out the impact on the stresses and displacement.

The finite element mesh information needed for the *FEM model data base* can be obtained from any CAD package which can output NASTRAN bulk data format. After the initial FEM solution is obtained, the data for the *Analysis results data base*

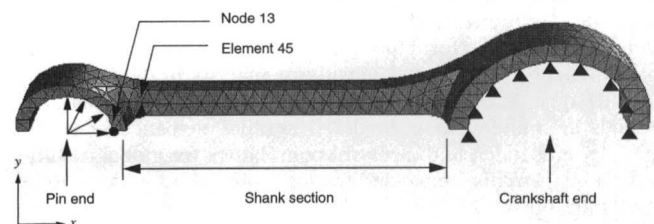


Fig. 3 Quarter model of the engine connecting rod



Fig. 4 Virtual environment for engine connecting rod shape design

are taken from NASTRAN punch files which contain grid point displacements and vonMises stresses. As shown in Fig. 4, the program first inputs the engine connecting rod finite element model and its analysis results through these two modules. The *Graphic data base* module prepares the model for display in the virtual environment. The stress contour data is also generated in this module by linearly interpolating the adjacent grid point stresses to show the color contour on the deformed model.

Figure 4 also shows a 3D menu and a PinchGlove which are served as the interaction and manipulation tools in the virtual environment. The *User interface* module contains those tools that the designer needs to interact with the virtual world. It includes drivers for the VR peripherals including the picking, navigating and viewing devices. A 3D menu system is used to allow the designer to change the virtual display mode, switch between different tasks, etc. The principles of the 3D menu system are directly related to those of the 2D menu system in a window-based applications. The 3D menu system contains multiple entities, such as buttons, sliders, and dialog boxes. In the 2D menu user interface, only the mouse and keyboard can be used to interact with the menu. The 3D menu system is essentially a group of virtual objects floating in the virtual environment. It can be reached by the picking device like a physical control panel, and can be invoked or closed using different hand gestures. Depending on the user's need, the 3D menu can stay fixed in front of the user's viewpoint, or fixed in the virtual world.

The *Graphic visualization* module is used to display the scene of the virtual environment. It displays the graphic data as virtual objects, shows the deformation and color contour stresses, and updates the position and orientation of the viewpoint and the picking device. The *User interface*, *Sensor*, and *Graphic visualization* modules are the only platform dependent modules in this program. For future development on different platforms, only these three modules need to be changed to accommodate the new system.

The *Free-form deformation* module is where the deformed shape of the finite element model and the sensitivity coefficients of the shape design variables are calculated. The internal modules, *Control point lattice construction*, *Design model parametrization*, *Design variable specification*, *What-if study*, and *Update new design* perform the five basic steps required to accomplish the shape design and sensitivity analysis in this program. The designer first constructs the control point lattice that surrounds the finite element model. The finite element model can be fully embedded in the control point lattice for global modification, or partially embedded for local modification depending on the designer's knowledge of the design model. The designer can use the picking device to resize the lattice that surrounds the design model or move the lattice to the desired area for local

modification. In the connecting rod case, we are only aware of the shape change of the shank section, thus the control point lattice is constructed only around this area (see Fig. 5(a)). After the design area has been specified, the *Design model parametrization* module is used to determine the parametric value of the finite element grid points which are embedded inside the control point lattice. The color of the design area will be highlighted after the parametrization process is completed (see Fig. 5(b)). The designer can then choose either a control point or a grid point as the design variable. If a control point is selected as the design variable, the normal free-form deformation method is used to deform the embedded object by changing the control point position, otherwise, the direct manipulation technique will be used to deform the object.

The design velocity field with respect to the selected design variable is calculated when the design variable is specified. The design velocity field is used in the input data file for MSC/NASTRAN sensitivity analysis solution sequence to determine the stress sensitivity information for the structural responses. Here the sensitivities of the grid point displacement and grid point vonMises stress are of interest. Since the finite element analysis of MSC/NASTRAN may take a considerable amount of time, a subprocess is created to run the MSC/NASTRAN in the *Sensitivity Analysis* module so the main program is not interrupted.

When the *Sensitivity Analysis* module is completed, the sensitivity analysis results are brought into the virtual environment

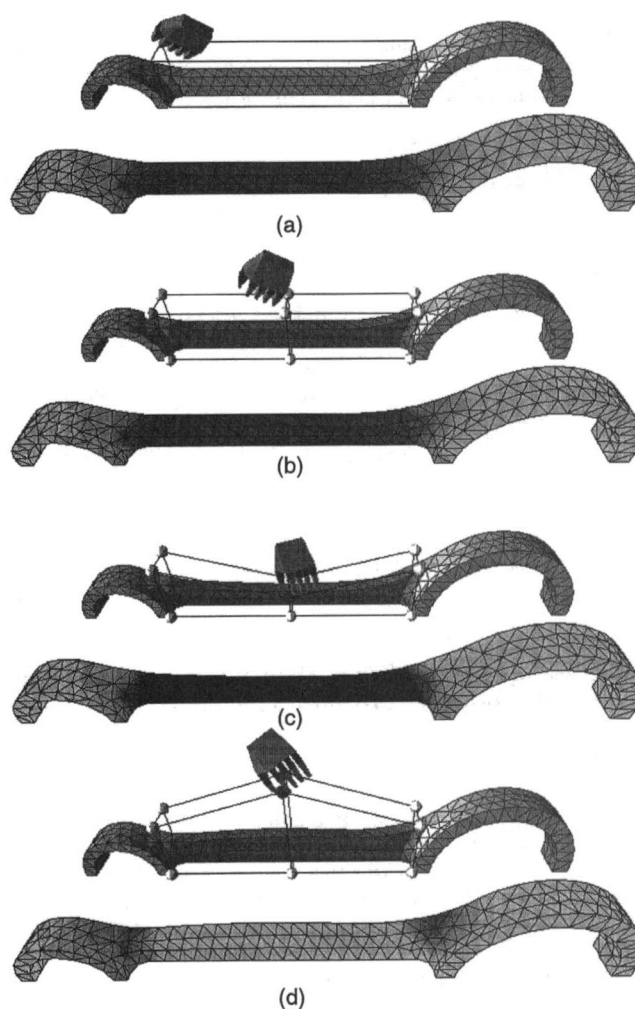


Fig. 5 Process of the shape design (a) design area specification; (b) parametrization and design variable selection; (c) shape change (decreasing design variable); (d) shape change (increasing design variable)

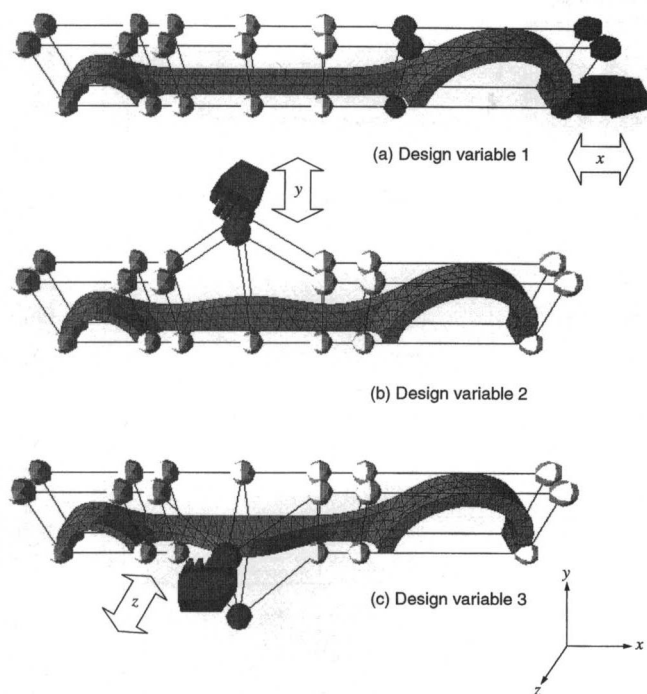


Fig. 6 The layout of the control point lattice and the design variables

automatically in order to perform the what-if studies. In the *What-if study*, the designer can change the design variable by moving the control point or the nodal point of the finite element model by using natural hand motion, and the finite element mesh and analysis results will be updated correspondingly through linear sensitivity approximation without performing a re-analysis in *Update new design* module. As shown in Fig. 5(c) and (d), the designer will be able to move the control point to change the shape of the connecting rod and also view the changes of the structural responses in real time.

Feature Preservation Using NFFD

The generic NFFD method can not arbitrarily be applied in the shape change of the finite element model since it may result in a meaningless design. For example, the round ends of the connecting rod should still be round after shape changes, otherwise, the design and analysis of the modified rod is meaningless. In the previous example, the round ends of the connect rod are preserved by embedding only the shank section of the rod inside the NFFD bounding box. Thus the round ends will never be modified. However, the NURBS volume setting in the previous example results in C1 discontinuity over the boundary of the NURBS volume since the knot vector U only has the degree of 2. This problem can be easily avoided by increasing the degree of the knot vector U . In this section, a NURBS volume setting is proposed to allow the user to modify the width, length and the contour of the shank section without modifying the feature of the round ends:

$$U = \{0.0, 0.0, 0.0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.0, 1.0\}$$

$$V = \{0.0, 0.0, 1.0, 1.0\}$$

$$W = \{0.0, 0.0, 1.0, 1.0\}$$

The layout of the control point lattice is shown in Fig. 6. The design variables are shown as the highlighted control points in Fig. 6(a), (b) and (c) respectively. Note from Fig. 6 that eight control points on the left and right of the control point lattice cover the whole area of the pin end and the crankshaft end respectively. The first design variable is the x direction move-

ment of the eight control points on the right of the lattice, as shown in Fig. 6(a), to change the length of the connecting rod. By moving these control points simultaneously, the feature of the round end can be preserved. Also the movement of these control points will not affect the shape of the pin end. The second design variable is the y direction movement of the two control points as shown in Fig. 6(b). Similarly, the movement of these two control points will not affect the features of the two round ends. Since the knot vector V indicates the linear change in y direction, the interior finite element nodes will be changed linearly with the design variable change in order to maintain the linearity of the overall finite element mesh. The third design variable is the z direction movement of the 2 front control points as shown in Fig. 6(c). It will change the width of the shank section. Also the interior nodes will be changed linearly in z direction to maintain the mesh integrity. Note that the direct manipulation method is not recommended if the feature preservation is important. Since the pseudo inverse method for the direct manipulation method calculates the minimum movements of all control points, it may result in changing all the features of the design model.

Verification of Design Sensitivity

The accuracy of the linear sensitivity approximation of displacement and stress is verified by reanalyzing the modified connecting rod. The nodal point 13 and the element 45, as shown in Fig. 3, contain the highest displacement and the highest element vonMises stress respectively. The finite element solution and the approximated solution of the displacement at node 13 and the stress at element 45 are illustrated in Fig. 7 for three design variables shown in Fig. 6.

All of the exact solutions are nonlinear to the changes of the three design variables, but design variable 3 (as shown in Fig. 7(e) and (f)) is the most nonlinear in the area of interest. The error of the displacement approximation is 2.16 percent when design variable 3 moves 5 mm in $+z$ direction, and 2.85 percent when design variable 3 moves 5 mm in $-z$ direction. The error of the stress approximation is 0.02 percent and 0.04 percent respectively when design variable 3 moves 5 mm in $+z$ and $-z$ directions. The linear approximations are reasonably accurate for small changes of the structural shape, thus we agree that the sensitivity method presented in this paper provides a good approximation for the linear static analysis of this case.

Conclusion and Future Development

Sensitivity methods, finite element analysis, and free-form deformation have been combined within a virtual environment to facilitate structural shape design. NURBS-based free-form deformation (NFFD) methods and direct manipulation techniques are used as the interface between the VR interaction and the finite element model. The addition of analysis results to the virtual environment provides further information to the designer which helps to guide the choice of design changes. The incorporation of sensitivities which are used to approximate the analysis results facilitates real time interaction and what-if studies. Using VR for interactive what-if studies gives the designer an intuitive sense of the performance of the design change, thus it helps the designer to achieve a viable, optimal solution in a timely manner.

Current limitation of the isoparametric mapping method is that it is not easy to maintain the geometry features of the design model. For example a round hole in a beam will become an elliptical hole after deforming the beam. The feature preservation methods used in this research are based on the setting of the NFFD bounding box or the local modification property of a higher order NURBS function to avoid the feature which should be preserved. Future development will focus on the improvement of the isoparametric mapping method used in the

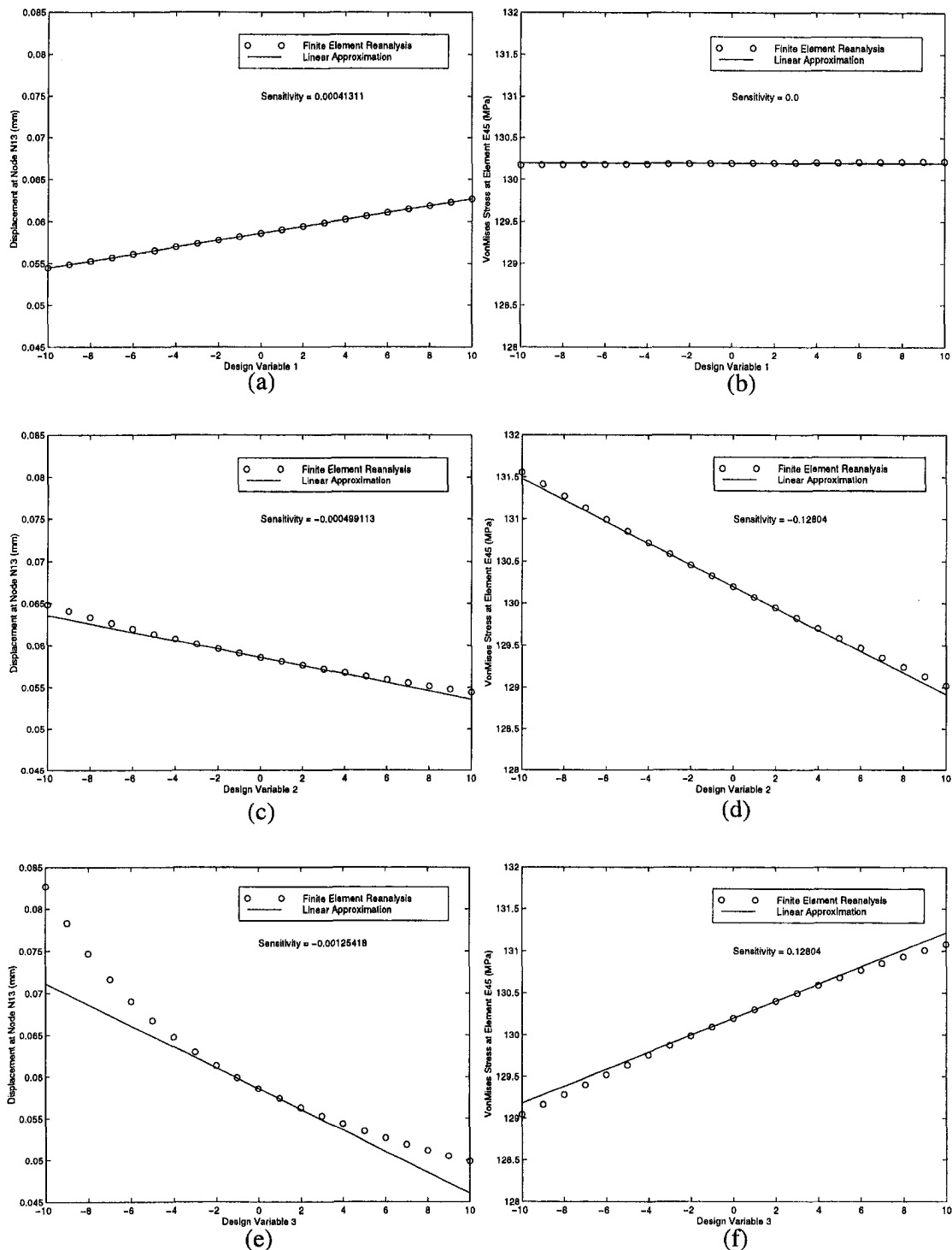


Fig. 7 Comparison plots of linear approximation and finite element reanalysis for three design variables

current free-form deformation technique such that it can accommodate constraints in the geometric boundaries of the model in order to preserve geometry features such as holes or slots. Furthermore, interactive design optimization (Yeh and Vance, 1995) will also be incorporated in this program to provide yet another tools to help guide the designer to achieve the most optimal design.

Acknowledgements

This project is funded through a National Science Foundation Research Initiation Award (DDM-9309663). The authors wish

to thank the Iowa Center for Emerging Manufacturing Technology at Iowa State University for the use of the equipment in the Visualization Lab.

References

- Arora, J. S., and Haug, E. J., 1979, "Methods of Design Sensitivity Analysis in Structural Optimization," *AIAA Journal*, Vol. 17, No. 9, pp. 970–974.
- Aukstakalnis, S., and Blatner, D., 1992, *Silicon Mirage: The Art and Science of Virtual Reality*, Peachpit Press, Berkeley.
- Belegundu, A. D., and Rajan, S. D., 1988, "A Shape Optimization Approach Based on Natural Design Variables and Shape Functions," *Computer Methods in Applied Mechanics and Engineering*, Vol. 66, No. 1, pp. 87–106.

- Chen, S. S., and Tortorelli, D. A., 1996, "Three-Dimensional Shape Optimization With Variational Geometry," *6th Annual AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Bellevue, WA, pp. 99–113.
- Choi, K. K., and Chang, K. H., 1994, "A Study of Design Velocity Field Computation for Shape Optimal Design," *Finite Elements in Analysis and Design*, Vol. 15, pp. 317–341.
- Griessmair, J., and Purgathofer, W., 1989, "Deformation of Solids with Trivariate B-Splines," *Proceeding of Eurographics 89*, Elsevier Science Publishers, North-Holland, pp. 137–148.
- Gupta, R., 1996, "Survey on Use of Virtual Environments in Design and Manufacturing," *Proceedings of the 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference*, Aug. 18–22, 1996, Irvine, CA.
- Haug, E. J., Choi, K. K., and Komkov, V., 1986, *Design Sensitivity Analysis of Structural Systems*, Academic Press, New York.
- Hardee, E., Chang, K. H., Choi, K. K., Yu, X., and Grindeanu, I., 1996, "A CAD-Based Design Sensitivity Analysis and Optimization for Structural Shape Design Applications," *6th Annual AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Bellevue, WA, pp. 77–87.
- Hsu, W. M., Hughes, J. F., and Kaufman, H., 1992, "Direct Manipulation of Free-Form Deformations," *Computer Graphics*, Vol. 26, No. 2, pp. 177–184.
- Kodiyalam, S., Kumar, V., and Finnigan, P. M., 1992, "Constructive Solid Geometry Approach to Three-Dimensional Structural Shape Optimization," *AIAA Journal*, Vol. 30, No. 5, pp. 1408–1415.
- Lamoussin, H. J., and Waggenspack, Jr., W. N., 1994, "NURBS-Based Free-Form Deformations," *IEEE Computer Graphics and Applications*, Vol. 14, No. 6, pp. 59–65.
- Lasser, D., 1995, "Rational Tensor Product Bézier Volumes," *Computer Math. Applic.*, Vol. 29, No. 8, pp. 95–108.
- Mikaili, A., and Bernard, J. E., 1992, "Structural Redesign Using Nearly Zero Section Reinforcements," *Journal of Finite Elements in Analysis and Design*, Vol. 11, No. 3, pp. 221–233.
- MSC/NASTRAN User's Manual Version 68, 1991, The MacNeal Schwendler Corporation, Los Angeles, CA.
- Piegl, L., and Tiller, W., 1995, *The NURBS Book*, Springer-Verlag, New York, p. 68.
- Sederberg, T. W., and Parry, S. R., 1986, "Free-Form Deformation of Solid Geometric Models," *Computer Graphics*, Vol. 20, No. 4, pp. 151–160.
- Tits, A. L., Lawrence, C., and Zhou, J. L., 1996, "User's Guide for CFSQP Version 2.4: A C Code for Solving (Large Scale) Constrained Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality Constraints," Electrical Engineering Department, *Institute for System Research TR-94-16rl*, University of Maryland, College Park.
- Vance, J. M., 1996, "Virtual Reality: What Potential Does It Hold for Engineering?," *Current Advances in Mechanical Design and Production*, Sixth Cairo University International MDP Conference, Cairo, Jan. 2–4, 1996, Elsevier Science, New York, N.Y., pp. 333–348.
- WorldToolKit Reference Manual, Release 6, 1996, Sense8 Corporation, Sausalito, CA.
- Yao, T. M., and Choi, K. K., 1989, "3D Shape Optimal Design and Automatic Finite Element Regridding," *International Journal of Numerical Methods in Engineering*, Vol. 28, No. 2, pp. 369–384.
- Yeh, T. P., and Vance, J. M., 1995, "Interactive Design of Structural Systems in a Virtual Environment," *Proceedings of 24th Midwest Mechanics Conference*, Iowa State University, Ames, Iowa, pp. 185–187.