A METHOD FOR ACCELERATING CONVERGENCE TO THE SOLUTION OF NONLINEAR STRUCTURAL EQUATIONS BY REDUCING RESIDUAL WORK

By

CLINE TURNER YOUNG II

Bachelor of Science
United States Merchant Marine Academy
Kings Point, New York
1973

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1976

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
DOCTOR OF PHILOSOPHY
May, 1981

A METHOD FOR ACCELERATING CONVERGENCE TO THE SOLUTION OF NONLINEAR STRUCTURAL EQUATIONS BY REDUCING RESIDUAL WORK

Thesis Approved:

Thomas E. Blejwas
Thesis Adviser

John E. Kelly

Richard J. Lowery

Lynn R. Eubanks

Norman N. Durham
Dean of the Graduate College

ACKNOWLEDGMENTS

The successful completion of this degree would not have been possible without the aid and support of many people. I would like to express my deep appreciation to Dr. Rembert F. Jones for his technical advice and friendly encouragement; to Dr. Lynn R. Ebbesen for all his time spent tutoring me in computer processes; to Dr. Richard L. Lowery for his patience and understanding; and to Dr. Thomas C. Blejwas, my thesis adviser, for his advice, instruction, and encouragement during my graduate study. I would once again like to thank Drs. L. R. Ebbesen, R. L. Lowery, and T. E. Blejwas, as well as Dr. A. E. Kelly for serving on my committee.

My appreciation is expressed to Charlene Fries and Eldon Hardy for their assistance in preparing the final manuscript. My appreciation is also expressed to Becky Farrell for her preparation of the rough draft.

I owe a very special word of gratitude to my wife, Libby. Her constant support, patience, understanding, encouragement, and sacrifice were the main ingredients of my endurance and success in this endeavor. I also wish to thank my mother for her continuous support of my education throughout the years.

The School of Mechanical and Aerospace Engineering provided a position as Instructor which helped make my graduate study possible. The Submarine Structures Division of the David Taylor Naval Ship Research and Development Center provided a contract which also helped make my graduate study possible.

TABLE OF CONTENTS

Chapter		Page
I.	INTRODUCTION	1
	Overview	1
	Approach to the Problem	2
	Organization	3
II.	BACKGROUND THEORY	5
	Solution Techniques	5
	Development of Structural Equations	14
	Literature Survey	18
III.	DEVELOPMENT OF EXTRAPOLATION TECHNIQUES	23
	Linear Scalar Extrapolation	23
	Multiple Search Extrapolation	32
IV.	MODIFICATION OF COMPUTER PROGRAM	38
V.	APPLICATION EXAMPLES	45
	Linear Scalar Extrapolation	45
	Multiple Search Extrapolation	53
	Plasticity	58
VI.	CONCLUSIONS AND RECOMMENDATIONS	60
BIBLIOGRAPHY		62
APPENDIX A - PRINCIPLE OF VIRTUAL WORK		66
APPENDIX B - DEVELOPMENT OF THE STRUCTURAL EQUATIONS		69
APPENDIX C - NONLINEAR ANALYSIS		74
APPENDIX D - PARABOLIC EXTRAPOLATION		76

LIST OF FIGURES

Figure		Page
1.	Explicit Solution to a Static Nonlinear Problem	6
2.	Newton-Raphson Technique (Normal Iterative Solution)	8
3.	Newton-Raphson Technique (Failure Mode of Solution)	10
4.	Direct Iteration (Normal Iterative Solution)	12
5.	Direct Iteration (Stability Example)	13
6.	Configuration History	16
7.	Modified Newton-Raphson	19
8.	Direct Iteration	24
9.	Direct Iteration With Linear Scalar Extrapolation	26
10.	Direct Iteration	27
11.	Direct Iteration With Linear Scalar Extrapolation Extended	28
12.	Effect of Convergence Criteria	31
13.	Direct Iteration With Multiple Search Extrapolation	35
14.	Direct Iteration With Multiple Search Extrapolation	36
15.	Bar-Spring Problem	46
16.	Load Versus Displacement for Bar-Spring Problem	48
17.	Finite Element Model of a Ring	51

LIST OF TABLES

Table		Page
I.	Subroutine ITERCK	39
II.	Bar-Spring Problem	49
III.	Ring Problem (1.001)	52
IV.	Pressure Loaded Ring (59.5 psi)	55
V.	Pressure Loaded Ring (59.5 psi)	57
VI.	Radially Loaded Ring (73.5 psi)	59

CHAPTER I

INTRODUCTION

Overview

At present, certain finite element codes can be used in conjunction with large computers to produce solutions to virtually all static problems conceivable in structural analysis. In addition, well-behaved problems such as those involving small elastic deformations can be solved relatively inexpensively and accurately. Computational difficulties do not arise until the stiffness of the structure has become a function of displacement or of displacement history. The former, known as geometric nonlinearity, occurs under large deflections, and the latter, known as material nonlinearity, is inherent to some greater or lesser degree in all deflections.

The problem is termed nonlinear once the stiffness of the structure has become a function of displacement because there is no longer a linear relationship between the applied forces and the resulting structural deformation. This additional complexity may require an iterative solution procedure where an attempt to reduce some error term to zero at each iteration takes place. Therefore, a nonlinear problem is more expensive to solve and can be much more so depending upon the degree of nonlinearity, the solution method employed, and the convergence criteria used. Furthermore, there is no way to avoid nonlinear problems where ultimate or failure analysis is needed because of the host of accompanying

nonlinear effects such as buckling, plasticity, creep, fatigue, fracture, etc. The reduction in cost of solution thus becomes a prime consideration.

There are at least three major areas in which the cost of solution of the nonlinear structural equations may be reduced. They are:

1. The development of more sophisticated hardware.

2. The reduction of the cost per iteration by linearizing some terms and neglecting others depending on the problem.

3. Reduction of the number of iterations.

This study in no way attempts the first course of action. The second two were investigated in detail and the findings constitute the major portion of this dissertation.

Approach to the Problem

The nonlinear structural equations solved in this study were developed in incremental form by Jones [1] using the principle of virtual work. Large strain expressions were used and allowance was made for the inclusion of plastic deformation effects. The iterative procedure used to solve the structural equations is of the secant stiffness type, sometimes referred to as direct iteration. The convergence criteria used to determine when iteration should stop was in all cases based on displacements because only buckling problems were considered in this study. Only static problems were investigated, but as implied above, both geometric and material nonlinearities were allowed.

The finite element codes and computer hardware necessary for solution of the structural equations were used in three different phases.

Phase I: Code--MARCSTRUC

Computers--CDC 6600/6700 and 6400

Purpose--To determine validity of linear extrapolation.

Phase II: Code--AGGIE 1 (see Haisler [2])

Computer--IBM 370/168

Purpose--To determine validity of individual degree of freedom (dof) linear extrapolation.

Phase III: Code--TRAINS (see Roderick et al. [3, 4])

Computers--CDC 6400, CYBER 74

Purpose--To determine validity of multiple search extrapolation.

Computer codes and computer hardware were changed due to a change of location in which that phase of the research was performed, which in turn affected the availability of both hardware and software.

Organization

Chapter II consists of a literature review of the major procedural techniques available for the solution of nonlinear structural equations. Also, it contains the development of the structural equations with reference to appendices where the background theory is presented.

Chapter III presents the theory development in this study. It begins with the motivation for the linear extrapolation development and continues through discussion of failure conditions for the multiple search extrapolation. Reference is made to appropriate appendices for a more detailed explanation of theory.

Chapter IV briefly treats the modification of the existing computer code and is followed by Chapter V where the results of application to

specific problems are presented. The problems considered were:

1. Bar-Spring (1 dof)
2. Ring with Radial Loading (28 dof)
3. Ring with Pressure Loading (28 dof)
4. Ring-Stiffened Cylinder (94 dof).

Chapter VI completes the report of the research performed with a statement of pertinent findings and conclusions. A section on recommendations for future research is also presented.

CHAPTER II

BACKGROUND THEORY

Solution Techniques

Nonlinear structural equation solution techniques belong to one of two broad categories, either explicit or implicit. An explicit method receives its name from the fact that all displacements calculated are done so in terms of known values only, with some error term being monitored at every operational step. The consequence of solving highly nonlinear static structural equations explicitly may well be an increasingly larger error as shown in Figure 1 since iteration is not performed. The method of accuracy control is to monitor some error function, stop the solution procedure when the error term becomes unacceptably large, correct for the error, then restart the procedure. This method generally results in less accurate answers than does an implicit scheme and is unsuitable for the extrapolation techniques presented in this study. Therefore, an implicit type of solution was selected.

The implicit solution category may be subdivided once more. The resulting divisions are according to stiffness matrix type, either tangent or secant, see Cook [5] or Zienkiewicz [6].

The most obvious advantage of the tangent stiffness approach over the secant stiffness method is that the former offers quadratic convergence while the latter offers at best super linear convergence. However,

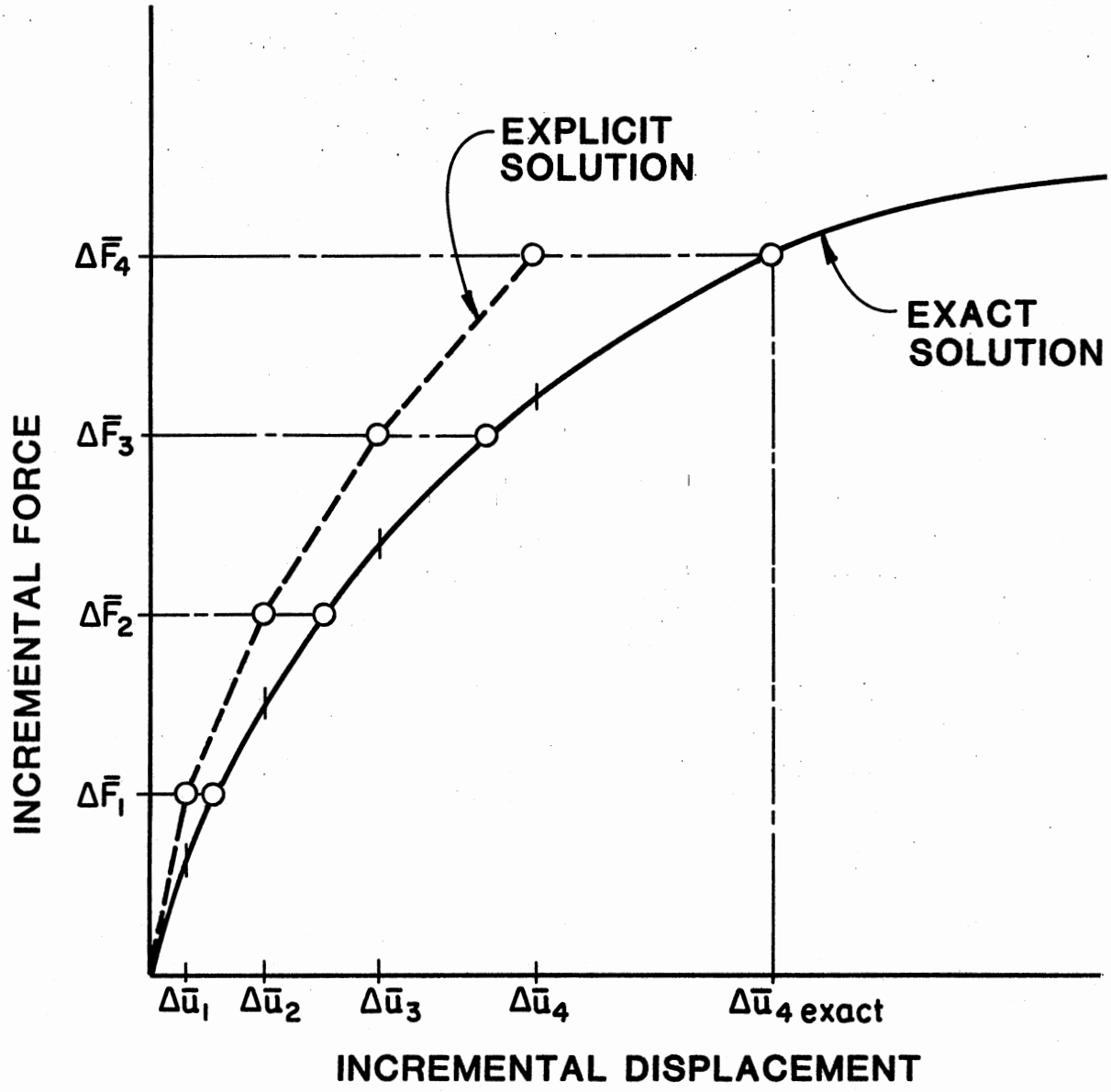

Figure 1. Explicit Solution to a Static Nonlinear Problem

in spite of this a secant stiffness approach was used. A justification for this selection through comparison of the methods follows.

Tangent Stiffness (Newton-Raphson)

There are many references for the treatment of the Newton-Raphson development such as Gerald [2] or Beveridge et al. [7]. Graphically, the method iterates to solution as shown in Figure 2.

Utilizing the notation of Figure 2, the general Newton-Raphson algorithm can be written as

$$\{\Delta u_{i+1}\} = \{\Delta u_i\} + [K_{i+1}]^{-1} \{\Delta F_i - R_i\} \qquad i = 0, 1, 2, \ldots \quad (2.1)$$

where

i = the i^{th} iteration;

$\{\Delta u\}$ = incremental displacement;

$\{\Delta F\}$ = externally applied incremental force;

$\{R\}$ = residual force (out of balance loads); and

$[K]$ = tangent stiffness matrix.

In the most general case, $[K_{i+1}]$, $\{\Delta F_i\}$, and $\{R_i\}$ are all functions of $\{\Delta u_i\}$. However, upon assuming $\{\Delta F\}$ to be a constant and biasing it to zero, Equation (2.1) becomes

$$\{\Delta u_{i+1}\} = \{\Delta u_i\} - [K_{i+1}]^{-1} \{R_i\} \quad (2.2)$$

which is a common form for expressing the Newton-Raphson method. This clearly indicates that the incremental difference between two successively calculated displacements is due to the stiffness matrix and the residual force.

As shown by Tauchert [8], according to Castigliano's first theorem written in index notation

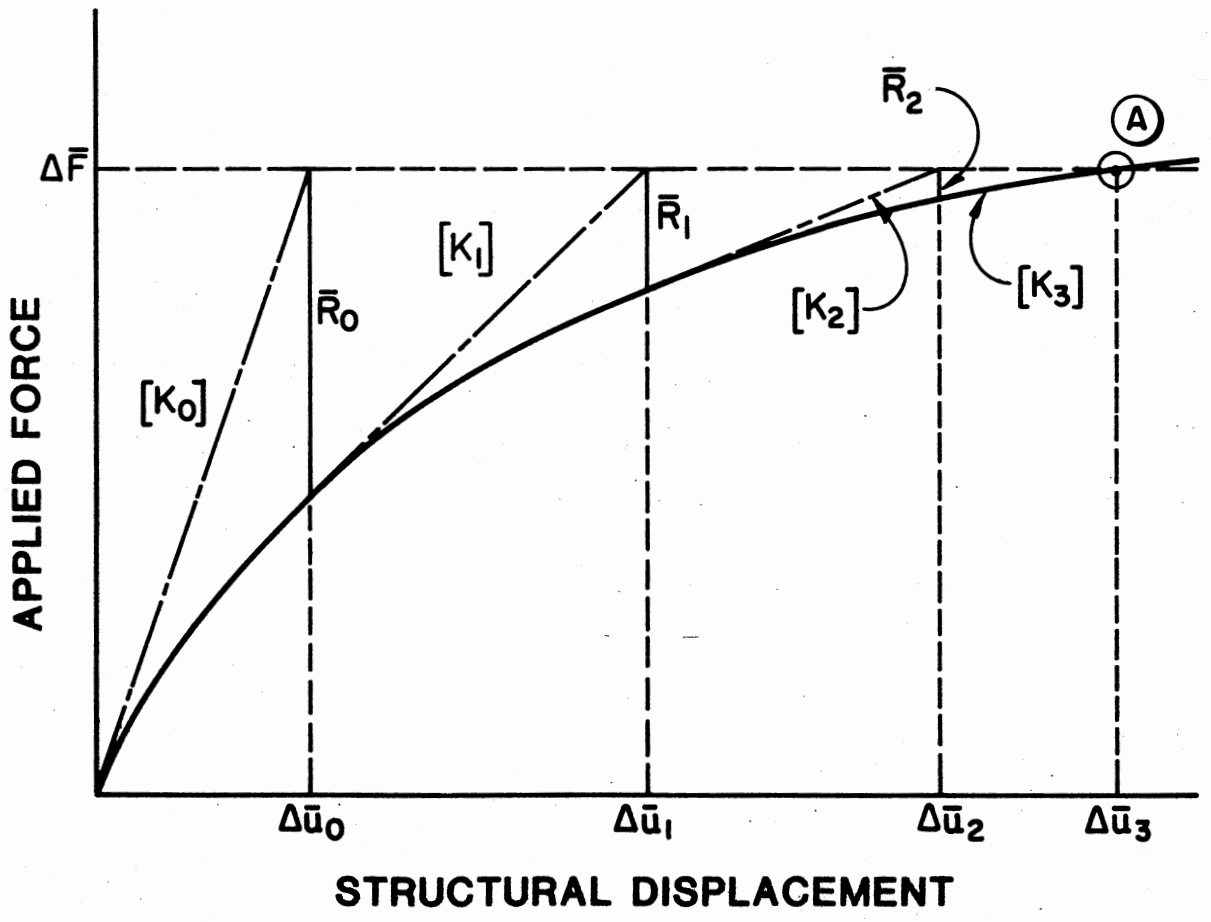

Figure 2. Newton-Raphson Technique (Normal Iterative Solution)

$$R_q = \frac{\partial u}{\partial x_q} \tag{2.3}$$

where

q = a particular degree of freedom;

x = generalized displacement;

u = internal strain energy; and

R = residual force after above mentioned biasing.

Inspection of the above development leads to the observation that there is an infinitesimal relationship between the generalized residual and the generalized displacement which can be expressed as

$$K_{qp} = \frac{\partial R_q}{\partial x_p} = \frac{\partial^2 u}{\partial x_q \partial x_p}$$

where

p = an index independent from q; and

K_{qp} = the qp component of the tangent stiffness matrix $[K_T]$.

It can be seen from Figure 2 that the stiffness matrices represented are tangent to the equilibrium curve.

A possible difficulty with buckling problems is illustrated in Figure 3. If the tangent to the force-displacement curve becomes nonpositive, the solution procedure fails. This difficulty and similar ones are likely for the problems of concern in this study.

The quadratic convergence offered by the Newton-Raphson method is the most common single reason for its use. However, the stiffness matrix may become ill-conditioned at near-buckling loads or nonpositive in snap-through problems. Since these are the type problems considered in this study, effective use of the Newton-Raphson method depends upon overcoming difficulties due to the stiffness matrix.

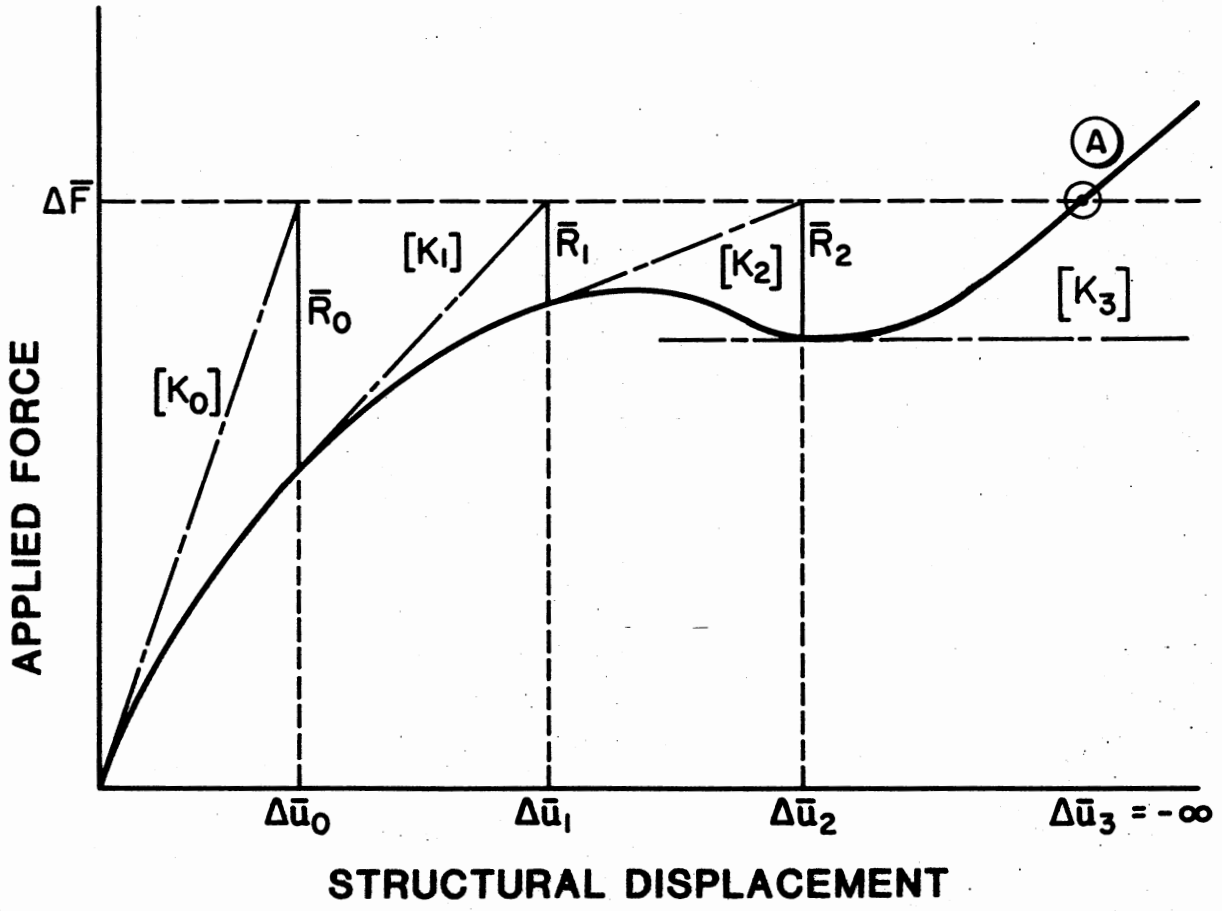

Figure 3. Newton-Raphson Technique (Failure Mode of Solution)

Alteration of step size or the introduction of a modified Newton-Raphson technique (see Reference [5]) cannot alleviate the difficulty. Some accepted methods of maneuvering past the difficult region amounted to stiffness matrix averaging from two different points in the solution history. This is similar to a secant stiffness approach. Therefore, the major theme of this study was to find a way to accelerate the convergence rate of a suitably stable secant stiffness solution procedure for the class of problems that contain geometric and material nonlinearities in a pre-buckling region.

Secant Stiffness (Direct Iteration)

As mentioned before, the secant stiffness matrix is an average stiffness matrix between two points in the solution history. A typical solution history is shown graphically in Figure 4. An average relationship between the residual force and the generalized displacement must exist as shown in Equation (2.3). It can be shown that a secant stiffness approach yields only linear convergence.

The major reason for the solution of the direct iteration procedure is its stability in regions where the Newton-Raphson method fails, as shown in Figure 5. This stability is apparently due to the recalculation of the whole incremental displacement at each iteration.

There are conditions under which the direct iteration method also will fail. In general, these can be overcome by an appropriate selection of incremental step size for the problem at hand. The step size selection should be such as to render a nonpositive definite stiffness matrix highly unlikely.

The penalty inherent in using a secant stiffness method with its

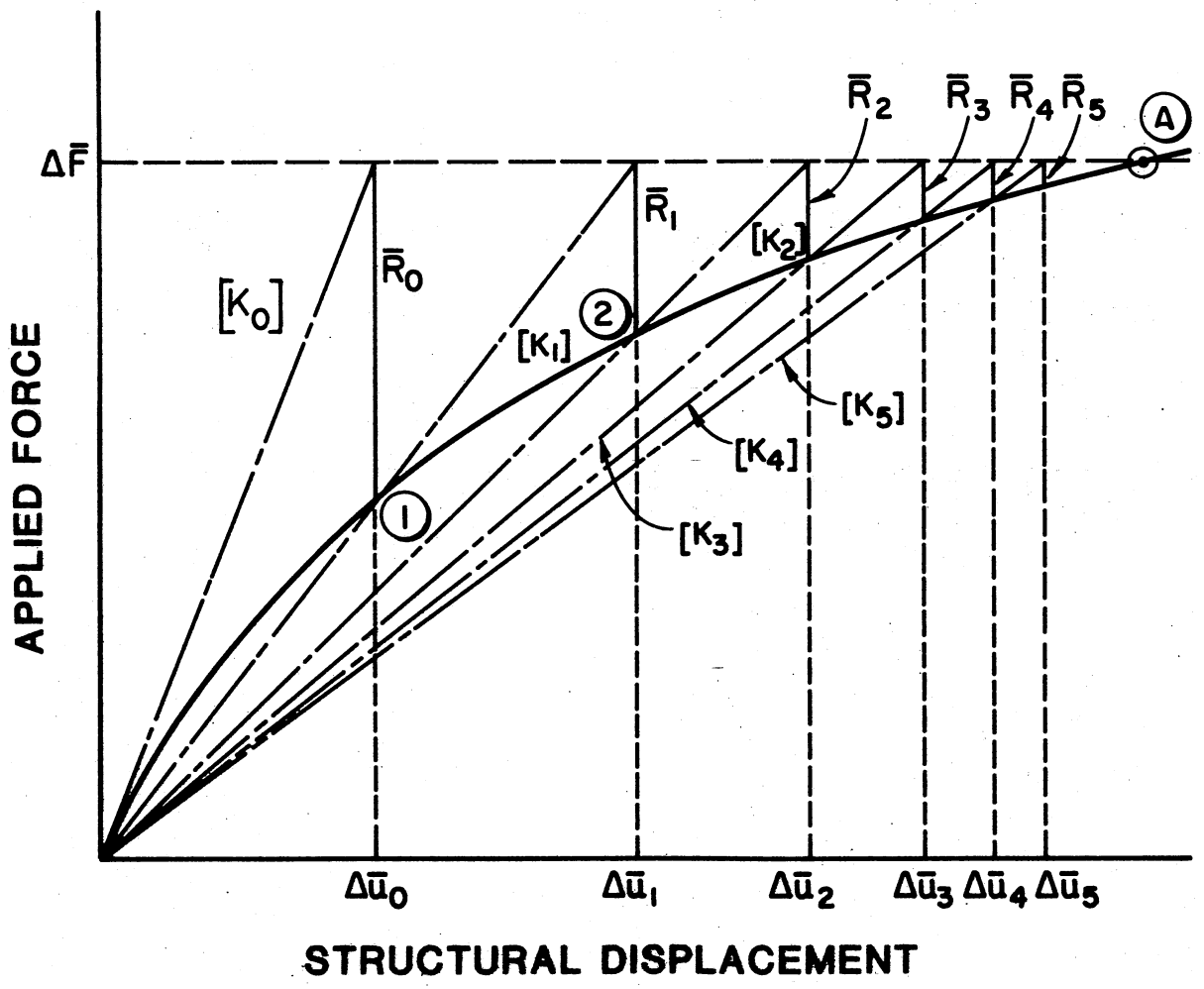

Figure 4. Direct Iteration (Normal Iterative Solution)

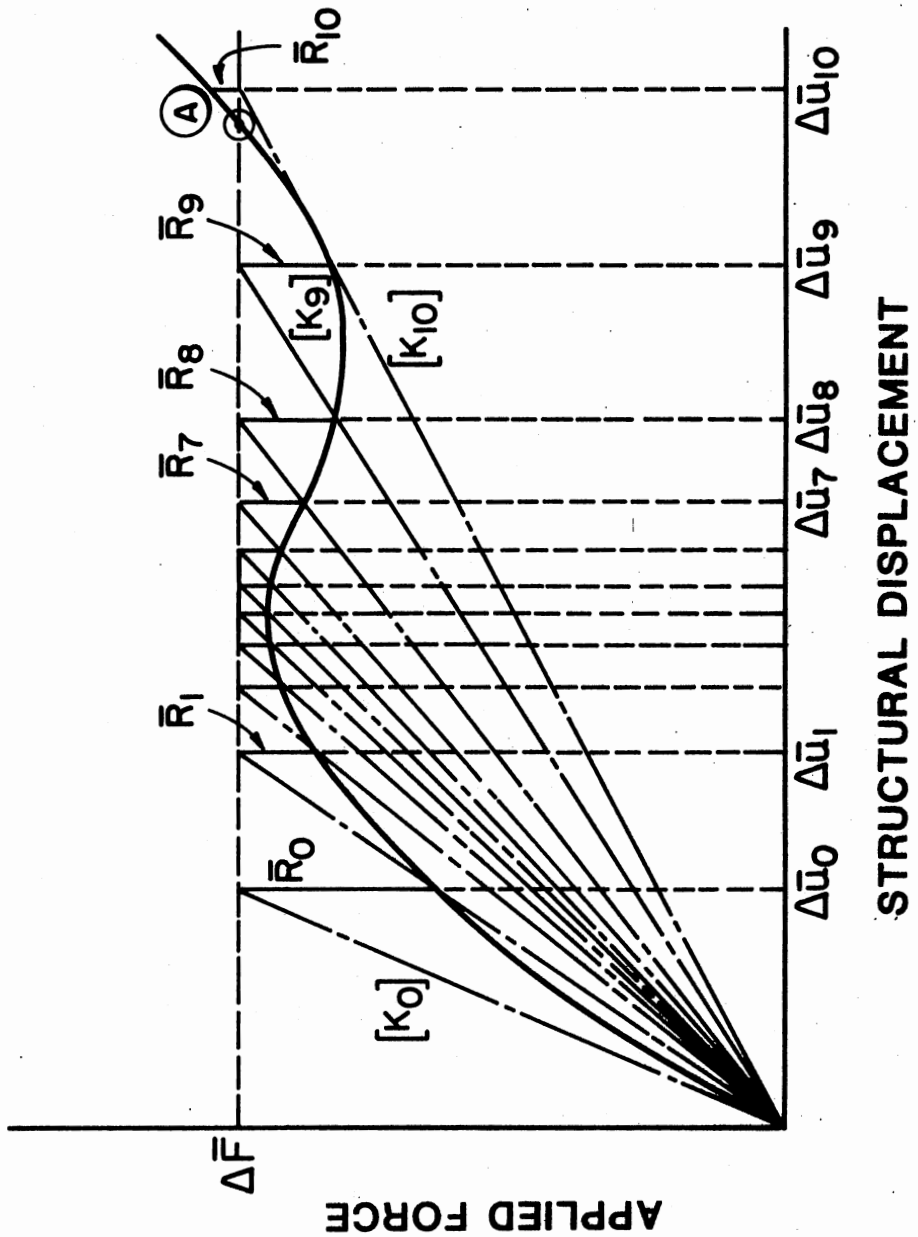

Figure 5. Direct Iteration (Stability Example)

step size restriction is a large number of iterations for a stiffness relaxation problem. Therefore, the development of an algorithm which will reduce the number of iterations per increment or reduce the cost of the iterations is desirable if a secant stiffness solution procedure is to be used.

Objectives

The secant stiffness solution procedure was selected in lieu of the tangent stiffness procedure because of the ability of the former to solve buckling problems at and beyond the buckling load. However, modification of the secant stiffness approach was necessary under the following guidelines to make it cost effective:

1. Reduce the number of iterations by using the residuals and accompanying displacements to extrapolate to a zero residual position.

2. Reduce the cost per iteration by reducing the complexity of the calculation of the next incremental displacement.

3. Preserve the accuracy inherent in the user's choice of finite element and finite element program.

4. Preserve the convergence characteristics inherent in the choice of finite element.

Development of Structural Equations

The structural equations were developed in incremental form using the principle of virtual work with the Kirchhoff stress tensor and the Lagrangian strain tensor [1, 9]. An abbreviated development of the principle of virtual work may be found in Appendix A.

Due to the importance of the structural equations to the application

in Chapter V of the theory developed in Chapter III, a detailed reproduction of the development is shown in Appendix B. Briefly, utilizing the principle of virtual work, Equation (A.7), and the various terms indicated in Figure 6, Equation (A.7) for the current configuration may be rewritten as

$$\int_s (T_q + \Delta_G T_q + \Delta T_q)\ \delta(\Delta u_q)dA + \int_v (\rho_o P_q + \rho_o \Delta P_q)\ \delta(\Delta u_q)dV$$
$$= \int_v (S_{qp} + \Delta S_{qp})\ \delta(\Delta E_{qp})dV \qquad (2.5)$$

where Δ is increment of change in magnitude, and Δ_G is increment of change in direction. Substituting for the strain as shown in Appendix B, Equation (2.5) can be written as

$$\int_v \Delta S_{qp}[\tfrac{1}{2}\ (\delta(\Delta u_q)_{,p} + \delta(\Delta u_p)^{(1)}_{,q}) + u_{m,q}\ \delta(\Delta u_m)_{,p}]dV$$
$$+ \int_v \Delta S_{qp}(\Delta u_m)^{(2)}_{,q}\ \delta(\Delta u_m)_{,p}dV$$
$$+ \int_v\ S_{qp}(\Delta u_m)^{(3)}_{,q}\ \delta(\Delta u_m)_{,p}dV$$
$$- \int_s \Delta_G F_q\ \delta^{(4)}(\Delta u_q)dS$$
$$= \int_v (\rho_o P_q + \rho_o\ \Delta P^{(5)}_q)\ \delta(\Delta u_q)dV$$
$$+ \int_s (F_q + \Delta F_q)^{(6)}\ \delta(\Delta u_q)dS$$
$$- \int_v S_{qp}[\tfrac{1}{2}\ (\delta(\Delta u_q)_{,p} + \delta^{(7)}(\Delta u_p)_{,q})$$
$$+ u_{m,q}\ \delta(\Delta u_m)_{,p}]dV \qquad (2.6)$$

Equation (2.6) is the result of the direct application of the principle of virtual work by Jones [1], although many of the terms were previously identified by others as mentioned by Jones and as shown below:

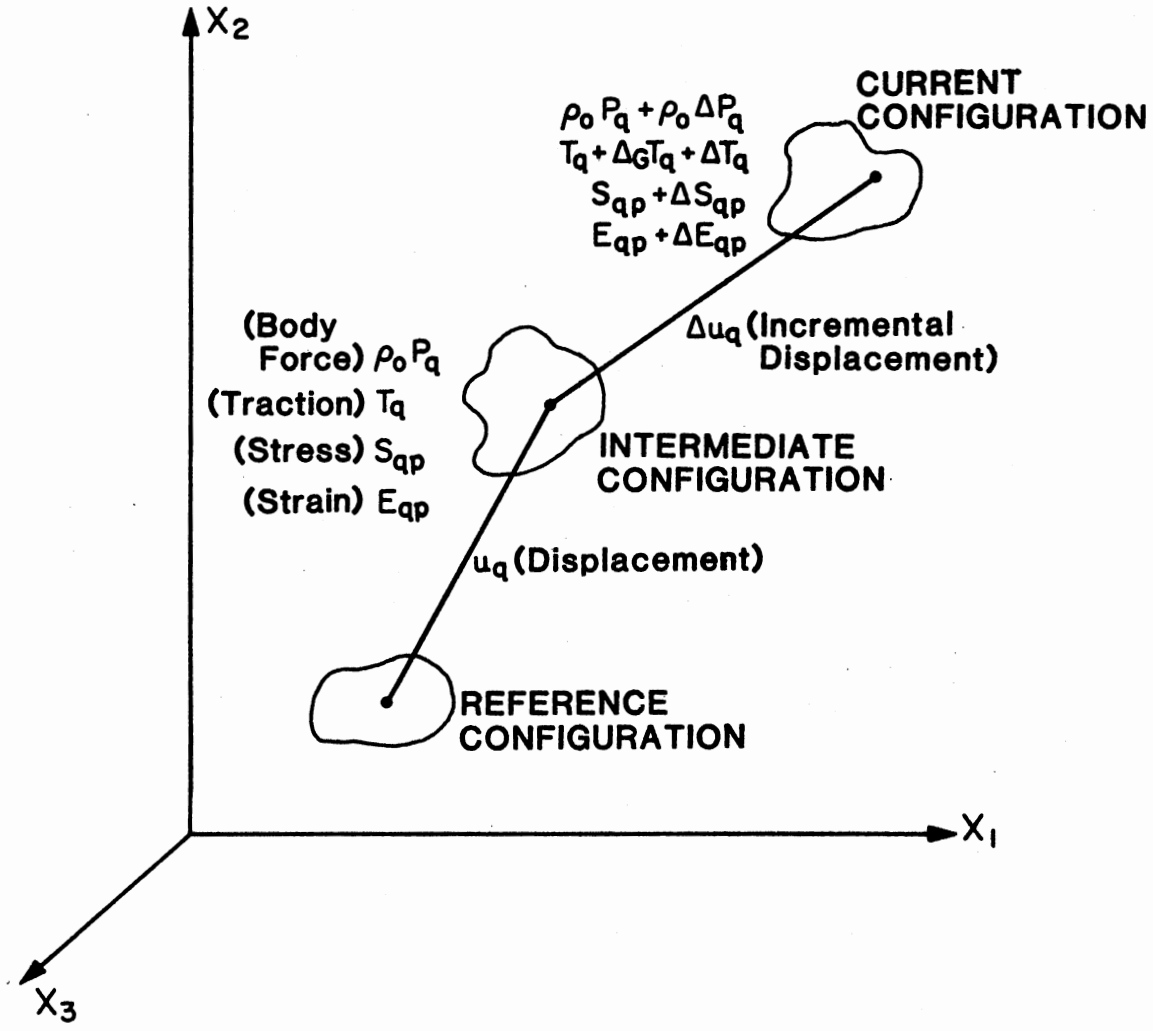

Figure 6. Configuration History

Term (1)--Small displacement plus initial displacement matrix by Marcal [10].

Term (2)--Initial stress or geometric stiffness matrix by Martin [11].

Term (3)--Apparently unidentified previously.

Term (4)--Initial load matrix by Hibbit et al. [12] and Oden and Keys [13].

Term (5)--Body load plus increment of body load.

Term (6)--Surface traction plus increment of surface traction.

Term (7)--Equilibrium load correction of Hofmeister et al. [14] and Stricklin et al. [15].

The point of expressing the principle of virtual work in the form of Equation (2.6) was to show that before the complete variational formulation various terms were overlooked, ignored, or unnecessary, depending upon the problem being solved. However, it has been shown that without using all terms, problems involving buckling or high nonlinearity of material characteristics may be solved incorrectly or with gross misrepresentations due to the ill-conditioning of the stiffness matrix. Terms that are normally insignificant may have very significant effects in nonlinear regions. The possibility of reducing cost by removing the "insignificant" terms must be viewed with great care for nonlinear problems.

As shown in Appendix B, the final form of Equation (2.6) is

$$[K]_s\{\Delta\bar{V}\} = \{P\} + \{\Delta P\} + \{T\} + \{\Delta T\} + \{\Delta_G T\} - \{E\} - [K]_A\{\Delta\bar{V}_{est}\} \tag{2.7}$$

where

$[K]_s$ = symmetric part of stiffness matrix and a function of the intermediate configuration;

$\{\Delta\bar{V}\}$ = unknown incremental nodal displacements;

$\{P\}$ = body loads;

$\{\Delta P\}$ = increment in body loads;

$\{T\}$ = surface loads;

$\{\Delta T\}$ = increment in magnitude of surface forces;

$\{\Delta_G T\}$ = increment in direction of surface forces and a function of the intermediate configuration;

$\{E\}$ = equilibrium correction vector and a function of the intermediate configuration and the assumed incremental displacements of the current configuration;

$[K]_A$ = skew symmetric part of the stiffness matrix and a function of the intermediate configuration; and

$\{\Delta\bar{V}_{est}\}$ = assumed incremental nodal displacement.

The major difficulty involved in the solution of nonlinear equations is that the stiffness matrix is not a constant and may be a function of displacement or displacement history, or a number of other items. See Appendix C.

Literature Survey

Background History

The reformulation and the factorization of the stiffness matrix is a first order contribution to the cost of solution of nonlinear structural equations. Consequently, the quadratic convergence of the Newton-Raphson method in its various forms is very popular.

The most frequently used Newton-Raphson method is the modified Newton-Raphson approach depicted in Figure 7. In order to reduce the number of reformulations of the stiffness matrix, the modified Newton-Raphson

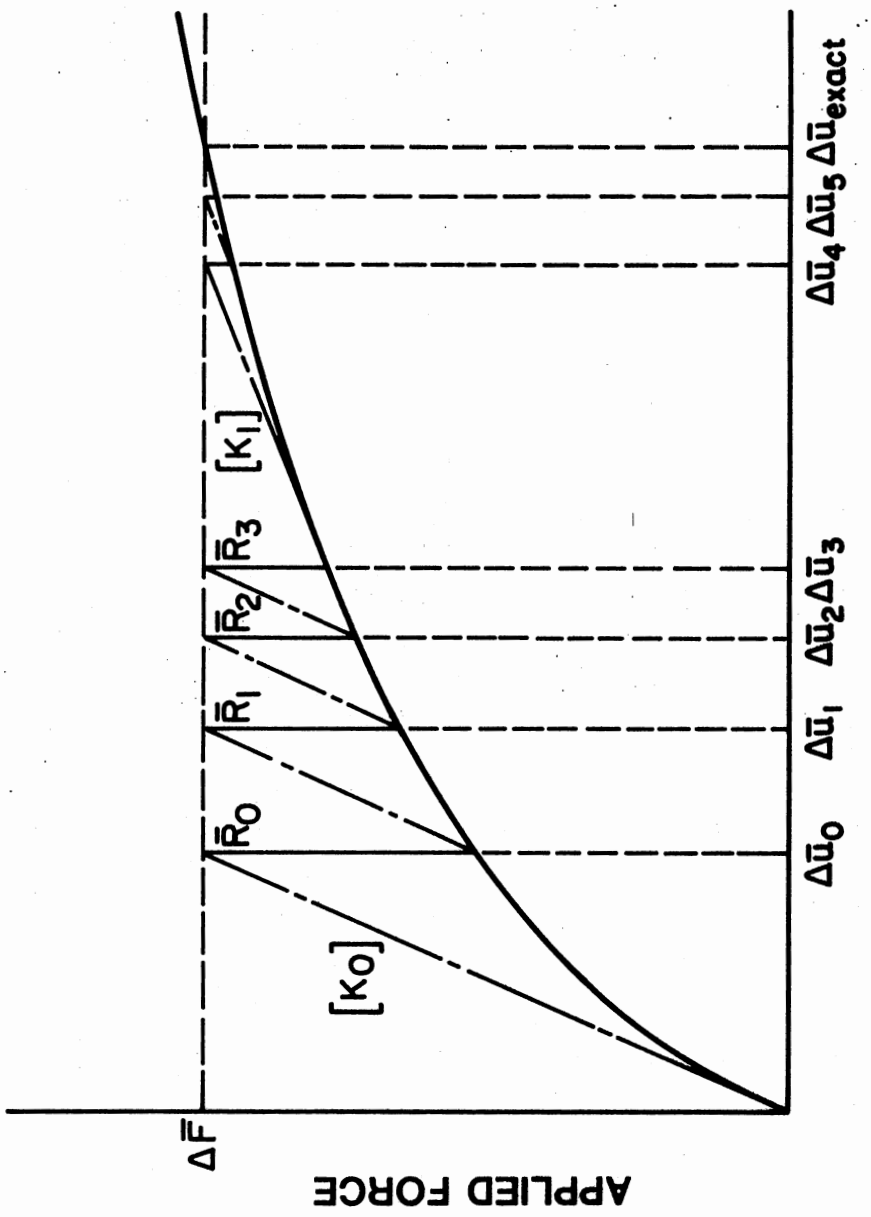

Figure 7. Modified Newton-Raphson

uses one stiffness matrix until convergence becomes too slow, as shown at position $\Delta\bar{u}_3$.

An early paper by Thurston [22] showed that the Newton-Raphson method could be used to find post-buckled solutions along different branches and pre-buckled solutions away from the instability region. Recently the cost of solution has been the subject in a number of papers. Mondkar and Powell [23] have used the constant alpha technique to update the stiffness matrix inexpensively for the modified Newton-Raphson problem. Almroth et al. [24] surveyed the possibilities of reducing the cost of solution and state that an automatic choice of solution procedures holds promise, see Reference [25]. Schmidt [26] stated that an automatic choice of step size is most useful while Almroth et al. [27] included an automatic selection of global shape functions. Schmidt [28] also proposed a method of extending the convergence domain of the Newton-Raphson method so that a larger step size could be used. Many others sought a less expensive solution for particular problems [29, 30, 31, 32, 33].

Current State-of-the-Art

The purpose of this study was, in part, to reduce the number of iterations required for solution and not to develop a finite element expression. The literature search was for methods concerning iteration reduction. When this research began in May of 1979, there were no published articles, books or reports devoted to solution cost reduction cross-referenced in the major periodical indices with the single exception of a residual (out-of-balance-load) reduction technique proposed by Felippa [34]. Felippa's method consists of setting up a functional whose magnitude is determined by a pseudo-Euclidean norm of the residual, R, and

minimizing it. Unfortunately, the norm, defined as

$$= \frac{1}{2} \{R\}^T [W]\{R\} \tag{2.8}$$

requires the definition of a weighting matrix, [W], such that nonhomogeneous terms in {R} can be converted to some common dimension or to a dimensionless value. Thus the user must interact with the program to an undesirable extent. However, Felippa's work is the starting point for this investigation.

Recently, two papers concerning the reduction in cost of the solution procedure have appeared and two more have been submitted for publication. Three of these papers, Geradin et al. [35], Matthies et al. [36], and Strang [37] have taken similar approaches to that introduced by Dennis, Jorge, and More [38] on Quasi-Newton methods. All of these recent works attempt to update the stiffness matrix without going through the full process of reformulation and factorization. The most popular approach is to update the stiffness matrix by a matrix of rank two,

$$[K_1] = ([I] + \{w_1\}\{v_1\}^T)[K_o]^{-1}([I] + \{v_1\}\{w_1\}^T) \tag{2.9}$$

where [I] is the identity matrix, and $\{v_1\}$ and $\{w_1\}$ are vectors dependent on displacement and residual values. This calculation requires m by n operations, where m is the bandwidth and n is the number of degrees of freedom. A normal stiffness update requires m by m by n operations. This method involves changing the stiffness matrix slightly and then reevaluating the accompanying error. A large number of iterations may be required to satisfy convergence within a given load step for highly nonlinear problems.

Crisfeld [39] combined the above approach and an elementary line search, except the stiffness matrix is altered by a matrix of rank one.

He restricted his work to accelerating the iterations in a solution history obtained from a modified Newton-Raphson approach. Young [40] introduced a more sophisticated line search based on minimizing the contribution of the residual loads to the total energy of the system. This method, which is explained in detail in Chapter III, requires only n operations for a scalar extrapolation term and somewhat more for the multiple search extrapolation technique.

The two newest papers in the area are by Pappas [41] and Kamat et al. [42]. The procedure of the former searches in different directions for a correct solution and the latter attempts to minimize the energy of the structure. One can see from the above that the methods surfacing for reducing cost are beginning to embody common concepts if not similar theory. Clearly, there is a growing recognition by the technical community for the need to modify the solution processes for nonlinear structural equations so as to reduce cost.

CHAPTER III

DEVELOPMENT OF EXTRAPOLATION TECHNIQUES

The modifications for this study to the direct interation method are based upon minimizing the work done by the residuals (out-of-balance loads). The residual work term is an error in the strain energy expression. Extrapolation methods are used to minimize this source of error more rapidly than is possible with a direct iteration solution.

Linear Scalar Extrapolation

Development

A graphical representation of the direct iteration method is presented in Figure 4. The curve represents the calculated resistance of the structure. The original stiffness matrix, $[K_o]$, is developed using material properties and assuming linearly elastic deformation; consequently, as shown in Figure 4, this stiffness matrix is tangent to the curve at the origin. Upon application of the incremental load, $\Delta\bar{F}$, the displacement $\Delta\bar{u}_o$ with associated residual, $\bar{R}_o$, is calculated. The displacement $\Delta\bar{u}_o$ is used to update the stiffness matrix to yield $[K_1]$. This stiffness matrix, when factored and subjected to the incremental load, produces the new incremental displacement, $\Delta\bar{u}_1$, with its associated residual, $\bar{R}_1$. This process continues until the convergence criteria are satisfied. (See Figure 8 for this method in equation form).

$$\left.\begin{aligned} \{\Delta u_o\} &= [K_o]^{-1}\ \{\Delta F\} \\ [K_1] &= K(\Delta \bar{u}_o) \\ \{R_o\} &= \{\Delta F\} - \Sigma[k_1]\{\delta_o\} \end{aligned}\right\} \text{Iteration 1}$$

$$\left.\begin{aligned} \{\Delta u_1\} &= [K_1]^{-1}\ \{\Delta F\} \\ [K_2] &= K(\Delta \bar{u}_1) \\ \{R_1\} &= \{\Delta F\} - \Sigma[k_2]\{\delta_1\} \end{aligned}\right\} \text{Iteration 2}$$

$$\left.\begin{aligned} \{\Delta u_2\} &= [K_2]^{-1}\ \{\Delta F\} \\ [K_3] &= K(\Delta \bar{u}_2) \\ \{R_2\} &= \{\Delta F\} - \Sigma[k_3]\{\delta_2\} \end{aligned}\right\} \text{Iteration 3}$$

$$\{\Delta u_3\} = [K_3]^{-1}\ \{\Delta F\}$$

$$\vdots$$

To Convergence

where

$[K]$ = global stiffness matrix;

$[k]$ = elemental stiffness matrix;

$\{\Delta u\}$ = global displacements; and

$\{\delta\}$ = elemental displacements.

Figure 8. Direct Iteration

The manner of modification of the direct iteration procedure may be seen by comparing Figures 4 and 9 and by comparing the equation forms in Figures 8 and 10. The method developed to attempt minimization of the work done by the residuals is identical to the direct iteration method of Figures 4 and 9 up to and including the calculation of $\Delta\bar{u}_1$ and $\bar{R}_1$. It is the manner of determination of the new estimated displacement, $\Delta\bar{u}_2$, that is altered.

The linear extrapolation method is illustrated for a single-degree-of-freedom in Figure 9. Linear extrapolation using the residuals $\bar{R}_o$, and $\bar{R}_1$ and the distance between them, $\bar{d}$, results in an estimated position, $\Delta\bar{u}_2$. However, $\Delta\bar{u}_2$ is not the exact solution and a residual, $\bar{R}_3$, can be calculated. The residuals $\bar{R}_1$ and $\bar{R}_2$ and the distance between them, $\bar{d}_2$, supply the necessary information for a second extrapolation when it is needed because of unsatisfied convergence criteria. This procedure is continued until some stopping criteria have been met as shown in Figure 11.

The development of the procedure for calculating the scalar w is not as simple as Figure 9 may suggest. In a one-dimensional case it is a simple matter to determine that:

$$w = \frac{\bar{R}_o}{\bar{R}_o - \bar{R}_1} \tag{3.1}$$

However, since in general, the vectors $\bar{R}_o$ and $\bar{R}_1$ are heterogeneous in dimensional form, a division as shown above is impossible because this vector space is not Euclidean. Thus vector lengths have no meaning even though the inner products do exist. The solution to this difficutly was obtained by considering the commonality of work done by generalized forces moving through generalized displacements.

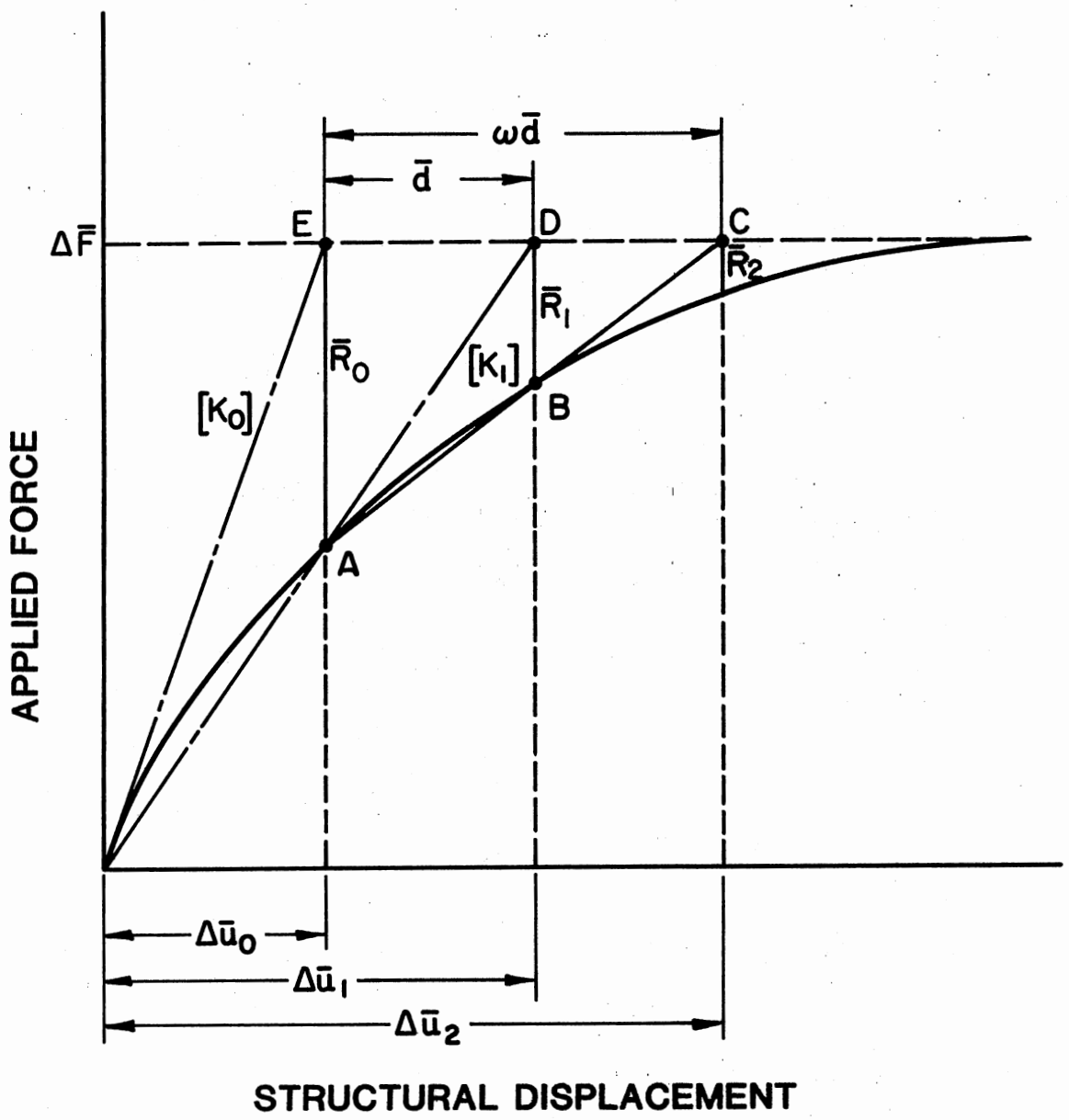

Figure 9. Direct Iteration With Linear Scalar Extrapolation

$$\{\Delta u_o\} = [K_o]^{-1}\ \{\Delta F\}$$
$$[K_1] = K(\Delta \bar{u}_o)$$
$$\{R_o\} = \{\Delta F\} - \Sigma[k_1]\{\delta_o\}$$

Iteration 1

$$\{\Delta u_1\} = [K_1]^{-1}\ \{\Delta F\}$$
$$[K_2] = K(\Delta \bar{u}_1)$$
$$\{R_1\} = \{\Delta F\} - \Sigma[k_2]\{\delta_1\}$$

Iteration 2

$$\{d_1\} = \{\Delta u_1\} - \{\Delta u_o\}$$
$$\omega_1 = \frac{\{R_o\}\cdot\{d_1\}}{[\{R_o\} - \{R_1\}]\cdot\{d_1\}}$$
$$\{\Delta u_2\} = \{\Delta u_o\} + \omega\{d_1\}$$
$$[K_3] = K(\Delta \bar{u}_2)$$
$$\{R_2\} = \{\Delta F\} - \Sigma[k_3]\{\delta_2\}$$

Iteration 3 (Extrapolated)

$$\{d_2\} = \{\Delta u_2\} - \{\Delta u_1\}$$
$$\omega_2 = \frac{\{R_1\}\cdot\{d_2\}}{[\{R_1\} - \{R_2\}]\cdot\{d_2\}}$$
$$\{\Delta u_3\} = \{\Delta u_1\} + \omega\{d_2\}$$
$$[K_4] = K(\Delta \bar{u}_3)$$
$$\{R_3\} = \{\Delta F\} - \Sigma[k_4]\{\delta_3\}$$

Iteration 4 (Extrapolated)

$$\{d_3\} = \{\Delta u_3\} - \{\Delta u_2\}$$
$$\vdots$$

To Convergence

Figure 10. Direct Iteration

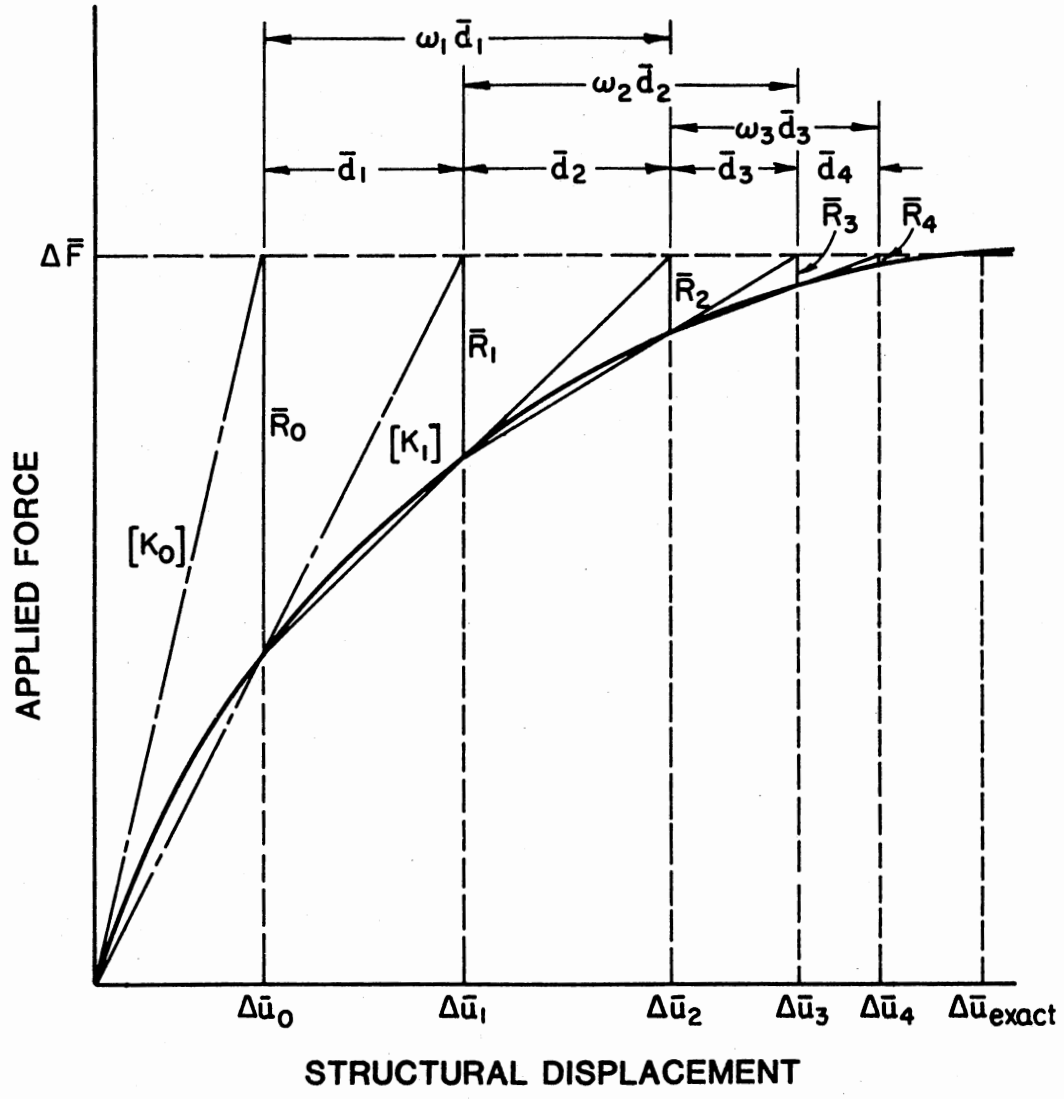

Figure 11. Direct Iteration With Linear Scalar Extrapolation Extended

Referring to Figure 9, it can be seen that by considering areas as quantities of work, the extrapolation may take place without concern for units. In other words, equating the work represented by the area of the trapezoid ABDE plus the work represented by the area of the triangle BCD to that of the triangle ACE yields,

$$A_{ABDE} + A_{BCD} = A_{ACE} \tag{3.2}$$

Next, substituting proper values for the work leads to:

$$\frac{1}{2}(\bar{R}_o + \bar{R}_1)\cdot\bar{d} + \frac{1}{2}\bar{R}_1\cdot(w - 1)\,\bar{d} = \frac{1}{2}\bar{R}_o\cdot w\bar{d} \tag{3.3}$$

which reduces to:

$$w = \frac{\bar{R}_o \cdot \bar{d}}{(\bar{R}_o - \bar{R}_1)\cdot\bar{d}} \tag{3.4}$$

Comparison of Equation (3.4) with (3.1) shows that no difference exists for a one degree-of-freedom problem and that the difficulty of generalized units has been removed.

Discussion

From Figure 10 it should be clear that by extrapolating to obtain the deflection $\Delta\bar{u}_2$, a major source of solution expense, factoring of the stiffness matrix, is eliminated. Furthermore, as seen in Figure 11, a possible savings in the total number of iterations may result from use of the linear scalar extrapolation. The reason is that the linear scalar extrapolation is a method which resembles a trangent stiffness solution procedure without the stability problems. The earliest that program logic in a Newton-Raphson procedure may detect a buckling point is when the stiffness matrix becomes very ill-conditioned. The linear scalar

extrapolation program logic will recognize the same point when the scalar w multiplied by the deflection $\bar{d}$ becomes very large; however, this had not been observed to be a frequent occurrence. The condition is easily recognized and can be compensated when it does occur.

Although not apparent from the previous development, the linear scalar extrapolation is a line search in the direction of $\bar{d}_1$. If the convergence criteria are treated as a performance function, an optimization technique such as the golden section search [7] could be used to determine w. This approach is not attempted in this dissertation.

When the need exists for highly accurate deformations, the convergence criteria is altered so as to require more iterations as illustrated in Figure 12, because, in general, more accurate results occur with each successive iteration. With this increased requirement position 5 must be reached before convergence can be satisfied whereas position 4 was sufficient for the original convergence criteria. In fact, if the convergence criteria are reduced to the inner circle, then the direction of search would not intersect the n-dimensional sphere of convergence, and position 6 would be the closest the procedure could approach to the exact solution at position A.

If the convergence criteria were such as to disallow convergence to solution, a new search direction would need to be established. This could be done in one of many ways. Two that could be used are:

1. Rename position 6 in Figure 12 as $\Delta\bar{u}_o$ and then repeat the procedure listed in Figure 8. That is, calculate a new stiffness matrix as a function of $\Delta\bar{u}_o$, factor it, multiply by the load, obtain $\Delta\bar{u}_1$; then the difference between $\Delta\bar{u}_o$ and $\Delta\bar{u}_1$ would be the new search direction.

2. Through a means described in the following text, determine a

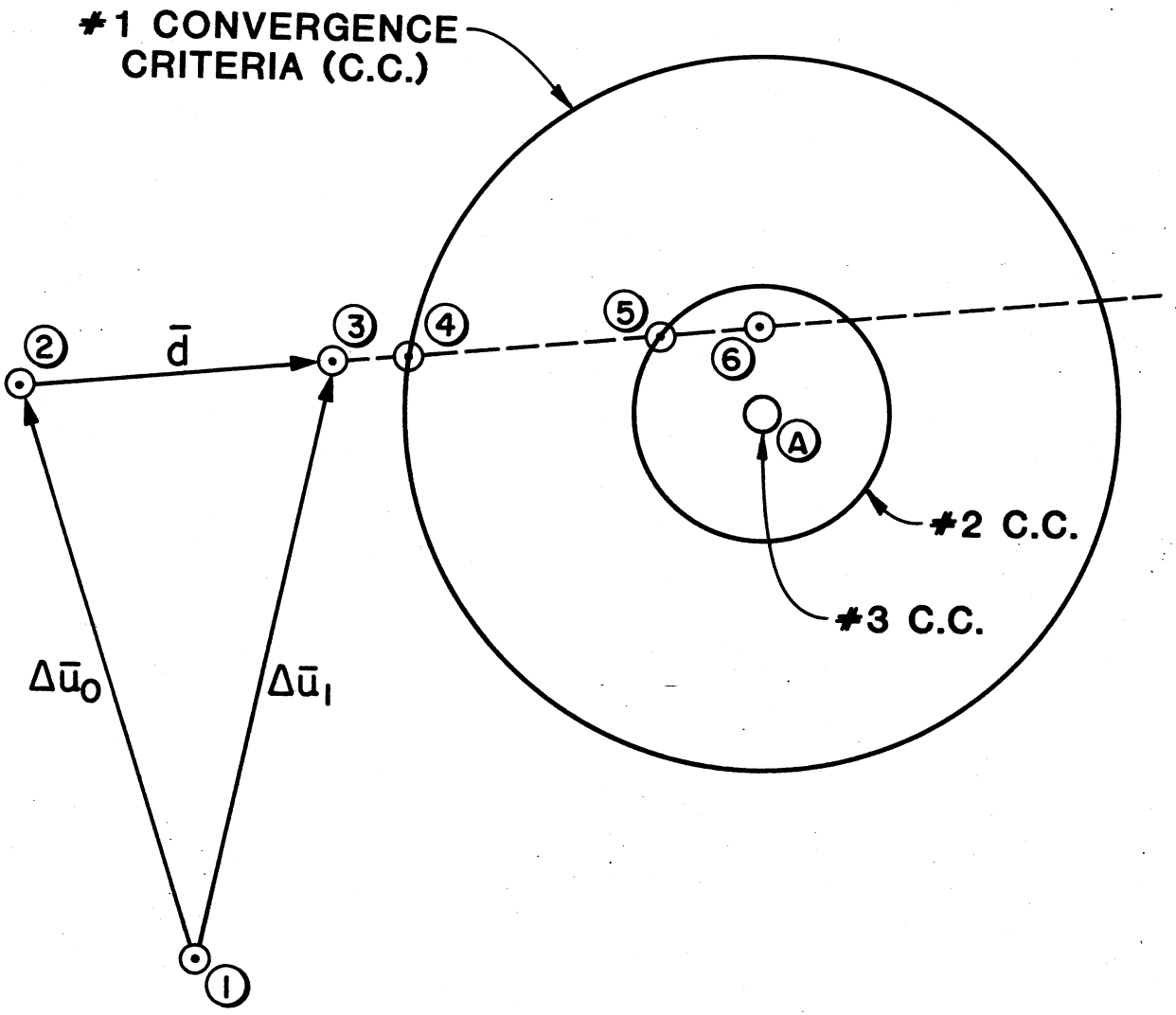

Figure 12. Effect of Convergence Criteria

search direction orthogonal to the original search direction without factoring a global stiffness matrix. The solution procedure would then be the same as before.

The second of the two possibilities mentioned above was selected for investigation because it continues in the mainstream of cost reduction procedures. In fact, if this method could solve all problems of interest, only two global stiffness matrices will have been used for the entire solution procedure at an incremental load step.

Multiple Search Extrapolation

Development

The development of the multiple search extrapolation procedure begins with a close inspection of the linear scalar extapolation. Reproducing part of Figure 10 for convenience,

$$\left.\begin{aligned} \{u_o\} &= [K_o]^{-1}\ \{\Delta F\} \\ [K_1] &= K(\Delta \bar{u}_o) \\ \{R_1\} &= \{\Delta F\} - \Sigma[k_1]\{\delta_o\} \end{aligned}\right\} \quad \text{Iteration 1}$$

$$\left.\begin{aligned} \{\Delta u_1\} &= [K_1]^{-1}\ \{\Delta F\} \\ [K_2] &= K(\Delta \bar{u}_1) \\ \{R_2\} &= \{\Delta F\} - \Sigma[k_2]\{\delta_1\} \end{aligned}\right\} \quad \text{Iteration 2}$$

$$\left.\begin{aligned} \{d_1\} &= \{\Delta u_1 - \Delta u_o\} \qquad (3.5) \\ w_1 &= \frac{\{R_o\}\cdot\{d_1\}}{[\{R_o\} - \{R_1\}]\cdot\{d_1\}} \qquad (3.6) \\ \{\Delta u_2\} &= \{\Delta u_o\} + w\{d_1\} \qquad (3.7) \end{aligned}\right\} \quad \begin{matrix}\text{Iteration 3} \\ \text{(Extrapolated)}\end{matrix}$$

Inspection of Equation (3.5) reveals that vector $\bar{d}_1$ is a correction direction to the vector Δu_o, as determined by the vector $\Delta\bar{u}_1$. The change in residual from $\bar{R}_o$ to $\bar{R}_1$ is an indicator of the accuracy of the change in displacement. The step size, w_1, along a search direction is determined by calculation from the change in two successive residuals and from the distance, $\bar{d}_1$, between the residuals.

An inspection of Equation (3.6) shows that the entire work done by $\bar{R}_o$ and $\bar{R}_1$ in moving through the distance $\bar{d}_1$ is considered without regard to the amount of work contributed by each generalized force in the calculation of w_1. Consequently, this lumping of terms requires the alteration of each generalized displacement by the same multiplier, w, even though some are greater contributors to the energy expression than others.

A method to select a new direction of search based on appropriately weighting the scalar w for each degree of freedom must consider the effect of that particular degree of freedom on the calculation of w_1. This is done in a reverse fashion by removing the energy terms due to an individual degree of freedom from Equation (3.6). This results in the equation,

$$w_p^* = \frac{\{R_o\}\cdot\{d_1\} - (R_{op})d_{1p}}{[\{R_o\} - \{R_1\}]\cdot\{d_1\} - (R_{op} - R_{1p})d_{1p}} \tag{3.8}$$

where the subscript, p, represents the individual component of the various vectors associated with the p^{th} degree of freedom and w_p^* represents a new scalar without the effect of the p^{th} degree of freedom being used.

The term w_p^* provides a means by which to determine whether the original w extrapolation of Equation (3.6) is appropriate, relative to the other degrees of freedom. Therefore, Equation (3.8) is the additional

information necessary to help determine a new direction of search in the event that the old direction fails.

Ideally, the search history would proceed as shown in Figure 13. The original search direction, $\bar{d}$, is as shown in Figure 13. Position 4 represents the closest point of approach to the exact solution at position A. After reaching position 4, a new search direction, $\bar{d}^*$, is found. Once the new direction has been established, convergence is satisfied when position 5 or any other position inside the convergence criteria sphere is reached.

A loss of effectiveness will result from searching in the direction passing through the position 6. This unnecessarily poor direction of search could result from failure to insure that the direction $\bar{d}^*$ be orthogonal to the original direction, $\bar{d}$, and all prior search directions. The manner of calculating the new search direction is as follows:

$$w = \frac{\{R_o\}\cdot\{d_1\}}{[\{R_o\} - \{R_1\}]\cdot\{d_1\}} \tag{3.6}$$

$$w_p^* = \frac{\{R_o\}\cdot\{d_1\} - (R_{op})d_{1p}}{[\{R_o\} - \{R_1\}]\cdot\{d_1\} - (R_{op} - R_{1p})d_{1p}} \tag{3.8}$$

$$\Delta w_{pp} = w - w_p^* \tag{3.9}$$

$$\{d\}^* = [\Delta w]\{d_1\} - \left[\frac{([\Delta w]\{d\})\cdot\{d\}}{\{d\}\cdot\{d\}}\right]\{d\} \tag{3.10}$$

where the matrix $[\Delta w]$ is a diagonal matrix consisting of Δw_{pp} diagonal terms and zeros as the off-diagonal terms. The term in brackets in Equation (3.10) represents the scalar term that defines the portion of the vector $[\Delta w]\{d\}$ that is in the $\bar{d}$ direction. This term is subtracted to preserve an orthogonal search pattern.

The entire solution procedure is presented in Figure 14. Iteration

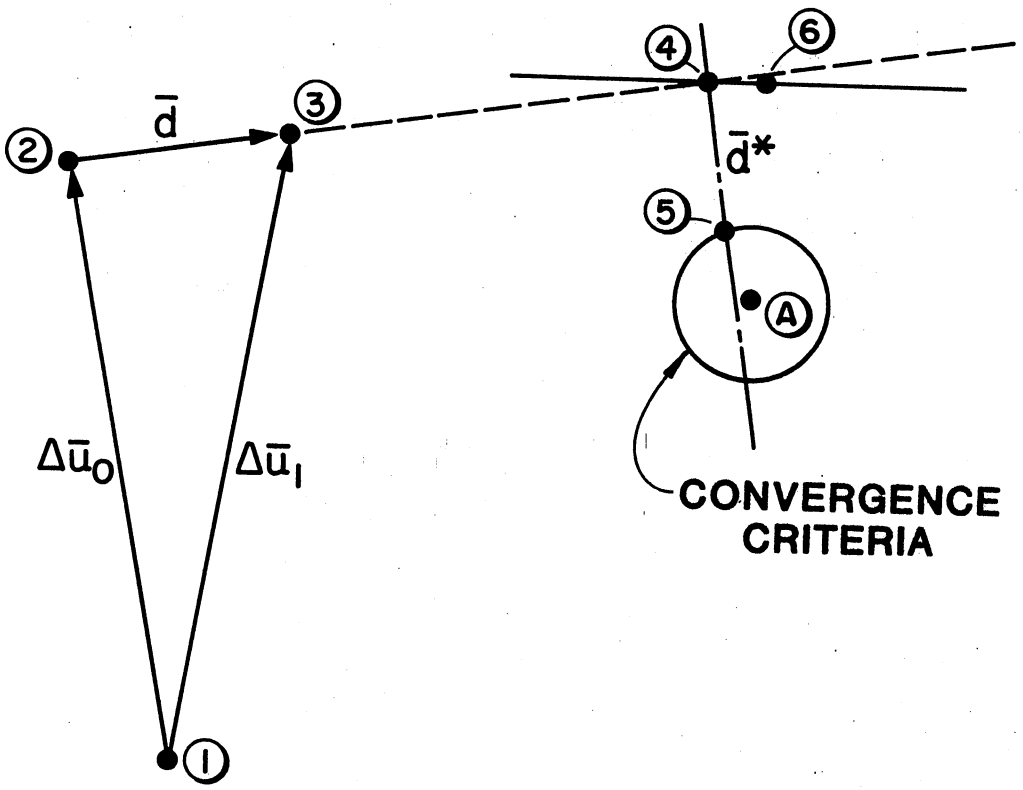

Figure 13. Direct Iteration With Multiple Search Extrapolation

$$\left.\begin{aligned} \{\Delta u_o\} &= [K_o]^{-1}\ \{\Delta F\} \\ [K_1] &= K(\Delta \bar{u}_o) \\ \{R_o\} &= \{\Delta F\} - \Sigma[k_1]\{\delta_o\} \end{aligned}\right\} \text{Iteration 1}$$

$$\left.\begin{aligned} \{\Delta u_1\} &= [K_1]^{-1}\ \{\Delta F\} \\ [K_2] &= K(\Delta \bar{u}_1) \\ \{R_1\} &= \{\Delta F\} - \Sigma[k_2]\{\sigma_1\} \end{aligned}\right\} \text{Iteration 2}$$

$$\left.\begin{aligned} \{d_1\} &= \{\Delta u_1 - \Delta u_o\} \\ \omega_1 &= \frac{\{R_o\}\cdot\{d_1\}}{[\{R_o\} - \{R_1\}]\cdot\{d_1\}} \\ \{\Delta u_2\} &= \{\Delta u_o\} + \omega_1\ \{d_1\} \\ [K_3] &= K(\Delta \bar{u}_2) \\ \{R_2\} &= \{\Delta F\} - \Sigma[k_3]\{\delta_2\} \end{aligned}\right\} \begin{array}{l}\text{Iteration 3} \\ \text{(Extrapolated)}\end{array}$$

$$\left.\begin{aligned} &\vdots \end{aligned}\right\} \begin{array}{l}\text{Iteration 4 -} \\ \text{Iteration n}\end{array}$$

$$\left.\begin{aligned} \Delta\omega_{pp} &= \omega_n - \frac{\{R_{n-1}\}\cdot\{d_n\} - (R_{n-1p})d_{np}}{[\{R_{n-1}\} - \{R_n\}]\cdot\{d_n\} - (R_{n-1p} - R_{np})d_{np}} \\ \{d\}^* &= [\Delta\omega]\{d_n\} - \left[\frac{([\Delta\omega]\{d_n\})\cdot\{d_n\}}{\{d_n\}\cdot\{d_n\}}\right]\{d_n\} \end{aligned}\right\} \text{Direction Change}$$

$$\left.\begin{aligned} \omega_{n+1} &= \text{Some arbitrary constant} \\ \{\Delta u_{n+1}\} &= \{\Delta u_n\} + \omega_{n+1}\ \{d\}^* \\ [K_{n+2}] &= K(\Delta u_{n+1}) \\ \{R_{n+1}\} &= \{\Delta F\} - \Sigma[k_{n+2}]\{\delta_{n+1}\} \end{aligned}\right\} \begin{array}{l}\text{Iteration n+1} \\ \text{(Extrapolated)}\end{array}$$

$$\{d_{n+1}\} = \{\Delta u_{n+1} - \Delta u_n\}$$

$$\omega_{n+2} = \frac{\{R_n\}\cdot\{d_{n+1}\}}{[\{R_n\} - \{R_{n+1}\}]\cdot\{d_{n+1}\}}$$

$$\vdots$$

To Convergence

Figure 14. Direct Iteration With Multiple Search Extrapolation

4 through iteration n are represented as three dots to indicate a continuity pattern. Once the new search direction, $\bar{d}^*$, has been determined, an arbitrary magnitude in that direction is necessary to start the solution procedure forward again. Care in this step size selection, w_{n+1}, is required because an excessively large one might cause accuracy or convergence problems.

Discussion

The multiple search extrapolation procedure is extremely sensitive to the calculation of the residuals; therefore, the residuals must be calculated as accurately as possible. One requirement is that all pertinent data be updated at the end of each iteration rather than at the end of the load step. However, the method is fail-safe. If an incorrect direction of search is determined, the method simply returns the solution for the incremental displacement to zero because of the large residuals which result.

CHAPTER IV

MODIFICATION OF COMPUTER PROGRAM

The major finite element programs used in this study are mentioned in Chapter I and referenced in the Bibliography. All three codes are very large multi-purpose programs used for solving nonlinear structural equations with both geometric and material nonlinearities and possessing both static and dynamic capabilities. In all three cases, in order to implement the algorithms presented in Chapter III, it was necessary to identify the subroutine in which the incremental displacements are calculated. If the residual loads were not previously available a significant change in several subroutines was necessary for their determination.

In order to provide a better understanding of the computer code modification, only the TRAINS (Transient Analysis of Nonlinear Systems) code is discussed. The major portion of the modifications are in subroutine ITERCK (see Table I).

Prior to modification subroutine ITERCK performed the dual roles of determining if convergence had been satisfied and of updating the total displacements so that terms dependent upon displacement could be calculated. In the modified form the subroutine ITERCK performs all of the following:

1. Checks for convergence and maximum iteration limit.
2. Initializes almost all variables needed for extrapolation.
3. Calculates the residuals.

TABLE I

SUBROUTINE ITERCK

```
      SUBROUTINE ITERCK (TF,DF,TA,DD,NAPRH,NPRH,SXK,NDEG,NUMNP,
     1IHIB,IXLOAD)
      DIMENSION TF(NDEG,NUMNP),DF(NDEG,NUMNP),TA(NDEG,NUMNP),
     1DD(NDEG,NUMNP),NAPRH(NUMNP),SXK(NDEG,NDEG,1),VARS(1)
      COMMON/CDC/DUM1(1),STEP,DUM4(4)
      COMMON/MINBAK/IG,ITEM1T,ITEM2T,ITEM3T
      COMMON/VECOUT/NUMOUT
      COMMON/VECRAN/ITEMP1,ITEMP2,ITEMP3,ND64,NSZCRD
      COMMON/NEWVEC/IOMEGA,ITFSAV,INCF,KOUNT,IPMULT,IBUFFR
      COMMON/NEWVCT/ITEMK1,ITEMK2
      COMMON/REMEMB/IREM1,IREM2,IREM3,IREM4,IREM5
      COMMON/NEWFAC/XPMULT
      COMMON/CYCLE/MINCYC,MAXCYC,TOLER,ICYC,IFSAVE,IR0,IR1,
     1IDDSAV,IOK,ENERPR
     1,ITERATE
      COMMON/STATIC/ISTAT
      COMMON/NZR/NEQST,NSTRES,LODCOR
      COMMON/ARRAY2/DUM25(25),IRES1,DUM5(5)
      COMMON/MONDEG/ITDEG,ITNODE
      COMMON/SEQFIL/KFILE
      COMMON/SPACE/INTS(1)
      EQUIVALENCE(VARS(1),INTS(1))
C*****  INITIALIZE VARIOUS TERMS NEEDED WITH DUE REGARD TO WHEN
C*****  THEY ARE NEEDED.  ************************************
      IOK = 0
      ND=NDEG*NUMNP
      IDIR1=IBUFFR
      IDIR2=IDIR1+ND
      IDIR3=IDIR2+ND
      IDIR4=IDIR3+ND
      IDIR5=IDIR4+ND
      IBUF51=IDIR5
      IBUF52=IBUF51+1
      DWDUD1=0.0
      DWDUD2=0.0
      DWDUD3=0.0
      DWDUD4=0.0
      D1D1=1.0
      D2D2=1.0
      D3D3=1.0
      D4D4=1.0
      IF(ICYC.NE.0) GO TO 920
      XPMULT=0.0
      CALL SCLA(VARS(IDIR1),0.0,NDEG,NUMNP,0)
      CALL SCLA(VARS(IDIR2),0.0,NDEG,NUMNP,0)
      CALL SCLA(VARS(IDIR3),0.0,NDEG,NUMNP,0)
      CALL SCLA(VARS(IDIR4),0.0,NDEG,NUMNP,0)
      IREM1=1
      IREM2=0
  920 CONTINUE
```

TABLE I (Continued)

```
  920 CONTINUE
C    KOUNT IS A VARIABLE INITIALIZED IN SUBROUTINE MARC.
       KOUNT=KOUNT+1
       WRITE(6,1021) KOUNT
 1021 FORMAT(* KOUNT=*,I10)
       ROD=0.0
       R1D=0.0
       FACT = (1.0/STEP**2)
       CALL READMS(8,DD,ND,3)
       ENERGY = DD(ITDEG,ITNODE)
       IF(IHIB.EQ.0) ENERPR=ENERGY
       PERCH=ABS((ENERGY-ENERPR)/ENERPR)*100.0
       WRITE (6,100) ENERGY
  100 FORMAT (* INCREMENT OF DISPLACEMENT IN THIS ITERATION*,E15.7)
       WRITE (6,105) ENERPR
  105 FORMAT (*  INCREMENT OF DISPLACEMENT IN PREVIOUS ITERATION*,E15.7)
       WRITE (6,110) PERCH
  110 FORMAT (* PERCENTAGE DIFFERENCE*,E15.7)
       ENERPR = DD(ITDEG,ITNODE)
       IF(PERCH.LE.TOLER) IOK = 1
       IF(IHIB.EQ.0) GO TO 900
       IF(IOK.NE.1.AND.ICYC.EQ.MAXCYC) IOUT=1
       IF(IOUT.EQ.1) WRITE(6,600)
 600   FORMAT(/////, 20X,*  MAXIMUM SPECIFIED ITERATIONS EXCEEDED*)
       IF(IOUT.EQ.1) IOK=1
       IF(IOUT.EQ.1) GO TO 900
       IF(IOK.NE.1) GO TO 200
       GO TO 900
 200   CONTINUE
C  *   *  *  COMPUTE RESIDUALS
       CALL WRITMS(8,VARS(IDDSAV),ND64,7,-1)
       IF(ICYC.EQ.0) CALL WRITMS(8,VARS(IDDSAV),ND64,8,-1)
       CALL WRITMS (8,VARS(IXLOAD),NUMOUT,1,-1)
       IF(ISTAT.EQ.1) GO TO 800
       CALL WRITMS(8,TA,ND64,7,-1)
       CALL READMS(8,VARS(ITEM2T),ND64,7)
C  *   *  *  COMPUTE MASS TIMES ACCELERATION
       KFILE = 22
       CALL UPTX(SXK,NPRH,VARS(ITEM1T),VARS(ITEM2T),NDEG,NDEG,NUMNP,
      1NDEG,NAPRH,NDEG,NDEG)
  800 CONTINUE
C * * * STRESS PART OF RESIDUAL COMPUTED IN STRAIN, IN VARS(ITEMP3)
C * * * BRING BACK B2TSIGMA VECTOR (LODCOR)
       IF(LODCOR.EQ.2) GO TO 805
       IF(ISTAT.EQ.0) CALL GMADD(VARS(ITEM2T),VARS(ITEM1T),VARS(ITEM2T)
      *                          ,NDEG,NUMNP)
  805 CONTINUE
       CALL READMS(8,VARS(IXLOAD),NUMOUT,1)
       CALL GMADD(LF,TF,VARS(ITEMP2),NDEG,NUMNP)
       IF(LODCOR.EQ.2) CALL GMADD(VARS(ITEMP2),VARS(IRES1),VARS(ITEMP2),
```

TABLE I (Continued)

```
      IF(LODCOR.EQ.2) CALL GMADD(VARS(ITEMP2),VARS(IRES1),VARS(ITEMP2),
     *                           NDEG,NUMNP)
      IF(ISTAT.EQ.0) CALL SMPY(VARS(ITEMP2),FACT,VARS(ITEMP2),NDEG,
     *                         NUMNP,0)
      CALL GMADD(VARS(ITEMP2),VARS(ITEMP3),VARS(ITEMP2),NDEG,NUMNP)
C  *  *  *  THE RESIDUAL FOR THE ITERATION JUST COMPLETED
C  *  *  *  IN VARS(ITEMP2)
C  *  *  *  NOW PUT THE RESIDUAL FOR THE PREVIOUS ITERATION IN R0
      CALL MCPY(VARS(IR1),VARS(IR0),NDEG,NUMNP,0)
C  *  *  *  AND PUT THE CURRENT RESIDUAL IN R1
      CALL MCPY (VARS(ITEMP2),VARS(IR1),NDEG,NUMNP,0)
      ICYC=ICYC+1
C  *  *  *  IF TWO RESIDUAL VECTORS HAVE NOT BEEN COMPUTED,
C  *  *  *  CANNOT EXTRAPOLATE
      IF(ICYC.LE.1) GO TO 900
C  *  *  *  COMPUTE THE INCREMENT OF DISPLACEMENT IN THE ITERATION
      CALL READMS(8,DD,ND,3)
      IF(ICYC.GT.2) GO TO 880
      CALL MCPY(VARS(IDDSAV),DD,NDEG,NUMNP,0)
      CALL READMS(8,VARS(IDDSAV),ND,8)
      CALL GMSUB(DD,VARS(IDDSAV),VARS(ITEMP2),NDEG,NUMNP)
      CALL WRITMS(8,DD,ND64,8,-1)
      CALL MCPY(VARS(IDDSAV),VARS(ITEMP1),NDEG,NUMNP,0)
      GO TO 885
 880  CONTINUE
      CALL READMS(8,VARS(ITEMP1),ND64,8)
      CALL GMSUB(VARS(IDDSAV),VARS(ITEMP1),VARS(ITEMP2),NDEG,NUMNP)
      CALL WRITMS(8,VARS(IDDSAV),ND64,8,-1)
 885  CONTINUE
C  *  *  *  COMPUTE THE ROD AND R1D SCALARS
      NUMBER=NDEG*NUMNP
      CALL GMPRD(VARS(IR0),VARS(ITEMP2),ROD,1,NUMBER,1)
      CALL GMPRD(VARS(IR1),VARS(ITEMP2),R1D,1,NUMBER,1)
      OMEGA=ROD/(ROD-R1D)
C*****  KEEP TRACK OF HOW FAR WE EXTRAPOLATE IN A GIVEN DIRECTION
C       BY DETERMINING THE MULTIPLE OF THE ORIGINAL INCREMENT IN
C       THAT DIRECTION AND SPECIFYING IT AS XPMULT.              *******
      IREM2=IREM2+1
      IN=1
      INN=0
  946 GO TO (947,949),IN
  947 IF(VARS(ITEMP2+INN).EQ.0.) GO TO 948
      IN=IN+1
      IF(IREM2.EQ.1) VARS(IBUF51)=VARS(ITEMP2+INN)
      XPMULT=XPMULT+VARS(ITEMP2+INN)/VARS(IBUF51)
  948 INN=INN+1
      IF(INN.GT.NUMBER) GO TO 952
      GO TO 946
  952 WRITE(6,951)
  951 FORMAT(* GARBAGE1*)
```

TABLE I (Continued)

```
  951 FORMAT(* GARBAGE1*)
      STOP
  949 CONTINUE
      WRITE(6,950) OMEGA,IREM1,XPMULT
  950 FORMAT(* OMEGA=*,E15.7,5X,*FOR DIRECTION *,I3,* XPMULT=*,E15.7)
      CALL GMPRD(VARS(ITEMP2),VARS(ITEMP2),DUDU,1,NUMBER,1)
C*****  DETERMINE IF A NEW SEARCH DIRECTION IS NEEDED.  ***************
      IF(ABS((XPMULT-VARS(IBUF52))/XPMULT).GE.1.0E-2.OR.
     *      IREM2.LT.4.OR.IREM1.GT.4) GO TO 1010
C*****  CALCULATE THE NEW SEARCH DIRECTION BASED ON THE RESIDUALS
C       AND HOW MUCH THEY AFFECT THE DETERMINATION OF OMEGA.     *******
      DO 1000 I=1,NUMBER
      NOMEGA=IOMEGA+I-1
      VARS(NOMEGA)=0.0
      IF(ABS(VARS(ITEMP2+I-1)).LT.1.0E-60) GO TO 1000
      PROD=VARS(IR0+I-1)*VARS(ITEMP2+I-1)
      PR1D=VARS(IR1+I-1)*VARS(ITEMP2+I-1)
      POMEGA=(ROD-PROD)/(ROD-R1D-PROD+PR1D)
      VARS(NOMEGA)=(OMEGA-POMEGA)*VARS(ITEMP2+I-1)
 1000 CONTINUE
C*****  DEFINE THE TERMS NECESSARY TO INSURE THE NEW SEARCH DIRECTIONS
C       ARE IN FACT ORTHOGONAL.                                       **
      CALL GMPRD(VARS(IOMEGA),VARS(ITEMP2),DWDUDU,1,NUMBER,1)
      GO TO (5,4,3,2,1),IREM1
    1 CALL GMPRD(VARS(IOMEGA),VARS(IDIR4),DWDUD4,1,NUMBER,1)
      IF(D4D4.EQ.1.) CALL GMPRD(VARS(IDIR4),VARS(IDIR4),D4D4,1,NUMBER,1)
    2 CALL GMPRD(VARS(IOMEGA),VARS(IDIR3),DWDUD3,1,NUMBER,1)
      IF(D3D3.EQ.1.) CALL GMPRD(VARS(IDIR3),VARS(IDIR3),D3D3,1,NUMBER,1)
    3 CALL GMPRD(VARS(IOMEGA),VARS(IDIR2),DWDUD2,1,NUMBER,1)
      IF(D2D2.EQ.1.) CALL GMPRD(VARS(IDIR2),VARS(IDIR2),D2D2,1,NUMBER,1)
    4 CALL GMPRD(VARS(IOMEGA),VARS(IDIR1),DWDUD1,1,NUMBER,1)
      IF(D1D1.EQ.1.) CALL GMPRD(VARS(IDIR1),VARS(IDIR1),D1D1,1,NUMBER,1)
    5 CONTINUE
C*****  CALCULATE THE NEW SEARCH DIRECTION.    **************************
      DO 1005 I=1,NUMBER
      NOMEGA=IOMEGA+I-1
      DWNL=VARS(IOMEGA+I-1)-(DWDUDU/DUDU)*VARS(ITEMP2+I-1)
     *                     -(DWDUD1/D1D1)*VARS(IDIR1+I-1)
     *                     -(DWDUD2/D2D2)*VARS(IDIR2+I-1)
     *                     -(DWDUD3/D3D3)*VARS(IDIR3+I-1)
     *                     -(DWDUD4/D4D4)*VARS(IDIR4+I-1)
C*****  SAVE THE NEW DIRECTION FOR FUTURE USE.  ***********************
      NIDIR=(IBUFFR+(IREM1-1)*ND)+I-1
      IF(IREM1.LT.5) VARS(NIDIR)=DWNL
      VARS(NOMEGA)=DWNL
 1005 CONTINUE
      IREM1=IREM1+1
C*****  SAVE THE FIRST DIRECTION FOR FUTURE USE.  ************
      IF(IREM1.EQ.2) CALL MCPY(VARS(ITEMP2),VARS(IDIR1),
     *                         NDEG,NUMNP,0)
```

TABLE I (Continued)

```
     *                                NDEG,NUMNP,0)
      IREM2=0
      CALL MCPY(VARS(IOMEGA),VARS(ITEMP2),NDEG,NUMNP,0)
      IN=1
      INN=0
 1046 GO TO (1047,1049),IN
 1047 IF(VARS(ITEMP2+INN).EQ.0.) GO TO 1048
      IN=IN+1
      OMEGA=(0.001*XPMULT)*(VARS(IBUF51)/VARS(ITEMP2+INN))
 1048 INN=INN+1
      IF(INN.GT.NUMBER) GO TO 1050
      GO TO 1046
 1050 WRITE(6,1051)
 1051 FORMAT(* GARBAGE2*)
      STOP
 1049 CONTINUE
      XPMULT=0.0
      CALL MCPY(VARS(IR0),VARS(IR1),NDEG,NUMNP,0)
 1010 CONTINUE
      CALL SMPY(VARS(ITEMP2),OMEGA,VARS(ITEMP2),NDEG,NUMNP,0)
      VARS(IBUF52)=XPMULT
      WRITE(6,1011) XPMULT
 1011 FORMAT(* PMULT=*,E15.6)
      WRITE(6,1784) VARS(IR1),VARS(IR1+1),VARS(IR1+2),
     *              VARS(IR1+3),VARS(IR1+4),VARS(IR1+6),
     *              VARS(IR1+ND-6),VARS(IR1+ND-5),VARS(IR1+ND-4),
     *              VARS(IR1+ND-3),VARS(IR1+ND-2),VARS(IR1+ND-1)
 1784 FORMAT(* IR1= *,6(E11.4),/,* IR1(LAST SIX DOF)= *,6(E11.4))
      CALL GMADD(VARS(ITEMP2),VARS(ITEMP1),DD,NDEG,NUMNP)
      ENERPR=DD(ITDEG,ITNODE)
      CALL WRITMS(8,DD,ND64,3,-1)
  900 RETURN
      END
```

4. Determines the scalar w.

5. Determines the new search directions and insures orthogonality to all previous search directions.

6. Updates the total displacements with the incremental displacements.

Subroutine ITERCK is listed in Table I.

Almost all large nonlinear structural programs use dynamic allocation. Because of dynamic allocation, the implementation of the algorithms presented in this study may be tedious. There are a number of subtle special cases at branch points, particularly in the transition from one search direction to another. However, the effort will be cost effective.

CHAPTER V

APPLICATION EXAMPLES

This chapter contains two examples verifying the usefulness of the linear scalar extrapolation, two examples of application of the multiple search extrapolation, and a discussion of a problem involving material and geometric nonlinearities. The first two examples are a bar-spring buckling problem with one degree of freedom and a pressure loaded ring with twenty-four degrees of freedom. The second two examples are a radially loaded ring with twenty-eight degrees of freedom and a pressure loaded ring with twenty-eight degrees of freedom. The final problem is a ring-stiffened cylinder under pressure loading. This chapter also contains a discussion of the ability of the proposed algorithms to facilitate the solution of each problem as indicated by the results.

Linear Scalar Extrapolation

Bar-Spring Problem

To demonstrate the validity of the linear scalar extrapolation procedure, the one degree-of-freedom bar-spring problem which was previously solved by Jones [1] is presented. The bar-spring problem is illustrated in Figure 15. The length of the spring is unimportant if nonlinear effects do not enter into the calculations for the deflection of the spring. The bar was modeled so as to allow only a change in length and no flexural or shear deformation.

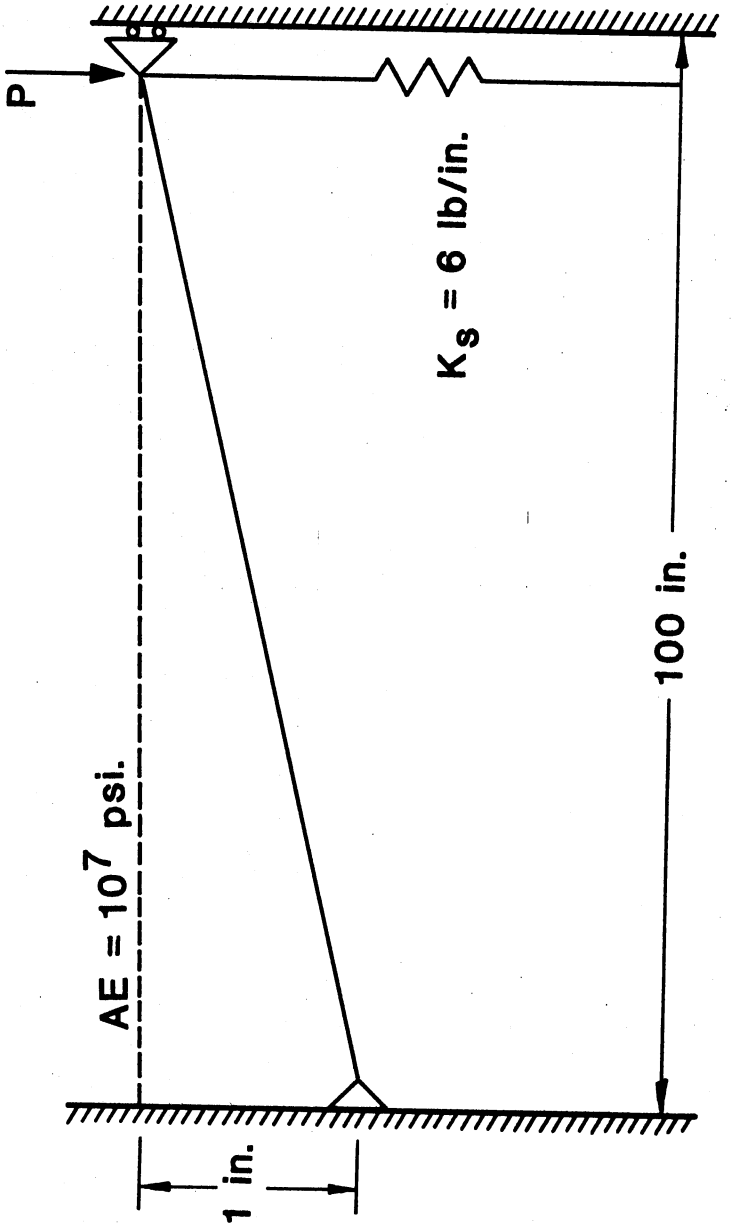

Figure 15. Bar-Spring Problem

A varying load, P, was applied at the end of the bar where the spring is attached as shown in Figure 15. Clearly, the problem has only a single degree of freedom with a highly nonlinear deflection response to the applied load, as can be seen in Figure 16 in which the exact response is graphically illustrated. The buckling load for this problem with the stated parameters is exactly six pounds.

In determining the results (Table II), three forms of calculation were used. The exact values were determined analytically. The "old" values were calculated using the direct iteration procedure without the linear scalar extrapolation. The "new" values were calculated with the linear scalar extrapolation modification included. The parameter used for comparison in Table II is the number of iterations (small raised superscript) required to satisfy the convergence criteria indicated at the column heading. The convergence criteria represent a ratio of two successive incremental displacements in the solution history.

It should be noted that at the buckling load the tolerance required to obtain two significant digits accuracy, 1.001, the linear scalar extrapolation required five iterations as compared to twenty-six iterations for the unmodified approach. However, the new method required three iterations more than the four performed by the unmodified procedure in the post-buckled region because the linear extrapolation does not follow the hardening stiffness of the structure very well. For buckling analysis where displacements are required, the greatest interest is generally in the pre-buckled region which is where the greatest savings in the number of iterations occurred, since not only were fewer iterations made, but each iteration was less expensive due to the linear extrapolation bypassing the factorization of the global stiffness matrix.

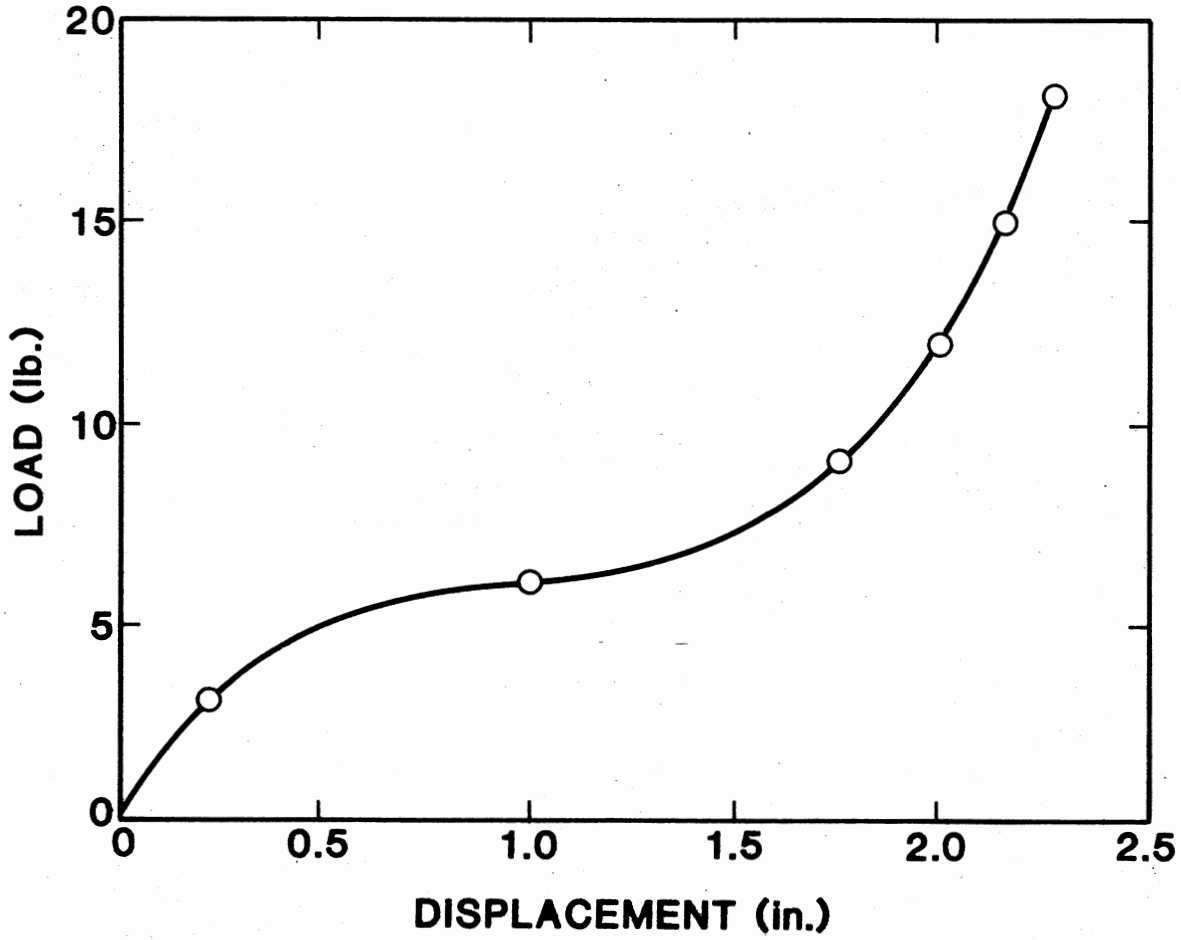

Figure 16. Load Versus Displacement for Bar-Spring Problem

TABLE II

BAR-SPRING PROBLEM

Load (lb)	Method	Exact (in.)	Convergence Criteria 2.0	1.5	1.1	1.01	1.001	1.0001	1.00001
0.1	Old		$0.0062506^{}$						
	New		$0.0062506^{}$						
		0.23536							
3.0	Old		0.22458^{0}	0.22458^{0}	0.23288^{1}	0.23479^{2}	0.23534^{4}	0.23537^{6}	0.23537^{7}
	New		0.22458^{0}	0.22458^{0}	0.23288^{1}	0.23481^{3}	0.23535^{5}	0.23537^{6}	0.23537^{7}
		1.0000							
6.0*	Old		0.59355^{0}	0.68192^{1}	0.79436^{3}	0.96634^{13}	0.99685^{26}	0.99970^{39}	1.0000^{51}
	New		0.59355^{0}	0.68192^{1}	0.87911^{2}	0.98498^{4}	0.99892^{5}	1.0000^{6}	1.0000^{6}
		1.7646							
9.0	Old		1.7957^{1}	1.7910^{1}	1.7690^{1}	1.7640^{1}	1.7646^{1}	1.7646^{1}	1.7646^{1}
	New		1.7957^{1}	1.7910^{1}	1.7633^{1}	1.7644^{1}	1.7646^{1}	1.7646^{1}	1.7646^{3}
		2.0000							
12.0	Old		1.9399^{0}	2.0040^{1}	1.9997^{2}	2.0000^{3}	2.0000^{4}	2.0000^{4}	2.0000^{5}
	New		1.9399^{0}	2.0040^{1}	2.0011^{3}	2.0001^{5}	2.0000^{7}	2.0000^{8}	2.0000^{9}
		2.1617							
15.0	Old		2.1752^{0}	2.1540^{0}	2.1631^{1}	2.1617^{3}	2.1617^{4}	2.1617^{6}	2.1617^{6}
	New		2.1752^{0}	2.1540^{0}	2.1633^{1}	2.1618^{5}	2.1617^{7}	2.1617^{9}	2.1617^{10}
		2.2891							
18.0	Old		2.2686^{0}	2.2863^{0}	2.2904^{1}	2.2892^{3}	2.2891^{4}	2.2891^{6}	2.2891^{7}
	New		2.2686^{0}	2.2863^{0}	2.2904^{1}	2.2893^{5}	2.2892^{7}	2.2891^{9}	2.2891^{11}

*Buckling load.

Note: When the number of iterations (displayed as superscripts) is less than $(^{2})$, there is NO difference between the new and old methods.

Pressure Loaded Ring

The problem of a pressure loaded ring is presented to demonstrate that with the linear scalar extrapolation, deflections in a pre-buckled region can be calculated less expensively and more accurately than without the extrapolation. The uniformly applied external pressure is allowed to vary up to the buckling load of 60 psi (pounds per square inch). The ring is modeled through 90 degrees at shown in Figure 17. The 90-degree arch is divided into two substructures. The degrees of freedom per node are:

1. Z--Lateral translation
2. R--Vertical translation
3. dZ/ds--Rotation
4. dR/ds--Rotation.

The rotations are defined as positive in the direction indicated by θ shown in Figure 17. The ring is modeled with a modulus of elasticity of 30×10^6 psi and a radius of 20 inches. A kicker force is applied at node 1 of substructure 1 in the negative R direction with a magnitude of 3.4×10^{-6} pounds. The purpose of the kicker load is to force the ring into a buckled mode shape without significantly affecting the magnitude of the deflections.

As there was no exact solution other than the known collapse load of 60 psi, the tolerance chosen is 1.001. This is the tolerance that yielded two significant digits accuracy for the bar-spring problem. The results obtained are shown in Table III.

As can be seen from the results in Table III, the reduction in iterations realized by using the linear scalar extrapolation is excellent. In fact, for the 59.5 psi load, the unaltered version did not satisfy the

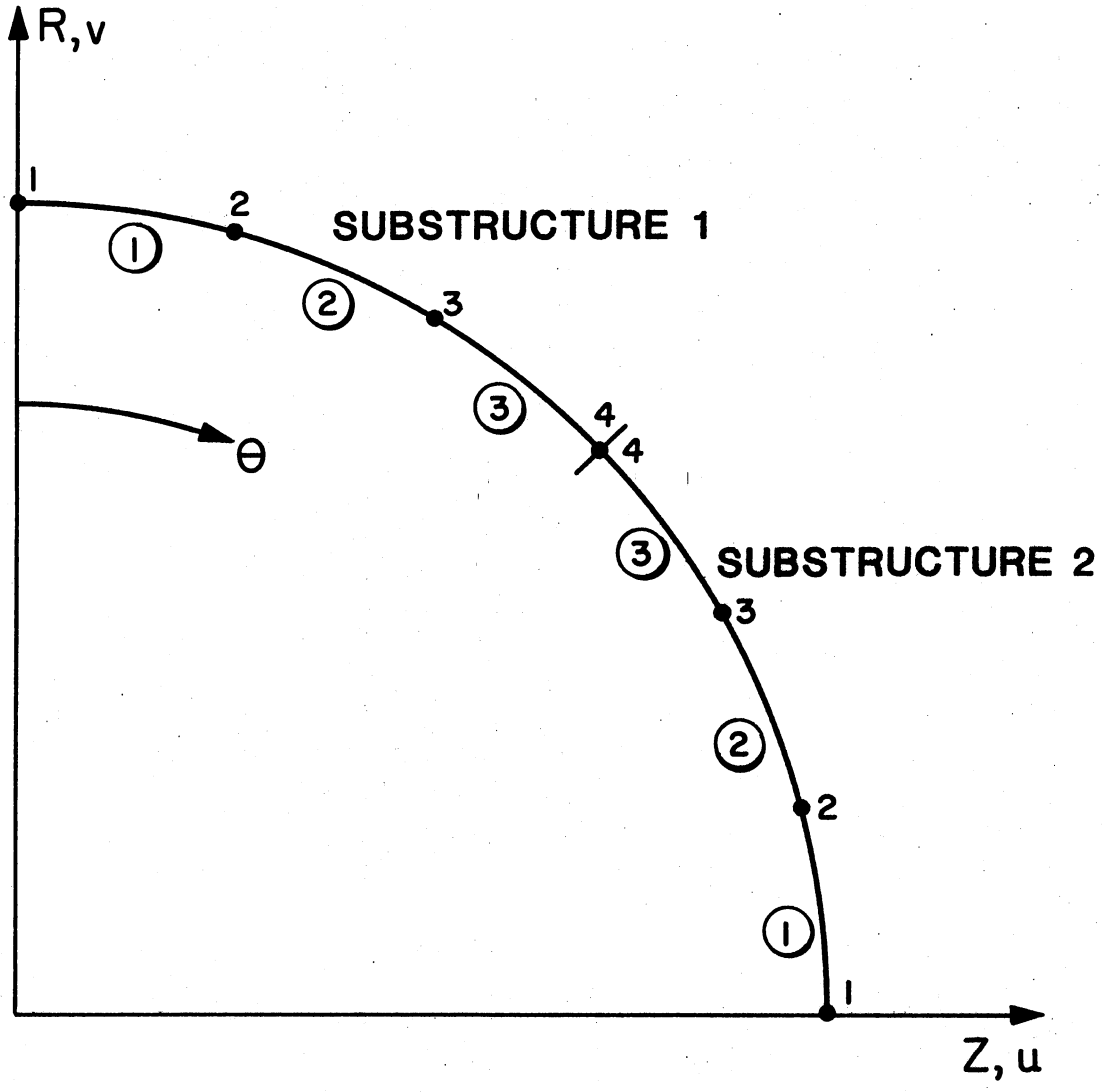

Figure 17. Finite Element Model of a Ring

TABLE III

RING PROBLEM (1.001)

Load (psi)	Method	Iterations	Substructure 1 (in.) Node 1 D.O.F. R ($\times 10^6$ in.)	Substructure 2 (in.) Node 1 D.O.F. Z
7	Old	2	-309.24	-162.98
	New	4	-309.27	-162.98
21	Old	5	-1005.40	-412.13
	New	4	-1005.70	-411.79
35	Old	9	-1946.40	-417.61
	New	4	-1947.70	-416.36
49	Old	23	-4027.60	715.44
	New	3	-4030.40	718.24
56	Old	54	-8839.00	5050.00
	New	3	-8896.30	5107.10
59.5	Old	149*	-34785.00	30700.00
	New	4	-37665.00	33570.00

*Maximum number of iterations allowed. Convergence not yet satisfied.

convergence criteria after 150 iterations. Probably, many more iterations would have been required to satisfy convergence. Clearly, the linear scalar extrapolation is very powerful under such circumstances.

Multiple Search Extrapolation

As explained in Chapter III and illustrated in Figure 12, there are conditions under which the convergence criteria cannot be satisfied by use of the linear scalar extrapolation. Thus the need exists for a multiple search extrapolation method. Ideally, the multiple search extrapolation will locate the closest point of approach to an exact solution along a line search, change the direction of search, and then locate a new closest point of approach. Eventually, the convergence criteria should be satisfied. However, it is important that the residuals be calculated in as accurate a manner as possible since the new search direction is highly sensitive to variations in the residuals.

Pressure Loaded Ring

The first test for the multiple search extrapolation became necessary because of a possible need to decrease the convergence criteria of the earlier mentioned pressure loaded ring to a tolerance of 1.0001. As shown in Figure 12, this has the result of withdrawing the sphere of convergence to a position where the line passing through $\bar{d}$ does not intersect the convergence region. It is not possible to satisfy the convergence criteria without selecting a new search direction and proceeding as before. The ideal situation is depicted in Figure 13.

The solution procedure used earlier in the linear scalar extrapolation increased the load to 59.5 psi gradually by passing through load

values of 7 psi, 14 psi, 21 psi, 35 psi, 49 psi, 56 psi, and finally 59.5 psi. Due to the unavailability of an exact solution for this problem, it was decided that the above mentioned solution would be used as a "Base" value against which to compare the linear scalar extrapolation and the multiple search extrapolation. In order to provide a stringent test for these two methods, the series of load values was selected so as to pass through 7 psi, 14 psi, and 59.5 psi. The step sizes were chosen so as to increase the effect of the nonlinearity as much as possible without introducing plasticity effects. The results for the final displacements at 59.5 psi are shown in Table IV.

As can be seen from the results in Table IV, the multiple search extrapolation was slightly better than the linear scalar extrapolation at every degree of freedom shown. The rotations are not shown in the interest of brevity. It was apparent from the entire solution history output that an improvement toward the correct solution was made after the selection of the second search direction but no significant gain was made thereafter in any of the following three search directions. In fact, from an inspection of the residuals it was apparent that a source of error was obscuring the true residuals from the multiple search extrapolation procedure to such an extent that superfluous directions were being determined and then searched with the net result of zero progress after the second direction.

One reason for this difficulty may be seen by inspecting Equation (2.7). The next to last term on the right-hand side is a function of both the intermediate and current configurations and the stiffness matrix is a function of the current configuration as is the fifth term on the right-hand side. Until this observation, the stress was only updated at

TABLE IV

PRESSURE LOADED RING (59.5 PSI)

Substructure and Node	Method	Displacements ($\times 10^6$ in.) DOF R	Displacements ($\times 10^6$ in.) DOF Z	Percent Closer DOF R	Percent Closer DOF Z
1,1	Base	-37665.0	0.0	0.18	0.00
	Multiple Search	-24643.0	0.0		
	Linear	-24619.0	0.0		
1,2	Base	-34061.0	104.96	0.20	0.15
	Multiple Search	-22311.0	-117.15		
	Linear	-22288.0	-117.49		
1,3	Base	-24870.0	3461.50	0.20	0.17
	Multiple Search	-16375.0	1853.20		
	Linear	-16358.0	1850.40		
1,4	Base	-13999.0	11180.00	0.19	0.19
	Multiple Search	-9355.2	6609.70		
	Linear	-9346.2	6600.90		
2,3	Base	-5443.5	21384.0	0.19	0.19
	Multiple Search	-3795.4	12950.0		
	Linear	-3792.3	12934.0		
2,2	Base	-1130.3	30128.0	0.19	0.19
	Multiple Search	-901.2	18390.0		
	Linear	-900.8	18368.0		
2,1	Base	0.0	33570.0	0.00	0.19
	Multiple Search	0.0	20531.0		
	Linear	0.0	20506.0		

the end of each load step and not at the end of each iteration. Since the multiple search extrapolation is very dependent on correct residuals, a modification was made to update the stresses at the end of each iteration. The pressure ring problem was run once more and the results are shown in Table V.

Clearly, the results are more accurate overall and the multiple search extrapolation has increased its advantage over the linear scalar extrapolation. However, the advantage is still only marginal, which is far short of the expected results. The problem with the residuals mentioned earlier was still apparent in the solution history; therefore, a critical review of Equation (2.7) is necessary.

In Equation (2.7) all terms on the right-hand side are taken to be applied forces even though some came originally from the left-hand side as per Appendices A and B. The key term of interest is the last which came from separating the stiffness matrix into a symmetric matrix and a skew symmetric matrix and placing the skew symmetric part on the right-hand side so as to take advantage of the efficiency of a symmetric equation solver. But, notice that the term $\{\Delta\bar{V}_{est}\}$ is not equal to $\{\Delta\bar{V}\}$ except in linear problems. This is clearly an important source of error in the residuals, albeit the approximation has been very well justified in the past for cost considerations. Consequently, if a problem not as nonlinear as the pressure ring problem were solved, a greater advantage of the multiple search extrapolation over the linear scalar extrapolation is to be expected.

Radially Loaded Ring

The difference between the radially loaded ring and the pressure

TABLE V

PRESSURE LOADED RING (59.5 PSI)

Substructure and Node	Method	Displacements ($\times 10^6$ in.) DOF R↓	Displacements ($\times 10^6$ in.) DOF Z→	Percent Closer DOF R	Percent Closer DOF Z
	Base	-37665.0	0.00		
1,1	Multiple Search	-28315.0	0.00	0.32	0.00
	Linear	-28285.0	0.00		
	Base	-34061.0	104.96		
1,2	Multiple Search	-25618.0	-53.45	0.33	0.28
	Linear	-25590.0	-53.90		
	Base	-24870.0	3461.50		
1,3	Multiple Search	-18754.0	2312.00	0.31	0.31
	Linear	-18735.0	2308.40		
	Base	-13999.0	11180.00		
1,4	Multiple Search	-10648.0	7904.50	0.30	0.33
	Linear	-10638.0	7893.60		
	Base	-5443.5	21384.00		
2,3	Multiple Search	-4251.6	15324.00	0.28	0.33
	Linear	-4248.2	15304.00		
	Base	-1130.3	30128.00		
2,2	Multiple Search	-964.4	21680.00	0.27	0.32
	Linear	-964.0	21653.00		
	Base	0.0	0.00		
2,1	Multiple Search	0.0	24179.00	0.00	0.31
	Linear	0.0	24150.00		

loaded ring is that the applied load for the radially loaded ring never changes direction, and the applied load for the pressure loaded ring is always normal to the surface. The response of the radially loaded ring is slightly nonlinear and buckles at approximately 90 psi while the pressure loaded ring is highly nonlinear and buckles at 60 psi.

The results obtained from the pressure loaded ring are shown in Table VI. The "Base" load steps in psi were 7, 14, 21, 35, 49, 56, 59.5, 63, 66.5, 70, and 73.5. The multiple search extrapolation and linear scalar extrapolation load steps in psi were 7, 14, and 73.5. The tabulated results are for the 73.5 psi condition.

Clearly, the multiple search extrapolation converged closer to the base solution than did the linear scalar extrapolation. The manner in which the residuals behaved in the solution history also indicated that more accurate residuals were being used. Once again reviewing Equation (2.7), it can be seen that the third term from the end is identically zero, the first and second terms from the end have less effect on the solution due to the estimated displacement being very nearly correct.

Plasticity

Ring-Stiffened Cylinder

The only problem investigated which dealt with plasticity effects was a ring-stiffened cylinder with 94 degrees of freedom. Due to a lack of an exact solution against which to compare the results no detailed description can be made. However, due to the manner of convergence of the problem there is little reason to believe that plasticity will present larger difficulties for the extrapolation procedures described in this study under a static load consideration.

TABLE VI

RADIALLY LOADED RING (73.5 PSI)

Substructure and Node	Method	Displacements ($\times 10^6$ in.) DOF R	Displacements ($\times 10^6$ in.) DOF Z	Percent Closer DOF R	Percent Closer DOF Z
1,1	Base	-10075.0	0.00		
	Multiple Search	-10070.0	0.00	16.67	0.00
	Linear	-10069.0	0.00		
1,2	Base	-9216.8	-498.92		
	Multiple Search	-9209.0	-498.45	8.24	0.00
	Linear	-9208.3	-498.45		
1,3	Base	-7029.3	-260.54		
	Multiple Search	-7017.6	-258.48	3.31	0.00
	Linear	-7017.2	-258.48		
1,4	Base	-4387.4	969.26		
	Multiple Search	-4376.8	969.90	1.85	0.00
	Linear	-4376.6	969.90		
2,3	Base	-2159.0	2806.3		
	Multiple Search	-2153.8	2800.0	0.00	7.35
	Linear	-2153.8	2799.5		
2,2	Base	-763.1	4442.1		
	Multiple Search	-762.3	4425.5	0.00	3.49
	Linear	-762.3	4424.9		
2,1	Base	0.0	5093.8		
	Multiple Search	0.0	5072.3	0.00	2.71
	Linear	0.0	5071.7		

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

It is possible to accelerate the rate of convergence to solution for nonlinear structural equations in the pre-buckled region by attempting to extrapolate to a "zero residual" position. In accelerating the rate of convergence, it is also possible to reduce the cost of each iteration through bypassing the factorization of the global stiffness matrix. Furthermore, the total number of iterations can be reduced dramatically.

In the event that the linear scalar extrapolation is not accurate enough to satisfy the convergence criteria, a new search direction orthogonal to the first can be selected by considering the effect of each residual individually on the calculation of the scaling constant, w. By extrapolating along the new direction, a solution closer to the exact solution is possible. However, in order to obtain significantly better results with the new direction of search, the residuals must be calculated in as accurate a manner as possible. This accuracy may require a local increase in the cost of computation, but should be more than offset by the allowance of larger step sizes, a reduction in the number of iterations, and the omission of the factorization of the global stiffness matrix.

There are basically three approaches to reducing the cost of solution of nonlinear structural equations. They are as follows:

1. Reduce the cost of all iterations by placing as many terms on

the right-hand side as possible. (All displacements on the right-hand side are assumed thereby simplifying calculations.)

2. Reduce the number of iterations by use of a linear scalar extrapolation if convergence is possible.

3. Increase the load step size and reduce the average cost per iteration and perhaps the number of iterations by use of the multiple search extrapolation.

Unfortunately, the first and third methods are mutually exclusive but the second can be used to complement either. Which of the above combinations would be most effective is problem dependent and warrants further investigation.

The recommendations for future study are as follows:

1. Complete the modifications necessary to calculate the residuals as accurately as possible so as to determine more fully the advantage available through the multiple search extrapolation.

2. Investigate the possibility of using extrapolations on individual degrees of freedom. This investigation could result in an additional cost savings and might better enable the extrapolation procedures to solve strain-hardening problems (see Appendix D).

3. Complete the investigation of the ability of the extrapolation procedures to solve problems with plasticity effects.

4. Expand all procedures as necessary to reduce the cost of solution of dynamic problems.

BIBLIOGRAPHY

[1] Jones, R. F. Incremental Analysis of Nonlinear Structural Mechanics Problems With Application to General Shell Structures. Naval Ship Research and Development Center Report 4143, 1973.

[2] Haisler, W. E. AGGIE 1--A Finite Element Program for Nonlinear Structural Analysis. Texas A&M University, Aerospace Engineering Department Report, ONR, N00014-76-C-0150, 1977.

[3] Roderick, J. E., R. F. Jones, and M. G. Costello. TRAINS--A Finite Element Computer Program for the Transient, Large Deflection Analysis of Inelastic Structures. Naval Ship Research and Development Center Manual, m 17, Vol. 1, 1978.

[4] Roderick, J. E., R. F. Jones, and M. G. Costello. TRAINS--A Finite Element Computer Program for the Transient, Large Deflection Analysis of Inelastic Structures. Naval Ship Research and Development Center Manual, m 85, Vol. 2, 1978.

[5] Cook, R. D. Concepts and Applications of Finite Element Analysis. New York: John Wiley & Sons, Inc., 1974.

[6] Zienkiewicz, O. C. The Finite Element Method. Berkshire, England: McGraw-Hill Book Co. Ltd., 1977.

[7] Beveridge, G. S. G., and R. S. Schecter. Optimization: Theory and Practice. New York: McGraw-Hill Book Co., 1970.

[8] Tauchert, T. R. Energy Principles in Structural Mechanics. New York: McGraw-Hill, Inc., 1968.

[9] Fung, Y. C. Foundations of Solid Mechanics. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1965.

[10] Marcal, P. V. The Effect of Initial Displacements on Problems of Large Deflection and Stability. Brown University, Division of Engineering, ARPA E54, 1967.

[11] Martin, H. C. "Derivation of Stiffness Matrices for the Analysis of Large Deflection and Stability Problems." Proceedings of the First Conference on Matrix Methods in Structural Mechanics, AFFDL-TR-66-80, pp. 697-715, 1966.

[12] Hibbitt, H. D., P. V. Marcal, and J. R. Rice. A Finite Element Formulation for Problems of Large Strain and Large Displacment. Brown University Report No. 2, ONR N00014-67-A-0191-0007, June, 1969.

[13] Oden, J. T., and J. E. Keys. "Numerical Analysis of Finite Axisymmetric Deformation of Incompressible Elastic Solids of Revolution." International Journal of Solids and Structures, Vol. 6 (1970), pp. 405-518.

[14] Hofmeister, L. D., G. A. Greenbaum, and D. A. Evensen. "Large Strain, Elasto-Plastic Finite Element Analysis." Proceedings of the AIAA/ASME Eleventh Structures, Structural Dynamics, and Material Conference, Denver, Colorado, 1970, pp. 250-259.

[15] Stricklin, J. A., W. E. Haisler, and W. A. Von Riesemann. Geometrically Nonlinear Analysis by the Direct Stiffness Method. Texas A&M University, Aerospace Engineering Department, Report 70-16, 1970.

[16] Oden, J. T. Finite Elements of Nonlinear Continua. New York: McGraw-Hill Book Co., 1972.

[17] Rivello, R. M. Theory and Analysis of Flight Structures. New York: McGraw-Hill, Inc., 1969.

[18] Mendelson, A. Plasticity: Theory and Application. New York: Macmillan Co., 1968.

[19] Bushnell, D. "A Subincremental Strategy for Solving Problems Involving Large Deflections, Plasticity and Creep." International Journal for Numerical Methods in Engineering, Vol. 11 (1977), pp. 683-708.

[20] Anand, S. C. "Constitutive Relations and Solution Schemes in Plastic Analyais." Journal of the Engineering Mechanics Division, ASCE, Vol. 106 (1980), pp. 21-36.

[21] Hibbett, H. D. "Plasticity." Taped lecture series, University of Arizona, 1979.

[22] Thurston, G. A. "Newton's Method Applied to Problems in Nonlinear Mechanics." Journal of Applied Mechanics, Vol. 32, Series E, No. 2 (1965), pp. 383-388.

[23] Mondkar, D. P., and G. H. Powell. "Evaluation of Solution Schemes for Nonlinear Structures." Computers & Structures, Vol. 9 (1978), pp. 223-236.

[24] Almroth, B. O., P. Stern, and F. A. Brogan. "Future Trends in Computerized Structural Analysis." Presented at the Symposium on Future Trends in Computerized Structural Analysis and Synthesis, Washington, D.C., 1978.

[25] Almroth, B. O., and F. A. Brogan. "Automated Choice of Procedures in Computerized Structural Analysis." Computers & Structures, Vol. 7 (1977), pp. 335-342.

[26] Schmidt, W. F. "Adaptive Step Size Selection for Use With the Continuation Method." International Journal for Numerical Methods in Engineering, Vol. 12 (1978), pp. 677-694.

[27] Almroth, B. O., P. Stern, and F. A. Brogan. "Automatic Choice of Global Shape Functions in Structural Analysis." AIAA Journal, Vol. 16 (1978), pp. 35-47.

[28] Schmidt, W. F. "Extending the Convergence Domain of the Newton-Raphson Method in Structural Analysis." Proceedings of the Sixth Canadian Congress of Applied Mechanics, Vancouver, 1977.

[29] Nagy, D. A. "Model Representation of Geometrically Nonlinear Behavior by the Finite Element Method." Presented at the Fourth International Conference on Structural Mechanics in Reactor Technology, San Francisco, California, 1977.

[30] Mallett, R. H., and P. V. Marcal. "Finite Element Analysis of Nonlinear Structures." Proceedings of the American Society of Civil Engineers, Structural Division, 1968, pp. 2081-2105.

[31] Arantes e Olivera, E. R., and E. C. Borges Pires. "A Direct Discrete Method for Buckling and Post-Buckling Analysis of Structures." Theoretical and Applied Mechanics. Ed. by K. T. Koiter, North Holland, 1977.

[32] Wood, R. D., and O. C. Zienkiewicz. "Geometrically Nonlinear Finite Element Analysis of Beams, Frames, Arches and Axisymmetric Shells." Computers & Structures, Vol. 7 (1977), pp. 725-735.

[33] Bushnell, D. Large Deflection Elastic-Plastic Creep Analysis of Axisymmetric Shells. Lockheed Palo Alto Research Laboratory Report, N00014-73-C-0065, 1973, pp. 103-138.

[34] Felippa, C. A. "Procedures for Computer Analysis of Large Nonlinear Structural Systems." Large Engineering Systems. Ed. by A. Wexler. London: Pergamon Press, 1976, pp. 60-101.

[35] Geradin, M., S. Idelsohn, and M. Hagge. "Nonlinear Structural Dynamics Via Newton and Quasi-Newton Methods." To be published in Nuclear Engineering and Design.

[36] Matthies, H., and G. Strang. "The Solution of Nonlinear Finite Element Equations." International Journal for Numerical Methods in Engineering, Vol. 14 (1979), pp. 1613-1626.

[37] Strang, G., and H. Matthies. Numerical Computations in Nonlinear Mechanics. NSF Report, MCS 78-12362, 1978.

[38] Dennis, J. E., and J. J. More. "Quasi-Newton Methods, Motivation, and Theory." SIAM Review, Vol. 19 (1977), pp. 46-89.

[39] Crisfield, M. A. "A Faster Modified Newton-Raphson Iteration." Computer Methods in Applied Mechanics and Engineering, Vol. 20 (1979), pp. 267-278.

[40] Young, C. T., and R. F. Jones. "Study of Solution Procedures for Nonlinear Structural Equations." Presented at the Symposium on Computational Methods in Nonlinear Structural and Solid Mechanics, NASA Conference Publication 2147, 1980.

[41] Pappas, M. "An Improved Direct Search Numerical Optimization Procedure." Computers & Structures, Vol. 11 (1980), pp. 539-557.

APPENDIX A

PRINCIPLE OF VIRTUAL WORK

An excellent reference for the principle of virtual work is Tauchert [8]. Considering the law of conservation of energy, meaning dissipative effects are neglected, it may be stated that the work done by all externally applied forces is equal to the change in strain energy of the structure. This may be written as

$$W_e = U \tag{A.1}$$

which becomes for a single degree of freedom,

$$\frac{1}{2} F_q u_q = \int_v \left(\int_o^{e_{qp}} \sigma_{qp} \, de_{qp} \right) dV \tag{A.2}$$

where

q = a particular degree of freedom;

F = externally applied force at q;

u = displacement at q;

V = volume;

e_{qp} = the qp component of the strain tensor; and

σ_{qp} = the qp component of the stress tensor.

If the structure is in equilibrium and remains in equilibrium through a virtual displacement δu_q, then the force, F_q, remains constant, the stress, σ_{qp}, remains constant, and the actual work done by the externally applied force in moving through the virtual displacement may be written as

$$\delta W_e = \int_s T_q \, \delta u_q \, dA + \int_v \rho_o \, P_q \, \delta u_q dV \tag{A.3}$$

where

T_q = surface traction at q;

A = area;

ρ_o = mass per unit volume; and

P_q = body force at q.

Using the equilibrium expression,

$$T_q = \sigma_{qp}\, n_p \tag{A.4}$$

where n_p is the unit normal in the ρ direction at q, and the divergence theorem,

$$\int_v \bar{\nabla} \cdot \bar{\sigma}\, dV = \int_s \bar{\sigma} \cdot \bar{n}\, d\, s \tag{A.5}$$

where ∇ is the Laplacian operator. Equation (A.3) can be rearranged to show that

$$\delta W_e = \int_v \sigma_{qp}\, \delta e_{qp}\, dV \tag{A.6}$$

or

$$\int_s T_q\, \delta u_q\, dA + \int_v \rho_o\, P_q\, \delta u_q\, dV = \int_v \sigma_{qp}\, \delta e_{qp}\, dV \tag{A.7}$$

where Equation (A.7) may be taken to be written in index notation with application of the rule of summation on repeated subscripts.

Tauchert [8] has restated Equation (A.7) in words as:

> If a structure is in equilibrium and remains in equilibrium while it is subject to a virtual distortion, the external virtual work δW_e done by the external forces acting on the structure is equal to the internal work δU done by the internal stresses.

This is, of course, a variational calculus expression of the law of conservation of energy as shown in Equation (A.2), the only difference being that in Equation (A.2) the stresses and externally applied forces

are functions of displacement, whereas in Equation (A.7) the stresses and externally applied forces are not functions of the virtual displacements.

APPENDIX B

DEVELOPMENT OF THE STRUCTURAL EQUATIONS

Starting with the virtual work Equation (A.7) expressed in terms of the notation in Figure 6, it can be seen that

$$\int_s (T_q + \Delta_G T_q + \Delta T_q)\ \delta(\Delta u_q) dS + \int_v (\rho_o P_q + \rho_o \Delta P_q)\ \delta(\Delta u_q) dV$$
$$= \int_v (S_{qp} + \Delta S_{qp})\ \delta(\Delta E_{qp}) dV \qquad (B.1)$$

Furthermore, the Lagrangian or Green strain in the current configuration can be written as

$$E_{qp} = \frac{1}{2}(u_q + \Delta u_q)_{,p} + \frac{1}{2}(u_p + \Delta u_p)_{,q}$$
$$+ \frac{1}{2}(u_m + \Delta u_m)_{,i}\ (u_m + \Delta u_m)_{,j} \qquad (B.2)$$

so that the variation in strain, $\delta(\Delta E_{qp})$, due to a variation in incremental displacement, $\delta(\Delta u_q)$, is given by

$$\delta(\Delta E_{qp}) = \delta E_{qp} = \frac{1}{2}\delta(\Delta u_q)_{,p} + \frac{1}{2}\delta(\Delta u_p)_{,q}$$
$$+ \frac{1}{2} u_{m,q}\ \delta(\Delta u_m)_{,p} + \frac{1}{2} u_{m,p}\ \delta(\Delta u_m)_{,q}$$
$$+ \frac{1}{2}(\Delta u_m)_{,q}\ \delta(\Delta u_m)_{,p} + \frac{1}{2}(\Delta u_m)_{,p}\ \delta(\Delta u_m)_{,q} \qquad (B.3)$$

Substituting Equation (B.3) into Equation (B.1), the result is

$$\int_v (S_{qp} + \Delta S_{qp})\left[\frac{1}{2}(\delta(\Delta u_q)_{,p} + \delta(\Delta u_p)_{,q} + u_{m,q}\ \delta(\Delta u_m)_{,p}\right.$$
$$\left. + (\Delta u_m)_{,q}\ \delta(\Delta u_m)_{,p}\right] dV = \int_v (\rho_o P_q + \rho_o \Delta P_q)\ \delta(\Delta u_q) dV$$
$$+ \int_s (T_q + \Delta T_q + \Delta_G T_q)\ \delta(\Delta u_q) dS \qquad (B.4)$$

Equation (2.6) resulted from a rearrangement of Equation (B.4). However, a differeent rearrangement was necessary to put the equation in a form suitable for development into a computer algorithm. This arrangement is as follows:

$$\begin{aligned}
&\int_v \Delta S_{qp} \left[\frac{1}{2}\left(\delta(\Delta u_q)_{,p} + \delta(\Delta u_p)_{,q}\right)\right. \\
&\quad \left. + \left(u_{m,q} + \Delta u_{m,q}\right)\delta(\Delta u_m)_{,p}\right] dV \\
&\quad = \int_v (\rho_o P_q + \rho_o \Delta P_q)\,\delta(\Delta u_q)\,dV \\
&\quad + \int_s (T_q + \Delta T_q + \Delta_G T_q)\,\delta(\Delta u_q)\,dS \\
&\quad - \int_v S_{qp}\left[\frac{1}{2}\left(\delta(\Delta u_q)_{,p} + \delta(\Delta u_p)_{,q}\right)\right. \\
&\quad \left. + \left(u_{m,q} + \Delta u_{m,q}\right)\delta(\Delta u_m)_{,p}\right] dV
\end{aligned} \tag{B.5}$$

Reviewing the definition of these terms, we have

S_{qp} = known state of stress at intermediate configuration;

ΔS_{qp} = unknown;

u_q = known displacement at intermediate configuration;

Δu_q = unknown;

$\rho_o P_q$ = known body force at intermediate configuration;

$\rho_o \Delta P_q$ = known increment in body force;

T_q = known surface force at intermediate configuration;

$\Delta_G T_q$ = unknown; and

ΔT_q = known increment in surface force magnitude.

Since the incremental stress, ΔS_{qp}, is a function of displacement, the constitutive relationship is needed and may be expressed as

$$\Delta S_{qp} = D_{qpab}\,\Delta E_{ab} \tag{B.6}$$

where D_{qpab} is a material tensor, a known function of the current

configuration, and

$$E_{ab} = \frac{1}{2}\,(\Delta u_{a,b} + \Delta u_{b,a}) + \frac{1}{2}\,u_{m,a}\,\Delta u_{m,b} + \frac{1}{2}\,\Delta u_{m,a}\,u_{m,b} + \frac{1}{2}\,\Delta u_{m,a}\,\Delta u_{m,b} \tag{B.7}$$

Substituting Equation (B.7) into Equation (B.6) and then into Equation (B.5), the result is

$$\begin{aligned}
&\int_v D_{qpab}\,[\tfrac{1}{2}\,(\Delta u_{a,b} + \Delta u_{b,a}) + (u_{m,a}\,\Delta u_{m,b} \\
&\quad + \tfrac{1}{2}\,\Delta u_{m,a}\,\Delta u_{m,b})]\,[\tfrac{1}{2}\,(\delta(\Delta u_q)_{,p} + \delta(\Delta u_p)_{,q}) \\
&\quad + (u_{n,q} + \Delta u_{n,q})\,\delta(\Delta u_n)_{,p}] \\
&\quad = \int_v (\rho_o\,P_q + \rho_o\,\Delta P_q)\,\delta(\Delta u_q)\,dV \\
&\quad + \int_s (T_q + \Delta T_q + \Delta_G\,T_q)\,\delta(\Delta u_q)\,dS \\
&\quad - \int_v S_{pq}\,[\tfrac{1}{2}\,(\delta(\Delta u_q)_{,p} + \delta(\Delta u_p)_{,q} \\
&\quad + (u_{m,q} + \Delta u_{m,p})\,\delta(\Delta u_{m,q})]\,dV
\end{aligned} \tag{B.8}$$

Equation (B.8) is now in a form suitable for developing the computer algorithm. Without great explanation it is simply stated that the old form of finite element analysis with basis functions instead of shape functions was used to develop the algorithm which appears as

$$\begin{aligned}
&([A]^T \int_v [B2]^T\,[D]\,[B1]\,dV\,[A])\,\{\Delta\bar{v}\} \\
&\quad = [A]^T \int_v [r]^T\,(\rho_o\,\{P\} + \rho_o\,\{\Delta P\})\,dV \\
&\quad + [A]^T \int_s [r]^T\,(\{T\} + \{\Delta T\} + \{\Delta_G T\})\,dS \\
&\quad - [A]^T \int_v [B2]^T\,\{S\}\,dV
\end{aligned} \tag{B.9}$$

where

[r] = basis function matrix;

$[A]$ = conversion matrix of constants;

$[B2]^T$ = first bracketed term in Equation (B.8);

$[B1]$ = second bracketed term in Equation (B.8); and

$\{\Delta\bar{v}\}$ = vector of generalized displacements of nodes.

All other terms are as previously defined.

The entire left side of Equation (B.9) may also be written as $[K]\{\Delta\bar{v}\}$. However, it can be seen that the left side of the equation is nonsymmetric. Since the efficiency of a symmetric equation solver is very much desired, the stiffness matrix was broken down into a symmetric part and a skew symmetric part as shown,

$$K_{qp} = \frac{1}{2}(K_{qp} + K_{pq}) + \frac{1}{2}(K_{qp} - K_{pq}) \tag{B.10}$$

or

$$[K] = [K]_s + [K]_A \tag{B.11}$$

so that Equation (B.10) can finally be written as

$$[K]_s \{\Delta\bar{v}\} = \{P\} + \{\Delta P\} + \{T\} + \{\Delta T\} + \{\Delta_G T\} - \{E\} - [K]_A \{\Delta\bar{v}_{est}\} \tag{B.12}$$

where

$[K]_s$ = symmetric part of stiffness matrix;

$[E]$ = equilibrium correction term or the last line of Equation (B.9) or (B.8);

$[K]_A$ = skew symmetric part of stiffness matrix; and

$\{\Delta\bar{v}_{est}\}$ = estimated nodal displacements.

All other terms are as previously defined.

Equation (B.12) is the final form in which the structural equations exist before solution. A point of interest is that the right side of the

equation is either known or assumed, and the only unknown or unassumed term in the entire equation is the displacement of the nodes on the left side; therefore, solution and iteration are possible.

APPENDIX C

NONLINEAR ANALYSIS

In the finite element solution of the nonlinear structural equations (see References [5, 6, 16]), the displacement of any point is interpolated between the nodes using shape functions, [N]. For plane problems this may be written as

$$\{u(x,y)\} = [N(x,y)]\ \{u_{node}\} \tag{C.1}$$

or in terms previously used

$$u_q = r_{qp}\ A_{PK}\ \bar{v}_K \tag{C.2}$$

From the plane theory of elasticity (see References [8, 9, 17]), the nonlinear strain is written as

$$\{E\} = \begin{Bmatrix} u_{,x} + \frac{1}{2}\left[(u_{,x})^2 + (V_{,x})^2\right] \\ v_{,y} + \frac{1}{2}\left[(u_{,y})^2 + (v_{,y})^2\right] \\ u_{,y} + v_{,x} + u_{,y} \cdot v_{,x} \end{Bmatrix} \tag{C.3}$$

Substituting Equation (C.1) into Equation (C.3) and simplifying, the following may be obtained,

$$\{E\} = [B]\ \{u_{node}\} \tag{C.4}$$

where [B] is a matrix of derivative terms of the shape function [N] and of the nodal displacements $\{u_{node}\}$.

As can be seen from Equation (C.3), when the nonlinear terms are

used to express the strain tensor, factoring out the nodal displacements still leaves nodal displacements in the expression for the [B] matrix. This, of course, is what makes the [B] matrix a function of displacement.

It is stated, without derivation, that the elemental stiffness matrix may be expressed as

$$[K] = \int_V [B]^T [D] [B] \, dV \qquad (C5.)$$

(see References [5, 6, 16]). The fact that the stiffness matrix is a function of displacement makes the problem nonlinear. The fact that the [B] matrix is a function of displacement makes the problem geometrically nonlinear.

The material nonlinearity is due to the [D] matrix being a function of displacement, displacement history, load, or whatever else the particular program in use may dictate. For a thorough explanation of the constitutive relationships of plastic deformation, see Jones [1], Mendelson [18], Bushnell [19], Anand [20], or Hibbitt [21]. This study dealt only with the nonlinear stiffness matrix; therefore, it made no difference whether the nonlinearity was material or geometric in nature.

APPENDIX D

PARABOLIC EXTRAPOLATION

As can be observed from Table II, the linear scalar extrapolation is not as effective as the unmodified direct iteration procedure in the post-buckled region. The reason for this is as stated earlier. The stiffness is changing too rapidly for a linear extrapolation to be effective. Hence, the motivation for a parabolic extrapolation is established.

In the development of the expression for ω, two different forms of the parabolic equation were used. The first is for a stiffness relaxation (softening) condition and is shown as

$$(x - a) = p(y - b)^2 \tag{D.1}$$

The second is for the strain-hardening case and may be viewed as

$$(y - b) = p(x - a)^2 \tag{D.2}$$

where

y = incremental force;

x = incremental displacement; and

a, b, p = parabolic parameters.

Furthermore, two major assumptions were necessary before the final development of the ω scalars for the cases mentioned. The first is

1. To extend parabolic extrapolation to multiple dimensions, it must be assumed that $\Delta\bar{y}$ to be a direct result of $\Delta\bar{x}$.

Under this assumption, Equations (D.1) and (D.2) may respectively be written as

$$\Delta\bar{y} = \left[\frac{1}{2p\cdot(\bar{y} - \bar{b})}\right] \Delta\bar{x} \tag{D.3}$$

$$\Delta\bar{y} = [2\bar{p}\ (\bar{x} - \bar{a})]\ \Delta\bar{x} \tag{D.4}$$

The second assumption is

2. In the same spirit of the earlier made assumption, it must be assumed that the generalized forces as well as the generalized displacements may be completely uncoupled.

With much algebra and manipulation, the two cases may be summarized as shown in the following two figures (see Figures 18 and 19).

1. Locate the points

$$(x_1, y_1) = (\Delta u_o, F - R_o) \longrightarrow \text{(Linearly Elastic)}$$

$$(x_2, y_2) = (\Delta u_o + d, F - R_1) \longrightarrow \text{(Direct Iteration)}$$

$$(x_3, y_3) = (\Delta u_o + \omega d, F - R_2) \longrightarrow \text{(Linear Extrapolation)}$$

$$*\{\text{Save } \omega\}$$

2. Calculate the Parabolic Parameters by Components

$$b_i = \left\{ \frac{(y_1^2 - y_2^2)(x_1 - x_3) - (x_1 - x_2)(y_1^2 - y_3^2)}{2[(y_1 - y_2)(x_1 - x_3) - (x_1 - x_2)(y_1 - y_3)]} \right\}_i$$

$$p_i = \left\{ \frac{(x_1 - x_3)}{[(y_1 + y_3) - 2b](y_1 - y_3)} \right\}_i$$

3. Calculate the Constants K_1, K_2, K_3

$$K_1 = \frac{1}{[2\bar{p} \cdot (\bar{F} - \bar{R}_1 - \bar{b})]}$$

$$K_2 = \frac{1}{[2\bar{p} \cdot (\bar{F} - \bar{R}_2 - \bar{b})]}$$

$$K_3 = \frac{1}{[2\bar{p} \cdot (\bar{F} - \bar{b})]}$$

4. Calculate the Parabolic Extrapolation Constant, ω^*

$$\omega^* = \frac{[\bar{R}_o + (\omega + 1)\bar{R}_1 + (\omega - 2)R_2] \cdot \bar{d} + \frac{1}{2}[K_1 + (\omega - 1)^2 K_2 + \omega K_3]}{[\bar{R}_o - \bar{R}_2] \cdot \bar{d} + \frac{1}{2}(\omega - 1)K_3}$$

Figure 18. Stiffness Relaxation

1. Locate the Points

$$(\bar{x}_1, \bar{y}_1) = (\Delta\bar{u}_o, \bar{F} - \bar{R}_o) \longrightarrow \text{(Linearly Elastic)}$$

$$(\bar{x}_2, \bar{y}_2) = (\Delta\bar{u}_o + \bar{d}, \bar{F} - \bar{R}_1) \longrightarrow \text{(Direct Iteration)}$$

$$(\bar{x}_3, \bar{y}_3) = (\Delta\bar{u}_o + \omega\bar{d}, \bar{F} - \bar{R}_2) \longrightarrow \text{(Linear Extrapolation)}$$

2. Calculate the Parabolic Parameters by Components

$$a_i = \left\{ \frac{(x_1^2 - x_2^2)(y_1 - y_3) - (y_1 - y_2)(x_1^2 - x_3^2)}{2[(x_1 - x_2)(y_1 - y_3) - (y_1 - y_2)(x_1 - x_3)]} \right\}_i$$

$$p_i = \left\{ \frac{(y_1 - y_3)}{[(x_1 + x_3) - 2a](x_1 - x_3)} \right\}$$

3. Calculate the Constants C_1, C_2, C_3

$$C_1 \equiv -2(\bar{p} \cdot \bar{d})$$

$$C_2 \equiv [(\bar{R}_1 - \bar{R}_o) \cdot \bar{d} - 2\bar{p} \cdot (\Delta\bar{u}_o - \bar{a}) + \bar{p} \cdot \bar{d}]$$

$$C_3 \equiv [(\bar{R}_o + \bar{R}_1) \cdot \bar{d} + 2\bar{p} \cdot (\Delta\bar{u}_o - \bar{a}) + \bar{p} \cdot \bar{d}]$$

4. Calculate the Parabolic Extrapolation Constants, ω^*

$$\omega^* = \frac{-C_2 \pm (C_2^2 - 4\,C_1\,C_3)^{\frac{1}{2}}}{2C_1}$$

5. Choose the Desired (ω^*) by Comparison With (ω)

Figure 19. Strain Hardening

VITA ?

Cline Turner Young II

Candidate for the Degree of

Doctor of Philosophy

Thesis: A METHOD FOR ACCELERATING CONVERGENCE TO THE SOLUTION OF NON-LINEAR STRUCTURAL EQUATIONS BY REDUCING RESIDUAL WORK

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born August 24, 1950, in Duncan, Oklahoma, the son of Mr. and Mrs. R. K. Young.

Education: Graduated from Midwest City High School, Midwest City, Oklahoma, in May, 1968; received the Bachelor of Science degree from the U.S. Merchant Marine Academy, Kings Point, New York, in 1973; received the Master of Science degree from Oklahoma State University, Stillwater, Oklahoma, in 1976; completed the requirements for the Doctor of Philosophy degree at Oklahoma State University in May, 1981.

Professional Experience: Third Mate, Texaco Marine Fleet, 1973-1974; Graduate Teaching Assistant, Oklahoma State University, 1974-1976; Third Assistant Engineer, Texaco Marine Fleet, 1976-1977; Second Mate, Texaco Marine Fleet, 1977-1978; Instructor, Oklahoma State University, 1978-1981.

Professional Organizations: AIAA, ASME.