NONLINEAR ADAPTIVE ANALYSIS VIA QUASI-NEWTON APPROACH WITH MSC/NASTRAN

Ortwin Ohtmer
Department of Mechanical Engineering
California State University, Long Beach, U.S.A.

ABSTRACT

The Quasi-Newton method has proven to be the most efficient optimization method. The purpose of this paper is to apply this numerical procedure for optimization problems as well as large deflection analysis and animation. A FORTRAN program developed to calculate constrained optimization problems is used as the basic code within an iterative nonlinear adaptive analysis. The new numerical procedure calculates the displacements of an elastic structure due to given loading conditions. Then the displacements are added to the joint coordinates. In the deformed position the degrees of freedom of the structure are supported and the negative displacements are applied as loadings, to move the structure back to the old undeformed position. The difference of the reaction forces in both positions specifies the geometric nonlinear adaptive loading conditions. These additional forces are applied in an iteration procedure, until equilibrium is achieved. The software ME-BANK (Mechanical Engineering Program-Bank), written in C-language, was developed to execute MSC/NASTRAN and a constrained optimization FORTRAN-code via the SYSTEM-function within an iteration procedure.

INTRODUCTION

Within the last 20 years finite element concepts, and recently also boundary element formulations, have proven to be the most reliable analysis tools for static-, dynamic-, heat transfer-, fluid dynamic-, and electromagnetic problems. Improvements on the numerical procedures for example eigenvalue/eigen vectoranalysis via LANCZOS-algorithms [1], as well as increases in storage and computer efficiency, have made the finite element and boundary element tools affordable for everyone. The design optimization procedure is based on an accurate finite element analysis. This very powerful tool can be used for a large variety of problems (shape optimization, static dynamic, or buckling optimization) to specify many design variables and to not violate the nonlinear allowable stress (displacement) constraints.

The most powerful numerical procedure for optimization problems is the Quasi-Newton method [3], mainly known as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update. Recently, Quasi-Newton methods have also been applied with great success for minimization of functions, solution of nonlinear equations, design optimization, and large deflection analysis. In this paper, we intend to apply the Quasi-Newton method to solve arbitrary nonlinear finite element problems by adapting equilibrium via a linear finite element code. At the same time, this method is applied to generate the keyframes [4] of animation of elastic, plastic or kinematic deflections. The gradient projection method, gradient method, and the steepest descent method have been successfully applied in the area of optimal design for many years. Quasi-Newton methods have been developed recently. Based on statements in the state-of-the-art book Optimum Design [3], the Quasi-Newton methods are considered to be efficient, reliable and generally applicable. They have been extensively evaluated against other named methods. It was found that the Quasi-Newton methods are far superior to others due to the following reasons: The Quasi-Newton procedure requires the computation of only first derivatives. By making use of information obtained from previous iterations, however, convergence towards the minimum is speeded up. An approximation to the matrix of second derivatives can therefore be generated. These methods are learning processes as they accumulate the information from previous iterations. In this regard, the methods presented have desirable features of both the steepest descent and the Newton methods. They are called Quasi-Newton or update methods. First-order derivatives are used to generate approximations for a matrix of second partial derivatives for a function $f(\{x\})$ called the Hessian matrix. A new very efficient, modified Quasi-Newton approach for large deflection finite element analysis and animation is developed and successfully applied.

1. THE QUASI-NEWTON NUMERICAL PROCEDURE

Theoretical aspects and the development of the method by several researchers are given in Philip E. Gill, Walter Murray, and Margaret H. Wright's Practical Optimization [7]. Many practical applications are discussed in Jabir S. Arora's Introduction to Optimum Design [3]. Summaries of the Quasi-Newton numerical procedure can be found in Todd D. Coburn's The Quasi-Newton Method in Optimization [5].

1.1 The Newton Method

The algorithms methods are discussed based on a quadratic function of the objective function F to be minimized [7]. There are two major justifications for choosing a quadratic function: its simplicity and, more importnat, the success and efficiency in practice of the method based on it.

In the same way, a parameter notation, taking into account second derivatives, is preferred for the numerical integration of integral equations[8]. The very accurate numerical intefration based on modified cubic spline functions is also achieved in [9] by specifying continuous first and second order derivatives. A cubic spline is a deformed beam structure; the first derivative represents the tangent of the elastic curve, while the second derivative represents the moment distribution or and accurate estimate of the curvature of the beam.

If first and second derivatives of the objective function F to be minimized are available, a quadratic function of F can be obtained by taking the first three terms of the Taylor-series expansion about the current point in an n-dimensional Hilbert-space. The vector $\{p\}_k$ is specifying the $\{x\}_k$ search direction.

$$F(\{x\}_k + \{p\}_k) = F(\{x\}_k) + \left(\{g\}_k^T\{p\}_k\right) + \frac{1}{2}\left(\{p\}_k^T[G]_k\{p\}_k\right) \tag{1.1}$$

with $\{g\}_k = \left\{\frac{\partial F}{\partial p_i}\right\}_k$, $\{x\}_k = \begin{Bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{Bmatrix}_k$, and $\{p\}_k = \begin{Bmatrix} p_1 \\ p_2 \\ \cdot \\ \cdot \\ \cdot \\ p_n \end{Bmatrix}_k$

where $i = 1, 2, \ldots n$

$[G]_k$ is named the Hessian matrix

$$[G]_k = \begin{bmatrix} \frac{\partial^2 F}{\partial p_1^2} & \frac{\partial^2 F}{\partial p_1 \partial p_2} & \cdots & \frac{\partial^2 F}{\partial p_1 \partial p_n} \\ \cdot & \cdot & & \\ \cdot & \cdot & & \\ \cdot & \cdot & & \\ \frac{\partial^2 F}{\partial p_1 \partial p_n} & \frac{\partial^2 F}{\partial p_2 \partial p_n} & \cdots & \frac{\partial^2 F}{\partial p_n^2} \end{bmatrix}$$

Within the context of the model algorithm, it is helpful to formulate the function (equation (1.1)) in terms of the vector $\{p\}_k$ (the step to the miminum) rather than the predicted minimum itself. The minimum of the right-hand side of equation (1.1) will be achieved if $\{p\}_k$ is a minimum of the quadratic function:

$$\begin{aligned} \phi(\{p\}_k) &= F(\{x\}_k + \{p\}_k) - (F\{x\}_k) \\ &= \left(\{g\}_k^T\{p\}_k\right) + \frac{1}{2}\left(\{p\}_k^T[G]_k\{p\}_k\right) . \end{aligned} \tag{1.2}$$

Then $$\frac{\partial \phi}{\partial \left(\{p\}_k^T\right)} = 0$$

$$= \frac{\partial \left(\left(\{p\}_k^T\{g\}_k\right) + \frac{1}{2}\left(\{p\}_k^T[G]_k\{p\}_k\right)\right)}{\partial \left(\{p\}_k^T\right)} , \tag{1.3}$$

or

$$\{[G]_k\{p\}_k\} = -\{g\}_k = -\left\{\frac{\partial F}{\partial p_i}\right\}_k . \tag{1.4}$$

The stationary point $\{p\}_k$ satisfies the linear system of equation (1.4). $\left(\{p\}_k^T[G]_k\{p\}_k\right)$ in equation (1.1) is named the curvature. The gradient $\{g\}_k = \{\partial F/\partial p_i\}_k$ is the direction vector in n-dimensional Hilbert-space. A minimization algorithm in which $(p)_k$ is defined by equation (1.4) is termed the Newton method, and the solution of equation (1.4) is called the Newton direction.

1.2 Quasi-Newton Method

The key to the success of Newton-type methods is the curvature information provided by the Hessian matrix, which allows a local quadratic function of F to be developed. Quasi-Newton methods are based on the idea of building up curvature information as the iterations of a descent method proceed, using the observed behavior of F and {g}. The theory of Quasi-Newton methods is based on the fact that an approximation to the curvature on a nonlinear function can be computed without explicitly forming the Hessian matrix.

In order to minimize the objective function F, let $\{s\}_k$ be the vector step taken from $\{x\}_k$ and consider expanding the gradient function about $\{x\}_k$ in a Taylor series along $\{s\}_k$:

$$\left\{g\left(\{x\}_k + \{s\}_k\right)\right\} = \left\{g\left(\{x\}_k\right)\right\} + \left\{[G]_k\{s\}_k\right\} , \tag{1.5}$$

with the Hessian matrix $[G]_k$ and the gradient vector $\{g\}$

$$\{g\}_k = \{\partial F/\partial p_i\} \tag{1.6}$$

Due to equation (1.1) the curvature of F along $\{s\}_k$ is given by $\{s\}_k^T[G]_k\{s\}_k$, which can be approximated using only first-order information. Multiplying equation (1.5) by $\{s\}_k^T$ from the left, we get:

$$\left(\{s\}_k^T[G]_k\{s\}_k\right) = \left(\left\{\left\{g\left(\{x\}_k + \{s\}_k\right)\right\} - \left\{g\left(\{x\}_k\right)\right\}\right\}^T\{s\}_k\right) \tag{1.7}$$

At the beginning of the k^{th} iteration of a Quasi-Newton method, an approximate Hessian matrix $[B]_k$ is available, which is intended to reflect the curvature information already accumulated. If $[B]_k$ is taken as the Hessian matrix of a quadratic function, the search direction $\{p\}_k$ is the solution of a linear system analogous to equation (1.4):

$$[B]_k\{p\}_k \doteq -\{g\}_k = -\left\{\frac{\partial F}{\partial p_i}\right\}_k . \tag{1.8}$$

The initial Hessian approximation $[B]_0$ is usually taken as the unit matrix if no additional information is available.

After $\{x\}_{K+1}$ has been computed, a new Hessian approximation $[B]_K$ is obtained by updating $[B]_K$ to take account of the newly acuired curvature information. An update formula is:

$$[B]_{k+1} = [B]_k + [U]_k \tag{1.9}$$

where $[U]_K$ is the update matrix. Let the vector $\{s\}_K$ denote the change in x during the k^{th} iteration. Then we obtain equation (1.10) with the step length α_K:

$$\{s\}_k = \{x\}_{k+1} - \{x\}_k = \alpha_k\{p\}_k \tag{1.10}$$

and

$$\{y\}_k = \{g\}_{k+1} - \{g\}_k \tag{1.11}$$

where $\{y\}_K$ is the change in gradient.

The standard condition required of the updated Hessian approximation is that it should approximate the curvature of F along $\{s\}_K$. Based on equations (1.5) and (1.6) $[B]_{K+1}$ is required to satisfy the Quasi-Newton condition:

$$[B]_{k+1}\{s\}_k = \{y\}_k = \{g\}_{k+1} - \{g\}_k \tag{1.12}$$

During a single iteration, new information is obtained about the second-order behavior of F along only one direction; thus we would expect $[B]_{k+1}$ to differ from $[B]_k$ by a matrix of low rank. In fact, the Quasi-Newton condition can be satisfied by adding a rank-one matrix to $[B]_k$. Assume that

$$[B]_{k+1} = [B]_k + \left[\{u\}\{v\}^T\right] \tag{1.13}$$

where {u} and {v} are vectors. Similarly a symmetric rank-two update can be derived which finally results in the well known Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula specified in (1.14). The theoretical background is given in [7]. Many applications are summarized in [14].

$$[B]_{k+1} = [B]_k + \frac{\left[\{y\}_k\{y\}_k^T\right]}{\left(\{y\}_k^T\{s\}_k\right)} + \frac{\left[\{g\}_k\{g\}_k^T\right]}{\left(\{g\}_k^T\{p\}_k\right)} \tag{1.14}$$

1.3 Efficient Large Deflection Analysis via Quasi-Newton Method

Quasi-Newton update methods are considered the most sophisticated methods of the Newton iterative solution scheme and represent the culmination of extensive algorithm development for Newton iterations by approximating the inverse Hessian matrix in place of the true inverse that is required in the Newton method. Quasi-Newton methods have been efficiently and successfully used in nonlinear optimization problems.

By the Quasi-Newton method (BFGS update), the information acquired during the iteration is used to modify the inverse stiffness matrix. This approximate update to the inverse stiffness matrix results in a secant modulus in the search direction. As these updates accumulate, the BFGS method renders a stiffness matrix resembling tangential stiffness in the limit. When combined with the line search, the performance of the BFGS update with respect to effectiveness and efficiency depends largely on the implementation. The basic concept of the Quasi-Newton method is to build an approximate inverse Hessian stiffness matrix using information gathered during the descent process. The current approximation is used at the next iteration to define the next feasible direction in the modified Newton method. According to the first interaction, a feasible direction of descent is given by:

$$\{d^1\} = [K_0]^{-1} \{R(u^0)\}$$

where $\{R\}$ is an error vector to be minimized and K_0 is a Hessian matrix. In the absence of a line search or Quasi-Newton update, the second iteration would lead to the next feasible direction:

$$\{d^2\} = [K_0]^{-1} \{R(u^1)\} \text{ , with } \{u^1\} = \{u^0\} + \{d^1\}$$

Consider a Taylor series expansion of the load error $\{R\}$ about $\{u^i\}$; we get

$$\{R\} = \{R(u^i)\} - [K]\left\{\{u\} - \{u^i\}\right\} + \{0(h)\} \qquad (1.15)$$

where $$\{R(u)\} = \{P\} - \{F(u)\}$$

with $\{F(u)\}$ being the aggregate vector of element nodal forces and $\{P\}$ the external load vector. Assuming that the load stiffness due to following forces is negligible, the stiffness matrix is formed by

$$[K] = \frac{\partial\{R\}}{\partial\{u\}^T} = \frac{\partial\{F\}}{\partial\{u\}^T}$$

In equation (1.15) the data from two points, $\{u^{i-1}\}$ and $\{u^i\}$, should provide some information about $[K]$, because they should satisfy:

$$\{Y\} = [K]\{\delta\} \quad \text{or} \quad \{\delta\} = [K]^{-1}\{Y\} \qquad (1.16)$$

with

$$\{\delta\} = \{u^i\} - \{u^{i-1}\} \quad \text{and} \qquad (1.17)$$

$$\{Y\} = \{R^i\} - \{R^{i-1}\} \; . \qquad (1.18)$$

Based on the data obtained during the iterative procedure, we construct successive approximations to $[K]^{-1}$.

The Quasi-Newton scheme is the BFGS update in (1.14). By this scheme, the Hessian matrix is updated by adding two symmetric rank-one matrices at each iteration. Therefore, the scheme is a rank-two correction procedure:

$$[K_{i+1}]^{-1} = [K_i]^{-1} + \frac{\{\delta\}\{\delta\}^T}{\{\delta\}^T\{\gamma\}} - \frac{[K_i]^{-1}\{\gamma\}\{\gamma\}^T[K_i]^{-1}}{\{\gamma\}^T[K_i]^{-1}\{\gamma\}} \tag{1.19}$$

Note that the formula simply satisfies equation (1.16) while preserving the positive stiffness and symmetry of $[K]^{-1}$. Equation (1.19) is equivalent to equation (1.14).

It is also possible to update approximations to the stiffness matrix itself, rather than its inverse. Recalling the complementary roles of [K] and $[K]^{-1}$ with respect to the Quasi-Newton vectors in equation (1.16), the formula for [K] is found by interchanging $\{\delta\}$ and $\{\gamma\}$:

$$[K_{i+1}] = [K_i] + \frac{\{\gamma\}\{\gamma\}^T}{\{\gamma\}^T\{\delta\}} - \frac{[K_i]\{\delta\}\{\delta\}^T[K_i]}{\{\delta\}^T[K_i]\{\delta\}} \tag{1.20}$$

Equation (1.20) is identical to equation (1.14).

Inverting equation (1.20), we obtain the so-called BFGS update formula for the inverse Hessian matrix:

$$[K_{i+1}]^{-1} = [K_i]^{-1} + \left[1 + \frac{\{\gamma\}^T[K_i]^{-1}\{\gamma\}}{\{\gamma\}^T\{\delta\}}\right]\frac{\{\delta\}\{\delta\}^T}{\{\delta\}^T\{\gamma\}} - \frac{\{\delta\}\{\gamma\}^T[K_i]^{-1} + [K_i]^{-1}\{\gamma\}\{\delta\}^T}{\{\gamma\}^T\{\delta\}} \tag{1.21}$$

Another way to express equation (1.21) for the j[th] BFGS update is [11]:

$$[K_j]^{-1} = [C_j]^T[K_{j-1}]^{-1}[C_j] + z_j\{\delta_j\}\{\delta_j\}^T \qquad (1.21$$

where

$$[C_j] = [I] - z_j\{\gamma_j\}\{\delta_j\}^T \quad \text{and} \quad z_j = \frac{1}{\{\delta_j\}^T\{\gamma_j\}}$$

This is a recurrence formula which is applicable to every pair of Quasi-Newton (QN) vectors. We note that the index j for the BFGS update may be different from the iteration index i. The inverse stiffness matrix is assumed to be symmetric and positive definite throughout the derivation.

We can easily verify that the Hessian matrix $[G]_k$ in equation(1.14)is equal to the tangential stiffness matrix [K] in equation (1.20), that $\{s\}_k = \{x\}_{k+1} - \{x\}_k = \alpha_k\{p\}_k$ in equation (1.10) is equal to $\{u^i\} - \{u^{i-1}\} = \alpha\{d^i\}$ in equation (1.17), and that $\{y\}_k = \{g\}_{k+1} - \{g\}_k$ in (1.11) is equal to $\{R^i\} - \{R^{i-1}\}$ in equation (1.18).

1.4 Line Search Method

The Quasi-Newton method involves the use of first order approximations for the second order information function. This results in the rapid convergence rates of second order methods without the problems associated with evlauating these second order differentials. For this reason, Quasi-Newton methods are very efficient and fast of the optimization methods.

Minimizing the total potential energy of structure applying the Quasi-Newton method combined with the line search method is the most efficient optimization procedure analyzing geometric nonlinear problems. The line search method is well established as a basic descent method in nonlinear analysis. The method has been used to imporve the rate of convergence in nonlinear iterations. A description of the line search method is given in [13] and [14].

The process of determining the local minimum point in a given direction is called the line search. Considering the i^{th} iteration, the new solution set is determined by

$$\{u^i\} = \{u^{i-1}\} + \alpha\{d^i\} \quad \text{with} \quad \{d^i\} = [K]^{-1}\{R^{i-1}\} \tag{1.22}$$

where a positive search parameter (α) is determined such that the vectors $\{R^i\}$ and $\{d^i\}$ are orthogonal.

Equation (1.23) represents a linear interpolation in terms of {u} and {R}, which is the basis of the line search method. The sear ch parameter α is defined in (1.24).

$$\{u\} = \{u^i\} - \left[\frac{\{\{u^i\} - \{u^{i-1}\}\}}{\{\{R^i\} - \{R^{i-1}\}\}^T}\right]\{R^i\} \tag{1.23}$$

$$\alpha = \alpha^* - 1 = \frac{(-\{R^i\}^T\{d^i\})}{(\{\{R^i\} - \{R^{i-1}\}\}^T\{d^i\})} \tag{1.24}$$

Within a finite element analysis the gradient vector {u} and the search parameter α are computed via equations (1.23) and (1.24). The calculation of the "load correction vector" {R} is explained in chapter 2.

2. CALCULATION OF "LOAD CORRECTION VECTOR {Y} VIA LINEAR INVERSE EQUILIBRIUM ANALYSIS

In equation (1.18) {y} = {R } - {R } is introduced as the "load correction vector" due to large deflection increments $\{\delta\} = \{u^i\} - \{u^{i-1}\}$. Based on a given external load vector {P}, the internal force and reaction forces $\{R^{i-1}\}$ are calculated within the back substitution of a linear finite element analysis for the unknown displacement vector.

To estimate the error vector {Y} due to large deflections caused by large external loads, we can very efficiently analyze the "inverse" problem. We add the displacements (translations only assuming negligible rotations) to the joint coordinates. We obtain the deformed position of the structure. Then we support the degrees of freedom and apply the negative displacements as loadings. The linear static analysis at the deformed position moves the structure automatically back to the previous position. The calculation of the supports or negative reaction forces $\{R^i\}$, the forces necessary to move the structure in the previous position, is very efficient, because only the backsubstitution with known displacements is performed solving each row of the basic linear system of equations uncoupled.

For small deflections $\{\delta\} = \{u^i\} - \{u^{i-1}\}$, the force correction vector $\{\gamma\} = \{R^i\} - \{R^{i-1}\}$ is nearly zero, indicating "linear analysis." For larger deflections (nonlinear analysis), the force correction vector $\{\gamma\}$ is increasing considerably.

The so-called Quasi-Newton vectors $\{\delta\}$ and $\{\gamma\}$ are applied in an iteration procedure to correct the displacements by minimizing the potential energy (in dynamics also kinetic energy). A tolerance parameter controls the iteration, to decide when convergence is achieved, or to stop if the procedure is diverging. The appropriate criteria are specified in [13]. The "change in design" (equation 1.10) is replaced by the displacement increments $\{\delta\}$, and the "change in gradient" (equation 1.10) is now represented by the force correction vector {Y} divided the displacement increment $\{\delta\}$; $(\{\gamma i/\delta i\})$. The appropriate adjustments have to be coded within the source code of the Quasi-Newton numerical iteration procedure, which will be outlined.

Relating the displacements to the temperature distribution in heat transfer analysis (velocities in fluid dynamics) and the forces
to heat sources (fluid sources in fluid dynamics), the outlined procedure can easily be extended to nonlinear heat-transfer and fluid-dynamics problems. The calculation of the force correction vector is demonstrated on the following pages considering large deflections for a two-truss structure and a cantilever beam.

The following example shows the solution for deflection of a two-truss structure. We have a two-truss structure as shown in figure 1. Joints 1 and 3 are supported. A load F is applied at joint 2 . Joint 2 moves to joint 2'. Then we support joint 2' and apply the negative displacements as loadings. x and y are the horizontal and vertical displacements. When s = 1.5 m, h = 1.0 m, A = 0.00001 m^2, E = 2.07 x 10^{11} N/m, θ = 30 degrees, L = 1.25 m. We set up three loads from small to large: F1 = 10.0 kN, F2 = 100.0 kN, and F3 = 300.0 kN. Using MSC/NASTRAN , displacements, loads or reactions are displayed in table 5.

We calculate displacements at joint 2 by a linear system of equations to verify the results from the MSC/NASTRAN input:

$$u_{x_2} = \frac{8660}{0.72} \cdot \frac{1}{k} = 1.2028 \times 10^4/k = 7.2632 \times 10^{-3} \text{ m}$$

$$u_{y_2} = \frac{-5000}{1.28} \cdot \frac{1}{k} = -3.9063 \times 10^3/k$$

$$= -2.3588 \times 10^{-3} \text{ m} .$$

Comparing u_{x2} and u_{y2} with those in table 5, we see that they are matched.

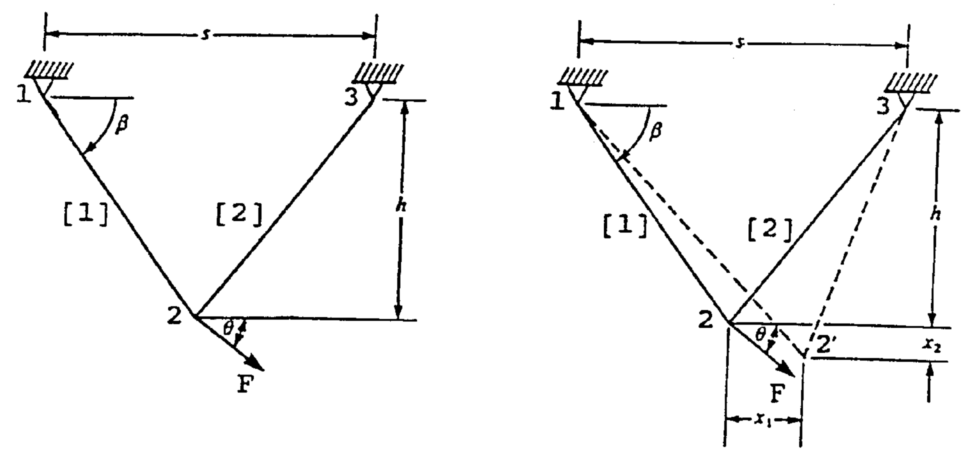

Figure 1. Two-bar structure: left, undeflected shape; right, deflected shape.

Table 1 shows that for a small force F1 (appendix B), the displacements at joint 2 are small, the reactions at joint 2 are also small, and the force correct vectors approach zero; so the deformation is linear.

Table 1 also shows that for a large force F2 (appendix C) or F3 (appendix D), the displacements at joint 2 are large, the reactions are also large, and the force vectors increase considerably; so the deformation is considered as nonlinear analysis. Therefore, the Quasi-Newton method will be applied to optimize the solutions.

Table 1.--Deformation Results for Two-Truss Structure at Joint 2

Load F (kN)	Load F_x (kN)	Reaction R_x (kN)	Force Correct Vector {Y} (kN)	Load F_y (kN)	Reaction R_y (kN)	Force Correct Vector {Y} (kN)	Dis-place-ment u_x (m)	Dis-place-ment u_y (m)
10	8.66	- 8.62	0.04	- 5.0	4.99	-0.01	0.007263	-0.002358
100	86.6	- 82.08	4.52	- 50.0	49.79	-0.21	0.07263	-0.02358
300	259.80	-216.88	42.92	-150.0	159.20	9.20	0.21789	-0.070765

The following example shows the solution for the deflection of a cantilever beam. The cantilever beam shown in figure 2 is supported (fixed) at joint 1. It is divided into 10 members. Each member of the beam is equal. The cross-sectional area of beam A is 2 x 2 in. The length of beam L is 100.0 in. Young's modulus E is 2.1×10^6 lb/in.^{2}.

Step 1: As shown in figure 2, a load (joint force) F is applied at joint 11. Using an associated MSC/NASTRAN code for a linear static solution, the calculated displacements are listed.

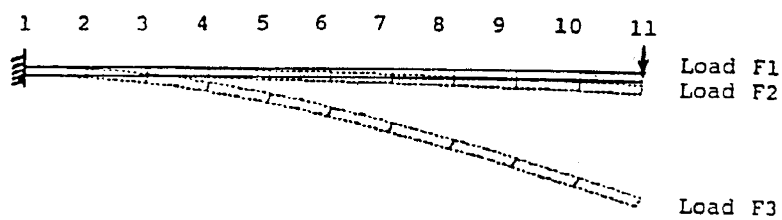

Figure . A 10-member cantilever beam.

The deformed beam is shown in figure 2 (dashed line).

Step 2: We consider the deformed position from step 1 as a start position. Also, the boundary conditions are changed. Joints 2 to 11 are changed from status-free to status-support, but moments Z are released. Applied negative displacements from step 1 become loads (joint displacement loads). Member properties, constants and member incidences remain unchanged. Applying the modified NASTRAN code, the calculated reactions are listed. The deformed beam is back to the original position of step 1 or horizontal, shown in figure 2 (solid line).

For linear deformations, the value of joint load F at the free end (joint 11) should equal the value of joint reaction R at the fixed end (joint 1). The reactions of in-between joints should be zero. For nonlinear deformations, the joint load at the free end (joint 11) is not equal to the joint reaction at the fixed end (joint 1). The joint reactions occur not only at two ends but also at the in-between joints (joints 2 to 10). The sum of the reactions at joints 2 to 10 are very small, so are ignored. We consider this deflection as linear deformation.

In order to have a clear view, we set up three different loads: F1 = 15.0 lbs (small load), F2 = 150.0 lbs (middle load), and F3 = 1,500.0 lbs (large load). The calculated displacements and loads or reactions are listed in tables 2, 3, and 4.

Using the MSC/NASTRAN code and applying a small load F1, the maximum displacement (at joint 11) is 0.111 in., which is small compared to the length of the beam (100.0 in.). As table 2 shows, the reactions in the X direction (load F1 direction), the reaction at joint 1 is nearly equal to that at joint 11, the reactions at joints 2 to 10 are very small, so are ignored. We consider this deflection as linear deformation.

Using the MSC/NASTRAN code and applying a middle load F2 or large load F3, the maximum displacement (at joint 11) is 1.11 in. or 11.1 in. which is large compared to the length of the beam (100.0 in.). From table 3 or 4, we see the reactions not only appear at two ends (joints 1 and 11) but also at in-between joints (joints 2 to 10). The

Table 2.---Deformed Cantilever Beam Results, F1 = 15.0 lb

Joint	X Force	Y Force	Z Moment	Displacement
1	0.2176416	-14.9951954	-1500.0228271	0.000000
2	1.5779505	- 0.0609638	0.0000036	0.001618
3	2.6833925	0.1812560	- 0.0000056	0.006251
4	3.3384278	- 0.2494498	- 0.0000032	0.013562
5	3.5951750	0.1467153	0.0000038	0.023218
6	3.5133188	- 0.1017839	0.0000094	0.034883
7	3.1495285	0.2196821	0.0000084	0.048222
8	2.5637851	- 0.2744259	0.0000030	0.062900
9	1.7989372	- 0.2352806	0.0000006	0.078584
10	0.9163748	0.6587136	0.0000166	0.094937
11	-23.354532	14.7107325	0.0000000	0.111620

Table 3.---Deformed Cantilever Beam Results, F2 = 150.0 lb

Joint	X Force	Y Force	Z Moment	Displacement
1	21.7640	-150.0390	-15000.345	0.00000
2	157.7890	- 1.4550	0.0001457	0.01618
3	268.3090	- 0.6138	0.0000658	0.06251
4	333.7700	- 6.6989	- 0.0000502	0.13562
5	359.3950	- 4.2085	- 0.0001752	0.23218
6	351.1680	- 7.4680	- 0.0000522	0.34883
7	314.7710	- 4.3438	- 0.0001457	0.48222
8	256.2030	- 8.4250	- 0.0001137	0.62900
9	179.7600	- 6.6560	- 0.0000113	0.78584
10	91.5587	4.4030	0.0000304	0.94937
11	-2334.4870	185.5060	0.0000000	1.11620

Table 4.--Deformed Cantilever Beam Results, F3 = 1,500.0 lb

Joint	X Force	Y Force	Z Moment	Displace-ment
1	2174.701	- 1587.651	-150121.75	0.0000
2	15718.458	- 855.997	- 0.0008115	0.1618
3	26531.250	- 2407.698	0.0006758	0.2651
4	32659.713	- 4194.857	0.0009868	1.3562
5	34744.824	- 5543.834	- 0.0000154	2.3218
6	33532.140	- 6303.272	- 0.0009786	3.4883
7	29710.881	- 6237.572	0.0017390	4.8222
8	23944.582	- 5499.351	- 0.0001169	6.2900
9	16673.930	- 4048.989	- 0.0015223	7.8584
10	8452.642	- 2046.108	- 0.0008894	9.4937
11	-224143.125	38725.512	0.0000000	11.1620

deformations are nonlinear.

3. APPLICATION OF THE QUASI-NEWTON METHOD

The Quasi-Newton procedure outlined in Chapter 1 and 2 are applied for the truss-and beam-problems specified in Chapter 2. After a few iterations convergence was achieved. The same problems were solved with the Quasi-Newton Method implemented is MSC/NASTRAN (Version 66,67).

It was impossible to prove that the proposed modification is much more effective and efficient. It is necessary to run a test series, solving large practical problems. The results will be presented at the Conference. To establish the environment at the University level, to test new numerical procedures based on existing software codes (MSC/NASTRAN), the "Mechanical Engineering Program-bank" was developed.

4. DISCUSSION

The Mechanical Engineering Program-Bank

Today, nearly all areas of expertise in engineering are covered via large application software packages. To integrate them into one black box makes no sense, due to modern parallel processing technology and the need of applied research, and maintenance. Therefore, we try to adapt all software systems via standards used for the pre-and post-processing of data. To run an automatic process chain, applying different software systems, can easily be achieved, programming in C-language.

MSC/NASTRAN, IDEAS (CAEDS), MSC/EMAS, MSC/DYTRAN, unconstrained and constrained optimization FORTRAN-codes can easily be executed via SYSTEM-function in C-language, and then applied within an iteration procedure. Via the NASTRAN-OUTPUT_2-file, IDEAS (CAEDS) and MSC/NASTRAN are linked together.

The Mechanical Engineering Program-Bank provides the environment for a large variety of complex engineering problems. The outlined adaptive numerical procedure for large deflection analysis of complex structures can be extended to nonlinear transient heat-transfer analysis. A tube consisting of different layers is heated via a wire of a special pattern (a spiral, et.). A sensor at a special location turns the heating on and off.

The numerical simulation can be established via the Program-Bank, using MSC/NASTRAN and a Quasi-Newton Code. The outlined procedure in Chapter 1 and 2 can easily be adapted. Relating the temperature distribution to the displacements, the heat sources to load and react, a nonlinear transient heat transfer problem can be solved cia MSC/NASTRAN using small time increments. If the temperature at the sensor location has reached a maximum, the loading cards are changed to simulate a cooling process (heat turned off). At the minimum level of the temperature the MSC/NASTRAN loading cards are again switched, to simulate heating.

Figure 5 shows the prompt of the command-and-menu driven Program-Bank after initiating the software via the command ME-BANK:

```
 ________________________________________________
|                                                |
|        MECHANICAL  ENGINEERING                 |
|              PROGRAM BANK                      |
|                                                |
|        Designed and maintained by:             |
|                                                |
|        Allen Teagle, Chris Fuld,               |
|        Karl Conroy,Ajay Hirve,                 |
|        Ortwin Ohtmer                           |
|                                                |
|        Version 1.0    02/24/1994               |
|                                                |
 ------------------------------------------------
```

```
Currently implemented:

NAStran
DYTran
EMAs
ANSys
GTStrudl
CHOlesky
GAUss
INVerse power
MODified splines
FOUr-bar-mechanism
NAA - Nonlinear Adaptive Analysis
FEA - Input Adaption

CAEds
FTT (connect to VAX (TIGER) via ftp)
207 (printer)

Quit
```

Figure 5

5. CONCLUSION

To demonstrate the flexibility of the Quasi-Newton Method, several application areas are summarized:

* Minimization of functions
* Finding Roots of Nonlinear Equations
* Solution of Non-linear Equations
* Minimization of Total Potential Energy
* Large Deflection Analysis of Elastic Structures
* Design Optimization

For Design optimization (constrained optimization) instead of minimizing the objective function (weight, potential energy, etc.), the Lagrange function is minimized.

Via a Programbank, universities and research institutions are able to use MSC/NASTRAN, MSC/DYTRAN, MSC/EMAS, etc. as a black box together with new C- or FORTRAN-source codes to perform extensive test series. Based on these results MSC should make a decision whether or not to implement new numerical procedures in general purpose software MSC/NASTRAN, MSC/DYTRAN, or MSC/EMAS.

REFERENCES

1. Hughes, T.J.R., The Finite Element Method, Prentice-Hall, Englewood Cliffs, N.J., 1987

2. "Host of New Features Available in Version 68 of MSC/NASTRAN," MSC/WORLD, Vol. 3, No. 3, march 1993, pp. 1,14.

3. Arora, J.S., Introduction to Optimum Design, McGraw-Hill, New York, 1989.

4. CQUEL.BYU, computer program handbook, Brigham Young University, Engineering Computer Graphics Laboratory, Provo, Utah, 1992.

5. Coburn, T.D., The Quasi Newton method in Optimization, unpublished M.S. Thesis, California State University, Long Beach, 1993.

6. Chang, J.C., Nonlinear Stress Analysis of solids with Nonlinear Elastic material, Large Deflections, unpublished M.S. Thesis, California State University, Long Beach, 1991.

7. Gill, P.E., Murray, W., and Wright, M.H., Practical Optimization, Academic Press, New York, 1987.

8. Lutter, G.L., and Ohtmer, O., Struceng & Femcad Structural Engineering and Optimization, 1st ed., (IITT) International, Paris, France, 1989.

9. Ohtmer, O., Boundary element Improved Integration Technique Using spline Functions, Computational Mechanics Publications, Boston, 1987.

10. Arora, J.S., and Tseng, C.H., User's Manual for IDESIGN: Version 3.5, College of Engineering, University of Iowa, Iowa City, 1987.

11. Morris, A.J., Foundations of Structural Optimization: A Unified Approach, Wiley-Interscience, New York, 1982.

12. Georgia Institute of Technology, GT STRUDL User's Manual, 4th ed., 6 vols., Georgia Institute of Technology, Atlanta, 1990.

13. MSC/NASTRAN Handbook for Nonlinear Analysis: Version 67, MacNeal-Schwendler Corporation, Los Angeles, March 31, 1992.

14. Ye Zlue, Nonlinear Adaptive Analysis and Animation via Quasi-Newton Approach, unpublished M.S. Thesis, California State University, Long Beach, 1994.1