

NX Nastran

Handbook of Nonlinear Analysis (Solutions 106 and 129)

About this handbook

The NX Nastran Handbook of Nonlinear Analysis is a legacy document which we do not update. Updated information for solutions 106 and 129 is included in the NX Nastran Basic Nonlinear Analysis User's Guide. The information in this document is still applicable, so is provided here.

Note that the NX Nastran Handbook of Nonlinear Analysis does support search and includes bookmarks for navigation inside of a PDF viewer.

Proprietary & Restricted Rights Notice

© 2013 Siemens Product Lifecycle Management Software Inc. All Rights Reserved.

This software and related documentation are proprietary to Siemens Product Lifecycle Management Software Inc. Siemens and the Siemens logo are registered trademarks of Siemens AG. NX is a trademark or registered trademark of Siemens Product Lifecycle Management Software Inc. or its subsidiaries in the United States and in other countries.

NASTRAN is a registered trademark of the National Aeronautics and Space Administration. NX Nastran is an enhanced proprietary version developed and maintained by Siemens Product Lifecycle Management Software Inc.

MSC is a registered trademark of MSC.Software Corporation. MSC.Nastran and MSC.Patran are trademarks of MSC.Software Corporation.

All other trademarks are the property of their respective owners.

TAUCS Copyright and License

TAUCS Version 2.0, November 29, 2001. Copyright (c) 2001, 2002, 2003 by Sivan Toledo, Tel-Aviv University, stoledo@tau.ac.il. All Rights Reserved.

TAUCS License:

Your use or distribution of TAUCS or any derivative code implies that you agree to this License.

THIS MATERIAL IS PROVIDED AS IS, WITH ABSOLUTELY NO WARRANTY EXPRESSED OR IMPLIED. ANY USE IS AT YOUR OWN RISK.

Permission is hereby granted to use or copy this program, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any derivative code must cite the Copyright, this License, the Availability note, and "Used by permission." If this code or any derivative code is accessible from within MATLAB, then typing "help taucs" must cite the Copyright, and "type taucs" must also cite this License and the Availability note. Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. This software is provided to you free of charge.

Availability (TAUCS)

As of version 2.1, we distribute the code in 4 formats: zip and tarred-gzipped (tgz), with or without binaries for external libraries. The bundled external libraries should allow you to build the test programs on Linux, Windows, and MacOS X without installing additional software. We recommend that you download the full distributions, and then perhaps replace the bundled libraries by higher performance ones (e.g., with a BLAS library that is specifically optimized for your machine). If you want to conserve bandwidth and you want to install the required libraries yourself, download the lean distributions. The zip and tgz files are identical, except that on Linux, Unix, and MacOS, unpacking the tgz file ensures that the configure script is marked as executable (unpack with tar zxvpf), otherwise you will have to change its permissions manually.

CONTENTS

Note: Contributors of various sections denoted in italics.

List of Tables and List of Figures

Introduction

1

- 1.1 Overview this Book
- 1.2 Overview of Nonlinear Capabilities
- 1.3 Program Architecture
- 1.4 Rudiments of User Interface
 - 1.4.1 Case Control Section
 - 1.4.2 **Bulk Data Section**
 - 1.4.3 Parameters
 - 1.4.4 Nonlinear Element Output Codes for Plotting
- 1.5 Nonlinear Characteristics and General Recommendations

2

Basic Considerations for Nonlinear Analysis

- 2.1 **Discrete System for a Nonlinear Continuum Model**
- 2.2 **Finite Element Formulation for Equilibrium Equations**
- 2.3 Coordinate Transformations
- 2.4 **Displacement Sets and Reduction of Equations**
- 2.5 **Nonlinear Solution Procedure**

3

Solution Methods

- 3.1 Adaptive Solution Strategies
- 3.2 Newton's Method of Iteration
- 3.3 **Stiffness Update Strategies**
 - 3.3.1 **Update Principles**
 - 3.3.2 Divergence Criteria
 - 3.3.3 Time Expiration Criteria
- 3.4 Line Search Method
 - 3.4.1 Theoretical Basis
 - 3.4.2 Search Criteria
 - 3.4.3 Implementation of Search Procedure
 - 3.4.4 One-Dimensional Example
 - 3.4.5 Other Provisions for Line Search
 - 3.4.6 **Quadratic Interpolation**
- 3.5 **Quasi-Newton Method**
 - 3.5.1 Evolution of Quasi-Newton Method
 - 3.5.2 Criteria for BFGS Update
 - 3.5.3 Implementation of the BFGS Update Strategy
 - 3.5.4 One-Dimensional Example

- 3.5.5 BFGS Updates for Arc-length Methods
- 3.6 Convergence Criteria
 - 3.6.1 Rudimentary Considerations
 - 3.6.2 Convergence Conditions
 - 3.6.3 Error Functions and Weighted Normalization
 - 3.6.4 Implementation
 - 3.6.5 Some Observations
- 3.7 Arc-Length Methods for Post-Buckling Analysis
 - 3.7.1 Basic Theory for Arc-Length Methods
 - 3.7.2 Riks' Method and Its Variations
 - 3.7.3 Crisfield's Method
 - 3.7.4 Adaptive Arc-length Method
 - 3.7.5 Verification Problems
- 4 **Finite Elements in Nonlinear Analysis**
 - 4.1 Overview
 - 4.1.1 Line Elements
 - 4.1.2 Surface Elements
 - 4.1.3 Solid Elements
 - 4.1.4 Other Elements
 - 4.2 Formulation of Isoparametric Finite Elements
 - 4.2.1 Isoparametric Coordinates
 - 4.2.2 Shape Functions
 - 4.2.3 An Example of Element Matrix
 - 4.2.4 Volume Integration of Element Matrices
 - 4.2.5 Element Loads and Equilibrium
 - 4.2.6 Element Coordinates
 - 4.2.7 Stress Data Recovery
 - 4.3 Gap and Friction Element
 - 4.3.1 Introduction
 - 4.3.2 Basic Considerations for Penalty GAP Element
 - 4.3.3 User Interface for GAP Element
 - 4.3.4 Theoretical Basis for Frictional Behavior
 - 4.3.5 Subincremental Algorithm for Stiffness Update and Bisection
 - 4.3.6 Adaptive Adjustment of Penalty Values
 - 4.3.7 Non-adaptive GAP Element with Anisotropic Friction
 - 4.3.8 Verification and Validation: Bouncing Mass Problem
 - 4.3.9 Verification and Validation: Vibration with Coulomb Damping
 - 4.3.10 Hertzian Problem: Contact Between Sphere and Rigid Plane
 - 4.3.11 Dynamic Case: Impact of a Sphere with Rigid Plane
 - 4.3.12 Recommendations
- 5 **Geometric Nonlinearity**
 - 5.1 Overview and User Interface

S.H.Lee

5.2	Updated Element Coordinates	<i>S.H.Lee</i>
5.2.1	Concept of Convective Coordinates	
5.2.2	Updated Coordinates and Net Deformation	
5.2.3	Provisions for Global Operation	
5.3	Follower Forces	<i>S.H.Lee, D.N.Herting</i>
5.3.1	Basic Definition	
5.3.2	Implementation	
5.3.3	Dynamic Follower Forces	
5.3.4	Verification: Elliptic Cylinder Subject to an Internal Pressure	
5.4	Treatment of Large Rotation	<i>D. V. Wallerstein, R. Allahabadi</i>
5.4.1	Gimbal Angle Approach	
5.4.2	Rotation Vector Approach	
5.4.3	Bisection Due to Large Rotations	
5.4.4	Output Interpretation	
5.5	Verification: Large Rotation of a Beam	<i>A. Raiten</i>
5.5.1	Problem Description	
5.5.2	Finite Element Model	
5.5.3	Analysis Procedure	
5.5.4	Analysis Results	
5.6	Canonical Approach to the Geometric Nonlinearity	<i>S.H.Lee</i>
5.6.1	Fundamentals of Continuum Mechanics	
5.6.2	Incremental Formulations for Large Displacements and Strains	
5.6.3	Total Lagrangian vs. Updated Lagrangian in Finite Elements	
5.6.4	Constitutive Relations for Large Deformation	
6	Material Nonlinearity and Constitutive Relations	
6.1	Introduction	<i>S.H.Lee</i>
6.1.1	Overview	
6.1.2	User Interface	
6.2	Plasticity	<i>S.H.Lee</i>
6.2.1	Some Preliminaries	
6.2.2	Yield Criteria	
6.2.3	Yield Function	
6.2.4	Strain Hardening	
6.2.5	Prandtl-Reuss Stress-Strain Relations	
6.2.6	Associated Flow Rule	
6.2.7	Generalized Effective Plastic Strain Increment	
6.2.8	Yield Function Derivatives	
6.2.9	Degenerate Cases: Plane Strain, Plane Stress and Uniaxial Stress	
6.2.10	Solution Algorithm for Elasto-Plastic Material	
6.2.11	Verification and Validation	
6.3	Nonlinear Elasticity	<i>S.H.Lee</i>
6.3.1	Theoretical Basis	
6.3.2	Solution Alogorithm	

6.3.3	Adaptation of Uniaxial Compression Stress-Strain Curve	
6.3.4	Computational Procedure for Bilateral Stress-Strain Relations	
6.4	Creep and Viscoelasticity	<i>S.H.Lee</i>
6.4.1	Introduction	
6.4.2	Effects of Stress and Temperature	
6.4.3	Equilibrium of a Kelvin–Maxwell Model for a Stress Component	
6.4.4	Adaptation of Rheological Model to Multiaxial Stress State	
6.4.5	Coupling of Plasticity	
6.4.6	Implementation	
6.4.7	Verification and Validation	
6.4.8	Nozzle-to-Spherical Shell Attachment as Benchmark Problem	
6.5	Thermo-Elasticity	<i>S.S.Hsieh</i>
6.5.1	Theoretical Basis	
6.5.2	User Interface	
6.5.3	Validation Problem	
6.6	Initial Anisotropy	<i>S.S.Hsieh</i>
6.6.1	Theoretical Basis	
6.6.2	Validation Problems	
7	Nonlinear Static Analysis	
7.1	User Interface	<i>S.H.Lee</i>
7.1.1	Case Control	
7.1.2	Iteration Control: NLPARM Data	
7.1.3	User Interface for Arc-length Methods: NLPCI data	
7.1.4	Iteration Related Output Data	
7.2	Iteration Module and Solution Sequence	<i>S.H.Lee, T.L.Bock</i>
7.2.1	DMAP Interface of NLITER Module	
7.2.2	Input Data Blocks	
7.2.3	Output Data Blocks	
7.2.4	Parameters	
7.2.5	Diagnostic Outputs	
7.3	Loads Overview	<i>D.V.Wallerstein</i>
7.3.1	Concentrated Loads	
7.3.2	Distributed Loads	
7.3.3	Mass Related Loads	
7.3.4	Thermal Loads	
7.3.5	Loads Combination	
7.4	Nonproportional Loading	<i>S.H.Lee</i>
7.4.1	Validation of Nonproportional Loading Case	
7.4.2	Biaxial Loading Case For Neutral Loading	
7.5	Constraints and Enforced Motion	<i>S.S.Hsieh</i>
7.5.1	Boundary Condition	
7.5.2	Multipoint Constraint	
7.5.3	Enforced Motion	

7.5.4	Example	
7.6	Restarts	<i>S.S.Hsieh</i>
7.6.1	Preliminaries	
7.6.2	Restart Procedure	
7.6.3	Example	
7.7	Verification Problem: Analysis of a Pressure Vessel	
8	Direct Time Integration	
8.1	Overview	<i>S.H.Lee</i>
8.2	Three-Point Method (NLTRD Module)	<i>D.N.Herting</i>
8.2.1	Basic Equations	
8.2.2	Nonlinear Iterations	
8.2.3	Stability Analysis	
8.2.4	DMAP Interface of NLTRD Module	
8.3	Two-Point Method (NLTRD2 Module)	<i>S.H.Lee</i>
8.3.1	Newmark Integration	
8.3.2	Adaptation of Newmark's Method	
8.3.3	Start-up Algorithm	
8.3.4	DMAP Interface of NLTRD2 Module	
8.4	Adaptive Time Stepping	<i>S.H.Lee</i>
8.4.1	Introduction	
8.4.2	Automatic Time Stepping Algorithm	
8.4.3	Bisection Algorithm	
8.5	Expedient Iteration Strategies	<i>S.H.Lee</i>
8.6	Verification Problems	<i>S.H.Lee</i>
8.6.1	Elastic-Plastic Rod with an End Mass Subjected to a Step Loading	
8.6.2	Impulsively Loaded Clamped Plate	
8.6.3	Simply Supported Beam with a Restrained Motion	
8.6.4	Bouncing Weight on an Elastic Platform	
8.6.5	Shallow Spherical Cap under a Concentrated Apex Load	
9	Nonlinear Transient Response Analysis	
9.1	User Interface	<i>S.H.Lee</i>
9.1.1	Case Control	
9.1.2	Implicit Integration Control: TSTEPNL Data	
9.1.3	Iteration Related Output Data	
9.2	Restarts	<i>S.S.Hsieh</i>
9.2.1	Restarting from SOL 66 into SOL 99	
9.2.2	Restarting within SOL 99	
9.2.3	Example	
9.3	Dynamic Effects: Mass and Damping	<i>A.Raiten</i>
9.3.1	Mass Modeling	
9.3.2	Mass Matrix	
9.3.3	Damping Modeling	

9.3.4	Damping Matrix	
9.4	Transient Loads and Initial Conditions	<i>D.Herting, S.H.Lee</i>
9.4.1	Transient Loads	
9.4.2	Enforced Motion	
9.4.3	Initial Conditions	
9.5	Diagnostic Outputs	<i>T.L.Bock</i>
9.5.1	NLTRD Module for AUTO or TSTEP Method	
9.5.2	NLTRD2 Module for ADAPT Method	
10	Special Applications	
10.1	Nonlinear Buckling Analysis	
10.1.1	Introduction	
10.1.2	Formulation of Nonlinear Buckling Analysis	
10.1.3	Analysis Procedure	
10.1.4	Verification Problem: Euler Column Buckling	
10.1.5	Example of a Nonlinear Buckling	
10.2	Nonlinear Modal Analysis	
10.2.1	Analysis Procedure	
10.2.2	Bending Vibration of Linear Material Beam	
10.2.3	Bending Vibration of Nonlinear Material Beam	
10.2.4	Plate Vibration	
10.2.5	Beam Bending Vibration with Superelements	
10.3	Static Analysis using SOL 99 or 129	
10.3.1	Input Data for SOL 99 or 129 Static Analysis	
10.3.2	A Z-Shaped Beam Loaded with A Static Force	
10.3.3	Rotation of a Beam with Friction	
11	Example Problems	
11.1	Snap-Through Analysis of a Spherical Diaphragm in SOL 66	
11.1.1	Problem Description	
11.1.2	Finite Element Modeling and Input Data	
11.1.3	Analysis Results	
11.1.4	Solution Progression	
11.1.5	Concluding Remarks	
11.2	Parametric Study on BFGS Updates and Line Search Method in SOL 66	
11.2.1	Introduction	
11.2.2	The Cologne Challenge: a Z-Shaped Cantilever Beam	
11.2.3	Snap-Through and Snap-Back: a Spherical Shell with Backing Plate	
11.2.4	Pre-Buckling Behavior: an Imperfect Spherical Cap	
11.2.5	Concluding Remarks	
11.3	Creep Analysis of a Thick-Walled Pressure Vessel	
11.3.1	Introduction	
11.3.2	Problem Description	
11.3.3	Model Description	

- 11.3.4 Solution Procedure
- 11.3.5 Results and Discussion
- 11.3.6 Input Data Listing of Pressure Vessel
- 11.4 Transient Response of an Impulsively Loaded Cylindrical Panel
 - 11.4.1 Problem Description
 - 11.4.2 Finite Element Model
 - 11.4.3 Analysis Results and Discussion
 - 11.4.4 Input Data Listing for the 8x16 Model
- 11.5 Contact Analysis for the Shaft-Journal Interface in SOL 66 *R. Louwers*
 - 11.5.1 Introduction
 - 11.5.2 Problem Description
 - 11.5.3 Selection of Gap Stiffness
 - 11.5.4 Superelement and ASET
 - 11.5.5 Analysis Procedure
 - 11.5.6 Summary of Results
 - 11.5.7 Friction in the GAP
- 11.6 Impact Analysis in SOL 99: Stress Wave Propagation in an Elastic Rod
 - 11.6.1 Problem Description
 - 11.6.2 Theoretical Solution
 - 11.6.3 Software Solution
 - 11.6.4 Comparison of Results
 - 11.6.5 Input Data Listing
- 11.7 Effects of K6ROT on Geometric Stiffening of the Thin Shell Model
 - 11.7.1 Theoretical Basis for Parameter K6ROT
 - 11.7.2 Difficulties in the Thin Shell Model
 - 11.7.3 Parametric Studies
 - 11.7.4 QUAD4 Model of Rectangular Plate with Line Load
 - 11.7.5 TRIA3 Model of Rectangular Plate with Line Load
 - 11.7.6 QUAD4 Model of Square Plate with Uniform Pressure
 - 11.7.7 QUAD4 Model of Square Plate with Concentrated Load
 - 11.7.8 Observations
- 11.8 Adaptation of Creep Analysis Capability to General Viscoelastic Materials
 - 11.8.1 Introduction
 - 11.8.2 Formulation for Least Square Fit of Rheological Parameters
 - 11.8.3 Computational Process
 - 11.8.4 Analysis Procedure and Results
 - 11.8.5 Remarks

Bibliography

Appendix A. Nonlinear Bulk Data Description

Index

Chapter 1

INTRODUCTION

1.1 Overview of this Book

This book covers nonlinear structural analysis using Solutions 106 and 129, including statics, dynamics, and stability. This book does NOT cover nonlinear statics using Solution 601. For information on Solution 601, see the book on this solution sequence.

1.2 OVERVIEW OF NONLINEAR CAPABILITIES

Nonlinear effects in structures occur mainly due to nonlinear materials and large rotations. Contact problems exhibit nonlinear effects due to changes in boundary conditions. All of these attributes may be represented by nonlinear elements.

The nonlinear element library of NX NASTRAN consists of:

- RODs, CONRODs, and TUBEs for unidirectional truss members;
- BEAMs for axially and laterally deforming line members;
- QUAD4s and TRIA3s for membrane, plate and shell modeling;
- HEXAs, PENTAs, and TETRAs for solid modeling;
- GAPs for contact and friction simulation.

Nonlinear elements may be combined with linear elements for computational efficiency if the nonlinear effects can be localized. Primary operations for nonlinear elements are updating element coordinates and applied loads for large displacements, and the internal relaxation iteration for material nonlinearity. Refer to Table 1.2.1 for a summary of the nonlinear element properties.

The geometric nonlinearity becomes discernible when the structure is subjected to large displacement and rotation. Geometric nonlinear effects are prominent in two different aspects: geometric stiffening due to initial displacements and stresses, and follower forces due to a change in loads as a function of displacements. The large deformation effect resulting in large strains has not been implemented.

Material nonlinearity is an inherent property of any engineering material. Material non-linear effects may be classified into many categories. Included are plasticity, nonlinear elasticity, creep, and viscoelasticity. Creep and viscoelasticity, implemented as a generalized nonlinear viscoelastic capability, may be coupled with plasticity. Many sophisticated options are available for yield criteria and hardening behavior in plasticity.

The primary solution operations are gradual load or time increments, iterations with convergence tests for acceptable equilibrium error, and stiffness matrix updates. The iterative process is based on the modified-Newton's method combined with optional expeditious methods such as the quasi-Newton (BFGS) update and the line search. The stiffness matrix updates are performed occasionally to improve the computational efficiency, and may be overridden at the user's discretion. A number of options of arc-length methods are also available for snap-through or post-buckling analysis of the static problems. For the transient response analysis, a number of options are available for implicit direct time integration, combined with adaptive and expedient iteration strategies similar to those implemented for static analysis.

Solution sequences 106 and 129 consolidate all the nonlinear features described above. SOL 106 is applicable to static, quasi-static, and nonlinear buckling analyses. SOL 129 is primarily applicable to dynamic transient response analysis with some limited static analysis capability. Both of these solution sequences can accommodate superelements

Table 1.2.1 Summary of Properties of the Nonlinear Elements

C o n n e c t i v i t y	P r o p e r t i e s	N o. o f G R I D P o i n t s	N o. o f G a u s s P o i n t s	G e o m e t r i c N o n l i n e a r P e r i o d i c y	Material Properties			Static Loads			DATA Recovery			
					N o n l i n e a r P r o p i e t c	A I s o t r o p i c	O r t h o t r o p i c	T h e r m a l	P r e s s u r e	G r a v i t y	S t r e s s	G r i d P o i n t S t r e s s	F o r c e	S t r u c t u r e P l o t
CBEAM	PBEAM	3	16*	X	1			X		X	X		X	X
	PBCOMP													
CGAP	PGAP	3	0								X			X
CHEXA	PSOLID	8	8	X	1	9		X	X	X	X	X		X
CONROD	-----	2	1	X	1			X		X	X		X	X
CPENTA	PSOLID	6	6	X	1	9		X	X	X	X	X		X
CQUAD4	PSHELL	4	5*	X	1	2	8	X	X	X	X	X	X	X
	PCOMP													
CROD	PROD	2	1	X	1			X		X	X		X	X
CTETRA	PSOLID	4	1	X	1	9		X	X	X	X	X		X
CTRIA3	PSHELL	3	5*	X	1	2	8	X	X	X	X	X	X	X
	PCOMP													
CTUBE	PTUBE	2	1	X	1			X		X	X		X	X

NOTES:

1. Integers listed under Material properties identify (i) on MATi Bulk Data entries.
2. MATS1 and CREEP may be attached to MAT1.
3. Edge nodes are not applicable to nonlinear elements.
4. * may be changed by the user.

1.3 PROGRAM ARCHITECTURE

The software has a modular structure to separate functional capabilities which are organized under an efficient executive system. The program is divided into a series of independent subprograms, called functional modules. A functional module is capable of performing a pre-defined subset of operations. It is the Executive System that identifies every module to execute by MPL (Module Properties List).

The Executive System processes the input data by IFP (Input File Processor) and the general initialization, which are known as Preface operations. It then establishes and controls the sequence of module executions in the OSCAR (Operation Sequence Control Array) based on the user-specified DMAP (Direct Matrix Abstraction Program) or solution sequence. The Executive System allocates system files to the data blocks in the FIAT (File Allocation Table) and maintains a parameter table for module interface. The Executive System is also responsible for the database management and all the input and output operations by GINO (General Input/Output Routines).

The functional module consists of a number of subroutines. Modules communicate with each other only through secondary storage files, called data blocks (matrix or table). Each module performs a certain function with input data blocks and produces output data blocks. A module may communicate with the Executive System and with other modules through parameters, which may be input and/or output variables of the module. Modules utilize main memory dynamically. If the size of the main memory is insufficient to complete an operation, the module uses scratch files, which reside in the secondary storage as an extension of the main memory. This is known as a spill operation.

DMAP is a kind of macro program using a data block oriented language. The solution sequence is a collection of module statements written in the DMAP language tailored to process a sequential series of operations, resulting in a specific type of structural analysis. A typical solution sequence consists of three phases of functional operations: formation, assembly, and reduction of matrices; solution of equations; and data recovery. Solution sequences that process superelements have built-in superelement loops in the first and the last phases.

The nonlinear solution sequences have built-in loops in the second phase for subcase changes, load increments, and stiffness matrix updates. Nested in this DMAP loop, nonlinear solution processes comprise a number of internal iteration loops. Confining the discussion to SOL 106 and SOL 129, the hierarchy of the nonlinear looping is shown in Fig. 1.3.1. Central to the nonlinear processes are modules NLITER and NLTRD2 for SOL 106 and SOL 129, respectively. These modules are self-contained to perform iterations for converged solutions.

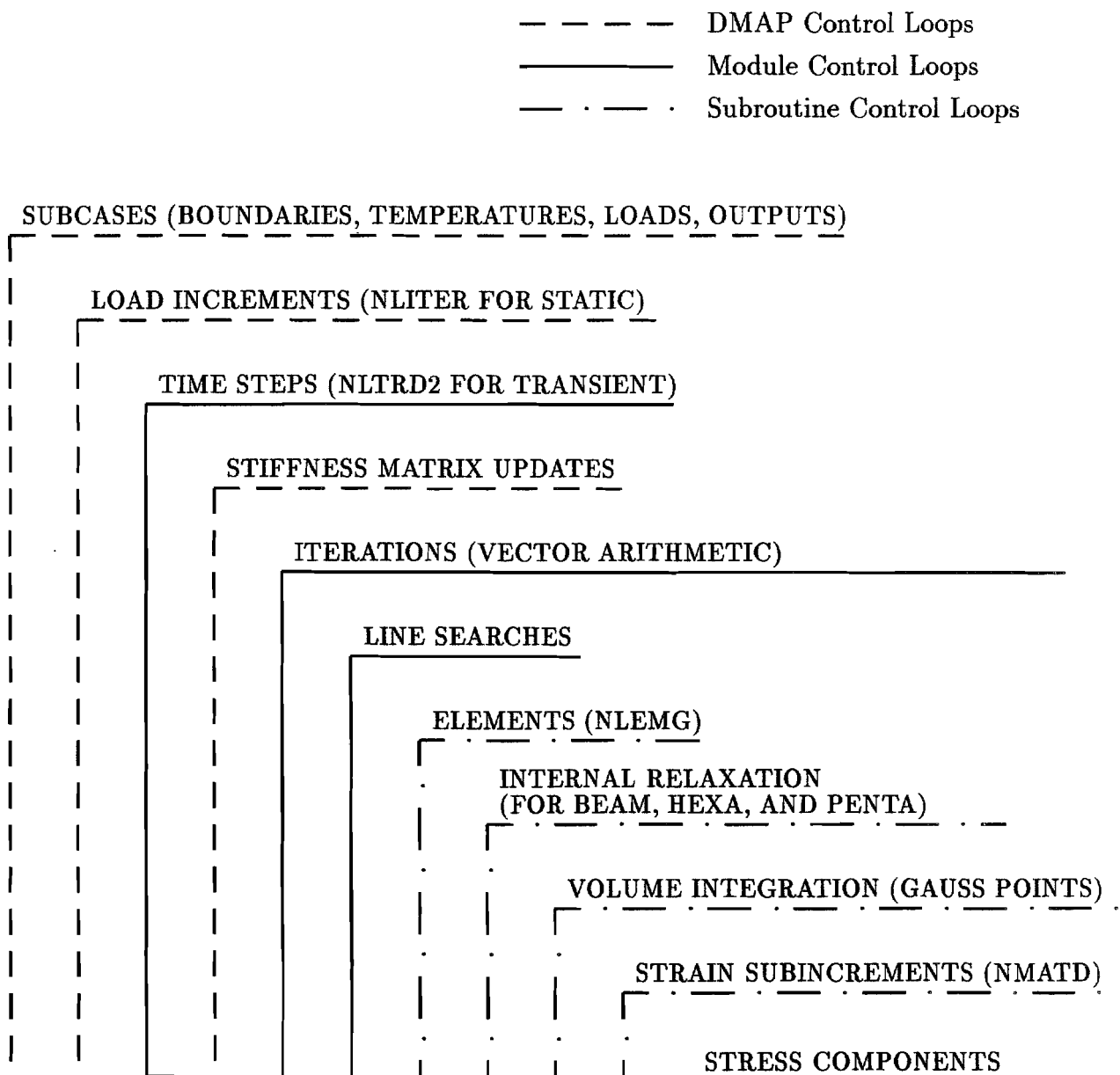


Figure 1.3.1 Hierarchy of Nonlinear Looping

1.4 RUDIMENTS of USER INTERFACE

The input data structure, as was designed originally, still has the most popular format consisting of an optional header, Executive Control Section, Case Control Section, and Bulk Data Section. All the features and principles for the user interface are common in the software. Needless to say, all the features of the user interface for nonlinear analysis are compatible with those for the linear analysis. Review Reference [1.1] for details of the general features. Any exceptions for nonlinear analysis will be explained in the relevant chapters that follow.

Mechanical design is dictated by the strength, dynamic, and stability characteristics of the structure. The software provides the analysis capabilities of these characteristics with solution sequences, each of which is designed for specific applications. The type of desired analysis is specified in the Executive Control Section by using a solution sequence identification. SOL 66 (or 106) is designed for static, quasi-static, and buckling analyses. SOL 99 (or 129) provides nonlinear transient response analysis.

The basic input data required for a finite element analysis may be classified as follows:

- Geometric data
- Element data
- Material data
- Boundary conditions and constraints
- Loads and enforced motions
- Solution methods

The first three classes of data may not be changed during the course of an analysis whereas the last three classes of data may be changed in midcourse via subcases under the Case Control Section. Examples of input data are illustrated in Tables 1.4.1 and 1.4.2 for nonlinear static and dynamic analyses, respectively.

Table 1.4.1 Example Input Data for Nonlinear Static Analysis

```

ID TETRAC,V65      $ CJS 18-JULY-85
SOL 66             $ NONLINEAR STATIC ANALYSIS
DIAG 8,50          $ DIAGNOSTIC PRINTOUT
TIME 8             $ FOR VAX
CEND               $ END OF EXECUTIVE CONTROL DATA
TITLE = TEST OF CTETRA ELEMENT (CUBE SUBJECT TO UNIAXIAL LOADING)
  SEALL = ALL      $ FOR ALL MATRICIES
    DISP = ALL
    STRESS = ALL
    SPC = 100
$ RESTART PARAMETERS
PARAM  SUBID
PARAM  LOADING
PARAM  LOOPID
SUBCASE 1
  SUBTITLE = ELASTIC -- LOAD TO 850. PSI
  LABEL = LOAD TO YIELD
  LOAD = 50
  NLPARM = 50
SUBCASE 2
  SUBTITLE = PLASTIC -- LOAD TO 1000. PSI
  LABEL = LOAD BEYOND YIELD
  LOAD = 100
  NLPARM = 100
SUBCASE 4
  SUBTITLE = ELASTIC -- UNLOAD COMPLETELY TO 0. PSI
  LABEL = FULL UNLOAD
  LOAD = 200
  NLPARM = 200
$OUTPUT(PLOT)      $ PLOT DATA ADDED FOLLOWING THIS LINE IF NEEDED
$ END OF CASE CONTROL DATA
BEGIN BULK
$ PARAMETERS FOR NONLINEAR ITERATION
NLPARM  50      1      AUTO      UPW      NO
NLPARM  100     8      SEMI      UPW      NO
NLPARM  200     2      AUTO      UPW      NO
.
.
.
ENDDATA

```

Table 1.4.2 Example Input Data for Nonlinear Dynamic Analysis

```

ID VGAP, V65      $ SHL
SOL 99            $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50         $ DIAGNOSTIC PRINTOUT
TIME 5            $ FOR VAX
CEND              $ END OF EXECUTIVE CONTROL DATA
TITLE = GAP VERIFICATION BY NONLINEAR TRANSIENT ANALYSIS
  SEALL = ALL     $ FOR ALL MATRICIES
  SET 1 = 201 $ ,202,200
  SET 3 = 22      $ ,23
  SET 4 = 20      $ CONROD
  DISP = 1
  STRESS = 3      $ FOR GAP FORCES
  ELFORCE = 4
  LOADSET = 20
  SPC = 100
$ RESTART PARAMETERS
$PARAM LOOPID
$PARAM STIME
SUBCASE 1          $ UP TO 0.5 SECONDS
  DLOAD = 210
  TSTEPNL = 22
SUBCASE 2          $ UP TO 1 SECOND
  DLOAD = 210
  TSTEPNL = 21
OUTPUT(XYPLOT)
  CSCALE = 1.3
  XTITLE = TIME IN SECONDS
  YTITLE = DISPLACEMENT IN INCHES
  YMIN = 0.
  YMAX = 15.
  XYPLOT DISP RESP/201(T1),202(T1),200(T1)
$ END OF CASE CONTROL DATA
BEGIN BULK
TSTEPNL 21         200      .0025   2      AUTO      10      W
.
.
.
ENDDATA

```

1.4.1 Case Control Section

The Case Control Section may be regarded as a central control panel where select switches for input and output are set. The primary purpose of the Case Control Data is to define subcases. The subcase structure provides a unique means of changing loads, boundary conditions, and solution methods by making selections from the Bulk Data. Confining the discussion to SOL 66 (or 106) and SOL 99 (or 129), loads and solution methods may change from subcase to subcase on an incremental basis. However, constraints can be changed from subcase to subcase only in the static solution sequence. As a result, the subcase structure determines a sequence of loading and constraint paths in a nonlinear analysis. The subcase structure also allows the user to select and change output requests for printout, plot, etc., by specifying set numbers with keywords. Any selections made above the subcase specifications are applicable to all the subcases. Selections made in an individual subcase supersede the selections made above the subcases. Table 1.4.3 summarizes the Case Control Data for nonlinear analysis.

Table 1.4.3 Summary of Nonlinear Case Control Data

<p>Load Selection</p> <p>LOAD CLOAD DLOAD LOADSET NONLINEAR IC</p>	<p>Selects static loading condition Selects static load combination for superelements Selects dynamic loading conditions Selects static load sets defined on the Bulk data LSEQ Selects nonlinear loading (NOLINi) for transient response Selects initial conditions for transient response</p>
<p>Solution Method Selection</p> <p>METHOD NLPARM TSTEPNL</p>	<p>Selects methods for eigenvalue analysis Selects iteration methods for nonlinear static analysis Selects iteration methods for nonlinear transient analysis</p>
<p>Output Requests</p> <p>DISPLACEMENT VELOCITY ACCELERATION ELFORCE STRESS SPCFORCES NLLOAD OUTPUT (PLOT)</p>	<p>Requests output for displacements of physical points Requests output for velocities of physical points Requests output for acceleration of physical points Requests output for element forces Requests output for element stresses Requests output for constraint forces of SPC points Requests output for NOLINi in transient response Requests the beginning of the plotter output</p>
<p>Superelement Control</p> <p>SUPER SEALL SEKR SELG SELR SEMG SEMR</p>	<p>Specifies the superelement identification number and the load sequence number Combines the functions of SEMG, SELG, SEKR, SEMR, and SELR Specifies the superelement identification numbers for which stiffness matrices will be assembled and reduced Specifies the superelement identification numbers for which load vectors will be generated Specifies the superelement identification numbers for which the static load matrices will be assembled and reduced Specifies the superelement identification numbers for which stiffness, mass, and damping matrices will be generated Specifies the superelement identification numbers for which the mass and damping matrices will be assembled and reduced</p>

1.4.2 Bulk Data Section

The primary input media is the Bulk Data, where all details of the input data are specified. Some of the Bulk Data may not be used during a program execution if they are not selected in the Case Control Data. The nonlinear Bulk Data are summarized in Table 1.4.4. All the input data specially designed for nonlinear analysis will be described in the relevant chapters that follow. Descriptions of the Bulk Data pertinent to nonlinear analysis are listed in Appendix A.

Table 1.4.4 Summary of Nonlinear Bulk Data

Element Connectivity CBEAM CGAP CHEXA CONROD CPENTA CQUAD4 CROD CTETRA CTRIA3 CTUBE	Defines connection for beam element Defines connection for gap or frictional element Defines connection for six-sided solid element Defines connection and properties for rod Defines connection for five-sided solid element Defines connection for quadrilateral element with bending and membrane stiffness Defines connection for rod with axial and torsional stiffness Defines connection for four-sided solid element Defines connection for triangular element with bending and membrane stiffness Defines connection for a tube
Element Properties PBCOMP PBEAM PCOMP PGAP PROD PSHELL PSOLID PTUBE	Defines properties for composite CBEAM Defines properties for CBEAM Defines properties for composite material laminate Defines properties for CGAP Defines properties for CROD Defines properties for CTRIA3 and CQUAD4 Defines properties for CHEXA, CPENTA and CTETRA Defines properties for CTUBE

Material Properties CREEP MAT2 MAT8 MAT9 MATS1 TABLES1 TABLEST	Defines creep material properties Defines anisotropic material properties for shell elements Defines orthotropic material properties for shell elements Defines anisotropic material properties for solid elements Defines properties for plastic and nonlinear elastic material Defines a function for stress-dependent material properties Combines many TABLES1 entries for temperature dependent material properties
Constraints SPC SPC1 MPC	Defines single-point constraints and enforced displacements Defines single-point constraints Defines a linear relationship for two or more degrees of freedom
Loads CLOAD FORCEi LSEQ MOMENTi NOLINi PLOAD PLOAD2 PLOAD4 RFORCE TEMP TEMPPi TEMPRB TIC TLOADi	Defines a static load combination for superelement loads Defines concentrated load at grid point Defines static load sets for dynamic analysis Defines moment at a grid point Defines nonlinear transient load Defines pressure loads on QUAD4 and TRIA3 Defines pressure loads on shell elements, QUAD4 and TRIA3 Defines pressure loads on surfaces of HEXA, PENTA, TETRA TRIA3, and QUAD4 elements Defines load due to centrifugal force field Defines temperature at grid points Defines temperature field for surface elements Defines temperature field for line elements Specifies initial values for displacement and velocity Defines loads as a function of time
Solution Methods NLPARM NLPCI TSTEPNL EIGB	Defines iteration methods for nonlinear static analysis Defines arc-length methods for nonlinear static analysis Specifies integration and iteration methods for nonlinear transient analysis Defines eigenvalue extraction method for buckling analysis

1.4.3 Parameters

Parameters constitute an important part of input data in their own right because they maintain communications among the user, DMAP, and modules. Parameters are single-valued input data, used for requesting special features or specifying miscellaneous data. Parameters are initialized in the MPL, which can be overridden by a DMAP initialization. Modules may change the parameter values while the program is running.

There are two different types of parameters: user parameters (V,Y,name in the DMAP) and DMAP (non-user) parameters. For the user parameters, users are authorized to change the initial values by specifying PARAM data in the Bulk Data Section or occasionally in the Case Control Section. Many options are controlled by user-specified parameters. Table 1.4.5 lists the parameters used in nonlinear solution sequences 64, 66, 99, 106 and 129. Key parameters which allow options for nonlinear analyses are listed below:

AUTOSPCR Default = NO in SOL 66 or 106

Provides for automatic single-point constraints for the residual structure.

BETA Default = 0.333333 in SOL 99 or 129

The value of BETA is the transient integration control factor for the Newmark-Beta method with AUTO and TSTEP methods.

BUCKLE Default = -1

If a value of 1 is specified in SOL 66 or 106, nonlinear buckling analysis is activated. This capability should be used in the restart runs.

CNSTRT Default = 1 for SOL 64 only

This parameter is used to control restarts but not recommended.

DBDR.NL Default = 0 for SOLs 66, 99, 106, and 129

By default in SOLs 66, 99, 106, and 129, the database directory is printed upon completion of the nonlinear analysis phase. PARAM, DBDRNL, -1 will suppress this printout.

K6ROT Default = 100.0 for SOLs 66, 99, 106, and 129

If K6ROT > 0.0, then stiffness is added to the normal rotation for QUAD4 and TRIA3 elements. This is an alternative method to suppress the grid point singularities, and is intended primarily for geometric nonlinear analysis. The recommended value is in the range of 10.0 to 1.0E+4.

- LANGLE** Default = 1
 In SOL 66 (or 106) and SOL 99 (or 129), large rotations (using PARAM,LGDISP) are treated by the gimbal angle approach through a default option. A rotation vector approach is implemented in Version 67, and can be selected by PARAM,LANGLE with a value of 2.
- LGDISP** Default = -1
 If LGDISP=1, all the nonlinear elements allowed in SOLs 66 and 99 will be assumed to have large displacement effects (updated element coordinates and follower forces). If LGDISP=-1, no large displacement effects will be considered. If LGDISP=2, follower force effects will be ignored while large displacement is accounted for.
- LOADINC** Default = 1
 Defines the beginning load step of the subcase in which the restart analysis is to be initiated in SOL 66.
- LOOPID** Default = 0
 Defines the loop number for initial conditions in SOL 66 or SOL 99 for restarts.
- MAXLP** Default = 5
 Maximum number of iterations allowed internally for element relaxation and material point subincrement processes in SOL 99 (or 129). This parameter is set to 10 in SOL 66 (or 106), which does not allow user's specification of any other value.
- NDAMP** Default = 0.025 for SOLs 99 and 129 only
 Numerical damping introduced in the two-point integration (ADAPT method in SOL 99 or 129) for numerical stability (implemented in Version 67). The numerical damping increases as the value of NDAMP increases, with zero being no numerical damping. Recommended range is from 0.0 to 0.1. For most cases the value of 0.01 is adequate.
- NLAYERS** Default = 5 (minimum = 1, maximum = 12)
 This parameter defines the number of layers used to integrate through the thickness of isotropic material nonlinear QUAD4 and TRIA3 elements. Set NLAYERS=1 for efficiency if no bending is selected (MID2 = 0 or -1 on all PSHELL data entries). A larger value of NLAYERS will give greater accuracy at the cost of computing time and storage requirements.

NMLOOP	<p>Default = 0</p> <p>In Version 67, nonlinear modal analysis capability is built in SOL 106 with PARAM, NMLOOP. In SOL 106, normal modes can be computed with the updated stiffness if PARAM, NMLOOP, n is specified where n is the desired LOOPID. This option also requires the presence of the METHOD command in the Case Control section and EIGR or EIGRL entry in the Bulk Data section.</p>
OUTPUT	<p>Default = 0</p> <p>In SOL 64 this parameter is used in geometric nonlinear analysis to control restarts in a loop. If it is desired to skip operations in the loop prior to stress recovery, set OUTPUT=-1 (and usually set NODATA=-1).</p>
SDATA	<p>Default = 1</p> <p>In SOL 99 (or 129), SDATA=-1 with the proper value (last value) for LOOPID allows data recovery restarts, skipping nonlinear computation loop.</p>
SKPLOAD	<p>Default = 1</p> <p>This parameter is used to control the calculation of loads, as an efficiency feature. If the loads of the present subcase are the same as in the previous case, set SKPLOAD=-1. The default causes recalculation of the loads, even if they do not change (SOL 64 only).</p>
SKPMTRX	<p>Default = 1</p> <p>This parameter is used to control the matrix reduction and decomposition, an efficiency feature in geometric nonlinear analysis (SOL 64). If the matrix of the present subcase is the same as in the previous case (sometimes called right side iterations), set SKPMTRX=-1. Use of this parameter results in a lower solution cost per iteration, but will usually require more iterations for convergence.</p>
SLOOPID	<p>Default = 0</p> <p>In SOL 99 (or 129) transient analysis, this parameter identifies the initial conditions from a previous SOL 66 (or 106) nonlinear static solution. The printout LOOPID=x is given for each subcase and load increment in SOL 66. Setting SLOOPID=x and providing the static data base will cause SOL 99 to start from the static deformed position.</p>

- SMALLDB** Default = -1 (No longer available from Version 66)
- This parameter is provided in SOL 66 to allow a small database option in connection with the "INTOUT" field of the NLPARM entry. With a value of 1, the database will retain only those data blocks corresponding to the load steps for which an output request is made by the "INTOUT" field in the NLPARM entry. Restarts will be restricted to those starting from these steps.
- SSG3** Default = 0
- This parameter is used to control restarts in a loop for geometric nonlinear analysis (SOL 64). If it is desired to skip operations in the loop prior to FBS (for example, restart after time limit), set SSG3 = -1 (and usually set NODATA = -1).
- STIME** Default = 0.0
- In SOL 99 (or 129) restarts from previous SOL 99 runs, the user provides STIME= t_N , where t_N is the last time step of the subcase to be continued with a new or changed subcase in the new run. Thus the loading and printout will start from t_N as though the original run contained the new subcase data.
- SUBID** Default = 1
- Defines the record number of the case control data (for SEID=0) in which the beginning load step is defined for the restart in SOL 66 (or 106). In SOL 64, SUBID is the subcase identification number (from a previous SOL 64 run) used for an initial guess.
- SUBSKP** Default = 0
- Controls the skipping of subcases which are not desired in geometric nonlinear analysis. SUBSKP can be used on restart to skip completed cases or can be used to delete the static (first) or differential (second) subcase. The number of subcases to be skipped is input, not the subcase ID number. The default discards all previous results and is equivalent to a cold start.
- TABS** Default = 0.0
- This is used to convert units of the temperature input (°F or °C) to the absolute temperature, i.e.,
- PARAM, TABS, 273.16 when Celsius is used
- PARAM, TABS, 459.69 when Fahrenheit is used
- Refer to the Bulk Data entry CREEP for a creep analysis with SOL 66 (or 106). Refer to PARAM,SIGMA for heat transfer analysis.

TESTNEG Default = -1 for SOL 64, -2 for Newton's method in SOL 66, +1 for arc-length method in SOL 66 and SOL 106

Controls SOL 64 and SOL 66 solution sequence when negative terms are encountered on the factor diagonal of matrix decomposition. Negative terms indicate that the differential stiffness has introduced a structural instability. The instability may be real (structural buckling) or mathematical (the current iteration appears unstable, but a stable solution exists).

Value	Result
-1	Stop if negative terms occur.
+1 or 0	Continue if negative terms occur.
-2	If negative terms exist, do not use differential stiffness.
+2	Do not use differential stiffness.

TESTSE Default = 1.0×10^{36}

Used to control iterations within subcases for SOL 64. For subcases 1 and 2, the output is skipped if the strain energy is greater than TESTSE. For subcases 3 onward, the solution iterates within the subcase until the incremental strain energy is less than TESTSE.

TSTATIC Default = -1 (SOL 99 or 129 only)

If TSTATIC=1, a static solution may be obtained in SOL 99 by ignoring inertial and damping forces. This option is available only with the adaptive method, specified by ADAPT in the TSTEPNL.

W3,W4 Default = 0.0

The damping matrix for transient analysis is assembled from the equation

$$[B_{dd}] = [B_{dd}^1] + [B_{dd}^2] + \frac{g}{w3} [K_{dd}^1] + \frac{1}{w4} [K_{dd}^4]$$

The values of W3 and W4 are used for $w3$ and $w4$, respectively. The default values of 0.0 for W3 and W4 cause the associated terms in the damping matrix to be set equal to zero, regardless of the presence of the PARAMeter G or $[K_{dd}^4]$ terms. The units for W3 and W4 are radians per unit time.

Table 1.4.5 PARAMeters used in Nonlinear Solution Sequences

NOTES:

B – Usable in the Bulk Data Section only

E – Usable in either the Bulk Data or Case Control Data Section

PARAMETER NAME	SOLUTION SEQUENCE			DEFAULT	DESCRIPTION
	64	66	99		
ASING	B	E	E	0	Removes singularities by appropriate techniques
AUTOSPC	B	E	E	NO	Specifies automatic single-point-constraints
AUTOSPCR		E	B	NO	Specifies automatic SPC's for residual structure
BAILOUT		E	E	0	Terminates superelement program with singularities
BETA			B	0.33333	Specifies control factor for Newmark-Beta method
BUCKLE		E		-1	Selects nonlinear buckling analysis for restarts
CB1,2			E	1.0	Selects superelement damping matrix coefficients
CK1,2,3		E	E	1.0	Selects superelement stiffness matrix coefficients
CM1,2		E	E	1.0	Selects superelement mass matrix coefficients
CNSTRT	B			1	Controls restarts for constraint change
COUPMASS	B	E	E	-1	Generates coupled rather than lumped mass matrices
CURV	E			-1	Computes stress data in a material coordinate system
CURVPLOT	E			-1	Requests x-y (curve) plots
DBCLEAN	B	B	B	0	Empties database
DBDICT	B	B	B	-1	Prints database dictionary at the end and beginning
DBDRNL	E	E	E	0	Prints database dictionary in the nonlinear loop
DBINIT	B	B	B	0	Initializes database with empty data blocks
DBNAME	B	B	B	DB01	Specifies database name
DBNBLKS	B	B	B	4000	Sets maximum number of GINO blocks
DBSETi	B	B	B	0	Specifies database subsets for storing and fetching
DBSORT	B	B	B	5	Sorts printout of database dictionary
DLOAD	B	B	B	0	Indicates no other changes except for loads
DOPT	E			0	Controls x spacing of curves for CURVPLOT

PARAMETER NAME	SOLUTION SEQUENCE			DEFAULT	DESCRIPTION
	64	66	99		
EPPRT	E	E	E	1.0E-8	Prints singularities of stiffness matrix
EPSBIG			E	1.0E+8	Turning parameter for generalized dynamic reduction
EPSMALC			E	1.0E-8	Turning parameter for generalized dynamic reduction
EPSMALU			E	1.0E-10	Turning parameter for generalized dynamic reduction
EPZERO	B	E	E	1.0E-8	Determines printed singularities compared to EPPRT
ERROR		E	E	-1	Terminates run for superelement errors
EST	B	E	E	2	Requests element measure and volume computation
FACTOR	B	B	B	10,000	Generates SEQUID on SEQGP entry
GPECT	B	E	E	-1	Prints all elements connected to each grid point
GRDPNT	B	E	E	-1	Executes Grid Point Weight Generator
G			E	0.0	Specifies uniform structural damping coefficient
HFREQ			E	1.0E+30	Defines upper limit of frequency range
INRLM			E	0	Requests appending of inertia relief modes
KDIAG	E			1.0	Applies value of a spring constant to all DOF's
K6ROT	E	E	E	100.0	Assigns stiffness to normal rotation of QUAD4 and TRIA3
LANGLE		E	E	1	Selects gimbal angle approach for large rotations if 1 and rotation vector approach if 2
LFREQ			E	0.0	Defines lower limit of frequency range
LGDISP		E	E	-1	Selects large displacement effects
LMODES			E	0	Defines number of lowest modes
LOADINC		E		1	Specifies load increment ID for restarts
LOOPID		E	E	0	Specifies loop ID in the database for restarts
MAXLP			E	5	Maximum number of iterations for internal loop
MAXRATIO	E	E	E	1.0E+5	Determines the singularity of the stiffness matrix
MODEL	B	B	B	0	Stores several models in the database
MPCX	B	B	B	0	Controls MPC and rigid element processing on restarts
NDAMP			E	0.025	Specifies numerical damping in ADAPT method
NEWDYN			E	1	Executes MTRXIN, GKAD, and GKAM modules

PARAMETER NAME	SOLUTION SEQUENCE			DEFAULT	DESCRIPTION
	64	66	99		
NEWSEQ	B	B	B	3	Selects option used in Sequencing Processor module
NINTPTS	E			10	Requests CURV to interpolate over all elements
NLAYERS		B	B	5	Specifies number of integration points through thickness for QUAD4 and TRIA3
NOCOMPS		E		1	Supresses composite stress output
NODATA	B	B	B	0	Indicates no changes in Bulk Data
NOMECH	E			-1	Controls spurious mechanisms
NOTRED			E	0	Controls t-set reduction
OG	E			0	Calculates stress/strain data at grid points for CURV
OLDELM	B			1	Computes element forces using old method
OPTION	E			ABS	Sums structural responses using given convention
OUTOPT	E			0	Sets format for output quantities computed by CURV
OUTPUT	B			0	Jumps to data recovery
PDRMSG			E	1	Suppresses data recovery messages
PLOT	B	E	E	1	Jumps to undeformed plot
PLOTSUP		E	E	0	Makes undeformed plots for superelements
PLTMSG		E	E	1	Suppresses undeformed plot messages
POST			E	1	Stores post-processing data blocks in database
PRGPST	B	E	E	YES	Suppresses singularity printout
PROUT	B	E	E	-1	Suppresses execution and printout from ELTPRT
PRPHIVZ			E	1.0E+37	Prints eigenvector matrix for general dynamic reduction
RESDUAL		E	E	1	Skips to nonlinear operations (Phase II)
RSPECTRA			E	-1	Requests calculation of response spectra
RSPRINT			E	0	Suppresses tabulated values of response spectra
SDATA			E	1	Obtains additional solution set output
SEFINAL		B	B	-1	Indicates no data changes if change SEFINAL
SEMAP		B	B	SEMAP	Determines partitioning of superelements

PARAMETER NAME	SOLUTION SEQUENCE			DEFAULT	DESCRIPTION
	64	66	99		
SEMAPOPT		B	B	42	Places exterior superelement grids on CSUPER
SEMAPPRT		B	B	3	Prints SEMAP table
SENAME		E	E	SENAME	Identifies superelement by name
SEQOUT	B	B	B	0	Controls output options for NEWSEQ
SKPLOAD	E			1	Skips load vector calculation
SKPMTRX	E			1	Skips matrix reduction and decomposition
SLOOPID			B	0	Specifies loop ID from SOL66 database for restart
SMALLDB		E		-1	Selects small database option
SOLID	B	B	B	0	Stores several solutions in same database
SPCGEN	B	E	E	0	Places automatic SPC's in SPC1 format
SSG3	B			0	Skips operations prior to FBS
START	B	B	B	0	Defines number of grid points for NEWSEQ
STIME			E	0.0	Specifies time step for restarts
SUBID	B	E		1	Specifies subcase ID for restarts
SUBSKP	B			0	Skips a number of subcases for restarts
SUPER	B	B	B	0	Deletes ungrouped grid points
TABID			E	2	Controls punch for response spectra
TABS		E		0.0	Converts units of temperature input to absolute
TESTNEG	E	E		-1/-2/1	Tests for negative terms on factor diagonal of matrix
TESTSE	E			1.0E+36	Tests for strain energy for iteration control
TSTATIC			E	0	Selects static solution with TSTATIC=1 in SOL 99 using ADAPT method in TSTEPNL
USETPRT	B	E	E	-1	Selects output type for degrees of freedom
USETSEL	B	E	E	0	Controls sets for row list output option
WTMASS	B	E	E	1.0	Multiplies terms of structural mass matrix by value
W3,W4			E	0.0	Selects frequency for conversion of damping

1.4.4 Nonlinear Element Output Codes for Plotting

Nonlinear elements (which reference nonlinear geometry and/or material) have different codes from linear elements. The codes for linear elements can be found in the User's Manual, Vol. II. The following codes may be used in SOL 66 (or 106) and SOL 99 (or 129).

(All items are stresses unless otherwise denoted)

Element Name (Code)	Word or Component	Item
CBEAM (94)	2	External grid point ID
	3	C (alpha-numeric value)
	4	Long. stress at point C
	5	Equivalent stress
	6	Total strain
	7	Effective plastic strain
	8	Effective creep strain
		(Item codes 3 thru 8 are repeated for points D, E and F. Then the entire record (from 2 through N) is repeated for end B of the element.)
CGAP (86)	2	COMP-X
	3	SHEAR-Y
	4	SHEAR-Z
	5	AXIAL-U
	6	SHEAR-V
	7	SHEAR-W
	8	SLIP-V
	9	SLIP-W
CHEXA (93)	2	Grid/Gauss
	3	External grid ID (0 = center)
	4	STRESS-X
	5	STRESS-Y
	6	STRESS-Z
	7	STRESS-XY
	8	STRESS-YZ
	9	STRESS-ZX

Element Name (Code)	Word or Component	Item
CHEXA (93) (cont.)	10	Equivalent stress
	11	Effective plastic strain
	12	Effective creep strain
	13	STRAIN-X
	14	STRAIN-Y
	15	STRAIN-Z
	16	STRAIN-XY
	17	STRAIN-YZ
	18	STRAIN-ZX
	19-146	Items 3 through 17 repeated for 8 corners
CONROD (92)		Same as CROD
CPENTA (91)		Same as CHEXA but with only 6 corners
CQUAD4 (90)	2	Z1 = Fibre distance 1 (Plane stress only)
	3	STRESS-X (at Z1, if plane stress)
	4	STRESS-Y (at Z1, if plane stress)
	5	STRESS-Z (Plane strain only)
	6	STRESS-XY (at Z1, if plane stress)
	7	Equivalent stress (at Z1, if plane stress)
	8	Plastic strain (at Z1, if plane stress)
	9	Creep strain (at Z1, if plane stress)
	10	STRAIN-X (at Z1, if plane stress)
	11	STRAIN-Y (at Z1, if plane stress)
	12	STRAIN-Z (Plane strain only)
	13	STRAIN-XY (at Z1, if plane stress)
	14-25	Items 2 through 13 repeated for fibre distance Z2 (Plane stress only)

Element Name (Code)	Word or Component	Item
CROD (89)	2	Axial stress
	3	Equivalent stress
	4	Total strain
	5	Effective plastic strain
	6	Effective creep strain
	7	Linear torsional stress
CTETRA (85)		Same as CHEXA except with only 4 corners
CTRIA3 (88)		Same as CQUAD4
CTUBE (87)		Same as CROD

1.5 NONLINEAR CHARACTERISTICS AND GENERAL RECOMMENDATIONS

Modeling for nonlinear analysis is not exempted from the guidelines for good modeling practice pertaining to linear analysis [1.1], which are summarized as follows:

- The analyst should have some insight into the behavior of the structure to be modeled; otherwise, a simple model should be the starting point.
- Substructuring should be considered for the modularity of the model and/or synergism between projects and agencies involved.
- The size of the model should be determined based on the purpose of the analysis, the trade-offs between accuracy and efficiency, and the scheduled deadline.
- Prior contemplation of the geometric modeling will increase efficiency in the long run. Factors to be considered include selection of coordinate systems, symmetric considerations for simplification, and systematic numbering of nodal points and elements for easy classification of locality.
- Discretization should be based on the anticipated stress gradient, i.e., a finer mesh in the area of stress concentrations.
- Element types and the mesh size should be judiciously chosen. For example, avoid highly distorted and/or stretched elements (with high aspect ratio); use TRIA3 and TETRA only for geometric or topological reasons.
- The model should be verified prior to the analysis by some visual means, such as plots and graphic displays.

Nonlinear analysis requires better insight into structural behavior. First of all, the type of nonlinearities involved must be determined. If there is a change in constraints due to contact during loading, the problem may be classified as a boundary nonlinear problem and would require GAP elements or some surface contact algorithm. The material nonlinearity is characterized by material properties. However, the material nonlinear effects may or may not be significant depending on the magnitude and duration of the loading, and occasionally on environmental conditions. The anticipated stress level would be a key to this issue. The geometric nonlinearity is characterized by large rotations which usually cause large displacements. Intuitively, geometric nonlinear effects should be significant if the deformed shape of the structure appears distinctive from the original geometry without amplifying the displacements. There is no distinct limit for large displacements because geometric nonlinear effects are related to the dimensions of the structure and the boundary conditions. The key to this issue is to know where the loading point is in the load-deflection curve of the critical area.

The model is accepted for nonlinear analysis by SOL 66 or SOL 99, if it consists of:

- a parameter LGDISP (for geometric nonlinearity);
- any GAP element (CGAP and PGAP);
- any active nonlinear material data (specified on MATS1 and/or CREEP);
- or any combination thereof.

The model may consist of superelements, but only the residual structure (superelement 0) may consist of nonlinear elements, mixed with any type of linear elements. All the GAP elements are always actively nonlinear if included in the residual structure. However, other potentially nonlinear elements in the residual structure become actively nonlinear only if a parameter, LGDISP, is used and/or if they use the nonlinear material data specified on the MATS1 and/or CREEP data entries.

The nonproportional loading introduces additional nonlinearity to the structural mechanics problem. This nonlinearity is manifested in the path-dependent problems such as plastic deformation, creep deformation, and contact problems with frictional forces. The cyclic loading, which belongs to this category of nonlinearity, makes the problem more complicated with another nonlinearity known as the Baushinger effect. Such a nonlinearity is characterized by a loading path, which can be controlled by multiple subcases in SOLs 66 and 99. When the analysis involves a path-independent nonlinear problem such as a purely geometric nonlinear problem, an incremental process may not be required. For computational efficiency, however, the analysis of such a problem does frequently need incremental solutions.

With these points in mind, additional recommendations are imperative for nonlinear analysis:

- Identify the type of nonlinearity and localize the nonlinear region for computational efficiency. If unsure, perform a linear analysis by SOL 61 (or 24) or SOL 69 (or 27) prior to the nonlinear analysis by SOL 66 or SOL 99, respectively.
- Segregate the linear region by using superelements and/or linear elements if possible. Notice that the potentially nonlinear elements can be used as linear elements.
- The nonlinear region usually requires a finer mesh. Use a finer mesh if severe element distortions or stress concentrations are anticipated.
- Be prepared for restarts with the database properly stored in cases of divergence and changing constraints or loading paths via subcases.
- The subcase structure should be utilized properly to divide the load or time history for conveniences in restarts, data recovery, and database storage control, not to mention changing constraints and loading paths.
- The load or time for the subcase should then be further divided into increments, not to exceed 20 load steps (for SOL 66) or 200 time steps (for SOL 99) in each subcase.

- Many options are available in solution methods to be specified on NLPARM (for SOL 66) or TSTEPNL (for SOL 99) data entries. The defaults should be used on all options before gaining experience.
- Caution should be exercised in specifying GAP properties. In particular, the closed gap stiffness should not exceed the stiffness of the adjacent degree of freedom by 1000 times.
- Normal rotation for QUAD4 and TRIA3 should be restrained by a parameter K6ROT when the geometric nonlinearity is involved.
- Understand the basic theory of plasticity or creep before using these capabilities.
- Caution should be exercised in preparing input data for creep, because they are unit dependent.
- For the transient analysis by SOL 99, some damping is desirable and the massless degree of freedom should be avoided.
- The time step size for a transient response analysis should be carefully determined based on the highest natural frequency of interest because it has significant effects on the efficiency as well as the accuracy.

For any anomalies, refer to errors and limitations listed in the User's Manual Volume II.

Chapter 2

BASIC CONSIDERATIONS FOR NONLINEAR ANALYSIS

2.1 DISCRETE SYSTEM FOR A NONLINEAR CONTINUUM MODEL

Theories in solid mechanics are dictated by three governing relationships [2.1]:

- The state equilibrium that requires

$$\frac{\partial \sigma_{ij}}{\partial x_j} + b_i = 0 \quad (2.1.1)$$

where σ_{ij} are stress tensor components, b_i are body forces, and x_j are space coordinates;

- The constitutive relations represented by stress-strain relations, e.g., for linear elasticity,

$$\sigma_{ij} = D_{ijkl} \varepsilon_{kl} \quad (2.1.2)$$

where ε_{kl} are strain tensor components and D_{ijkl} are elastic constants;

- The compatibility represented by strain-displacement relations, e.g., for a small deformation,

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (2.1.3)$$

where u_i are displacements.

These systems of governing differential equations must be satisfied for every infinitesimal element throughout the domain of the continuum. The complete set of state variables, namely displacements, may be determined by solving these systems of equations supplemented by

boundary conditions, and in dynamic situations by initial conditions as well. For the nonlinear problems, the governing equations should be satisfied throughout the history of load application. The material nonlinearity is manifested in the constitutive relations. The geometric nonlinearity is pronounced in the strain-displacement relations, but it also affects the equilibrium equation by changing applied loads. Changes in constraints affect the boundary conditions, which constitute contact problems.

Most of the known solutions for the solid mechanics problems are based on ideal geometry and linear approximations. However, the real nature is more complicated and inherently nonlinear. The linear system is a very particular case of a general problem. Even the nonlinear solutions that we seek deal with only a small subset of special cases in a general category of nonlinear problems. When the nonlinear system is confronted, no general mathematical solutions exist and superposition no longer applies. The system may even be nonconservative.

The first phase of the structural analysis is the idealization of a physical system into a simpler and more manageable engineering problem. The idealization process involves simplifications of the geometry, boundary and joint conditions, and loading conditions, etc. using engineering intuitions, experimental data, empirical observations, and classical solutions. If the idealized structural system renders a problem that cannot be resorted to a classical method of analysis, further idealization is required, namely discretization, for numerical analysis.

Finite elements represent spatial discretization of a continuum. As such, however, they do not immediately impose nonlinearity. When nonlinearity has to be taken into account for large displacements and/or stresses, a numerical model poses new dimensions to the discretization in addition to the n -dimensional Euclidean space. That is, the discretization is applied to time, load, and material properties by using piecewise linear curves. While discretization allows approximate solutions by numerical methods, it introduces numerous mathematical singularities which may complicate computational processes. Fortunately, the efficiency of modern digital computers makes it feasible to apply complicated computational procedures to the complex systems of engineering problems.

For the discrete system, governing differential equations are converted to algebraic equations. The finite element model represents a structure by an assemblage of finite elements interconnected at nodal points. State variables are the displacements (displacement method or stiffness approach) of the nodal points which carry fictitious forces representing distributed stresses actually acting on the element boundaries. The equilibrium requirements are satisfied at nodal points by the nodal force balance. The material constitutive laws are satisfied at the integration points of the element. The compatibility is ensured by the displacement continuity between elements. It is noted, however, that the compatibility of the non-conforming elements is ensured by a patch test.

2.2 FINITE ELEMENT FORMULATION FOR EQUILIBRIUM EQUATIONS

The variational principle renders the system governing equilibrium equations when applied to a functional Π , representing a total potential of a continuum [2.2-2.5], i.e.,

$$\Pi = U - W \quad (2.2.1)$$

where U is the strain energy of the system and W is the potential energy of the external loads. The equilibrium equations can be obtained by invoking the principle of virtual work or the Ritz method, i.e.,

$$\delta\Pi = 0 \quad \text{or} \quad \frac{\partial\Pi}{\partial\{u\}} = 0 \quad (2.2.2)$$

which implies that the total potential of the system must be stationary with respect to the state variables (displacement) for equilibrium to be ensured. The functional Π is so called because it involves the integral of implicit functions of the state variables, $\{u\}$.

Considering a three-dimensional continuum for a nonlinear problem, the stationarity condition results in

$$\int_V \sigma_{ij} \delta\epsilon_{ij} dV = \int_V b_i \delta u_i dV + \int_S t_i \delta u_i dS + \sum_i p_i \delta u_i \quad (2.2.3)$$

where the dots and δ denote infinitesimal increments and arbitrary variations, respectively. The left-hand side represents variations in the strain energy increment and the right-hand side represents variations in the external work which consists of body forces b_i (such as a gravity load), traction forces t_i at the boundary surface (such as pressure loads), and concentrated forces p_i . Now it remains to determine admissible functions expressing the arguments of the functional Π in terms of state variables $\{u\}$, which are valid throughout the whole region and satisfy the boundary conditions.

The finite element method can be characterized by the following features distinguished from the conventional Ritz methods or the matrix method for frame structures:

- The whole region of the system is divided into numerous subdomains, called finite elements, which have simple geometrical shapes.
- The variational process is limited to each finite element, which aggregates into a whole region when assembled.
- The admissible displacement field within each element, $\{\tilde{u}\}$, can be expressed in terms of nodal displacements using interpolation functions known as shape functions, N , i.e.,

$$\{\tilde{u}\} = [N]\{u\} \quad (2.2.4)$$

where $\{u\}$ is a displacement vector consisting of all nodal points of the element.

The strain displacement relations for the element can then be established in terms of nodal displacements using the shape functions in Eq. (2.2.4), i.e.,

$$\{\dot{\epsilon}\} = [B] \{\dot{u}\} \quad (2.2.5)$$

where

$$\{\dot{\epsilon}\}^T = \langle \dot{\epsilon}_x \quad \dot{\epsilon}_y \quad \dot{\epsilon}_z \quad \dot{\gamma}_{xy} \quad \dot{\gamma}_{yz} \quad \dot{\gamma}_{zx} \rangle$$

and the element matrix $[B]$ consists of derivatives of the shape functions, evaluated at the current deformed geometry. Notice that the geometric linear problem requires that the element matrix be evaluated only at the initial geometry. The software employs an approximate updated Lagrangian approach for geometric nonlinear problems, by which linear strains are computed in the updated element coordinate system in order to eliminate the effects of the rigid body rotation but the equilibrium is established at the final position in the stationary coordinate system. This method does not require reevaluation of the element matrix $[B]$ (constant in the absence of large strains) while the element coordinates are reevaluated continuously.

Equilibrium equations for an element may be obtained by reducing Eq. (2.2.3) after the substitution of Eqs. (2.2.4) and (2.2.5), based on the small deformation theory. Then the element boundary stresses are statically equivalent to the nodal forces which balance the applied external loads, i.e.,

$$\{F\}^e = \{P\}^e \quad (2.2.6)$$

with

$$\{F\}^e = \int_V [B]^T \{\sigma\} dV$$

and

$$\{P\}^e = \int_V [N]^T \{b\} dV + \int_S [N_s]^T \{t\} dS + \{p\},$$

where $[N_s]$ is an appropriate interpolation function for the traction force. Notice that the equilibrium equation for an incremental load may be expressed as

$$\{\dot{F}\} = \int_V [B]^T \{\dot{\sigma}\} dV = \{\dot{P}\} \quad (2.2.7)$$

where $\{\dot{\sigma}\}$ should be components of co-rotational stress which is independent of a rigid body rotation.

The element stiffness matrix can be obtained by substituting the constitutive relations into Eq. (2.2.7), i.e.,

$$\{\dot{\sigma}\} = [D] \{\dot{\epsilon}\} \quad (2.2.8)$$

where

$$\{\sigma\}^T = \langle \sigma_x \quad \sigma_y \quad \sigma_z \quad \tau_{xy} \quad \tau_{yz} \quad \tau_{zx} \rangle$$

and $[D]$ is a material tangent matrix. The nodal forces of an element can then be expressed as

$$\{\dot{F}\}^e = \int_V [B]^T \{\dot{\sigma}\} dV = [K]^e \{\dot{u}\} \quad (2.2.9)$$

where the element stiffness is

$$[K]^e = \int_V [B]^T [D] [B] dV. \quad (2.2.10)$$

Notice that this expression represents an element stiffness due to the material stiffness without geometric nonlinear effects. As will be shown later, an additional stiffness $[K^d]$ due to initial stresses should be included for an incremental process because the initial stresses exist from the second increment.

The equilibrium must be satisfied in the whole region throughout the complete history of load application. Equilibrium equations for the global discrete system are obtained when all the elements are assembled, i.e.,

$$\sum_m \int_V [B]^T \{\sigma\} dV = \sum_m \{P\}^e \quad (2.2.11)$$

where \sum over m denotes a summation over all elements. For the incremental process, the equilibrium equation may be rewritten as

$$\sum_m \int_V [B]^T \{\sigma - \sigma^0\} dV = \{\Delta P\} \quad (2.2.12)$$

with

$$\{\Delta P\} = \sum_m \{P\}^e - \sum_m \int_V [B]^T \{\sigma^0\} dV$$

where $\{\sigma^0\}$ represents an initial stress or the stress state at the preceding load step.

Because of the approximations involved in the interpolation functions, the finite element model provides an approximate solution even if the equilibrium equation (2.2.11) is satisfied exactly. Consequently, the differential equations of equilibrium are not satisfied exactly even for linear problems, but the error decreases as the finite element mesh is refined. This convergence condition is required and ensured by element formulations with regard to the element convergence criteria. The convergence, however, may not be monotonic due to non-conforming elements or reduced integration. By virtue of the non-conforming elements and the reduced integration, the finite element model will have added flexibility and compensate the stiffening effects by the displacement method. With a displacement approach, the finite element model is generally known to produce a stiffer structure than in reality.

2.3 COORDINATE TRANSFORMATIONS

The coordinate transformation is one of the most frequent operations in the finite element method [2.6]. Vectors and matrices defined in a particular coordinate system can be transformed into another system. Coordinate systems involved are:

- € Basic coordinate system: a cartesian coordinate system on which local coordinate systems are defined
- € Local coordinate system: defined by the user in the Bulk Data, which may include special coordinates such as cylindrical and spherical coordinate systems
- € Global coordinate system: a collective coordinate system which comprises all the local coordinate systems specified for output quantities
- € Element coordinate system: a cartesian coordinate system unique to each element
- € Displaced element coordinate system: similar to element coordinate system but defined in the displaced position
- € Material coordinate system: a cartesian coordinate system used to orient anisotropic material properties
- € Modal coordinate system: a generalized coordinate system defined for each eigenmode.

It is noted that the global system is a cartesian coordinate system, although non-cartesian coordinate systems are adopted to orient the local cartesian coordinates for output quantities. In the software, all the displacements and forces, hence the system matrices, such as the stiffness matrix, are expressed in the global coordinates. This implies that all the major computations involved in the analysis are processed in cartesian coordinates. Element and material coordinate systems are defined in the element connectivity description. Now we only have to consider linear transformations between cartesian coordinate systems.

Let us consider a coordinate transformation between the primed and unprimed systems which are right-handed cartesian coordinates. The transformation matrix T consists of direction cosines of unit vectors of the unprimed coordinate system, i.e.,

$$u' = T u \quad (2.3.1)$$

where

$$T = \begin{bmatrix} i_x & j_x & k_x \\ i_y & j_y & k_y \\ i_z & j_z & k_z \end{bmatrix}.$$

Notice that T is an orthogonal matrix and thus

$$[T]^{-1} = [T]^T. \quad (2.3.2)$$

Since the work and energy are invariants with respect to the coordinate transformation, i.e.,

$$P^T u = P'^T u'$$

it follows that

$$P' = T P. \quad (2.3.3)$$

Then the equilibrium equation,

$$K' u' = P'$$

may be expressed in the unprimed coordinate system by

$$K u = P$$

with

$$K = T^T K' T. \quad (2.3.4)$$

It is noted that the modal matrix Φ is used as a transformation matrix for a modal transformation which is not elaborated here.

The forces and displacements are transformed from element to global coordinates and vice versa, i.e.

$$\{u_e\} = T_{be}^T T_{bg} \{u_g\} \quad (2.3.5)$$

$$\{F_g\} = T_{bg}^T T_{be} \{F_e\} \quad (2.3.6)$$

where T_{be} transforms from element to basic coordinates;

T_{bg} transforms from global to basic coordinates.

It is noted that T_{be} is identical for all the nodes of an element but T_{bg} may vary from node to node in the same element. The element stiffness matrix is transformed into global coordinates by

$$K_{gg} = T_{bg}^T T_{be} K_{ee} T_{be}^T T_{bg} \quad (2.3.7)$$

for which the building blocks of T_{be} and T_{bg} are (3x3) matrices formed for each nodal point and have to be assembled for an entire element, e.g., for a three-noded triangular shell element

$$T_{bg} = \begin{bmatrix} T_{bg}^1 & 0 & & & 0 \\ 0 & T_{bg}^1 & & & \\ & & T_{bg}^2 & 0 & \\ & & 0 & T_{bg}^2 & \\ 0 & & & & T_{bg}^3 & 0 \\ & & & & 0 & T_{bg}^3 \end{bmatrix}$$

where the superscript is used to associate each (3x3) matrix with the nodal point and is repeated for the rotational degrees of freedom.

2.4 DISPLACEMENT SETS AND REDUCTION OF SYSTEM EQUATIONS

The equilibrium equations, and thus system matrices, are reduced in size using the displacement set, which is a unique feature of this software. Mutually exclusive subsets of the global displacement set, $\{u_g\}$, are defined as follows:

u_m : degrees of freedom eliminated by multipoint constraints;

u_s : degrees of freedom eliminated by single-point constraints;

u_o : degrees of freedom omitted by static condensation;

u_r : degrees of freedom eliminated by a Bulk Data SUPORT to suppress rigid body motion;

u_l : degrees of freedom which remain for solution after reductions.

For convenience, complementary sets are defined as follows:

$$\begin{aligned} u_n &= u_g - u_m \\ u_f &= u_n - u_s \\ u_a &= u_f - u_o \\ u_l &= u_a - u_r \end{aligned}$$

The subsets are defined by the user with a possible exception in the s -set if (PARAM, AUTOSPC, YES) is used. Notice that the rigid elements are equivalent to the multipoint constraints internally in the program, but they are not selectable in the subcases as for MPCs. Since the set-reduction operations involve many basic modules and DMAP blocks, the mathematics for elimination of constraints and static condensation is reviewed here.

The multipoint constraint equations are formed in the module GP4 as follows:

$$[R_{mg}] \begin{Bmatrix} u_m \\ u_n \end{Bmatrix} = \{0\} \quad (2.4.1)$$

where

$$[R_{mg}] = [R_{mm} \mid R_{mn}] .$$

The module MCE1 partitions $[R_{mg}]$ and solves for a transformation matrix $[G_{mn}]$, i.e.,

$$\{u_m\} = [G_{mn}] \{u_n\} \quad (2.4.2)$$

where

$$[G_{mn}] = -[R_{mm}]^{-1} [R_{mn}] .$$

Then the module MCE2 partitions the global stiffness matrix, $[K_{gg}]$, and reduce it to the n -set, i.e.,

$$\begin{bmatrix} K_{mm} & K_{mn} \\ K_{nm} & K'_{nn} \end{bmatrix} \begin{Bmatrix} u_m \\ u_n \end{Bmatrix} = \begin{Bmatrix} P_m \\ P'_n \end{Bmatrix} + \begin{Bmatrix} Q_m \\ Q'_n \end{Bmatrix} \quad (2.4.3)$$

from which the system is reduced to

$$K_{nn} u_n = P_n + Q_n \quad (2.4.4)$$

where

$$\begin{aligned} K_{nn} &= G_{mn}^T [K_{mm} G_{mn} + K_{mn}] + [K_{mn}^T G_{mn} + K'_{nn}] \\ P_n &= G_{mn}^T P_m + P'_n \end{aligned}$$

and

$$Q_n = G_{mn}^T Q_m + Q'_n.$$

The primes are used in K'_{nn} , P'_n , and Q'_n to distinguish from K_{nn} , P_n , and Q_n , which are resulting matrices after the reduction.

Equations in the n -set can be further reduced by eliminating single-point constraints, i.e.,

$$\begin{bmatrix} K_{ss} & K_{sf} \\ K_{fs} & K_{ff} \end{bmatrix} \begin{Bmatrix} u_s \\ u_f \end{Bmatrix} = \begin{Bmatrix} P_s \\ P'_f \end{Bmatrix} + \begin{Bmatrix} Q_s \\ 0 \end{Bmatrix} \quad (2.4.5)$$

which is reduced to

$$K_{ff} u_f = P_f \quad (2.4.6)$$

where

$$P_f = P'_f - K_{fs} Y_s \quad (\text{by SSG2})$$

with

$$u_s = Y_s.$$

Notice that the effects of constraint forces (Q_s and Q_m) are not visible in Eq. (2.4.6). The single-point constraint forces are recovered by

$$Q_s = -P_s + K_{fs}^T u_f + K_{ss} Y_s. \quad (2.4.7)$$

Further reduction of equations in the f -set is performed by an elimination of the o -set, known as static condensation. The f -set is partitioned by a module UPARTN as follows:

$$\begin{bmatrix} K_{oo} & K_{oa} \\ K_{ao} & K'_{aa} \end{bmatrix} \begin{Bmatrix} u_o \\ u_a \end{Bmatrix} = \begin{Bmatrix} P_o \\ P'_a \end{Bmatrix} \quad (2.4.8)$$

from which

$$\{u_o\} = \{u_o^o\} + [G_{oa}] \{u_a\} \quad (2.4.9)$$

where

$$u_o^o = K_{oo}^{-1} P_o \quad (\text{by SSG3})$$

and

$$G_{oa} = -K_{oo}^{-1} K_{oa} \quad (\text{by DECOMP and FBS}).$$

Then the reduced system of equations in the a -set is obtained as

$$K_{aa} u_a = P_a \quad (2.4.10)$$

where

$$K_{aa} = K'_{aa} + K_{oa}^T G_{oa}$$

and

$$P_a = P'_a + G_{oa}^T P_o .$$

There are some rules to remember regarding the displacement sets in nonlinear analysis. They are:

- The r -set is not supported in SOLs 66 (or 106) and 99 (or 129). Do not use a Bulk Data SUPPORT.
- PARAM, AUTOSPC is applicable only to the superelements.
- PARAM, AUTOSPCR is available in SOL 66 (or 106) and applicable to the o -set in the residual structure. However, the o -set may only be used in the linear part of the structure.
- In SOL 99 (or 129), MPCs and SPCs can not be changed in the subcase level. Therefore, all the constraints should be specified above subcases.
- In SOL 66 (or 106), SPCs and MPCs can be changed from subcase to subcase, but they will be changed on an incremental basis.
- Rigid elements are formulated with linear multipoint constraint equations and do not have large displacement capability. Consequently, erroneous results will be obtained if the rigid element undergoes a large rotation. To avoid this, stiff elements should be used in place of rigid elements for large displacement analysis.

2.5 NONLINEAR SOLUTION PROCEDURE

The general-purpose program developer faces the task of providing the best workable solution method for a wide spectrum of problems, while maintaining flexibility by allowing the user to specify optional parameters. Based on the extensive numerical experiments, an attempt was made to establish a general strategy suitable for most problems without requiring insight or experience. Variations in combining theories, algorithms, criteria and parameter values with numerous test problems resulted in a succinct implementation.

The major feature of the nonlinear analysis is the requirement for the incremental and iterative processes to obtain a solution. The main issue is how to choose the most efficient method from the options available for the incremental and iterative processes in the solution of nonlinear equilibrium equations. The increment size for loads or time steps has the most significant effect on the efficiency and the accuracy of the computation, particularly in the path-dependent problems. The incremental and iterative processes are complementary to each other because the larger the increment size the more iterations the solution requires. While an excessively small increment reduces the computing efficiency without any significant improvement in accuracy, a large increment may deteriorate the efficiency as well as the accuracy; it may even cause divergence.

It is impossible to optimize the incremental step size in the absence of prior knowledge of the structural response. The best engineering judgement should be exercised to determine the increment size based on the severity of the nonlinearity. Needless to say, no incremental load steps are required when the response is linear. In principle, the size of the load increment (or time increment for creep analysis) should be so chosen to yield a uniform rate of change in strains or stresses for the material nonlinear problems and a uniform rate of change in displacements for geometric nonlinear problems. Some adaptive methods are available, such as an automatic time step adjustment and bisection of loads upon divergence.

User specifications for solution methods in nonlinear analyses are allowed via:

- Bulk Data NLPARM for the static analysis in SOL 66 or 106 (to be selected by a Case Control Data NLPARM)
- Bulk Data TSTEPNL for the transient response analysis in SOL 99 or 129 (to be selected by a Case Control Data TSTEPNL)
- Bulk Data EIGB for the nonlinear buckling analysis in SOL 66 or 106 (to be selected by a Case Control Data METHOD)

The increment size can be varied from subcase to subcase by specifying different NLPARM or TSTEPNL. It is recommended to define separate NLPARM or TSTEPNL for every subcase even if the same values are specified, so that changes can be accommodated in the subcase level as needed. Determination of the proper incremental time step will be treated in chapter 8, because it is dictated by dynamic characteristics as well as the loading history.

Chapter 3

SOLUTION METHODS

3.1 ADAPTIVE SOLUTION STRATEGIES

Nonlinear finite element computations comprise material processes, element force computations, and various global solution strategies. The computational procedure involves incremental and iterative processes ranging from local subincrements to global solution processes. Performance of the finite element program can be scrutinized from three different perspectives: computational efficiency, solution accuracy and effectiveness. All of these attributes of the nonlinear program can be improved by adaptive algorithms.

There is a broad range of processes for which adaptive algorithms may be adopted in the computational procedure of nonlinear finite element analysis. The size of the load or time increment has the most profound effect on the efficiency as well as accuracy. However, it is difficult to determine optimal load or time increment size. The adaptive algorithm alleviates this difficulty. The most CPU consuming processes in nonlinear analysis are the stiffness matrix update operation and element force calculation. From the efficiency point of view, the number of stiffness matrix updates and the number of iterations should be minimized, which may be conflicting requirements. The adaptive algorithms can help to reduce these processes or to choose a more desirable iteration path. Effectiveness may be interpreted as reliability or user-friendliness. If the program provides a plausible solution or direction despite the poor estimates or selection of control parameters, the engineering time can be reduced considerably. The effectiveness of the program is significantly improved by the adaptive algorithms.

The adaptive algorithm is related to cybernetics or artificial intelligence in a broad sense. In engineering applications, the adaptive system is defined to be an automatic control system which has the capability of automatic measurement of process dynamics and of automatic readjustment (or redesign) of the control algorithm. Favorable performance of the automatic control system is obtained when control parameters are tuned to process dynamics. The control criterion is most critical and tuning the control parameters is the major design task.

The adaptive algorithm is analogous to the automatic control system with the process to be controlled being numerical rather than physical.

Major efforts have been made to implement adaptive solution algorithms for nonlinear analysis. The groundwork was carried out during Version 65 development.

The adaptive capabilities for the transient response analysis were implemented in Version 66, which included the automatic time stepping, adaptive update and bisection, and plasticity induced bisection strategy. Further development of the adaptive capabilities continued for Version 67, which included the load step bisection strategy for static analysis, creep time step bisection, automatic arc-length adjustment, adaptive GAP/friction element, and the adaptive bisection strategies based on the large rotation and material processes. As a result, a cascade of computational feedback control loops have been implemented in the incremental and iterative processes in conjunction with the adaptive subincremental processes in the local level processes. The design objectives are:

- Relieve users from uncertainties (guessing, trial-and-error).
- Maintain accuracy and efficiency without user intervention throughout the analysis.
- Promote effectiveness in terms of engineering time.

Adaptive features implemented in are as follows:

- Newton's iteration for static and implicit dynamic analysis
 - Convergence test: error functions and convergence criteria.
 - Divergence criteria.
 - Adaptive stiffness matrix update strategies.
 - Selective BFGS updates.
 - Selective line search processes.
 - Adaptive bisection and recovery of load increment.
 - Reiteration process.
- Arc-length methods for static post-buckling and snap-through problems
 - Crisfield's arc-length method.
 - Riks and modified Riks methods.
 - Adaptive arc-length adjustment.
 - Selective BFGS Updates.
 - Adaptive switching algorithm for limiting cases.
 - Adaptive correction in case of path reversal.
- Direct time integration for transient response analysis
 - Two-point Newmark integration.

- Automatic time step adjustment.
- Adaptive bisection and recovery.
- Large rotations for Geometric Nonlinearity
 - Gimbal angle approach with auxiliary angle.
 - Rotation vector approach.
 - Adaptive bisection based on rotations.
- Processes for Material Nonlinearity
 - Internal relaxation iteration for select elements.
 - Adaptive subincremental scheme in plasticity and creep routines.
 - Adaptive bisections in plasticity.
 - Adaptive bisection on creep time increment.
 - Error control and stiffness updates for unloading.
- Processes in Gap/Friction Element for contact problems
 - Adaptive subincremental process.
 - Adaptive stiffness matrix update strategy.
 - Adaptive bisection strategy.
 - Adaptive penalty value adjustment.
 - Adaptive friction to simulate stick-slip forces.

The main goal of the self-adaptive method is to relieve users from the chores and uncertainties of guessing or trial-and-error searches. By achieving this goal, an effective algorithm design is automatically attained for efficiency as well as for solution accuracy. The primary virtue of the self-adaptive method, however, is not the computational efficiency but the overall engineering efficiency. This is because the adaptability makes it possible to produce a solution at the very first trial without requiring advanced knowledge of structural behavior. As a result, the goal of improving user-friendliness, solution efficiency and the accuracy has been accomplished.

It is desirable to have a fully automated program which efficiently renders accurate solutions to nonlinear problems. Adaptive algorithms in nonlinear analysis have shone some light toward this goal. The adaptive algorithms in the lower level processes (element or material) in concert with adaptive solution methods in global incremental/iterative processes, make it possible to maintain an acceptable accuracy while maintaining the solution efficiency without user intervention throughout the dynamic environment of nonlinear computation. Although the optimal combination of all parameters is yet to be found, the success of the adaptive algorithms in various levels presents the future direction of nonlinear capabilities development.

3.2 NEWTON'S METHOD OF ITERATION

The equilibrium equations in the g -set may be written as

$$\{P_g\} + \{Q_g\} - \{F_g\} = \{0\} \quad (3.2.1)$$

where $\{P_g\}$, $\{Q_g\}$, and $\{F_g\}$ represent vectors of applied loads, constraint forces, and element nodal forces, respectively. Element nodal forces are nonlinear functions of displacements for nonlinear elements. Since the equilibrium condition is not immediately attained in the presence of nonlinear elements, an iterative scheme such as the Newton-Raphson method is required. Since the error vanishes at constrained points and the constraint forces vanish at free points, the unbalanced forces acting at nodal points at any iteration step are conveniently defined as an error vector by

$$\{R_a\} = \{P_a\} - \{F_a\}. \quad (3.2.2)$$

Notice that the a -set is equivalent to the l -set in the nonlinear analysis because the r -set does not exist. The subscript a will be dropped for simplicity in the following discussion.

Based on Newton's method [3.1, 3.2], a linearized system of equations is solved for incremental displacements by Gaussian elimination in succession. The Jacobian of the error vector emerges as the tangential stiffness matrix. The equation to solve at the i -th iteration is

$$[K_T] \{\Delta u^i\} = \{R^{i-1}\} \quad (3.2.3)$$

where

$$[K_T] = - \left[\frac{\partial R}{\partial u} \right]_{u=u^i} = \left[\frac{\partial F}{\partial u} \right]_{u=u^i}$$

$$\{\Delta u^i\} = \{u^i\} - \{u^{i-1}\}$$

and

$$\{R^i\} = \{P\} - \{F(u^i)\}.$$

The iteration continues until the residual error $\{R\}$ and the incremental displacements $\{\Delta u\}$ become negligible, which is signified by the convergence criteria.

The tangential stiffness consists of the geometric stiffness in addition to the material stiffness, i.e., without regard to the coordinate transformation,

$$[K_T] = \frac{\partial}{\partial u} \int_V [B]^T \{\sigma\} dV = [K^m + K^d] \quad (3.2.4)$$

where $[K^m]$ and $[K^d]$ refer to the material and the differential stiffnesses, respectively. The material stiffness is given in Eq. (2.2.10) with a material tangential matrix for $[D]$. The differential stiffness, which is caused by the initial stress, is defined as follows:

$$[K^d] = \int_V \left[\frac{\partial B_N}{\partial u} \right]^T \{\sigma\} dV = \int_V [G]^T [M] [G] dV \quad (3.2.5)$$

where $[B_N]$ represents the second order effects in the strain-displacement relations, $[G]$ consists of derivatives of shape functions and $[M]$ is a function of stresses. Notice that the initial displacement stiffness is not included in $[K_T]$ because its effects are already eliminated in the element formulation.

Newton's procedure is implemented using a corrective force in MSC/NASTRAN until Version 65. Recalling that the element forces for linear elements are expressed as

$$\{F\} = [K]\{u\},$$

a corrective force vector may be defined as

$$\{C\} = \{F\} - [K]\{u\}. \quad (3.2.6)$$

The iteration starts with initial values

$$\begin{aligned} \{u^0\} &= \text{last converged displacement;} \\ \{R^0\} &= P - F(u^0) = \{\Delta P\} + \{R_e\}; \end{aligned}$$

where $\{\Delta P\}$ is an incremental load vector and $\{R_e\}$ is a residual load error carried over from the last converged solution. Then the successive error vectors can be evaluated by

$$R^{i+1} = R^i - K(u^{i+1} - u^i) - (C^{i+1} - C^i). \quad (3.2.7)$$

It is noted that the corrective force vector vanishes for linear elements. The corrective force calculation was removed in Version 66 and the error vector is computed directly from the internal forces, i.e.,

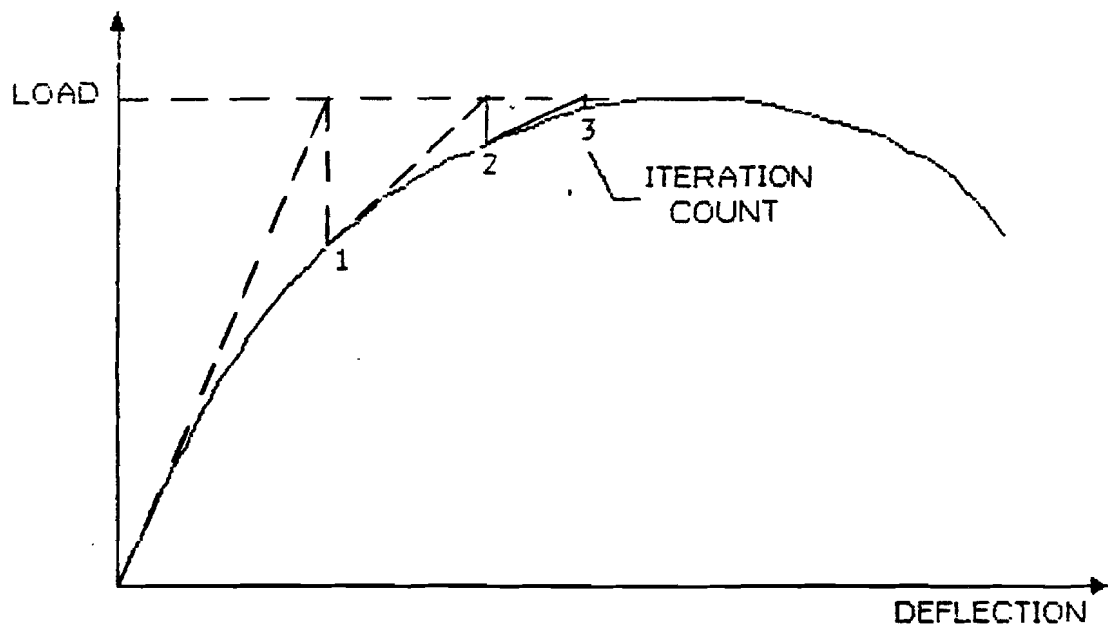
$$\{R^{i+1}\} = \{P\} - \{F^i\} \quad (3.2.7a)$$

Then the residual load error is automatically carried over to the next incremental process.

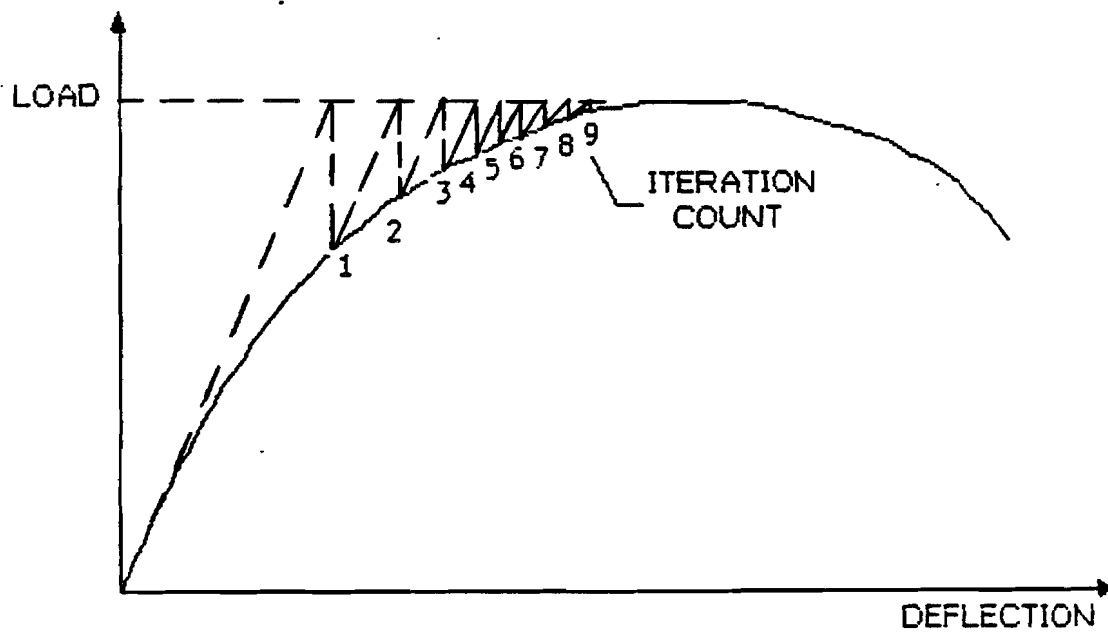
The merit of the Newton-Raphson method is the quadratic rate of convergence, i.e.,

$$\|u^* - u^{i+1}\| \leq q \|u^* - u^i\|^2 \quad (3.2.8)$$

where u^* is a true value of $\{u\}$, q is a constant, and $\| \cdot \|$ represents a vector norm. From a practical standpoint, however, determination of the tangential stiffness and its inverse at each iteration entails a considerable amount of computation. As Fig. (3.2.1) suggests, one may resort to the modified Newton's method which requires the tangential stiffness to be evaluated just once at the initial position, $\{u^0\}$, and used thereafter to solve for $\{\Delta u^i\}$. However, more iterations are required for a given accuracy by the modified Newton's method. The Gaussian elimination method is better suited for this approach than the iterative descent method because the decomposition is performed only once.



(a) Newton-Raphston Method



(b) Modified Newton's Method

Figure 3.2.1 Newton's Methods for Iteration

3.3 STIFFNESS UPDATE STRATEGIES

Among other features of the solution algorithm, the stiffness matrix update has probably the most profound effect on the success of the nonlinear solution. In spite of its significance, however, it is very difficult to implement a robust algorithm for update strategy due to the lack of *a priori* information regarding the right timing for an update. In this section, stiffness update strategies are reviewed with respect to the static analysis.

Variations of the modified Newton's method are adopted in the software. A variation of the modified Newton's methods is to update the stiffness matrix at every few iterations. However, the modified Newton's method could lead to divergence when the stiffness changes drastically, as demonstrated in Fig. 3.3.1, unless the tangential stiffness is reevaluated at the critical point. To this end, an adaptive matrix update method is indispensable. Stiffness update strategies are established to update the stiffness matrix on an as needed basis [3.3] such as probable divergence.

3.3.1 Update Principles

Newton's method could be trapped in an infinite loop, oscillating about the local maximum as illustrated in Fig. 3.3.2(a). This difficulty is overcome during the Newton's iteration by discarding the differential stiffness, $[K^d]$, when the tangential stiffness is not positive definite as shown in Fig. 3.3.2(b). However, the non-positive definite stiffness matrix will be retained and used for iteration if the arc-length method is used for the static solution. This problem is controlled by PARAM,TESTNEG.

From the user's point of view, the stiffness update process is primarily controlled by the input data in KMETHOD (AUTO, SEMI or ITER) and KSTEP (integer) fields of the NL-PARM Bulk Data entry. The stiffness matrix is updated by a DMAP loop in SOL 66 (or 106) controlled by a non-user parameter NEWK which has a value of -1, 1, or 2. The new stiffness is formed whenever the iteration module NLITER returns a positive value for NEWK.

The value of NEWK will be set to 1 under the following conditions, most of which have been determined empirically:

- The ITER method is specified for KMETHOD and the iteration count (i) is an integer multiple of KSTEP.
- The first iteration is finished with the SEMI method.
- The solution has converged with the AUTO or SEMI method and

$$i_k \geq \text{KSTEP}$$

where i_k is the number of iterations to convergence after the stiffness update.

- The solution has converged with the AUTO or SEMI method and

$$\text{NQNV} > \text{MAX}(\text{MAXQN} - 6, 5)$$

where NQNV is the number of quasi-Newton vectors accumulated and MAXQN (integer parameter to be specified in NLPARM) represents the maximum number of quasi-Newton vectors allowed.

- The KMETHOD is AUTO or SEMI and if $i_k > 2$ and

$$t_k < t_i \times (N_i - 7) \quad (3.3.1)$$

where t_k and t_i are the CPU time required for a K update and an iteration, respectively, and N_i is the expected number of iterations for convergence without a K update. The value of N_i (printed under the heading **ENIC** at every iteration) is estimated by

$$N_i = \frac{\log(\text{EPSP}/E_p^i)}{\log \lambda_i^*} \quad (3.3.2)$$

with E_p^i being the relative error in terms of loads at the i -th iteration, EPSP the specified tolerance for E_p^i and

$$\lambda_i^* = \min(\lambda_i, 0.7 + 0.1\lambda_i, 0.99)$$

where the **convergence rate** λ_i (printed under the heading **LAMBDA** at every iteration) is computed by

$$\lambda_i = \frac{1}{2} \left(\frac{E_p^i}{E_p^{i-1}} + \lambda_{i-1}^* \right) \quad (3.3.3)$$

with an initial value $\lambda_1 = 0.1$.

- The KMETHOD is AUTO or SEMI and if $i_k > 2$ and

$$(N_i - 7) > (\text{MAXITER} - i)$$

where i is an iteration count and MAXITER (integer parameter in NLPARM) is the maximum number of iterations allowed for an increment.

3.3.2 Divergence Criteria

The value of NEWK will be set to 2 if the solution tends to diverge. The MAXDIV field in the NLPARM entry requires an integer to specify a limit on the probable divergence conditions allowed for each iteration to continue. There are two classifications in the divergence condition: probable and absolute. The absolute divergence is treated as two occurrences of the probable divergence. **Divergence criteria** are defined based on the ratio of energy errors evaluated before and after the iteration, i.e., for the i -th iteration,

$$E^i = \frac{\{\Delta u^i\}^T \{R^i\}}{\{\Delta u^i\}^T \{R^{i-1}\}} \quad (3.3.4)$$

Then, the divergence criteria are established as follows:

- Absolute divergence if $E^i > 1 + 10^{-5}$ or $E^i < -10^{12}$
- Probable divergence if $-10^{12} < E^i < -1$ or excessive line search,

where the excessive line search condition is defined in Section 3.4. Another absolute divergence condition is established based on the heuristic observations to prevent thrashing with excessive K updates, i.e.,

$$\text{KMATUP} > (1 + \frac{i}{4}) \quad \text{and} \quad \text{KMETHOD}=\text{AUTO}$$

where KMATUP is a DMAP parameter representing the number of stiffness updates within an incremental step.

Recalling those criteria, the solution is considered diverging if

$$\text{NDIV} \geq \text{MAXDIV}$$

where NDIV is the number of **probable divergences** encountered during the iteration. It is noted that NDIV is incremented by 2 if the **absolute divergence** is encountered. When the solution is diverging, or NEWK=2, and if the value of MAXDIV is positive, the current solution base (displacement) is retracted so that the stiffness matrix is evaluated in reference to the preceding iteration step. If the divergence is detected at the first iteration ($i_k=1$) after a new [K] is formed, however, the current solution base is not retracted to prevent a repetition of the same diverging iteration. If the absolute divergence condition (NEWK=2) is encountered in two successive iterations in spite of the new stiffness, the computation will be terminated with a fatal message (UFM 4551) to that effect.

3.3.3 Time Expiration Criteria

If the remaining CPU time is not sufficient to continue the computation, the NLITER module will set the **NOGO flag** (DMAP parameter NSKIP=-2) and exit with a UFM 4311 to that effect. The purpose is to allow adequate time for the data recovery with solutions previously obtained. Sufficiency of the remaining time (t_r) is determined based on the K update time (t_k) and the iteration time (t_i), depending on the succeeding process anticipated. To reserve a minimum of 5% of the user-specified CPU time (t_s) for data recovery, the remaining time is defined as

$$t_r = (t_s - \text{elapsed CPU time}) - 0.05t_s .$$

Then the NOGO flag will be set under the following conditions:

- $t_r < 1.2t_k$
when the solution has not converged and the stiffness matrix is scheduled to be updated.
- $t_r < 10t_i$
when the solution needs more iterations without a new stiffness.

- $t_r < 1.8t_k$
when the solution has converged and the stiffness is to be updated.
- $t_r < 0.8t_k$
when the solution has converged and the stiffness update is not required. This condition includes pseudo convergence, where the number of iterations has reached MAXITER (>0). If the number of iterations reaches MAXITER while MAXITER is negative, the NOGO flag will be set regardless of the remaining time.

Flow diagrams for the stiffness update strategies, the **divergence and retreat procedure**, and the time expiration logic are shown for clarity in Figures 3.3.3, 3.3.4, and 3.3.5, respectively.

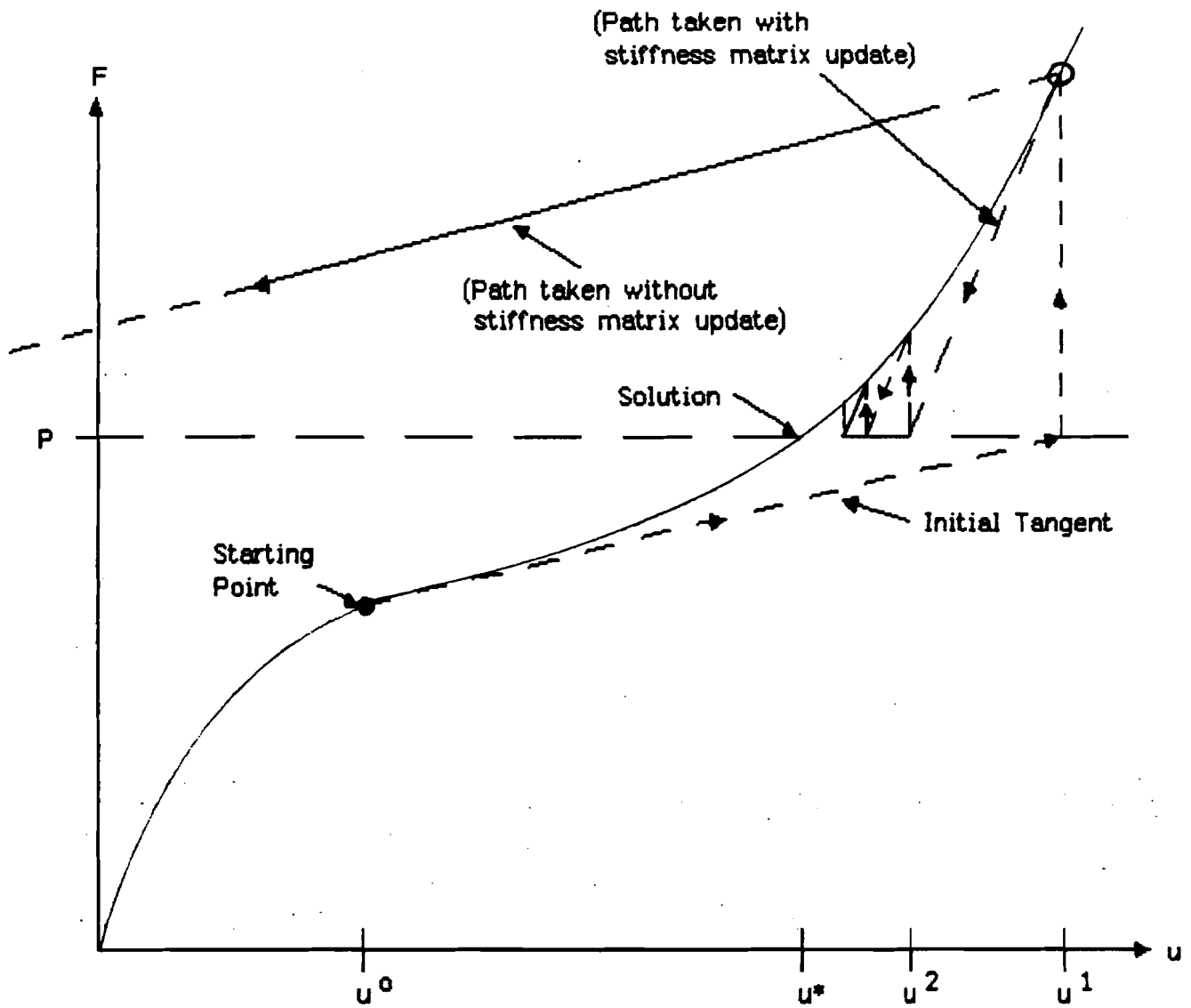
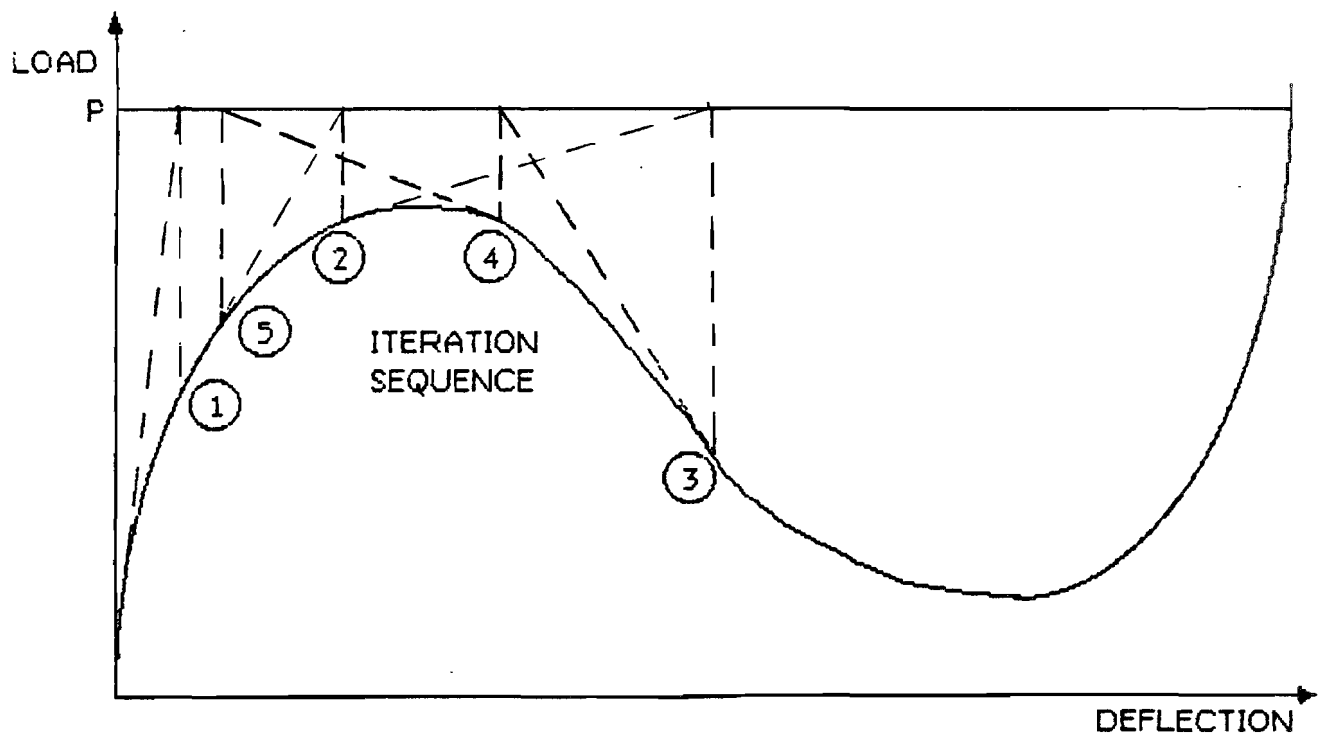
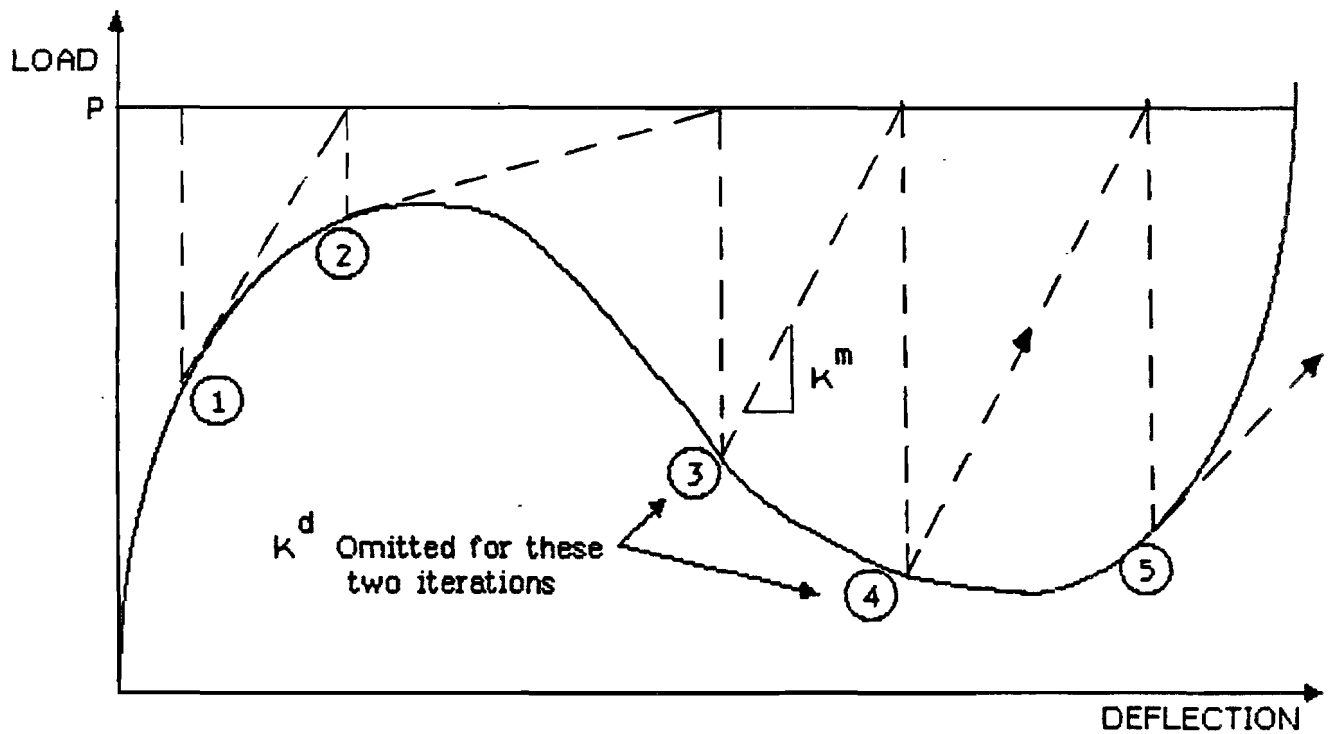


Figure 3.3.1 Hybrid Newton's Method based on Stiffness Matrix Update Strategy



- (a) Iteration stops at 3 with $\text{PARAM,TESTNEG}=-1$ if the stiffness becomes negative. Iteration will be trapped if continued with $\text{TESTNEG}=+1$ or 0.



- (b) Iteration sequence drops the differential stiffness with $\text{PARAM,TESTNEG}=-2$ when the negative diagonal terms appear in the stiffness matrix.

Figure 3.3.2 Iteration Trap Condition

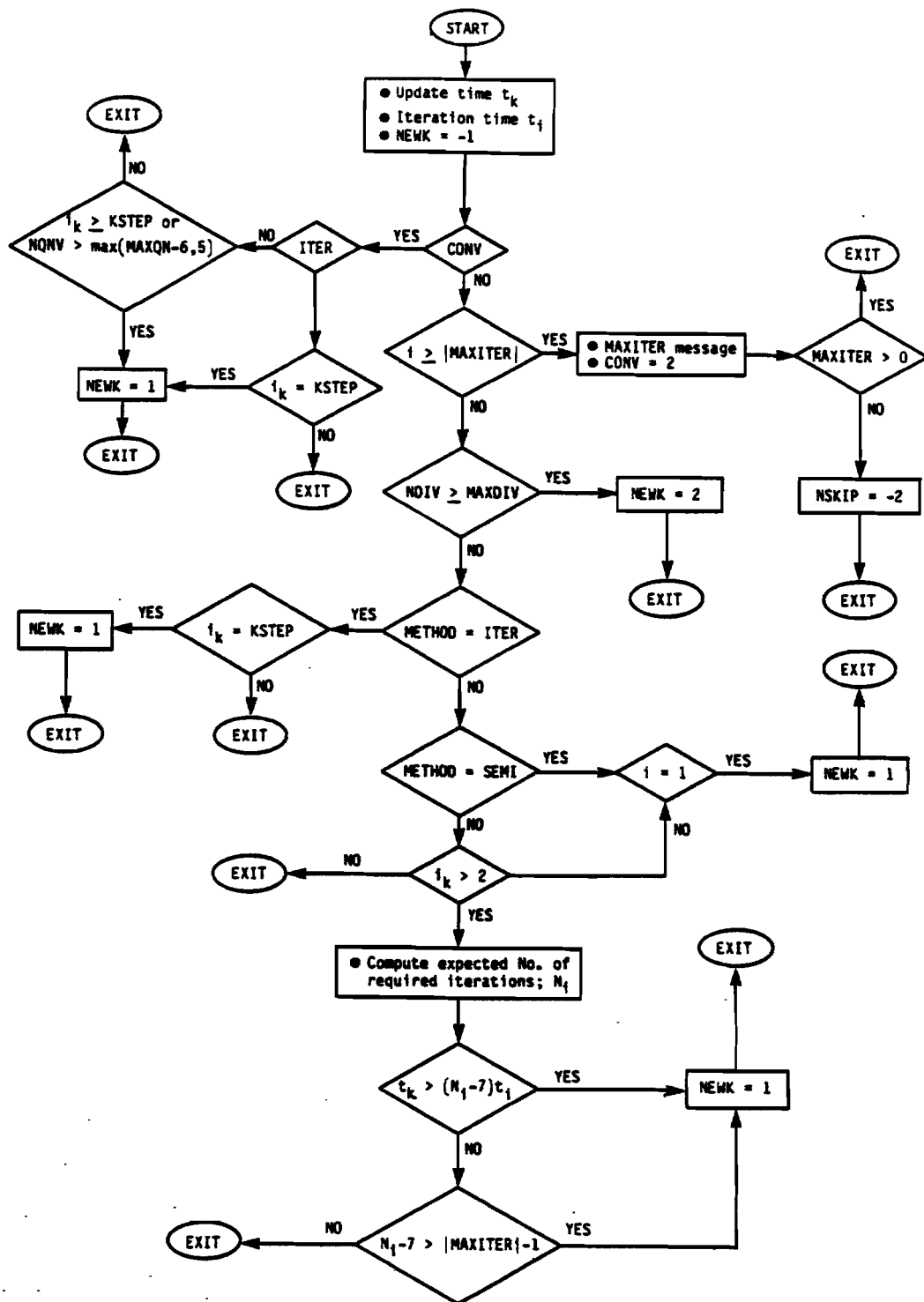


Figure 3.3.3 Flow Diagram for K Update Strategies (to set NEWK)

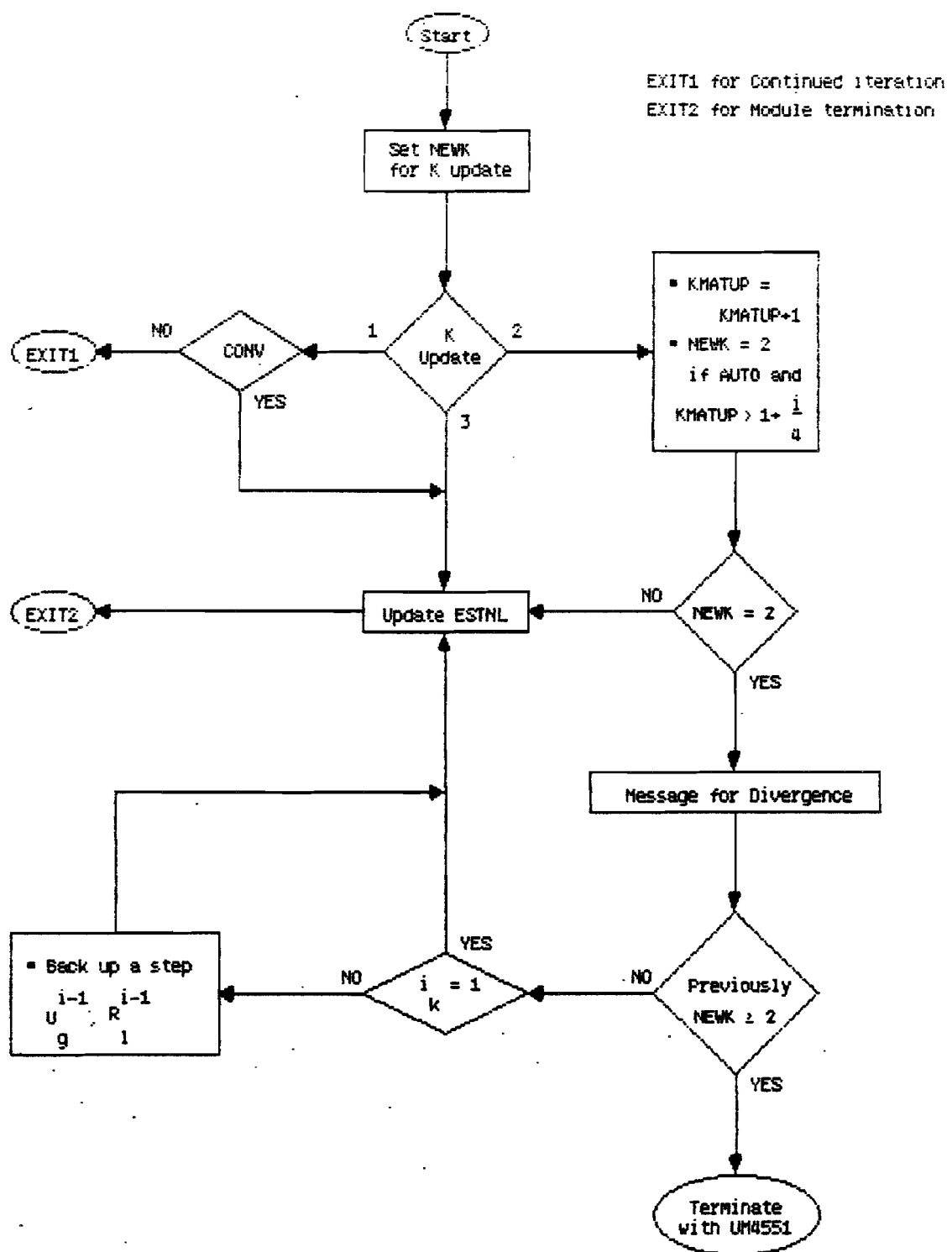


Figure 3.3.4 Divergence and Retreat Procedure

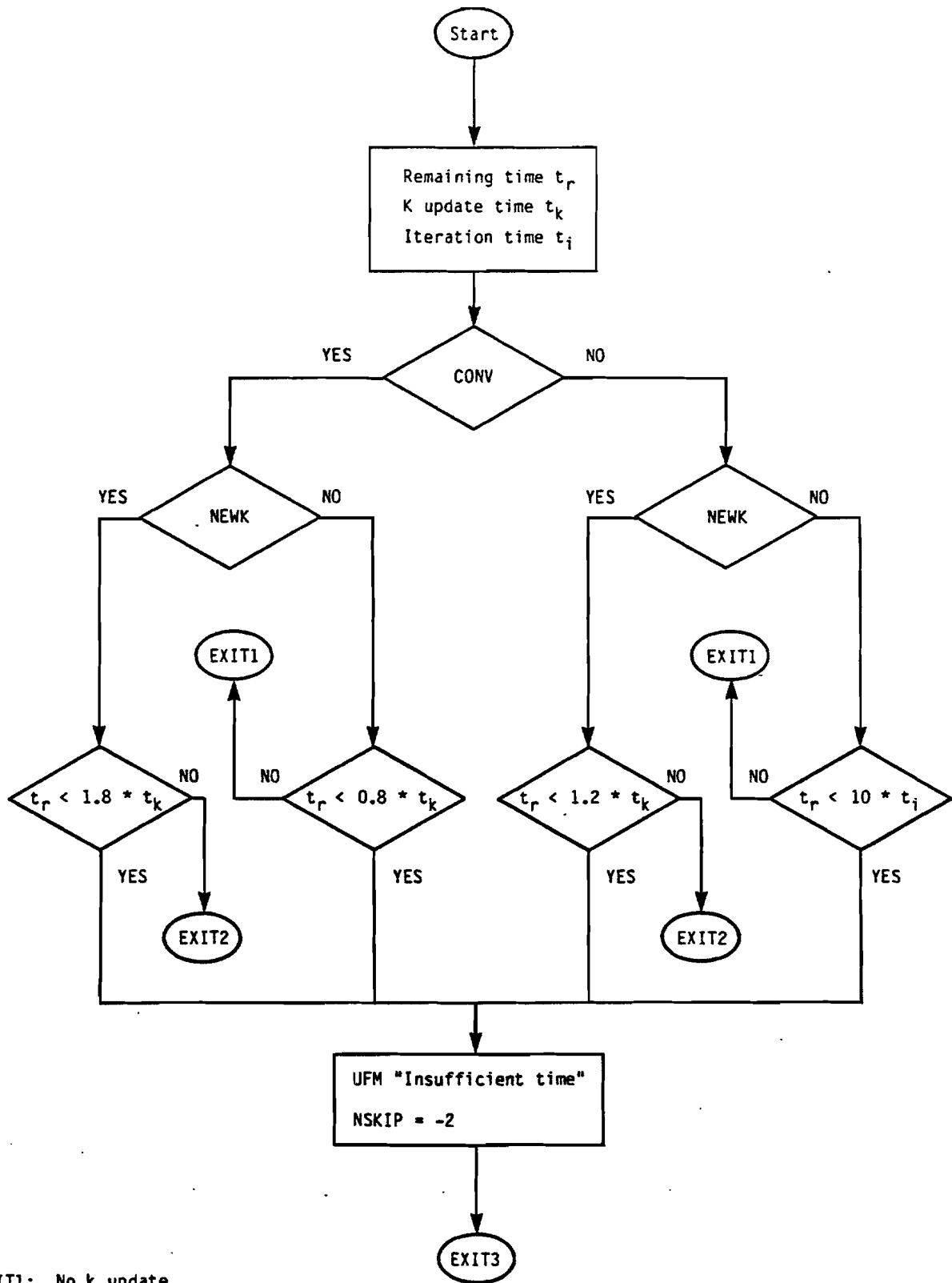


Figure 3.3.5 CPU Time Expiration Logic for NOGO Condition

3.4 LINE SEARCH METHOD

The line search method is well established as a basic descent method in nonlinear programming [3.4]. Although the method has been used for many years to improve the rate of convergence in the nonlinear iterations, the implementation scheme was not well elaborated in the finite element analysis [3.5-3.9]. While the line search method was implemented and used in this software, some insights were gained for making the option more viable and effective. The line search method was implemented and modified to optimize its usefulness for a general class of problems in conjunction with quasi-Newton updates [3.10].

3.4.1 Theoretical Basis

The functional to minimize in the finite element analysis is the total potential of a continuum, which is reduced to a quadratic function of displacements $\{u\}$ in linear problems as follows:

$$\Pi = U - W = \frac{1}{2}u^T K u - P^T u \quad (3.4.1)$$

where U is the strain energy of the system and W is the potential energy of the external loads. The basic idea behind Newton's method for the nonlinear iteration is that the objective function (Π) is approximated by a quadratic function in the vicinity of the trial solution point; i.e.,

$$\Pi(u) \simeq \Pi(u^i) + \frac{\partial \Pi}{\partial u} \{u - u^i\} + \frac{1}{2} \{u - u^i\}^T K \{u - u^i\} \quad (3.4.2)$$

The stationarity condition of Eq. (3.4.2) results in an iteration scheme for Newton's method with

$$\{u^{i+1}\} = \{u^i\} + K^{-1} \{R^i\}$$

where the unbalanced force $\{R^i\}$ represents the negative of the gradient vector at $\{u^i\}$, and the tangential stiffness matrix, $[K]$ at $\{u^i\}$, represents the Hessian matrix of the total potential.

The function Π must be convex in the vicinity of a solution to have a minimum point. The function is convex if and only if the Hessian matrix is positive semi-definite, i.e.,

$$d^T K d \geq 0 \quad (3.4.3)$$

where $\{d\}$ denotes a direction vector, corresponding to $\{u - u^i\}$. Then, a feasible direction $\{d\}$ must satisfy the necessary and sufficient condition:

$$\{R^i\}^T \{d\} \geq 0 \quad (3.4.4)$$

where the equality should be satisfied at the solution. This is the theoretical basis of the line search method.

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

$$u^i = u^{i-1} + \alpha d^i \quad (3.4.5)$$

with

$$d^i = K^{-1} R^{i-1}$$

where a positive search parameter (α) is determined such that

$$\{R^i\}^T \{d^i\} = 0. \quad (3.4.6)$$

The search parameter is introduced to expedite the convergence when Newton's method is used. It could be effectively used at a distance from the solution, where nonquadratic terms may not be negligible. By a line search procedure satisfying Eq. (3.4.6), the solution set $\{u\}$ is moved to a relative minimum point in the feasible direction $\{d\}$. Thus, the solution set is placed at a superior position for the next iteration.

The line search should be particularly effective when the modified Newton's method or the adaptive update method is used, because the approximation employed in Eq. (3.4.2) is not appropriate. The supplementary approximation for a quadratic function can be made by utilizing the gradients known at two successive iterations to estimate the Hessian matrix $[K]$, i.e.,

$$\Pi(u) \simeq \Pi(u^i) - \{R^i\}^T \{u - u^i\} - \frac{1}{2} \{u - u^i\}^T \frac{R^i - R^{i-1}}{u^i - u^{i-1}} \{u - u^i\} \quad (3.4.7)$$

which is known as the method of false position or the secant method [3.2]. The local minimum point can be obtained by making the derivative of Eq. (3.4.7) vanish, which results in

$$u = u^i - \frac{\{R^i\}^T \{d^i\}}{\{R^i - R^{i-1}\}}. \quad (3.4.8)$$

This equation represents a linear interpolation in terms of $\{u\}$ and $\{R\}$, which is the basis of the line search method. Along the direction of descent $\{d\}$, the objective function to be minimized can be regarded as a function of a single variable, the search parameter. The total potential may be expressed in terms of the search parameter by substituting Eq. (3.4.5) into Eq. (3.4.7); i.e.,

$$\Pi(\alpha) = \Pi(1) - (\alpha - 1) \{R^i\}^T \{d^i\} - \frac{1}{2} (\alpha - 1)^2 \{d^i\}^T \{R^i - R^{i-1}\} \quad (3.4.9)$$

When the derivative of Eq. (3.4.9) with respect to α disappears, the search parameter for a local minimum is obtained by

$$\alpha = 1 - \frac{\{R^i\}^T \{d^i\}}{\{R^i - R^{i-1}\}^T \{d^i\}}. \quad (3.4.10)$$

3.4.2 Search Criteria

It is impractical in general to obtain the exact minimum point for the line search procedure because it does not render a final solution. Furthermore, each line search costs a considerable amount of computing time, comparable to that of a modified Newton's iteration, to evaluate element forces. Therefore, it is often desirable to sacrifice accuracy in the line search and to conserve the overall computing time by terminating the search procedure before it has actually converged to the local minimum.

One method of restricting the line search is to limit the maximum number of consecutive searches at each iteration cycle. Another criterion for limiting line searches is based on the divergence rate defined by a ratio of energy errors before and after the iteration, i.e.,

$$E_k = \frac{\{d^i\}^T \{R_k^i\}}{E_0} \quad (3.4.11)$$

with

$$E_0 = \{d^i\}^T \{R^{i-1}\}$$

where i and k are indices for the iteration and the line search (within the i -th iteration), respectively. Then, a criterion is established to end the line search procedure if

$$- \text{LSTOL} \leq E_k \leq \text{LSTOL} \quad (3.4.12)$$

where LSTOL is a tolerance specified in the NLPARM entry.

In the neighborhood of the solution point, the search parameter should have a value close to unity. The line search will result in a dilemma if the selected search parameter is too small or too large. If it is too small (i.e., close to zero), the iteration is not effective because the solution vector retracts to the old value. To circumvent such a situation, the bisection process can be adopted when α approaches zero; i.e.,

$$\alpha_{new} = \frac{1}{2}(\alpha_{old} + \alpha_{new}) \quad \text{if} \quad \alpha_{new} < \text{ATOL} \quad (3.4.13)$$

where ATOL is a tolerance. An excessively large value of α (compared to unity) does not validate the underlying assumption of close proximity to the solution point and may cause a numerical instability.

3.4.3 Implementation of Search Procedure

The search method was varied depending on the value of the relative error E_k in Eq. (3.4.11). If E_k is initially (i.e., at $\alpha = 1$) greater than unity, the solution is diverging and the stiffness matrix should be updated without the line search. If E_k is initially less than unity but outside the tolerance, the search procedure is initiated by utilizing one of the following two schemes:

- Linear interpolation with Illinois algorithm if $E_k < -\text{LSTOL}$
- Doubling scheme if $\text{LSTOL} < E_k < 1$

where LSTOL has a default value of 0.5. Note that the line search would not be meaningful if the denominator (E_0) in Eq. (3.4.11) approaches zero. If it is close to zero, in fact, the solution should have almost converged and the line search would not be required. Under such circumstances, the line search is blocked by assigning zero to E_1 .

The domains for different operations are shown schematically in Fig. 3.4.1, where E_k is expressed as a function of α . The relative error E_k is normalized such that E_k is unity at $\alpha = 0$. If E_1 is negative, there should be an α , between 0 and 1, that makes the error E_k disappear. This value is predicted by a linear interpolation, i.e.,

$$\alpha_{k+1} = \alpha_k - E_k \frac{\alpha_k - \alpha_{k-1}}{E_k - E_{k-1}} \quad (3.4.14)$$

where k is a line search index. With $k = 1$, corresponding to $\alpha = 1$, Eq. (3.4.14) is equivalent to Eq. (3.4.10). A linear extrapolation can also be conceived by Eq. (3.4.14). However, two points (corresponding to indices k and $k - 1$) involved in Eq. (3.4.14) are not necessarily the most recent data points acquired from the preceding line searches. These two points were so chosen to maintain two most recent points with the errors in opposite sign and the extrapolation was avoided. This scheme is employed because the error function (E_k) may not be a monotonic function of α , as exemplified in Fig. 3.4.1. Thus, Eq. (3.4.14) should be rewritten as:

$$\alpha_{k+1} = \alpha_{\text{new}} - E_{\text{new}} \frac{\alpha_{\text{new}} - \alpha_{\text{old}}}{E_{\text{new}} - E_{\text{old}}} \quad (3.4.14a)$$

The Illinois algorithm was introduced for the subsequent searches to achieve faster convergence in the succeeding search process. If two successive line searches result in the error of the same sign, the next line search uses one-half the value of the error in opposite sign for interpolation, i.e.,

$$E_{\text{old}} = \frac{1}{2} E_{\text{old}} \quad (3.4.15)$$

before using Eq.(3.4.14a). An example of the Illinois algorithm is illustrated in Fig. 3.4.1. As shown in the example, the error E_k is allowed to increase in magnitude if it is negative because the root ($E_k = 0$) is guaranteed to exist in this case.

For a positive value of E_1 (for $\alpha = 1$), the zero-crossing could be predicted by a linear extrapolation based on the same mathematical reasoning as interpolation. This extrapolation may result in a very large value of α , if E_1 is close to 1. For this reason, a doubling scheme was employed as a conservative approach by simply doubling the old value of α . The search procedure is terminated if the doubling scheme results in an increasing relative error E_k . If E_k becomes negative ($< -\text{LSTOL}$), the doubling scheme is switched to the interpolation method in the subsequent searches as shown in Fig. 3.4.2. The initial and the final values of E_k are printed for every iteration under the headings E-FIRST and E-FINAL, respectively.

The number of consecutive line searches allowed for each iteration is limited by a user-specified parameter, MAXLS. This parameter, MAXLS, is defaulted to 5 for static analysis and 2 for dynamic analysis. Upper and lower bounds of α are effectively set by combined effects of MAXLS and the doubling scheme or the bisection. The bisection process is initiated based on the criterion:

$$\alpha < \text{ATOL}$$

in which

$$\text{ATOL} = \frac{1}{10 * \text{MAXLS}}$$

During the line search iteration, the data relevant to the computed minimum error (E_{min} , α_{min} , and R_{min}) are resumed in case the line search is terminated due to reaching MAXLS or increasing error (E_k). The line search algorithm and its iteration scheme are shown succinctly by flow diagrams in Figs. 3.4.2 and 3.4.3.

The method of false position is not globally convergent, or rather the process converges only if it is started in close proximity to the solution point. The order of its convergence approaches 1.618, the golden mean [3.4]. The convergence rate of the current line search process is somewhat reduced by the doubling scheme, but expedited by virtue of the Illinois algorithm. However, the line search process is terminated before accurate convergence is achieved, and the convergence rate as such is rather immaterial.

3.4.4 One-Dimensional Example

It is instructive to examine the line search procedure in a single degree of freedom system. Consider a fictitious structure that responds to the displacement with the internal force described by a parabolic function:

$$F(u) = -u^2 + 6u.$$

Suppose that the structure is in equilibrium at $u = 1$ with an internal force (F) of 5. Starting from this point, the external load (P) is increased from 5 to 8, as shown by a dashed line in Fig. 3.4.4. The first iteration, ending at point 1, represents a Newton's iteration with the initial stiffness ($K_0 = 6$); i.e.,

$$R^0 = P - F^0 = 8 - 5 = 3$$

$$d^1 = \frac{R^0}{K_0} = \frac{3}{6} = 0.5$$

which results in

$$u^1 = u^0 + d^1 = 1.5$$

and

$$R^1 = P - F^1 = 8 - 6.75 = 1.25$$

To perform the line search at point 1, the relative error in Eq. (3.4.11) is estimated:

$$E_1 = \frac{d^1 R_1^1}{d^1 R^0} = \frac{1.25}{3} = 0.4167.$$

Since this error is smaller than the line search error tolerance by default (LSTOL= 0.5), the line search will normally be skipped. If the line search is enforced (e.g., LSTOL= 0.4), the doubling scheme will be employed since E_1 is positive. The doubling scheme will result in:

$$u_2^1 = u^0 + 2. * d^1 = 2.0$$

which is an exact solution and the iteration is finished. For the sake of comparison, however, let us employ an extrapolation scheme for the line search, i.e., based on Eq. (3.4.14):

$$\alpha_2 = \alpha_1 - E_1 \frac{\alpha_1 - \alpha_0}{E_1 - E_0} = 1. - \frac{0.4167}{0.4167 - 1.} = 1.714$$

which results in:

$$u_2^1 = u^0 + 1.714 * d^1 = 1.86$$

and

$$R^2 = P - F^2 = 8 - 7.7 = 0.3$$

This point corresponds to point 2, which can be obtained by connecting points 0 and 1 on the dashed line.

The line search seems to be more effective in the stiffening structure, which is illustrated with another parabolic function representing an internal force:

$$F(u) = u^2 + u.$$

Suppose that the structure is in equilibrium at $u = 1$ with $F = 2$ and an external force (P) of 6 is applied as shown by a solid line in Fig. 3.4.4. The first iteration is a modified Newton's iteration with a stiffness evaluated at certain point previously, e.g., $K_0 = 2$:

$$\begin{aligned} R^0 &= P - F^0 = 6 - 2 = 4 \\ d^1 &= R^0 / K_0 = 4/2 = 2 \\ u^1 &= u^0 + d^1 = 1 + 2 = 3 \\ R^1 &= P - F^1 = 6 - 12 = -6 \end{aligned}$$

The first line search will result in:

$$\begin{aligned} E_1 &= R_1^1 / R^0 = -6/4 = -1.5 \\ \alpha_2 &= \alpha_1 - E_1 \frac{\alpha_1 - \alpha_0}{E_1 - E_0} = 0.4 \\ u_2^1 &= u^0 + \alpha_2 d^1 = 1.8 \\ R_2^1 &= P - F_2^1 = 0.96 \end{aligned}$$

The second line search will result in:

$$\begin{aligned} E_2 &= R_2^1 / R^0 = 0.24 \\ \alpha_3 &= \alpha_2 - E_2 \frac{\alpha_2 - \alpha_1}{E_2 - E_1} = 0.4828 \\ u_3^1 &= u^0 + \alpha_3 d^1 = 1.9656 \\ R_3^1 &= P - F_3^1 = 1.7018 \end{aligned}$$

The second line search would not have been performed if LSTOL= 0.5. Let us assume that LSTOL= 0.02 to continue discussion. The third line search would result in:

$$E_3 = R_3^1 / R^0 = 0.0427$$

Now that E_3 is positive again, the Illinois algorithm is invoked:

$$\begin{aligned} \alpha_4 &= \alpha_3 - E_3 \frac{\alpha_3 - \alpha_1}{E_3 - E_1/2} = 0.5107 \\ u_4^1 &= u^0 + \alpha_4 d^1 = 2.0213 \\ R_4^1 &= P - F_4^1 = -0.1071 \end{aligned}$$

As the line search continues, the solution converges to $u = 2$ with $\alpha = 0.5$, without requiring additional iterations. Such a convergence by using the line search alone is a peculiar behavior in the single degree of freedom problem. The line search has a single parameter, which is not sufficient to render a solution to a multi-variable system unless the direction vector happened to be oriented in the right direction.

3.4.5 Other Provisions for Line Search

Notice that the displacement vector in the g -set is required to compute internal element forces (F) at every iteration, while the residual error vector (R), to be minimized in the iterative process, comprises only the l -set. For this reason, conversions from the l -set to the g -set and vice versa should be performed at every iteration. In this context, there is another conversion of displacement increments to treat the cumulative gimbal angle for large rotations. The line search uses the direction vector in the g -set to avoid additional set conversions in the search process, i.e.,

$$u_g = u_g + \alpha d_g,$$

where d_g can be obtained by merging displacement increments in the m -set (Δu_m) and s -set (ΔY_s) to the l -set displacement increment from the Newton's iteration. It is noted that the line search cannot be performed at the first iteration if the enforced displacements (SPCD or SPC) are specified. This is because d_g includes ΔY_s at the first iteration, but is set to zero at the second iteration and on.

Considering Eq. (3.4.11), the line search would not be meaningful if the denominator (E_0) approaches zero, not to mention the numerical difficulty. In fact, if the denominator is close

to zero, the solution should have almost converged and the line search should not be required. There are provisions to block the line search under such circumstances as follows:

- Reset $E_1=0$ if $\text{ABS}(E_0) < \text{EPSW} \times E^*$ for the first iteration;
- Reset $E_1=0$ if $\text{ABS}(E_0) < 10^{-5} \text{EPSW} \times E^*$ for subsequent iterations;
- Reset $E_1=\text{LSTOL}$ if $\text{ABS}(E_k) > \text{LSTOL}$ and $\text{ABS}(E_0) < 0.01 \text{EPSW} \times E^*$;

where EPSW is a convergence tolerance in terms of work from NLPARM and E^* is introduced for dimensionless criteria (defined in Section 3.6). The first condition is employed for the creep analysis because the line search or the quasi-Newton update is not appropriate for the first iteration of the creep which always causes a small value of E_0 . As will become clear in Section 3.5, the quasi-Newton update is not used if $E_1=0$. The third condition is, therefore, introduced to allow quasi-Newton updates while suppressing the line search.

It was discovered that the solution algorithm tends to thrash line searches when the structure becomes physically unstable, such as at the collapse load. To alleviate exhausting CPU resources in such cases, excessive line searches are treated like a probable divergence condition by incrementing NDIV by 1. The excessive line search is defined to satisfy

$$\text{TLSCT} > (2 \times \text{MAXLS} - 1)$$

where TLSCT is the cumulative number of line searches after the new stiffness matrix is formed within an increment.

Upon termination of the line search procedure, the results will be printed with other iteration information under the following heading:

- FACTOR : α (final value for the search parameter)
- EFIRST : E_1 (relative error at $\alpha=1$, divergence rate)
- EFINAL : E_i (E_k at the end of line search)
- N-LS : LSCT (line search count k at the end)

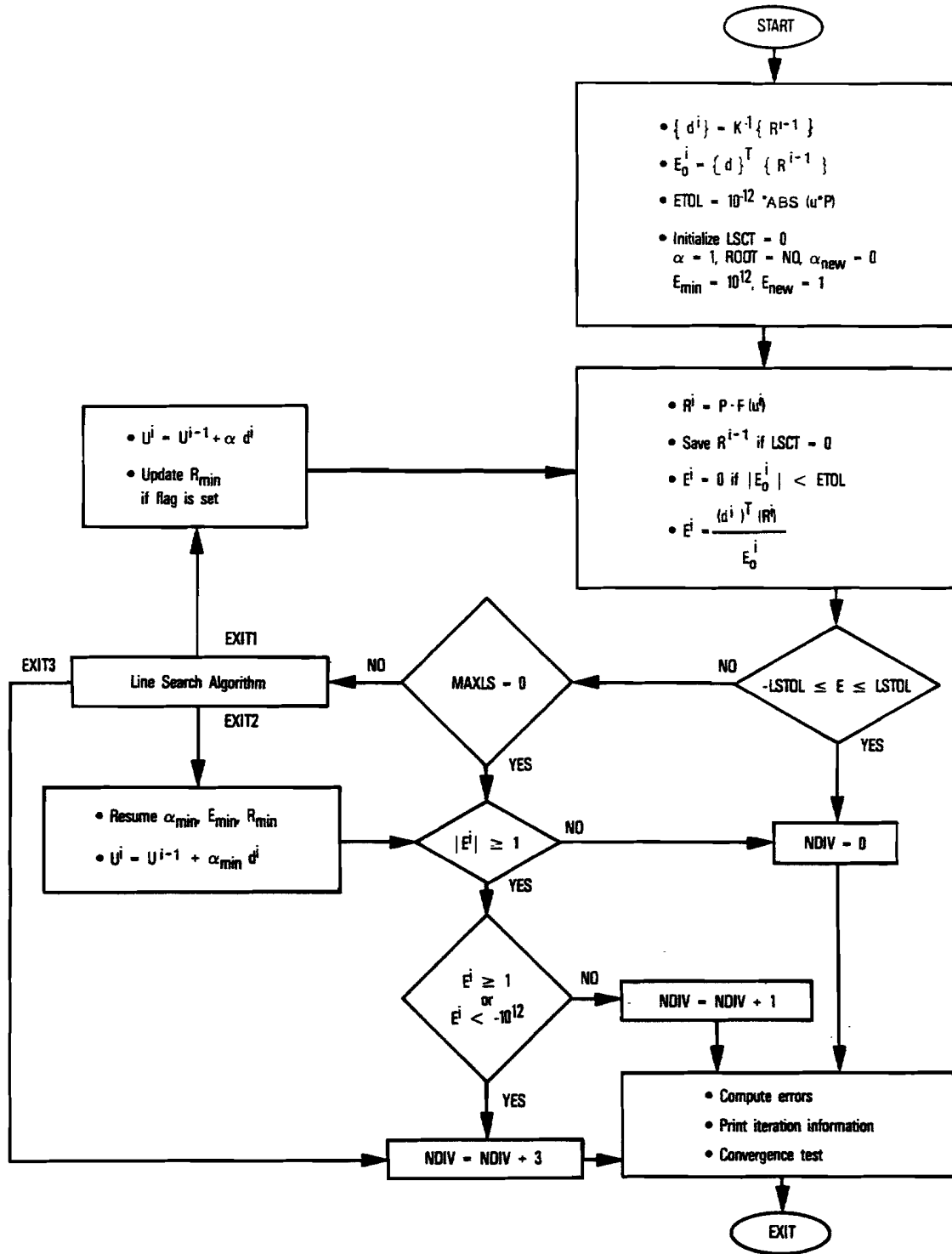


Figure 3.4.3 Flow Diagram for the Line Search Iteration

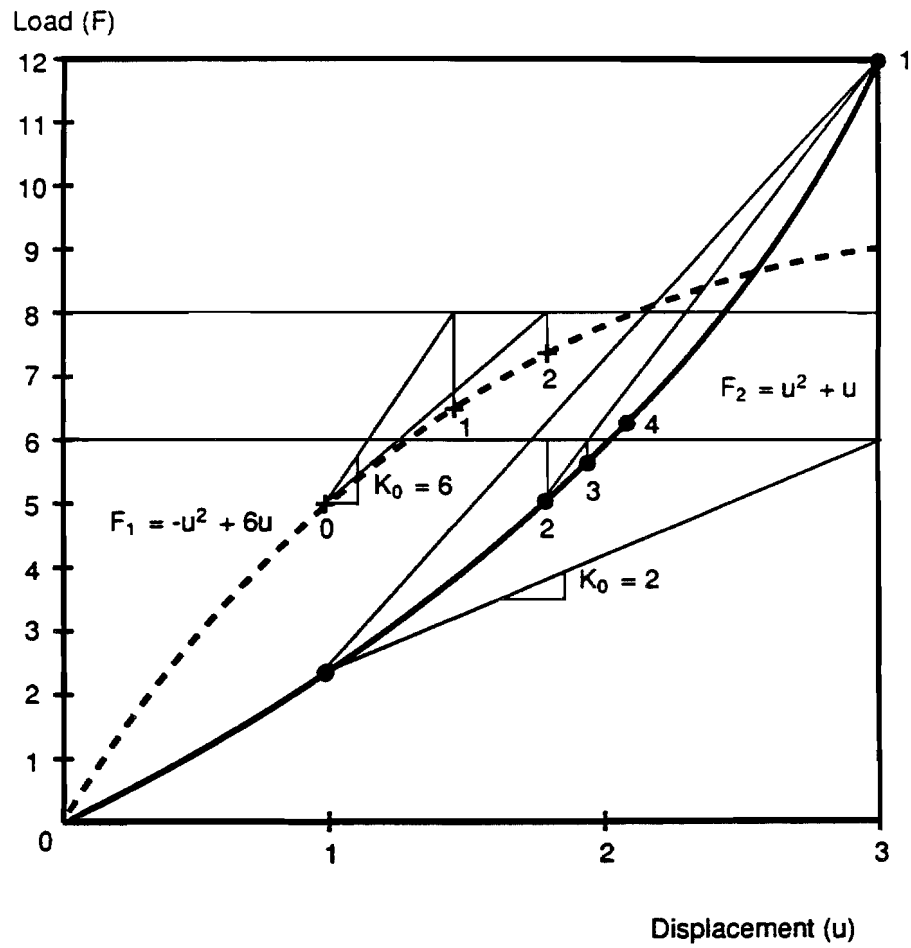


Figure 3.4.4 Examples for the Line Search Procedure

3.4.6 Quadratic Interpolation

A quadratic interpolation scheme is added to the line search method in Version 67. The quadratic interpolation is implemented as an optional feature to be used instead of the linear interpolation. A quadratic line search on the first iteration is the default option. In order to suppress the quadratic interpolation, SYSTEM(141)=1 must be included in the NASTRAN definition at the beginning of the input data, i.e.,

$$\text{NASTRAN SYSTEM}(141)=1 \$$$

which will revive the linear interpolation on the first iteration as in Version 66.

What follows is the theoretical basis for the quadratic interpolation. Let the error (generalized residual force vector) be defined (in terms of α) by

$$E(\alpha) = d^T R(\alpha) \quad \text{for } \alpha > 0. \quad (3.4.16)$$

Known values of this function are

$$E_0 = E(0) = d^T R^0 \quad \text{for } \alpha = 0$$

and

$$E_1 = E(1) = d^T R^1 \quad \text{for } \alpha = 1.$$

The derivative of E can be expressed as

$$\left. \frac{dE}{d\alpha} \right|_{\alpha=0} = d^T \left. \frac{\partial R}{\partial \alpha} \right|_{\alpha=0} \simeq -d^T K_0 d = -d^T R^0 = -E_0 \quad (3.4.17)$$

where K_0 is a stiffness matrix formed at some preceding step or iteration. In reference to Fig. 3.4.5, the coefficients for the quadratic curve fitting can be obtained as

$$E(\alpha) \simeq c_2 \alpha^2 - E_0 \alpha + E_0 \quad (3.4.18)$$

where $c_2 = E_1$ from $E(1) = E_1$. Solving for $E(\alpha) = 0$,

$$\alpha = \frac{1 \pm \sqrt{1 - 4E_1/E_0}}{2E_1/E_0} \quad (3.4.19)$$

Real roots can exist for $\frac{E_1}{E_0} \leq 0.25$. Considering the previous search criteria, the doubling scheme is used for $0.5 < \frac{E_1}{E_0} < 1$ and the linear interpolation with Illinois algorithm is used for $\frac{E_1}{E_0} < -0.5$. Since the doubling scheme is safe and conservative, it is necessary to introduce the quadratic interpolation only for $\frac{E_1}{E_0} < -0.5$. In this domain the determinant should be

$$1 - 4 \frac{E_1}{E_0} > 3.0$$

Since $\alpha > 0$, the positive sign in Eq. (3.4.19) should be discarded, i.e.,

$$\alpha = \frac{1 - \sqrt{1 - 4E_1/E_0}}{2E_1/E_0} \quad (3.4.19a)$$

which is valid for $\frac{E_1}{E_0} \leq 0$. Notice that the geometric stiffening case belongs to this domain. The quadratic interpolation is deemed effective for the geometric stiffening case.

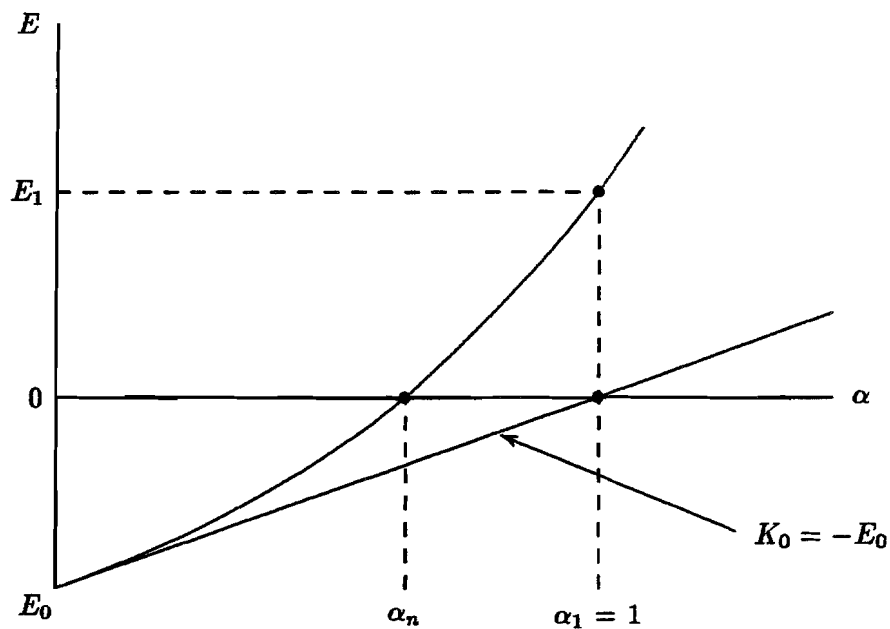


Figure 3.4.5 Quadratic Interpolation Algorithm

3.5 QUASI-NEWTON METHOD

The quasi-Newton update method is employed as an option to accelerate the convergence and improve the overall effectiveness of the modified Newton's iteration. When the BFGS method was implemented and tested, implementation method and adjustment of parameters had a significant impact on the program's performance [3.11].

Quasi-Newton update methods are considered the most sophisticated methods of the Newton's iterative solution schema and represent the culmination of extensive algorithm development for Newton's iterative process [3.4]. They have been developed for efficient nonlinear iterations by approximating the inverse Hessian matrix in place of the true inverse that is required in Newton's method. Quasi-Newton methods have been widely and successfully used in nonlinear optimization applications. One of the quasi-Newton techniques, known as BFGS (Broyden-Fletcher-Goldfarb-Shanno) update, was introduced into the finite element method by Matthies and Strang [3.8].

By the BFGS update, the information acquired during the iteration is used to modify the inverse stiffness matrix. This approximate update to the inverse stiffness matrix results in a secant modulus in the search direction. As these updates accumulate, the BFGS method renders a stiffness matrix resembling tangential stiffness in the limit. When combined with the line search [3.10], the performance of the BFGS update with respect to effectiveness and efficiency depends largely on the implementation. It is extremely difficult to tune the tolerances and parameters to the optimal condition, which may be problem dependent, for the general class of problems.

3.5.1 Evolution of Quasi-Newton Method

The basic concept of quasi-Newton methodology is to construct an approximate inverse Hessian (stiffness) matrix using information gathered during the descent process. The current approximation is used at the next iteration to define the next feasible direction in the modified Newton's method. Upon the first iteration, a feasible direction of descent is given by

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

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

$$d^2 = K_0^{-1} \{R(u^1)\}$$

with

$$u^1 = u^0 + d^1$$

Consider a Taylor series expansion of the load error $\{R\}$ about $\{u^i\}$, expressed as

$$R = R(u^i) - K(u - u^i) + O(h) \quad (3.5.1)$$

where

$$R(u) = P - F(u)$$

with $F(u)$ being the aggregate vector of element nodal forces. Assuming that the load stiffness due to follower forces is negligible, the stiffness matrix is formed by

$$K = -\frac{\partial R}{\partial u} = \frac{\partial F}{\partial u}$$

In view of Eq. (3.5.1), the data from two points, $\{u^{i-1}\}$ and $\{u^i\}$, should provide some information about $[K]$, because they should satisfy:

$$\gamma = K\delta \quad \text{or} \quad \delta = K^{-1}\gamma \quad (3.5.2)$$

with

$$\delta = u^i - u^{i-1} \quad (3.5.3)$$

and

$$\gamma = R^{i-1} - R^i. \quad (3.5.4)$$

It is natural to attempt constructing successive approximations to $[K]^{-1}$ based on the data obtained during the iterative procedure.

The earliest quasi-Newton scheme was proposed by Davidon and later elaborated by Fletcher and Powell. By this scheme which is referred to as the DFP method [3.4], the inverse Hessian is updated by adding two symmetric rank one matrices at each iteration. Therefore, the scheme is a rank two correction procedure, i.e.,

$$[K_{i+1}]^{-1} = [K_i]^{-1} + \frac{\delta \delta^T}{\delta^T \gamma} - \frac{K_i^{-1} \gamma \gamma^T K_i^{-1}}{\gamma^T K_i^{-1} \gamma}. \quad (3.5.5)$$

Notice that the formula simply satisfies Eq. (3.5.2) while preserving positive definiteness and symmetry of $[K]^{-1}$.

It is also possible to update approximations to the stiffness matrix itself, rather than its inverse. Recalling the complementary roles of $[K]$ and $[K]^{-1}$ with respect to the quasi-Newton vectors in Eq. (3.5.2), the formula for $[K]$ is found by interchanging γ and δ , i.e.,

$$[K_{i+1}] = [K_i] + \frac{\gamma \gamma^T}{\gamma^T \delta} - \frac{K_i \delta \delta^T K_i}{\delta^T K_i \delta}. \quad (3.5.6)$$

Another way of finding the formula for the inverse stiffness matrix is to invert Eq. (3.5.6). This can be done by applying the general inversion identity known as the Sherman-Morrison formula. The resulting formula is referred to as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update for the inverse Hessian, i.e.,

$$[K_{i+1}]^{-1} = [K_i]^{-1} + \left(1 + \frac{\gamma^T K_i^{-1} \gamma}{\gamma^T \delta}\right) \frac{\delta \delta^T}{\delta^T \gamma} - \frac{\delta \gamma^T K_i^{-1} + K_i^{-1} \gamma \delta^T}{\gamma^T \delta}. \quad (3.5.7)$$

Numerical experiments have indicated that the performance of the BFGS method is superior to that of the DFP method [3.4].

An equivalent form of Eq. (3.5.7) can be expressed for the j -th BFGS update as:

$$[K_j]^{-1} = [C_j]^T [K_{j-1}]^{-1} [C_j] + z_j \{\delta_j\} \{\delta_j\}^T \quad (3.5.8)$$

where

$$[C_j] = [I] - z_j \{\gamma_j\} \{\delta_j\}^T$$

and

$$z_j = \frac{1}{\{\delta_j\}^T \{\gamma_j\}}.$$

This is a recurrence formula employed in MSC/NASTRAN, which is applicable to every pair of quasi-Newton(QN) vectors. Notice that the index j for the BFGS update may be different from the iteration index i . The stiffness matrix (therefore its inverse) is assumed to be symmetric and positive definite throughout the derivation.

3.5.2 Criteria for BFGS Update

There are cases where the BFGS update is not appropriate. These cases include the following:

Case 1. When the solution diverges at the current iteration.

Case 2. When the BFGS update makes $[K]^{-1}$ near singular.

Case 3. When the BFGS update makes $[K]$ near singular.

Case 4. When two successive data points involved in QN vector generation are too closely spaced.

Case 5. When the change in $[K]^{-1}$ due to BFGS update is negligible.

Characteristics of such cases should be identified so that undesirable updates can be avoided.

The BFGS update formula in Eq. (3.5.8) can be expressed in an equivalent form as follows:

$$[K_j]^{-1} = [A_j]^T [K_{j-1}]^{-1} [A_j] \quad (3.5.9)$$

where

$$[A_j] = [I] + \{v_j\} \{w_j\}^T \quad (3.5.10)$$

with

$$\{v_j\} = \left(\frac{\delta_j^T \gamma_j}{\delta_j^T K_{j-1} \delta_j} \right)^{1/2} [K_{j-1}] \{\delta_j\} - \{\gamma_j\}$$

and

$$\{w_j\} = \frac{\{\delta_j\}}{\{\delta_j\}^T \{\gamma_j\}}.$$

The condition number (ratio of the largest to the smallest eigenvalue) of the matrix $[A]$ can be used to determine the characteristics of $[K]^{-1}$. Eigenvalues of $[A]$ consist of multiple roots of unity and a single non-unity eigenvalue, i.e.,

$$\lambda = 1 + \{w_j\}^T \{v_j\}. \quad (3.5.11)$$

This non-unity eigenvalue itself or its reciprocal becomes the condition number.

In MSC/NASTRAN, the criteria for BFGS updates are established based on the squared value of Eq. (3.5.11), which can be simplified as:

$$\lambda^2 = \frac{\delta_j^T K_{j-1} \delta_j}{\delta_j^T \gamma_j} = \frac{\alpha}{1 - E^i} \quad (3.5.12)$$

where E^i is the divergence rate defined as

$$E^i = \frac{\{d^i\}^T \{R^i\}}{\{d^i\}^T \{R^{i-1}\}} \quad (3.5.13)$$

with

$$\{d^i\} = [K]^{-1} \{R^{i-1}\}$$

and α is the line search parameter [3.10]. Based on Eq. (3.5.12), the following observations can be made:

Case 1. In case of divergence ($E^i > 1$), λ has an imaginary value.

Case 2. As the condition number approaches zero, $[K_j]^{-1}$ will become nearly singular.

Case 3. If E^i approaches 1, the condition number approaches infinity, which makes $[K_j]$ tend to be singular.

Case 4. A small change in energy (i.e., $\{\delta_j\}^T \{\gamma_j\}$ approaches zero) makes the condition number approach infinity, which causes $[K_j]$ to become singular.

Case 5. The BFGS should not be needed near the convergence or in the linear case where λ is close to unity.

Hence the undesired BFGS updates can be filtered based on the squared value of the condition number in Eq. (3.5.12).

3.5.3 Implementation of the BFGS Update Strategy

The overall picture of the iteration module, in which the BFGS update method is implemented, is shown by a flow diagram in Fig. 3.5.1. Undesirable BFGS updates can be avoided by establishing two admissible domains for the expression in Eq. (3.5.12), in which the updates are performed:

$$TOL1 < \lambda^2 < 1 - TOL2 \quad \text{for } E^i < 0$$

and

$$1 + TOL2 < \lambda^2 < TOL3 \quad \text{for } E^i > 0.$$

where the tolerances are determined based on the numerical experiments [3.14]. The algorithm begins with an initial stiffness $[K_0]$ (evaluated at some previous step), which has already been decomposed, i.e.,

$$K_0 = L D L^T \quad (3.5.14)$$

where L is a lower triangular matrix and D is a diagonal matrix with the pivot elements in the diagonal. Then, the direction vector is obtained by the forward and backward substitution, expressed symbolically as

$$d_i = K_0^{-1} R^{i-1} \quad (3.5.15)$$

The quasi-Newton vectors are determined based on the data from two iteration steps, i.e.,

$$\{\delta_j\} = \{u^i\} - \{u^{i-1}\} = \alpha \{d^i\} \quad (3.5.16)$$

and

$$\{\gamma_j\} = \{R^{i-1}\} - \{R^i\},$$

where α is unity if the line search is not performed. Notice that vectors $\{d^i\}$ and $\{R^{i-1}\}$ are generated before the line search is initiated and no QN vectors are created during the line search process. If the candidate quasi-Newton vectors satisfy the criterion based on the condition number, these vectors will be stored for the subsequent iterations by appending them in a file (named QNV). However, the number of QN vector pairs to be stored is limited by a user-specified parameter, MAXQN, which is defaulted to the maximum number of iterations allowed (defaulted to 20). If MAXQN has been exceeded, the QNV file is not altered until it is purged upon a stiffness matrix update. It is noted that the stiffness matrix is updated upon convergence if MAXQN has been exceeded in order to stop dragging too many QN vectors. The BFGS update can be suppressed by specifying MAXQN=0.

The BFGS update is a sequence of contiguous processes of dot products (4n multiplications for each pair of QN vectors with n components), combined with a forward and backward substitution (FBS). Using Eq. (3.5.8) for the BFGS update, multiple QN vectors can be processed as follows:

$$\begin{aligned} \{d^i\} &= [K_j]^{-1} \{R^{i-1}\} \\ &= [C_j]^T [K_{j-1}]^{-1} [C_j] \{R^{i-1}\} - z_j \{\delta_j\} \{\delta_j\}^T \{R^{i-1}\}. \end{aligned}$$

Letting

$$\{q_{j-1}\} = [C_j] \{q_j\}$$

and

$$p_j = z_j \{\delta_j\}^T \{q_j\} \quad (3.5.17)$$

with initially

$$\{q_j\} = \{R^{i-1}\},$$

the recurrence formula is obtained as:

$$[K_j]^{-1} \{q_j\} = [C_j]^T [K_{j-1}]^{-1} \{q_{j-1}\} - p_j \{\delta_j\}$$

where

$$\{q_{j-1}\} = [C_j] \{q_j\} = \{q_j\} - p_j \{\gamma_j\}. \quad (3.5.18)$$

This formula is processed recursively for p_j and $\{q_{j-1}\}$, sweeping the index backward from j through 1. Then, the FBS operation applied to $\{q_0\}$ to obtain $\{s_0\}$, i.e.,

$$[K_0]^{-1} \{q_0\} = \{s_0\}.$$

The computational procedure can be completed by applying the following recurrence formula:

$$\begin{aligned} \{s_j\} &= [C_j] \{s_{j-1}\} - p_j \{\delta_j\} \\ &= \{s_{j-1}\} - (p_j + r_j) \{\delta_j\} \end{aligned} \quad (3.5.19)$$

with

$$r_j = z_j \{\gamma_j\}^T \{s_{j-1}\}.$$

For this procedure, the recursion is processed forward for r_j and s_j , sweeping the index from 1 to j . The end result $\{s_j\}$ represents the direction vector $\{d^i\}$.

3.5.4 One-Dimensional Example

It is instructive to examine the effects of the BFGS update in the analysis of a single degree-of-freedom system. Consider a fictitious structure that responds to the displacement with the internal force described by a parabolic function:

$$F(u) = -u^2 + 6u.$$

Suppose that the structure is in equilibrium at $u=1$ with an internal force (F) of 5. Starting from this point, the external load (P) is increased from 5 to 8, as shown in Fig. 3.5.2. The iteration process is illustrated with data points labeled 0 through 4. The first process, ending at point 1, represents a modified Newton's iteration with the initial stiffness ($K_0=6$), i.e.,

$$R^0 = P - F^0 = 8 - 5 = 3$$

$$d^1 = \frac{R^0}{K_0} = \frac{3}{6} = 0.5$$

which results in

$$u^1 = u^0 + d^1 = 1.5 \quad \text{and} \quad F^1 = 6.75.$$

If the BFGS update is performed at this point, the QN vectors will reduce to:

$$\delta_1 = u^1 - u^0 = 1.5 - 1 = 0.5$$

and

$$\gamma_1 = R^0 - R^1 = 3 - (8 - 6.75) = 1.75 .$$

Introducing the BFGS update formula in Eq. (3.5.8),

$$[K_1^*]^{-1} = z_1 \delta_1 \delta_1^T = \frac{\delta_1}{\gamma_1} = \frac{1}{3.5}$$

from which the secant stiffness at 1 (connecting points 0 and 1) is found to be 3.5. This is a significant improvement compared to the old stiffness ($K_0=6$) and the tangential stiffness (of a value 3) at point 1. The second iteration using K_1^* will end at point 2 resulting in

$$d^2 = \frac{R^1}{K_1^*} = \frac{1.25}{3.5} = 0.36$$

$$u^2 = u^1 + d^2 = 1.86$$

and

$$F^2 = 7.70 .$$

The second pair of QN vectors can now be computed:

$$\delta_2 = u^2 - u^1 = 0.36$$

and

$$\gamma_2 = R^1 - R^2 = 1.25 - (8 - 7.7) = 0.95$$

which results in a modified $[K]$ with BFGS update as:

$$[K_2^*]^{-1} = \frac{\delta_2}{\gamma_2} = \frac{0.36}{0.95} = \frac{1}{2.639} .$$

Notice that the stiffness value of 2.639 (equivalent to connecting points 1 and 2) is arrived at by recursive application of Eq. (3.5.8) with two sets of QN vectors, although the first QN vector set has no effect on the end result in this case of a single degree-of-freedom problem. From this observation, it can be deduced that the order of QN vector application is not commutative. Invoking the BFGS update criteria using Eqs. (3.5.12) and (3.5.13), values of λ^2 for the first and the second QN vectors are 1.714 and 1.316, respectively. Both of the QN vectors satisfy the criteria to be used for BFGS updates in the subsequent iterations.

Let us now examine the effects of combining the line search [3.10] with the BFGS update. To perform the line search at point 1, the relative error is estimated:

$$E_1 = \frac{d^1 R^1}{d^1 R^0} = \frac{1.25}{3} = 0.4167 .$$

If the line search is enforced by using a linear extrapolation with $E_0=1$, $\alpha_0=0$, and $\alpha_1=1$, the line search factor is found to be:

$$\begin{aligned} \alpha_2 &= \alpha_1 - E_1 \left(\frac{\alpha_1 - \alpha_0}{E_1 - E_0} \right) \\ &= 1 - \frac{0.4167}{0.4167 - 1} = 1.714 \end{aligned}$$

which results in:

$$u^{1*} = u^0 + 1.714 \times d^1 = 1.86 .$$

This point corresponds to point 2, which is the very point that was predicted by the second iteration using the BFGS update. This result occurs because the concept of line search methodology is developed based on the secant method using a single parameter, and the BFGS update is equivalent to the line search in a single degree-of-freedom model. If the BFGS update is applied at this point, the quasi-Newton vectors will be reduced to:

$$\delta_1^* = u^2 - u^0 = 0.86$$

$$\gamma_1^* = R^0 - R^2 = 2.70$$

and

$$[K_2^*]^{-1} = \frac{0.86}{1.75} = \frac{1}{3.14}$$

which indicates a secant stiffness (connecting points 0 and 2) having a value of 3.14. The next iteration will result in point 3 in Fig. 3.5.2, i.e.,

$$u^3 = 1 + \frac{0.3}{3.14} = 1.956$$

with

$$R^3 = P - F^3 = 0.09 .$$

Compared to point 2 obtained without the line search, point 3 shows a significant improvement. However, the third iteration using BFGS updates without the line search would have given a more accurate result than point 3.

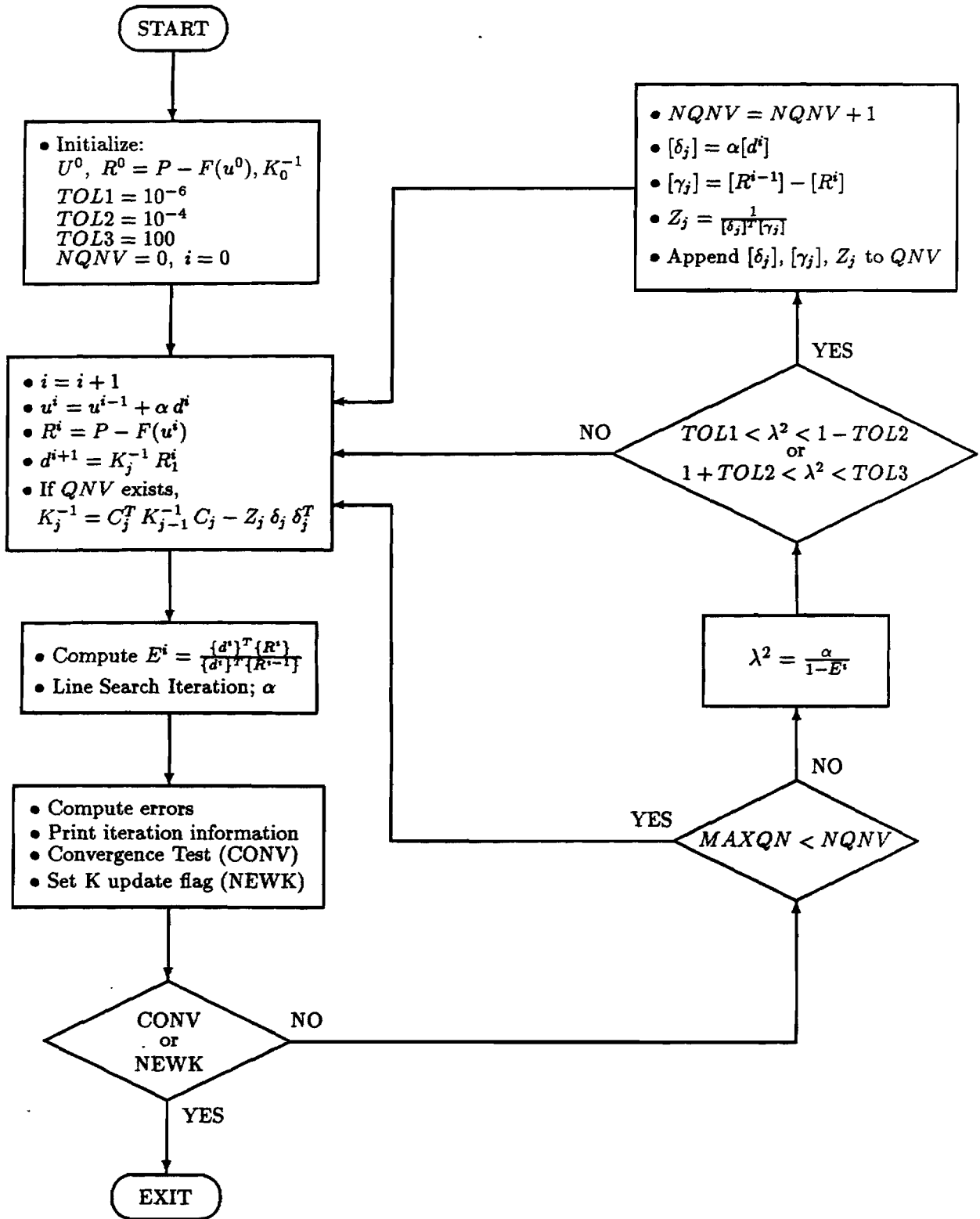


FIGURE 3.5.1. Flow Diagram for Iteration with Quasi Newton Updates.

Point	u	F	K	Derived from
0	1.0	5.0	4.0	Starting Point
1	1.5	6.75	3.0	First modified Newton iteration
2	1.86	7.70	2.29	After line search or BFGS update
3	1.956	7.91	2.088	Second iteration with LS and BFGS
4	2.0	8.0	2.0	Solution Point

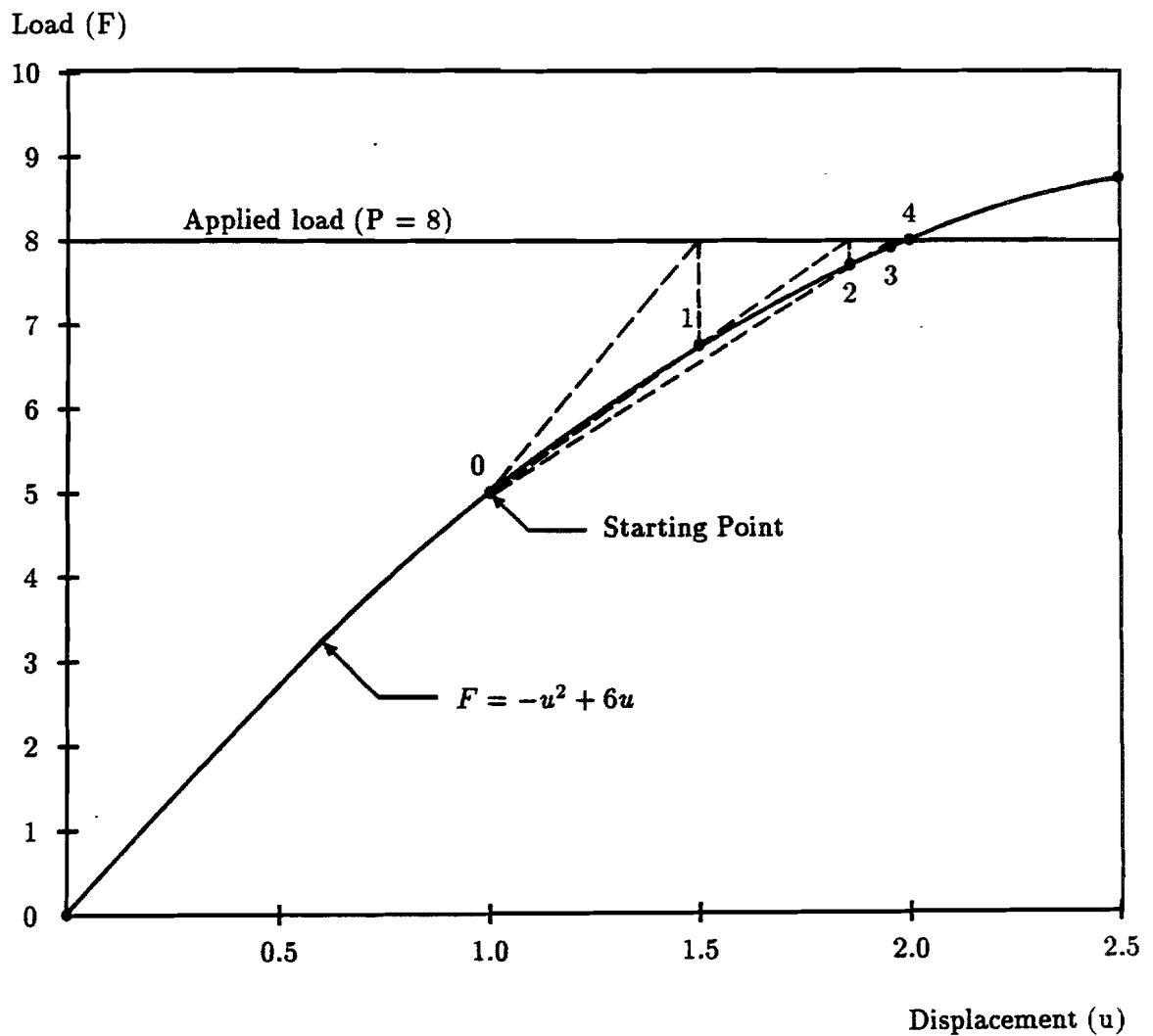


FIGURE 3.5.2. Comparison of Iteration Methods
(Quasi-Newton method with or without line search)

3.5.5 BFGS Updates for Arc-length Methods

The BFGS update is required to make the arc-length methods more effective. The BFGS updates have been implemented for the arc-length methods in Version 67. It has been proven that the BFGS update is effective and efficient for convergence, especially when the stiffness matrix is deficient. The BFGS updates compensate for the deficiency and frequently provide converged solutions in difficult problems.

The iteration equation for the arc-length method (Section 3.7.1) is shown to be:

$$\left[K - \frac{\partial P}{\partial u} \right] \Delta u = R(u, \mu) + \Delta \mu \Delta P \quad (3.5.20)$$

where the follower matrix may be omitted. The iterative process can be established by decomposing Eq. (3.5.20) into two parts:

$$K \Delta u_R = R(u, \mu) \quad (3.5.21)$$

and

$$K \Delta u_P = \Delta P \quad (3.5.22)$$

Then the trial solution is obtained by

$$u^i = u^{i-1} + \Delta u \quad (3.5.23)$$

with

$$\Delta u = \Delta u_R + \Delta \mu \Delta u_P \quad (3.5.24)$$

where $\Delta \mu$ is obtained from the constraint equation.

The residual error, R in Eq. (3.5.21), is evaluated at every iteration and Eq. (3.5.21) is solved again by FBS. However, Eq. (3.5.22) needs to be solved only once upon stiffness matrix update in the absence of the BFGS update. Two different approaches for adopting BFGS updates can be contemplated. The BFGS update may be applied to Eqs. (3.5.21) and (3.5.22). Then, FBS has to be processed on both equations at every iteration. It is also possible to adopt the BFGS update only on Eq. (3.5.21), i.e.,

$$\tilde{K} \Delta u_R = R(u, \mu) \quad (3.5.21a)$$

where \tilde{K} implies that the BFGS update is incorporated into K . The BFGS does not have to be implemented in Eq. (3.5.22), or Δu_P does not have to be reevaluated unless a new stiffness matrix is formulated. The solution should not be altered by introducing the BFGS, but the convergence will be expedited. This is because the effect of Δu_P diminishes as the solution converges ($\Delta \mu$ approaches zero).

It is necessary to modify the quasi-Newton vector γ for the arc-length method. For a constant external applied load, the quasi-Newton vector is defined as in Eq. (3.5.4). The applied load varies as a function of the load factor (μ) at every iteration in the arc-length

methods. The changes in μ during the iteration can be accommodated in the quasi-Newton vector (γ) by

$$\gamma^* = R^{i-1} - (1 - \text{NINC} * \Delta\mu^i)R^i \quad (3.5.4a)$$

where NINC is the number of increments for the subcase.

Various strategies of the BFGS updates for the arc-length methods are implemented with a control by SYSTEM(145). The NASTRAN definition at the top of the input data may include

NASTRAN SYSTEM(145) = n \$

where

n = 0 (default)	to update Δu_R and Δu_P at every iteration with γ^*
n = 1	to update Δu_R only with γ
n = 2	to update Δu_R only with γ^*

It is noted that these options are available in Version 67.

3.6 CONVERGENCE CRITERIA

The convergence test is an important factor that affects accuracy and overall efficiency in nonlinear finite element analysis. Out-of-balance forces and changes in displacements should vanish upon convergence in an iterative process. The energy error accommodates both quantities and is usually adequate for most problems. However, the displacements could be in gross error while the residual load error is negligible, or vice versa.

In order to ensure accurate and consistent convergence, multiple criteria with errors measured in terms of displacements, loads, and energy should be combined. It is the error function and the convergence tolerance that characterize the criteria. Error functions are formulated using the weighted normalization so that the error measures are dimensionless. Tolerances should be realistic for the solution scheme to be efficient. In this context, variations are considered in search of the best workable combinations of error functions and tolerances for a wide class of structural problems.

3.6.1 Rudimentary Considerations

The convergence test is a decision-making process, on which termination of the iterative process is based, while the true solution is not known. The convergence criteria are extremely important for the incremental/iterative solution strategy to be effective and efficient, because improper criteria could cause inefficiency as well as inaccuracy. It is rather astounding to find a scarcity of publications on this subject, considering the significant impact of the convergence criteria on the accuracy and the efficiency of the computation. Two distinct aspects are involved in the convergence criteria:

- Error functions to be minimized by the iteration
- Tolerances of error functions within which errors are acceptable.

Both aspects must be defined properly for the criteria to be effective, for the solution scheme to be efficient, and for the solution to be accurate.

A literature search [3.7, 3.8, 3.12, 3.13] has revealed that there are no universally accepted convergence criteria to date in the field of finite element analysis. Conditions to be met by ideal convergence criteria for a general-purpose finite element analysis have been contemplated. The convergence criteria should:

- be satisfied for linear cases at all times;
- be independent of structural units;
- be reliable (cancellation of errors are not acceptable);
- render consistent accuracy;
- be independent of structural characteristics (stiffening or softening);

- be able to handle all the loading cases including constant loading, unloading, and no external loading (applicable to creep analysis);
- have smooth transitions after the stiffness updates and loading changes.

These conditions dictate the formulation of error functions to be discussed [3.14].

3.6.2 Convergence Conditions

The iteration continues until the convergence is attained by satisfying the convergence criteria and the residual error vector at convergence is carried over to the next incremental step. When the convergence criteria are satisfied, the out-of-balance forces and the changes in displacements should be sufficiently small so that the remaining error is not physically significant nor will it cause any detrimental effects, numerically or physically, on the succeeding incremental steps. Convergence tolerances have the following effects:

- excessively tight tolerances cause a waste of computing resources for unnecessary accuracy;
- excessively loose tolerances cause not only inaccuracy but convergence difficulties in the subsequent steps due to cumulative errors.

The fundamental difficulty of the convergence tests for a structural analysis lies in the fact that the base vectors (forces and displacements) involve inconsistent units, namely, combinations of forces and moments or translations and rotations. Indiscriminate use of these vectors will cause unit-dependent convergence criteria. For example, while an error in forces is dominant when the model is expressed in newton-meter, the error would be dominated by moments if the same model is described in newton-millimeter.

The most natural and reasonable criterion for the convergence test is formulated in terms of an energy error. The energy error is the logical choice because both the out-of-balance forces $\{R\}$ and the change in displacements $\{\Delta u\}$ should be minimized by the iteration process. Furthermore, energy quantities do not pose problems of inconsistent units due to mixed units associated with translations and rotations.

Although the convergence test in terms of energy errors is usually adequate, some distinct errors are not detected with this criterion; i.e., displacements are in gross error while the residual load error is negligible, or vice versa. This would be the case if the degrees-of-freedom in error have a very small or a very large stiffness. Such cases compel the need for criteria in terms of loads and displacements. Nominally, by visualizing the load-deflection curve for a one-dimensional case, it can be noticed that the convergence criterion in terms of loads governs the stiffening structure and the criterion in terms of displacements governs the softening structure. Scalar error functions for these criteria are formulated to be dimensionless by introducing the weighted normalization.

3.6.3 Error Functions and Weighted Normalization

All the error functions are defined as relative errors in terms of vector norms. The vector norm is defined by a sum of absolute values of the components (l_1 norm or $\| \cdot \|_1$), where subscript 1's are omitted for simplicity in the following development. The vectors used in these norms, however, are not raw vectors but weighted vectors using other vector quantities as weighting functions. The weighting functions are chosen so that the products have common and consistent structural units.

To define a relative error in terms of energy (or work) which is printed at every iteration under the heading EWI, the energy error is normalized by another energy quantity (E^*) that is relatively stable throughout the analysis, i.e.,

$$E_w = \frac{\| R * \Delta u \|}{E^*} \quad (3.6.1)$$

with

$$\| R * \Delta u \| = \sum_j \text{ABS}(R_j * \Delta u_j) \quad (3.6.2)$$

$$E^* = \| P' * u \| = \sum_j \text{ABS}(P'_j * u_j) \quad (3.6.3)$$

and

$$\{P'\} = \{|\Delta P|\} + \{P\} \quad (3.6.4)$$

where $\{\Delta P\}$ represents the incremental load (including increments in reaction forces due to enforced displacements) and $\{P\}$ pertains to the total load at the previous loading step; and $\{u\}$ and $\{\Delta u\}$ represent the total and incremental displacements at the current iteration. Norms as defined in Eqs. (3.6.2) and (3.6.3) are not dot products and should be viewed as weighted vector norms. They are devised to prevent cancellation of errors, based on the concept of the Cauchy-Schwarz in equality. Notice that smooth transitions between incremental steps upon load changes are attempted by defining an absolute sum for $\{P'\}$ as in Eq. (3.6.4), utilizing the triangle inequality. Thus the normalization factor does not vanish unless the incremental load as well as the total load disappear simultaneously.

The error function in terms of loads (printed at every iteration under the heading EPI) is made dimensionless by a weighted normalization using the total displacement vector, i.e.,

$$E_p = \frac{\| R * u \|}{\| P' * u \|} \quad (3.6.5)$$

in which the numerator and denominator are defined respectively as in Eqs. (3.6.2) and (3.6.3). Notice that the current displacement vector is used as a weighting function; thus, the energy units resulting from the products are common and cancel each other. Cancellation of errors due to orthogonality is thereby prevented effectively.

In order to formulate an error function in terms of displacements, the contraction factor (q) of $\{\Delta u\}$ should be introduced because the error in displacements is not known. Assuming

a geometric progression of the displacement increments, the contraction factor is defined as a ratio of displacement increments between two successive steps, i.e.,

$$q = \frac{\|u^{i+1} - u^i\|}{\|u^i - u^{i-1}\|} = \frac{\|\Delta u^{i+1}\|}{\|\Delta u^i\|} \quad (3.6.6)$$

where i is an iteration index. If the contraction factor is assumed to remain constant with a value less than unity, the upper bound of the absolute error in displacements can be estimated. For all positive integers n , we have

$$\begin{aligned} \|u - u^i\| &\leq \|u - u^{n+1}\| + \|u^{n+1} - u^{n+i-1}\| + \dots \|u^{i+1} - u^i\| \\ &= \|\Delta u^i\| (q^n + q^{n-1} + \dots + q) \end{aligned}$$

where q_n denotes the n -th power of q . Taking the limit as n approaches infinity, the above expression is reduced to:

$$\|u - u^i\| \leq \|\Delta u^i\| \frac{q}{1 - q} \quad (3.6.7)$$

The convergence criterion in terms of displacements is not accurate, as is obvious from the derivation of Eq. (3.6.7) with many assumptions. Contrary to the assumption, the contraction factor may be ill-conditioned and does indeed fluctuate erratically. An averaging scheme is employed to smooth a fluctuating q . After trial and error, we adopted:

$$q^i = \frac{2}{3} \frac{\|\Delta u^i\|}{\|\Delta u^{i-1}\|} + \frac{1}{3} q^{i-1} \quad (3.6.8)$$

with an initial value $q^1 = 0.99$.

The error function in displacements (printed at every iteration under the heading EUI) is made dimensionless by a weighted normalization using the stiffness matrix, i.e.,

$$E_u = \frac{q}{1 - q} \frac{\|w * \Delta u\|}{\|w * u\|} \quad (3.6.9)$$

in which the weighting function $\{w\}$ is formed by collecting the square root of the diagonal terms of $[K]$, i.e.,

$$w_j = \sqrt{K_{jj}} \quad (3.6.10)$$

and the norms of the weighted displacements are defined similarly to Eq. (3.6.2). All the product terms have the same unit of the square root of the energy, which implies more equitable participation of all terms in the error estimation. Note that the contraction factor is not used for the divergence check because of its erratic and unreliable behavior.

An alternative approach has been considered to formulate error functions in loads and displacements using a characteristic length specified by the user. The weighting functions in this case will be formed using a mapping vector $\{L\}$ consisting of 1's for the translational degree-of-freedom and the specified characteristic length for the rotational degree-of-freedom. The vector $\{L\}$ and its reciprocal, denoted by $\{L'\}$ are used as weighting functions for errors

in displacements and loads, respectively. Then the error functions in terms of loads and displacements are defined similarly to Eqs. (3.6.5) and (3.6.9) using $\{L'\}$ and $\{L\}$, respectively. With this approach, the units are unified and errors become dimensionless. This method is intuitively appealing from an engineering perspective, because the weighting functions are known functions and the errors are measured in terms of physical units of length and force. However, it may be difficult or cumbersome to determine what the characteristic length should be.

3.6.4 Implementation

The convergence tolerance determines the efficiency of the solution scheme as well as the accuracy of the solution. The tolerance should be realistic, not too tight nor too loose. It is difficult to choose optimal default values for the convergence tolerances. However, efforts have been made to set the default values to provide reliable solutions to the general class of problems. Thus, default tolerances should be adhered to until good reasons are found to change them.

The convergence test is performed at every iteration after the line search process (if applicable). Three error functions (in terms of displacements, loads, and energy) are computed and compared to tolerances, i.e.,

$$E_u < \text{EPSU} (= 10^{-3} \text{ by default})$$

$$E_p < \text{EPSP} (= 10^{-3} \text{ by default})$$

$$E_w < \text{EPSW} (= 10^{-7} \text{ by default})$$

where EPSU, EPSP, and EPSW are tolerances specified in the NLPARM entry. However, only those criteria chosen by the user (combinations of U,P, and/or W) are designed to be satisfied for convergence. The default values have been determined as a result of numerous tests on 47 problems with a wide variety of characteristics and options. The goal of the default tolerances is to consistently provide sufficiently accurate solutions to a wide spectrum of problems without sacrificing efficiency.

The solution should converge in a single iteration if the structural response is linear. The default option is designed to comply with this requirement. However, if the user inadvertently specifies an undesirable condition such as an extremely small tolerance, the solution to the linear case should still converge regardless of the convergence criteria. Thus, the absolute convergence criterion is designed for this purpose using the error function in terms of the load error, i.e.,

$$E_p < 10^{-7} * \text{EPSP} (= 10^{-10} \text{ by default}).$$

It is noted that divergence conditions are established independent of convergence criteria.

3.6.5 Some Observations

Some objectives of the ideal convergence criteria have not been fully met by the criteria as currently implemented. In view of Eq. (3.6.3), errors E_w and E_p are amplified when the load increment is decreased. Therefore, the reduced load increment effectively results in a tightened convergence test. Without $\{\Delta P\}$ in Eq. (3.6.4), however, the convergence tests will fail in the unloading cases. Without $\{P\}$ in Eq. (3.6.4), the convergence tests will fail in the constant loading cases as in a creep analysis. Convergence tests on E_w and E_p will fail if no external loads are applied in more than two consecutive loading cases, resulting in a null vector for $\{P'\}$ in Eq. (3.6.4). This is the case when the creep relaxation is analyzed under residual stresses. Such difficulties can be avoided by introducing a fictitious load to a disjoint nodal point in the model. Convergence tests can be, in effect, bypassed by assigning large values to the tolerances.

The error in displacements is not an appropriate measure for the first iteration because of an arbitrarily assigned initial value for the contraction factor in Eq. (3.6.8). Another potential problem exists in the displacement error for using the stiffness matrix as a weighting function. With the modified Newton method, this approach uses an old stiffness matrix while the tangential stiffness may change significantly, which can cause improper weight distribution. Numerical experimentation proves, however, that most problems do not require convergence criterion in Eq. (3.6.9) if criteria in Eqs. (3.6.1) and (3.6.5) are satisfied. Introducing divergence criteria independent of the convergence rate is proven effective because of the erratic nature of convergence behavior. The divergence criteria play an important role in trapping the run-away solution. Two categories of divergence (absolute and probable) help to handle cases with different structural characteristics. The probable divergence is designed to expedite the convergence of the stiffening structure, for instance, as a complement to the usual divergence classified as absolute divergence.

Due to the nature of the subject, the performance measure in terms of convergence test is rather qualitative and heuristic than quantitative. Admitting that the test data are not available in a systematic and statistical form, the results presented here are derived from testing and tuning of parameters and tolerances with fifty nonlinear problems as described in Tables 3.6.1 and 3.6.2. The work was carried out during the development of MSC.NASTRAN® (MSC.Nastran is a registered trademark of MSC.Software Corporation) Version 65, and tests were conducted in comparison with Version 64. As illustrated in table 3.6.2, the current convergence criteria combined with line search [3.10] and BFGS updates[3.11] improved the convergence characteristics remarkably. Notice that 11 problems out of 50, which failed to converge in Version 64, have been analyzed to completion successfully in Version 65. The significant contribution of the current convergence criteria to the overall performance of the program is demonstrated in providing reliable solutions with consistent accuracy for all the converged solutions, covering a wide spectrum of problems. This effect is pronounced in three problems which gave erroneous solutions in Version 64 and ran correctly in Version 65

Table 3.6.1 Test Problem Description

1. BBUCKL	- Beam - for Euler buckling
2. BCTEST	- Boundary condition test
3. BEAMZ	- 10-BEAM model to test SMALLDB - reduced from ZBEAM for short run
4. BMLIFT	- Beam with Gap/Lift-off
5. CBEAM	- Cantilever beam with boundary condition changes
6. CLUTCH	- Electro-mechanical clutch
7. CRDEMOS	- Static analysis of nozzle to sphere attachment
8. CSR3065	- Gap error with friction
9. CTESHP	- 1 HEXA and 2 PENTA elements to test plastic material
10. CTHEXA	- 1 HEXA element to test plastic material
11. CTPENTA	- 2 PENTA elements to test plastic material
12. CTSHELL	- A single QUAD4 to test plastic material, loading and unloading
13. DRUM	- Drum head model with thermal load
14. FOLFO	- Follower force test problem with CRDEMO
15. GAPS	- Gap elements - opening and closing in sequence
16. GAPTST	- Gap test with electro-mechanical clutch
17. GOODRICH	- 2 layers of HEXA's to model bi-metal
18. HAMMER	- Cantilevered rod hit by a mass with initial velocity specified - mass connected to rod by gap
19. HDRULE	- Kinematic hardening rule test - tension,unloading, compression
20. LGDROD	- 2 DOF model with thermal load
21. LGDTEST	- 3 shell elements to test LGDISP
22. MNEWTON	- Modified Newton method with line search on RODTEST
23. NLCTIRE	- Tire model with composite element and large displacement effects
24. NLELAS	- Nonlinear-elastic material with different tension- compression properties
25. NLELAS2	- Shell elements with bilateral nonlinear-elastic material
26. NLELAS3	- Nonlinear-elastic HEXA with hydrostatic load
27. NLPATH1	- Non-proportional loading specified by strain path
28. NLPATH2	- Non-proportional loading specified by stress path
29. NLROD	- 1-D plasticity in rod
30. NLSHELL	- Imperfect spherical shell with shell element
31. NLSOLID	- Imperfect spherical shell with solid model, includes buckling analysis
32. RODTEST	- 2 DOF test problem for BFGS method with line search
33. SBUCKL	- NLSHELL - for buckling analysis
34. SE66S1	- Superelement model of two tips plus a residual(TRIA3) in sort 1
35. SE66S2	- Superelement model of two tips plus a residual(QUAD4) in sort 2
36. SFRAME	- Curved box-beam test for large displacements
37. SHLBEAM	- Cantilevered beam with shell elements
38. SNAPS	- Snap-thru of a spherical dome
39. SUPLCOP	- Non-superelement model of two tips plus a residual
40. SWAVE	- Cantilevered rod with a step forcing function at the end, testing stress wave
41. TCRP	- Creep of plate under thermal load
42. TRIPOD	- Basic nonlinear test with rod elements
43. UNIAX1P	- Creep at constant stress
44. UNIAX2P	- Elastoplastic-creep, reproducing isochronous curve
45. UNIAX2R	- Code test for creep - relaxation prediction
46. UNIAX2T	- Creep under variable temperature
47. UNIAX4	- Effects of stress reversal
48. VGAP	- Gap verification with nonlinear transient analysis
49. ZBEAM	- Z-shaped beam analysis with 10 BEAM elements
50. APLATE	- Z-shaped beam analysis with 10 QUAD4 elements

Table 3.6.2 Test Results Comparision : V64 vs. V65

Problem Name	No.Iters		No.K Updates		No. L.S.		Remarks
	V64	V65	V64	V65	V64	V65	
BBUCKL	5	5	0	0	0	0	
BBUCKL(R)	2	2	0	0	0	0	
BCTEST	26	22	0	0	0	0	
BEAMZ	-	30	-	6	-	0	
BEAMZ(R)	-	10	-	2	-	0	
BMLIFT	2	2	2	2	0	0	
CBEAM	6	6	0	0	0	0	
CLUTCH	9+	54	3	12	14	56	1
CLUTCH(R)	-	141	-	6	-	8	
CRDEMOS	14	16	0	0	9	1	2
CSR3065	8+	5	8	5	0	0	3
CTESHP	5+	15	2	2	0	3	
CTHEXA	5+	15	3	3	0	0	
CTPENTA	12+	9	3	1	10	4	
CTSHELL	15	23	2	0	7	0	
CTSHELL(R)	3	3	0	0	0	0	
DRUM	-	36+	-	3	-	31	
DRUM(R)	-	15	-	2	-	10	
FOLFO	12	9	0	0	0	0	
GAPS	39	26	3	1	20	44	4
GAPTST	29	24	3	1	38	34	5
GOODRICH	11+	155	3	14	24	20	
HAMMER	-	-	-	-	-	-	6
HDRULE	21+	15	9	2	0	6	
HDRULE(R)	13	5	6	1	0	1	7
LGDRD	51	67	17	3	63	31	8
LGDTST	17+	40+	3	5	0	36	9
LGDTST	4+	15+	1	1	0	5	
MNEWTON	4+	79	1	15	0	18	
NLCTIRE	13	39	6	3	5	5	10
NLELAS	-	35	-	1	-	8	
NLELAS2	16+	40	0	1	12	7	
NLELAS3	-	11	-	0	-	1	
NLPATH1	25	20	4	4	0	0	
NLPATH2	35	28	9	4	0	3	
NLROD	10	9	3	3	0	0	
NLSHELL	78	68	13	4	13	0	
NLSOLD	77+	74+	21	11	65	20	11
NLSOLID(R)	-	15	-	3	-	4	12
RODTEST	35(+)	47(46)	14	12(9)	81	29(30)	13
SBUCKL	51	60	13	13	34	6	
SBUCKL(R)	10	11	2	2	8	0	
SE66S1	12	13	2	2	0	0	
SE66S2	11	13	0	0	5	0	
SFRAME	54+	71+	19	15	49	13	14
SFRAME(R)	-	7+	-	3	-	5	
SHLBEAM	55+	82	2	0	7	4	
SHLBEAM(R)	60	41	6	4	0	0	
SNAPS	36	40	5	2	25	15	
SUPLCOP	9	10	0	0	3	0	
SWAVE	-	-	-	-	-	-	6
TCRP	14	13	1	0	0	0	
TRIPOD	14	22	2	1	26	14	15
UNIAX1P	20	13	4	2	0	3	
UNIAX2P	75	89	5	3	0	21	
UNIAX2R	31	46	0	0	14	0	15
UNIAX2T	146	179	3	2	0	0	
UNIAX4	31	37	5	4	0	0	
VGAP	22	22	4	4	0	0	6
ZBEAM	267+	1907	54	200	358	310	16
ZPLATE	822	2621	292	335	940	291	16

(R) : Restart run, + : Diverging solution, - : V64 unavailable for this analysis

3.7 ARC-LENGTH METHODS FOR POST-BUCKLING ANALYSIS

The static nonlinear response of a structure can not be analyzed beyond the critical limit load by the conventional Newton's method. Although the post-buckling state is not usually allowed in the structural design, the prediction of such response could be of interest in some cases. In the design process, for instance, it may be desirable to trace the response of the snap-through or post-buckling behavior. The arc-length methods allow solutions in the unstable regime for such class of problems. It is noted that they are not efficient solution methods, but effective methods to render solutions in the unstable regime which is not necessarily a physically viable state. An arc-length method, known as Crisfield's method, was implemented in Version 66. However, some convergence difficulties have been identified and resolved in Version 66A. In addition, a number of options for arc-length methods and the adaptive bisection algorithm have been implemented in Version 67.

3.7.1 Basic Theory for Arc-Length Method

The concept of the arc-length method is to modulate the applied loads in order to produce solutions with displacement increments of manageable size for a given load step. In order to modulate the applied load, an additional variable (the load factor) and a constraint equation are introduced. There are various approaches to providing a constraint equation [3.15-3.18].

Consider a residual load $\{R\}$

$$R(u, \mu) = P(u, \mu) - F(u) \quad (3.7.1)$$

where F represents the internal forces, and the total external load P is expressed as

$$P(u, \mu) = P_0 + \mu \Delta P \quad (3.7.2)$$

where P_0 denotes the applied load at the end of the preceding subcase, ΔP represents the load increment in the current subcase, and μ is the load factor varying from 0 to 1 (but not to be limited to this range) within the subcase. Linearizing $\{R\}$ about (u, μ) , Eq. (3.7.1) can be expressed as

$$R(u + \Delta u, \mu + \Delta \mu) \simeq R(u, \mu) + \frac{\partial R}{\partial u} \Delta u + \frac{\partial R}{\partial \mu} \Delta \mu \quad (3.7.3)$$

Based on Eqs. (3.7.1) and (3.7.3), the equilibrium condition at $(u + \Delta u, \mu + \Delta \mu)$ dictates that

$$R(u, \mu) + \frac{\partial P}{\partial u} \Delta u - \frac{\partial F}{\partial u} \Delta u + \frac{\partial P}{\partial \mu} \Delta \mu = 0 \quad (3.7.4)$$

where $[\frac{\partial P}{\partial u}]$ is the follower matrix, $[\frac{\partial F}{\partial u}]$ is the stiffness matrix $[K]$, and $\frac{\partial P}{\partial \mu} = \Delta P$.

The iteration equation could be derived by rearranging Eq. (3.7.4):

$$[K - \frac{\partial P}{\partial u}] \Delta u = R(u, \mu) + \Delta \mu \Delta P \quad (3.7.5)$$

where the follower matrix may be omitted. The iterative process can be established by decomposing Eq. (3.7.5) into two parts:

$$K \Delta u_R = R(u, \mu) \quad \text{and} \quad K \Delta u_P = \Delta P \quad (3.7.6)$$

Then the trial solution is obtained by

$$u^i = u^{i-1} + \Delta u \quad (3.7.7)$$

with

$$\Delta u = \Delta u_R + \Delta \mu \Delta u_P \quad (3.7.8)$$

where $\Delta \mu$ can be obtained from the constraint equation.

3.7.2 Riks' Method and Its Variations

The displacement increment is limited by a constraint equation:

$$\{\Delta u^1\}^T \{\Delta u^1\} + (w \Delta \mu^1)^2 = \Delta \ell^2 \quad (3.7.9)$$

where w is a user-specified scaling factor (to be specified in NLPCI) and $\Delta \ell$ is defined by

$$\Delta \ell^2 = (\Delta \mu^1)^2 [\{\Delta u_P^1\}^T \{\Delta u_P^1\} + w^2]$$

if the initial value of $\Delta \mu$ is given. The constraint of Eq. (3.7.9) has a disparity in the dimension by mixing the displacements with the load factor. For this reason, the scaling factor (w) is introduced so that the user can scale μ to the appropriate dimension or delete the $\Delta \mu$ term. The default value of w is zero (Fig. 3.7.3b). The iteration follows the path on the plane normal to the initial tangent as shown in Fig. 3.7.1. Therefore the subsequent iterations ($i > 1$) must satisfy

$$\{\Delta u^1\}^T \{\Delta u^i\} + w^2 \Delta \mu^1 * \Delta \mu^i = 0 \quad (3.7.10)$$

Recalling that the first iteration should result in

$$\{\Delta u^1\} = \Delta \mu^1 \{\Delta u_P^1\},$$

Eq. (3.7.10) may be reduced to

$$\{\Delta u_P^1\}^T \{\Delta u_R^i + \Delta \mu^i \Delta u_P^i\} + w^2 \Delta \mu^i = 0$$

from which the load factors for the subsequent iterations are determined by

$$\Delta \mu^i = \frac{-\{\Delta u_P^1\}^T \{\Delta u_R^i\}}{w^2 + \{\Delta u_P^1\}^T \{\Delta u_P^i\}} \quad (3.7.11)$$

and

$$\mu^i = \mu^{i-1} + \Delta\mu^i$$

Notice that the normal plane does not change during the iteration by Riks' method. In addition, $\{\Delta u_P\}$ remains constant if the iteration process is the modified Newton's method.

Alternatively, the normal plane may be updated at every iteration. If the normal plane is to be normal to the cumulative incremental displacements for the preceding iterations as shown in Fig. 3.7.2, the orthogonality condition in Eq. (3.7.10) should be modified to:

$$\{u^{i-1} - u^0\}^T \{\Delta u^i\} + w^2(\mu^{i-1} - \mu^0)\Delta\mu^i = 0 \quad (3.7.10b)$$

The increment in the load factor for $i > 1$ is obtained by solving Eq. (3.7.10b), i.e.,

$$\Delta\mu^i = \frac{-\{u^{i-1} - u^0\}^T \{\Delta u_R^i\}}{w^2(\mu^{i-1} - \mu^0) + \{u^{i-1} - u^0\}^T \{\Delta u_P^i\}} \quad (3.7.11b)$$

This variation of Riks' method has an advantage over Crisfield's method as it avoids the solution of a quadratic equation.

3.7.3 Crisfield's Method

Instead of iterating on the normal plane, the solution is sought on the surface defined by Eq. (3.7.9) with an arc-length of $\Delta\ell$ as depicted in Fig. 3.7.3a, i.e.,

$$\{u^i - u^0\}^T \{u^i - u^0\} + w^2(\mu^i - \mu^0)^2 = \Delta\ell^2 \quad (3.7.12)$$

This constraint can be interpreted as keeping the incremental displacement constant, if $w=0$, as shown in Fig. 3.7.3b. Substituting Eq. (3.7.8) into the preceding equation, we obtain a quadratic equation in terms of $\Delta\mu$:

$$a(\Delta\mu^i)^2 + b\Delta\mu^i + c = 0 \quad (3.7.13)$$

where

$$\begin{aligned} a &= \{\Delta u_P^i\}^T \{\Delta u_P^i\} + w^2 \\ b &= 2\{\Delta u_P^i\}^T \{(u^{i-1} - u^0) + \Delta u_R^i\} + 2w^2(\mu^{i-1} - \mu^0) \\ c &= \{(u^{i-1} - u^0) + \Delta u_R^i\}^T \{(u^{i-1} - u^0) + \Delta u_R^i\} + w^2(\mu^{i-1} - \mu^0)^2 - \Delta\ell^2 \end{aligned}$$

Since Crisfield's method leads to a quadratic equation, the selection of the proper root of this equation becomes the most critical process for the success of this method. There are two roots to Eq. (3.7.13), i.e.,

$$\Delta\mu^i = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (3.7.14)$$

The root is chosen so that the angle between two vectors $\{u^{i-1} - u^0\}$ and $\{u^i - u^0\}$ is less than 90 degrees, i.e.,

$$\begin{aligned}\cos \theta &= \frac{\{u^i - u^0\}^T \{u^{i-1} - u^0\}}{\Delta \ell^2} \\ &= 1 + \frac{1}{\Delta \ell^2} \{u^{i-1} - u^0\}^T \{\Delta u^i\} > 0\end{aligned}$$

There are cases where no roots can be found. Such is the case when the trial solution is far from the true solution and stays outside the region covered by the arc-length. In this case, the trial solution vector is scaled so that the direction vector intersects with the surface defined by Eq. (3.7.12).

The wrong choice of the root could cause an unintentional loading path reversal, by which the solution returns to the previous state. Such cases can be detected by checking the orthogonality of the incremental displacements of the two successive solutions. If this case is detected, the root is chosen so that the angle between $\{u^1 - u^0\}$ and $\{u^i - u^0\}$ is an acute angle.

3.7.4 Adaptive Arc-Length Method

It would be difficult for a user to estimate a proper arc-length for multi-degree-of-freedom problems. The initial arc-length for the Crisfield method can be determined by

$$\Delta \ell_0^2 = (\Delta \mu^1)^2 [\{\Delta u_P^1\}^T \{\Delta u_P^1\} + w^2] \quad (3.7.15)$$

with

$$\Delta \mu^1 = \mu_1 = \frac{1}{\text{NINC}}$$

Notice that NINC is a user-specified value for the number of increments in the subcase.

The arc-length should be continuously updated at every increment using the information gathered during the preceding increment. One method is to reduce the arc-length if it requires an excessive number of iterations to attain a converged solution, i.e.,

$$\Delta \ell_{new} = \sqrt{\frac{I_d}{I_{max}}} \Delta \ell_{old} \quad (3.7.16)$$

where I_d is the desired number of iterations for convergence and I_{max} is the number of iterations required for convergence at the preceding step. It is also desirable to decrease the arc-length if the stiffness ratio increases, i.e.,

$$\Delta \ell_{new} = \frac{1}{K_{ratio}} \Delta \ell_{old} \quad (3.7.17)$$

with

$$K_{ratio} = \text{ABS} \left[\frac{K_{new}}{K_{old}} \right] = \text{ABS} \left[\frac{\{u_{j-1} - u_{j-2}\}^T \{F_j - F_{j-1}\}}{\{u_j - u_{j-1}\}^T \{F_{j-1} - F_{j-2}\}} \right] \quad (3.7.18)$$

where the subscripts ($j, j-1, j-2$) denote the converged solution steps. In terms of the arc-length, the computation is simplified as

$$K_{ratio} = \text{ABS} \left[\frac{\Delta\mu_n * \Delta\ell_{n-1}}{\Delta\mu_{n-1} * \Delta\ell_n} \right]$$

The adaptive process should be based on the arc-length ratio, i.e.,

$$\text{ALRATIO} = \frac{\Delta\ell_{new}}{\Delta\ell_{old}}$$

Combining two criteria, the new arc-length ratio may be adapted to the nonlinearity by

$$\begin{aligned} \text{If } \left(\sqrt{\frac{I_d}{I_{max}}} \text{ and } \frac{1}{K_{ratio}} \cdot GT.1. \right) \text{ then } \text{ALRATIO} &= \text{MIN} \left(\sqrt{\frac{I_d}{I_{max}}}, \frac{1}{K_{ratio}} \right) \\ \text{If } \left(\sqrt{\frac{I_d}{I_{max}}} \text{ and } \frac{1}{K_{ratio}} \cdot LT.1. \right) \text{ then } \text{ALRATIO} &= \text{MAX} \left(\sqrt{\frac{I_d}{I_{max}}}, \frac{1}{K_{ratio}} \right) \\ \text{else } \text{ALRATIO} &= 1. \end{aligned}$$

In order to maintain the stability for the adaptive process, ALRATIO should also be bounded, i.e.,

$$\text{MINALR} < \text{ALRATIO} < \text{MAXALR} \quad (3.7.19)$$

where MINALR and MAXALR can be user-specified parameters, defaulted to 0.25 and 4., respectively. If the adjusted ALRATIO falls outside the bounds, ALRATIO should be reset to the limit. Then the arc-length can be updated at the beginning of the next step based on ALRATIO as follows:

$$\Delta\ell_{new} = \text{ALRATIO} * \Delta\ell_{old}$$

If the iteration produces μ^i which exceeds the value of μ for output points, $\Delta\mu^i$ should be adjusted to the specified value of μ for the output, i.e.,

$$\Delta\mu^* = \text{MIN}(\Delta\mu, \mu^* - \mu^{i-1}) \quad (3.7.20)$$

and

$$\Delta u^* = \Delta u_R^i + \Delta\mu^* \Delta u_P^i$$

where μ^* is the specified value for output (end of subcase or intermediate points). In the stiffening case this limit may be exceeded at the first increment of the subcase. This limiting process may be implemented as follows:

Process Eq. (3.7.20) at every iteration with

$$\mu^* = \Sigma \frac{1}{\text{NINC}}$$

if INTOUT=ALL, otherwise $\mu^* = 1$.

Then if

$$\| u^i + \Delta u^* - u^0 \| < \Delta \ell$$

$$\Delta \mu^i = \Delta \mu^* \text{ and } \Delta u^i = \Delta u^*$$

otherwise ignore $\Delta \mu^*$ and Δu^* and proceed to update u^i and μ^i .

The arc-length should be updated upon convergence whenever this limiting process takes place. Notice that this limiting process in effect reduces the arc-length method to the conventional Newton's iteration.

In the unstable regime where the stiffness is negative, the load factor decreases with a forward step. When this happens, the sign of $\Delta \mu^1$ should be reversed. This possibility should be examined at the beginning of each increment. The sign can be determined by the sign of a dot product, i.e.,

$$\Delta \mu_j^1 = -(\mu_j - \mu_{j-1}) \quad \text{if} \quad \{u_{j-1} - u_{j-2}\}^T \{\Delta u_P^1\} < 0 \quad (3.7.21)$$

An adaptive bisection algorithm is also incorporated to cope with divergent cases. If the iterative process using the arc-length method tends to diverge, the arc-length is bisected. The bisection is combined in concert with the stiffness matrix update strategy. The bisection procedure will continue until the iterative process is stabilized and a converged solution is found. However, the number of contiguous bisections is limited by a user-specified parameter, MAXBIS in the NLPARM Bulk Data entry. The variable arc-length at every increment invokes the recovery from the bisection process once the difficulties in convergence are overcome.

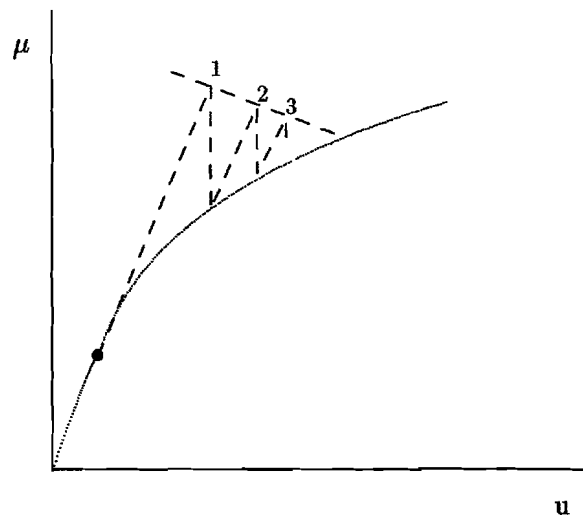


Figure 3.7.1. Riks' Method

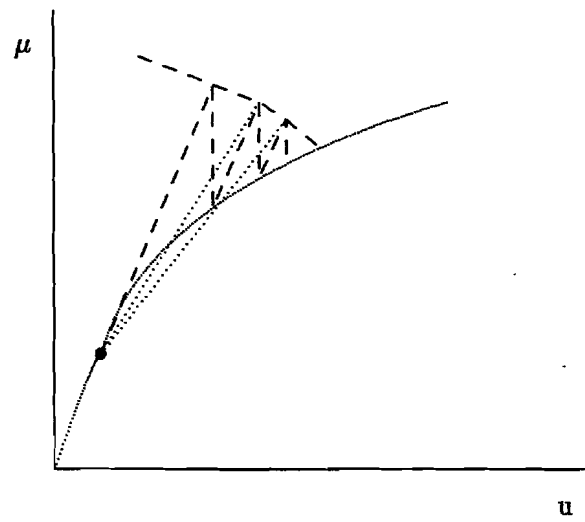
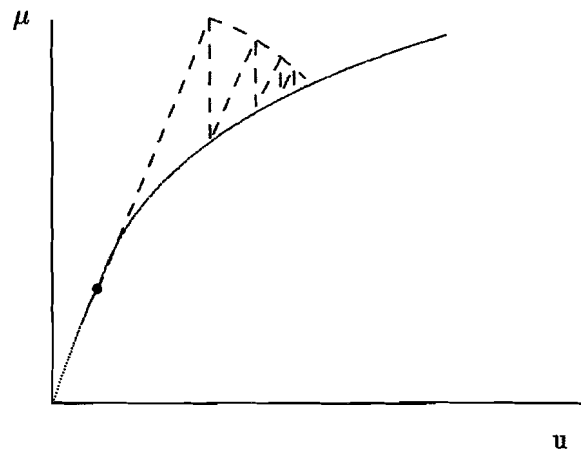
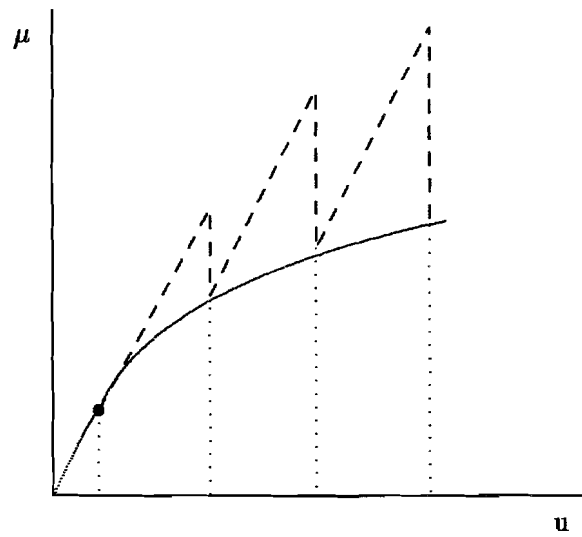


Figure 3.7.2. Modified Riks' Method



(a) Arc-length in terms of Combined Variables



(b) Arc-length in terms of Displacements

Figure 3.7.3. Crisfield's Methods.

3.7.5 Verification Problems

The most notable applications of the arc-length method are analyses of the snap-through behavior and the post-buckling behavior. Such examples are illustrated below.

SNAP-THROUGH AND SNAP-BACK: A SPHERICAL SHELL WITH BACKING PLATE

A spherical diaphragm resting on a rigid backing plate is subjected to a uniform pressure. The analysis objective is to find the structural response for snap-through and snap-back under the clamped boundary condition. Additional constraints are applied after the snap-through by the backing plate. The material is linear elastic, and the deformation pattern is assumed to be axisymmetric because the diaphragm is connected to a plunger which prevents rotation of the apex.

A 10-degree sector of the diaphragm was modeled using shell elements with axisymmetric boundary conditions. The contact process between the diaphragm and the backing plate was simulated by gap elements as shown in Fig. 3.7.4. The main features of these problems were geometric nonlinearities due to large displacements, follower forces, and changes in constraints due to the backing plate. This model was analyzed by Crisfield's method and the results are compared to those obtained by the conventional Newton's method [3.10]. In the latter, the convergence behavior was rather erratic and unpredictable around the critical loads. In addition, since Newton's method is unable to trace the structural response throughout the snap-through process, the obtained solutions were discontinuous. The missing solution portions were conjectured and represented by dashed lines.

The load-deflection curve at the apex point, constructed by combining loading and unloading paths, is shown in Fig. 3.7.5. The snap-through occurs at around 0.97 psi, and the corresponding displacement is 0.18 in. at the apex point. During the snap-through process (between points A and B), the displacement increased with a decreasing load until the stiffness became positive again when a tensile stress started to build up in the membrane direction of the shell. This turning point (point B) corresponds to the snap-back point with a pressure of 0.41 psi and a displacement of 0.61 in. Beyond point B, the pressure rose rapidly as the solution proceeded. Since the apex point came into contact with the backing plate at 0.98 psi, the point did not translate beyond the gap distance (0.757 in.) despite the increasing pressure.

This problem does not involve the material nonlinearity and, therefore, the unloading curve should follow the reversed loading curve. This proved the validity of Crisfield's method. Note that the unloading steps did not coincide with the loading steps since the load increment varied adaptively during the analyses.

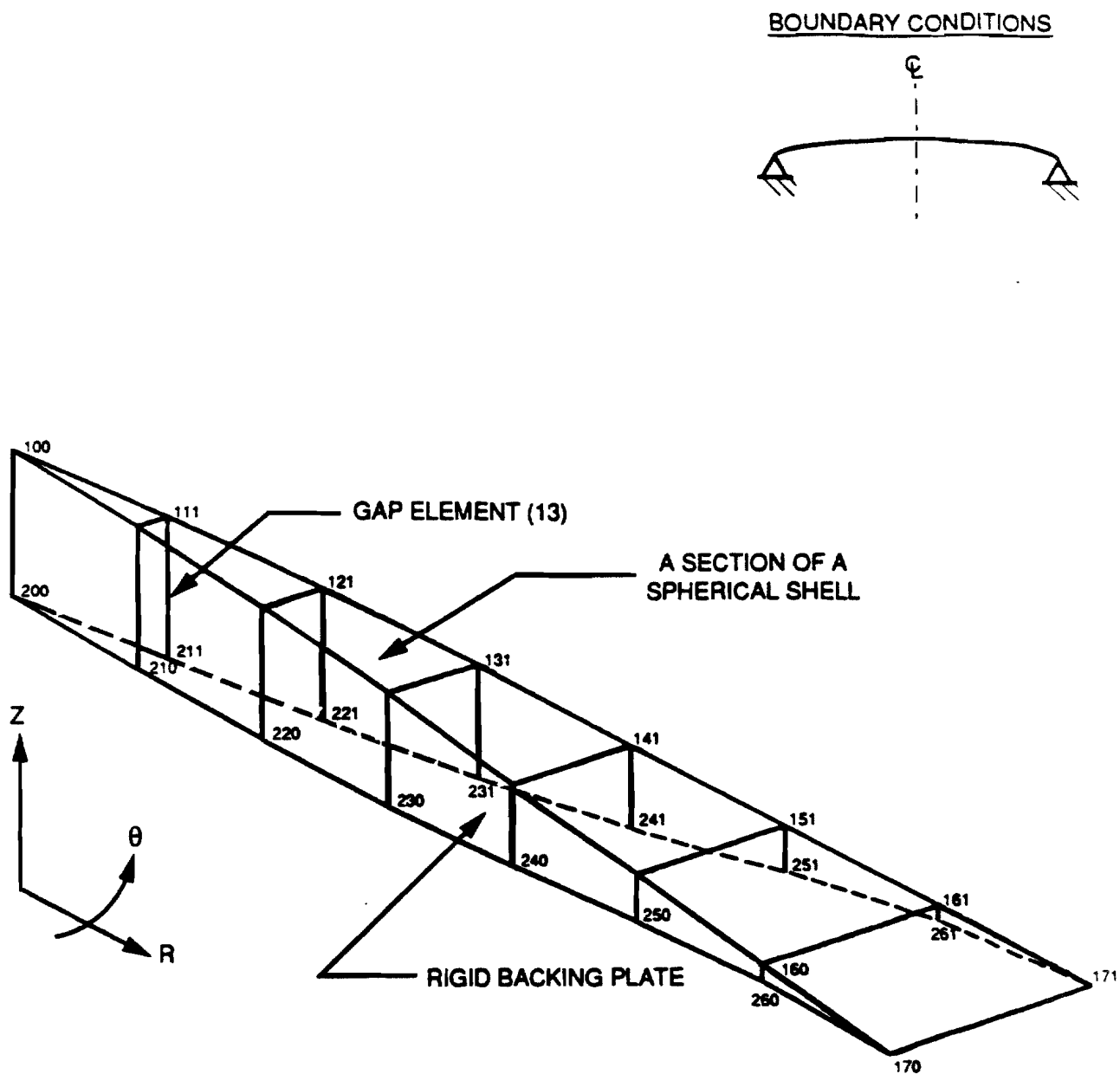


Figure 3.7.4. Spherical Diaphragm Model with Axisymmetric Boundary Condition and Constraints by Gap Elements.

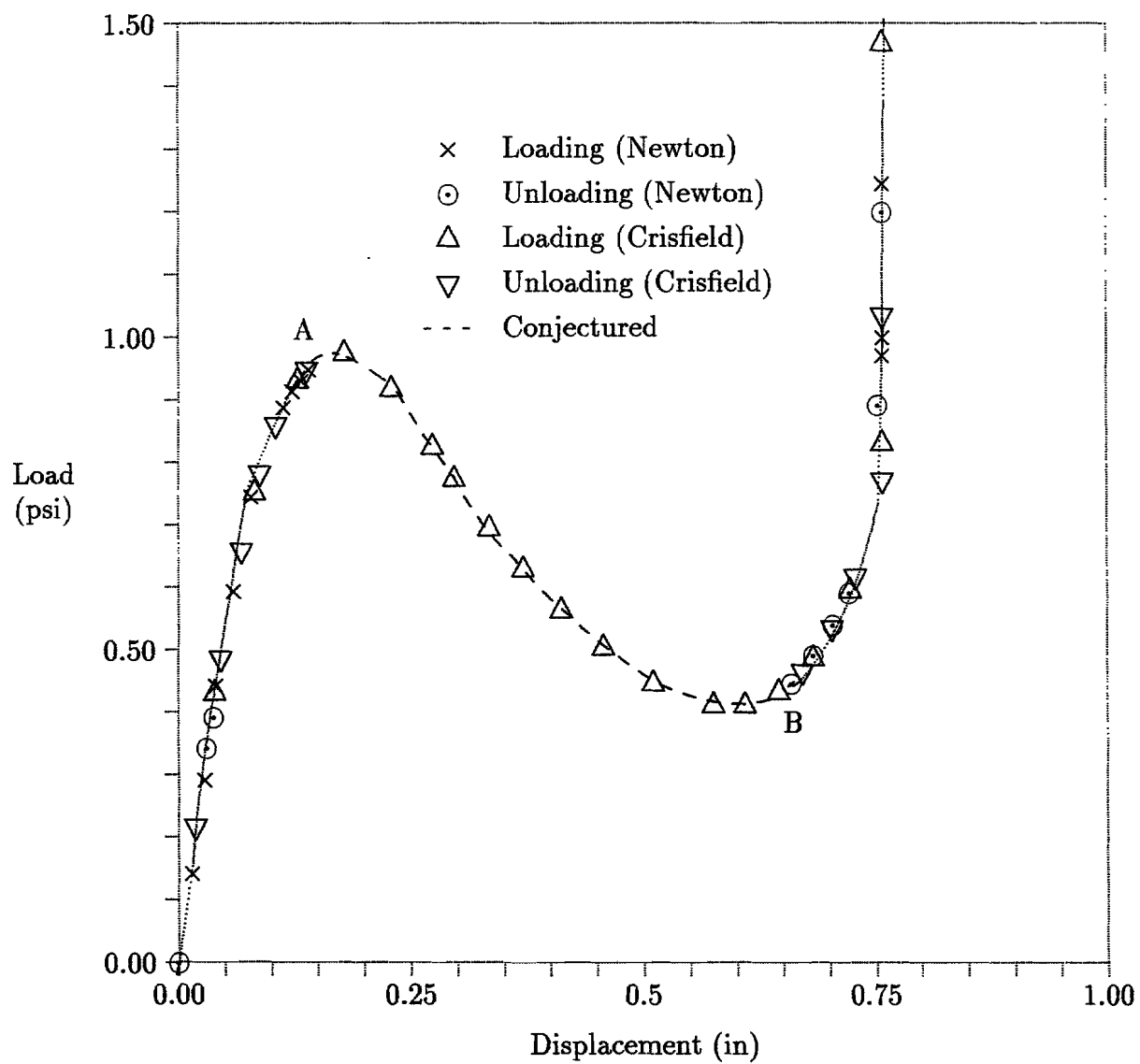


Figure 3.7.5. Load vs. Vertical Displacement at the Center (Clamped Case).

POST-BUCKLING BEHAVIOR: AN IMPERFECT SPHERICAL CAP

A spherical shell with an initial imperfection was analyzed [3.19]. The shell was subjected to an external uniform pressure, while the periphery was clamped. The problem was assumed to remain axisymmetric in geometry and loading throughout the deformation. The details of the geometry and the material are shown in Fig. 3.7.6. The initial imperfection was introduced by making the radius of curvature near the apex greater than the shell radius. The material was elasto-plastic with von Mises yield criterion and kinematic hardening. The large displacement effect was also included in the analysis. The finite element model shown in Fig. 3.7.7 represents a 10-degree sector with axisymmetric boundary conditions.

In the analysis, the external pressure was gradually increased: 2000 psi, 3000 psi, and 4000 psi at the end of SUBCASE 1, SUBCASE 2, and SUBCASE 3, respectively. Since Newton's method is usually more efficient than the arc-length methods if the stiffness matrix is positive-definite and not close to being singular, it was decided to use Newton's method in the first two subcases and Crisfield's in the third. The solution proceeded successfully and passed beyond the peak point, unlike Newton's method. The run automatically stopped when the number of load increments reached the maximum limit MXINC. Fig. 3.7.8 shows the load-deflection curve at the apex. The peak pressure and its corresponding displacement are identified as 3574 psi and 0.0139 in., respectively. Since a finite number of load steps was used, the predicted solution would not have passed through the exact peak point, but the error in the buckling pressure should be small because the curve is plateaued in the vicinity of the peak. On the other hand, the maximum displacement at buckling lies between 0.0139 in. and 0.0159 in., where the DECOMP module encountered negative terms in the factor diagonal of the stiffness matrix. By using the arc-length method, the buckling load as well as the post-buckling behavior can be predicted in a single run. The deformed shapes are shown in Fig. 3.7.9.

When the same model was analyzed by using Newton's iteration method in all three subcases, the solution diverged in the third subcase. A restart run was made in SOL 66 for the eigenvalue analysis [3.20], starting from the solution at $P=3300$ psi. Solutions at $P=3400$ psi and 3500 psi were repeated and a buckling analysis was performed using the stiffnesses between those two steps. The buckling pressure and displacement estimated by this method are 3547 psi and 0.0122 in., respectively. This solution is not as accurate as that predicted by the arc-length method.

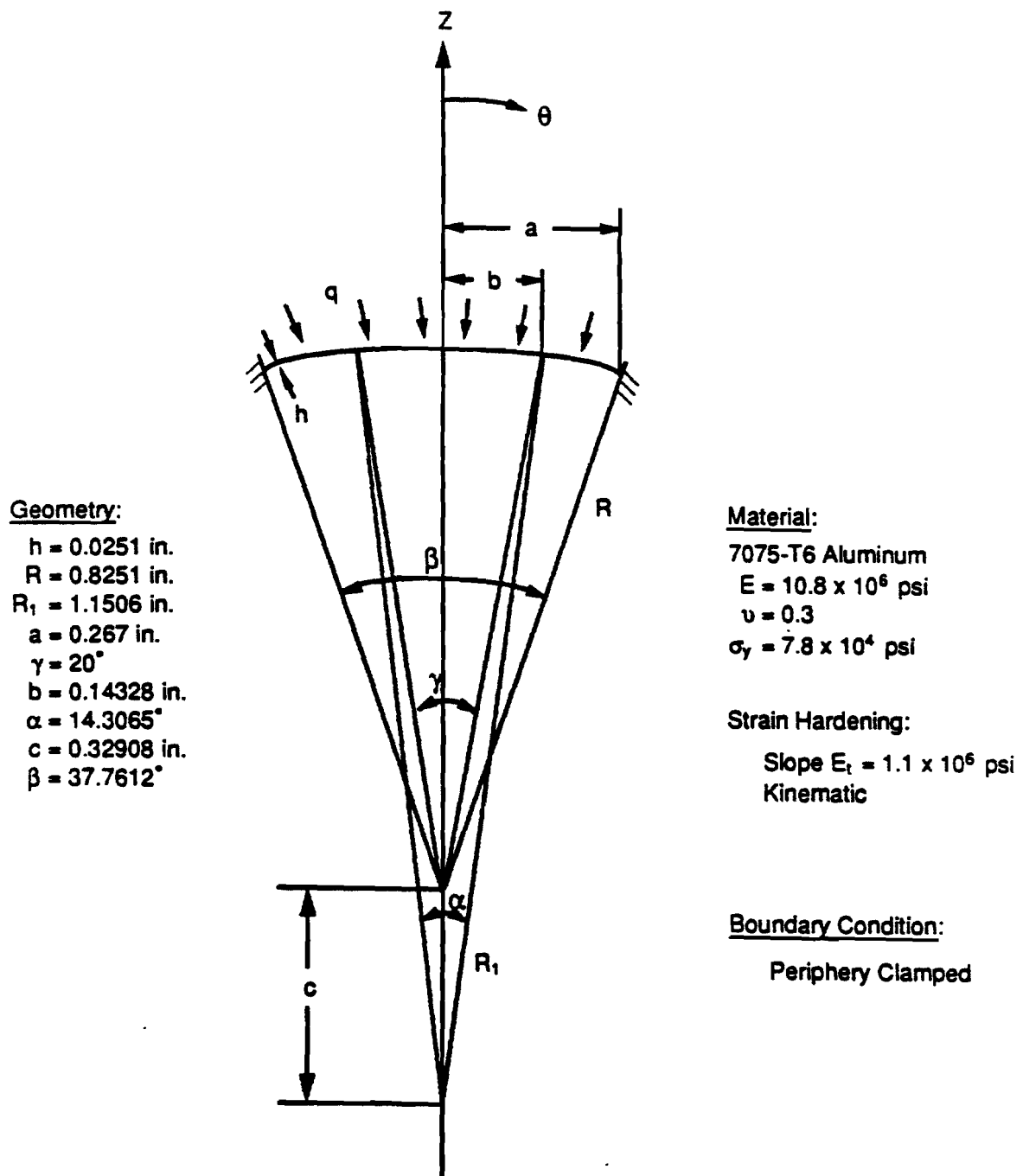


Figure 3.7.6. Imperfect Spherical Shell.

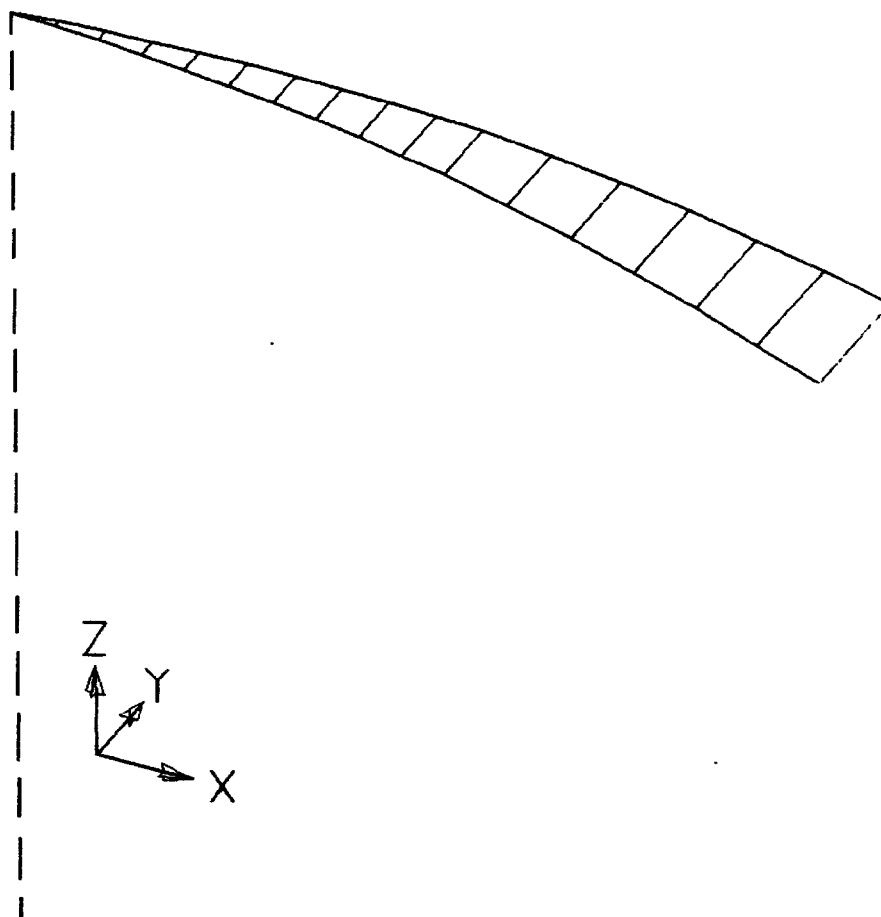


Figure 3.7.7 Model for Imperfect Spherical Shell.

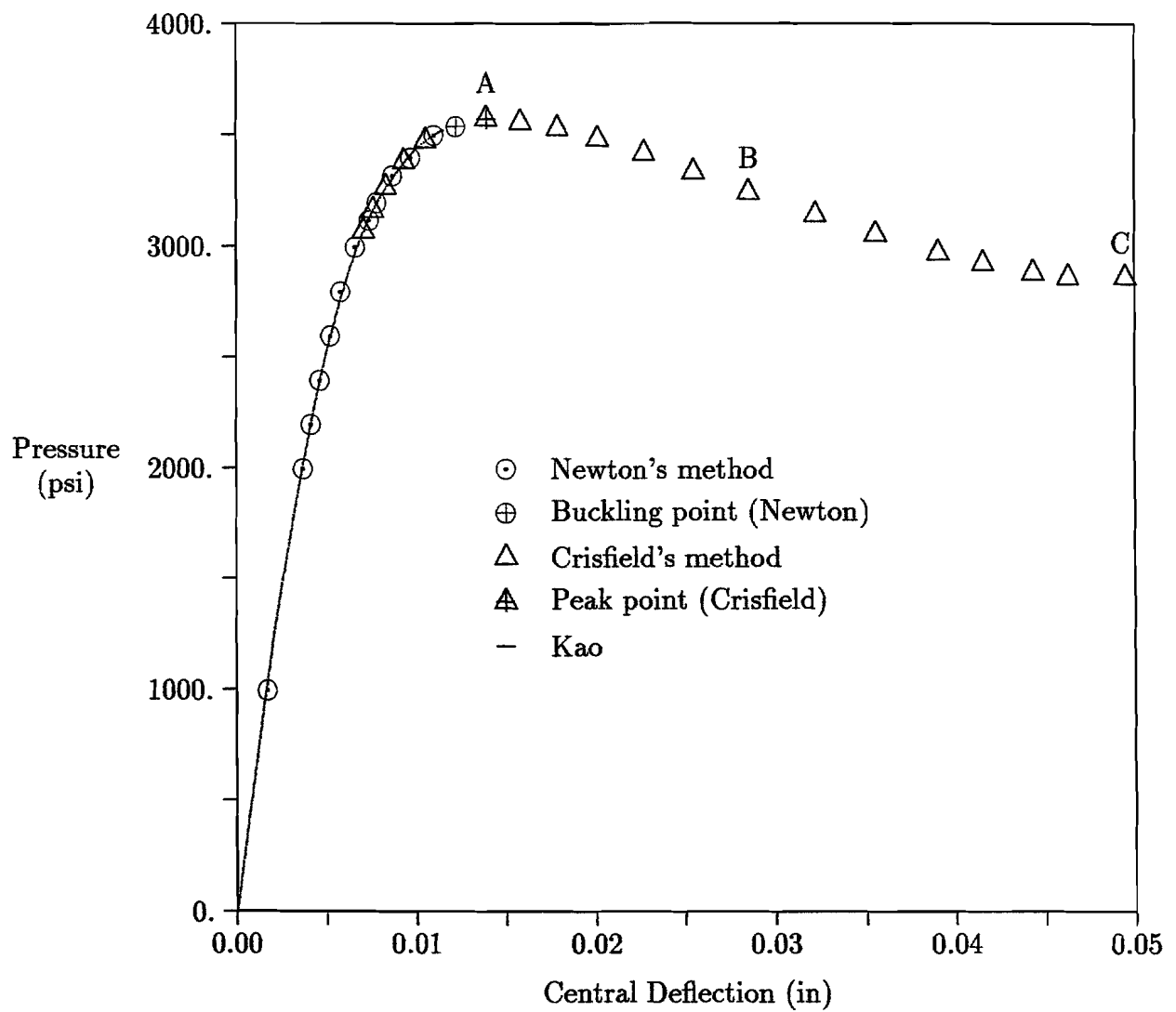
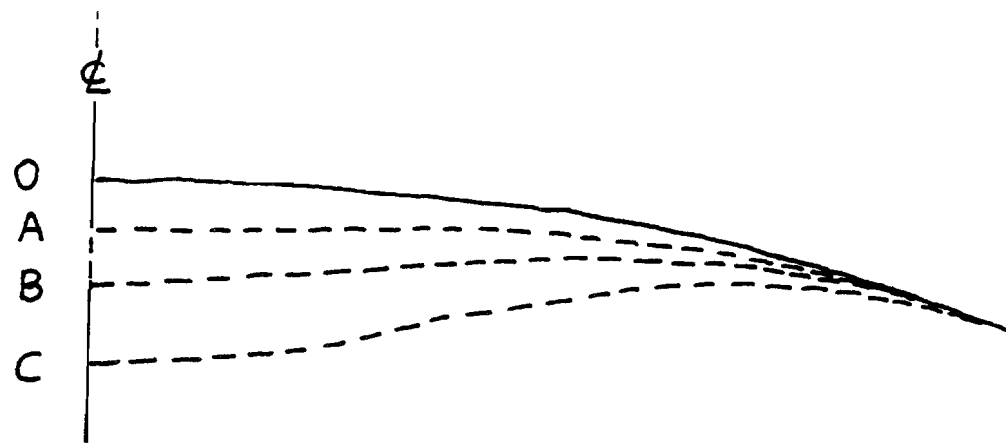


Figure 3.7.8. Load-Deflection Curve of Imperfect Spherical Shell.



O: Undeformed Shape

A: at $q=3574$ psi, $u=0.0139$ in.

B: at $q=3243$ psi, $u=0.0286$ in.

C: at $q=2863$ psi, $u=0.0495$ in.

Figure 3.7.9. Deformed Shapes at Different Buckling Stages.

Chapter 4

FINITE ELEMENTS IN NONLINEAR ANALYSIS

4.1 OVERVIEW

In nonlinear finite element analysis, lower-order elements are often preferred over higher-order ones because of their robustness and reasonable accuracy at reduced costs. The software employs linear elements, rather than quadratic or cubic elements, to process nonlinearity. When using lower-order elements, quadrilateral and hexahedral elements are generally preferred over triangular, pent ahedral or tetrahedral elements. Triangular and tetrahedral elements can exhibit excessively stiff behavior, and caution is needed when using these elements to hull.

Caution is also needed when different element types are combined in a model, and if these elements are incompatible (e.g., HEXA and QUAD4). In such cases, some provision (e.g., appropriate constraints) may be necessary at the interface boundary. Modeling the joints (such as bolted, riveted, or welded) is particularly difficult. For lack of better information, the joints are usually modeled as rigid or free in certain degrees-of-freedom. If improved accuracy is required at such joints, the characteristics of the joint (stiffness and/or damping) may have to be identified from experiments or the local analysis of a detailed model at the joint. Modeling of the boundary conditions at the supports poses similar difficulties. Ideal boundaries are represented as free, clamped, pinned, roller or ball joints. The reality tends to be in smeared condition.

Another common modeling problem involves offsets at the interface boundary. Offset options in beam, plate and shell elements are provided by the software. These offset options are not fully validated for nonlinear analysis, especially for material nonlinearity. Alternatives to these offset options are

- using very stiff rod elements for the offset distance
- using rigid elements or MPC relations if the large rotation is not involved.

The solver's element library includes about forty elements, of which some are considered obsolete and some are not applicable to the nonlinear analysis. Elements which may be used and are recommended for nonlinear analysis are tabulated below:

Line element	Surface element	Solid element	Other elements
* CROD	* CQUAD4	* CHEXA	CGAP
* CONROD	CQUAD8	* CPENTA	CELASi
* CTUBE	* CTRIA3	* CTETRA	CTRIAX6
CBAR	CTRIA6		+* CQUAD4,CTRIA3
* CBEAM	CSHEAR		CONM2
CBEND	RTRPLT		GENEL
RROD,RBAR			RBE2,RBE3

Notes:

- * denotes elements capable of geometric and material nonlinear processes.
- + denotes a plane-strain element.

All the elements listed above may be used in the nonlinear analysis (SOL 66 and SOL 99) if they are combined properly in the model. Elements that are potentially capable of nonlinear processes (those denoted by *) become actively nonlinear if the parameter LGDISP is turned on or nonlinear material is assigned to those via MATS1 or CREEP bulk data. As for geometric nonlinearity, the software does not currently support large strain capability. However, large displacement is treated effectively by computing element stresses and strains in the updated element coordinates. Active nonlinear elements may only be used in the residual superelement. Linear elements [4.1], however, may be used in the residual superelement for the region that remains linear **throughout** the analysis. All the upstream superelements are assumed to remain linear **throughout** the analysis.

4.1.1 Line Elements

The truss type elements (CROD, CONROD and CTUBE) differ only in input convenience. The ROD element is a straight element connecting two GRID points, with axial and torsional stiffnesses in displacements u and θ_x , respectively. The ROD element can sustain axial force P and torque T , producing axial and shear stresses as follows:

$$P = A\sigma_x \quad \text{and} \quad T = \int_A r\tau_{\theta z}dA \quad (4.1.1)$$

with

$$\sigma_x = \int E d\epsilon_x \quad \text{and} \quad \tau_{\theta z} = G\gamma_{\theta z} \quad (4.1.2)$$

where E is the Young's modulus or an appropriate modulus for the nonlinear material and G is the shear modulus.

Geometric nonlinear effects are reflected in the stiffness via the differential stiffness and the internal forces using the method of updated element coordinates. However, nonlinear material behavior will only be exhibited in the axial deformation. The torsional stress is assumed linear and not coupled to the axial stress, i.e., the linear shear stress is simply computed by

$$\tau_{\theta z}|_{max} = \frac{Tc}{J} \quad \text{with} \quad J = \int_A r^2 dA \quad (4.1.3)$$

where T is the torsional moment, J the torsional rigidity, and c the user-specified value to determine the stress. The stress is determined based on a single Gauss integration; therefore the stress and strain are assumed constant throughout the length. The strain is computed based on the small strain theory, i.e.,

$$\epsilon_x = \frac{L - L_0}{L_0} \quad (4.1.4)$$

where L and L_0 denote the current and original length of the element, respectively. The stiffness matrix for a ROD element is the (8x8) matrix shown below:

$$K = \begin{bmatrix} \tilde{K} & -\tilde{K} \\ -\tilde{K} & \tilde{K} \end{bmatrix} \quad \text{with} \quad \tilde{K} = \begin{bmatrix} k_x & & & \\ & k^d & & \\ & & k^d & \\ & & & k_\theta \end{bmatrix} \quad (4.1.5)$$

where

$$k_x = \frac{EA}{L}, \quad k_\theta = \frac{GJ}{L} \quad \text{and} \quad k^d = \frac{A\sigma_x}{L}$$

The BAR element is derived from classical beam theory; the plane cross-section remains planar under deformation. It provides exact linear solutions for end loads of any kind. Its major limitations are that it is prismatic (i.e., properties do not vary along the length), that the shear center and the neutral axis coincide, and that the torsional stiffening due to warping is neglected. These limitations are relaxed with the BEAM element in addition to the nonlinear capabilities to be described in the following section. Geometrical properties and the directions of internal forces of the BEAM are shown in Fig. (4.1.1).

The BEND element forms a circular arc, connecting two nodal points. Its primary use is to model a piping elbow. RROD and RBAR are rigid elements, corresponding to ROD and BAR, respectively.

4.1.2 Surface Elements

The plate and shell elements (QUAD4, QUAD8, TRIA3, TRIA6) can be used to model membranes, plates and shells. The quadratic elements (QUAD8 and TRIA6) are applicable only for the linear part of the structure and provide mid-side nodes that may be deleted. These quadratic elements can represent the singly-curved shells or linear curvature shells more accurately than the linear shell elements at the same cost. However, this is not the case in the doubly-curved shells.

Linear shell elements (QUAD4 and TRIA3) may be used for nonlinear analysis with geometric and material nonlinearities. In general, the quadrilateral elements are more reliable than the triangular elements, because the TRIA3 element exhibits behavior too stiff in membrane action. Properties of shell elements are specified in the PSHELL data, but PCOMP can be alternatively used to model the composite layers. If the PCOMP is selected, however, no nonlinear material may be referenced. In other words, no material nonlinear capability exists in the composite elements.

The element coordinate systems are implicitly defined by positions and the order of the nodal connectivity. The different material coordinate system with respect to the element coordinate system may be defined in the shell element connection card as shown in Fig. (4.1.2). Components of forces and stresses, as shown in Fig. (4.1.3), are always recorded and output in the element coordinate system.

The SHEAR element can be used to model the shear panel in the linear part of the structure. The RTRPLT element may be used to model a rigid plate in the linear structure.

4.1.3 Solid Elements

Solid elements (HEXA, PENTA and TETRA) are quadratic isoparametric elements that may be degenerated to linear elements. Various schemes of Gaussian quadrature may be selected for the volume integration in the element property data specified in the PSOLID entry. The element coordinate systems are implicitly defined by positions and the order of the nodal connectivity (see description of HEXA, PENTA and TETRA for details). The material coordinate system (defaulted to the element coordinates) can be defined in the PSOLID entry. Stresses and strains for these elements are recorded in the material coordinate system.

The quadratic element with mid-side nodes may not be used in the nonlinear region. Linear solid elements applicable to the nonlinear region will be discussed in the following section.

4.1.4 Other Elements

The GAP element simulates a point-to-point contact including the friction effect, but without effects of large rotation. The formulation is based on the penalty method. The GAP element is a nonlinear element and loses its significance if included in a linear analysis. Plane strain elements are available for nonlinear analysis using QUAD4 and TRIA3 elements by specifying -1 in the MID2 field of the associated PSHELL entry. Plane strain elements were implemented by suppressing bending and enforcing the plane strain condition instead of the plane stress condition.

Other elements, which are linear but available for nonlinear analysis, include a lumped elastic spring (ELASi), an axisymmetric element (TRIAX6), a lumped mass (CONM2), a general element (GENEL), and rigid elements (RBE2, RBE3). The GENEL element is used to represent the stiffness of a substructure which has an arbitrary number of connected grid points. The input data may be obtained either from another computer run or from test data.

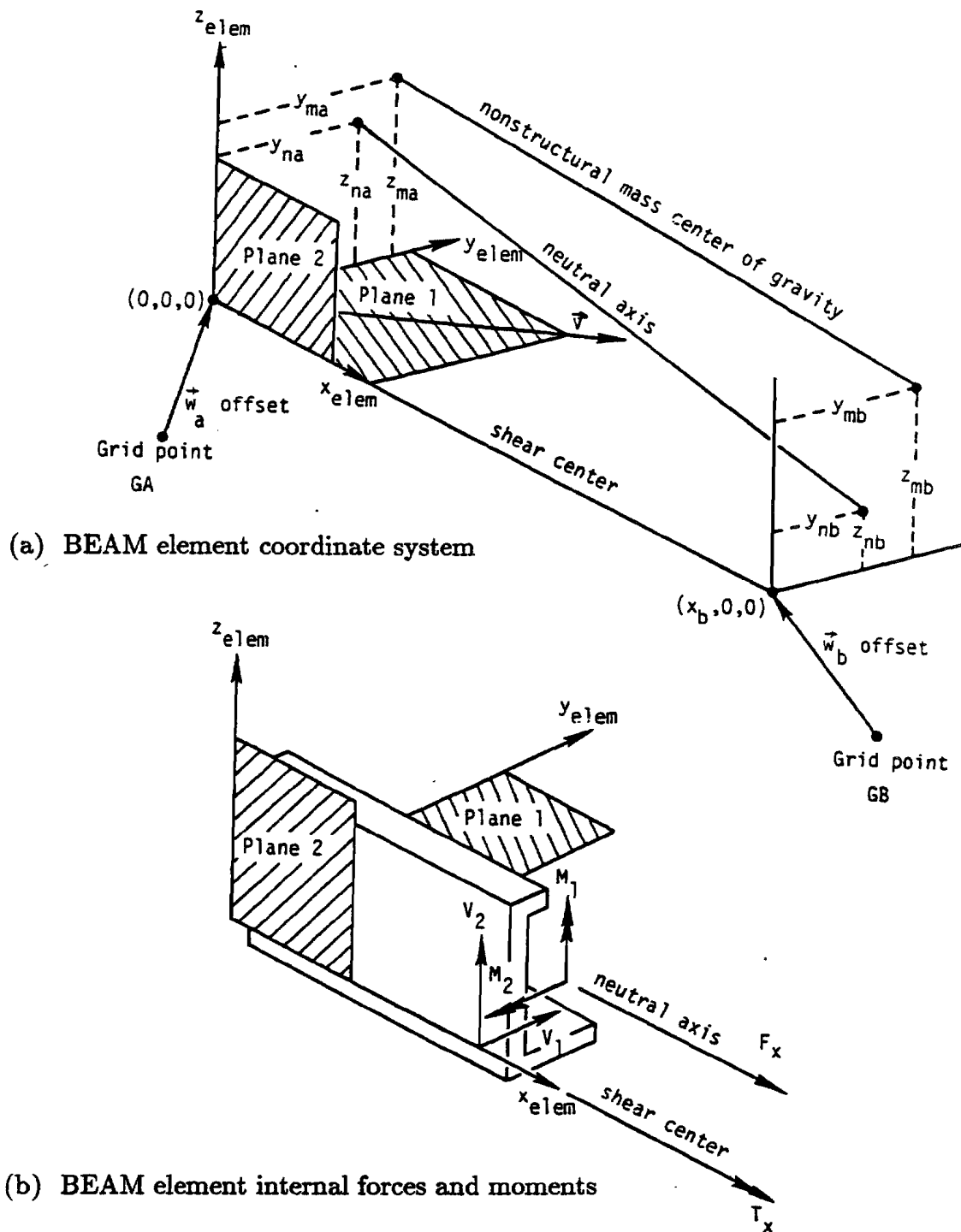
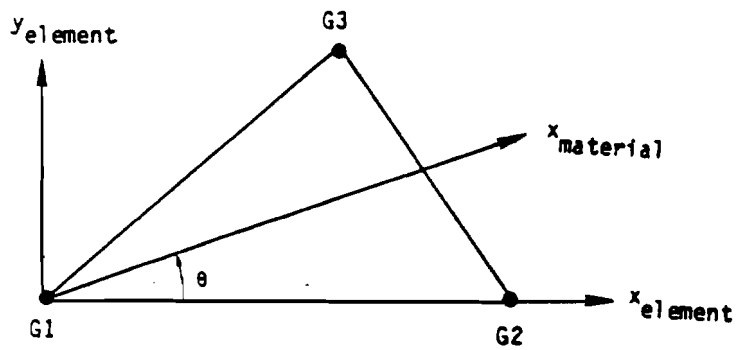
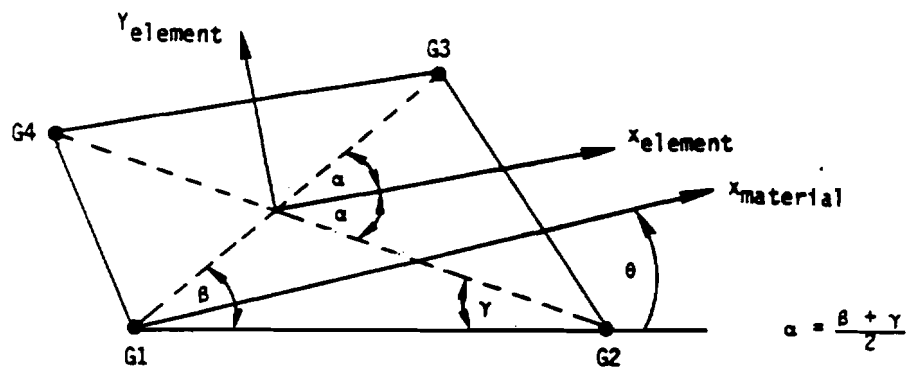


Figure 4.1.1 BEAM Element Coordinate System and Element Forces

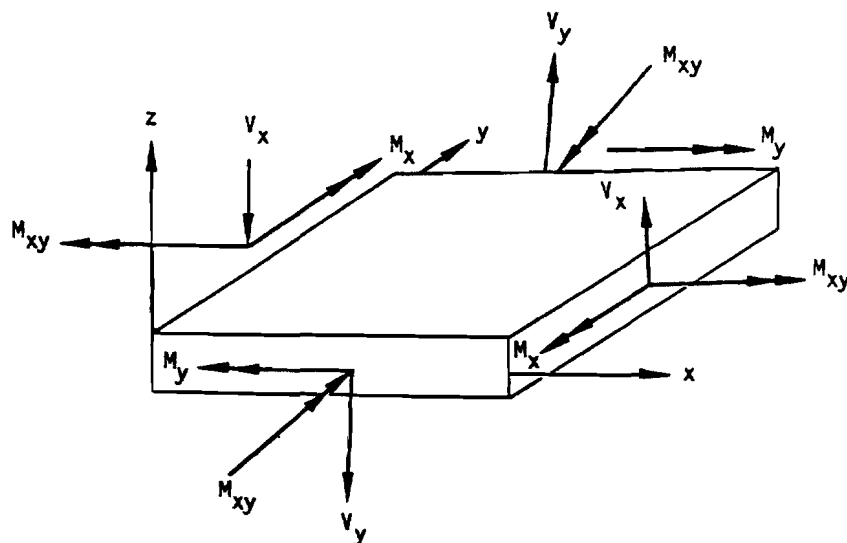


(a) TRIA3

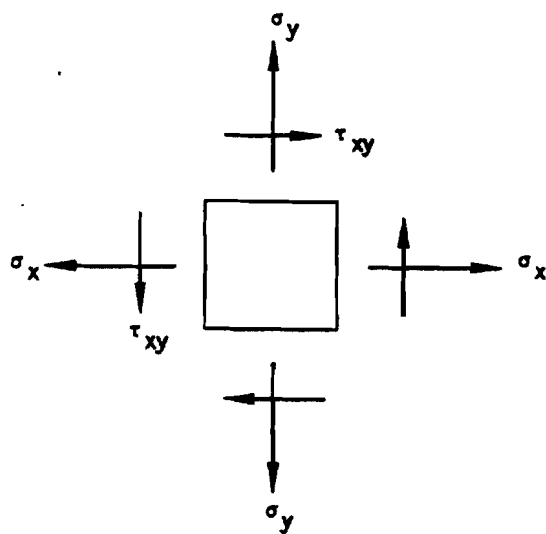


(b) QUAD4

Figure 4.1.2 Plate and Shell Element Coordinate Systems



(a) Forces and moments



(b) Stresses

Figure 4.1.3 Forces and Stresses in Plate Elements

4.2 FORMULATION OF ISOPARAMETRIC ELEMENTS

In the finite element method, parametric mapping is frequently used to map an irregular region into a regular one. The coordinate system used in the parametric mapping is a natural coordinate system suitable for the geometry. For instance, a natural coordinate system (ξ, η) is used for a quadrilateral surface in which each corner node has an extremum value of +1 or -1 in ξ and η . Then, the geometry of the internal points of the finite element can be described in terms of the nodal coordinates by the parametric mapping. The mathematical expression for this parametric mapping or interpolation is called a shape function. The displacement field inside the element should also be interpolated in terms of nodal displacements. The mapping is isoparametric if the same shape function is used to interpolate the displacement field as well as the geometry. The merit of isoparametric mapping is that the displacement field is invariant to the orientation of the cartesian coordinate system x and y .

Most of the elements are isoparametric elements, in which the shape functions are expressed in terms of isoparametric coordinates. In what follows, derivations are shown for the isoparametric coordinates, shape functions, element matrix describing the strain-displacement relations, volume integration for stiffness and mass matrices by Gauss quadrature, and element loads are derived for a tetrahedron element to illustrate element related operations.

4.2.1 Isoparametric Coordinates

Cartesian coordinates are not convenient to describe the geometry or the displacement field of a tetrahedron element. Let us introduce a set of volume coordinates $(L_1 \ L_2 \ L_3 \ L_4)$, such that

$$L_1 = \frac{\text{Volume (P234)}}{\text{Volume (1234)}}, \text{ etc.}$$

as shown in Fig. 4.2.1. The relation between volume and cartesian coordinates can be established for a linear tetrahedron (without midside nodes) as follows:

$$\begin{aligned} x &= L_1 x_1 + L_2 x_2 + L_3 x_3 + L_4 x_4 \\ y &= L_1 y_1 + L_2 y_2 + L_3 y_3 + L_4 y_4 \\ z &= L_1 z_1 + L_2 z_2 + L_3 z_3 + L_4 z_4 \\ 1 &= L_1 + L_2 + L_3 + L_4 \end{aligned} \tag{4.2.1}$$

It is obvious from Eq. (4.2.1) that the shape functions are simply the volume coordinates, i.e.,

$$N_1 = L_1, \ N_2 = L_2, \ \dots, \text{ etc.} \tag{4.2.2}$$

because $x = \sum N_i x_i$.

Shape functions for the quadratic tetrahedron can be derived using Lagrangian interpolation. These are

$$\begin{aligned} N_1 &= (2L_1 - 1)L_1, \text{ etc. for corner nodes} \\ \text{and } N_5 &= 4L_1L_2, \text{ etc. for midside nodes} \end{aligned} \quad (4.2.3)$$

Notice that these shape functions, Eqs. (4.2.2) and (4.2.3), satisfy element convergence criteria: integrability (C^{n-1} continuity for n-th derivative) and completeness (no straining by a rigid body mode, constant strain condition, and continuous displacement field).

With the isoparametric element, the same shape functions are used to describe the displacement field as well as the geometry, i.e.,

$$x = \sum_1^{10} N_i x_i, \text{ etc.} \quad \text{and} \quad u = \sum_1^{10} N_i u_i, \text{ etc.} \quad (4.2.4)$$

The strains are calculated as

$$\{\varepsilon\} = [B]\{u\} \quad (4.2.5)$$

where strain vector

$$\{\varepsilon\}^T = \langle \varepsilon_x \ \varepsilon_y \ \varepsilon_z \ \gamma_{xy} \ \gamma_{yz} \ \gamma_{zx} \rangle$$

element matrix

$$[B] = [B_1 B_2 \dots B_i \dots B_{10}]$$

nodal displacement vector

$$\{u\}^T = \langle a_1^T a_2^T \dots a_i^T \dots a_{10}^T \rangle$$

with

$$\begin{aligned} \{a_i\} &= \begin{Bmatrix} u_i \\ v_i \\ w_i \end{Bmatrix} \\ [B_i] &= \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 & 0 \\ 0 & \frac{\partial N_i}{\partial y} & 0 \\ 0 & 0 & \frac{\partial N_i}{\partial z} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial z} & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial z} & 0 & \frac{\partial N_i}{\partial x} \end{bmatrix} \end{aligned} \quad (4.2.6)$$

Since the shape functions are defined in terms of local coordinates, a coordinate transformation is required to obtain global derivatives. Volume coordinates involve four coordinates

(one of which is dependent), and the Jacobian matrix will become rectangular. To avoid this difficulty, let us introduce local coordinates (ξ, η, ζ) as follows:

$$\begin{aligned} L_1 &= \xi \\ L_2 &= \eta \\ L_3 &= \zeta \\ L_4 &= 1 - \xi - \eta - \zeta \end{aligned} \quad (4.2.7)$$

Derivatives with respect to the local coordinates can be expressed in terms of global derivatives using the chain rule, i.e.,

$$\begin{aligned} \left\{ \begin{array}{c} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \\ \frac{\partial N_i}{\partial \zeta} \end{array} \right\} &= \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix} \left\{ \begin{array}{c} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial z} \end{array} \right\} \end{aligned} \quad (4.2.8)$$

$\begin{matrix} 3 \times 10 & & 3 \times 3 & & 3 \times 10 \end{matrix}$

where the Jacobian matrix is expressed in terms of shape functions as

$$[J] = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_2}{\partial \xi} & \dots & \frac{\partial N_{10}}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} & \frac{\partial N_2}{\partial \eta} & \dots & \frac{\partial N_{10}}{\partial \eta} \\ \frac{\partial N_1}{\partial \zeta} & \frac{\partial N_2}{\partial \zeta} & \dots & \frac{\partial N_{10}}{\partial \zeta} \end{bmatrix} \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_{10} & y_{10} & z_{10} \end{bmatrix} \quad (4.2.9)$$

This Jacobian matrix must be inverted to obtain global derivatives, from which the element matrix is formed. Notice that the determinant of the Jacobian matrix is called Jacobian which represents a volume change, i.e.,

$$dx dy dz = \det J d\xi d\eta d\zeta \quad (4.2.10)$$

4.2.2 Shape Functions

To make tetrahedron element compatible with other solid elements (HEXA and PENTA), deletion of any or all of the midside nodes is permitted [4.2]. The shape functions are modified with Kronecker deltas $(\delta_5 - \delta_{10})$ where

$$\begin{aligned} \delta_i &= 0 \quad \text{if the midside node } i \text{ is deleted} \\ &= 1 \quad \text{if the midside node } i \text{ is not deleted} \end{aligned}$$

The goal is to construct functions which are unity at the associated node and zero at all other nodes, regardless of any combination of deleted midside nodes, by the following scheme:

at the corner nodes

$$N_1 = \xi - \frac{1}{2}N_5 - \frac{1}{2}N_6 - \frac{1}{2}N_7, \text{ etc.}$$

at the midside nodes

$$N_5 = 4\xi\eta\delta_5, \text{ etc.}$$

Shape functions are identified as follows:

$$\begin{aligned} N_1 &= \xi[1 - 2\eta\delta_5 - 2\zeta\delta_6 - 2(1 - \xi - \eta - \zeta)\delta_7] \\ N_2 &= \eta[1 - 2\xi\delta_5 - 2\zeta\delta_8 - 2(1 - \xi - \eta - \zeta)\delta_{10}] \\ N_3 &= \zeta[1 - 2\xi\delta_6 - 2\eta\delta_8 - 2(1 - \xi - \eta - \zeta)\delta_9] \\ N_4 &= (1 - \xi - \eta - \zeta)(1 - 2\xi\delta_7 - 2\zeta\delta_9 - 2\eta\delta_{10}) \\ N_5 &= 4\xi\eta\delta_5 \\ N_6 &= 4\xi\zeta\delta_6 \\ N_7 &= 4\xi(1 - \xi - \eta - \zeta)\delta_7 \\ N_8 &= 4\eta\zeta\delta_8 \\ N_9 &= 4\zeta(1 - \xi - \eta - \zeta)\delta_9 \\ N_{10} &= 4\eta(1 - \xi - \eta - \zeta)\delta_{10} \end{aligned}$$

Derivatives of the shape functions with respect to the local coordinates are obtained as follows:

$$\begin{aligned} \frac{\partial N_1}{\partial \xi} &= 1 - 2\eta\delta_5 - 2\zeta\delta_6 - 2(-2\xi - \eta - \zeta + 1)\delta_7 \\ \frac{\partial N_2}{\partial \xi} &= -2\eta(\delta_5 - \delta_{10}) \\ &\vdots \\ \frac{\partial N_{10}}{\partial \xi} &= -4\eta\delta_{10} \\ \frac{\partial N_1}{\partial \eta} &= -2\xi(\delta_5 - \delta_7) \\ \frac{\partial N_2}{\partial \eta} &= 1 - 2\xi\delta_5 - 2\zeta\delta_8 - 2(1 - \xi - 2\eta - \zeta)\delta_{10} \\ &\vdots \\ \frac{\partial N_{10}}{\partial \eta} &= 4(1 - \xi - 2\eta - \zeta)\delta_{10} \\ \frac{\partial N_1}{\partial \zeta} &= -2\xi(\delta_6 - \delta_7) \\ \frac{\partial N_2}{\partial \zeta} &= -2\eta(\delta_8 - \delta_{10}) \\ &\vdots \\ \frac{\partial N_{10}}{\partial \zeta} &= -4\eta\delta_{10} \end{aligned}$$

These shape functions and derivatives may be reduced to those for 4-noded and 10-noded tetrahedron elements. It can be verified that, for any combination of deleted midside nodes, there exist a condition

$$\sum N_i = 1$$

which satisfies the constant strain requirement.

4.2.3 An Example of Element Matrix

To illustrate the computational procedure, an element matrix for a linear tetrahedron (4-noded) element is explicitly derived here. The shape functions and the derivatives are tabulated below:

Node	N_i	$\frac{\partial N_i}{\partial \xi}$	$\frac{\partial N_i}{\partial \eta}$	$\frac{\partial N_i}{\partial \zeta}$
1	ξ	1	0	0
2	η	0	1	0
3	ζ	0	0	1
4	$1 - \xi - \eta - \zeta$	-1	-1	-1

Then the Jacobian matrix may be found as

$$[J] = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ x_4 & y_4 & z_4 \end{bmatrix} = \begin{bmatrix} x_{14} & y_{14} & z_{14} \\ x_{24} & y_{24} & z_{24} \\ x_{34} & y_{34} & z_{34} \end{bmatrix}$$

where $(x_{ij} = x_i - x_j)$ is used for convenience. Upon inverting the Jacobian matrix, we have

$$[J]^{-1} = \frac{1}{\det J} \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}$$

where

$$\begin{aligned} a_1 &= y_{24}z_{34} - z_{24}y_{34} \\ b_1 &= -(x_{24}z_{34} - z_{24}x_{34}) \\ c_1 &= x_{24}y_{34} - y_{24}x_{34} \\ a_2 &= -(y_{14}z_{34} - z_{14}y_{34}) \\ b_2 &= x_{14}z_{34} - z_{14}x_{34} \\ c_2 &= -(x_{14}y_{34} - y_{14}x_{34}) \end{aligned}$$

$$\begin{aligned}
a_3 &= y_{14}z_{24} - z_{14}y_{24} \\
b_3 &= -(x_{14}z_{24} - z_{14}x_{24}) \\
c_3 &= x_{14}y_{24} - y_{14}x_{24}
\end{aligned}$$

The global derivatives of the shape functions are

$$\begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial x} & \frac{\partial N_4}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial y} & \frac{\partial N_4}{\partial y} \\ \frac{\partial N_1}{\partial z} & \frac{\partial N_2}{\partial z} & \frac{\partial N_3}{\partial z} & \frac{\partial N_4}{\partial z} \end{bmatrix} = \frac{1}{\det J} \begin{bmatrix} a_1 & a_2 & a_3 & -(a_1 + a_2 + a_3) \\ b_1 & b_2 & b_3 & -(b_1 + b_2 + b_3) \\ c_1 & c_2 & c_3 & -(c_1 + c_2 + c_3) \end{bmatrix}$$

Hence the element matrix will be

$$[B]_{6 \times 12} = \frac{1}{\det J} \begin{bmatrix} a_1 & 0 & 0 & : & a_2 & 0 & 0 & : & a_3 & 0 & 0 & : & -\sum a_i & 0 & 0 \\ 0 & b_1 & 0 & : & 0 & b_2 & 0 & : & 0 & b_3 & 0 & : & 0 & -\sum b_i & 0 \\ 0 & 0 & c_1 & : & 0 & 0 & c_2 & : & 0 & 0 & c_3 & : & 0 & 0 & -\sum c_i \\ b_1 & a_1 & 0 & : & b_2 & a_2 & 0 & : & b_3 & a_3 & 0 & : & -\sum c_i & -\sum a_i & 0 \\ 0 & c_1 & b_1 & : & 0 & c_2 & b_2 & : & 0 & c_3 & b_3 & : & 0 & -\sum c_i & -\sum b_i \\ c_1 & 0 & a_1 & : & c_2 & 0 & a_2 & : & c_3 & 0 & a_3 & : & -\sum c_i & 0 & -\sum a_i \end{bmatrix}$$

Notice that the rank of $[B]$ matrix is 6.

4.2.4 Volume Integration of Element Matrices

By virtue of variational principles, the element stiffness matrix is derived as follows:

$$\begin{aligned}
[K^e] &= \int_V B^T D B dV \\
&= \int_0^1 \int_0^{1-\zeta} \int_0^{1-\eta-\zeta} |\det J| B^T D B d\xi d\eta d\zeta
\end{aligned} \tag{4.2.11}$$

where D is a (6×6) material tangential matrix.

While analytical integration is possible, there are some advantages in using numerical integration. The Gaussian quadrature is used for the tetrahedron as usual. The minimum number of intergration points required for non-singular stiffness matrix may be determined based on

$$\text{minimum } n = \frac{\text{Total dof - rigid body dof}}{\text{No. of strain components for each point}} \tag{4.2.12}$$

These numbers are found to be 1 and 4 for linear and quadratic tetrahedron, respectively. Using the integration formulas, shown in Fig. 4.2.2, the element stiffness matrices for linear and quadratic tetrahedron may be computed as

$$[K^e] = \frac{1}{6} |\det J| B^T D B \quad \text{for 4-noded TETRA}$$

$$[K^e] = \frac{1}{6} \times \frac{1}{4} \sum_{\text{gauss}}^{a-d} |\det J| B^T D B \quad \text{for 5-10 noded TETRA} \quad (4.2.13)$$

where the scaling factor $1/6$ is introduced to compensate $|J| = 6 \times (\text{Volume})$. One-point scheme should be used for 4-noded tetrahedron (all the midside nodes deleted) and four-point scheme otherwise.

It is difficult to determine lumped masses when some of the midside nodes are deleted. However, the consistent mass matrix may be obtained using the same shape functions, i.e.,

$$M^e = \int_V \tilde{N}^T \rho \tilde{N} dV \quad (4.2.14)$$

where $[\tilde{N}] = [N_1 I \ N_2 I \ \dots \ N_{10} I]$ with I being (3×3) identity matrix. Again the Gaussian quadrature 4-point formula is used to find

$$[M]^e = \frac{1}{24} \rho \sum_{\text{gauss}} |\det J| [\tilde{N}]^T [\tilde{N}] \quad (4.2.15)$$

Notice that 4-point scheme is to be used even for 4-noded tetrahedron. For computational convenience, the consistent mass may be converted to the lumped mass. One way to achieve this is to take the diagonal terms and scale them so that the total mass is preserved.

4.2.5 Element Loads and Equilibrium

Nodal forces are computed from the element stresses using element matrix, i.e.,

$$\{F\}^e = \int_V B^T \{\sigma\} dV \quad (4.2.16)$$

The Gaussian integration is performed as

$$\{F\}^e = \frac{1}{6} |\det J| [B]^T \{\sigma\} \quad (4.2.17)$$

for 4-noded tetrahedron and

$$\{F\}^e = \frac{1}{24} \sum_{\text{gauss}} |\det J| [B^T] \{\sigma\}$$

12×1
12×6

30×1
6×1

for (5-10)-noded tetrahedron.

Thermal load is like an initial strain. Therefore, nodal forces due to thermal load are obtained by

$$\{F\}_{thermal}^e = \int_V B^T [D] \{\varepsilon_0\} dV \quad (4.2.18)$$

where $\{\varepsilon_0\}^T = \alpha \Delta T < 1 \ 1 \ 1 \ 0 \ 0 \ 0 >$. With anisotropic thermal properties, $\{\varepsilon_0\}^T$ becomes $\Delta T < \alpha_1, \alpha_2 \dots \alpha_6 >$ in general.

The pressure load applied to any surface of the tetrahedron may be distributed to the grid points using the shape functions applicable to the 6-noded triangular element with removable midside nodes, i.e.,

$$\{P\}_{pressure} = \int_A p [\tilde{N}']^T \{n\} dA \quad (4.2.19)$$

where $\{n\}$ is a unit direction vector associated with a pressure (p) and

$$[\tilde{N}'] = [N'_1 I \ N'_2 I \ \dots \ N'_6 I]$$

with I being (3×3) identity matrix. Shape functions (N'_i) are shown in Fig. 4.2.3. Area integration should be performed using Gaussian quadrature formulas shown in Fig. 4.2.4, i.e.,

$$\{P\}_{pressure} = \frac{1}{6} \sum_{gauss}^{a-c} |\det J| \tilde{N}'^T \{p\} \quad (4.2.20)$$

During the nonlinear iteration process, the equilibrium is not reached until convergence is achieved. The equilibrium is sought in the global level when the residual load R approaches zero. The residual load vector is defined as

$$\{R\} = \{P\} - \sum_e^{linear} \{K^e u^e\} - \sum_e^{nonlinear} \left\{ \int B^T \sigma dV + F_{thermal}^e \right\} \quad (4.2.21)$$

where $\{P\}$ is the applied load vector including the thermal load and \sum implies assemblage in the global coordinate system (coordinate transformation required).

4.2.6 Element Coordinates

The element coordinate system for the tetrahedron element is shown in Fig. 4.2.5. It is defined with the initial element geometry such that

- The origin is at the first grid point in the connectivity, G1.
- The x-axis is determined by connecting the origin G1 to node G2, i.e.,

$$\vec{i} = \left[\left\{ \begin{array}{c} x_2 \\ y_2 \\ z_2 \end{array} \right\} - \left\{ \begin{array}{c} x_1 \\ y_1 \\ z_1 \end{array} \right\} \right]_{normalized} \quad (4.2.22)$$

- The y-axis is determined by orthonormalization (Gram-Schmidt process) of the edge direction (\vec{V}_{13}) with respect to x-axis,

$$\vec{j} = [\vec{V}_{13} - (\vec{i} \bullet \vec{V}_{13})\vec{i}]_{normalized} \quad (4.2.23)$$

- The z-axis is orthogonal to x and y according to the right-hand rule, i.e.,

$$\vec{k} = \vec{i} \times \vec{j}$$

Then, the transformation from element coordinates to the basic coordinates is simply

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{basic} = [T_{be}] \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{element} + \begin{Bmatrix} x^e \\ y^e \\ z^e \end{Bmatrix}_{basic} \quad (4.2.24)$$

where

$$[T_{be}] = \begin{bmatrix} i_x & j_x & k_x \\ i_y & j_y & k_y \\ i_z & j_z & k_z \end{bmatrix} \quad (4.2.25)$$

and $\langle x^e, y^e, z^e \rangle_{basic}^T$ is the position vector of the element coordinate system with respect to the basic coordinate system. Notice that $[T_{be}]$ is an orthogonal matrix, i.e.,

$$[T_{be}]^{-1} = [T_{be}]^T$$

4.2.7 Stress Data Recovery

In case of linear analysis, the stresses at the integration point are recovered as

$$\{\sigma\}_{gauss} = [D]\{[B]\{u\} - \{\alpha\}\Delta T\} \quad (4.2.26)$$

with

$$\Delta T = \{N_i\}^T \{T_i - T_{oi}\}$$

where $\{\sigma\}$, $[B]$, $\{u\}$ and $\{\alpha\}$ are defined in the element coordinate system, and the shape function $\{N_i\}$ interpolates Gauss point temperatures from the nodal temperatures. In case of nonlinear analysis, stresses are computed again upon convergence, starting from the last converged state (last converged solution of σ and u), i.e.,

$$\{\sigma_{new}\}_{gauss} = \{\sigma_{old}\} + \int_{\varepsilon_{old}}^{\varepsilon_{old} + \Delta\varepsilon} D d\varepsilon$$

Given $\{\sigma_{old}\}$ and

$$\{\Delta\varepsilon\} = \{\varepsilon\} - \{\varepsilon_{old}\} \quad (4.2.27)$$

with

$$\{\varepsilon\} = [B]\{u\} - \{\alpha\}\Delta T$$

the nonlinear material routine computes updated stresses $\{\sigma_{new}\}$, which are stored in ESTNL.

To output grid point stresses, the stresses at the integration points must be extrapolated to the nodal points. A linear extrapolation can be applied using stresses at the element c.g. and the corner Gauss point associated with a grid point. For a tetrahedron, referring to Fig. 4.2.2,

$$\sigma_1 = \frac{\sigma_a - \sigma_0}{\alpha - 1/4} \left(1 - \frac{1}{4}\right) + \sigma_0 \quad (4.2.28)$$

where

$$\sigma_0 = \frac{1}{4}(\sigma_a + \sigma_b + \sigma_c + \sigma_d)$$

This yields

$$\sigma_1 = \frac{1 - \beta}{1 - 4\beta} \sigma_a - \frac{\beta}{1 - 4\beta} (\sigma_b + \sigma_c + \sigma_d) \quad (4.2.29)$$

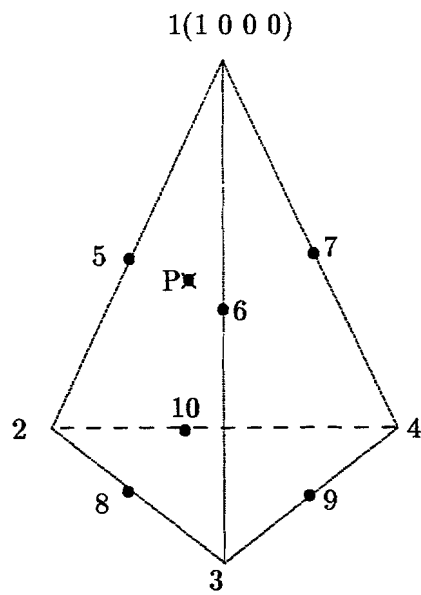
In matrix form for all the grid points in the element

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \end{Bmatrix}_{grid} = \begin{bmatrix} A & -B & -B & -B \\ & A & -B & -B \\ & & A & -B \\ \text{Sym.} & & & A \end{bmatrix} \begin{Bmatrix} \sigma_a \\ \sigma_b \\ \sigma_c \\ \sigma_d \end{Bmatrix}_{gauss} \quad (4.2.30)$$

where

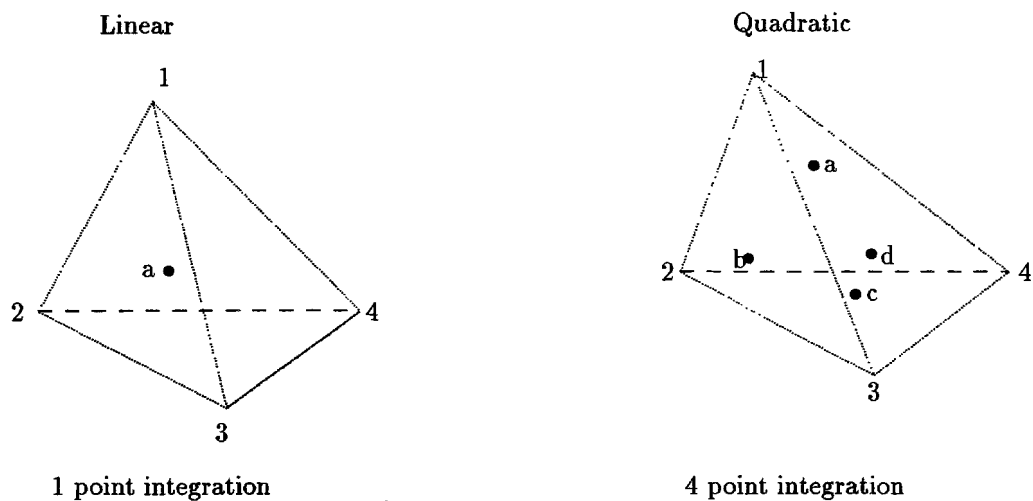
$$A = \frac{1 - \beta}{1 - 4\beta} \quad \text{and} \quad B = \frac{\beta}{1 - 4\beta}$$

This process must be operated on every component of stress vector. No extrapolation is required for 1-point integration.



GRID	L_1	L_2	L_3	L_4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1/2	1/2	0	0
6	1/2	0	1/2	0
7	1/2	0	0	1/2
8	0	1/2	1/2	0
9	0	0	1/2	1/2
10	0	1/2	0	1/2

Figure 4.2.1. Volume Coordinates for Tetrahedron

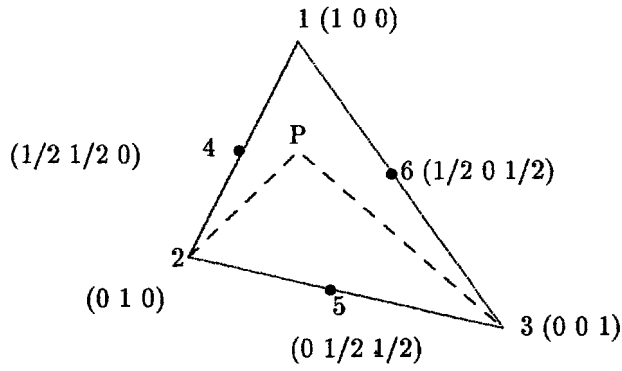


	Point	Weight	ξ	η	ζ	Error
Linear	a	1	1/4	1/4	1/4	$O(h^2)$
Quadratic	a	1/4	α	β	β	$O(h^3)$
	b	1/4	β	α	β	
	c	1/4	β	β	α	
	d	1/4	β	β	β	

$$\alpha = 1 - 3\beta = .58541020$$

$$\beta = 1 / (5 + \sqrt{5}) = 0.13819660$$

Figure 4.2.2 Gaussian Quadrature for Tetrahedron



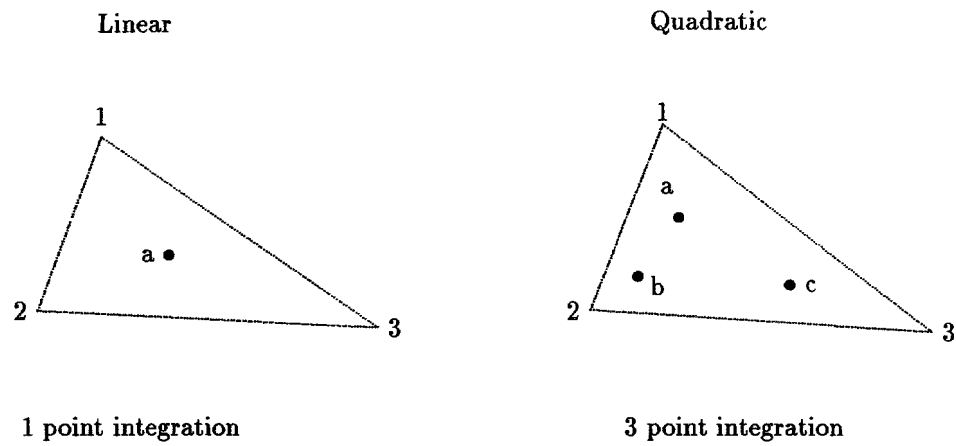
* Area coordinate system $(\xi \ \eta \ \zeta)$ such that

$$\xi = \frac{\text{Area}(P23)}{\text{Area}(123)}, \text{ etc.}$$

* Shape Functions:

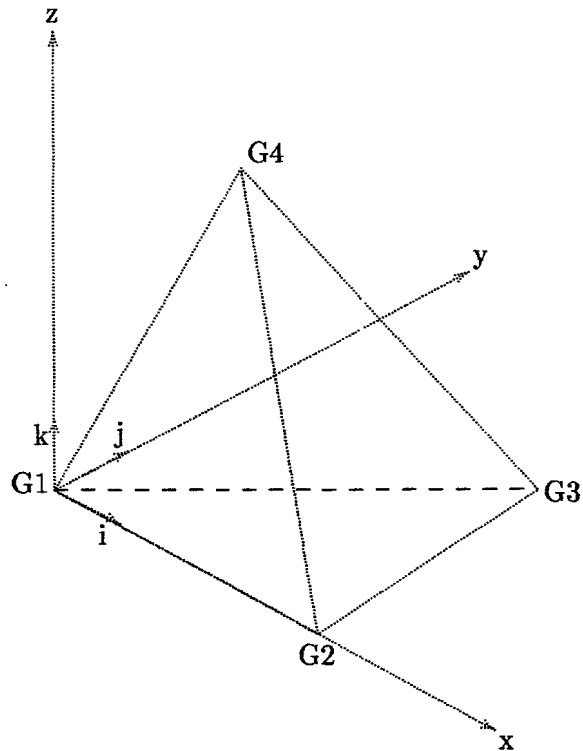
$$\begin{aligned} N'_1 &= \xi - \frac{1}{2}N_4 - \frac{1}{2}N_6 \\ N'_2 &= \eta - \frac{1}{2}N_4 - \frac{1}{2}N_5 \\ N'_3 &= \zeta - \frac{1}{2}N_5 - \frac{1}{2}N_6 \\ N'_4 &= 4\xi\eta\delta_4 \\ N'_5 &= 4\eta\zeta\delta_5 \\ N'_6 &= 4\xi\eta\delta_6 \end{aligned}$$

Figure 4.2.3 Shape Functions for Pressure Load



	Point	Weight	ξ	η	ζ	Error
Linear	a	1	1/3	1/3	1/3	$0(h^2)$
Quadratic	a	1/3	2/3	1/6	1/6	$0(h^3)$
	b	1/3	1/6	2/3	1/6	
	c	1/3	1/6	1/6	2/3	

Figure 4.2.4 Gaussian Quadrature for Pressure Load



Origin at G1

x-axis: in the direction of the edge from G1 to G2

y-axis: orthogonalized from edge (G1-G3), \vec{V}_{13}

z-axis: orthogonal to x and y according to the right hand rule

Figure 4.2.5 Element Coordinate System Definition

4.3 GAP AND FRICTION ELEMENT

One of the major areas in nonlinear analysis that poses difficulties in convergence is the application of GAP elements. This has been particularly true when friction is involved. Numerical difficulty is inherent in nature for the penalty method which is employed in the GAP element.

Furthermore, the GAP element simulates a point-to-point contact, and is often used to simulate surface contact problems. Deficiencies of the penalty GAP element are coped with by implementing an adaptive GAP element. The adaptability is based on the subincremental scheme with the capability to update the stiffness at proper timing, to bisect the increment when it is excessive, and to adjust the penalty stiffnesses to the proper values in concert with the nonlinear and dynamic environment. In addition, the kinetic friction effect (different from the static friction) is introduced to simulate the physics more accurately by distinguishing stick-slip forces.

4.3.1 Introduction

Contact problems are common in engineering practice. The GAP element simulates a unidirectional point-to-point contact using the Bulk Data CGAP and PGAP. The GAP element is one of the major areas that pose difficulties to the convergence of the iterative process, particularly when the friction is involved. The penalty method is used to simulate the rigidity between two degrees of freedom by adding a large value to the stiffness so that the two have approximately the same displacements. Penalty values are introduced to avoid penetration and to enforce the sticking condition (static friction) between two contact points. Difficulties (or inaccuracies) arise when the penalty values are not properly chosen. The success of this method is highly dependent on the user's choice of penalty values and requires a compromise between accuracy and numerical performance.

The Lagrange multiplier method has been used by researchers [4.3-4.5] in contact problems. The method enforces the constraints exactly. However, it also poses some difficulties. The main drawbacks of the Lagrange multiplier method are: (1) the problem size increases by Lagrange multiplier variables and (2) the stiffness matrix has zero diagonal terms, and requires pivoting and resequencing. Some variations of the Lagrange multiplier method have been proposed [4.6-4.8] to mitigate the deficiencies. Another school of thought for the dilemma is to alleviate the difficulties by implementing the adaptive penalty method to choose a proper penalty value and adjust it to control the numerical stability and the accuracy [4.9-4.10]. Although the constraint is not enforced exactly, the effectiveness of the penalty method lies in preserving the size and the bandwidth of the stiffness matrix. The software employs this approach in the adaptive GAP element implemented in Version 67.

4.3.2 Basic Considerations for Penalty GAP Element

The GAP element changes its status when the load is applied. Consider the internal forces of a GAP element with an isotropic friction. Force components in the GAP element are the axial compressive force (F_x) and the friction forces in lateral directions (F_y and F_z) in terms of displacements (u, v, w) in the element coordinate system. These internal forces can be computed based on the GAP status as follows:

- When the GAP is open (no contact, no lateral stiffness),

$$F_x = K_b u \leq 0. \quad \text{and} \quad F_y = F_z = 0. \quad (4.3.1)$$

where K_b is an arbitrary open stiffness.

- When the GAP is closed and sliding (no friction),

$$F_x = K_a u > 0. \quad \text{and} \quad F_y = F_z = 0 \quad (4.3.2)$$

where K_a is the penalty value for the closed stiffness.

- When the GAP is closed and sticking (static friction),

$$F_y^2 + F_z^2 \leq (\mu_s F_x)^2 \quad (4.3.3)$$

where $F_x = K_a u$, $F_y = K_t v$, and $F_z = K_t w$. It is noted that the sticking condition is dictated by the static coefficient of friction (μ_s) and the transverse shear stiffness (K_t).

- When the GAP is closed and slipping (kinetic friction),

$$F_y^2 + F_z^2 > (\mu_k F_x)^2 \quad (4.3.4)$$

where

$$F_x = K_a u, \quad F_y = \frac{v \mu_k F_x}{\sqrt{v^2 + w^2}}, \quad \text{and} \quad F_z = \frac{w \mu_k F_x}{\sqrt{v^2 + w^2}}$$

with the friction dictated by the kinetic coefficient of friction μ_k .

The element stiffness matrix can be formed as

$$[K] = \begin{bmatrix} \tilde{K} & -\tilde{K} \\ -\tilde{K} & \tilde{K} \end{bmatrix} \quad \text{with} \quad \tilde{K} = \frac{\partial \{F\}}{\partial \{u\}}$$

where the stiffness components are derived for each GAP status as follows:

- When the GAP is open,

$$\tilde{K} = \begin{bmatrix} K_b & 0. & 0. \\ & 0. & 0. \\ sym. & & 0. \end{bmatrix} \quad (4.3.5)$$

- When the GAP is closed and sliding,

$$\tilde{K} = \begin{bmatrix} K_a & 0. & 0. \\ & 0. & 0. \\ sym. & & 0. \end{bmatrix} \quad (4.3.6)$$

- When the GAP is closed and sticking,

$$\tilde{K} = \begin{bmatrix} K_a & 0. & 0. \\ & K_t & 0. \\ sym. & & K_t \end{bmatrix} \quad (4.3.7)$$

- When the GAP is closed and slipping,

$$\tilde{K} = \frac{K_a}{(v^2 + w^2)^{3/2}} \begin{bmatrix} (v^2 + w^2)^{3/2} & 0. & 0. \\ \mu_k v(v^2 + w^2) & \mu_k u w^2 & -\mu_k u v w \\ \mu_k w(v^2 + w^2) & -\mu_k u v w & \mu_k u v^2 \end{bmatrix} \quad (4.3.8)$$

It is noted that the slipping due to friction introduces an unsymmetric stiffness matrix.

There are some inherent difficulties involved in the penalty GAP element. Since the Newton's method is sensitive to abrupt changes in stiffness during iterations, it could cause a divergent or oscillatory solution when the GAP changes its status. The timing of the stiffness update is crucial to the convergence, which necessitates an adaptive stiffness update criterion based on the GAP status. Furthermore, if the GAP status undergoes two or more consecutive changes, adaptive stiffness update alone cannot stabilize the solution and the increment size should be reduced.

The stick-slip behavior of the frictional GAP element is path-dependent in a manner similar to the plasticity. The lateral forces due to friction should be considered for equilibrium. If the internal forces in the GAP elements are computed in a single step, the accuracy deteriorates when the increment produces large changes in displacements with friction. In order to trace the path-dependent solution, the subincremental scheme within an incremental load step is desirable [4.11].

Optimal penalty values (K_a and K_t) are difficult to assess. Furthermore, they have to be updated continuously during the deformation process, because the stiffness of the adjacent structure changes during the incremental process. It becomes necessary to adjust penalty values adaptively by the program throughout the analysis.

4.3.3 User Interface for GAP Element

Connectivity is specified in the Bulk Data entries CGAP as shown below:

1	2	3	4	5	6	7	8	9	10
CGAP	EID	PID	GA	GB	X1,G0	X2	X3	CID	
CGAP	17	2	.110	112				3	

Line GA-GB defines the GAP element x-axis as shown in Fig. 4.3.1. If GA and GB are coincident in space, CID is used to define the element coordinate system. The orientation of the y and z axes is determined by G0 (or X1, X2, X3) defining the x-y plane. The element coordinate system does not rotate as a result of deflections. In other words, the element coordinate system is not updated even when the geometric nonlinearity is considered (LGDISP=1).

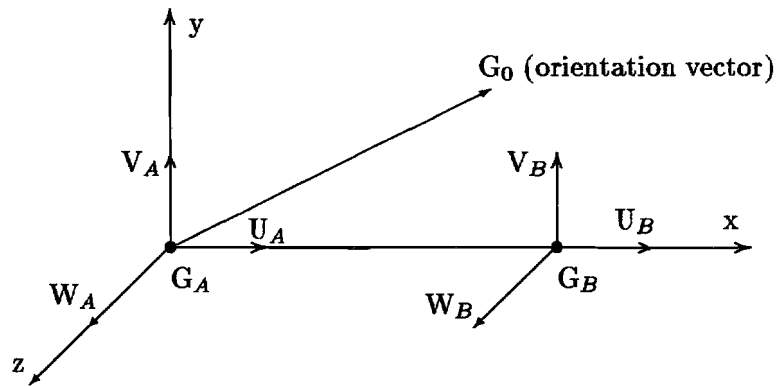


Figure 4.3.1 The Gap Element

The axial and lateral force-deflection curves of the GAP element are shown in Figs. 4.3.2 and 4.3.3. Properties that characterize these curves are specified in the Bulk Data PGAP.

1	2	3	4	5	6	7	8	9	10
PGAP	PID	U_0	F_0	K_a	K_b	K_t	μ_1	μ_2	
PGAP	2	0.	0.	1.E6	1.E-2	1.E5	0.	0.	+PG

	TMAX	MAR	TRMIN						+PG
+PG	0.	100.	0.001						

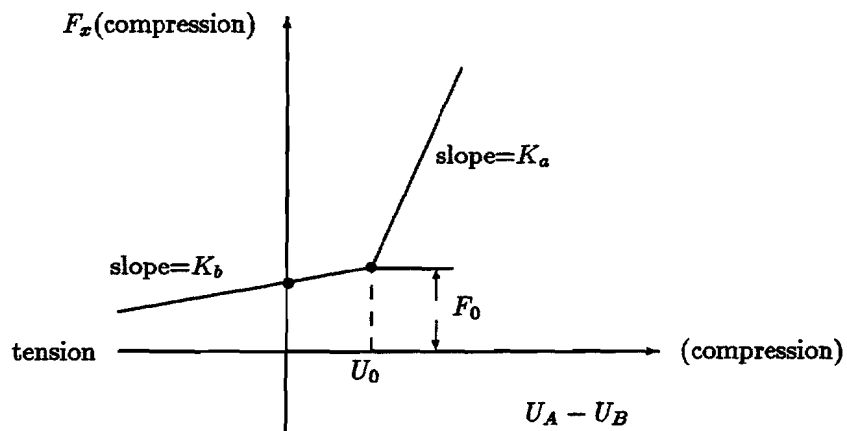


Figure 4.3.2 Gap Axial Force vs. Deflection
(slope K_a is used when $U_A - U_B \geq U_0$)

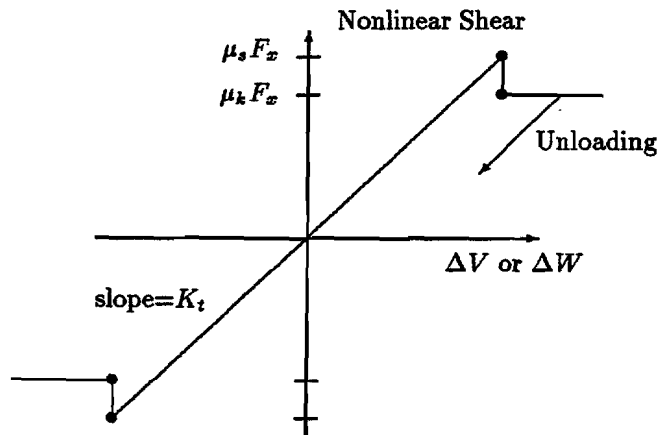


Figure 4.3.3 Gap Lateral Force vs. Deflection

The initial GAP opening is defined by U_0 (defaulted to 0), not by the separation distance. The preload is defined by F_0 (defaulted to 0). Notice that positive values of force and displacement represent compression in the GAP element coordinates. If the GAP is closed ($U_A - U_B \geq U_0$), the axial stiffness (K_a) has a large value relative to the adjacent structure. When the GAP is open, there is a small open stiffness (K_b , which is defaulted to $10^{-8}K_a$) in the axial direction. For most contact problems, K_a should be chosen to be three orders of magnitude higher than the stiffness of the neighboring grid points. The factor may be reduced to facilitate convergence at the expense of the accuracy. A much larger K_a value may slow down convergence or cause divergence, while a much smaller K_a value may result in an inaccuracy.

The transverse shear stiffness K_t (defaulted to $\mu_1 K_a$) is used when the friction is active upon contact. When the GAP is open, there is no transverse stiffness. When the GAP is closed and if there is friction, the GAP has the elastic stiffness (K_t) in the transverse direction until the lateral force exceeds the friction force and the slip starts to occur. If $K_t = 0$, the friction effect is ignored (equivalent to $\mu_1 = \mu_2 = 0$). The recommended range of the transverse stiffness value is $0.01K_a \leq K_t \leq K_a$.

When the adaptive GAP (default option) is used, μ_1 and μ_2 represent static and kinetic coefficients of friction μ_s and μ_k , respectively. If the nonadaptive GAP element is chosen, the anisotropic coefficients of friction (μ_y and μ_z) are the input for μ_1 and μ_2 . Directions y and z are defined by the orientation vector in the CGAP entry, which corresponds to the GAP element coordinate system.

Penalty values (K_a and K_t) are adjusted adaptively if $TMAX > 0.0$. There are two kinds of GAP elements: adaptive GAP and nonadaptive GAP. If $TMAX \geq 0.0$, the adaptive GAP element is chosen. When $TMAX = 0.0$ (default option), penalty values will not be adjusted but other adaptive features will be active (the gap induced stiffness update, gap induced bisection, and subincremental process). The value of $TMAX = -1.0$ selects the nonadaptive (old) GAP element. The recommended allowable penetration $TMAX$ is about 10% of the element thickness for plates and shells (or equivalent thickness for other elements) which are connected to the GAP element. If the structure is a massive solid, $TMAX$ should be determined such that the penetration is two or three orders less than the elastic deformation of the solid model. The initial estimate of $TMAX$ could be made such that the penetration is inconspicuous relative to the overall dimension of the structure, e.g., $\frac{1}{10000}$ of the overall length. The penalty value adjustment will thrash if $TMAX$ is too small, and no adjustment will occur if $TMAX$ is too large.

The maximum adjustment ratio MAR ($1 < MAR < 10^6$, defaulted to 100) is used only for penalty value adjustment of the adaptive GAP element. Upper and lower bounds of the adjusted penalty value are $MAR * K^0$ and K^0/MAR , respectively, where K^0 is the user-specified value for K_a or K_t . The parameter $TRMIN$ is a fractional number ($0.0 \leq TRMIN \leq 1.0$, defaulted to 0.001), representing a lower bound for the allowable penetration. The lower bound for the allowable penetration is computed by $TRMIN * TMAX$. The penalty values are decreased if the penetration is below the lower bound.

GAP element output may be obtained by the STRESS output request in the Case Control Section. An example of the GAP output format is shown below:

ELECTRO-MECHANICAL CLUTCH				SEPTEMBER 5, 1990 MSC/NASTRAN 8/30/90 PAGE 66					
DISPLACEMENT IN MM., FORCE IN NEWTON									
									SUBCASE 1
LOAD STEP = 2.00000E-01									
STRESSES (FORCES) IN GAP ELEMENTS (CGAP)									
ELEMENT - FORCES IN ELEM SYST - - DISPLACEMENTS IN ELEM SYST -									
ID	COMP-X	SHEAR-Y	SHEAR-Z	AXIAL-U	TOTAL-V	TOTAL-W	SLIP-V	SLIP-W	STATUS
31	1.2051E-06	0.0	0.0	1.2977E-02	-2.3891E-01	0.0	-2.3891E-01	0.0	SLIDE

The element forces and displacements are computed in the element coordinate system. Positive values of the axial force (F_x) and the axial displacement (U) represent compression. Lateral forces are computed for the frictional case, and printed as SHEAR-Y and SHEAR-Z. Lateral displacements in the y and z directions are printed under TOTAL-V, TOTAL-W (representing total lateral displacements) and SLIP-V, SLIP-W (representing translation of the slip center). When the GAP is open (regardless of friction) or sliding without friction, the total displacements (TOTAL-V, TOTAL-W) and the slip center displacements (SLIP-V, SLIP-W) have the same value. For the sticking or slipping cases with friction, the magnitude of the slip center translation (SLIP-V, SLIP-W) should be smaller than the magnitude of the total displacement. The last column of the output shows the GAP status (available in V67): OPEN, SLIDE, STICK or SLIP.

4.3.4 Theoretical Basis for Frictional Behavior

According to the Coulomb friction law, the reaction force due to friction is proportional to the normal force (F_N). Slipping does not occur as long as the external force exerted in the lateral direction (F_T) is smaller than the friction force, i.e.,

$$F_T \leq \mu_s F_N$$

where μ_s is the static coefficient of friction. Slipping starts when the external lateral force exceeds the static friction force. The friction force is reduced to the kinetic friction force ($\mu_k F_N$) during the slip, since $\mu_k < \mu_s$ in general. In order to simulate the stick-slip behavior of the current GAP element, the frictional yield function is formed as follows:

$$f = \sqrt{F_y^2 + F_z^2} - \mu_s F_x \quad (4.3.9)$$

with

$$\begin{Bmatrix} F_y \\ F_z \end{Bmatrix} = K_t \begin{Bmatrix} v - v_s \\ w - w_s \end{Bmatrix} \quad (4.3.10)$$

where K_t is a penalty value (representing an elastic stiffness) during sticking, and (v_s, w_s) represents a translation of the slip center in the y-z plane. The GAP is sticking, if $f < 0$.

When the GAP starts to slip ($f \geq 0$), the lateral displacement can be decomposed into two parts: recoverable sticking part (denoted by superscript st) and irrecoverable slipping part (denoted by superscript sl); i.e.,

$$\begin{Bmatrix} \Delta v \\ \Delta w \end{Bmatrix} = \begin{Bmatrix} \Delta v^{st} \\ \Delta w^{st} \end{Bmatrix} + \begin{Bmatrix} \Delta v^{sl} \\ \Delta w^{sl} \end{Bmatrix} \quad (4.3.11)$$

Slipping is similar to the plastic deformation [4.11]. Adopting the associated flow rule, the slip displacements can be defined in terms of the effective slip increment ($\Delta\bar{\gamma}^{sl}$) by

$$\begin{Bmatrix} \Delta v^{sl} \\ \Delta w^{sl} \end{Bmatrix} = \Delta\bar{\gamma}^{sl} \frac{1}{\mu_k F_x} \begin{Bmatrix} F_y \\ F_z \end{Bmatrix} \quad (4.3.12)$$

where

$$\mu_k F_x = \sqrt{F_y^2 + F_z^2}$$

and

$$\Delta\bar{\gamma}^{sl} = \sqrt{\Delta v^{sl2} + \Delta w^{sl2}}$$

Notice that slipping occurs along the direction normal to the slip locus, where (F_y, F_z) represent the forces in the normal direction to the slip locus.

The slip yield surface, which can be represented by a circular cone in the element force space, is kept up-to-date by updating slip forces and the slip center. Substituting Eqs. (4.3.11) and (4.3.12) into (4.3.10), we obtain

$$\begin{Bmatrix} F_y \\ F_z \end{Bmatrix} = \frac{1}{1 + \Delta\bar{\gamma}^{sl} \frac{K_t}{\mu_k F_x}} \begin{Bmatrix} F_y^{trial} \\ F_z^{trial} \end{Bmatrix} \quad (4.3.13)$$

where

$$\begin{Bmatrix} F_y^{trial} \\ F_z^{trial} \end{Bmatrix} = K_t \begin{Bmatrix} \Delta v \\ \Delta w \end{Bmatrix}$$

Substituting Eq. (4.3.13) into Eq. (4.3.9) with $f = 0$, we can express $\Delta\bar{\gamma}^{sl}$ as below:

$$\Delta\bar{\gamma}^{sl} = \frac{\sqrt{F_y^{trial2} + F_z^{trial2}} - \mu_k F_x}{K_t} \quad (4.3.14)$$

Combining Eq. (4.3.13) and (4.3.14), the final form of Eq. (4.3.13) becomes

$$\begin{Bmatrix} F_y \\ F_z \end{Bmatrix} = \frac{\mu_k F_x}{\sqrt{F_y^{trial2} + F_z^{trial2}}} \begin{Bmatrix} F_y^{trial} \\ F_z^{trial} \end{Bmatrix} \quad (4.3.15)$$

The slip center must be updated accordingly using Eq. (4.3.12), (4.3.14) and (4.3.15):

$$\begin{Bmatrix} v_s \\ w_s \end{Bmatrix} = \begin{Bmatrix} v \\ w \end{Bmatrix} - \frac{1}{K_t} \begin{Bmatrix} F_y \\ F_z \end{Bmatrix} \quad (4.3.16)$$

4.3.5 Subincremental Algorithm for Stiffness Update and Bisection

In the absence of friction, the GAP element may change its status from open to sliding and vice versa. When the friction is present, the status of the GAP element may change from open to stick and further to slip, and vice versa. General principles adopted in the adaptive stiffness matrix update and bisection strategies induced by GAP elements are:

- Allow only one state change in any GAP element within each increment. Bisect the load or time step if the GAP undergoes two state changes (e.g., open to slip) in a single step.
- Update the stiffness matrix when the GAP changes its status (e.g., open to stick, or slip to stick).
- Subdivide the increment for the slip process [4.11].

In order to achieve the adaptability, it is necessary to use the subincremental approach within a global increment. Furthermore, slipping with friction is a path-dependent process and the subincremental scheme allows a more accurate solution to the path-dependent process.

Two major steps are required to implement the subincrement method. Let the displacement increment for the i -th iteration be defined as follows:

$$\begin{Bmatrix} \Delta u \\ \Delta v \\ \Delta w \end{Bmatrix} = \begin{Bmatrix} u^i - u_n \\ v^i - v_n \\ w^i - w_n \end{Bmatrix}$$

where n represents the previously converged step. The first step is to find the open/close boundary. Assume that the GAP closes from the opening state at the i -th iteration. The scale factor α ($0 \leq \alpha \leq 1$) is used to determine the point at which the GAP starts to close as follows:

$$\alpha = \begin{cases} \frac{u_0 - u_n}{\Delta u} & \text{if } u_n \leq u_0 \leq u^i \quad (\text{open} \rightarrow \text{closed}) \\ 0 & \text{if } u_n \geq u_0 \quad (\text{closed} \rightarrow \text{closed}) \end{cases} \quad (4.3.17)$$

where u_0 represents the initial opening of the GAP element.

The second step is to find the stick/slip boundary. Assume that the GAP changes status from stick to slip at the i -th iteration. The scale factor β ($0 \leq \beta \leq 1$) is used to determine the point where the GAP begins to slip. The frictional yield function should be satisfied ($f = 0$) at the stick/slip boundary, i.e.,

$$\mu_s(F_x^n + \beta K_a \Delta u) = \sqrt{(F_y^n + \beta K_t \Delta v)^2 + (F_z^n + \beta K_t \Delta w)^2} \quad (4.3.18)$$

which can be rearranged to solve for β , resulting in a quadratic function of β :

$$a\beta^2 + 2b\beta + c = 0 \quad (4.3.19)$$

with

$$\begin{aligned} a &= (\mu_s K_a \Delta u)^2 - K_t^2 (\Delta v^2 + \Delta w^2) \\ b &= \mu_s^2 F_x^n K_a \Delta u - K_t (F_y^n \Delta v + F_z^n \Delta w) \\ c &= \mu_s^2 F_x^{n2} - F_y^{n2} - F_z^{n2} \end{aligned}$$

where F_x^n , F_y^n and F_z^n represent GAP internal forces at the last converged solution. It can be shown that the quadratic equation is guaranteed to produce a real and positive root. Since the GAP is slipping at the i -th iteration,

$$f = \sqrt{F_y^{i2} + F_z^{i2}} - \mu_k F_x^i > 0 \quad (4.3.20)$$

Expanding Eq. (4.3.20) using the relation in Eq. (4.3.19), the following inequalities can be obtained:

$$a + 2b + c < 0 \quad \text{and} \quad b^2 - ac > \left(\frac{a-c}{2}\right)^2 \geq 0$$

These inequalities lead to a proper root

$$\beta = \frac{-b - \sqrt{b^2 - ac}}{a}$$

and for $a \simeq 0$

$$\beta = -\frac{c}{2b}$$

Once the initial slip point is found, the slip region is subdivided into m subincrements ($1 < m \leq 10$). The number of subdivisions are determined adaptively by

$$m = \text{INT}\left(\frac{f^i}{0.5 * \mu_k * F_x}\right) + 1 \quad (4.3.21)$$

The displacement increment for each subincrement becomes

$$\begin{Bmatrix} \Delta u^m \\ \Delta v^m \\ \Delta w^m \end{Bmatrix} = \frac{1}{m}(1 - \alpha)(1 - \beta) \begin{Bmatrix} u^i - u_n \\ v^i - v_n \\ w^i - w_n \end{Bmatrix}$$

The GAP normal force (F_x) is gradually increased, and the the slip force (F_y and F_z) is updated in m subincrements. After the subincremental process, the element stiffness and the slip center are updated. The subincremental process is schematically described in Fig. 4.3.4.

It is noted that the “slip-to-slip” case is catagorized into two groups. One is the slip in the same direction and the other is the slip in the opposite direction. The condition for the slip in the same direction can be determined by the dot product of the last converged friction force and the friction force increment, i.e.,

$$[F_y^n (F_y^{trial} - F_y^n) + F_z^n (F_z^{trial} - F_z^n)] \geq 0$$

Slipping in the same direction is also signified by a double root of β at zero. If the slip occurs in the opposite direction, β has two roots with one being zero. In this case, the root at zero is discarded and the subincrement is processed from the slip locus at the opposite side signified by nonzero β .

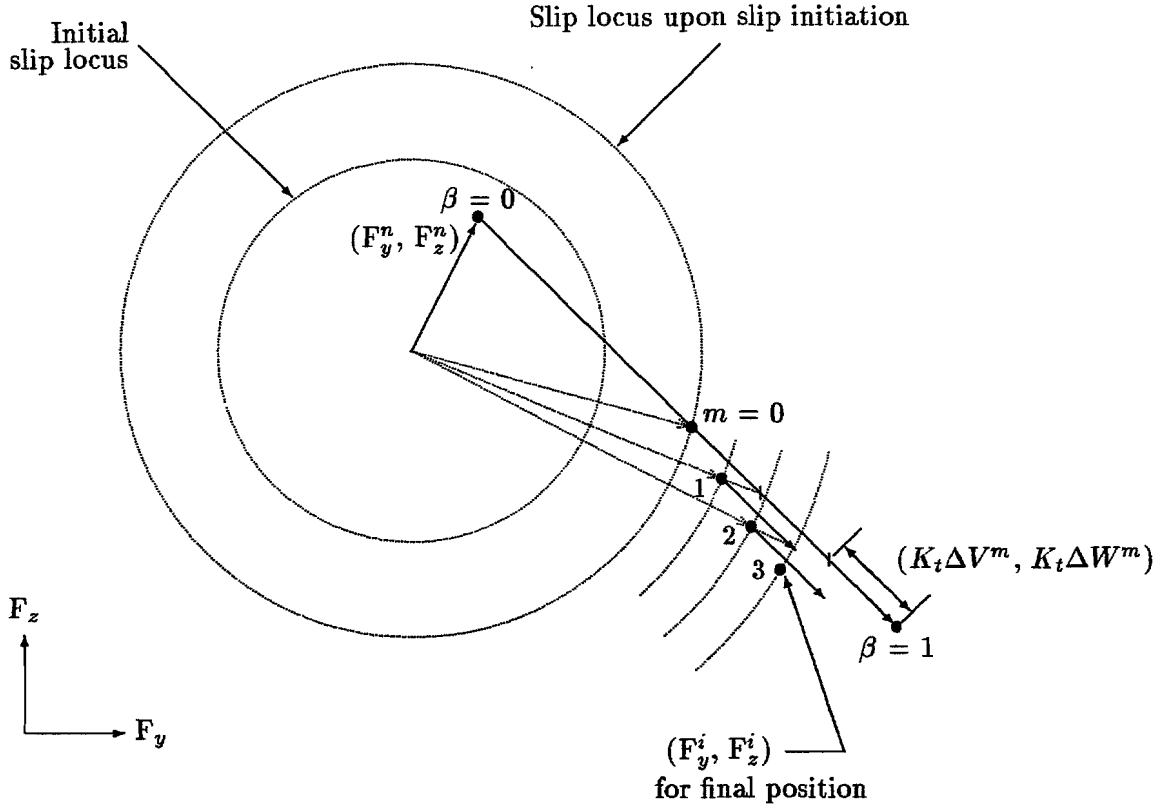


Figure 4.3.4 Schema for Subincremental Process ($m = 3$)

4.3.6 Adaptive Adjustment of Penalty Values

Penalty values are adaptively adjusted upon convergence for the next increment. If the penalty value is too small and produce an appreciable penetration, the accuracy of the solution may not be acceptable. Then the penalty value (K_a) should be increased to reduce the penetration. On the other hand, numerical difficulties arise if the penalty values are set too high. It is highly desirable to reduce the penalty values in such cases to improve the numerical performance.

The adaptive penalty value adjustment is designed to adjust penalty values automatically, if necessary, using the user-specified initial values and the maximum allowable penetration

(TMAX). If the GAP penetration exceeds the allowable value supplied by the user, penalty values (K_a and K_t) are increased, and vice versa if the penetration is smaller than the minimum allowable value, a fraction of TMAX. The adjustments are made at converged points. It is necessary to introduce a stepping function in order to avoid thrashing [4.12]. The stepping function is designed such that the adjustment ratio is in powers of 10, i.e.,

$$K^{n+1} = f(R)K^n \quad (4.3.22)$$

where $f(R)$ is ..., 0.1, 1., 10, ... and n denotes the incremental step with

$$R = \frac{u - u_0}{\text{TMAX}} \quad \text{if} \quad u - u_0 > \text{TMAX}$$

or

$$R = \frac{u - u_0}{\text{TMAX} * \text{TRMIN}} \quad \text{if} \quad u - u_0 < \text{TMAX} * \text{TRMIN}$$

where TRMIN represents the user-specified parameter to define a lower bound for allowable penetration.

It is noted that two penalty values (K_a and K_t) are adjusted by the same ratio, starting from the user-specified initial values. Since the ratio of K_a to K_t constitutes the slip criterion, an incompatibility will occur in the constitutive relations between two consecutive solutions if K_a and K_t are adjusted independently. The proper ratio between two penalty values can be established logically. Assume that the lateral displacement under the sticking condition is ϵ when the GAP starts to slip. The slip condition requires that

$$\mu_k K_a (u - u_0) = K_t \epsilon$$

If it is desired to limit the lateral displacement under the sticking condition (ϵ) by the same magnitude as the normal penetration ($u - u_0$), the magnitude of shear stiffness K_t has to be related to K_a by

$$K_t \simeq \mu_k K_a. \quad (4.3.23)$$

If the user inputs TMAX = 0, the penalty value is fixed and the adjustment algorithm is not invoked. The penalty value adjustment is limited by an upper and lower bound defined by the user-specified maximum adjustment ratio (MAR), i.e.,

$$\frac{K^0}{\text{MAR}} \leq K \leq \text{MAR} * K^0 \quad (4.3.24)$$

where K^0 is the user-specified value for K_a or K_t . The global stiffness matrix is updated whenever the penalty values are adjusted.

4.3.7 Non-adaptive GAP Element with Anisotropic Friction

The original GAP element can have an anisotropic friction by specifying direction-dependent friction coefficients associated with the GAP orientation. When the GAP is in contact (closed), the compressive force in the axial direction is defined by

$$F_x = F_0 + K_a(u - u_0) \quad (4.3.25)$$

where F_0 is the preload and u_0 is the initial GAP opening. The slip yield function may be modified for anisotropic friction as follows:

$$f = \sqrt{\left(\frac{F_y}{\mu_y}\right)^2 + \left(\frac{F_z}{\mu_z}\right)^2} - F_x \quad (4.3.26)$$

where μ_y and μ_z are coefficients of friction in the y and z direction. If the GAP is sticking, the friction forces become

$$\begin{Bmatrix} F_y \\ F_z \end{Bmatrix} = \frac{2K_t}{\mu_y^2 + \mu_z^2} \begin{Bmatrix} \mu_y^2(v - v_s) \\ \mu_z^2(w - w_s) \end{Bmatrix} \quad (4.3.27)$$

with the associated stiffness as follows:

$$\tilde{K} = \begin{bmatrix} K_a & 0 & 0 \\ \frac{2\mu_y^2 K_t}{\mu_y^2 + \mu_z^2} & 0 & 0 \\ sym & \frac{2\mu_z^2 K_t}{\mu_y^2 + \mu_z^2} & 0 \end{bmatrix} \quad (4.3.28)$$

In the case of slipping, the lateral forces are given by

$$\begin{Bmatrix} F_y \\ F_z \end{Bmatrix} = \frac{F_x}{\sqrt{\mu_y^2(v - v_s)^2 + \mu_z^2(w - w_s)^2}} \begin{Bmatrix} \mu_y^2(v - v_s) \\ \mu_z^2(w - w_s) \end{Bmatrix} \quad (4.3.29)$$

The associated GAP stiffness is given by

$$\tilde{K} = \begin{bmatrix} K_a & 0 & 0 \\ \frac{F_x(\mu_y\mu_z(w-w_s))^2}{((\mu_y(v-v_s))^2 + (\mu_z(w-w_s))^2)^{1.5}} & -\frac{F_x(\mu_y\mu_z)^2(v-v_s)(w-w_s)}{((\mu_y(v-v_s))^2 + (\mu_z(w-w_s))^2)^{1.5}} & 0 \\ sym & \frac{F_x(\mu_y\mu_z(v-v_s))^2}{((\mu_y(v-v_s))^2 + (\mu_z(w-w_s))^2)^{1.5}} & \frac{F_x(\mu_y\mu_z)^2(v-v_s)(w-w_s)}{((\mu_y(v-v_s))^2 + (\mu_z(w-w_s))^2)^{1.5}} \end{bmatrix} \quad (4.3.30)$$

where the symmetry is enforced.

4.3.8 Verification and Validation: Bouncing Mass Problem

A 4-lb weight is resting on a spring-supported platform which is initially compressed 4 inches in the position shown in Fig. 4.3.5. The platform is released, and after moving 2 inches, the platform is halted by a stopper. The response of the weight is to be determined [4.13]. The spring modulus of the platform support is 10 lb/in.

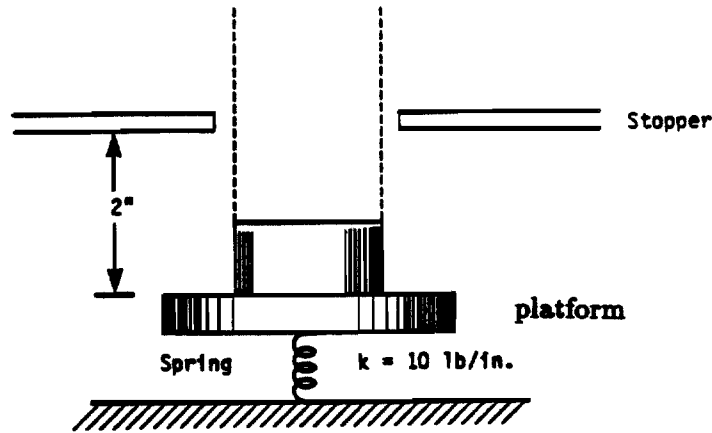


Figure 4.3.5 Physical Model

From the conservation of energy

$$\Delta V_e + \Delta V_g + \Delta T = 0. \quad (4.3.31)$$

The change in energy from the initial state when the platform is stopped is

$$\Delta V_e = \frac{1}{2}k(x_1^2 - x_0^2) = -60 \text{ lb-in}$$

$$\Delta V_g = mgh = 8 \text{ lb-in}$$

$$\Delta T = \frac{1}{2}m\dot{x}_0^2$$

from which the velocity of the mass at the time of departure from the platform can be obtained as

$$\dot{x}_0 = \sqrt{\frac{2}{m}(-\Delta V_e - \Delta V_g)} = 100.1798 \text{ in/sec.}$$

Up to this point, the motion of the mass is governed by the equation

$$m\ddot{x} + kx = -mg. \quad (4.3.32)$$

With the initial conditions $x_0 = -4$ in. and $\dot{x}_0 = 0$ in/sec, the response of the mass is expressed as

$$x = (x_0 + \frac{mg}{k}) \cos \omega_n t - \frac{mg}{k} \quad (4.3.33)$$

The elapsed time at the moment the platform hits the stopper may be obtained by solving the above equation for t with $x_1 = 2$ in, i.e.,

$$t_1 = \frac{1}{\omega_n} \cos^{-1} \left(\frac{x_1 + \frac{mg}{k}}{x_0 + \frac{mg}{k}} \right) = 0.0357 \text{ sec}$$

where $\omega_n = \sqrt{\frac{k}{m}} = 31.0644 \text{ rad/sec}$. As the mass departs from the platform, the motion of the mass is described by

$$h = \dot{x}_0 t - \frac{1}{2} g t^2 \quad (4.3.34)$$

relative to the stopper. Elapsed time when the mass reaches the peak is obtained as

$$t_2 = t_1 + \frac{\dot{x}_0}{g} = 0.0357 + 0.2595 = 0.2953 \text{ sec.}$$

The height of the mass at this moment is

$$h_{max} = \frac{\dot{x}_0^2}{2g} = 13.0 \text{ in}$$

from the stopper. The theory predicts that one full cycle will take 0.5905 sec.

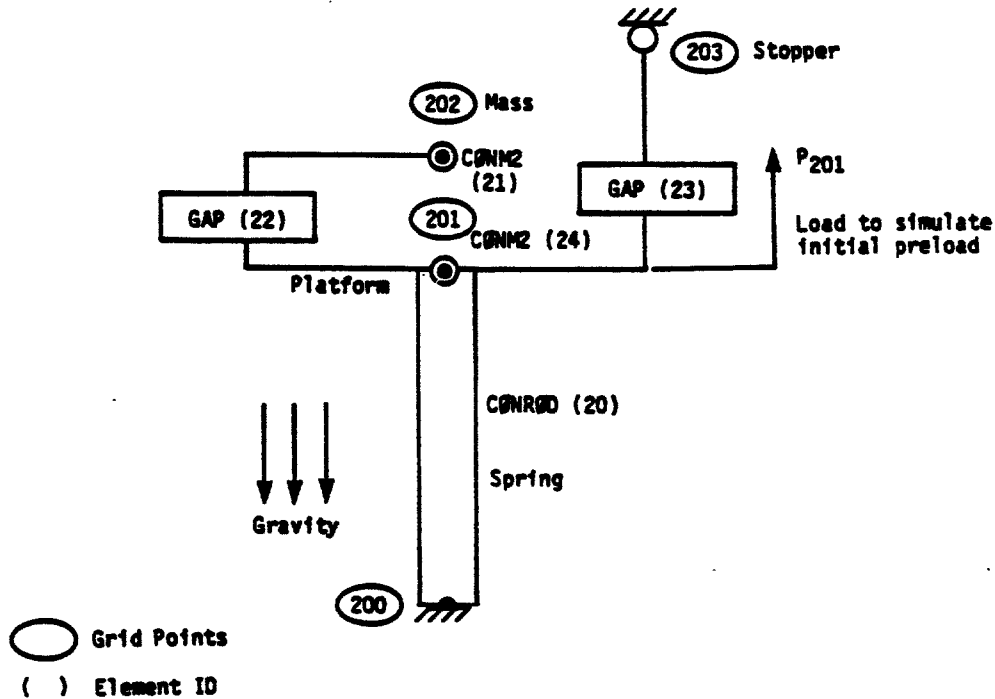


Figure 4.3.6 Symbolic Model

The model is depicted schematically in Fig. 4.3.6. The spring is modeled with a CONROD element, for which the stiffness is given as $k = \frac{AE}{l}$, with A (area) = 1.0 in², E (Young's modulus) = 100 psi, and l (length) = 10 in. The mass is modeled as CONM2 21, which is connected to the platform (GRID 201) by GAP 22. The stopper is represented by a fixed point (GRID 203), to which the platform is connected by GAP 23. The base (GRID 200) is fixed to the ground. An initial displacement (4 inches) is simulated by a step force (40 lbs) applied at GRID 201. The gravity field is applied by a GRAV entry. Static loads are converted to the dynamic loads using Bulk Data LSEQ and TLOAD2 with a Case Control command LOADSET. A small mass CONM2 24 (0.1 % of the mass block) is provided at the platform to suppress extraneous jittering. Small viscous damping (0.5 % at 31 rad/sec.) is introduced and adaptive time stepping is used. The input data listing is given in Table 4.3.1.

The solution to the model is obtained by the nonlinear transient analysis method (SOL 99). The analysis started with initial $\Delta t = 0.0025$ sec. for the duration of 1 sec. GAP elements had the initial stiffness of 10⁴ lb/in for the closed position with TMAX of 0.005 in. for adaptive adjustment. At the onset, the penalty value (K_a) of GAP 22 was adjusted to 10⁵. The bisection was also activated by the adaptive time stepping during the initial phase. After the initial settlements, the automatic time stepping algorithm quadrupled the time step size. The stiffness matrix was updated three times due to GAP status changes in the vicinity of $t = 0.035$ sec., $t = 0.545$ sec. and $t = 0.614$ sec. The automatic time stepping algorithm caused the reduction of Δt to a half of the specified value (0.00125 sec.) while the platform was in motion around $t = 0.58$ sec.

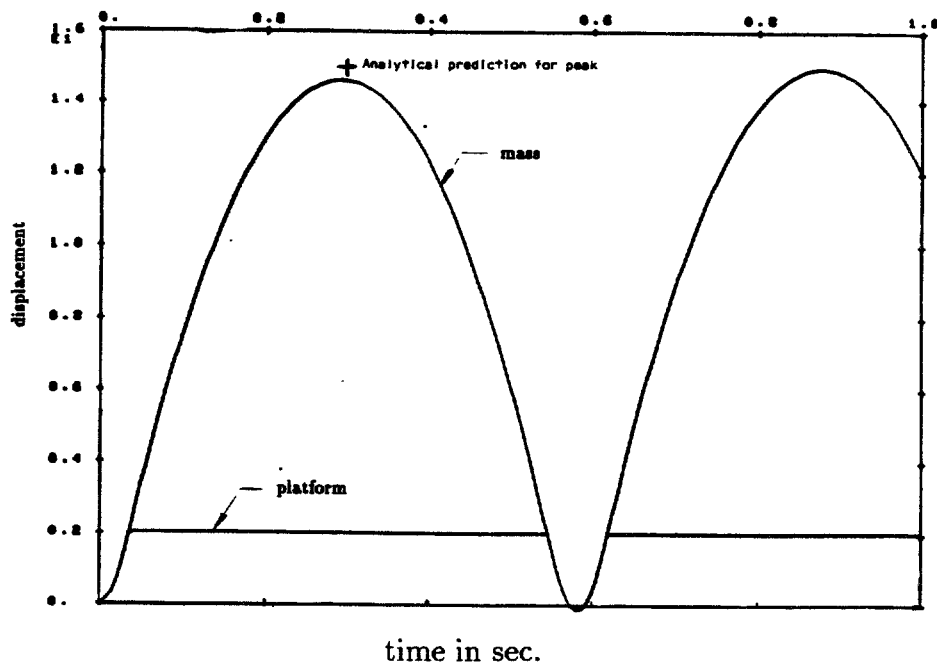


Figure 4.3.7 Displacement Response

The displacement response in Fig. 4.3.7 shows a peak height of approximately 14.56 inches and the period of approximately 0.583 sec. Discrepancies between this solution and the analytical solution may be attributed to the following:

- Some momentum is dissipated due to the damping and the small mass introduced at the platform.
- The artificial stiffness introduced to the GAP elements changes the dynamic characteristics slightly.
- Due to the finite interval of the time increment, the time stepping scheme can not be synchronized exactly with the GAP closing/opening.

In order to verify the adaptive penalty value adjustment algorithm, a parametric study is conducted with the penalty value (closed stiffness) of GAP 22. The value was varied from 10^3 to 10^7 with $TMAX = 0.005$. The effectiveness of adaptive penalty value adjustment is shown in Fig. 4.3.8. Notice that penalty values were accepted in a wide range (10^4 to 10^6) by the adaptive adjustment scheme. A tighter tolerance ($TMAX = 0.001$) was also tested to examine its effect. Indeed, the penalty value was adjusted to the narrower range (10^5 to 10^6).

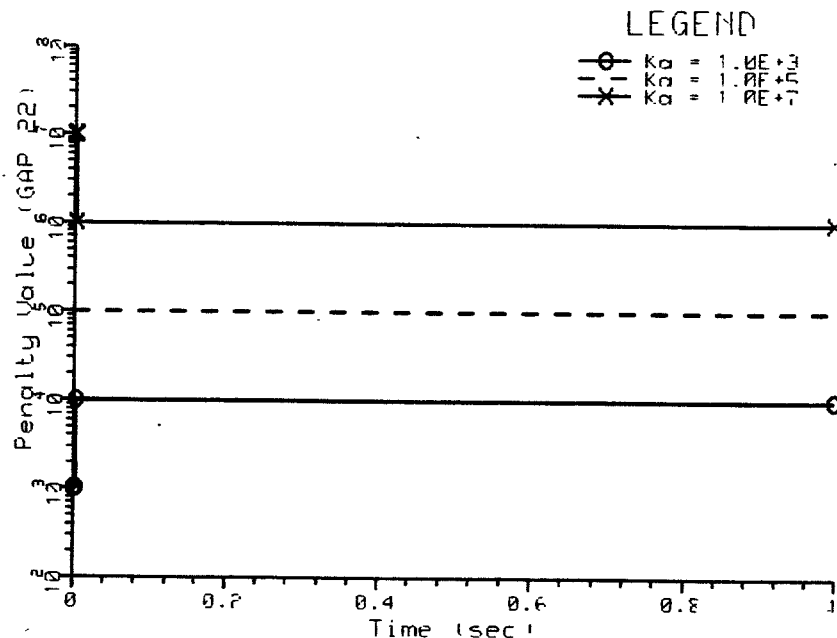


Figure 4.3.8 Adaptive Penalty Value Adjustment on GAP 22

Table 4.3.1 Input Data Listing for Bouncing Mass Problem

```

ID      VGAPM,V66      $ SHL 12-11-90
TIME    5              $ CPU MINUTES FOR VAX
DIAG    8,50          $ PRINT MATRIX TRAILERS AND ITERATIONS
SOL 99
CEND
TITLE = GAP VERIFICATION
SUBTITLE = NONLINEAR TRANSIENT ANALYSIS
SEALL = ALL
ECHO = UNSORT
  SET 1 = 201,202
  SET 3 = 22,23
  SET 4 = 20          $ CONROD
    DISP = 1
    STRESS = 3 $ FOR GAP FORCES
    ELFORCE = 4
LOADSET = 20
SUBCASE 1
  DLOAD = 210
  TSTEPNL = 21
SUBCASE 2
  DLOAD = 210
  TSTEPNL = 21
OUTPUT(XY PLOT)
  CSCALE = 1.3
  PLOTTER = NAST
  XTITLE = TIME IN SEC
  YTITLE = DISP IN INCH
  XY PLOT DISP RESP/201(T1),202(T1)
BEGIN BULK
$ SOLUTION STRATEGY
TSTEPNL 21      200      .0025      1      ADAPT      PW      +TS1
+TS1
+TS2      0
$ PROPERTIES
PARAM  W4      31.
MAT1   1      100.      .3      .01
PGAP   3      .      1.E+4      +PG3
+PG3   5.-3
PGAP   4      2.      1.E+4      +PG4
+PG4   5.-3
$ LOADING
FORCE  201      201      40.      1.      0.      0.
GRAV   200      386.      -1.      0.      0.
LOAD   202      1.      1.      200      1.      201
LSEQ   20      301      202
TLOAD1 210      301      0      220
TABLED1 220
+TAB1  0.      0.      .001      1.      100.      1.      ENDT

```

```

$ BOUNDARY CONDITIONS
GRDSET                                23456
$ GEOMETRY
GRID    200          -6.    0.    0.    123456
GRID    201           4.    0.    0.
GRID    202          4.1    0.    0.
GRID    203           6.    0.    0.    123456
$ CONNECTIVITY
CONROD  20    200    201    1    1.
CONM2   24    201          1.-5
CONM2   21    202          .01036
CGAP    22     3    201    202    0.    1.    0.
CGAP    23     4    201    203    0.    1.    0.
ENDDATA

```

4.3.9 Verification and Validation: Vibration with Coulomb Damping

A mass block (m) connected to a spring (k) is resting on a frictional surface as shown in Fig. 4.3.9. The motion of the mass block is governed by

$$m\ddot{u} + ku = -F \quad \text{for} \quad \dot{u} > 0. \quad (4.3.35)$$

$$m\ddot{u} + ku = F \quad \text{for} \quad \dot{u} < 0.$$

where F is the friction force [4.14].

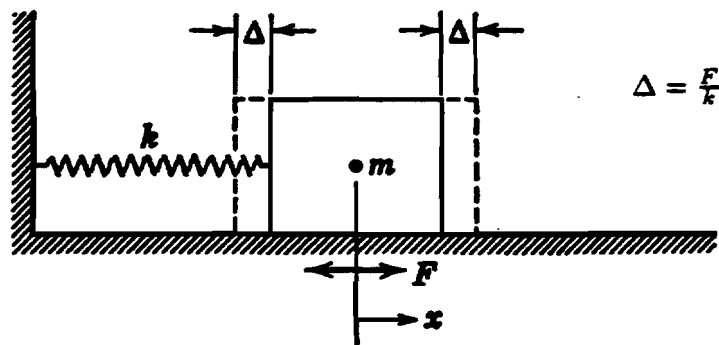


Figure 4.3.9 Physical Model for Vibration with Coulomb Damping [4.14]

Suppose the block is displaced by $u_0 \gg F/k$ and released. The solution while $\dot{u} < 0$ is

$$u = \frac{F}{k} + \left(u_0 - \frac{F}{k}\right) \cos \omega t \quad (4.3.36)$$

where $\omega = \sqrt{k/m}$. The mass block comes to a stop ($\dot{u} = 0$) at $\omega t = \pi$. For $\pi \leq \omega t \leq 2\pi$ while $\dot{u} > 0$, the solution is

$$u = -\frac{F}{k} - (u_0 - 3\frac{F}{k}) \cos(\omega t - \pi) \quad (4.3.37)$$

It is noted that the acceleration (\ddot{u}) is discontinuous at $t = \frac{\pi}{\omega}, \frac{2\pi}{\omega}, \frac{3\pi}{\omega}, \dots$, while the velocity is continuous. From the observation that $u = -(u_0 - 2F/k)$ at $t = \frac{\pi}{\omega}$, and $u = -(u_0 - 4F/k)$ at $t = \frac{2\pi}{\omega}$, the decay is a linear decrement at the rate of $4F/k$ per cycle until $|u_{max}| \leq \frac{F}{k}$.

The solver's model uses a GAP element to simulate friction with $k_a = 10^6$, $k_t = 10^5$, $\mu_s = 0.3$ and $\mu_k = 0.2$. Let m (CONM2) be 100 kg in a gravity field of $1m/sec^2$, which gives rise to a normal force (F_N) of 100 N. A rod element is used to simulate an elastic spring with $k = 10^3$ N/m. The input data listing is given in Table 4.3.2. The analysis is performed in three subcases using SOL 99. The first two subcases use PARAM, TSTATIC, 1 for static analyses, which apply the preloads (gravity and a horizontal force of $P=220$ N), resulting in an initial displacement

$$u_0 = \frac{1}{k}(P - \mu_k F_N) = 0.2$$

The third subcase uses PARAM, TSTATIC, 0 for dynamic response upon release of the load P . The natural frequency (ω) of the system is 3.1623 rad/sec and the period is 1.987 sec. After the first cycle, the amplitude decreases to

$$u_{max} = u_0 - 4\frac{\mu_k F_N}{k} = 0.2 - 4 \times 0.02 = 0.12$$

After two and a half cycles, the mass block is stagnated. Ideally, the mass block should come to a stop with a residual displacement due to friction. However, because of the shear stiffness (k_t) in the GAP element, the mass block oscillates at $\omega_r = \sqrt{k_t/m} = 31.623$ rad/sec. This effect is shown as ripples after two and a half cycles in Fig. 4.3.10.

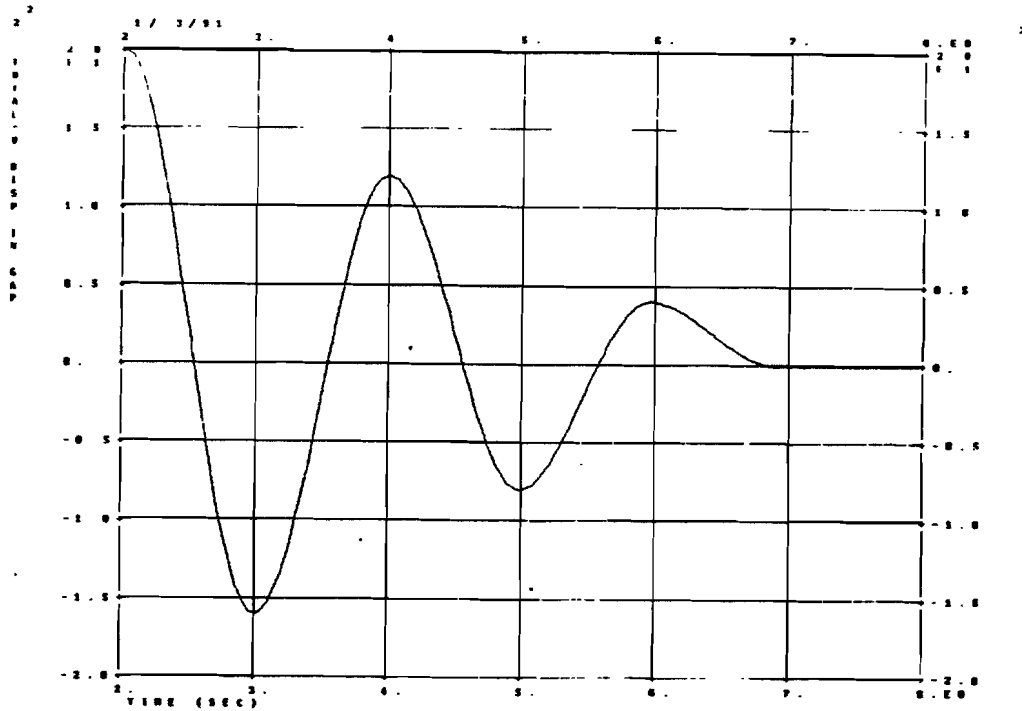


Figure 4.3.10 Horizontal Displacement Response

In order to examine the effects of the frictional force, the analysis is performed with some variations in the model. First, the mass (m) is increased to 366.7 kg and, in turn, the normal force (F_N) is increased to 366.7 N. Then the initial displacement (u_0) is decreased to 0.1467 and the natural frequency (ω) is reduced to 1.6514 rad/sec. In this case, the decay rate is such that the amplitude is reduced to zero in a half cycle ($\omega t = \pi$) as shown in Fig. 4.3.11. This is analogous to the critically-damped case in viscous damping. If the friction coefficients are also increased ($\mu_s = 0.45$ and $\mu_k = 0.3$) and the horizontal applied force (P) is increased to 300 N, the initial displacement (u_0) becomes 0.19 m. The mass will come to rest when $\dot{u} = 0$, which occurs at $t = \frac{\pi}{\omega} = 1.9$ sec. The residual displacement should have an asymptotic value of 0.03 from Eq. (4.3.36). As shown in Fig. 4.3.11, this is analogous to the over-damped case in viscous damping.

The effect of distinctive friction coefficients (static and kinetic) during the static loading is shown in Fig. 4.3.12. For the over-damped case, the GAP will be sticking up to the friction force of 165 N, governed by μ_s . As the block starts to slip, the friction force is reduced to 110 N, complying with μ_k . It is noted that an excessive number of load steps was taken during the static loading in order to show the reduction in the frictional force at the stick/slip transition. Despite the excessive steps, the peak at 165 N is not shown due to the discrete load increment size.

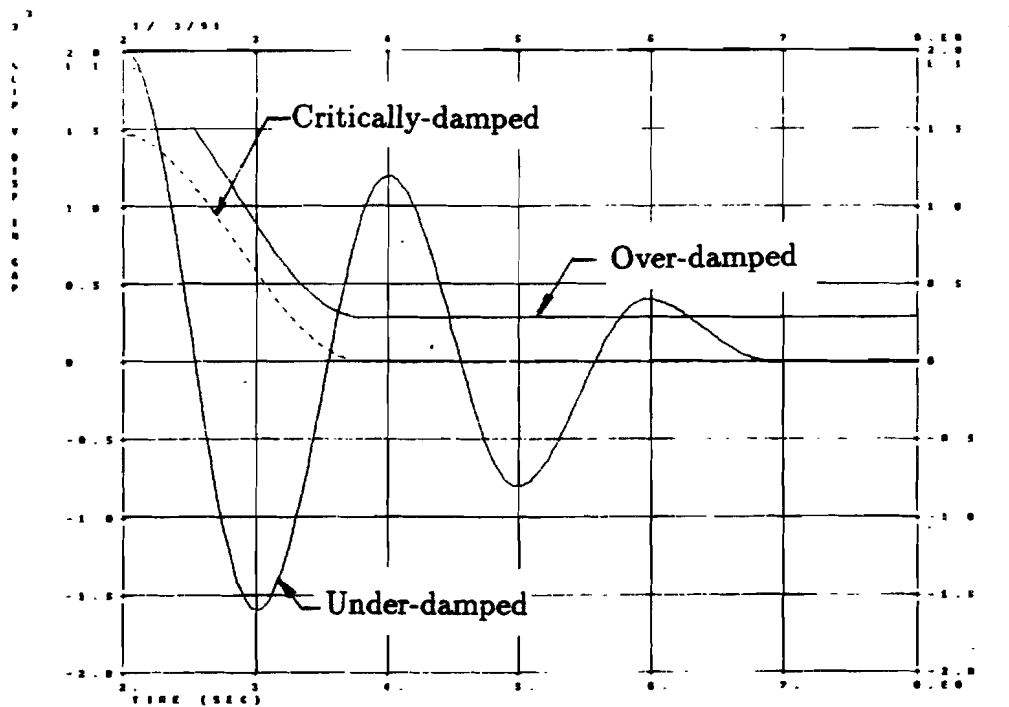


Figure 4.3.11 Comparison of Responses: Under-damped, Critically-damped, Over-damped

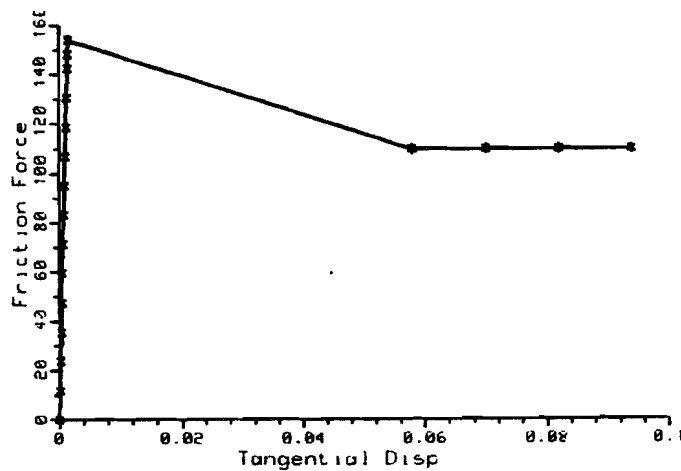


Figure 4.3.12 Horizontal Displacement during Static Loading

Table 4.3.2 Input Data Listing for Coulomb Damping Problem

```

ID      COULOMB, V67      $ SHL 12-21-90
TIME    5                $ CPU MINUTES FOR VAX
SOL      99              $ NONLINEAR TRANSIENT ANALYSIS
DIAG     8,50            $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = COULOMB FRICTION VIBRATION PROBLEM
SUBTITLE = NONLINEAR STATIC AND TRANSIENT ANALYSIS
LABEL = TIMOSHENKO, YOUNG, AND WEAVER; P. 198
ECHO = BOTH
SEALL = ALL
  SET 1 = 2, 12, 22
  SET 2 = 10, 110, 210
  SET 3 = 20, 120, 220
    DISP = 1
    VELO = 1
    ACCE = 1
    STRES = 2
    FORCE = 3
LOADSET = 1000
SUBCASE 10 $ COMPRESS GAP
  PARAM TSTATIC 1
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 20 $ SHEAR GAP
  PARAM TSTATIC 1
  DLOAD = 200
  TSTEPNL = 20
SUBCASE 30 $ DYNAMIC ANALYSIS
  PARAM TSTATIC 0
  DLOAD = 100
  TSTEPNL = 30
OUTPUT(XYPLOT)
  CSCALE = 1.5
  XTITLE = TIME (SEC)
  XGRID LINES = YES
  YGRID LINES = YES
  XMIN=2.
  XMAX=8.
  YTITLE = DISPLACEMENT AT THE MASS BLOCK
  XYPLOT DISP RESP /2(T1),12(T1),22(T1)
  YTITLE = VELOCITY AT THE MASS BLOCK
  XYPLOT VELO RESP /2(T1),12(T1)
  YTITLE = ACCELERATION AT THE MASS BLOCK
  XYPLOT ACCE RESP /2(T1)
  YTITLE = STRESS IN ROD
  XYPLOT STRESS RESP 30/20(2),120(2),220(2)
  YTITLE = SHEAR FORCE IN GAP
  XYPLOT STRESS RESP 30/10(3)

```

```

YTITLE = TOTAL-V DISP IN GAP
XYPLOT STRESS RESP 30/10(6)
YTITLE = SLIP-V DISP IN GAP
XYPLOT STRESS RESP 30/10(8),110(8),210(8)
BEGIN BULK
$ SOLUTION CONTROL
TSTEPNL 10      4      .25      1      ADAPT      20
TSTEPNL 20     25     .04      1      ADAPT      20
TSTEPNL 30     30     .2       1      ADAPT
$-----$
$ GEOMETRY AND CONNECTIVITY FOR OSCILLATORY SYSTEM
GRID  1          0.      0.      0.          123456
GRID  2          100.    0.      0.          3456
GRID  3          100.   -1.      0.          123456
CGAP  10      10      2       3       1.
CONROD 20      1       2      20      1.
CONM2  30      2          100.
$ PROPERTIES
PGAP  10      0.      0.      1.+6          1.+5      .3      .2
MAT1  20      1.+5
$-----$
$ GEOMETRY AND CONNECTIVITY FOR CRITICALLY DAMPED CASE
GRID  11          0.      0.      0.          123456
GRID  12          100.    0.      0.          3456
GRID  13          100.   -1.      0.          123456
CGAP  110     110     12      13      1.
CONROD 120     11     12     120     1.
CONM2  130     12          366.7
$ PROPERTIES
PGAP  110     0.      0.      1.+6          1.+5      .3      .2
MAT1  120     1.+5
$-----$
$ GEOMETRY AND CONNECTIVITY FOR OVER-DAMPED CASE
GRID  21          0.      0.      0.          123456
GRID  22          100.    0.      0.          3456
GRID  23          100.   -1.      0.          123456
CGAP  210     210     22      23      1.
CONROD 220     21     22     220     1.
CONM2  230     22          366.7
$ PROPERTIES
PGAP  210     0.      0.      1.+6          1.+5      .45      .3
MAT1  220     1.+5
$-----$
$ LOADING
LSEQ  1000     110     111
GRAV  111          1.      0.     -1.      0.
TLOAD2 100     110          0      0.     100.
$
LSEQ  1000     210     211

```

```

FORCE  211      2              220.    1.      0.      0.
FORCE  211     12              220.    1.      0.      0.
FORCE  211     22              300.    1.      0.      0.
DLOAD  200      1.      1.      100     1.      2000
TLOAD1 2000     210      0      201
TABLED1 201
+TBD21 0.      0.      1.      0.      2.      1.      ENDT
ENDDATA
+TBD21

```

4.3.10 Hertzian Problem: Contact Between Sphere and Rigid Plane

Mechanics of the frictionless contact between two elastic bodies were first solved by H. Hertz and published in 1881. Hertz presented solutions to a family of two-body contact problems with general curved surfaces for the loading normal to the surface. Stresses at the contact area depend on the deformation of the bodies in contact. Typical examples are ball and roller bearings, gears, cams, and rolling wheels. Hertz's analysis revealed that the pressure distribution on the contact surface between two curved bodies is represented by a semiellipsoid constructed on the surface of contact.

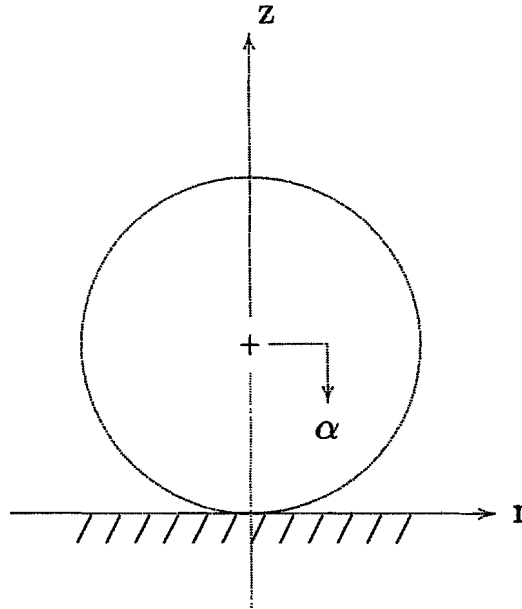


Figure 4.3.13 Contact Between Elastic Sphere and Rigid Plane

Let us confine our discussion to the contact of an elastic sphere with a rigid plane as shown in Fig. 4.3.13. The hemispherical pressure distribution on the surface of contact with a radius a gives the maximum pressure (q_0) at the center by

$$q_0 = \frac{3P}{2\pi a^2} \quad (4.3.38)$$

where P is the total compressive force. The solution for two balls in contact with radii R_1 and R_2 is given [4.15] with the radius of contact area as follows:

$$a = \left[\frac{3\pi}{4} \frac{P(k_1 + k_2)R_1R_2}{R_1 + R_2} \right]^{1/3} \quad (4.3.39)$$

and for the rigid body motion (or approach distance)

$$\alpha = \left[\frac{9\pi^2}{16} \frac{P^2(k_1 + k_2)^2(R_1 + R_2)}{R_1R_2} \right]^{1/3} \quad (4.3.40)$$

with

$$k_1 = \frac{1 - \nu_1^2}{\pi E_1} \quad \text{and} \quad k_2 = \frac{1 - \nu_2^2}{\pi E_2}$$

for two balls (1 and 2) with different Young's moduli and Poisson's ratios, respectively. For the contact of an elastic sphere with a rigid flat plane, Eqs. (4.3.39) and (4.3.40) are reduced with $R_2 = \infty$ and $E_2 = \infty$ to

$$a = \left[\frac{3(1 - \nu^2)}{4E} PR \right]^{1/3} = \sqrt{R\alpha} \quad (4.3.39a)$$

and

$$\alpha = \left[\frac{9}{16} \frac{P^2}{R} \frac{(1 - \nu^2)^2}{E^2} \right]^{1/3} \quad (4.3.40a)$$

where α is identical to the displacement of points on the sphere remote from the deformed area.

Stresses at the center of the contact surface are given by

$$\sigma_r = \sigma_\theta = \frac{1}{2}(1 + 2\nu)\sigma_z \quad \text{and} \quad \sigma_z = -q_0$$

The maximum shear stress occurs on the z -axis at a distance of $0.47a$ below the surface of contact, given as

$$\tau_{rz} = \tau_{\theta z} = 0.31q_0$$

The maximum tensile stress occurs at the circular contact boundary, given as

$$\sigma_r = \frac{1 - 2\nu}{3}q_0$$

At this boundary, the circumferential stress (σ_θ) is compressive with the same magnitude as σ_r , which represents a pure shear stress state.

For the static analysis, a symmetric half of a 10 degree sector of the sphere is modeled by HEXA and PENTA elements with axisymmetric boundary conditions as shown in Fig. 4.3.14. Contact with a rigid plane is simulated by adaptive GAP elements. Uniform vertical displacement of the symmetric plane is maintained by tying the vertical displacements of all the nodes

on the top plane (loading plane) by MPC relations to the one at the centerline (z-axis), on which 1/36 of the total compressive force is applied.

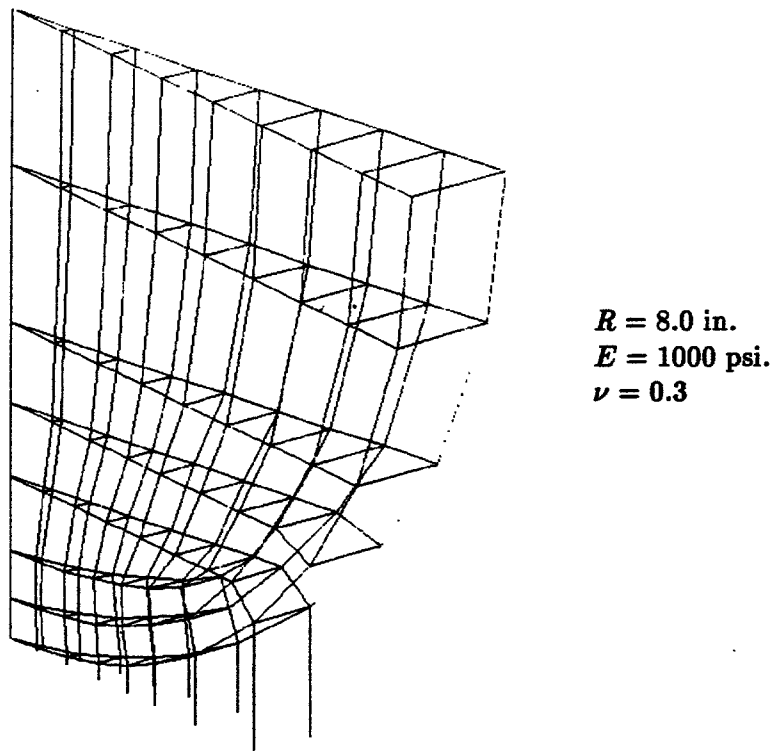


Figure 4.3.14 Model for Static Analysis

The static analysis was performed in 20 increments up to the maximum loading ($P = 2160$). The analysis ran to completion in 27 iterations with four GAP induced stiffness updates. The deformed shape at the maximum loading is shown in Fig. 4.3.15 in comparison with the undeformed model. Two additional cases were analyzed: one with geometric nonlinearity (PARAM,LGDISP) and the other with friction ($\mu_s = 0.3$ and $\mu_k = 0.2$). Deformed shapes for other cases were not discernible from the one in Fig. 4.3.15. All cases ran to completion without requiring any user intervention. In the frictional case, however, bisections were activated three times during the initial phase, and 30 incremental steps were required.

The radius of contact surface and the approach distance (vertical displacement at the loading plane) are plotted as functions of the total force (P) in Fig. 4.3.16, which shows this solution in comparison with the Hertz solution. Notice that the effects of friction and geometric nonlinearity are gradually manifested as the load increases. It is noted that the frictional case appears more accurate at the initial stage. This is because the frictional case took much smaller increments due to bisections during the initial phase, and the point of contact could be more closely predicted.

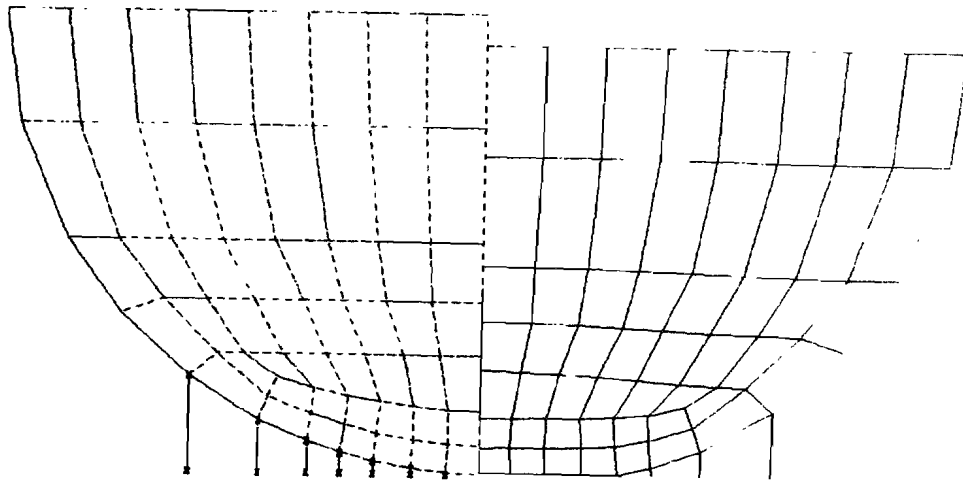


Figure 4.3.15 Undeformed Shape vs. Deformed Shape at $P = 2160$

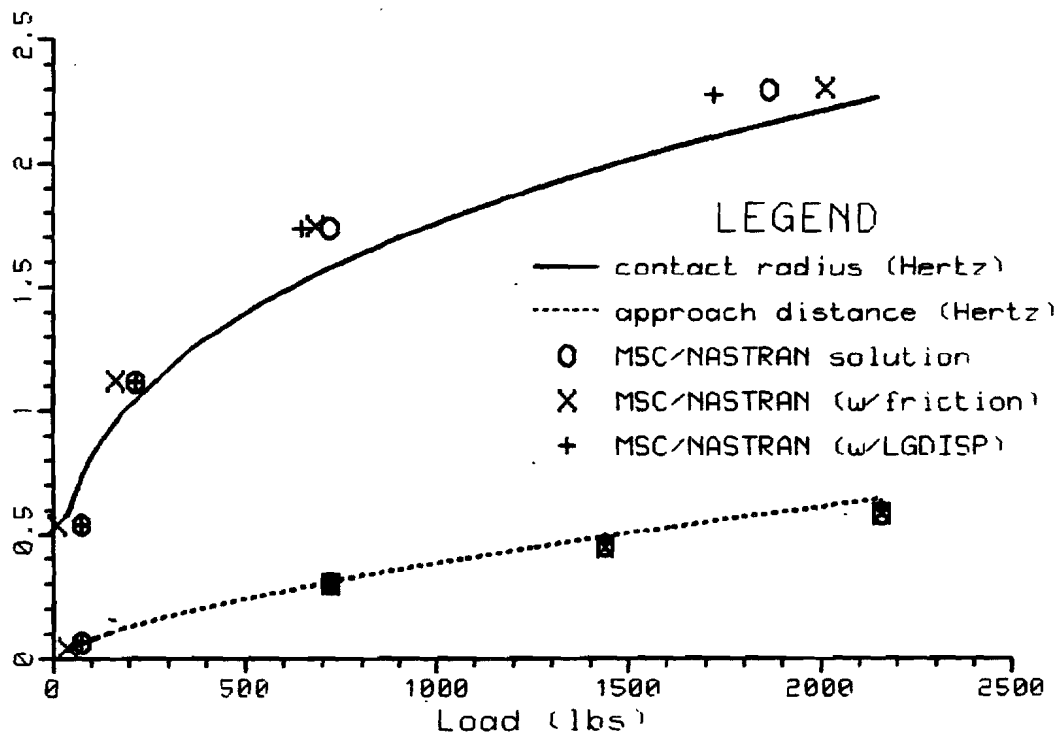


Figure 4.3.16 Contact Radius & Approach vs. Total Force in Static Analysis

4.3.11 Dynamic Case: Impact of a Sphere with Rigid Plane

The Hertzian problem shown in Fig. 4.3.13 is considered for dynamic response. In the impact of an elastic sphere with the rigid plane, the duration of contact is very long compared to the period of lowest mode of vibration in the sphere. Vibrations and the stress wave can therefore be neglected, and the statical solutions in Eqs. (4.3.39a) and (4.3.40a) are assumed to be valid during impact [4.15]. The applied force P can be obtained from

$$m\dot{v} = m\ddot{\alpha} = -P$$

where m and v are the mass and the velocity of the sphere, respectively. Introducing a notation

$$n = P\alpha^{-3/2} = \frac{4}{3} \frac{ER^{1/2}}{(1-\nu^2)}$$

the acceleration of approach becomes

$$\ddot{\alpha} = -\frac{n}{m}\alpha^{3/2}$$

from which

$$\frac{1}{2}(\dot{\alpha}^2 - v_0^2) = -\frac{2}{5} \frac{n}{m} \alpha^{5/2} \quad (4.3.41)$$

where v_0 is the velocity of the sphere at the beginning of impact.

The value of the maximum compressive force is obtained by

$$P_{max} = n[\alpha_{max}]^{3/2} \quad (4.3.42)$$

where α_{max} at the maximum compression can be found by substituting $\dot{\alpha} = 0$ into Eq. (4.3.41), resulting in

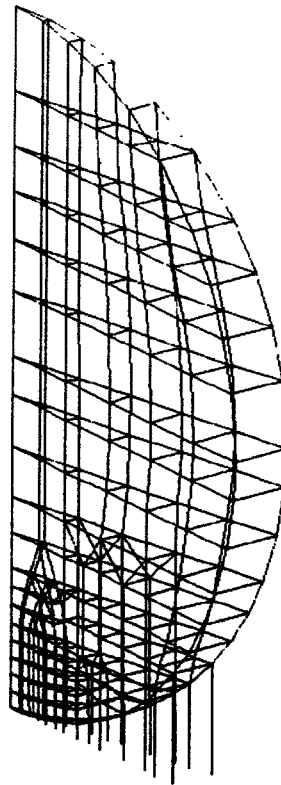
$$\alpha_{max} = \left[\frac{5m}{4n} v_0^2 \right]^{2/5} = \left[\frac{5}{4} \pi \rho \frac{1-\nu^2}{E} v_0^2 \right]^{2/5} R \quad (4.3.43)$$

where ρ is the mass density of the sphere. The duration of impact is given by

$$t = 2.94 \frac{\alpha_{max}}{v_0} = 2.94 \left[\frac{5}{4} \pi \rho \frac{1-\nu^2}{E} \right]^{2/5} R v_0^{-1/5} \quad (4.3.44)$$

For the impact analysis, a 10 degree sector of the sphere is modeled by HEXA and PENTA elements with axisymmetric boundary conditions as shown in Fig. 4.3.17. Contact with a rigid plane is simulated by adaptive GAP elements. The initial velocity of 10 in/sec. is specified for the sphere at the beginning of the impact. The analysis started with the initial time step size of 1 msec. and progressed to completion without any difficulty. The adaptive time stepping algorithm adjusted Δt to 2 and 4 msec. during the initial stage. The bisection was activated during the maximum impact to adjust Δt to 0.25 msec. In the meantime the GAP induced

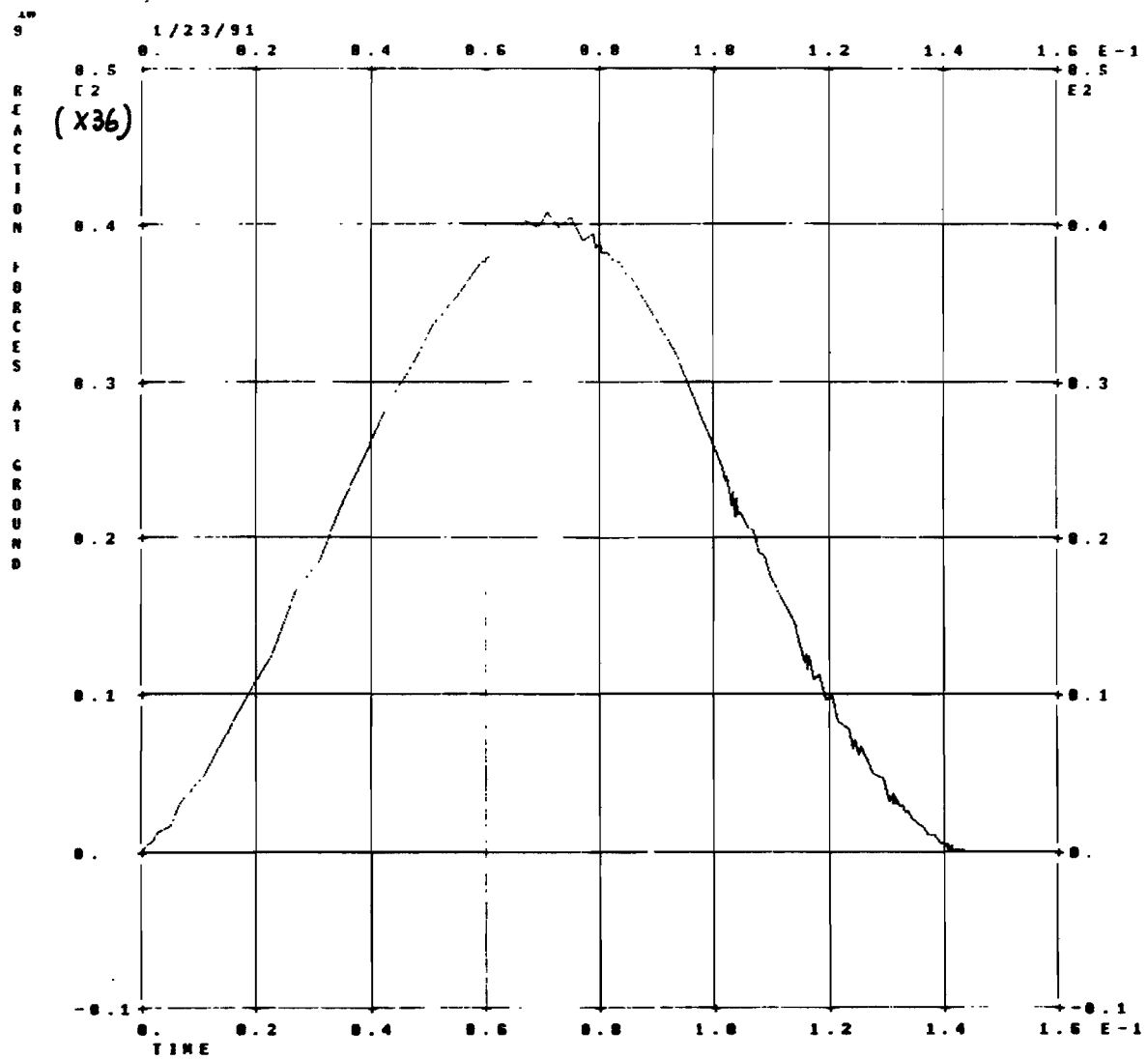
stiffness updates were activated (a total of 15 times) when GAPs were closing. After the peak impact, the time step size fluctuated between 0.25 msec. and 4.0 msec. due to thrashing of bisection and automatic adjustment, which seems to have been caused by opening GAPs. This thrashing has been remedied by introducing the numerical damping (PARAM,NDAMP).



$$\begin{aligned} R &= 5.0 \text{ in.} \\ E &= 1000 \text{ psi.} \\ \nu &= 0.3 \\ \rho &= 0.01 \text{ lb-sec}^2/\text{in}^4 \\ v_0 &= 10. \text{ in/sec.} \end{aligned}$$

Figure 4.3.17 Model for Dynamic Analysis

According to Eqs. (4.3.42) and (4.3.43), the maximum compression force (P_{max}) is 1246.5 lbs, which occurs when the approach distance (α_{max}) reaches 0.5251. The analysis shows $P_{max} \simeq 1440$ and $\alpha_{max} \simeq 0.53$ as shown in Figs. 4.3.18 and 4.3.19, respectively. The maximum indentation occurs in the vicinity of $t = 0.072$ sec. with the contact radius of approximately 1.71 while Eq. (4.3.39a) predicts 1.62. The duration of impact is 0.1544 sec. from Eq. (4.3.44). The analysis as shown in Fig. 4.3.18 indicates that the duration is around 0.143 sec. Small discrepancies are attributed to the discrete contact rather than continuous contact due to the discreteness in space and time. Fig. 4.3.20 shows snap shots of the deformed configuration at time 0.051 sec., 0.072 sec. and 0.2 sec.



NONLINEAR TRANSIENT IMPACT OF A SPHERE ON A RIGID FLAT PLANE
VELOCITY IN THE -Z IS 10.

Figure 4.3.18 Reaction Force vs. Time

Curves in the order of
Near contact point
Center point on mid-plane
Point at the top

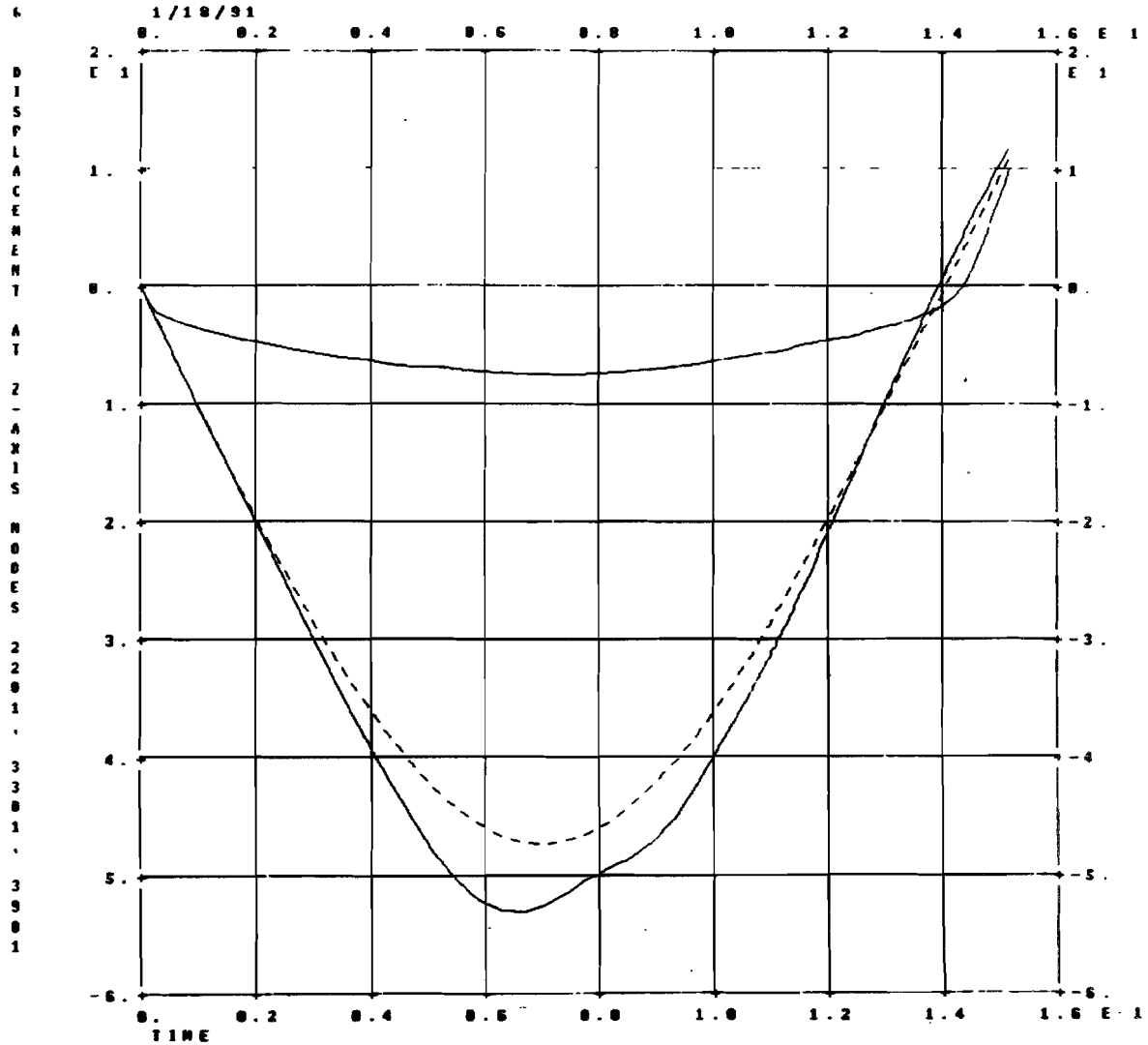
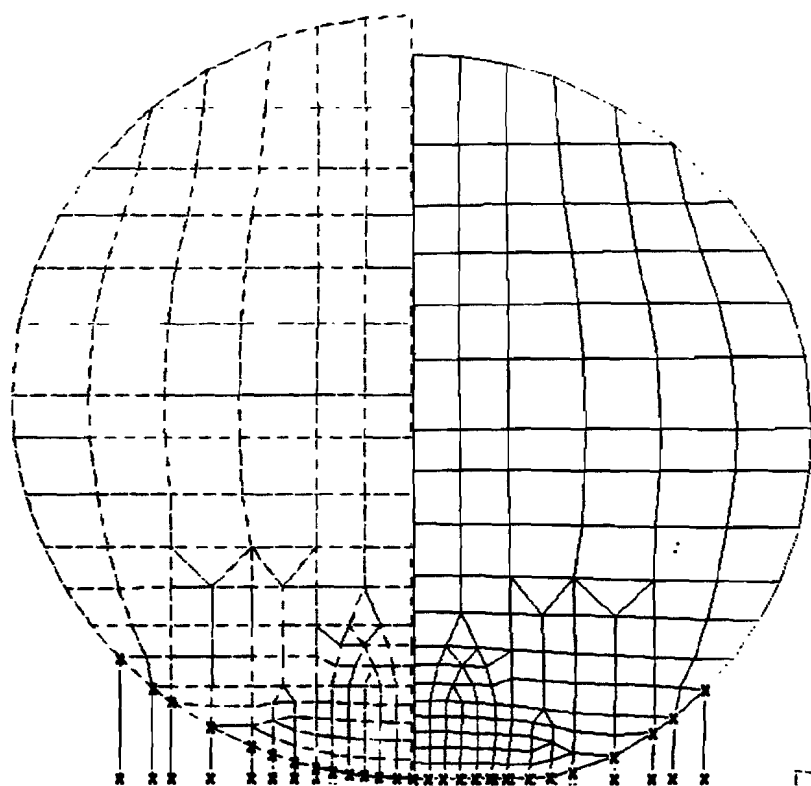


Figure 4.3.19 Vertical Displacement Response at Select Points on Center Line



Clockwise from
Upperleft corner:
 $t = 0., 0.051$
 $t = 0.2, 0.072$

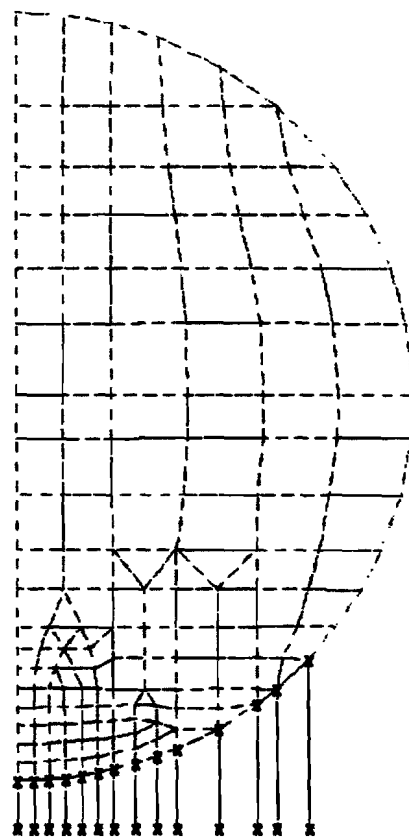
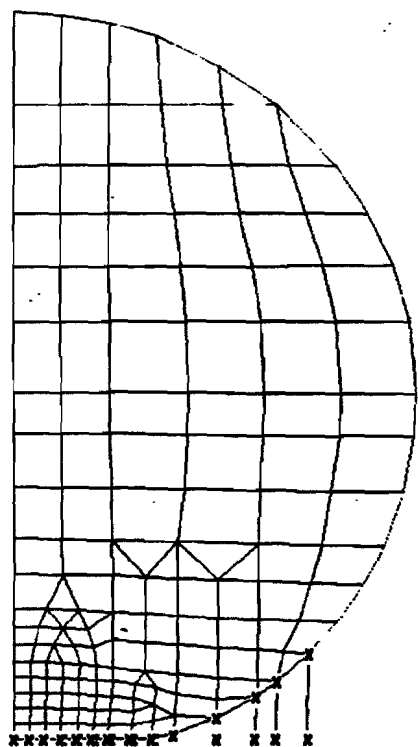


Figure 4.3.20 Snap Shots at $t=0., 0.051, 0.072$ and 0.2 sec.

4.3.12 Recommendations

The most crucial task in using the GAP element is to determine proper penalty values, which dictates the accuracy, efficiency and the effectiveness of the analysis. Penalty values should be chosen as large as possible for solution accuracy but as small as possible for solution efficiency. The recommended value for the penalty stiffness is 1000 times the stiffness of the adjacent structure, which will produce an error of 0.1%. Even with the advanced knowledge of the structural characteristics of the contact region, user specification of the penalty stiffness is merely an educated guess. This guess has to be a conservative one at the expense of the computational efficiency, because the proper value varies throughout the analysis.

The main goal of the self-adaptive method is to relieve users from the chores and uncertainties of guessing or trial-and-error searches. By achieving this goal, an optimum algorithm design is automatically attained for efficiency as well as for solution accuracy. A self-adaptive algorithm has been proven effective and efficient for nonlinear analysis in MSC/NASTRAN. The primary virtue of the adaptive method, however, is not the computing efficiency but the overall engineering efficiency. This is because the adaptability makes it possible to obtain the solution at the very first trial.

The adaptive GAP element implemented in Version 67, in concert with adaptive solution methods [4.12,4.16], makes it possible to maintain an acceptable accuracy while maintaining the solution efficiency without user intervention throughout the dynamic environment of nonlinear computation. The applicability of the present algorithm is deemed viable to simulation of surface contacts with friction. A limitation exists, however, that one of the contact surfaces should not rotate by a large angle because the GAP element orientation is not updated for large rotations. It is noted that there is an advantage in maintaining the original orientation of GAP elements when a contact with a fixed flat surface is simulated.

Chapter 5

GEOMETRIC NONLINEARITY

5.1 OVERVIEW AND USER INTERFACE

Geometric nonlinearities are manifested in problems involving large rotations and large deformation. The characteristics are follower forces due to large rotations, geometric stiffening due to initial stress effect (as a result of large rotations), and large strains due to large deformation.

Geometric nonlinear effects should be significant if the deformed shape of the structure appears distinctive from the original geometry by a visual inspection. A more rigorous and quantitative definition for the large displacements can be derived from the plate theory of Kirchhoff and Love: the small deflection theory is valid for a maximum deflection of less than 20% of the plate thickness or 2% of the small span length [5.1]. However, this definition seems to be a little conservative for numerical analysis, and there is no distinct limit for large displacements because geometric nonlinear effects are related to the boundary conditions as well as the dimensions of the structure. If the load-deflection curve of the critical point can be estimated, the loading point should be in the nonlinear portion of the curve.

Geometric nonlinear effects [5.2] in the structure involving large rotations, whether rigid body rotations or deformation induced rotations, are self-evident. Stiffening of a membrane, stiffness in a pendulum or snap-through of an arch belong to this category. The motion of a pendulum under gravity is caused by geometric (differential) stiffness. Follower forces are manifested when the applied loads are displacement dependent, such as pressure load and thermal load applied on the surface that rotates. Centrifugal force is another example of follower forces. Large strain effects are pronounced in metal forming problems which could have strains exceeding 100%. Finite strain formulation is required to treat the problems in this category. The software does not currently support the large strain capability. In most structural applications, however, moderately large strains (20 to 30 %) appear in local areas if there is any large deformation. The software can be used for that category of problems. Other geometric nonlinear effects are treated by updated element coordinates, gimbal angles

(or rotation vector), and the differential stiffness $[K^d]$.

The geometric nonlinearity is controlled by only one parameter (PARAM, LGDISP) with the following values in SOL 66 and SOL 99:

- 0 for geometrically linear analysis
- 1 for geometrically nonlinear analysis
- 2 for geometrically nonlinear, but without follower force effect

With values of 1 or 2 for LGDISP, all the potentially nonlinear elements become actively nonlinear elements unlike the material nonlinear model.

This solver has a distinct approach to the large rotation, for which the element coordinates are continuously updated to the current configuration during the iteration. The equilibrium is sought in the deformed position. Consider the internal force computation as follows [5.3]:

$$F = \int_V \bar{B}^T \sigma dV \quad (5.1.1)$$

The element matrix (\bar{B}) is defined from the strain definition as

$$\{d\epsilon\} = [\bar{B}]\{du\} \quad (5.1.2)$$

in which $[\bar{B}]$ could be divided into two parts (linear and nonlinear), i.e.,

$$\bar{B} = B_L + B_N \quad (5.1.3)$$

Upon differentiation of Eq. (5.1.1), we have

$$dF = \int_V \bar{B}^T d\sigma dV + \int_V d\bar{B}^T \sigma dV \quad (5.1.4)$$

where $\{\sigma\}$ represents stresses with reference to the original coordinates. Substituting Eq. (5.1.3) and

$$d\sigma = D d\epsilon = D \bar{B} du, \quad (5.1.5)$$

Eq. (5.1.4) becomes

$$dF = \underbrace{[K_L + K_R + K_\sigma]}_{\text{tangential matrix}} du \quad (5.1.6)$$

with

$$K_L = \int_V B_L^T D B_L dV$$

$$K_R = \int_V [B_L^T D B_N + B_N^T D B_N + B_N^T D B_L] dV \quad (5.1.7)$$

and

$$K_\sigma du = \int_V dB_N^T \sigma dV \quad (5.1.8)$$

in which K_L represents the usual linear stiffness matrix, K_R a stiffness due to large rotation, and K_σ a geometric stiffness dependent on the initial stress level.

Now it remains to define the nonlinear part of the element matrix (B_N). The definition of finite strains based on the Lagrangian formulation (referred to the initial configuration) is as follows:

$$\begin{aligned}\varepsilon_x &= \frac{\partial u}{\partial x} + \frac{1}{2} \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial w}{\partial x} \right)^2 \right] \\ \gamma_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} + \left[\frac{\partial u}{\partial x} \cdot \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \cdot \frac{\partial v}{\partial y} + \frac{\partial w}{\partial x} \cdot \frac{\partial w}{\partial y} \right]\end{aligned}\quad (5.1.9)$$

with other components obtained similarly. In matrix notation

$$\{\varepsilon\} = \{\varepsilon_L\} + \{\varepsilon_N\}$$

where $\{\varepsilon_L\}$ is the usual infinitesimal strain vector and $\{\varepsilon_N\}$ is the nonlinear strain vector consisting of the second order terms, i.e.,

$$\begin{aligned}\{\varepsilon_L\} &= [B_L]\{u\} \\ \{\varepsilon_N\} &= \frac{1}{2}[A]\{\theta\}\end{aligned}\quad (5.1.10)$$

where

$$[A] = \begin{bmatrix} a_x^T & 0 & 0 \\ 0 & a_y^T & 0 \\ 0 & 0 & a_z^T \\ a_y^T & a_x^T & 0 \\ 0 & a_z^T & a_y^T \\ a_z^T & 0 & a_x^T \end{bmatrix} \quad \begin{matrix} 6 \times 9 \end{matrix} \quad (5.1.11)$$

and

$$\begin{matrix} 9 \times 1 \end{matrix} \{\theta\} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \quad \text{with} \quad \begin{matrix} 3 \times 1 \end{matrix} \{a_x\} = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial x} \\ \frac{\partial w}{\partial x} \end{bmatrix}$$

Introducing shape functions (N_i) and nodal displacements $\{u\}$ (using an example of a 10-noded tetrahedron), displacement derivatives are expressed by

$$\{a_x\} = \left[\frac{\partial N_1}{\partial x} I \quad \frac{\partial N_2}{\partial x} I \quad \dots \quad \frac{\partial N_{10}}{\partial x} I \right] \{u\} \quad (5.1.12)$$

and

$$\{\theta\} = [G]\{u\} \quad (5.1.13)$$

where

$$[G] = \begin{matrix} 9 \times 30 \\ \left[\begin{array}{cccc} \frac{\partial N_1}{\partial x} I & \frac{\partial N_2}{\partial x} I & \dots & \frac{\partial N_{10}}{\partial x} I \\ \frac{\partial N_1}{\partial y} I & \frac{\partial N_2}{\partial y} I & \dots & \frac{\partial N_{10}}{\partial y} I \\ \frac{\partial N_1}{\partial z} I & \frac{\partial N_2}{\partial z} I & \dots & \frac{\partial N_{10}}{\partial z} I \end{array} \right] \end{matrix} \quad (5.1.14)$$

and

$$\begin{matrix} 30 \times 1 \\ \{u\} = \left\{ \begin{array}{c} a_1 \\ \vdots \\ a_i \\ \vdots \\ a_{10} \end{array} \right\} \end{matrix} \quad \text{with} \quad \{a_i\} = \left\{ \begin{array}{c} u_i \\ v_i \\ w_i \end{array} \right\}$$

From the properties of matrices A and θ , it can be shown that

$$d\varepsilon_N = \frac{1}{2} dA \theta + \frac{1}{2} A d\theta = A d\theta = A G du$$

from which

$$\begin{matrix} [B_N] = [A] & [G] \\ 6 \times 30 & 6 \times 9 \quad 9 \times 30 \end{matrix} \quad (5.1.15)$$

The initial stress stiffness $[K_\sigma]$ can be derived as follows:

$$K_\sigma da = \int_V dB_N^T \sigma dV = \int_V G^T dA^T \sigma dV$$

in which

$$dA^T \sigma = \left[\begin{array}{cccccc} da_x & 0 & 0 & da_y & 0 & da_z \\ 0 & da_y & 0 & da_x & da_z & 0 \\ 0 & 0 & da_z & 0 & da_y & da_x \end{array} \right] \left\{ \begin{array}{c} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{array} \right\}$$

$$= M d\theta = M G du$$

where

$$[M] = \begin{bmatrix} \sigma_x I & \tau_{xy} I & \tau_{xz} I \\ & \sigma_y I & \tau_{yz} I \\ \text{Sym} & & \sigma_z I \end{bmatrix} \quad (5.1.16)$$

9×9

with I being the (3×3) identity matrix. Finally the geometric stiffness is

$$[K_\sigma] = \int_V G^T M G dV \quad (5.1.17)$$

30×30

It has been found that stiffness matrices caused by geometric nonlinearity (K_R and K_σ) can be computed from the matrices $[A]$, $[G]$, and $[M]$ with the following observations:

- $[G]$ is dependent upon the initial geometry, hence stays constant unless the geometry is updated. This matrix is used in forming $[K_R]$ and $[K_\sigma]$.
- $[A]$ is used in forming $[K_R]$. $[A]$ is dependent on the rotations and should be updated continuously.
- $[M]$ is used in forming $[K_\sigma]$. $[M]$ is dependent on the stresses and should be updated continuously.

The primary functions of nonlinear stiffness matrices can be interpreted as follows:

- The matrix $[K_R]$ takes into account the effects of large rotations. The large displacement effects, due to rigid body translation and rotation, are treated effectively in the absence of large strains by updating element coordinates in the software.
- Geometric stiffness matrix $[K_\sigma]$ takes into account the effects of the initial stresses. This effect becomes important with geometric stiffening, and is used for instability analysis. The geometric stiffness matrix $[K_\sigma]$ is equivalent to the differential stiffness $[K^d]$ in the software.

5.2 UPDATED ELEMENT COORDINATES

When the large displacement effect is included in the nonlinear analysis, the solver employs a method of displaced element coordinate system. This method allows large rotations by updating element coordinates to the deformed geometry, and the equilibrium is computed in the deformed configuration.

5.2.1 Concept of Convective Coordinates

The concept is based on the fact that the rigid body motion does not contribute to the strain energy and is eliminated from the internal force computation. Consider a rod which underwent rigid body motion as well as deformation as shown below:

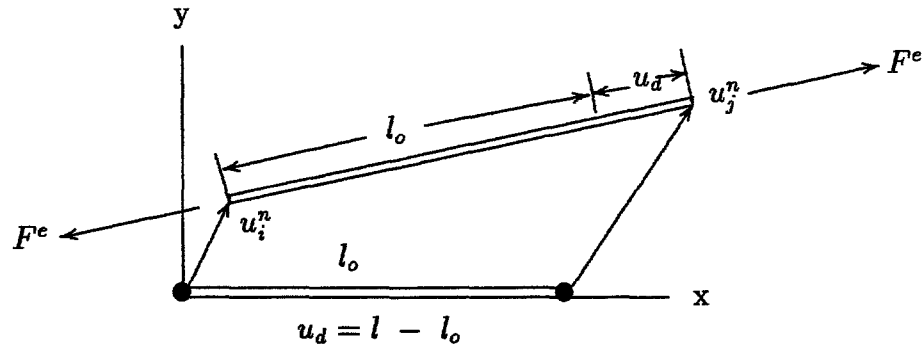


Figure 5.2.1 Net Deformation of a Rod

The net displacement u_d is measured in the displaced element coordinate system by overlaying the original element on top of the deformed element. The element force can simply be computed by

$$\{F_d\}^e = [K]^e \{u_d\} \quad (5.2.1)$$

where the superscript e denotes an elemental operation and the subscript d denotes the vectors in the displaced element coordinate system. Then the element forces should be transformed into the common coordinate system (namely global coordinate system denoted by a subscript g) before assembly for global operations, i.e.,

$$\{F_g\} = \sum T_{bg}^T T_{bd} \{F_d\}^e \quad (5.2.2)$$

where the summation sign implies an assembly operation, and T_{bd} and T_{bg} are transformation matrices from displaced to basic and from global to basic coordinate systems, respectively.

5.2.2 Updated Coordinates and Net Deformation

Referring to Fig. 5.2.2, a quadrilateral element is shown in its original and deformed positions (denoted by subscripts e and d , respectively) with reference to the basic coordinate system (denoted by a subscript b).

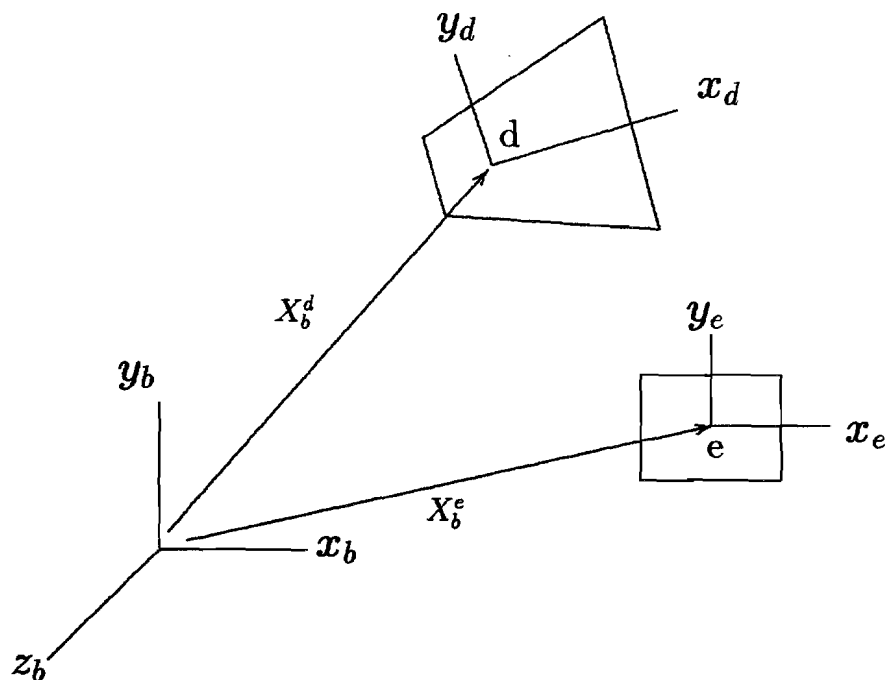


Figure 5.2.2 Element Coordinates vs. Displaced Coordinates

The element coordinate system is established by bisecting the diagonals of the quadrilateral. Transformation from the element coordinate system to basic coordinates is simply

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{basic} = [T_{be}] \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{element} + \begin{Bmatrix} x^e \\ y^e \\ z^e \end{Bmatrix}_{basic} \quad (5.2.3)$$

where the position vector (X_b^e in Fig. 5.2.2) of the element coordinate system with respect to the basic coordinate system is denoted by $\langle x^e, y^e, z^e \rangle_{basic}^T$, and transformation matrix $[T_{be}]$ is composed of direction cosines of unit vectors of the element coordinate system with respect to the basic coordinate system, i.e.,

$$[T_{be}] = \begin{bmatrix} i_x & j_x & k_x \\ i_y & j_y & k_y \\ i_z & j_z & k_z \end{bmatrix}. \quad (5.2.4)$$

As the element deforms or displaces, the element coordinate system moves and this is defined as a displaced coordinate system. The displaced coordinate system is established in the same manner as the element coordinate system. Again the transformation should be performed similarly, i.e.,

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{basic} = [T_{bd}] \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{displaced} + \begin{Bmatrix} x^d \\ y^d \\ z^d \end{Bmatrix}_{basic} \quad (5.2.5)$$

where $\langle x^d, y^d, z^d \rangle_{basic}^T$ is the position vector of the displaced element coordinate system with respect to the basic coordinate system (X_b^d in Fig. 5.2.2) and $[T_{bd}]$ is formed similarly to $[T_{be}]$.

In order to isolate the deformation from the rigid body displacements, nodal displacements are computed in the displaced element coordinate system by overlaying the original element as shown in Fig. 5.2.3.

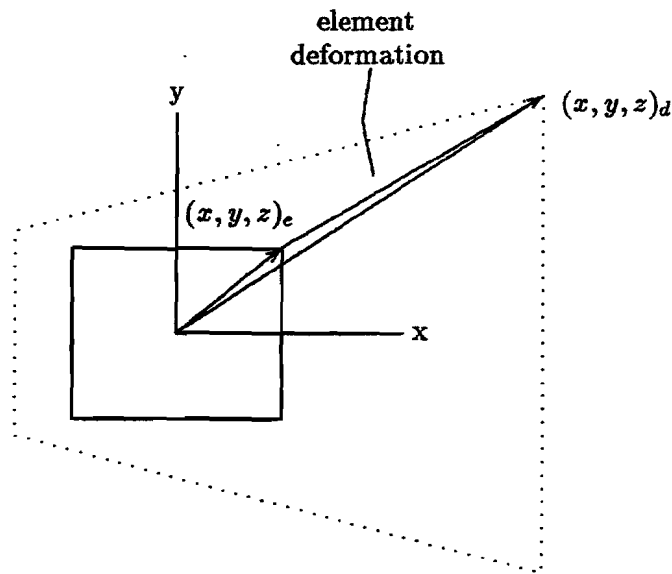


Figure 5.2.3 Computation of Net Deformation

The net displacements can be computed by subtracting the original nodal coordinates in the element coordinate system from the displaced nodal coordinates in the displaced element coordinate system, i.e.,

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix}_d = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_d - \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_e \quad (5.2.6)$$

in which the nodal coordinates in the element and displaced element coordinate systems can be computed by the following transformations:

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_e = [T_{be}]^T \left[\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{basic} - \begin{Bmatrix} x^e \\ y^e \\ z^e \end{Bmatrix}_{basic} \right] \quad (5.2.7)$$

and

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_d = [T_{bd}]^T \left[\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}_{basic} + [T_{bg}] \begin{Bmatrix} u \\ v \\ w \end{Bmatrix}_{global} - \begin{Bmatrix} x^d \\ y^d \\ z^d \end{Bmatrix}_{basic} \right] \quad (5.2.8)$$

Substitution of Eqs. (5.2.7) and (5.2.8) into Eq. (5.2.6) results in

$$\{u_d\} = T_{bd}^T \{X_b + T_{bg} u_g - X_b^d\} - T_{be}^T \{X_b - X_b^e\} \quad (5.2.9)$$

where $\{u_g\}$ is a total displacement (translational components only) in the global coordinates. In the absence of the large displacement effect, the net displacement u_d in Eq. (5.2.9) is reduced to:

$$\{u_e^i\} = T_{be}^T T_{bg} \{u_g^i\}$$

5.2.3 Provisions for Global Operation

It is noted that the net rotations (θ_x , θ_y and θ_z of each node associated with the shell and beam elements) are computed by a gimbal angle approach (or rotation vector approach) before computing element forces. Subsequently, the element forces have to be transformed to the global coordinate system before assembly for equilibrium check. The internal forces are computed using net displacements and rotations, u_d , i.e.,

$$\{F_g\} = \sum T_{bg}^T T_{bd} [K]^e \{u_d\} \quad (5.2.10)$$

if the material is linear or

$$\{F_g\} = \sum T_{bg}^T T_{bd} \int B^T \sigma(u_d) dV \quad (5.2.11)$$

if nonlinear material is involved. Consequently, the tangent stiffness matrix is formed in the global system by assembling the element stiffness matrices transformed into the global coordinate system from the displaced coordinate system, i.e.,

$$[K] = \sum T_{bg}^T T_{bd} [K]^e T_{bd}^T T_{bg} \quad (5.2.12)$$

The update process is performed at every iteration and the updated nodal displacements u_d are used whenever strains and stresses are computed. Effectively, the second order effect

due to large rigid body motion is eliminated. However, the displacement output shows the total displacements in the global coordinates, i.e.,

$$\{u_g^i\} = T_{bg}^T T_{bd} \{u_d^i\} + T_{bg}^T T_{bd} \left[T_{be}^T \{x_b^i - x_b^e\} - T_{bd}^T \{x_b^i - x_b^d\} \right] \quad (5.2.13)$$

where the superscript i denotes operations on each nodal point. The transformation matrix $[T_{bd}]$ is computed for each element after each iteration and stored in the ESTNL data block for stiffness matrix update when required by the stiffness matrix update strategy. On the other hand, the transformation $[T_{bg}]$ is computed for each nodal point and it is not stored but recomputed whenever it is needed. The nodal coordinates in the undeformed geometry, X_b , are available from the data block BGPDT.

This approach can be interpreted as approximate updated Lagrangian method, since the motion of the body follows Lagrangian description. Stresses are computed in the deformed geometry just like Cauchy stress. However, this method of displaced coordinate system is a unique and salient feature in the software. The referential geometry in the updated Lagrangian method is brought up-to-date at every incremental step upon convergence but fixed during the iterative process, which is inherently different from the current method of updating the coordinate system.

5.3 FOLLOWER FORCES

The term “follower force” usually refers to the applied loads that change direction and magnitude with structural displacements and rotations, e.g.,

$$P(u) = \int_A N^T p \hat{n} dA \quad (5.3.1)$$

where p is the magnitude of the pressure on the surface A , interpolated by a shape function N , and $\hat{n}dA$ changes as a function of u . They generally occur with fluid pressures such as the pressurized balloon, inflated tire or the lift load on the airplane wing. Other physical applications involve kinematics such as the classical fire hose instability problem or inertia loads on spinning bodies. In the software, the term applies to specific load inputs as defined below.

5.3.1 Basic Definition

For geometrical nonlinear analysis, static loads belong to one of two categories, namely:

- Forces and moments defined by fixed vector inputs, which may be calculated once per run and cannot change direction or magnitude.
- Forces and moments defined by the location of one or more GRID points, which include pressures, thermal loads, and centrifugal forces.

The first category includes simple forces, moments and enforced displacements. The second, follower force category, includes the following Bulk Data inputs:

FORCE1, FORCE2, MOMENT1, MOMENT2: The direction changes with displacements of the referenced GRID points. The magnitudes of these concentrated loads are constant.

PLOAD, PLOAD2, PLOAD4: The pressure loads follow the surface of the solid elements (HEXA, PENTA and TETRA) or shell elements (QUAD4 and TRIA3) defined by the GRID points and will change in magnitude if the area changes.

RFORCE: Centrifugal loads change in magnitude and direction with motion of the masses attached to the GRID points. The effect may be destabilizing if large motions occur. It is recommended that lumped masses be used with these loads.

TEMP, TEMP1, etc: Thermal loads are built into the element force calculations and will automatically follow the elements.

Also note that upstream superelements are assumed to be linear and therefore the upstream loads will remain fixed in magnitude and direction. In addition, forces on omitted degrees of freedom (when ASET or OMIT data are present) should not be follower forces.

5.3.2 Implementation

The follower forces depend on the GRID displacements and therefore must be recalculated for each nonlinear iteration and line search. The basic equation for residual error, defined by Eq. (3.2.2), becomes:

$$\{R_a^i\} = \{P_a(u^i)\} - \{F_a(u^i)\} \quad (5.3.2)$$

where the applied load vector $\{P_a\}$ is now a variable. Corrective Loads are computed based on the updated geometry and added to the initially applied loads to account for the follower forces, i.e.,

$$P(u) = P(0) + f(u)$$

where

$$f(u) = P(u) - P(0).$$

Note that thermal effects are included in the vector $\{F\}$.

In turn, the tangent matrix could be calculated from Eq. (3.2.3) using derivatives of the loads, which is termed follower matrix [5.4, 5.5]. However, the nonlinear solution process in MSC/NASTRAN ignores the stiffness effects of the changing loads and use the approximation:

$$[K_T] = -\left[\frac{\partial R}{\partial u}\right] \simeq \left[\frac{\partial F}{\partial u}\right] \quad (5.3.3)$$

The effect of the approximation is minor in most cases. However, it could become a major concern in thin shell models with pressure loads causing large rotations, where the converged solutions will be correct but the rate of convergence may be slow or cause divergence. Also the buckling solutions or modal analysis on preloaded structure with pressure load may not be correct due to the approximate tangent matrix if the effect of the follower matrix is significant.

The follower force effects in the analysis can be controlled by the parameter LGDISP. Three options are available in PARAM LGDISP:

- 0 for no geometric nonlinearity
- 1 for full geometric nonlinearity (including follower forces)
- 2 for deactivating follower forces while maintaining other geometric effects (rotation of element coordinates and differential stiffness).

The third option was made available from Version 66, primarily to verify the effects of the follower forces.

5.3.3 Dynamic Follower Forces

All of the follower forces listed above, which are available in static nonlinear analysis (SOL 66 and SOL 106), are also available in nonlinear transient analysis (SOL 99 and SOL 129) except for the following restrictions:

- Centrifugal forces (RFORCE) are not recalculated in SOL 99 or SOL 129.
- Thermal effects are not supported in the nonlinear transient analysis.
- The applied load is a function of displacement (follower forces) as well as time, i.e.,

$$P_{n+1}^i(u) = P_{n+1}(0) + f_F^i(u, t_{n+1}) + N(u)^i$$

with

$$f_F^i(u, t_{n+1}) = g(t_{n+1}) \{p^i(u) - p(0)\}$$

where $p^i(u)$ represents the follower forces, $g(t)$ is a forcing function in terms of time, and $N(u)^i$ is the applied load via NOLINI options. The Load Set Identification number is defined on LSEQ Bulk Data. The TLOAD1 and TLOAD2 data define the time dependency and refer to the DAREA IDs of the LSEQ data.

- Dynamic loading with DAREA Bulk Data and initial conditions do not follow the structure and are treated as linear loads.

5.3.4 Verification: Elliptic Cylinder Subject to an Internal Pressure

A thin-walled cylinder having an elliptical shape [5.2] is subjected to an increasing internal pressure, up to a maximum of 5 psi. Fig. 5.3.1 shows one quarter of the cross-section of the elliptic cylinder consisting of a row of 12 QUAD4 elements. The material of the cylinder is elastic-plastic obeying the von Mises yield criterion and the isotropic hardening rule, with the plasticity modulus being 1% of the Young's modulus. The material properties are specified by the MAT1 and MATS1 Bulk Data entries. All the translational degrees-of-freedom in the longitudinal direction are constrained to represent a typical segment of the long cylinder. The pressure loading is applied both statically (SOL 66) and dynamically (SOL 99) by PLOAD4 Bulk Data entries and the displacement responses are observed. Geometric nonlinearity effects are included in the analysis by specifying PARAM, LGDISP, 1. The solution strategy is provided by default methods in the NLPARM Bulk Data entry in SOL 66 and by the TSTEPNL Bulk Data entry in SOL 99. Table 5.3.1 and 5.3.2 list the input data for SOL 66 and SOL 99, respectively. The effect of follower forces is a conspicuous feature in this problem.

In the static analysis, the pressure load of 5 psi is applied in 5 subcases (total of 51 increments) with gradually increasing incremental pressure. The bisection is activated seven times while the geometric effects are predominant during loading up to 0.015 psi (in the first subcase). The loading history and the numerical performance of each subcase are tabulated below.

Numerical Performance During Static Analysis

subcase ID	load (psi)	no. of increments	no. of iterations	no. of K updates	no. of bisections	no. of line searches
1	0.05	20	656	174	7	401
2	0.1	10	143	26	0	81
3	0.5	8	126	26	0	57
4	1.0	5	53	8	0	16
5	5.0	8	60	7	0	13

The dynamic analysis is performed in five subcases of equal duration (1 sec) with an initial time step size of 10 msec. The internal pressure of the cylinder is increasing linearly at the rate of 1.0 psi/sec. The bisection process is activated 37 times during the first second of the time history. In the beginning of the analysis, bisections take place due to rapid changes in geometric effects. The bisection is again activated when the plastic deformation starts while approaching the peak (maximum and minimum) displacements. The plastic deformation occurs in the region near the small radius on the major axis of the ellipse (point B in Fig. 5.3.1). It is noted that the plastic reloading occurs after the unloading while approaching the minimum peak displacements. The automatic time step adjustment is activated 35 times during the first second of the analysis, mainly to recover from the bisection process by stretching the previously reduced time step size.

Fig. 5.3.2 shows the displacement responses at two points (denoted by A and B) on the minor and major axes of the ellipse. The vibration frequency increases as the internal pressure increases, but the amplitude diminishes due to the geometric stiffening. The static response is plotted in the same figure, which shows good agreement between the static response and the mean dynamic amplitude.

The static and dynamic deformed shapes (snap shots at the extreme magnitudes of the vibration) in the vicinity of 5 psi are shown in Fig. 5.3.3. The static deformation turns the elliptical into a nearly-circular cylinder, and the dynamic deformation is dominated by the ovaling mode. The effect of follower forces is verified and validated by the fact that the static deformed shape is approaching a circular shape. It is noted that the circular deformation pattern is formed at less than 0.5 psi in static analysis, and the geometric nonlinearity disappears beyond that point.

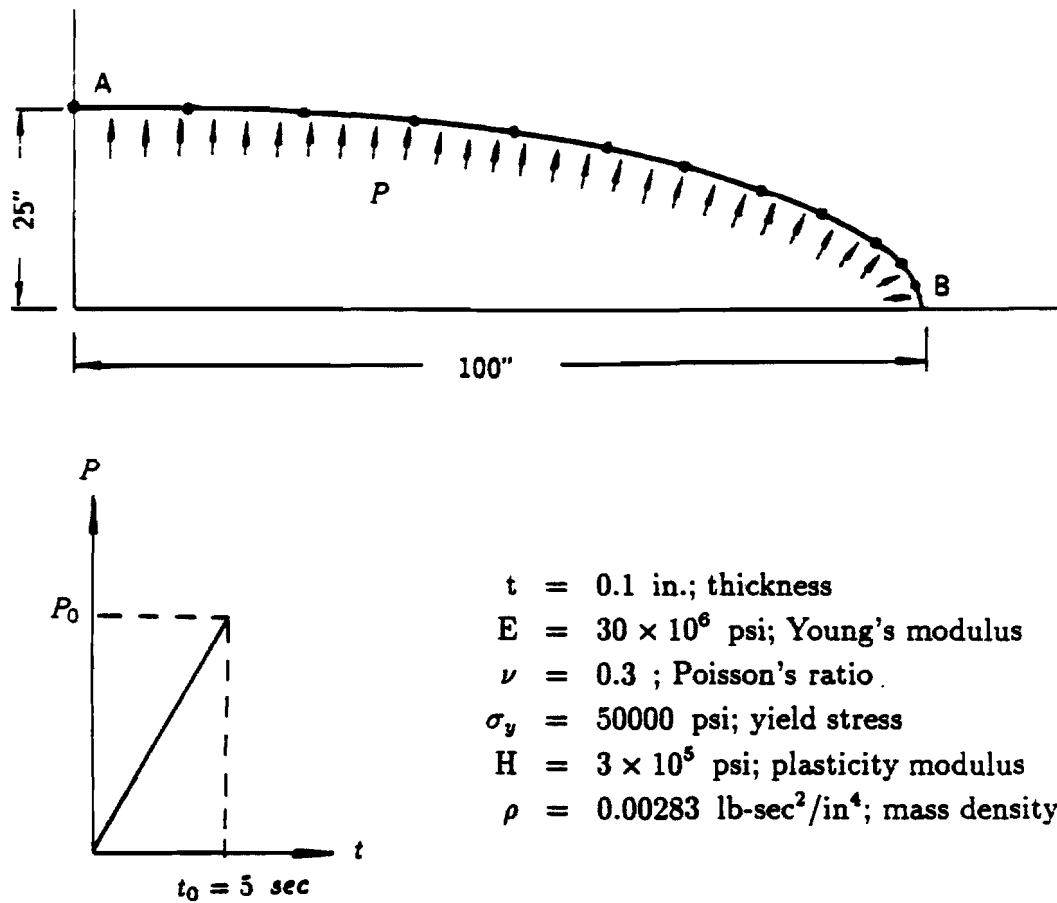


Figure 5.3.1 Elliptic Cylinder Subject to an Internal Pressure

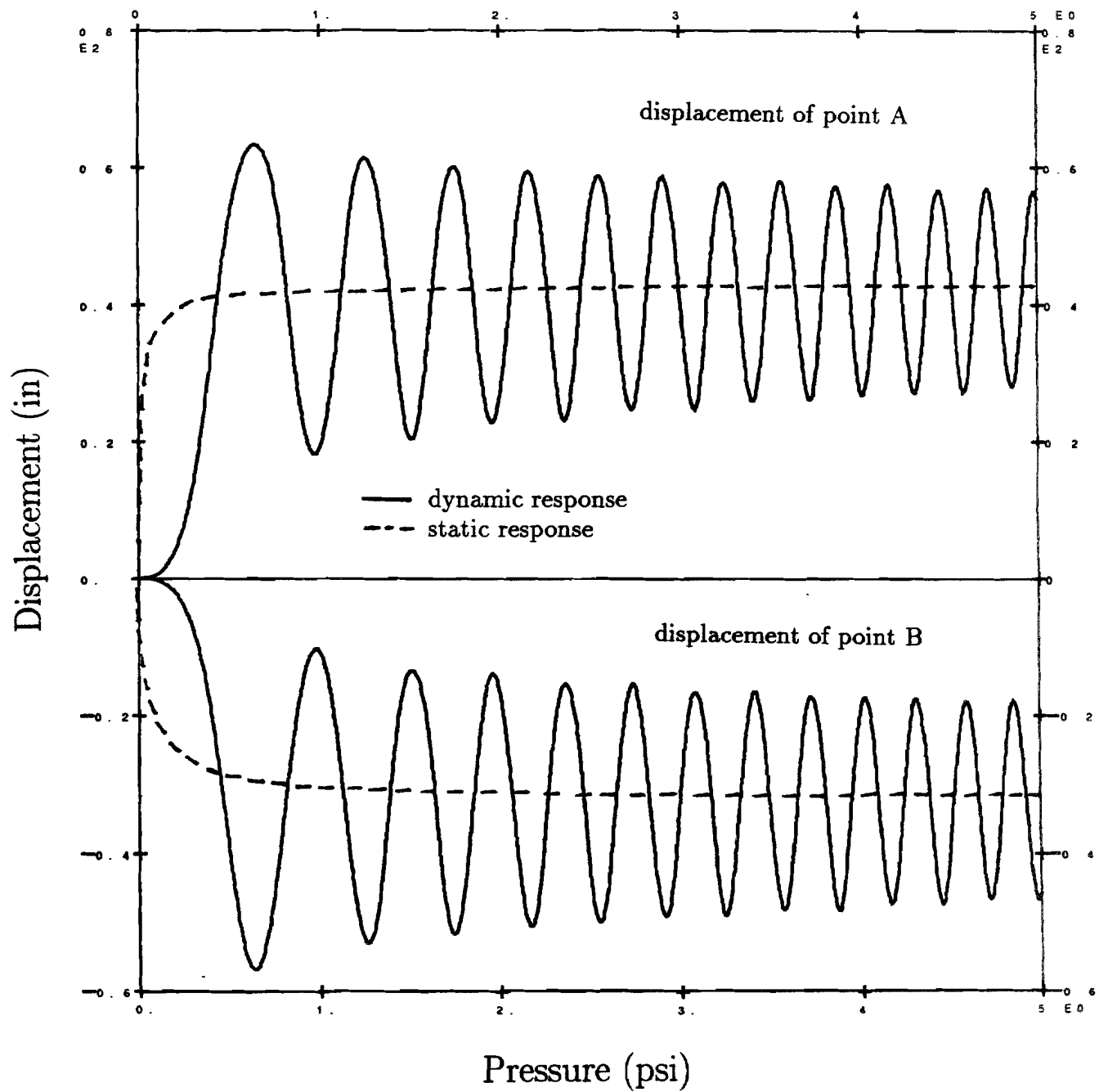


Figure 5.3.2 Displacement Responses for Elliptic Cylinder

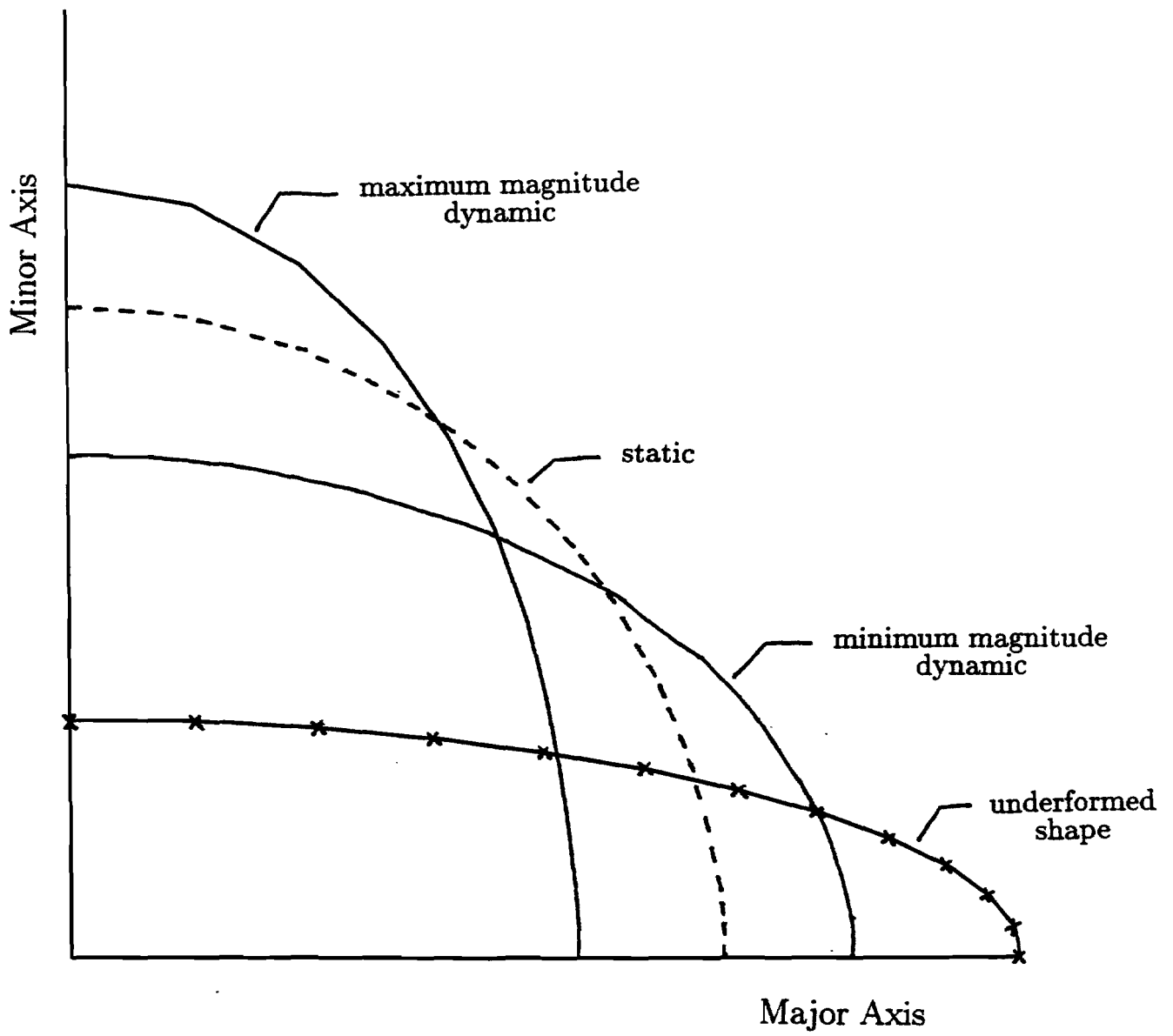


Figure 5.3.3 Deformed Shapes of the Elliptic Cylinder

Table 5.3.1 SOL 66 Input Data Listing for Elliptic Cylinder

```

ID      ELLIP66, V67  $ APR 6-25-91, SSH 4/20/88, SHL 3/9/88, KIM 2/87
SOL     66           $ NONLINEAR STATIC ANALYSIS
TIME    30           $ CPU MINUTES
DIAG    8,50         $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = ELLIPTIC CYLINDER UNDER INTERNAL PRESSURE
SEALL = ALL
ECHO = BOTH
      SET 1 = 1000,1120
      DISP = 1
SPC     = 200
SUBCASE 1
      LOAD = 10
      NLPARM = 100
SUBCASE 2
      LOAD = 20
      NLPARM = 200
SUBCASE 3
      LOAD = 30
      NLPARM = 300
SUBCASE 4
      LOAD = 40
      NLPARM = 400
SUBCASE 5
      LOAD = 50
      NLPARM = 500
OUTPUT(PLOT)
      CSCALE 1.3
      PLOTTER NAST
      SET 1 PLOTEL
      VIEW 0., 0., 0.
      AXES Z , X , Y
      MAXI DEFO .1
      FIND SCALE ORIGIN 1 SET 1
      PLOT  STATIC 0 1 RANGE 1.0 MAXI DEFO .1 SET 1
      PLOT  STATIC 0 2 RANGE 2.0 MAXI DEFO .1 SET 1
      PLOT  STATIC 0 3 RANGE 3.0 MAXI DEFO .1 SET 1
      PLOT  STATIC 0 4 RANGE 4.0 MAXI DEFO .1 SET 1
      PLOT  STATIC 0 5 RANGE 5.0 MAXI DEFO .1 SET 1
OUTPUT(XY PLOT)
      CSCALE 1.3
      PLOTTER NAST
      YMIN = -60.
      YMAX = 80.
      XTITLE = TIME IN SEC
      YTITLE = DISP OF GRID 1000 AND 1120
      XY PLOT DISP RESP/1000(T1),1120(T2)

```



```

BEGIN BULK
PARAM  LGDISP  1
$$ SOLUTION CONTROL
NLPARM  100      20          AUTO          YES
NLPARM  200      10          AUTO          YES
NLPARM  300      8           AUTO          YES
NLPARM  400      5           AUTO          YES
NLPARM  500      8           AUTO          YES
$$ LOADING
PLOAD4  10      101      .05          THRU      112
PLOAD4  20      101      .1           THRU      112
PLOAD4  30      101      .5           THRU      112
PLOAD4  40      101      1.           THRU      112
PLOAD4  50      101      5.           THRU      112
$$ GEOMETRY
GRID    1000          100.    0.    10.          345
GRID    1001          100.    0.    0.           345
GRID    1010          99.3625 3.30491 10.          345
GRID    1011          99.3625 3.30491 0.           345
GRID    1020          96.8149 6.51543 10.          345
GRID    1021          96.8149 6.51543 0.           345
GRID    1030          92.5105 9.59323 10.          345
GRID    1031          92.5105 9.59323 0.           345
GRID    1040          86.6025 12.5    10.          345
GRID    1041          86.6025 12.5    0.           345
GRID    1050          79.2443 15.1974 10.          345
GRID    1051          79.2443 15.1974 0.           345
GRID    1060          70.5889 17.6472 10.          345
GRID    1061          70.5889 17.6472 0.           345
GRID    1070          60.7898 19.8111 10.          345
GRID    1071          60.7898 19.8111 0.           345
GRID    1080          50.      21.6506 10.          345
GRID    1081          50.      21.6506 0.           345
GRID    1090          38.3729 23.1276 10.          345
GRID    1091          38.3729 23.1276 0.           345
GRID    1100          26.0617 24.2037 10.          345
GRID    1101          26.0617 24.2037 0.           345
GRID    1110          13.2197 24.8406 10.          345
GRID    1111          13.2197 24.8406 0.           345
GRID    1120          0.      25.    10.          345
GRID    1121          0.      25.    0.           345
GRID    2000          0.      0.      0.          123456
GRID    3000          100.    0.      0.          123456
GRID    4000          0.      100.    0.          123456
$$ CONNECTIVITY
CQUAD4  101      100      1000      1001      1011      1010
CQUAD4  102      100      1010      1011      1021      1020
CQUAD4  103      100      1020      1021      1031      1030
CQUAD4  104      100      1030      1031      1041      1040

```

CQUAD4	105	100	1040	1041	1051	1050
CQUAD4	106	100	1050	1051	1061	1060
CQUAD4	107	100	1060	1061	1071	1070
CQUAD4	108	100	1070	1071	1081	1080
CQUAD4	109	100	1080	1081	1091	1090
CQUAD4	110	100	1090	1091	1101	1100
CQUAD4	111	100	1100	1101	1111	1110
CQUAD4	112	100	1110	1111	1121	1120
\$\$ PROPERTIES						
PSHELL	100	110	.1	110		110
MAT1	110	3.+7		.3	.283-2	
MATS1	110		PLASTIC 3.+5			50000.
\$\$ CONSTRAINTS						
SPC1	200	16	1120	1121		
SPC1	200	26	1000	1001		
\$\$ FOR PLOT						
PLOTEL	1000	1000	1010			
PLOTEL	1001	1010	1020			
PLOTEL	1002	1020	1030			
PLOTEL	1003	1030	1040			
PLOTEL	1004	1040	1050			
PLOTEL	1005	1050	1060			
PLOTEL	1006	1060	1070			
PLOTEL	1007	1070	1080			
PLOTEL	1008	1080	1090			
PLOTEL	1009	1090	1100			
PLOTEL	1010	1100	1110			
PLOTEL	1011	1110	1120			
PLOTEL	1012	2000	3000			
PLOTEL	1013	2000	4000			
\$						
ENDDATA						

Table 5.3.2 SOL 99 Input Data Listing for Elliptic Cylinder

```

ID      ELLIP99, V67      $ APR 6-25-91, SSH 4/28/88, SHL 3/9/88, KI-00K KIM 2/87
SOL     99                $ NONLINEAR TRANSIENT ANALYSIS
TIME    200              $ CPU MINUTES
DIAG    8,50             $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND

TITLE = ELLIPTIC CYLINDER UNDER INTERNAL PRESSURE
SUBTITLE = PROPORTIONALLY LOADED UP TO 5 PSI IN 5 SECONDS
SEALL = ALL
ECHO = BOTH
    SET 1 = 1000,1120
    DISP = 1
SPC     = 200
LOADSET = 1000
SUBCASE 1
    DLOAD = 100
    TSTEPNL = 100
SUBCASE 2
    DLOAD = 100
    TSTEPNL = 200
SUBCASE 3
    DLOAD = 100
    TSTEPNL = 300
SUBCASE 4
    DLOAD = 100
    TSTEPNL = 400
SUBCASE 5
    DLOAD = 100
    TSTEPNL = 500
OUTPUT(PLOT)
    CSCALE 1.3
    PLOTTER NAST
    SET 1 PLOTTEL
    VIEW 0., 0., 0.
    AXES Z , X , Y
    MAXI DEFO .1
    FIND SCALE ORIGIN 1 SET 1
    PLOT TRANS 0 TIME 4.8 5.0 MAXI DEFO .1 SET 1
OUTPUT(XY PLOT)
    CSCALE 1.3
    PLOTTER NAST
    XTITLE = TIME IN SEC
    YTITLE = DISP OF GRID 1000 AND 1120
    XY PLOT DISP RESP/1000(T1),1120(T2)
BEGIN BULK
PARAM LGDISP 1
$$ SOLUTION CONTROL
TSTEPNL 100      100      .01      ADAPT

```

TSTEPNL	200	100	.01		ADAPT	
TSTEPNL	300	100	.01		ADAPT	
TSTEPNL	400	100	.01		ADAPT	
TSTEPNL	500	100	.01		ADAPT	
\$\$ LOADING						
LSEQ	1000	900	110			
PLCAD4	110	101	5.			THRU 112
TLOAD1	100	900	0	0	120	
TABLED1	120					+TBD1
+TBD1	0.	0.	5.	1.	ENDT	
\$\$ GEOMETRY						
GRID	1000		100.	0.	10.	345
GRID	1001		100.	0.	0.	345
GRID	1010		99.3625	3.30491	10.	345
GRID	1011		99.3625	3.30491	0.	345
GRID	1020		96.8149	6.51543	10.	345
GRID	1021		96.8149	6.51543	0.	345
GRID	1030		92.5105	9.59323	10.	345
GRID	1031		92.5105	9.59323	0.	345
GRID	1040		86.6025	12.5	10.	345
GRID	1041		86.6025	12.5	0.	345
GRID	1050		79.2443	15.1974	10.	345
GRID	1051		79.2443	15.1974	0.	345
GRID	1060		70.5889	17.6472	10.	345
GRID	1061		70.5889	17.6472	0.	345
GRID	1070		60.7898	19.8111	10.	345
GRID	1071		60.7898	19.8111	0.	345
GRID	1080		50.	21.6506	10.	345
GRID	1081		50.	21.6506	0.	345
GRID	1090		38.3729	23.1276	10.	345
GRID	1091		38.3729	23.1276	0.	345
GRID	1100		26.0617	24.2037	10.	345
GRID	1101		26.0617	24.2037	0.	345
GRID	1110		13.2197	24.8406	10.	345
GRID	1111		13.2197	24.8406	0.	345
GRID	1120		0.	25.	10.	345
GRID	1121		0.	25.	0.	345
GRID	2000		0.	0.	0.	123456
GRID	3000		100.	0.	0.	123456
GRID	4000		0.	100.	0.	123456
\$\$ CONNECTIVITY						
CQUAD4	101	100	1000	1001	1011	1010
CQUAD4	102	100	1010	1011	1021	1020
CQUAD4	103	100	1020	1021	1031	1030
CQUAD4	104	100	1030	1031	1041	1040
CQUAD4	105	100	1040	1041	1051	1050
CQUAD4	106	100	1050	1051	1061	1060
CQUAD4	107	100	1060	1061	1071	1070
CQUAD4	108	100	1070	1071	1081	1080

CQUAD4	109	100	1080	1081	1091	1090
CQUAD4	110	100	1090	1091	1101	1100
CQUAD4	111	100	1100	1101	1111	1110
CQUAD4	112	100	1110	1111	1121	1120
\$\$ PROPERTIES						
PSHELL	100	110	.1	110		110
MAT1	110	3.+7		.3	.283-2	
MATS1	110		PLASTIC	3.+5		50000.
\$\$ CONSTRAINTS						
SPC1	200	16	1120	1121		
SPC1	200	26	1000	1001		
\$\$ FOR PLOT						
PLOTEL	1000	1000	1010			
PLOTEL	1001	1010	1020			
PLOTEL	1002	1020	1030			
PLOTEL	1003	1030	1040			
PLOTEL	1004	1040	1050			
PLOTEL	1005	1050	1060			
PLOTEL	1006	1060	1070			
PLOTEL	1007	1070	1080			
PLOTEL	1008	1080	1090			
PLOTEL	1009	1090	1100			
PLOTEL	1010	1100	1110			
PLOTEL	1011	1110	1120			
PLOTEL	1012	2000	3000			
PLOTEL	1013	2000	4000			
\$						
ENDDATA						

5.4 TREATMENT OF LARGE ROTATIONS

Geometric nonlinearities may produce large displacements and rotations. Unlike displacements, finite rotations (components 4, 5 and 6 of the displacement vector) can not be updated by simple vector addition. There are two different approaches to update rotations: namely, gimbal angle approach and rotation vector approach (implemented in Version 67).

The user has the option to select either of the two approaches. This is done by specifying PARAM,LANGLE,1 (default option) for the gimbal angle approach and PARAM,LANGLE,2 for the rotation vector approach in either the Bulk Data or the Case Control. The value of LANGLE can not be changed between subcases or at the restart, and is determined at the time of the cold start.

5.4.1 Gimbal Angle Approach

The gimbal angle approach is selected by specifying PARAM,LANGLE,1. In the gimbal angle approach, the three components of rotations at a grid point ($\theta_x, \theta_y, \theta_z$) are interpreted as finite rotations of a rigid body attached to the grid point subjected to three successive rotations as follows: first the body is rotated about the grid point's global z-axis by an amount θ_z ; then it is rotated about the reoriented y-axis by an amount θ_y ; and finally it is rotated about the twice reoriented x-axis by an amount θ_x . The three angles θ_x, θ_y and θ_z are called gimbal angles [5.6].

It can be shown that the same final orientation of the rigid body can also be obtained by first rotating the body by an amount of θ_x about the global x-axis, then by an amount of θ_y about the global y-axis, and finally by an amount of θ_z about the global z-axis. The output is in terms of θ_x, θ_y and θ_z , the three gimbal angles.

The rotation matrix \mathbf{R} corresponding to the gimbal angles is given by

$$\mathbf{R} = \mathbf{R}_z \mathbf{R}_y \mathbf{R}_x \quad (5.4.1)$$

where the matrices \mathbf{R}_x , \mathbf{R}_y , and \mathbf{R}_z are

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_x & -s_x \\ 0 & s_x & c_x \end{bmatrix} \quad (5.4.2)$$

$$\mathbf{R}_y = \begin{bmatrix} c_y & 0 & s_y \\ 0 & 1 & 0 \\ -s_y & 0 & c_y \end{bmatrix} \quad (5.4.3)$$

$$\mathbf{R}_z = \begin{bmatrix} c_z & -s_z & 0 \\ s_z & c_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.4.4)$$

with $c_i = \cos \theta_i$ and $s_i = \sin \theta_i$ for $i=x,y,z$ and θ_i are the three gimbal angles. Multiplying \mathbf{R}_z , \mathbf{R}_y , and \mathbf{R}_x one obtains

$$\mathbf{R} = \begin{bmatrix} c_y c_z & s_x s_y c_z - c_x s_z & c_x s_y c_z + s_x s_z \\ c_y s_z & s_x s_y s_z + c_x c_z & c_x s_y s_z - s_x c_z \\ -s_y & s_x c_y & c_x c_y \end{bmatrix} \quad (5.4.5)$$

Three tasks are performed for every iteration in the gimbal angle approach as follows:

Step 1. Update the Gimbal Angles

The gimbal angles are updated using

$$\begin{aligned} \theta_x^{i+1} &= \theta_x^i + \Delta\theta_x \\ \theta_y^{i+1} &= \theta_y^i + \Delta\theta_y \\ \theta_z^{i+1} &= \theta_z^i + \Delta\theta_z \end{aligned} \quad (5.4.6)$$

The expressions for the incremental gimbal angles are derived by transforming gimbal angles to global angles and equating them with those obtained from the equilibrium solution. If we assume the global incremental angles ($\delta\theta_i$, $i=x,y,z$) are small, we have

$$\begin{Bmatrix} \delta\theta_x \\ \delta\theta_y \\ \delta\theta_z \end{Bmatrix} = [\mathbf{R}_z \mathbf{R}_y] \begin{Bmatrix} \Delta\theta_x \\ 0 \\ 0 \end{Bmatrix} + [\mathbf{R}_z] \begin{Bmatrix} 0 \\ \Delta\theta_y \\ 0 \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ \Delta\theta_z \end{Bmatrix} \quad (5.4.7)$$

It should be noted that matrices \mathbf{R}_z and \mathbf{R}_y are based on the gimbal angles at the beginning of the iteration. This is reasonable because of the small angle assumption. Solving for the incremental gimbal angles, the following expressions are obtained

$$\begin{aligned} \Delta\theta_x &= (\delta\theta_x c_z + \delta\theta_y s_z)/c_y \\ \Delta\theta_y &= -\delta\theta_x s_z + \delta\theta_y c_z \\ \Delta\theta_z &= \delta\theta_z + (\delta\theta_x c_z + \delta\theta_y s_z)s_y/c_y \end{aligned} \quad (5.4.8)$$

where c_i and s_i , $i=x,y,z$ correspond to the gimbal angles at the beginning of iteration.

It can be seen from the above equations that there is a singularity at $\theta_y = 90^\circ$. This is called *gimbal lock*. The gimbal lock is avoided by switching temporarily to a different set of angles, called auxiliary angles, and then reverting back to the gimbal angles. This is possible because the rotation matrix is unique and there exist several ways to obtain the final orientation of a rigid body attached to a grid point. The interpretation of the auxiliary angles is that the rotation of a rigid body attached to a grid point is computed by first rotating the body by an amount of ϕ_x about the global x-axis, then by an amount of ϕ_y about the global y-axis, and finally by an amount of ϕ_z about the global x-axis. The algorithm to avoid gimbal lock is as follows:

1. Form rotation matrix $\mathbf{R}(\theta)$ based on the three gimbal angles at the beginning of the iteration.

2. Obtain the auxiliary angles ϕ from $\mathbf{R}(\theta)$.
3. Update the auxiliary angles, i.e., $\phi^{i+1} = \phi + \Delta\phi$.
4. Form updated rotation matrix from the updated auxiliary angles, i.e., $\mathbf{R}(\phi + \Delta\phi)$.
5. Recover the updated gimbal angles, i.e., θ^{i+1} .

This algorithm may sometimes lead to sudden changes in the gimbal angles. However, it does not affect the solution because the element routines use the rotation matrix which is unique.

Step 2. Eliminate Rigid Body Rotation

The rigid body rotation is removed from the rotation matrix as follows:

$$\tilde{\mathbf{R}} = \mathbf{T}_{bd}^T \mathbf{T}_{bg} \mathbf{R} \mathbf{T}_{bg}^T \mathbf{T}_{be} = \mathbf{T}_{dg} \mathbf{R} \mathbf{T}_{eg}^T \quad (5.4.9)$$

where $\tilde{\mathbf{R}}$ is the rotation matrix corresponding to angular deformations. The matrices \mathbf{T}_{bd} , \mathbf{T}_{bg} , and \mathbf{T}_{be} transform the deformed element coordinates, the global coordinates, and the undeformed element coordinates, respectively, to basic coordinates. The matrices \mathbf{T}_{dg} and \mathbf{T}_{eg} transform the global coordinates to deformed and undeformed element coordinates, respectively.

Step 3. Calculate Angular Deformation

The gimbal angle approach assumes that the angular deformations are small. Hence, the angular deformations are recovered as follows:

$$\begin{aligned} \delta\tilde{\theta}_x &= \arcsin \frac{1}{2}(\tilde{R}_{32} - \tilde{R}_{23}) \\ \delta\tilde{\theta}_y &= \arcsin \frac{1}{2}(\tilde{R}_{13} - \tilde{R}_{31}) \\ \delta\tilde{\theta}_z &= \arcsin \frac{1}{2}(\tilde{R}_{21} - \tilde{R}_{12}) \end{aligned} \quad (5.4.10)$$

These angular deformations are then used to calculate the element forces.

5.4.2 Rotation Vector Approach

The rotation vector approach is selected by specifying PARAM,LANGLE,2. In the rotation vector approach the three components of rotation at a grid point, denoted as ψ_x , ψ_y , ψ_z to distinguish from the gimbal angles, are interpreted as three components of a rotation vector (ψ). The orientation of a rigid body attached to a grid point is obtained by rotating the body by an amount Ψ , called the magnitude of rotation, about an axis of rotation \mathbf{p} :

$$\psi = \Psi \mathbf{p} \quad (5.4.11)$$

where

$$\Psi = \|\psi\| = (\psi^T \psi)^{\frac{1}{2}} \quad (5.4.12)$$

and

$$\boldsymbol{\psi} = \begin{Bmatrix} \psi_x \\ \psi_y \\ \psi_z \end{Bmatrix} ; \quad \mathbf{p} = \begin{Bmatrix} p_x \\ p_y \\ p_z \end{Bmatrix} \quad (5.4.13)$$

The output is in terms of ψ_x , ψ_y , and ψ_z , the three components of rotation vector $\boldsymbol{\psi}$.

Let p_i be a component of the unit vector parallel to the axis of rotation and Ψ be the magnitude of rotation. Then the rotation matrix is as follows [5.3]:

$$\mathbf{R} = \begin{bmatrix} \cos \Psi + p_x^2(1 - \cos \Psi) & -p_z \sin \Psi + p_x p_y(1 - \cos \Psi) & p_y \sin \Psi + p_x p_z(1 - \cos \Psi) \\ p_z \sin \Psi + p_x p_y(1 - \cos \Psi) & \cos \Psi + p_y^2(1 - \cos \Psi) & -p_x \sin \Psi + p_y p_z(1 - \cos \Psi) \\ -p_y \sin \Psi + p_x p_z(1 - \cos \Psi) & p_x \sin \Psi + p_y p_z(1 - \cos \Psi) & \cos \Psi + p_z^2(1 - \cos \Psi) \end{bmatrix} \quad (5.4.14)$$

Like the gimbal angle approach, three tasks are also performed in the rotation vector approach. The details of the three tasks are:

Step 1. Update the Rotation Vector

The rotation vector can be updated by successive rotations. First update the rotation matrix, i.e.,

$$\mathbf{R}^{i+1} = \delta \mathbf{R} \mathbf{R}^i \quad (5.4.15)$$

where $\delta \mathbf{R}$ is the rotation matrix for the current iteration, and then recover the updated rotation vector from the updated rotation matrix \mathbf{R}^{i+1} . The software employs an algorithm which assumes that the incremental rotations ($\delta \psi_i$ $i=x,y,z$) in any iteration are small. That is $\delta \mathbf{R}$ can be approximated by

$$\delta \mathbf{R} = \begin{bmatrix} 1 & -\delta \psi_z & \delta \psi_y \\ \delta \psi_z & 1 & -\delta \psi_x \\ -\delta \psi_y & \delta \psi_x & 1 \end{bmatrix} \quad (5.4.16)$$

In this algorithm the updated rotation vector $\boldsymbol{\psi}^{i+1}$ is split into a scalar and a vector (no longer unit vector):

$$\boldsymbol{\psi}^{i+1} = \Psi^{i+1} \mathbf{p}^{i+1} \quad (5.4.17)$$

with

$$\Psi^{i+1} = \Psi^i + \Delta \Psi \quad (5.4.18)$$

and

$$\mathbf{p}^{i+1} = \mathbf{p}^i + \Delta \mathbf{p} \quad (5.4.19)$$

The scalar increment, $\Delta \Psi$, and the vector increment, $\Delta \mathbf{p}$, are recovered from given Ψ^i , \mathbf{p}^i and $\delta \boldsymbol{\psi}$, where $\delta \boldsymbol{\psi}$ is obtained from equilibrium iteration. Comparing the elements of the matrix Eq. (5.4.15) and after some transformations the following expressions are obtained:

$$\Delta \Psi = \mathbf{p}^{iT} \delta \boldsymbol{\psi} \quad (5.4.20)$$

$$\Delta p_x = \frac{(1 + \cos \Psi)(\delta \psi_x - \Delta \Psi p_x) + (\delta \psi_y p_z - \delta \psi_z p_y) \sin \Psi}{2(\sin \Psi + \Delta \Psi \cos \Psi)} \quad (5.4.21)$$

$$\Delta p_y = \frac{(1 + \cos \Psi)(\delta \psi_y - \Delta \Psi p_y) + (\delta \psi_z p_x - \delta \psi_x p_z) \sin \Psi}{2(\sin \Psi + \Delta \Psi \cos \Psi)} \quad (5.4.22)$$

$$\Delta p_z = \frac{(1 + \cos \Psi)(\delta \psi_z - \Delta \Psi p_z) + (\delta \psi_x p_y - \delta \psi_y p_x) \sin \Psi}{2(\sin \Psi + \Delta \Psi \cos \Psi)} \quad (5.4.23)$$

Special care has to be taken at the singularity points $\Psi = 0$ and $\Psi = \pi$. The angles ψ^i as well as the updated angles ψ^{i+1} have to be checked for the singularity points. Therefore, one needs an estimate for the updated angle ψ^{i+1} through vector addition of ψ^i and $\delta \psi$:

$$\psi^{i+1} = \psi^i + \delta \psi \quad (5.4.24)$$

The singularity points are checked for the angle ψ^i and ψ^{i+1} . Whenever ψ^i or ψ^{i+1} are close to a singularity with a tolerance of $\pm \epsilon$, the angles are updated through simple vector addition according to Eq. (5.4.24). The vector addition is consistent with the first order update, and does not affect the converged solution. The increment in Ψ is limited to π per iteration.

Step 2. Eliminate Rigid Body Rotation

The procedure to eliminate rigid body rotation is the same as that used in the gimbal angle approach.

Step 3. Calculate Angular Deformation

The angular deformations for the current iteration are calculated from the rotation matrix $\tilde{\mathbf{R}}$, which is of the same form as given in Eq. (5.4.14), as follows: the magnitude $\tilde{\Psi}$ is obtained from the first invariant of $\tilde{\mathbf{R}}$. That is:

$$\tilde{\Psi} = \arccos \left(\frac{1}{2} \left[\sum_{i=1}^3 \tilde{R}_{ii} - 1 \right] \right) \quad (5.4.25)$$

The components of the rotation vector \mathbf{p} are obtained from the off-diagonal components of $\tilde{\mathbf{R}}$:

$$p_x = \frac{1}{2} \frac{\tilde{R}_{32} - \tilde{R}_{23}}{\sin \Psi} \quad (5.4.26)$$

$$p_y = \frac{1}{2} \frac{\tilde{R}_{13} - \tilde{R}_{31}}{\sin \Psi} \quad (5.4.27)$$

$$p_z = \frac{1}{2} \frac{\tilde{R}_{21} - \tilde{R}_{12}}{\sin \Psi} \quad (5.4.28)$$

It becomes obvious that the recovery procedure outlined above has singularities at $\tilde{\Psi} = 0, \pi$, and therefore is only valid in the range $0 < \Psi < \pi$. For the singularity at $\Psi = 0$ the small angle rotation matrix in Eq. (5.4.16) is used, and the singularity at $\Psi = \pi$ is avoided by restricting $\tilde{\Psi}$ to

$$0 \leq \tilde{\Psi} \leq \frac{\pi}{2} \quad (5.4.29)$$

This is not only reasonable but necessary to avoid unrealistic element distortions.

5.4.3 Bisection Due to Large Rotations

Both the rotation vector and gimbal angle approaches assume that the incremental rotations from the equilibrium iterations satisfy the small angle theory. However, in actual computations this may not be true. Therefore, the user has an option (default) to bisect the load if the incremental rotations are large. The user can choose to bisect the load or time step if $\delta\theta_i$, $i=x,y,z$ exceeds RTOLB specified in degrees in the eighth field of the second continuation entry for the NLPARM and the TSTEPNL bulk data entries. The current default value for RTOLB is 20° .

In nonlinear problems the solution is obtained by iterations, and since the small angle rotation matrix in Eq. (5.4.16) is a first order approximation, in many problems it may be a computing advantage to use a larger value for RTOLB. Using a larger value of RTOLB may result in more iterations. However, no accuracy is lost if the solution converges.

The gimbal angle approach also assumes small angular deformations (in performing Step 3). Some accuracy in the results may be lost if the angular deformations are large. Therefore, it is more appropriate to use the rotation vector approach if the angular deformations are expected to be large.

5.4.4 Output Interpretation

The user can specify enforced non-zero angular displacements. These angular displacements are always enforced about the global axes. However, the printed results from the gimbal angle approach are the three gimbal angles. The gimbal angles in general for 3D rotations are not the same as the global angles. Therefore, in the gimbal angle approach, the printed results may not be equal to the specified value. This should not cause any alarm to the user, as the results represent gimbal angles while the specified rotation is about the global axes.

Consider a rigid body attached to a grid point. Let the rigid body be subjected to two successive rotations of 20° (0.349 radians) about the $(0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ axis. The total rotation is 40° (0.698 radians) about the $(0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ axis. The output for the three rotations, i.e., components 4, 5 and 6 of the displacement vector for the grid point at the end of each rotation is as follows:

1. After First Rotation:

	θ_x	θ_y	θ_z
Gimble angle approach	0.0	2.4683×10^{-1}	2.4683×10^{-1}
Rotation vector approach	0.0	2.4683×10^{-1}	2.4683×10^{-1}

2. After Second Rotation:

	θ_x	θ_y	θ_z
Gimble angle approach	6.2192×10^{-2}	4.8617×10^{-1}	5.0885×10^{-1}
Rotation vector approach	0.0	4.9365×10^{-1}	4.9365×10^{-1}

The details of the calculations for the two approaches are as follows:

Gimbal Angle Approach

1. First Rotation Increment:

- Gimbal angles at the beginning of increment

$$\begin{aligned}\theta_x &= 0.0 \\ \theta_y &= 0.0 \\ \theta_z &= 0.0\end{aligned}$$

- Incremental angles from global iteration

$$\begin{aligned}\delta\theta_x &= 0.0 \\ \delta\theta_y &= 2.4683 \times 10^{-1} \\ \delta\theta_z &= 2.4683 \times 10^{-1}\end{aligned}$$

- Incremental gimbal angles from Eq. (5.4.8)

$$\begin{aligned}\Delta\theta_x &= 0.0 \\ \Delta\theta_y &= 2.4683 \times 10^{-1} \\ \Delta\theta_z &= 2.4683 \times 10^{-1}\end{aligned}$$

- Updated gimbal angles at the end of increment

$$\begin{aligned}\theta_x &= 0.0 \\ \theta_y &= 2.4683 \times 10^{-1} \\ \theta_z &= 2.4683 \times 10^{-1}\end{aligned}$$

2. Second Rotation Increment:

- Gimbal angles at the beginning of increment

$$\begin{aligned}\theta_x &= 0.0 \\ \theta_y &= 2.4683 \times 10^{-1} \\ \theta_z &= 2.4683 \times 10^{-1}\end{aligned}$$

- Incremental angles from global iteration

$$\begin{aligned}\delta\theta_x &= 0.0 \\ \delta\theta_y &= 2.4683 \times 10^{-1} \\ \delta\theta_z &= 2.4683 \times 10^{-1}\end{aligned}$$

- Incremental gimbal angles from Eq. (5.4.8)

$$\begin{aligned}\Delta\theta_x &= 6.2192 \times 10^{-2} \\ \Delta\theta_y &= 2.3941 \times 10^{-1} \\ \Delta\theta_z &= 2.6202 \times 10^{-1}\end{aligned}$$

- Updated gimbal angles at the end of increment

$$\begin{aligned}\theta_x &= 6.2192 \times 10^{-2} \\ \theta_y &= 4.8617 \times 10^{-1} \\ \theta_z &= 5.0885 \times 10^{-1}\end{aligned}$$

Rotation Vector Approach

1. First Rotation Increment:

- At the beginning of increment

$$\begin{aligned}\psi_x &= 0.0 \\ \psi_y &= 0.0 \\ \psi_z &= 0.0\end{aligned}$$

- Incremental angles from global iteration

$$\begin{aligned}\delta\psi_x &= 0.0 \\ \delta\psi_y &= 2.4683 \times 10^{-1} \\ \delta\psi_z &= 2.4683 \times 10^{-1}\end{aligned}$$

- Updated angles at the end of increment from Eq. (5.4.24)

$$\begin{aligned}\psi_x &= 0.0 \\ \psi_y &= 2.4683 \times 10^{-1} \\ \psi_z &= 2.4683 \times 10^{-1}\end{aligned}$$

2. Second Rotation Increment:

- At the beginning of increment

$$\begin{aligned}\psi_x &= 0.0 \\ \psi_y &= 2.4683 \times 10^{-1} \\ \psi_z &= 2.4683 \times 10^{-1}\end{aligned}$$

for which

$$\psi = (\psi^T \psi)^{\frac{1}{2}} = 3.4907 \times 10^{-1}$$

$$p = \begin{Bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{Bmatrix}$$

- Incremental angles from global iteration

$$\begin{aligned}\delta\psi_x &= 0.0 \\ \delta\psi_y &= 2.4683 \times 10^{-1} \\ \delta\psi_z &= 2.4683 \times 10^{-1}\end{aligned}$$

for which

$$\Delta\psi = p^T \delta\psi = 3.4907 \times 10^{-1}$$

$$\begin{aligned}\Delta p_x &= 0 \\ \Delta p_y &= 0 \\ \Delta p_z &= 0\end{aligned}$$

- At the end of increment

$$p = \begin{Bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{Bmatrix}$$

$$\psi = \psi + \Delta\psi = 3.4907 \times 10^{-1} + 3.4907 \times 10^{-1} = 6.9814 \times 10^{-1}$$

$$\text{updated angles} = \psi p$$

$$= 6.9814 \times 10^{-1} \begin{Bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{Bmatrix} = \begin{Bmatrix} 0.0 \\ 4.9366 \times 10^{-1} \\ 4.9366 \times 10^{-1} \end{Bmatrix}$$

As a result

$$\begin{aligned}\psi_x &= 0.0 \\ \psi_y &= 4.9366 \times 10^{-1} \\ \psi_z &= 4.9366 \times 10^{-1}\end{aligned}$$

5.5 VERIFICATION: LARGE ROTATION OF A BEAM

This example compares geometric nonlinear analysis using SOL 64 and SOL 66. SOL 64 is an older trial DMAP developed for geometric nonlinear analysis only, whereas SOL 66 (or 106) is more recently developed and contains many features not available in SOL 64. SOL 66 is also simpler to use because the solution is controlled by the NLPARM Bulk Data instead of subcases and parameters as in SOL 64. It is therefore preferable to use SOL 66 (or 106) for the geometric nonlinear analysis. Two problems are studied: a beam with an axial end force and a beam with an end moment. SOL 64 and SOL 66 in Version 66A are used for the analysis of the two problems. The main feature of these problems is geometric nonlinearity due to large rotations.

5.5.1 Problem Description

A uniform beam is clamped at one end and free at the other end. The beam has a length of 12 inches and a cross-sectional area of 0.15 in^2 . The Young's modulus is $20 \times 10^6 \text{ lb/in}^2$ and the moment of inertia I_{zz} is $2.813 \times 10^{-4} \text{ in}^4$. Two cases are analyzed according to the loading conditions at the free end of the beam: axial end force and end moment.

Beam with Axial End Force

Initially, the beam is subjected to an axial force $P=0.9P_{cr}$ and a small lateral force $P_i=0.1P_{cr}$ (initial disturbance) at the free end as shown in Fig. 5.5.1. The lateral force is subsequently removed and large rotations of the beam are produced when the axial force is increased above the critical value P_{cr} . For this problem, the critical load which is calculated by Timoshenko and Gere [5.7] is

$$P_{cr} = \frac{\pi^2 EI}{4L^2}.$$

The beam rotation is determined for the ratios of P/P_{cr} of 1.152, 1.518, 2.541, 4.029, and 9.116 in order to compare the results with those of Timoshenko and Gere.

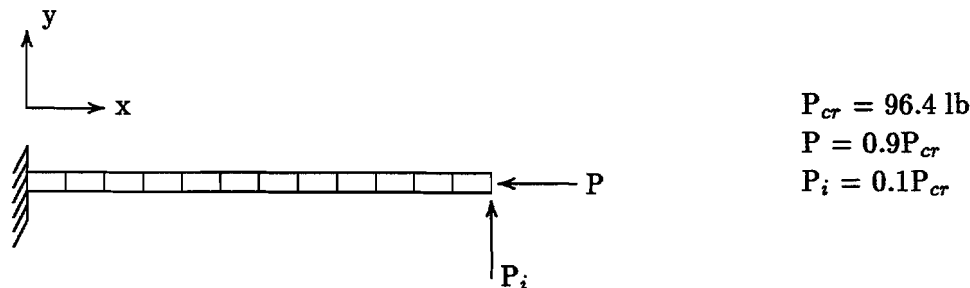


Figure 5.5.1 Beam with an Axial End Force.

Beam with End Moment

A moment with a magnitude of 3000 in-lb is applied in the +z direction at the free end of the beam, as shown in Fig. 5.5.2, and large rotations are produced.

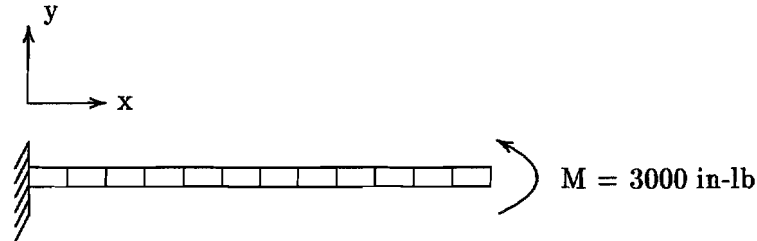


Figure 5.5.2 Beam with an End Moment.

5.5.2 Finite Element Model

The beam is modeled by twelve BEAM elements with linear elastic material. The clamped boundary condition at one end of the beam is imposed by constraining all six components of GRID 100. The applied load at the free end of the beam (GRID 112) is specified using the FORCE Bulk Data entry for the axial end force case and the MOMENT Bulk Data entry for the end moment case. Geometric nonlinearity effects are included in the SOL 66 analysis by specifying PARAM,LGDISP,1. The solution strategy is provided by the NLPARM Bulk Data entry in SOL 66. In the SOL 64 analysis, the solution is controlled by subcases. The input data for SOL 64 and SOL 66 are given in Tables 5.5.1 and 5.5.2, respectively, for the axial end force case and in Tables 5.5.3 and 5.5.4, respectively, for the end moment case.

5.5.3 Analysis Procedure

SOL 64 Analysis

Subcases and parameters control the number of iterations, load vector updates, and stiffness matrix updates in SOL 64. The stiffness matrix is updated after every iteration and there is one load increment per subcase. The first two subcases correspond to the linear elastic and the differential stiffness solutions, respectively. PARAM,TESTSE,1.-10 is included in the Bulk Data for an automatic convergence test based on strain energy.

The analysis is performed in seven subcases for the beam with axial end force. The initial loading conditions (axial force of $0.9P_{cr}$ and lateral force of $0.1P_{cr}$) are contained in the first two subcases. Five additional subcases are included to specify the five axial force ratios of P/P_{cr} with no lateral force.

For the beam with end moment, the analysis is performed in twelve subcases. The first two subcases correspond to the linear elastic and differential solutions, and the last ten subcases correspond to the ten moment increments.

SOL 66 Analysis

In SOL 66, the NLPARM Bulk Data controls the load increment size and the iteration strategy. Automatic convergence tests are provided and no special parameters need to be included to control the solution strategy.

The analysis is performed in six subcases for the beam with axial end force. The first subcase, which contains the initial loading, triggers large bending beyond the critical buckling load. The lateral force is removed and each of the remaining subcases correspond to an increasing axial force ratio of P/P_{cr} . Each subcase has one increment and the ITER option of KMETHOD is selected with KSTEP=1, specifying the full Newton's method.

For the beam with end moment, the analysis is performed in one subcase which has ten increments. Runs are made with different KMETHOD options (ITER, SEMI, or AUTO), which specify the stiffness matrix update strategy, while KSTEP=1. The analysis executes successfully with each of the KMETHOD options when the line search and BFGS updates are suppressed (MAXLS=0 and MAXQN=0). However, when the AUTO option is selected with line search and BFGS updates, the solution diverges at the first load increment. The performance in terms of the number of iterations and stiffness updates is tabulated below.

Performance for Beam with End Moment

KMETHOD	No. of K updates	No. of iterations
ITER	50	50
SEMI	40	138
AUTO	40	121

5.5.4 Analysis Results

The analysis results for the beam with axial end force and the values calculated by Timoshenko and Gere [5.7] are tabulated below for the ratios of P/P_{cr} . The distance y_{end} is the displacement of the free end of the beam in the vertical direction and x_{end} is the horizontal distance from the free end of the beam to its clamped end. The tangent at the free end of the beam has an angle θ_{end} with the horizontal. The solver results are in good agreement with the values of Timoshenko and Gere. The relation between the axial and lateral displacement of the free end of the beam (u_{112} and v_{112}) and the axial force is shown in Fig. 5.5.3. The displaced shapes of the beam are shown in Fig. 5.5.4.

Load-Deflection Data of Beam with Axial End Force

P/P_{cr}	x_{end}/L	y_{end}/L	θ_{end}
1.152	0.743 (0.741)	0.592 (0.593)	59.8 (60.0)
1.518	0.351 (0.349)	0.792 (0.792)	99.9 (100.0)
2.541	0.105 (0.107)	0.752 (0.750)	140.0 (140.0)
4.029	0.339 (0.340)	0.627 (0.625)	160.1 (160.0)
9.116	0.575 (0.577)	0.424 (0.421)	176.1 (176.0)

Notes: values in () denote solution in Timoshenko and Gere [5.7]

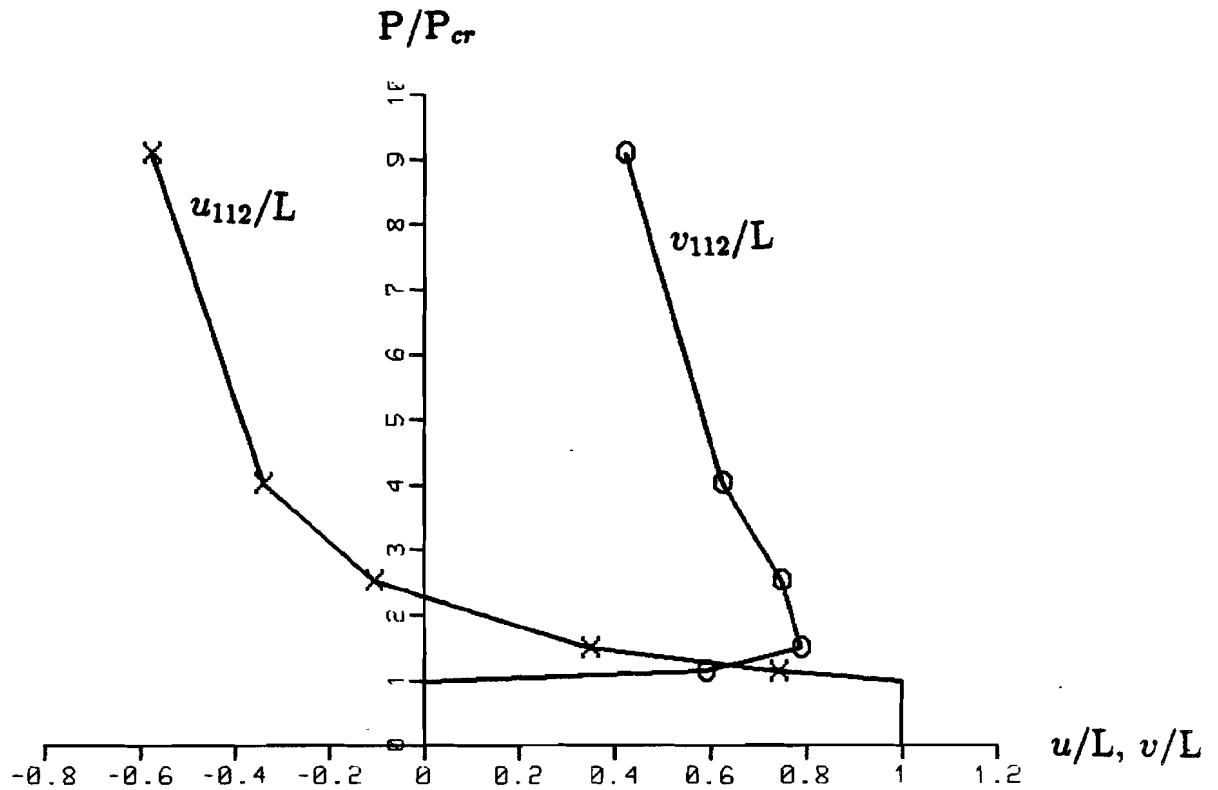


Figure 5.5.3 Load vs. Displacement at Free End of Beam with Axial End Force.

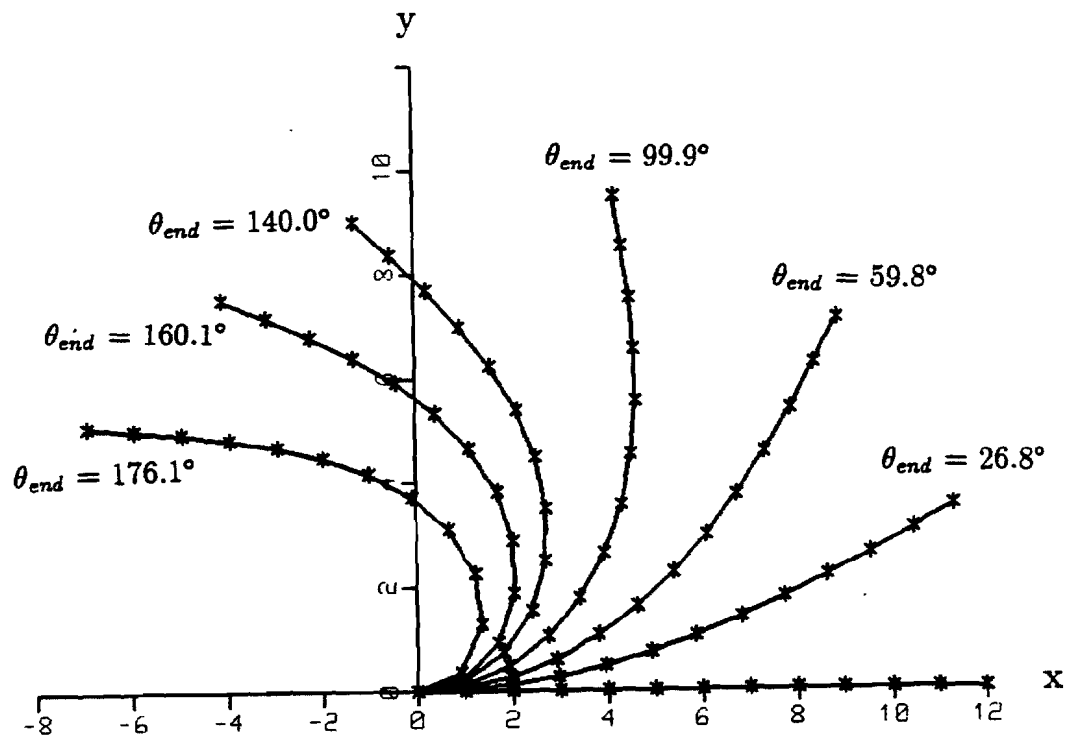


Figure 5.5.4 Deflection Curves for Beam with Axial End Force.

The displaced shapes of the beam with end moment are shown in Fig. 5.5.5. The deflection curves are labeled with the increment number (from 1 to 10). The moment magnitude and θ_{end} , the angle that the tangent at the free end of the beam has with the horizontal, are listed for each incremental step. Note that after ten increments, the beam has deformed into a complete circle.

inc	moment (in-lb)	θ_{end}
1	300	36.7°
2	600	73.3°
3	900	110.0°
4	1200	146.7°
5	1500	183.3°
6	1800	220.0°
7	2100	256.6°
8	2400	293.3°
9	2700	330.0°
10	3000	366.6°

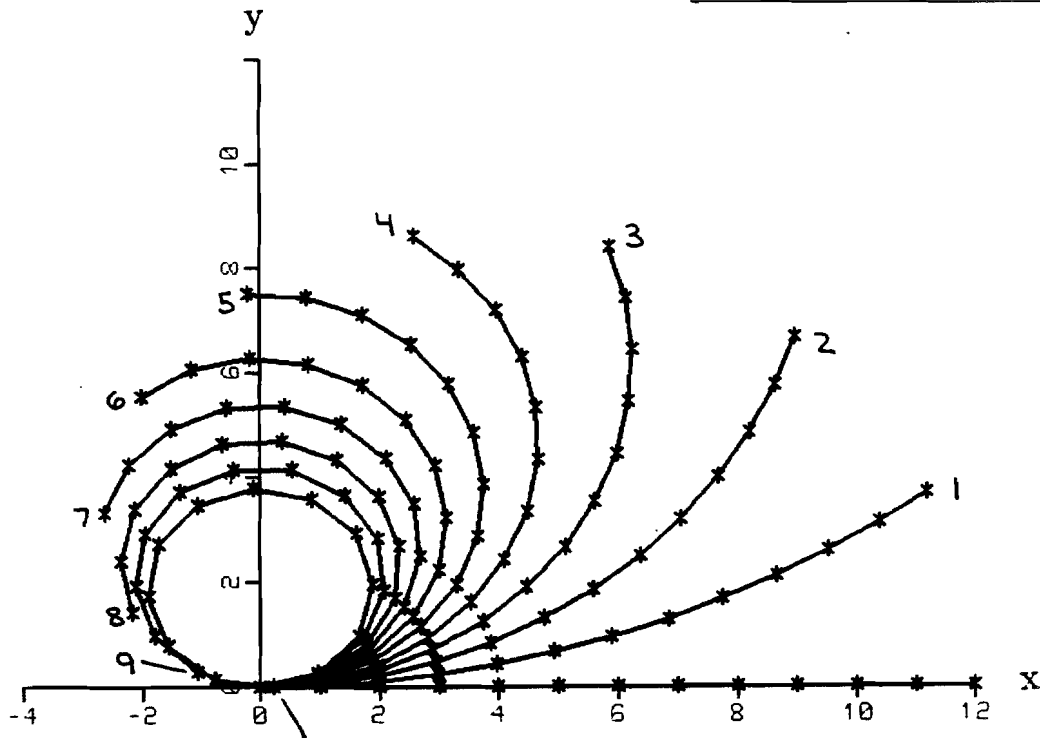


Figure 5.5.5 Deflection Curves for Beam with End Moment.

Table 5.5.1 SOL 64 Input Data Listing for Beam with Axial End Force

```

ID      PSTBUCK64, V66A $ APR 20-MAR-90
TIME    5          $ CPU MINUTES
SOL     64          $ GEOMETRIC NONLINEAR ANALYSIS
DIAG    8
CEND
TITLE = LARGE DEFLECTIONS OF A BEAM --- V6402S
SUBTITLE = TIMOSHENKO AND GERE, THEORY OF ELASTIC STABILITY; P.76
LABEL = P(CRIT) = EI*(.5*PI/L)**2 = 96.4
ECHO = BOTH
  SEALL = ALL
  DISP = ALL
  SPCF = ALL
  ELFOR = ALL
  SPC = 100
SUBCASE 1      $ P/P(CR)=0.900 PLUS LATERAL
  LOAD = 900
SUBCASE 2      $ P/P(CR)=0.900 PLUS LATERAL
  LOAD = 900
SUBCASE 3      $ P/P(CR)=1.152
  LOAD = 1152
SUBCASE 4      $ P/P(CR)=1.518
  LOAD = 1518
SUBCASE 5      $ P/P(CR)=2.541
  LOAD = 2541
SUBCASE 6      $ P/P(CR)=4.029
  LOAD = 4029
SUBCASE 7      $ P/P(CR)=9.116
  LOAD = 9116
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 = ALL
  VIEW 0. , 0. , 0.
  AXES Z , X , Y
  PTITLE = TOP VIEW
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1 ORIGIN 1 SYMBOL 1
BEGIN BULK
$ PARAMETERS
PARAM TESTSE 1.-10
$ GEOMETRY
GRID 100      0.      0.      0.      345
GRID 101      1.      0.      0.      345
GRID 102      2.      0.      0.      345
GRID 103      3.      0.      0.      345
GRID 104      4.      0.      0.      345
GRID 105      5.      0.      0.      345
GRID 106      6.      0.      0.      345

```

GRID	107		7.	0.	0.		345
GRID	108		8.	0.	0.		345
GRID	109		9.	0.	0.		345
GRID	110		10.	0.	0.		345
GRID	111		11.	0.	0.		345
GRID	112		12.	0.	0.		345
\$ CONNECTIVITY							
CBEAM	1	10	100	101	0.	0.	1.
CBEAM	2	10	101	102	0.	0.	1.
CBEAM	3	10	102	103	0.	0.	1.
CBEAM	4	10	103	104	0.	0.	1.
CBEAM	5	10	104	105	0.	0.	1.
CBEAM	6	10	105	106	0.	0.	1.
CBEAM	7	10	106	107	0.	0.	1.
CBEAM	8	10	107	108	0.	0.	1.
CBEAM	9	10	108	109	0.	0.	1.
CBEAM	10	10	109	110	0.	0.	1.
CBEAM	11	10	110	111	0.	0.	1.
CBEAM	12	10	111	112	0.	0.	1.
\$ PROPERTIES							
PBEAM	10	11	.15	2.813-4 2.813-4			
MAT1	11	2.+7	1.+7				
\$ CONSTRAINTS							
SPC1	100	126	100				
\$ LOADING -- P(CR)=96.4							
FORCE	900	112		96.4	-0.900	.1	0.
FORCE	1152	112		96.4	-1.152	0.	0.
FORCE	1518	112		96.4	-1.518	0.	0.
FORCE	2541	112		96.4	-2.541	0.	0.
FORCE	4029	112		96.4	-4.029	0.	0.
FORCE	9116	112		96.4	-9.116	0.	0.
\$							
ENDDATA							

Table 5.5.2 SOL 66 Input Data Listing for Beam with Axial End Force

```

ID      PSTBUCK66, V66A $ APR 20-MAR-90
TIME    5          $ CPU MINUTES
SOL      66        $ NONLINEAR STATIC ANALYSIS
DIAG     8,50      $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = LARGE DEFLECTIONS OF A BEAM --- V6402S
SUBTITLE = TIMOSHENKO AND GERE, THEORY OF ELASTIC STABILITY; P.76
LABEL = P(CRIT) = EI*(.5*PI/L)**2 = 96.4
ECHO = BOTH
  SEALL = ALL
  DISP = ALL
  SPCF = ALL
  ELFOR = ALL
  SPC = 100
  NLPARM = 1000
SUBCASE 1      $ P/P(CR)=0.900 PLUS LATERAL
  LOAD = 900
SUBCASE 2      $ P/P(CR)=1.152
  LOAD = 1152
SUBCASE 3      $ P/P(CR)=1.518
  LOAD = 1518
SUBCASE 4      $ P/P(CR)=2.541
  LOAD = 2541
SUBCASE 5      $ P/P(CR)=4.029
  LOAD = 4029
SUBCASE 6      $ P/P(CR)=9.116
  LOAD = 9116
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 = ALL
  VIEW 0. , 0. , 0.
  AXES Z , X , Y
  PTITLE = TOP VIEW
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1 ORIGIN 1 SYMBOL 1
BEGIN BULK
$ PARAMETERS
PARAM  LGDISP  1
$
NLPARM 1000    1          ITER    1          YES    +NLP
+NLP                                0          0
$ GEOMETRY
GRID    100          0.      0.      0.          345
GRID    101          1.      0.      0.          345
GRID    102          2.      0.      0.          345
GRID    103          3.      0.      0.          345
GRID    104          4.      0.      0.          345

```

GRID	105		5.	0.	0.		345
GRID	106		6.	0.	0.		345
GRID	107		7.	0.	0.		345
GRID	108		8.	0.	0.		345
GRID	109		9.	0.	0.		345
GRID	110		10.	0.	0.		345
GRID	111		11.	0.	0.		345
GRID	112		12.	0.	0.		345
\$ CONNECTIVITY							
CBEAM	1	10	100	101	0.	0.	1.
CBEAM	2	10	101	102	0.	0.	1.
CBEAM	3	10	102	103	0.	0.	1.
CBEAM	4	10	103	104	0.	0.	1.
CBEAM	5	10	104	105	0.	0.	1.
CBEAM	6	10	105	106	0.	0.	1.
CBEAM	7	10	106	107	0.	0.	1.
CBEAM	8	10	107	108	0.	0.	1.
CBEAM	9	10	108	109	0.	0.	1.
CBEAM	10	10	109	110	0.	0.	1.
CBEAM	11	10	110	111	0.	0.	1.
CBEAM	12	10	111	112	0.	0.	1.
\$ PROPERTIES							
PBEAM	10	11	.15	2.813-4	2.813-4		
MAT1	11	2.+7	1.+7				
\$ CONSTRAINTS							
SPC1	100	126	100				
\$ LOADING -- P(CR)=96.38286							
FORCE	900	112		96.4	-0.900	.1	0.
FORCE	1152	112		96.4	-1.152	0.	0.
FORCE	1518	112		96.4	-1.518	0.	0.
FORCE	2541	112		96.4	-2.541	0.	0.
FORCE	4029	112		96.4	-4.029	0.	0.
FORCE	9116	112		96.4	-9.116	0.	0.
\$							
ENDDATA							

Table 5.5.3 SOL 64 Input Data Listing for Beam with End Moment

```

ID      BMMOM64, V66A $ APR 20-MAR-90
TIME    5          $ CPU MINUTES
SOL     64          $ GEOMETRIC NONLINEAR ANALYSIS
DIAG    8
CEND
TITLE   = LARGE DEFLECTION DEMO
ECHO    = BOTH
  SEALL = ALL
  DISP  = ALL
  SPCF  = ALL
  ELFOR = ALL
  SPC   = 100
SUBCASE 1
  LOAD  = 103
SUBCASE 2
  LOAD  = 103
SUBCASE 3
  LOAD  = 103
SUBCASE 4
  LOAD  = 106
SUBCASE 5
  LOAD  = 109
SUBCASE 6
  LOAD  = 112
SUBCASE 7
  LOAD  = 115
SUBCASE 8
  LOAD  = 118
SUBCASE 9
  LOAD  = 121
SUBCASE 10
  LOAD  = 124
SUBCASE 11
  LOAD  = 127
SUBCASE 12
  LOAD  = 130
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 = ALL
  AXES Z, X, Y
  VIEW 0., 0., 0.
  PTITLE = TOP VIEW
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1 ORIGIN 1 SYMBOL 1
BEGIN BULK
$ PARAMETERS
PARAM TESTSE 1.-10

```

\$ GEOMETRY

GRID	100	0.	0.	0.	345
GRID	101	1.	0.	0.	345
GRID	102	2.	0.	0.	345
GRID	103	3.	0.	0.	345
GRID	104	4.	0.	0.	345
GRID	105	5.	0.	0.	345
GRID	106	6.	0.	0.	345
GRID	107	7.	0.	0.	345
GRID	108	8.	0.	0.	345
GRID	109	9.	0.	0.	345
GRID	110	10.	0.	0.	345
GRID	111	11.	0.	0.	345
GRID	112	12.	0.	0.	345

\$ CONNECTIVITY

CBEAM	1	10	100	101	0.	0.	1.
CBEAM	2	10	101	102	0.	0.	1.
CBEAM	3	10	102	103	0.	0.	1.
CBEAM	4	10	103	104	0.	0.	1.
CBEAM	5	10	104	105	0.	0.	1.
CBEAM	6	10	105	106	0.	0.	1.
CBEAM	7	10	106	107	0.	0.	1.
CBEAM	8	10	107	108	0.	0.	1.
CBEAM	9	10	108	109	0.	0.	1.
CBEAM	10	10	109	110	0.	0.	1.
CBEAM	11	10	110	111	0.	0.	1.
CBEAM	12	10	111	112	0.	0.	1.

\$ PROPERTIES

PBEAM	10	11	.15	2.813-4	2.813-4
MAT1	11	2.+7	1.+7		

\$ CONSTRAINTS

SPC1	100	126	100
------	-----	-----	-----

\$ LOADING

MOMENT	103	112	0	300.	0.	0.	1.
MOMENT	106	112	0	600.	0.	0.	1.
MOMENT	109	112	0	900.	0.	0.	1.
MOMENT	112	112	0	1200.	0.	0.	1.
MOMENT	115	112	0	1500.	0.	0.	1.
MOMENT	118	112	0	1800.	0.	0.	1.
MOMENT	121	112	0	2100.	0.	0.	1.
MOMENT	124	112	0	2400.	0.	0.	1.
MOMENT	127	112	0	2700.	0.	0.	1.
MOMENT	130	112	0	3000.	0.	0.	1.

\$

ENDDATA

Table 5.5.4 SOL 66 Input Data Listing for Beam with End Moment

```

ID      BMMOM66, V66A $ APR 20-MAR-90
TIME    5          $ CPU MINUTES
SOL     66          $ NONLINEAR STATIC ANALYSIS
DIAG    8,50        $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = VERY LARGE DEFLECTION DEMO
ECHO = BOTH
  SEALL = ALL
  DISP = ALL
  SPCF = ALL
  ELFOR = ALL
  SPC = 100
SUBCASE 10
  LOAD = 120
  NLPARM = 1000
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 = ALL
  AXES Z, X, Y
  VIEW 0., 0., 0.
  PTITLE = TOP VIEW
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1 ORIGIN 1 SYMBOL 1
BEGIN BULK
$ PARAMETERS
PARAM  LGDISP  1
$
NLPARM 1000    10          ITER    1          YES    +NL1
+NL1          0          0
$ GEOMETRY
GRID    100          0.      0.      0.          345
GRID    101          1.      0.      0.          345
GRID    102          2.      0.      0.          345
GRID    103          3.      0.      0.          345
GRID    104          4.      0.      0.          345
GRID    105          5.      0.      0.          345
GRID    106          6.      0.      0.          345
GRID    107          7.      0.      0.          345
GRID    108          8.      0.      0.          345
GRID    109          9.      0.      0.          345
GRID    110         10.      0.      0.          345
GRID    111         11.      0.      0.          345
GRID    112         12.      0.      0.          345
$ CONNECTIVITY
CBEAM   1           10      100      101      0.      0.      1.
CBEAM   2           10      101      102      0.      0.      1.
CBEAM   3           10      102      103      0.      0.      1.

```

CBEAM	4	10	103	104	0.	0.	1.
CBEAM	5	10	104	105	0.	0.	1.
CBEAM	6	10	105	106	0.	0.	1.
CBEAM	7	10	106	107	0.	0.	1.
CBEAM	8	10	107	108	0.	0.	1.
CBEAM	9	10	108	109	0.	0.	1.
CBEAM	10	10	109	110	0.	0.	1.
CBEAM	11	10	110	111	0.	0.	1.
CBEAM	12	10	111	112	0.	0.	1.
\$ PROPERTIES							
PBEAM	10	11	.15	2.813-4	2.813-4		
MAT1	11	2.+7	1.+7				
\$ CONSTRAINTS							
SPC1	100	126	100				
\$ LOADING							
MOMENT	120	112	0	3000.	0.	0.	1.
\$							
ENDDATA							

5.6 CANONICAL APPROACH TO GEOMETRIC NONLINEARITY

In geometric linear analysis where the displacements are assumed small regardless of the material nonlinearity, it is implied that the strain-displacement relations $[B]$ stay constant. However, the element matrix, $[B]$, as well as the applied external loads vary with large displacements. Conventional methods [5.8–5.12] such as the total Lagrangian, updated Lagrangian, or Eulerian method treat these effects of large displacements by starting with the finite strain, in contrast to an approximate updated Lagrangian approach employed by the software based on the infinitesimal strain. These conventional methods which are based on the classical continuum mechanics theory [5.13–5.16], referred to as canonical approach, are reviewed in this section.

5.6.1 Fundamentals of Continuum Mechanics

The kinematics of continua may be described from two different points of view. If the motion of a body is described in terms of the undeformed configuration using a referential coordinate system (denoted by X) attached to the object material, the method is known as the Lagrangian (or material) formulation. In contrast, the description of the motion in terms of the deformed configuration using spatial coordinates fixed in space (denoted by x) is called the Eulerian (or spatial) formulation. The Lagrangian description may be interpreted as a mapping of the initial configuration into the current configuration, i.e.

$$x = x(X, t). \quad (5.6.1)$$

Conversely, the Eulerian formulation gives the initial position of the particle at time t in terms of the current position x , i.e.

$$X = X(x, t). \quad (5.6.2)$$

The most fundamental quantity of these descriptions is the material deformation gradient defined as

$$F = \left[\frac{x_i}{X_j} \right] = \begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix}. \quad (5.6.3)$$

Then, the spatial deformation gradient can be defined by F^{-1} .

Consider a squared length of an infinitesimal line element in terms of the deformation gradient, i.e.

$$dx^T dx = (F dX)^T (F dX) = dX^T C dX \quad (5.6.4)$$

with

$$C = F^T F \quad (5.6.5)$$

where Green's deformation tensor is defined by C . The deformation tensor similarly defined for the Eulerian formulation is called Cauchy's deformation tensor and is denoted by c . As is

apparent from the derivation, these deformation tensors are symmetric and positive-definite. The change in the squared length is used as a measure of deformation, i.e.

$$dx^T dx - dX^T dX = dX^T (C - I) dX = 2 dX^T E dX \quad (5.6.6)$$

in which the Lagrangian (also called Green's) finite strain tensor is defined to be

$$E = \frac{(C - I)}{2}. \quad (5.6.7)$$

The Eulerian formulation can be derived likewise, i.e.

$$dx^T dx - dX^T dX = dx^T (I - c) dx = 2 dx^T e dx \quad (5.6.8)$$

in which the Eulerian (also called Almansi's) finite strain tensor is defined to be

$$e = \frac{(I - c)}{2}. \quad (5.6.9)$$

Noting that $x = X + u$, the finite strain tensors can be expressed in terms of the displacement gradient, i.e.

$$E_{ij} = \frac{1}{2} \left[\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} + \frac{\partial u_k}{\partial X_i} \frac{\partial u_k}{\partial X_j} \right] \quad (5.6.10)$$

and

$$e_{ij} = \frac{1}{2} \left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{\partial u_k}{\partial x_i} \frac{\partial u_k}{\partial x_j} \right] \quad (5.6.11)$$

where the repeated indices represent summation. When components of the displacement gradient are sufficiently small compared to unity, the product terms are negligible and the finite strain tensors reduce to the infinitesimal strain definition. Then, the distinction between the Lagrangian description and the Eulerian description disappears; which is the basis of the small deformation theory. Notice that the expression for the finite strain tensor is exact and not a second order approximation.

The displacement gradient can be decomposed into a symmetric and a skew-symmetric part, i.e.

$$\left[\frac{\partial u_i}{\partial x_j} \right] = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right). \quad (5.6.12)$$

Within the framework of the infinitesimal theory, the symmetric part is the linear strain tensor and the skew-symmetric part is the linear rotation tensor. When the displacement gradient is finite, it can no longer be reduced to a strain and a rotation by an additive decomposition. According to the polar decomposition theorem, however, the multiplicative decomposition is possible if the deformation gradient is not a singular tensor, i.e.

$$F = R M = N R \quad (5.6.13)$$

where R is the orthogonal rotation tensor, and M and N are positive definite symmetric tensors, known as the right and left stretch tensors, respectively. It follows that

$$C = F^T F = M^2 \quad \text{and} \quad c^{-1} = F F^T = N^2 \quad (5.6.14)$$

by virtue of $R^T R = I$. A significant interpretation can be given to Eq. (5.6.14); e.g., each principal value of M , which is known as the stretch ratio ($\lambda = dx/dX$), is the square root of the corresponding principal value of C , and the principal directions of C and M coincide. Based on the polar decomposition, it can be shown that the finite strain tensors are invariants with respect to the rigid-body motion.

Stresses are usually measured in the current configuration, because the equilibrium must be established in the deformed configuration. The stress in the deformed state is known as the Cauchy stress, consistent with the Eulerian strain tensor. In the course of analysis, however, if the strains are referred to the original position of particles as in the Lagrangian description, stresses should be defined similarly. Two Piola-Kirchhoff stress tensors are defined in the undeformed (or referential) configuration. The first Piola-Kirchhoff stress tensor (also known as Lagrangian stress), denoted by T_{ij} , is defined to give the actual force acting on the deformed surface element with an area dS , but the force is assigned to the unit area of the corresponding undeformed surface dS_0 . This stress tensor is easy to define but has the disadvantage of being non-symmetric. The second Piola-Kirchhoff stress tensor (also called Kirchhoff stress), denoted by S_{ij} , is defined to give the corresponding force acting on and assigned to the undeformed surface element dS_0 , but the force is related to the actual force in the deformed surface by the deformation gradient. These Piola-Kirchhoff stress tensors are related to the Cauchy stress tensor as follows:

$$(\det F) [\sigma_{ij}] = F [S_{ij}] F^T = F [T_{ij}] \quad (5.6.15)$$

or inversely,

$$[S_{ij}] = (\det F) F^{-1} [\sigma_{ij}] (F^{-1})^T = [T_{ij}] (F^{-1})^T \quad (5.6.16)$$

where the determinant of the deformation gradient F represents the change in volume due to the deformation, i.e.

$$\det F = \frac{dx_1 dx_2 dx_3}{dX_1 dX_2 dX_3} = \frac{V}{V_0}. \quad (5.6.17)$$

The second Piola-Kirchhoff stress tensor has little physical meaning; however, it has advantages of being symmetric and independent of rigid-body rotation. It is the second Piola-Kirchhoff stress tensor that is energetically conjugate to the Lagrangian strain tensor, i.e., the strain energy is preserved by

$$\int_{V_0} S_{ij} \delta E_{ij} dV = \int_V \sigma_{ij} \delta \epsilon_{ij} dV \quad (5.6.18)$$

5.6.2 Incremental Formulations for Large Displacements and Strains

The kinematic tensors in the incremental process of flow problems such as in plasticity, creep or viscoplasticity are frequently expressed in terms of time rates in the process of linearization to follow the deformation paths. For instance, the velocity gradient rather than the displacement gradient is used, i.e.

$$L = \left[\frac{\partial v_i}{\partial x_j} \right] = D + W \quad (5.6.19)$$

with a symmetric part:

$$D = \frac{1}{2} (L + L^T)$$

and a skew symmetric part:

$$W = \frac{1}{2} (L - L^T)$$

where D and W are called the rate of deformation (or stretching) tensor and the vorticity (or spin) tensor, respectively. While the rate of deformation tensor is independent of rigid-body motion, the vorticity tensor represents the angular velocity of the rigid-body rotation. Recalling that the natural-strain increment is defined as

$$d\varepsilon = \frac{dl}{l}$$

to give the following logarithmic strain by an integration over the length, i.e.

$$\varepsilon = \int_{\ell_0}^{\ell} \frac{d\ell}{\ell} = \ln \frac{\ell}{\ell_0}. \quad (5.6.20)$$

The rate of deformation tensor can be used for a 3-dimensional generalization of the natural-strain increment in time dt , i.e.

$$d\varepsilon = D dt. \quad (5.6.21)$$

The rate of deformation and the vorticity tensors can be interpreted as the material derivatives of the Eulerian linear strain and rotation tensors, respectively. The material derivative is the time rate of change of any property of a moving continuum measured by an observer traveling with a specific particle under consideration. In contrast, the time rate of change measured at a fixed point in space is called the local rate of change or the spatial derivative. Consider an acceleration which is a material derivative of the velocity. When the velocity is described only in terms of spatial variables x which are implicit functions of time, i.e.

$$v = v(x, t),$$

then the acceleration can be calculated by

$$\dot{v} \equiv \left[\frac{dv}{dt} \right]_X = \left[\frac{\partial v}{\partial t} \right]_x + \frac{\partial v}{\partial x} \left[\frac{\partial x}{\partial t} \right]_X = \frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} \quad (5.6.22)$$

where the subscripts identify the variables held constant. The first term gives the local rate of change and the second term gives the convective rate of change. The material derivative of the deformation gradient can be obtained by

$$\dot{F} = \frac{\partial v}{\partial X} = \frac{\partial v}{\partial x} \frac{\partial x}{\partial X} = L F. \quad (5.6.23)$$

The principle of material frame-indifference states that constitutive equations must be invariant under changes of reference frame. Functions (vectors or tensors) are called frame-indifferent if their values are unchanged by the superposed rigid-body motion when the frame of reference is changed from $\{x, t\}$ to $\{x^*, t^*\}$. Specifically, when the change of frame is described by

$$x^*(X, t^*) = c(t) + Q(t) x(X, t), \quad (5.6.24)$$

the vectors and tensors transform according to

$$v^* = Q(t) v \quad (\text{for vectors}) \quad (5.6.25)$$

and

$$S^* = Q(t) S Q(t)^T \quad (\text{for tensors}) \quad (5.6.26)$$

where $Q(t)$ is a rotation tensor and $c(t)$ represents a translation of the coordinate origin. It can be shown, however, that the deformation gradient F transforms like a vector. It is noted that the spin tensor is not frame-indifferent and transforms by

$$W^* = \Omega + Q W Q^T \quad (5.6.27)$$

with

$$\Omega = \dot{Q} Q^T = -Q \dot{Q}^T = -\Omega^T \quad (5.6.28)$$

where \dot{Q} is the material derivative of Q , and Ω is the angular velocity tensor of the unstarred frame relative to the starred frame.

Of our particular interest is the time rate of change of the stress among constitutive variables in the incremental formulation because the material derivative of the stress ($\dot{\sigma}_{ij}$) is not frame-indifferent even though σ_{ij} is. However, it can be demonstrated that the frame-indifferent stress rate is defined as

$$\overset{\nabla}{\sigma} = \dot{\sigma} - W \sigma + \sigma W \quad (5.6.29)$$

which is known as the Jaumann stress rate or the corotational stress rate tensor and is denoted by ∇ over stress. The first term on the right side represents the material rate of change with respect to a fixed coordinate system and the last two terms represent the convective rate of change. Finite strains have a similar problem because the material derivative of the finite strain is not frame-indifferent. However, it can be shown that the rate of deformation tensor (D) is frame-indifferent, despite the fact that the velocity is not. As pointed out earlier, the rate of deformation tensor is a material derivative of the linear strain. Thus, the frame-indifferent

constitutive relations for the incremental formulation can be obtained by using the Jaumann stress rate and the infinitesimal strain rate, i.e.

$$\dot{\bar{\sigma}} = D \dot{\epsilon} \quad (5.6.30)$$

where D represents the constitutive relations.

Now we are ready to derive the incremental formulations for solving general nonlinear problems with large displacements and large strains. From the principle of virtual displacements, the basic equation to solve in accordance with the Eulerian approach is

$$\int_V \sigma_{ij} \delta \epsilon_{ij} dV = P_i \delta u_i \quad (5.6.31)$$

which is to be satisfied in the currently deformed configuration. The fundamental difficulty with the Eulerian approach is that the current geometry of the body, which is the referential configuration, is yet to be determined. Two popular approaches to the incremental formulations are considered here: the total Lagrangian and the updated Lagrangian approaches, both based on the Lagrangian description of kinematics. In the total Lagrangian formulation, all the variables are referred to the initial configuration. Without loss of generality, however, such a natural procedure of the Lagrangian formulation may be applied with an arbitrary choice of the referential configuration. When the choice of the referential configuration is the preceding incremental load (or time) step, the method is known as the updated Lagrangian formulation.

The basic equation can be rewritten for the total Lagrangian formulation as

$$\int_{0V} {}_0^t S_{ij} \delta_0^t E_{ij} d_0V = {}_0^t P_i \delta_0^t u_i \quad (5.6.31a)$$

and for the updated Lagrangian formulation as

$$\int_{tV} {}_t^t S_{ij} \delta_t^t E_{ij} d_tV = {}_t^t P_i \delta_t^t u_i \quad (5.6.31b)$$

where the left superscripts refer to the current configuration of the body and the left subscripts refer to the referential coordinate axes, with 0, t , and t' denoting the initial, the t -th incremental, and the $(t+1)$ th incremental steps, respectively. It is noted that the expression for the updated Lagrangian formulation is obtained by substituting t for the left subscripts 0 in the total Lagrangian formulation. In the total Lagrangian formulation, the incremental process of multiple load steps is applied to the Lagrangian formulation. On the other hand, the updated Lagrangian formulation may be regarded as an application of the Lagrangian method to a single load step with the referential configuration brought up-to-date continuously, treating the previously converged configuration as an initial geometry. Now it remains to linearize Eqs. (5.6.31a) and (5.6.31b) to find the tangential stiffness. It will be assumed for simplicity that the applied load is independent of deformation.

5.6.3 Total Lagrangian vs. Updated Lagrangian in Finite Elements

Recalling that ${}^t u = {}^t u + u$ for the total Lagrangian formulation, the stress and strain (omitting indices) may be decomposed as follows:

$${}^t_0 S = {}^t_0 S + {}_0 S \quad (5.6.32)$$

and

$${}^t_0 E = {}^t_0 E + {}_0 E_1 + {}_0 E_2 + {}_0 E_3 \quad (5.6.33)$$

with

$$\begin{aligned} {}^t_0 E &= \frac{1}{2} \left[\frac{\partial {}^t u_i}{\partial {}^0 x_j} + \frac{\partial {}^t u_j}{\partial {}^0 x_i} + \frac{\partial {}^t u_k}{\partial {}^0 x_i} \frac{\partial {}^t u_k}{\partial {}^0 x_j} \right] \\ {}_0 E_1 &= \frac{1}{2} \left[\frac{\partial u_i}{\partial {}^0 x_j} + \frac{\partial u_j}{\partial {}^0 x_i} \right] \\ {}_0 E_2 &= \frac{1}{2} \left[\frac{\partial u_k}{\partial {}^0 x_i} \frac{\partial u_k}{\partial {}^0 x_j} \right] \\ {}_0 E_3 &= \frac{1}{2} \left[\frac{\partial {}^t u_k}{\partial {}^0 x_i} \frac{\partial u_k}{\partial {}^0 x_j} + \frac{\partial u_k}{\partial {}^0 x_i} \frac{\partial {}^t u_k}{\partial {}^0 x_j} \right]. \end{aligned}$$

Noting that

$$\delta {}^t_0 E = \delta ({}_0 E_1 + {}_0 E_2 + {}_0 E_3)$$

and

$$\delta {}^t u = \delta u,$$

the substitution of Eqs. (5.6.32) and (5.6.33) into Eq. (5.6.31a) gives the following equation with some rearrangement of terms:

$$\begin{aligned} \int_{{}_0 V} {}_0 S \delta ({}_0 E_1 + {}_0 E_2 + {}_0 E_3) d{}_0 V + \int_{{}_0 V} {}^t_0 S \delta {}_0 E_2 d{}_0 V \\ = {}^t_0 P \delta u - \int_{{}_0 V} {}^t_0 S \delta ({}_0 E_1 + {}_0 E_3) d{}_0 V. \end{aligned} \quad (5.6.34)$$

Notice that ${}_0 E_1$ is the linear strain increment causing a linear stiffness $[K^l]$, ${}_0 E_2$ is the nonlinear strain increment causing a stiffness $[K^d]$ due to the initial stress, and ${}_0 E_3$ is a cross-product term which causes a stiffness $[K^g]$ due to the initial displacement. If the strain-displacement relations are linearized by

$$\left[\frac{\partial {}_0 E}{\partial u} \right]_t = {}^t_0 B = {}^t_0 B_L + {}^t_0 B_N, \quad (5.6.35)$$

the equilibrium equation can be rewritten in matrices as

$$[K^l + K^g + K^d] \{u\} = \{{}_0 R\} \quad (5.6.36)$$

where (omitting the left-superscript t and left-subscript 0 of B),

$$\begin{aligned} K^i &= \int_{0V} [B_L^T {}_0D B_L] dV \\ K^g &= \int_{0V} [B_L^T {}_0D B_N + B_N^T {}_0D B_N + B_N^T {}_0D B_L] dV \\ K^d &= \int_{0V} \left[\frac{\partial B_N}{\partial u} \right]^T \{ {}^t_0 S \} dV \\ {}_0R &= {}^t_0 P - \int_{0V} [B_L + B_N]^T \{ {}^t_0 S \} dV \end{aligned}$$

with a constitutive relation

$${}^t_0 S = {}_0D {}^t_0 E. \quad (5.6.37)$$

During the iteration, the displacements are updated continuously, i.e., for the i -th iteration

$${}^t u^i = {}^t u^{i-1} + u^i \quad (5.6.38)$$

with which ${}^t_0 B_N$ and ${}^t_0 S$ are reevaluated to yield ${}^t_0 B_N$ and ${}^t_0 S$. Thus, the unbalanced force ${}_0R$ vanishes as the solution approaches convergence.

The total Lagrangian approach is not appropriate for the path-dependent inelastic problems such as a plasticity problem. The reason is that the constitutive relations ${}_0D$ in Eq. (5.6.37) must be defined with reference to the initial geometry, which is meaningful only for elastic materials such as a hyperelastic material. In fact, many researchers reported difficulties with the total Lagrangian method in those problems. Thus, we have to resort to the updated Lagrangian approach for general geometric nonlinear problems. The concept of the updated Lagrangian method is realized by updating the state variables at the end of each incremental step with respect to the converged configuration or convected coordinates, i.e.

$${}^t x = {}^0 x + {}^t u = {}^t x + u. \quad (5.6.39)$$

For the updated Lagrangian process, stresses and strains may be decomposed (omitting the left-subscript t) as follows:

$${}^t S = {}^t S + S \quad (5.6.40)$$

and

$${}^t E = E_1 + E_2 \quad (5.6.41)$$

with

$$E_1 = \frac{1}{2} \left[\frac{\partial u_i}{\partial {}^t x_j} + \frac{\partial u_j}{\partial {}^t x_i} \right]$$

and

$$E_2 = \frac{1}{2} \left[\frac{\partial u_k}{\partial {}^t x_i} \frac{\partial u_k}{\partial {}^t x_j} \right].$$

Noting that ${}^tS = \sigma$, the substitution of Eqs. (5.6.40) and (5.6.41) into Eq. (5.6.31b) gives, with some rearrangement of terms,

$$\int_{{}_tV} S \delta(E_1 + E_2) dV + \int_{{}_tV} \sigma \delta E_2 dV = {}^tP \delta u - \int_{{}_tV} \sigma \delta E_1 dV. \quad (5.6.42)$$

Introducing the constitutive relations at t by

$$S = {}^tD (E_1 + E_2) \quad (5.6.43)$$

and linearizing Eq. (5.6.42) by neglecting E_2 in the first term, the equilibrium equation can be written in a matrix form (omitting the left-subscript t of B and D) as

$$[K^l + K^d] \{u\} = {}^tP - \int_{{}_tV} B_L^T \{\sigma\} dV \quad (5.6.44)$$

with

$$K^l = \int_{{}_tV} [B_L^T {}^tD B_L] dV$$

and

$$K^d = \int_{{}_tV} \left[\frac{\partial B_N}{\partial u} \right]^T \{\sigma\} dV$$

where the current strain-displacement relation is obtained by

$$\left[\frac{\partial E}{\partial u} \right]_t = {}^tB = B_L + B_N \quad (5.6.45)$$

and the volume integration is performed over the current volume.

5.6.4 Constitutive Relations for Large Deformation

Difficulties arise in using the Lagrangian constitutive relations in Eq. (5.6.43). First of all, the Lagrangian strain increments are not additive while the elasto-plastic deformation process requires

$$d\varepsilon_{total} = d\varepsilon_e + d\varepsilon_p. \quad (5.6.46)$$

However, since the rate of deformation tensor is additive, we can establish the rate constitutive relations in terms of the Jaumann stress rate ($\dot{\sigma}$) and the rate of deformation tensor as presented in Eq. (5.6.30). The rate of deformation tensor (D) is related to the Lagrangian strain rate by

$$\dot{E} = F^T D F. \quad (5.6.47)$$

By allowing the coordinate axes of the referential configuration to be momentarily coincident with that of the current geometry, the distinction disappears and the Lagrangian strain rate may be used as a constitutive variable, i.e.

$$D = \dot{E} = \dot{E}_e + \dot{E}_p. \quad (5.6.48)$$

Recalling the generalization of the natural-strain increment defined in Eq. (5.6.21), an integration of Eq. (5.6.48) may be interpreted to give the logarithmic strain which is denoted by E^* , i.e.

$$E^* = \int \dot{E} dt. \quad (5.6.49)$$

This is another advantage of using the rate equations since we can now formulate the constitutive relations [tD] based on the true stress-strain curve obtained from the uniaxial test. It is noted that the generalized logarithmic strain, which has a physical meaning in the updated Lagrangian coordinates, is different from either the Lagrangian or the Eulerian strain.

For every iteration, displacements are updated by

$${}^t u^i = {}^t u^{i-1} + u^i. \quad (5.6.50)$$

and the stresses on the right-hand side of Eq. (5.6.44) by

$${}^t S^i = {}^t \sigma + S^i. \quad (5.6.51)$$

However, the frame-indifferent constitutive relations involve the Jaumann stress rate, which is related to $\dot{\sigma}$ by Eq. (5.6.29), in order to account for the rotation of the stress during the increment. Since the Cauchy stress and the Kirchhoff stress are related by Eq. (5.6.15), the material derivative of the Cauchy stress can be obtained by taking a material derivative of Eq. (5.6.15), i.e.

$$F \dot{S} F^T = (\det F)(\operatorname{div} v \sigma + \dot{\sigma} - L \sigma - \sigma L^T) \quad (5.6.52)$$

using Eq. (5.6.23) and the material derivative of $(\det F)$ from

$$\frac{d}{dt}(\det F) = \dot{J} = J v_{i,i} = J \operatorname{div} v. \quad (5.6.53)$$

By using Eq. (5.6.29) and the momentarily coincident referential configuration, Eq. (5.6.52) reduces to

$$\dot{S} = \dot{\sigma} - D \sigma - \sigma D + \text{div } v \sigma . \quad (5.6.54)$$

For incompressible materials, as in plasticity, Eq. (5.6.54) further reduces to

$$\dot{S} = \dot{\sigma} - \dot{E} \sigma - \sigma \dot{E} \quad (5.6.55)$$

using Eq. (5.6.19) with a substitution of \dot{E} for D . Then the stress increment is calculated by

$$S^i = \int_t^{t'} \dot{S}^i dt = \int_t^{t'} ({}^t D \dot{E} - \dot{E} \sigma - \sigma \dot{E}) dt . \quad (5.6.56)$$

It is noted that the volume integration is performed over the configuration at t in Eq. (5.6.42). As the solution converges, the stress becomes ${}^t S$, and should be transformed to the Cauchy stress at t' according to Eq. (5.6.15).

For the hyperelastic material, the Eulerian constitutive relations may be obtained from the Lagrangian constitutive relations by

$${}^t D = \frac{1}{\det F} F F_0 D F^T F^T . \quad (5.6.57)$$

The Lagrangian strain can be transformed to the Eulerian strain at t' by

$${}^{t'} E = [{}^t F]^{-1} {}^{t'} E [{}^t F^{-1}]^T \quad (5.6.58)$$

where $[{}^t F]$ is the deformation gradient at t' with reference to the configuration at t .

Chapter 6

MATERIAL NONLINEARITY AND CONSTITUTIVE RELATIONS

6.1 INTRODUCTION

6.1.1 Overview

Material nonlinearity becomes an issue in the analysis if the stress-strain relationship or the constitutive equations are nonlinear. Constitutive equations characterize the macroscopic behavior of the material, which stems from the microscopic constitution of the material. All the engineering materials are inherently nonlinear. From the mathematical point of view, it is not feasible to characterize a nonlinear material by a single set of equations for the entire range of environmental conditions, such as the loading, temperature, and the rate of deformation. Even in the restricted region of these conditions for specific needs, the mathematical model of the material is still an idealization (e.g., incompressibility) of more general nonlinearity.

Depending on the application or the situation, the material behavior may be idealized or simplified to account for certain effects which are important in the analysis. The linear elastic (Hookean) material is the simplest case of all in solid mechanics. As the strain exceeds the infinitesimal range, the material exhibits nonlinear behavior: nonlinear elastic if it is recoverable and plastic if it is irrecoverable. If the temperature effects on the material properties become an important issue, coupling between thermal and mechanical behavior should be properly taken into consideration, which is the realm of thermo-elasticity and thermo-plasticity. If the strain-rate has significant effects on the material response, we have to resort to the theories of visco-elasticity and visco-plasticity. Anisotropic effects add complexity to the material models of all kinds in solid mechanics.

Development of material constitutive models has been prolific in recent years. Detailed discussion in this book will be confined to the capabilities provided in MSC/NASTRAN.

6.1.2 User Interface

The data for nonlinear elastic or plastic material applications are contained in the Bulk Data entry MATS1, in connection with the MAT1 entry with the same MID. The MATS1 entry specifies the stress-dependent material properties according to the format shown below. A brief description of each of the fields is provided, along with default values and further elaborating remarks as applicable.

1	2	3	4	5	6	7	8	9	10
MATS1	MID	TID	TYPE	H	YF	HR	LIMIT1	LIMIT2	
MATS1	17	28	PLASTIC	0.0	1	1	2.+4		

MID Identification number of a MAT1 entry (Integer > 0).

TID Identification number of a TABLES1 or TABLEST entry (Integer ≥ 0). If H is given, then this field should be blank. See Remark 1.

TYPE Specifies the type of material nonlinearity: NLELAST (Nonlinear elastic) or PLASTIC (Elastoplastic). See Remarks 2 and 3.

H Work hardening slope (slope of stress vs. plastic strain) in units of stress (Real). For elastic-perfectly plastic cases, H=0.0 (default). For more than a single slope in the plastic range, the stress-strain data must be supplied on a TABLES1 entry referenced by TID and this field must be blank. See Remark 3.

YF Yield function (Integer) is selected by one of the following values:

- 1 von Mises (Default)
- 2 Tresca
- 3 Mohr-Coulomb
- 4 Drucker-Prager

HR Hardening Rule (Integer) is selected by one of the following values:

- 1 Isotropic hardening (Default)
- 2 Kinematic hardening
- 3 Combined hardening

LIMIT1 Parameter representing an initial yield point (Real). See Remark 4.

LIMIT2 Parameter representing the internal friction angle for the Mohr-Coulomb and Drucker-Prager yield criteria. ($0 \leq \text{Real} < 45^\circ$). See Remark 4.

Remarks:

1. If TID is given, TABLES1 entries (x_i, y_i) of stress-strain data (ε_k, Y_k) must conform to the following rules (see Figure 6.1.1):
 - (a) If TYPE is PLASTIC, the curve must be defined in the first quadrant. The first point must be at the origin ($x_1 = 0, y_1 = 0$) and the second point (x_2, y_2) must be at the initial yield point (Y_1 or 2c) specified on the MATS1 entry. The slope of the line joining the origin to the yield stress must be equal to the value of E on the MAT1 entry.
 - (b) If TYPE is NLELAST, the full stress-strain curve ($-\infty < x < \infty$) may be defined in the first and the third quadrant to accommodate different uniaxial compression data. If the curve is defined only in the first quadrant, then the curve must start at the origin ($x_1 = 0, y_1 = 0$) and the compression properties will be assumed identical to tension properties.
 - (c) If TYPE is PLASTIC and TID is given, it may not reference a TABLEST entry.
2. If TYPE is NLELAST, the stress-strain data given in the TABLES1 entry will be used to determine the stress for a given value of strain. The values H, YF, HR, LIMIT1, and LIMIT2 will not be used in this case.

For nonlinear elastic temperature-dependent material properties, TID must reference a TABLEST entry and TABLEST subsequently references TABLES1 entries for different temperatures. Also a MATT1 entry should be supplied to specify the temperature dependence of material properties.

3. If TYPE is PLASTIC, either the table identification, TID, or the work hardening slope, H, may be specified but not both. If the table ID is omitted, the work hardening slope, H, should be specified in field 5 unless the material is perfectly-plastic. The plasticity modulus (H) is related to the tangential modulus (E_T) by

$$H = \frac{E_T}{1 - \frac{E_T}{E}}$$

where E is the elastic modulus and $E_T (= \frac{dY}{d\varepsilon})$ is the slope of the uniaxial stress-strain curve in the plastic region. See Figure 6.1.2.

4. LIMIT1 and LIMIT2 are parameters used in the yield function and must be defined as follows:

Yield Function	LIMIT1	LIMIT2
von Mises (1) or Tresca (2)	Yield stress in tension, Y_1	Not used
Mohr-Coulomb (3) or Drucker-Prager (4)	2*Coheision, $2c$ (in stress units)	Angle of internal friction ϕ (in degrees)

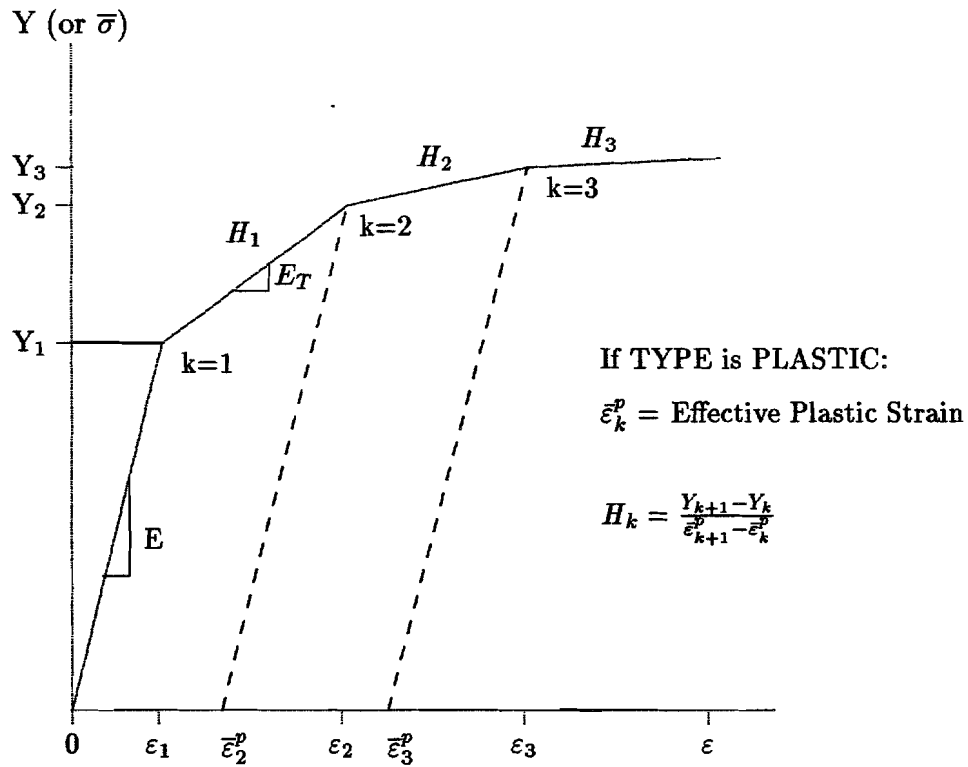


Figure 6.1.1. Stress-Strain Curve Definition with TABLES1 Entry.

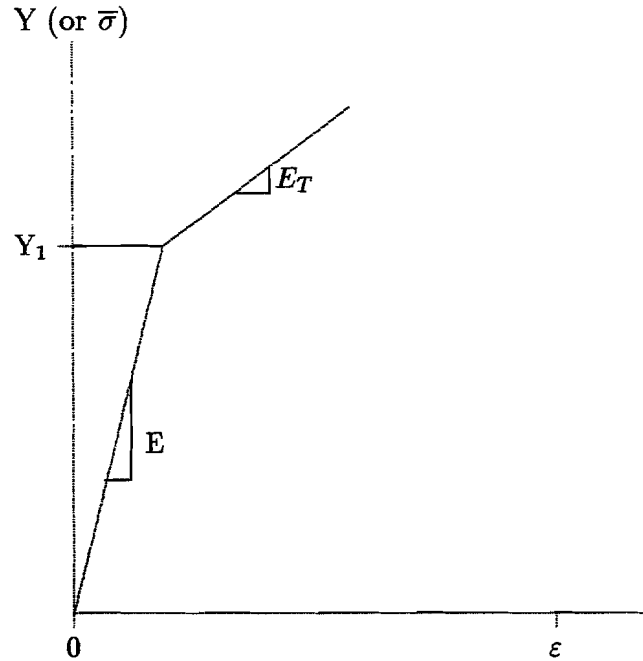


Figure 6.1.2. Stress-Strain Curve Definition for a Single H.

The parameters in the MATS1 Bulk Data entry pertain only to nonlinear elastic and plastic material nonlinearities. The required data for creep analysis are contained in the Bulk Data entry CREEP, which defines the creep characteristics based on experimental data or known empirical creep laws. In order to activate the creep analysis, the CREEP entry must contain the same MID as an associated MAT1 Bulk Data entry and the NLPARM Bulk Data entry must have a positive real value in the DT field. The DT value specifies the incremental time interval for creep analysis. The time unit in the CREEP entry used to define the creep characteristics with coefficients a through g must be consistent with the DT time unit. The CREEP entry format is shown below with default values, followed by a brief description of each field and applicable remarks.

1	2	3	4	5	6	7	8	9	10
CREEP	MID	RT	EXP	FORM	TIDKP	TIDCP	TIDCS	THRESH	
CREEP	8	0.	1.E-9	CRLAW					

	TYPE	a	b	c	d	e	f	g	
	121	6.985E-6	2.444	7.032E-4	0.1072	6.73E-9	0.1479	3.	

MID Identification number of a MAT1 entry (Integer > 0).

RT Reference temperature (T_0) at which creep characteristics are defined (Real). Default = 0.0. See Remark 1.

EXP Temperature-dependent term, $e^{-\frac{\Delta H}{RT_0}}$, in the creep rate expression ($0.0 < \text{Real} \leq 1.0$). Default = 1.0E-9. See Remark 1.

FORM Specifies the form of the input data defining creep characteristics: CRLAW (empirical creep law) or TABLE (tabular input data of creep model parameters). See Remark 2.

TIDKP Identification number of a TABLES1 entry for specification of the creep model parameter $K_p(\sigma)$ (Integer > 0). See Remark 3.

TIDCP Identification number of a TABLES1 entry for specification of the creep model parameter $C_p(\sigma)$ (Integer > 0). See Remark 3.

TIDCS Identification number of a TABLES1 entry for specification of the creep model parameter $C_s(\sigma)$ (Integer > 0). See Remark 3.

THRESH Threshold limit for creep process corresponding to the threshold strain ($0.0 < \text{Real} < 1.0\text{E}-3$). Default = 1.0E-5. Threshold stress under which creep does not occur is computed as THRESH*E, where E is the Young's Modulus.

TYPE Specifies the identification number (three-digit integer) of the empirical creep law type ($111 \leq \text{Integer} \leq 300$). See Remark 4.

a-g Coefficients of the empirical creep law specified in the TYPE field (Real). See Remark 4.

Remarks:

1. Creep law coefficients (a-g) are usually determined by least squares fit of experimental data obtained under a constant temperature. The reference temperature (T_0) at which creep behavior is characterized must be specified in the RT field if the temperature of the structure is different from this reference temperature. The unit of the temperature input (°F or °C) must be indicated in the PARAM entry TABS as follows:

PARAM, TABS, 273.16	(if Celsius is used)
PARAM, TABS, 459.69	(if Fahrenheit is used)

When the correction for the temperature effect is required, the temperature distribution must be defined in the Bulk Data entries (TEMP, TEMPP1, and/or TEMPRB) which are selected by the Case Control command, TEMP(LOAD) = SID, within the subcase.

From the thermodynamic consideration, the creep rate is expressed as

$$\dot{\epsilon}^c = A(e^{-\frac{\Delta H}{RT}})$$

where ΔH is the free energy of activation

R is the gas constant ($= 1.98 \text{ cal/mole } ^\circ\text{K}$)

T is the absolute temperature

A is strain/sec per activation

If the creep characteristics are defined at the temperature T_0 (RT field), the creep rate at the temperature T is corrected by a factor

$$\frac{\dot{\epsilon}^c}{\dot{\epsilon}_0^c} = (e^{-\frac{\Delta H}{RT_0}})^{(\frac{T_0}{T}-1)}$$

where $(e^{-\frac{\Delta H}{RT_0}})$ is to be specified in the EXP field.

2. If FORM = CRLAW, the creep law type and the coefficients (a-g) must be specified in the CREEP continuation fields. If FORM = TABLE, the creep model parameters (K_p , C_p , and C_s) are to be specified in the TABLES1 entries whose identification numbers appear in the TIDKP, TIDCP, and TIDCS fields, respectively. In this case, the CREEP continuation field should not be used.
3. The creep model parameters (K_p , C_p , and C_s) represent parameters in the uniaxial rheological model shown in Figure 6.1.3. Tabular entries (x_i, y_i) in the TABLES1 entry will consist of (σ_i, K_{pi}) , (σ_i, C_{pi}) , or (σ_i, C_{si}) for the input of K_p , C_p , and C_s , respectively. In the case of a linear viscoelastic material, the parameter (K_p , C_p , and C_s) values are constant and two data points with an identical value should be specified for each parameter.

The creep model parameters must have positive values as shown in Figures 6.1.4-6.1.6. If the table look-up results in a negative value, the value will be reset to zero and a warning message (TABLE LOOK-UP RESULTS IN NEGATIVE VALUE OF CREEP MODEL PARAMETER IN ELEMENT ID = ****) will be detected.

4. Two different classes of empirical creep laws are available at present. The first class of creep law is

$$\epsilon^c(\sigma, t) = A(\sigma)[1 - e^{-R(\sigma)t}] + K(\sigma)t$$

Oak Ridge National Laboratory recommends the parameters $A(\sigma)$, $R(\sigma)$, and $K(\sigma)$ in the form given in the following table:

Parameter	Type 1	Type 2
$A(\sigma)$	$a\sigma^b$	$a\text{Exp}(b\sigma)$
$R(\sigma)$	$c\text{Exp}(d\sigma)$	$c\sigma^d$
$K(\sigma)$	$e[\sinh(f\sigma)]^g$	$e\text{Exp}(f\sigma)$

Each of the three digits (e.g., 111, 121, etc.) in the TYPE field selects type 1 or 2 for the parameters $A(\sigma)$, $R(\sigma)$, and $K(\sigma)$, respectively.

The second class of creep law (TYPE = 300) is expressed as

$$\varepsilon^c(\sigma, t) = a\sigma^b t^d$$

where the values b and d are limited to the ranges

$$1.0 < b < 8.0$$

and

$$0.2 < d \leq 1.0$$

The coefficient g should be blank for creep law type xx2 material, and coefficients c , e , f , and g should be blank for creep law type 300. The coefficients (a-g) are dependent on the structural units. Caution must be exercised to make these units consistent with the rest of the input data.

5. Creep analysis requires an initial static solution at $t=0$, which can be obtained by a subcase with an NLPARM on which $DT=0$.

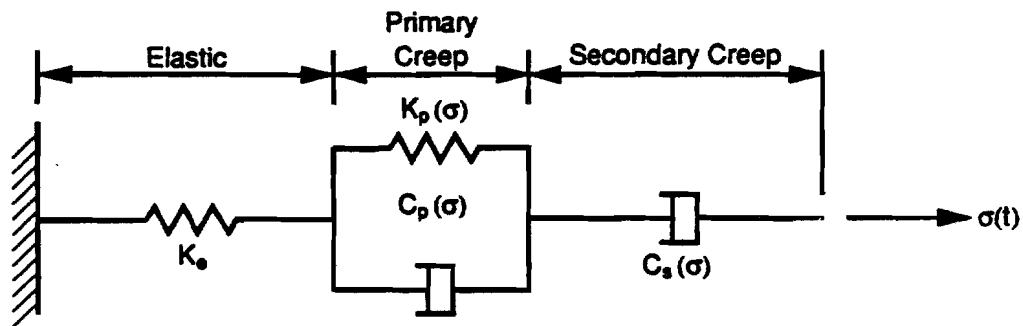


Figure 6.1.3. Uniaxial Rheological Model.

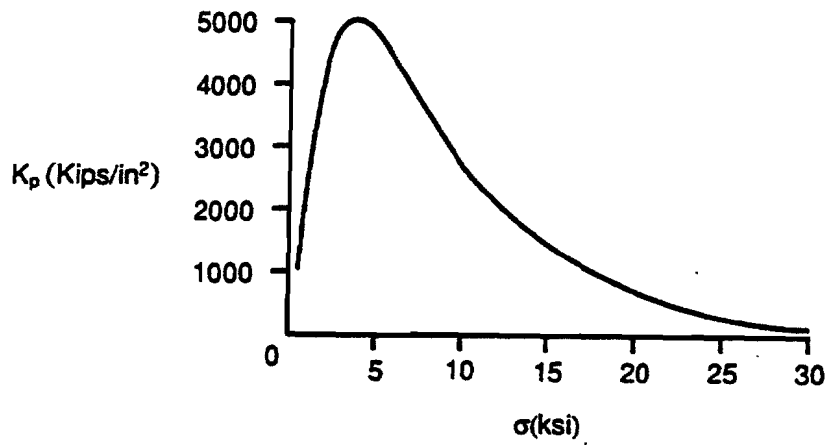


Figure 6.1.4. Creep Model Parameter K_p as a Function of Stress σ .

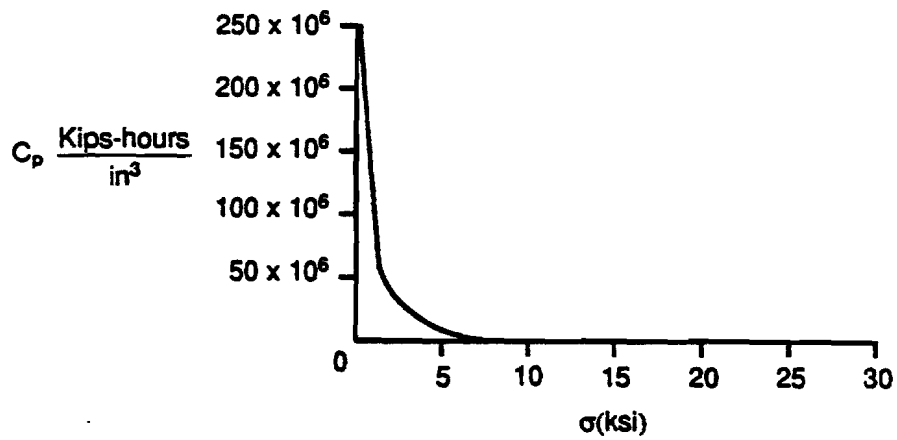


Figure 6.1.5. Creep Model Parameter C_p as a Function of Stress σ .

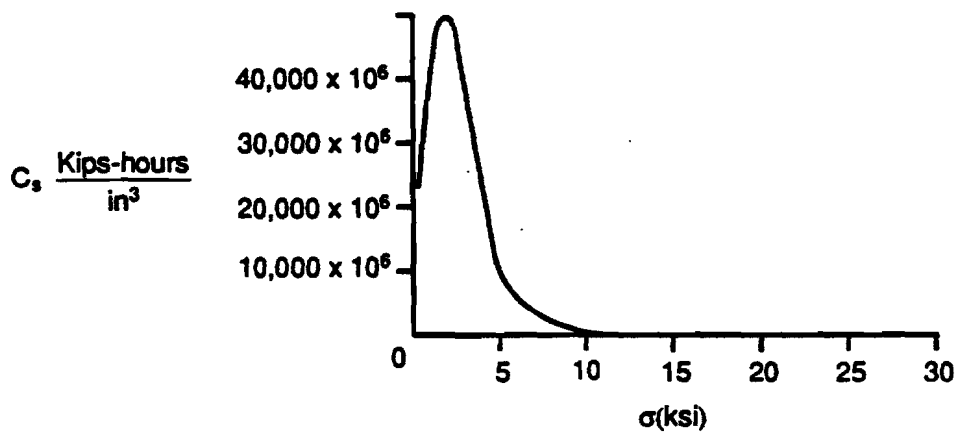


Figure 6.1.6. Creep Model Parameter C_s as a Function of Stress σ .

6.2 PLASTICITY

Two broad categories of material nonlinearity are nonlinear elastic analysis and plastic analysis. In both cases, the stress-strain relation is nonlinear; however, the unloading follows the stress-strain curve in the case of nonlinear elastic analysis, whereas elastic unloading takes place in plasticity.

The plasticity is characterized by its path dependency. The strain is not a function of the current state of stress alone, but depends on the previous loading or stress history. This is exemplified by the simple case of zero stress, when permanent sets of differing magnitude can be established by varying histories in which the stress starts and finishes at zero.

6.2.1 Some Preliminaries

Some of the commonly used terms in plasticity [6.1, 6.2] are given below.

Yield Stress: Yield stress is usually measured as the value of stress which produces the smallest measurable permanent strain.

Yield Criteria: In simple tension, there exists a yield point at which the material will begin to deform plastically. However, if the stress state at a point is not uniaxial but consists of stress components in different directions, a criterion is required to define which combination of multiaxial stresses will cause yielding. Such criteria are called yield criteria. The first step of any plasticity analysis is to decide a yield criterion.

Yield Surface: Any yield criterion is expressible in the form

$$F(\sigma, \kappa) = f(\sigma) - Y(\kappa) = 0$$

where $f(\sigma)$ is a function of the stress state and $Y(\kappa)$ is a function of the strain hardening. Geometric representation of the yield function is a surface in the stress space ($\sigma_1 - \sigma_2 - \sigma_3$ coordinates). This surface is called the yield surface.

Loading and Unloading: If the stress is changed so that the stress tends to move out of the elastic region, the process is called loading. If the stress value is changed such that the stress value moves into the elastic region, the process is called unloading.

Proportional Loading: Loading condition under which all the stress components increase in the same proportion.

Strain Hardening: When the material undergoes plastic deformation or change in strain, the yield surface expands and/or translates with increasing strain. This is known as strain hardening or work hardening and the material which does not exhibit strain hardening is called perfectly-plastic material.

Bauschinger Effect: When a plastically deformed specimen is unloaded, residual stresses on a microscopic scale remain and influence the plastic yielding for the different loading. If the previous strain was a uniform extension and the specimen is then reloaded in compression in the opposite direction, it is observed that yielding occurs at a reduced stress. This is known as the Bauschinger effect.

The phenomenon of the Bauschinger effect can be described with reference to Fig. 6.2.1, which shows an idealized stress-strain curve of a ductile material: first deformed by uniform tension, the load removed, and reloaded in compression. According to the kinematic hardening model (which is one extreme viewpoint), it is assumed that the elastic unloading range will be double the initial yield stress. If the initial yield stress in tension is σ_y and the specimen is loaded up to a stress σ_1 and unloaded, the plastic yielding will begin in compression at stress σ_2 given by $\sigma_2 = \sigma_1 - 2\sigma_y$. At the other extreme is the model that shows the hardening mechanism acting equally in tension and compression. Thus compressive yielding will occur when $\sigma_2 = -\sigma_1$. This model shows no Bauschinger effect. This is the simplest of the theories to apply and is most frequently used. This behavior is called the isotropic hardening rule. Between these theories there are models that compromise the attributes of the kinematic and isotropic hardening for the onset of compressive yield after tensile hardening. This behavior can be modeled using the combined hardening option.

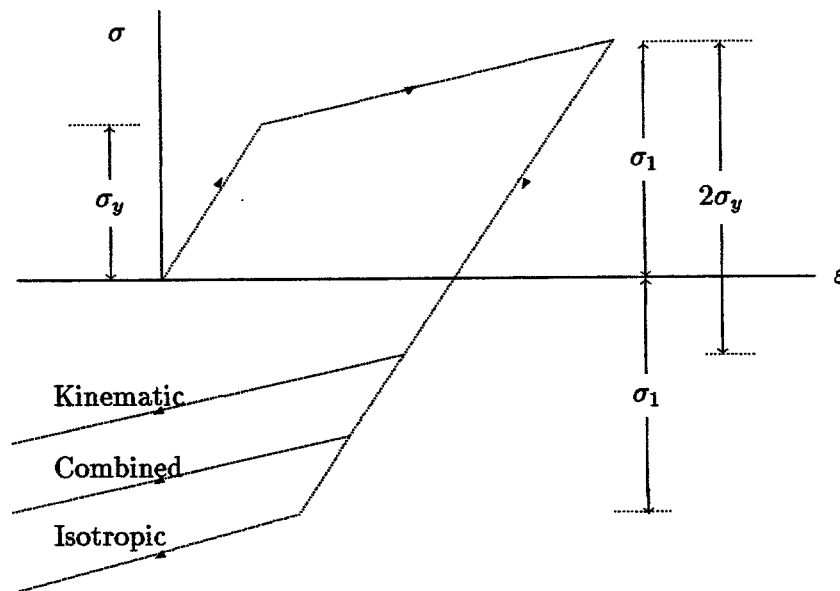


Figure 6.2.1. Representation of Hardening Rules

6.2.2 Yield Criteria

The software provides four different yield criteria [6.3 - 6.6]: von Mises, Tresca, Mohr-Coulomb and Drucker-Prager. The von Mises criterion is most commonly used in plastic analysis of ductile materials. The Tresca criterion is more suitable for brittle materials. The criteria by Mohr-Coulomb and Drucker-Prager are suitable for frictional materials such as soil and concrete.

The **von Mises yield criterion** states that yielding begins when the elastic distortion energy (or second invariant of deviatoric stress, J'_2) exceeds a critical value. In simple tension, this critical value is related to the yield stress (Y), i.e.,

$$\sqrt{3}(J'_2)^{1/2} \geq Y \quad (6.2.1)$$

where

$$J'_2 = \frac{1}{2}\sigma'_{ij}\sigma'_{ij} = \frac{1}{6}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]. \quad (6.2.2)$$

Geometrically, this equation represents a right circular cylinder perpendicular to the octahedral (π) plane. This cylinder is the von Mises yield surface, as shown in Fig. 6.2.2. The left-hand side of Eq. (6.2.1) may be expressed in terms of the general stress state, i.e.,

$$\bar{\sigma}^2 = \frac{1}{2}[(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2] + 3(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2) \quad (6.2.3)$$

where $\bar{\sigma}$ is termed the effective stress. For pure shear of the plane stress case,

$$\sigma_1 = -\sigma_2 = \tau \quad ; \quad \sigma_3 = 0 \quad (6.2.4)$$

and, from Eqs. (6.2.1) and (6.2.3),

$$\tau = \frac{Y}{\sqrt{3}} \quad (6.2.5)$$

where Y is the yield stress of the material, i.e., yield stress in pure shear is $\frac{1}{\sqrt{3}}$ times the yield stress in simple tension.

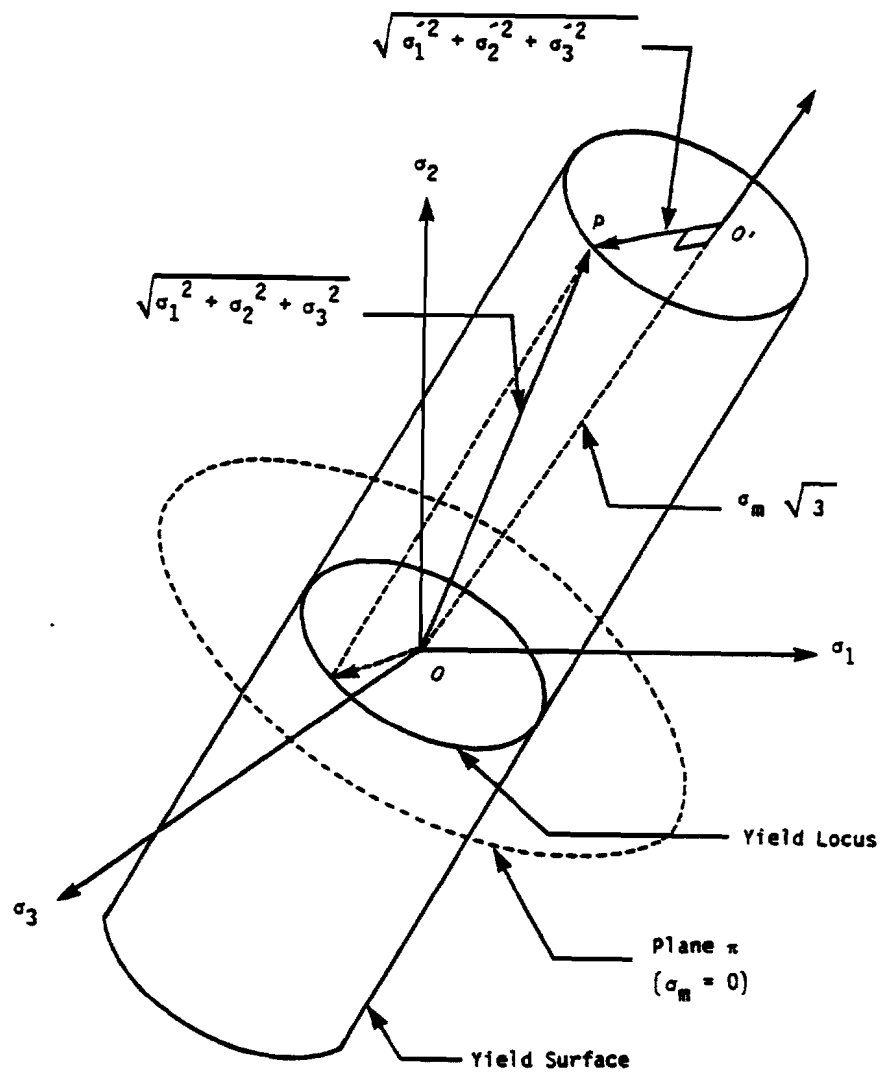


Figure 6.2.2. Geometric Representation of von Mises Yield Surface.

The **Tresca yield criterion** (sometimes referred to as the maximum shear theory) assumes that yielding will occur when the maximum shear stress exceeds the value of the maximum shear stress occurring under simple tension, i.e.,

$$\tau_{max} = \frac{\sigma_1 - \sigma_3}{2} \geq \frac{1}{2}Y \quad (6.2.6)$$

where σ_1 and σ_3 are principal stresses such that $\sigma_3 < \sigma_2 < \sigma_1$. Geometric representation of this equation is a hexagonal surface perpendicular to the π plane ($\sigma_1 + \sigma_2 + \sigma_3 = 0$) in the stress space.

For frictional materials, the cohesion c and the internal friction angle ϕ are the governing properties. The yield criterion for such materials is given by **Mohr-Coulomb** as

$$\tau = c - \sigma_n \tan \phi \quad (6.2.7)$$

where τ is the magnitude of the shear stress,
 σ_n is the normal stress,
 c is the cohesion, and
 ϕ the angle of internal friction. ($0 \leq \phi < 45^\circ$)

Graphically Eq. (6.2.7) represents a straight line tangent to the largest Mohr stress circle as shown in Fig. 6.2.3. In terms of principal stresses ($0 > \sigma_1 \geq \sigma_2 \geq \sigma_3$), Eq. (6.2.7) becomes

$$\frac{1}{2}(\sigma_1 - \sigma_3) \cos \phi \geq c - \left(\frac{\sigma_1 + \sigma_3}{2} + \frac{\sigma_1 - \sigma_3}{2} \sin \phi \right) \tan \phi \quad (6.2.8)$$

which may be reduced to

$$\sigma_1 - \sigma_3 + (\sigma_1 + \sigma_3) \sin \phi = 2c \cos \phi \quad (6.2.9)$$

This may be reduced to Tresca's criterion with $\phi = 0$ and $Y = 2c$.

An approximation to the Mohr-Coulomb yield criterion was presented by Drucker and Prager, known as the **Drucker-Prager yield criterion**, which is a modification of the von Mises yield criterion, i.e.,

$$3\alpha\sigma_m + (J'_2)^{1/2} = \kappa \quad (6.2.10)$$

where

$$\alpha = \frac{2 \sin \phi}{\sqrt{3}(3 - \sin \phi)}$$

$$\kappa = \frac{6c \cos \phi}{\sqrt{3}(3 - \sin \phi)}$$

This yield surface has the form of a circular cone coinciding with the outer apices of the Mohr-Coulomb hexagonal cone at any section. Rearranging terms of Eq. (6.2.10), the Drucker-Prager yield function is given as

$$\frac{6 \sin \phi}{3 - \sin \phi} \sigma_m + \sqrt{3}(J'_2)^{1/2} \geq \frac{6c \cos \phi}{3 - \sin \phi} \quad (6.2.11)$$

which is reduced to von Mises criterion for $\phi = 0$.

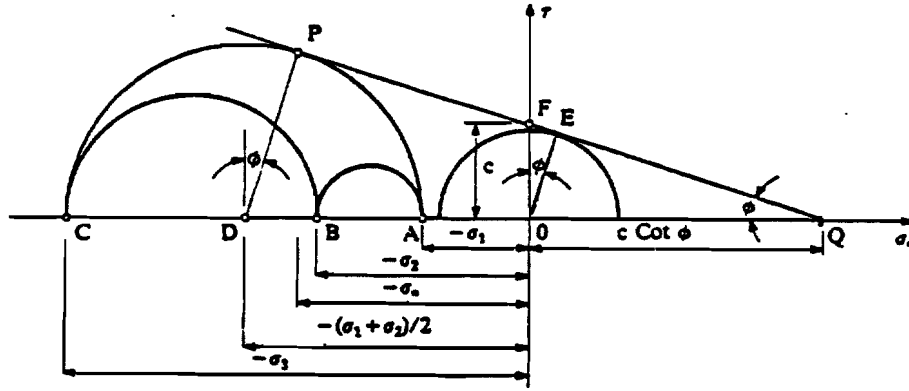


Figure 6.2.3. Mohr Circle Representation of the Mohr-Coulomb Yield Criterion.

6.2.3 Yield Function

Associated with the yield criteria, there exist yield functions in the form of

$$F(\sigma, \bar{\epsilon}^p) = f(\sigma) - Y(\bar{\epsilon}^p) = 0$$

where $f(\sigma)$ is an effective stress, a function of the stress state
and $Y(\bar{\epsilon}^p)$ is a tensile yield stress, a function of the strain hardening.

Geometric representation of the yield function is a surface in the stress space ($\sigma_1 - \sigma_2 - \sigma_3$ coordinates), which is called a yield surface. The von Mises yield surface is a circular cylinder and Tresca's is a hexagonal cylinder. The yield surface is reduced to the yield locus in the case of plane stress, as shown in Fig. 6.2.4 and 6.2.5 for von Mises and Tresca criteria, respectively. Both the Mohr-Coulomb and Drucker-Prager yield functions represent conical surfaces as shown in Fig. 6.2.6.

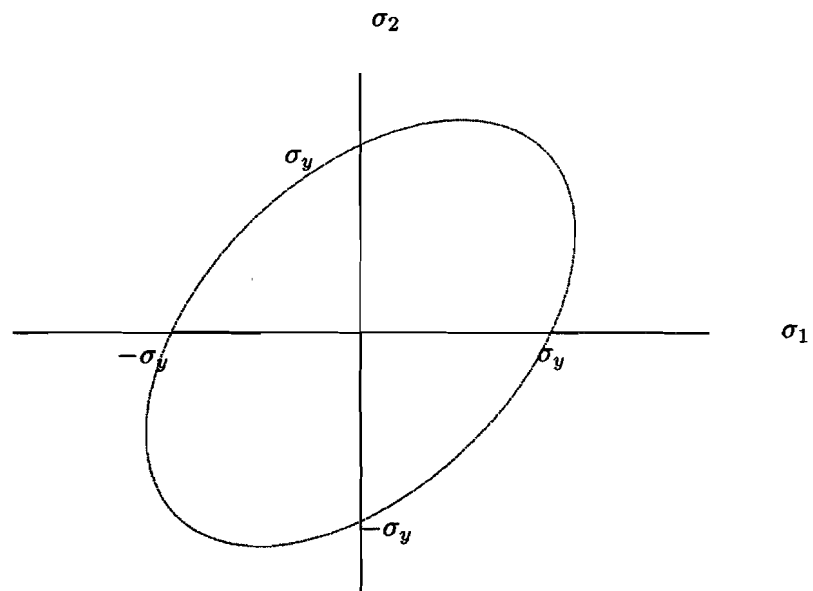


Figure 6.2.4 Von Mises Yield Locus for Plane Stress.

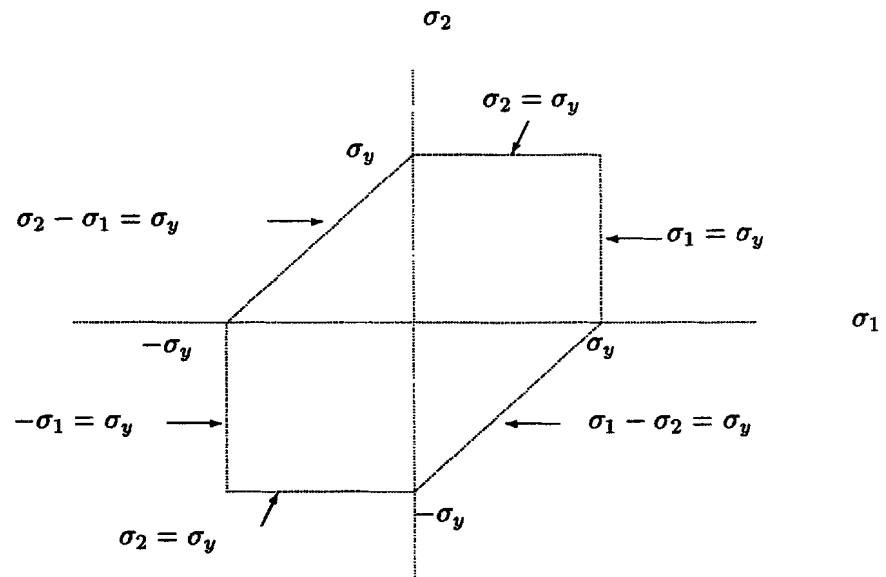


Figure 6.2.5 Tresca's Yield Locus for Plane Stress.

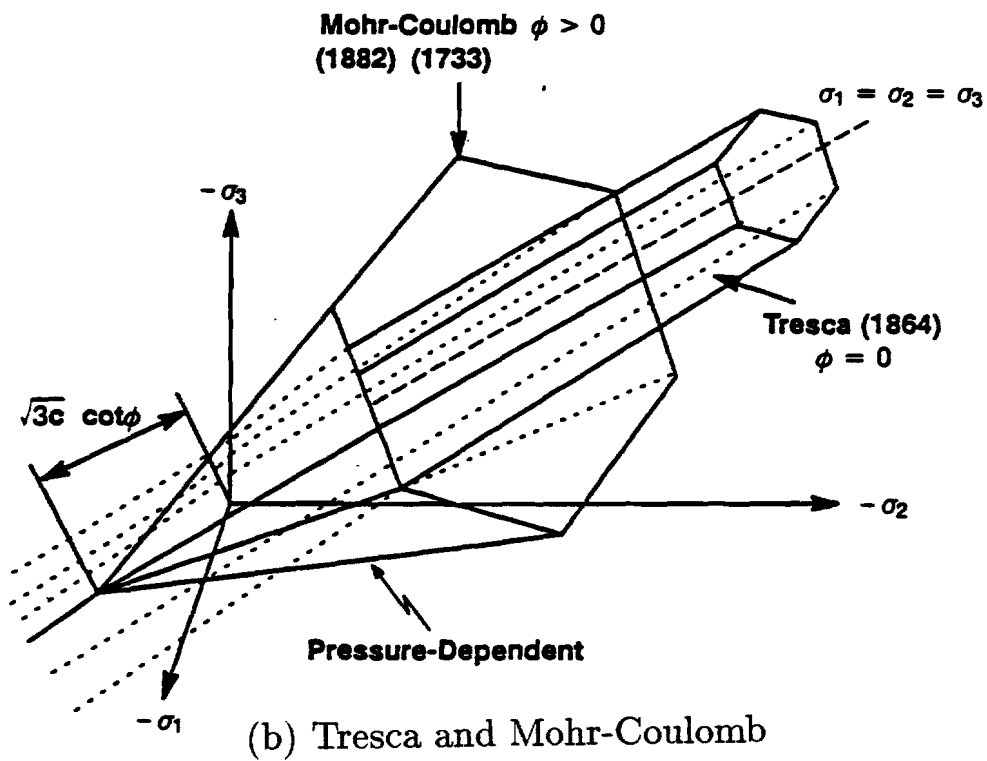
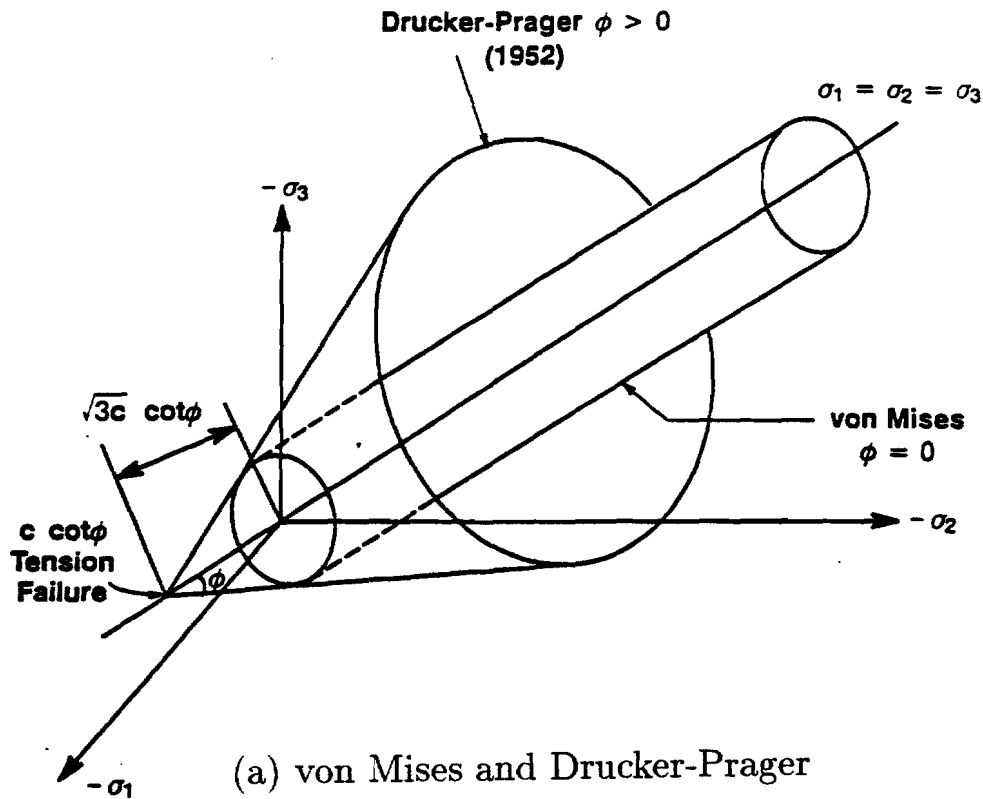


Figure 6.2.6. Isotropic Yield Surfaces in Principal Stress Space

The stress function $f(\sigma)$ represents the effective stress or equivalent stress, which is equivalent to the uniaxial stress in tension. The value of $f(\sigma)$ can be computed as follows:

i) For von Mises criterion

$$f(\sigma) = \sqrt{3}(J'_2)^{1/2} \quad (6.2.12)$$

ii) For Tresca's criterion

$$f(\sigma) = \sigma_1 - \sigma_3 \quad (6.2.13)$$

iii) For Mohr-Coulomb criterion

$$f(\sigma) = \sigma_1 - \sigma_3 + (\sigma_1 + \sigma_3) \sin \phi \quad (6.2.14)$$

iv) For Drucker-Prager criterion

$$f(\sigma) = \frac{6 \sin \phi}{3 - \sin \phi} \sigma_m + \sqrt{3}(J'_2)^{1/2} \quad (6.2.15)$$

The function $Y(\bar{\varepsilon}^p)$ is a measure of the tensile yield stress. The value of $Y(\bar{\varepsilon}^p)$ can be found from the user-specified stress-strain curve by an interpolation, i.e.,

$$Y(\bar{\varepsilon}^p) = Y_1 + \beta \left[\sum_{i=1}^{k-1} H_i(\bar{\varepsilon}_{i+1}^p - \bar{\varepsilon}_i^p) + \beta H_k(\bar{\varepsilon}^p - \bar{\varepsilon}_k^p) \right] \quad (6.2.16)$$

with

$$\begin{aligned} \bar{\varepsilon}_k^p &= \varepsilon_k - \frac{Y_k}{E} \\ H_k &= \frac{Y_{k+1} - Y_k}{\bar{\varepsilon}_{k+1}^p - \bar{\varepsilon}_k^p} = \frac{Y_{k+1} - Y_k}{(\varepsilon_{k+1} - \varepsilon_k) - \frac{1}{E}(Y_{k+1} - Y_k)} \end{aligned} \quad (6.2.17)$$

where (ε_k, Y_k) is a data point such that $\bar{\varepsilon}_k^p \leq \bar{\varepsilon}^p < \bar{\varepsilon}_{k+1}^p$, Y_1 is the initial yield point, and $\beta = 0.$, 0.5 or $1.$ (for kinematic, combined, or isotropic hardening, respectively). Since $Y_1 = 2c$ for frictional materials, it is obvious from Eqs. (6.2.9) and (6.2.11) that Eq. (6.2.16) must be modified by multiplying correction factors as follows:

$$\bar{Y}(\bar{\varepsilon}^p) = \cos \phi * Y(\bar{\varepsilon}^p) \quad \text{for Mohr-Coulomb} \quad (6.2.18)$$

$$\bar{Y}(\bar{\varepsilon}^p) = \frac{3 \cos \phi}{3 - \sin \phi} * Y(\bar{\varepsilon}^p) \quad \text{for Drucker-Prager} \quad (6.2.19)$$

Notice that $Y > 0.0$, otherwise the program issues a user error message: "Implausible stress-strain curve in EID=xxx".

6.2.4 Strain Hardening

The yield surfaces discussed previously correspond to the yield surface of a material when yielding starts. For a material that strain hardens, the yield surface must change for continued straining. Such a change of the yield surface is governed by the hardening rules.

The **isotropic hardening** theory, due to Hill, assumes that the loading surface expands uniformly about the origin in the stress space during plastic flow, maintaining the same shape, center and orientation as the original yield surface. Fig. 6.2.7 shows the initial and subsequent yield surfaces on a two-dimensional plot when the stress state shifts from point 1 to point 2. Unloading and subsequent reloading in the reverse direction will result in yielding at the stress state represented by point 3. The path 2-3 will be elastic and 0-2 is equal to 0-3. The main drawback to this theory is its inability to account for the Bauschinger effect exhibited by most materials. The equation for subsequent yield surfaces is given by

$$F = f(\sigma) - Y(\bar{\epsilon}^p) = 0 \quad (6.2.20)$$

where $Y(\bar{\epsilon}^p)$ is a measure of the expansion of the yield surface in the stress space, and the total effective plastic strain ($\bar{\epsilon}^p$) is a monotonically increasing scalar function during plastic deformation.

The **kinematic hardening** theory, due to Prager, assumes that the loading surface translates as a rigid body in stress space during plastic deformation, maintaining the size, shape and orientation of the yield surface (see Fig. 6.2.8). Thus, this theory accounts for the Bauschinger effect. The yield function for kinematic hardening is expressed as

$$F = f(\sigma_{ij} - \alpha_{ij}) - Y = 0 \quad (6.2.21)$$

where α_{ij} represents the translation of the center of the yield surface, and Y is assumed constant. The theory as set forth by Prager postulates that the incremental translation of the loading surface in the stress space occurs in the direction of the exterior normal to the surface at the instantaneous stress state. This relation is expressed analytically by

$$d\alpha_{ij} = c \, d\epsilon_{ij}^p \quad (6.2.22)$$

where c is a scalar parameter called the hardening modulus, and $d\epsilon_{ij}^p$ is the increment of plastic strain.

However, inconsistencies arise when Prager's theory is applied in various subspaces of stress. In other words, the loading surface will not, in general, translate in the direction of the exterior normal in the stress subspace with the full nine dimensional stress space. To avoid this difficulty, Ziegler has proposed a modification of Prager's rule as

$$d\alpha_{ij} = d\mu(\sigma_{ij} - \alpha_{ij}), \quad d\mu \geq 0 \quad (6.2.23)$$

The scalar $d\mu$ in Eq. (6.2.23) is determined from the condition that the stress state must remain on the translated loading surface during plastic deformation. Substituting Eq. (6.2.23) into

the total differential of Eq. (6.2.21),

$$dF = \frac{\partial F}{\partial \sigma} d\sigma + \frac{\partial F}{\partial \alpha} d\alpha = \frac{\partial f(\sigma - \alpha)}{\partial \sigma} d\sigma - \frac{\partial f(\sigma - \alpha)}{\partial \alpha} d\alpha = 0$$

and solving for $d\mu$,

$$d\mu = \frac{\frac{\partial f}{\partial \sigma_{ij}} d\sigma_{ij}}{(\sigma_{ij} - \alpha_{ij}) \frac{\partial f}{\partial \alpha_{ij}}} \quad (6.2.24)$$

A theory that combines kinematic and isotropic work hardening states that

$$F = f(\sigma_{ij} - \alpha_{ij}) - Y(\bar{\epsilon}^p) = 0 \quad (6.2.25)$$

for which $d\mu$ is found to be

$$d\mu = \frac{\frac{\partial f(\sigma - \alpha)}{\partial \sigma} d\sigma - \beta H d\bar{\epsilon}^p}{(\sigma - \alpha) \frac{\partial f(\sigma - \alpha)}{\partial \alpha}} \quad (6.2.26)$$

where H is the instantaneous slope ($\frac{dY}{d\bar{\epsilon}^p}$) of the uniaxial stress-strain curve and $\beta = 0.5$. Eq. (6.2.25) reduces to kinematic hardening when $Y(\bar{\epsilon}^p)$ is a constant and to isotropic hardening when $\alpha_{ij} = 0$ and $Y(\bar{\epsilon}^p)$ is monotonically increasing.

The specification of the initial yield surface (by means of a choice of the yield criterion) defines the stress state at which plastic flow will begin. The specification of a hardening rule defines the stress state for subsequent yielding.

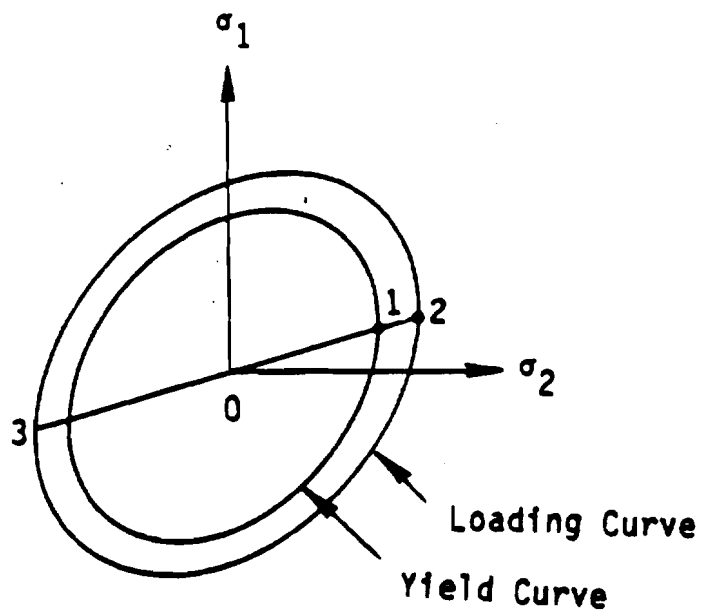


Figure 6.2.7. Isotropic Hardening.

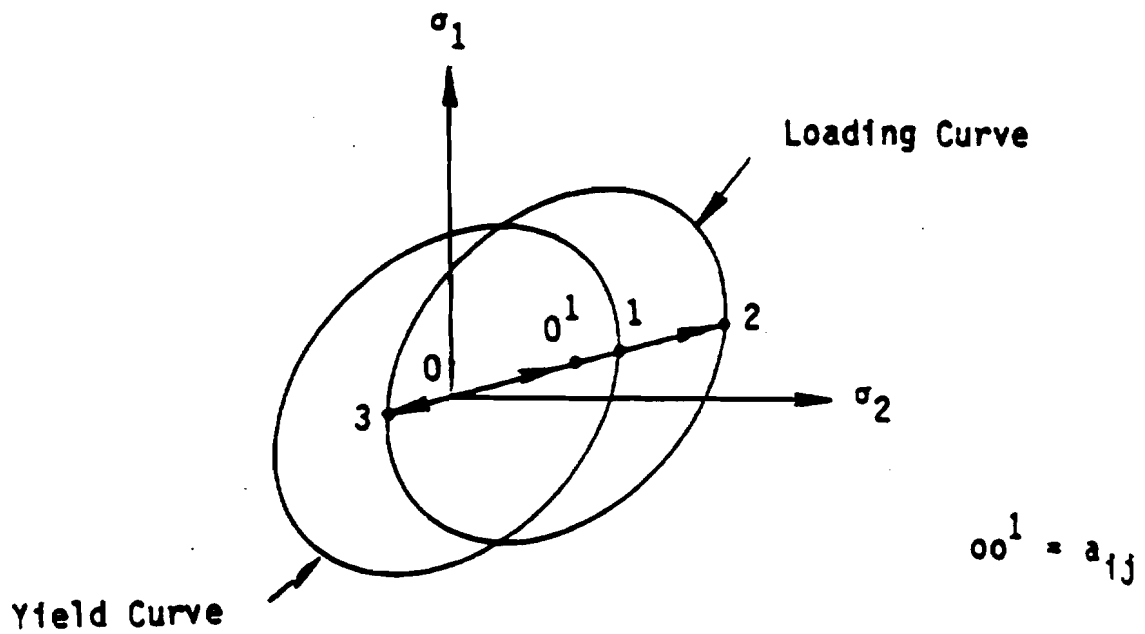


Figure 6.2.8. Kinematic Hardening.

6.2.5 Prandtl-Reuss Stress-Strain Relations

One of the major differences between stress-strain relations in the elastic range and those in the plastic range is that strains in the plastic range are, in general, not uniquely determined by the stresses but are dependent on the past history as well. Due to the path-dependent nature of the plastic strains, increments of plastic strain are calculated by an incremental procedure and the total strains obtained by integration or summation.

After Saint Venant proposed that the principal axes of the strain increment coincided with the axes of principal stress, a general relationship between components of the strain increment and stress was established by Levy and von Mises, independently [6.1]. Levy-Mises equations, however, use the total strain increments and are applicable only to rigid-plastic materials. Levy-Mises equations were modified to allow applications for elasto-plastic materials by Prandtl and Reuss. These equations are known as the Prandtl-Reuss stress-strain relations, i.e.,

$$\begin{aligned}
 d\varepsilon_x^p &= \frac{3}{2} \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} (\sigma_x - \sigma_m) = \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} \left[\sigma_x - \frac{1}{2}(\sigma_y + \sigma_z) \right] \\
 d\varepsilon_y^p &= \frac{3}{2} \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} (\sigma_y - \sigma_m) = \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} \left[\sigma_y - \frac{1}{2}(\sigma_z + \sigma_x) \right] \\
 d\varepsilon_z^p &= \frac{3}{2} \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} (\sigma_z - \sigma_m) = \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} \left[\sigma_z - \frac{1}{2}(\sigma_x + \sigma_y) \right] \\
 d\gamma_{xy}^p &= 3 \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} \tau_{xy} \\
 d\gamma_{yz}^p &= 3 \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} \tau_{yz} \\
 d\gamma_{zx}^p &= 3 \frac{d\bar{\varepsilon}^p}{\bar{\sigma}} \tau_{zx}
 \end{aligned} \tag{6.2.27}$$

where

$$\bar{\sigma} = \sqrt{\frac{1}{2} [(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2] + 3(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2)}$$

and

$$d\bar{\varepsilon}^p = \frac{2}{3} \sqrt{\frac{1}{2} [(d\varepsilon_x^p - d\varepsilon_y^p)^2 + (d\varepsilon_y^p - d\varepsilon_z^p)^2 + (d\varepsilon_z^p - d\varepsilon_x^p)^2] + \frac{3}{4} (d\gamma_{xy}^{p2} + d\gamma_{yz}^{p2} + d\gamma_{zx}^{p2})} \tag{6.2.28}$$

These stress-strain relations are consistent with the von Mises yield criterion. For other yield criteria, a general derivation follows.

6.2.6 Associated Flow Rule

Based on the condition that the principal axes of the plastic strain-increment must coincide with the principal stress axes, and defining a function of stress components (σ_{ij}) as the plastic

potential function (Q), the plastic strain-increments can be expressed as

$$d\varepsilon_{ij}^p = d\lambda \frac{\partial Q}{\partial \sigma_{ij}} \quad (6.2.29)$$

where $d\lambda$ is a Lagrange multiplier. For a so-called stable plastic material which has a stationary value of the plastic work done in a given increment, such a function (Q) exists and is assumed identical to the yield function (F), i.e.,

$$d\varepsilon_{ij}^p = d\lambda \frac{\partial F}{\partial \sigma_{ij}} \quad (6.2.30)$$

where F is given by Eq. (6.2.25). This is called the associated flow rule. The geometric implication of this relationship is that the plastic strain-increment vector is normal to the yield surface, known as normality principle.

During the incremental loading, the incremental strain is assumed to be equal to the sum of the incremental elastic and incremental plastic strains, i.e.,

$$\{d\varepsilon\} = \{d\varepsilon\}^e + \{d\varepsilon\}^p \quad (6.2.31)$$

But

$$\{d\varepsilon\}^e = [D_e]^{-1} \{d\sigma\} \quad (6.2.32)$$

with $[D_e]$ being the elasticity matrix;

$$[D_e] = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 0 & 0 & 0 \\ & 1 & \frac{\nu}{1-\nu} & 0 & 0 & 0 \\ & & 1 & 0 & 0 & 0 \\ & & & \frac{1-2\nu}{2(1-\nu)} & 0 & 0 \\ \text{Sym.} & & & & \frac{1-2\nu}{2(1-\nu)} & 0 \\ & & & & & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix}$$

and upon substituting Eqs. (6.2.30) and (6.2.32) into (6.2.31), it is found that

$$\{d\varepsilon\} = [D_e]^{-1} \{d\sigma\} + d\lambda \left\{ \frac{\partial F}{\partial \sigma} \right\} \quad (6.2.33)$$

When plastic yield is in progress, the stresses are on the yield surface given by the yield function, which, when differentiated, provides the expression

$$dF = \left\{ \frac{\partial F}{\partial \sigma} \right\}^T \{d\sigma\} + \frac{\partial F}{\partial \kappa} d\kappa = 0 \quad (6.2.34)$$

where κ (work-hardening parameter) can be α_{ij} or $\bar{\varepsilon}^p$, whichever case it may be. Then Eq. (6.2.34) may be conveniently expressed as

$$\left\{ \frac{\partial F}{\partial \sigma} \right\}^T \{d\sigma\} - H^* d\lambda = 0 \quad (6.2.35)$$

with

$$H^* = -\frac{\partial F}{\partial \kappa} d\kappa \frac{1}{d\lambda} \quad (6.2.36)$$

where H^* has a physical implication which will be shown later.

After a few matrix manipulations to eliminate $\{d\sigma\}$ from Eqs. (6.2.33) and (6.2.35), the scalar multiplier $d\lambda$ can be obtained as

$$d\lambda = \frac{\{\frac{\partial f}{\partial \sigma}\}^T [D_e] \{d\varepsilon\}}{H^* + \{\frac{\partial f}{\partial \sigma}\}^T [D_e] \{\frac{\partial f}{\partial \sigma}\}} \quad (6.2.37)$$

where $\{\frac{\partial F}{\partial \sigma}\}$ is reduced to $\{\frac{\partial f}{\partial \sigma}\}$, and the terms H^* and $\{\frac{\partial f}{\partial \sigma}\}$ are functions of $\{\sigma\}$. Elimination of $d\lambda$ from Eq. (6.2.33) gives

$$\{d\sigma\} = [D_{ep}] \{d\varepsilon\} \quad (6.2.38)$$

with the elasto-plastic matrix defined as [6.5]

$$[D_{ep}] = [D_e] - [D_e] \{\frac{\partial f}{\partial \sigma}\} \{\frac{\partial f}{\partial \sigma}\}^T [D_e] [H^* + \{\frac{\partial f}{\partial \sigma}\}^T [D_e] \{\frac{\partial f}{\partial \sigma}\}]^{-1} \quad (6.2.39)$$

6.2.7 Generalized Effective Plastic Strain Increment

The effective plastic strain increment can be defined using the work-hardening hypothesis. Work-hardening is postulated as the amount of work done during plastic deformation, i.e.,

$$dW_p = \{\sigma\}^T \{d\varepsilon^p\} = d\lambda \{\sigma\}^T \{\frac{\partial f}{\partial \sigma}\}. \quad (6.2.40)$$

According to the Euler's theorem, if the function $f(\sigma_{ij})$ is homogeneous and of degree one, e.g.,

$$f(\lambda \sigma_{ij}) = \lambda f(\sigma_{ij}) \quad (6.2.41)$$

then the following equality exists:

$$\{\sigma\}^T \{\frac{\partial f}{\partial \sigma}\} = f(\sigma_{ij}) \quad (6.2.42)$$

All four yield criteria, provided in MSC/NASTRAN, satisfy Eq. (6.2.41).

By virtue of Euler's theorem, Eq. (6.2.40) may be restated as

$$dW_p = d\lambda f(\sigma_{ij}). \quad (6.2.43)$$

The stress-strain data from the uniaxial tension test (in x-direction) may be related to Eq. (6.2.43) by considering the work done by the plastic deformation during the test, i.e.,

$$dW_p = \sigma_x d\varepsilon_x^p = \bar{\sigma} d\bar{\varepsilon}^p \quad (6.2.44)$$

When the effective stress ($\bar{\sigma}$) corresponds to the stress under uniaxial tension (Y), the corresponding effective strain increment must be equivalent to $d\lambda$ by equating corresponding terms of Eqs. (6.2.43) and (6.2.44). Hence,

$$d\bar{\epsilon}^p = d\lambda \geq 0 \quad (6.2.45)$$

and $d\lambda$ is defined by Eq. (6.2.37). In fact, it can be shown that $d\lambda$ in Eq. (6.2.37) is identical to $d\bar{\epsilon}^p$ as defined in Eq. (6.2.28) with the von Mises yield function. It is necessary to clarify H^* in Eq. (6.2.36) using Eq. (6.2.45). Combining Eqs. (6.2.30), (6.2.35) and (6.2.45), Eq. (6.2.36) can be restated as

$$H^* = \left(\frac{1}{d\bar{\epsilon}^p}\right)^2 \{d\sigma\}^T \{d\epsilon^p\} = \frac{d\bar{\sigma}}{d\bar{\epsilon}^p} \quad (6.2.36a)$$

It is revealed that H^* is identical to H which is the slope of the uniaxial stress-strain curve, $\frac{dY}{d\bar{\epsilon}^p}$. This is a natural consequence of the work-hardening, which should follow the simple tension test data regardless of the hardening rule.

6.2.8 Yield Function Derivatives

The derivatives of the yield function with respect to stress components are frequently required in the computation of plastic strains. This frequent computation is conveniently simplified by adopting a unified approach to the various yield criteria. All the yield functions are expressed in terms of stress invariants [6.6], and the gradient vector is obtained by differentiating the yield function with respect to these invariants first. This unified approach also facilitates to avoid the difficulty of singularities in the Tresca and the Mohr-Coulomb yield criteria.

The principal stresses can be expressed in terms of stress invariants as follows:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \end{Bmatrix} = \frac{2(J_2')^{1/2}}{\sqrt{3}} \begin{Bmatrix} \sin(\theta + \frac{2}{3}\pi) \\ \sin \theta \\ \sin(\theta + \frac{4}{3}\pi) \end{Bmatrix} + \sigma_m \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \quad (6.2.46)$$

where

$$\sigma_m = \frac{1}{3}(\sigma_x + \sigma_y + \sigma_z)$$

and

$$\theta = \frac{1}{3} \sin^{-1} \left[-\frac{3\sqrt{3}}{2} \frac{J_3'}{(J_2')^{3/2}} \right], \quad -\frac{\pi}{6} < \theta < \frac{\pi}{6} \quad (6.2.47)$$

with

$$\begin{aligned} J_2' &= \frac{1}{2}(\sigma_x'^2 + \sigma_y'^2 + \sigma_z'^2) + \tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2 \\ &= -(\sigma_x'\sigma_y' + \sigma_y'\sigma_z' + \sigma_z'\sigma_x') + \tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2 \end{aligned}$$

and

$$J'_3 = \sigma'_x \sigma'_y \sigma'_z + 2\tau_{xy} \tau_{yz} \tau_{zx} - (\sigma'_x \tau_{yz}^2 + \sigma'_y \tau_{zx}^2 + \sigma'_z \tau_{xy}^2)$$

Substitution of this expression into Eqs. (6.2.13) and (6.2.14) results in the Tresca and Mohr-Coulomb yield criteria in terms of stress invariants:

$$f(\sigma) = 2(J'_2)^{1/2} \cos \theta \quad (6.2.13a)$$

and

$$f(\sigma) = 2\sigma_m \sin \phi + 2(J'_2)^{1/2} [\cos \theta - \frac{1}{\sqrt{3}} \sin \theta \sin \phi] \quad (6.2.14a)$$

The gradient vector is computed by differentiating the stress function $f(\sigma)$ representing effective stress for each yield criterion as follows:

$$\left\{ \frac{\partial F}{\partial \sigma} \right\} = \left\{ \frac{\partial f}{\partial \sigma} \right\} = \frac{\partial f}{\partial \sigma_m} \left\{ \frac{\partial \sigma_m}{\partial \sigma} \right\} + \frac{\partial f}{\partial J'_2} \left\{ \frac{\partial J'_2}{\partial \sigma} \right\} + \frac{\partial f}{\partial J'_3} \left\{ \frac{\partial J'_3}{\partial \sigma} \right\} \quad (6.2.48)$$

with

$$\frac{\partial f}{\partial J'_3} = \frac{\partial f}{\partial \theta} \frac{\partial \theta}{\partial J'_3} \quad \text{and} \quad \frac{\partial f}{\partial J'_2} = \frac{\partial f}{\partial J'_2} + \frac{\partial f}{\partial \theta} \frac{\partial \theta}{\partial J'_2}.$$

The gradient vectors of the stress invariants are found independent of yield criteria as follows:

$$\begin{aligned} \left\{ \frac{\partial \sigma_m}{\partial \sigma} \right\}^T &= \frac{1}{3} < 1 \ 1 \ 1 \ 0 \ 0 \ 0 > \\ \left\{ \frac{\partial J'_2}{\partial \sigma} \right\}^T &= < \sigma'_x \ \sigma'_y \ \sigma'_z \ 2\tau_{xy} \ 2\tau_{yz} \ 2\tau_{zx} > \\ \left\{ \frac{\partial J'_3}{\partial \sigma} \right\}^T &= < (\sigma'_y \sigma'_z - \tau_{yz}^2 + \frac{J'_2}{3}), (\sigma'_x \sigma'_z - \tau_{zx}^2 + \frac{J'_2}{3}), \\ &\quad (\sigma'_x \sigma'_y - \tau_{xy}^2 + \frac{J'_2}{3}), 2(\tau_{yz} \tau_{zx} - \sigma'_z \tau_{xy}), \\ &\quad 2(\tau_{xy} \tau_{zx} - \sigma'_x \tau_{yz}), 2(\tau_{xy} \tau_{yz} - \sigma'_y \tau_{zx}) > \end{aligned}$$

The derivatives of the effective stress with respect to the stress invariants are found as below:

i) Tresca's criterion:

$$\begin{aligned} \frac{\partial f}{\partial \sigma_m} &= 0 \\ \frac{\partial f}{\partial J'_2} &= \frac{\cos \theta}{\sqrt{J'_2}} (1 + \tan \theta \tan 3\theta) \\ \frac{\partial f}{\partial J'_3} &= \frac{\sqrt{3} \sin \theta}{J'_2 \cos 3\theta} \end{aligned}$$

ii) von Mises' criterion:

$$\begin{aligned}\frac{\partial f}{\partial \sigma_m} &= \frac{\partial f}{\partial J'_3} = 0 \\ \frac{\partial f}{\partial J'_2} &= \frac{\sqrt{3}}{2\sqrt{J'_2}}\end{aligned}$$

iii) Mohr-Coulomb criterion:

$$\begin{aligned}\frac{\partial f}{\partial \sigma_m} &= 2 \sin \phi \\ \frac{\partial f}{\partial J'_2} &= \frac{\cos \theta}{\sqrt{J'_2}} [(1 + \tan \theta \tan 3\theta) + \frac{1}{\sqrt{3}} \sin \phi (\tan 3\theta - \tan \theta)] \\ \frac{\partial f}{\partial J'_3} &= \frac{\sqrt{3} \sin \theta + \sin \phi \cos \theta}{J'_2 \cos 3\theta}\end{aligned}$$

iv) Drucker-Prager criterion:

$$\begin{aligned}\frac{\partial f}{\partial \sigma_m} &= \frac{6 \sin \phi}{3 - \sin \phi} \\ \frac{\partial f}{\partial J'_2} &= \frac{\sqrt{3}}{2\sqrt{J'_2}} \\ \frac{\partial f}{\partial J'_3} &= 0\end{aligned}$$

Notice that when $\theta = \pm 30^\circ$, representing corners of the Tresca and Mohr-Coulomb yield surfaces, it is not possible to compute

$$\frac{\partial f}{\partial J'_2} \quad \text{and} \quad \frac{\partial f}{\partial J'_3}$$

This difficulty arises because the gradient vector cannot be uniquely defined at these corners. To prevent such difficulties, a provision is made to use the expressions for von Mises and Drucker-Prager yield criteria when $|\theta| \geq 29^\circ$.

6.2.9 Degenerate Cases: Plane Strain, Plane Stress and Uniaxial Stress

In the case of plane strain, the general expressions derived so far must be modified with

$$\tau_{yz} = \tau_{zx} = 0$$

and

$$\varepsilon_z = \Delta \varepsilon_z = 0$$

but

$$\Delta \varepsilon_z^e + \Delta \varepsilon_z^p = \Delta \varepsilon_z - \Delta \varepsilon_z^{thermal} = -\alpha \Delta T$$

The deviatoric stress invariants can be reduced to

$$J_2' = \frac{1}{2}(\sigma_x'^2 + \sigma_y'^2 + \sigma_z'^2) + \tau_{xy}^2$$

$$J_3' = \sigma_z'(\sigma_x'\sigma_y' - \tau_{xy}^2).$$

The gradient vector $\left\{\frac{\partial f}{\partial \sigma}\right\}$ will consist of four components $< \frac{\partial f}{\partial \sigma_x} \frac{\partial f}{\partial \sigma_y} \frac{\partial f}{\partial \sigma_z} \frac{\partial f}{\partial \tau_{xy}} >$, which can be computed with

$$\begin{aligned} \left\{\frac{\partial \sigma_m}{\partial \sigma}\right\}^T &= \frac{1}{3} < 1 \ 1 \ 1 \ 0 > \\ \left\{\frac{\partial J_2'}{\partial \sigma}\right\}^T &= < \sigma_x' \ \sigma_y' \ \sigma_z' \ 2\tau_{xy} > \\ \left\{\frac{\partial J_3'}{\partial \sigma}\right\}^T &= < (\sigma_y'\sigma_z' + \frac{J_2'}{3}), (\sigma_x'\sigma_z' + \frac{J_2'}{3}), (\sigma_x'\sigma_y' - \tau_{xy}^2 + \frac{J_2'}{3}), -2\sigma_z'\tau_{xy} > \end{aligned}$$

The elasticity matrix for the plane strain case is

$$[D_e] = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 0 \\ & 1 & \frac{\nu}{1-\nu} & 0 \\ & \text{Sym.} & 1 & 0 \\ & & & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix}$$

In the case of plane stress, the equations can be reduced with

$$\sigma_z = \tau_{yz} = \tau_{zx} = 0$$

but

$$\Delta \varepsilon_z = \Delta \varepsilon_z^e + \Delta \varepsilon_z^p \neq 0$$

and

$$\Delta \varepsilon_z^p = -\Delta \varepsilon_x^p - \Delta \varepsilon_y^p.$$

Notice that ε_z^p does not have to be stored as in the plane strain case. The deviatoric stress invariants can be computed with $\sigma_z = 0$. The elasticity matrix is reduced to:

$$[D_e] = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$$

In the case of uniaxial loading, all the stress components become zero except σ_x . The deviatoric stress invariants become simply

$$J_2' = \frac{1}{3} \sigma_x^2 \quad \text{and} \quad J_3' = \frac{2}{27} \sigma_x^3$$

The effective stresses become

$$\begin{aligned}
 f(\sigma) &= \text{ABS } |\sigma_x| && \text{for von Mises and Tresca} \\
 f(\sigma) &= \text{ABS } |\sigma_x| + \sigma_x \sin \phi && \text{for Mohr-Coulomb} \\
 f(\sigma) &= \text{ABS } |\sigma_x| + \frac{2 \sin \phi}{3 - \sin \phi} \sigma_x. && \text{for Drucker-Prager}
 \end{aligned}$$

The gradient vector can be reduced to a scalar, i.e.,

$$\begin{aligned}
 \frac{\partial f}{\partial \sigma_x} &= \frac{\sigma_x}{\sqrt{3} \sqrt{J'_2}} && \text{for von Mises and Tresca} \\
 \frac{\partial f}{\partial \sigma_x} &= \frac{\sigma_x}{\sqrt{3} \sqrt{J'_2}} + \sin \phi && \text{for Mohr-Coulomb} \\
 \frac{\partial f}{\partial \sigma_x} &= \frac{\sigma_x}{\sqrt{3} \sqrt{J'_2}} + \frac{2 \sin \phi}{3 - \sin \phi} && \text{for Drucker-Prager}
 \end{aligned}$$

The elasticity matrix is reduced to a scalar:

$$D_e = E.$$

6.2.10 Solution Algorithm for Elasto-Plastic Material

The material routine interfaces with element routines with the following I/O data:

- Input: $\bar{\varepsilon}^p$, $\{\alpha\}$, $\{\sigma\}$, $\{\Delta \varepsilon\}$
- Output: $\bar{\varepsilon}_{new}^p$, $\{\alpha_{new}\}$, $\{\sigma_{new}\}$, $[D_{ep}]$, $\bar{\sigma}$

The governing equation for the incremental process of elasto-plastic deformation is as follows:

$$d\sigma = D_e (d\varepsilon - d\varepsilon^p) = D_e d\varepsilon - d\lambda D_e \frac{\partial f}{\partial \sigma}$$

with

$$d\lambda = \frac{\left\{ \frac{\partial f}{\partial \sigma} \right\}^T D_e d\varepsilon}{H + \left\{ \frac{\partial f}{\partial \sigma} \right\}^T D_e \left\{ \frac{\partial f}{\partial \sigma} \right\}}$$

where H and $\left\{ \frac{\partial f}{\partial \sigma} \right\}$ are functions of $\{\sigma\}$.

The computational process in the plasticity routine is depicted in the flow diagram in Fig. 6.2.9. The procedure can be summarized in steps as follows:

Step 1. Initialize

$$\begin{aligned}
 \text{Zero} &= 10^{-7} * Y_1, & \bar{\varepsilon}_i^p &= \bar{\varepsilon}^p, \\
 \{\alpha\}_i &= \{\alpha\} & \text{and} & D_{ep} = D_e
 \end{aligned}$$

Step 2. Estimate the stress increment using the elasticity matrix.

$$\{\Delta\sigma_e\} = D_e\{\Delta\varepsilon\}$$

Then the trial stress is

$$\{\sigma^t\} = \{\sigma\} + \{\Delta\sigma_e\}$$

Step 3. Translate the yield surface for kinematic or combined hardening:

$$\{\sigma_0\} = \{\sigma - \alpha\}$$

$$\{\sigma_1\} = \{\sigma^t - \alpha\}$$

Step 4. Find the current yield stress (Y) from the input stress-strain curve as follows:

i) Increment k until $\bar{\varepsilon}_k^p \leq \bar{\varepsilon}^p < \bar{\varepsilon}_{k+1}^p$ where $\bar{\varepsilon}_k^p = \varepsilon_k - \frac{Y_k}{E}$.

ii) Compute the current plasticity modulus:

$$H_k = \frac{Y_{k+1} - Y_k}{\bar{\varepsilon}_{k+1}^p - \bar{\varepsilon}_k^p} = \frac{Y_{k+1} - Y_k}{(\varepsilon_{k+1} - \varepsilon_k) - \frac{1}{E}(Y_{k+1} - Y_k)}$$

iii) Compute the current yield stress:

$$Y = Y_1 + \beta(Y_k - Y_1) + \beta H_k(\bar{\varepsilon}^p - \bar{\varepsilon}_k^p)$$

where Y_1 is the initial yield point and $\beta = 0.$, 0.5 , or $1.$ for kinematic, combined, or isotropic hardening, respectively.

iv) Make corrections for the case of Mohr-Coulomb or Drucker-Prager criterion.

Step 5. Estimate the yield function:

$$F_0 = f(\sigma_0) - Y$$

$$F_1 = f(\sigma_1) - Y$$

Step 6. Determine whether the plastic deformation is involved.

If $F_1 \leq 0$, the process is elastic. Return with:

$$\sigma = \sigma^t$$

$$D_{ep} = D_e$$

Step 7. Determine whether the material is yielding.

If $|F_0| \leq 0$, the material is on the yield surface. Go to Step 10 with

$$\{\sigma\}_i = \{\sigma_0\} \quad \text{and} \quad \gamma = 0$$

Step 8. Check the error.

If $F_0 > 0$, exit with a message "Effective stress is greater than yield stress in element ID=xxx".

Step 9. Bring the stress state to the first yield point, if the material is elastic initially.

If $F_0 < 0$, material goes to plastic from the elastic state. Update the stress components to place the stress state on the yield point.

$$\{\sigma\}_i = \{\sigma_0\} + \gamma \{\Delta\sigma_e\} \quad \text{where } \gamma = -\frac{F_0}{F_1 - F_0}$$

Step 10. Find the number of incremental steps required as follows:

$$m = \text{Int.} \left(\frac{F_1}{\text{FSTRESS} * Y} \right) + 1$$

and

$$2 \leq m \leq \text{LOOPCT}$$

where FSTRESS is a user parameter in the NLPARM data entry and is defaulted to 0.2.

Step 11. If m indicates excessive increment size, set the flag to activate the bisection in order to reduce the load or time increment, which is based on

$$M = m/20 + 1$$

where bisection is activated if $M > 10$. Prior to Version 66, a warning message was issued instead of bisection: "Excessive incremental load is applied in element ID=xxx. Reduce the load increment by a factor of M for better solution."

Step 12. The governing equation will be integrated by the Euler method of step by step integration in m subincrements.

The incremental stress (equivalent to the equally divided subincrement of $\Delta\epsilon$) is

$$\{\Delta\sigma\} = \frac{1}{m} (1 - \gamma) \{\Delta\sigma_e\}$$

where $\{\Delta\sigma_e\} = D_e \{\Delta\epsilon\}$.

Step 13. Repeat the following loop m times.

- i) Compute $\left\{ \frac{\partial f}{\partial \sigma} \right\}_i$ and H_i based on $\{\sigma\}_i$ and $\bar{\epsilon}_i^p$
- ii) Find

$$\{d\}_i = [D_e] \left\{ \frac{\partial f}{\partial \sigma} \right\}_i$$

iii) Compute

$$\Delta\lambda_i = \frac{\left\{\frac{\partial f}{\partial \sigma}\right\}_i^T \{\Delta\sigma\}}{H_i + \left\{\frac{\partial f}{\partial \sigma}\right\}_i^T \{d\}_i}$$

iv) If $\Delta\lambda_i < -5 \times 10^{-5}$, set the unloading flag up (notice that $\Delta\lambda = 0$ by the end of this loop) and eliminate negative $\Delta\lambda$, i.e.,

$$\Delta\lambda_i = \text{MAX}(\Delta\lambda_i, 0.0)$$

v) Compute the stress subincrement

$$\{\Delta\sigma\}_i = \{\Delta\sigma\} - \Delta\lambda_i \{d\}_i$$

vi) If the hardening rule is not isotropic

$$\Delta\mu = \frac{\left\{\frac{\partial f}{\partial \sigma}\right\}_i^T \{\Delta\sigma\}_i - \beta H \Delta\lambda_i}{\left\{\frac{\partial f}{\partial \sigma}\right\}_i^T \{\sigma\}_i}$$

and

$$\Delta\mu = \text{MAX}(\Delta\mu, 0.0)$$

$$\{\Delta\alpha\}_i = \Delta\mu \{\sigma\}_i$$

$$\{\Delta\sigma\}_i = \{\Delta\sigma\}_i - \{\Delta\alpha\}_i$$

$$\{\alpha\}_{i+1} = \{\alpha\}_i + \{\Delta\alpha\}_i$$

vii) Update the stress and the plastic strain

$$\{\sigma\}_{i+1} = \{\sigma\}_i + \{\Delta\sigma\}_i$$

$$\bar{\varepsilon}_{i+1}^p = \bar{\varepsilon}_i^p + \Delta\lambda_i$$

and

$$\{\varepsilon^p\} = \{\varepsilon^p\} + \Delta\lambda_i \left\{\frac{\partial f}{\partial \sigma}\right\}_i$$

Step 14. Error Correction step

i) Compute the error: $\delta = F_3 = f(\sigma_{i+m}) - Y(\bar{\varepsilon}_{i+m}^p)$

ii) If $|\delta| \text{ FSTRESS} * Y$ in the converging state, EXIT with a fatal error message, "Error exceeded xx% of current yield stress in element ID=xxx".

iii) If $|\delta| \leq 0$, ignore the error and jump to Step 15.

iv) If $\delta \geq 0$, make a correction as follows [6.7]:

$$\{\sigma\}_{i+m} = \{\sigma\}_{i+m} * \frac{Y}{\delta + Y}$$

$$\{\alpha\}_{i+m} = \{\alpha\}_{i+m} - \frac{\delta}{Y} \{\sigma\}_{i+m}$$

Step 15. Offset the center of the yield surface.

$$\{\sigma\}_{i+m} = \{\sigma\}_{i+m} + \{\alpha\}_{i+m}$$

Step 16. If the unloading flag is up, the material is assumed to be unloaded. Since the material becomes elastic during unloading, return with the elastic tangential matrix, i.e.,

$$[D_{ep}] = [D_e]$$

Step 17. Otherwise, return with the elasto-plastic tangential matrix:

$$D_{ep} = D_e - \frac{d d^T}{H + \left\{ \frac{\partial f}{\partial \sigma} \right\}^T d}$$

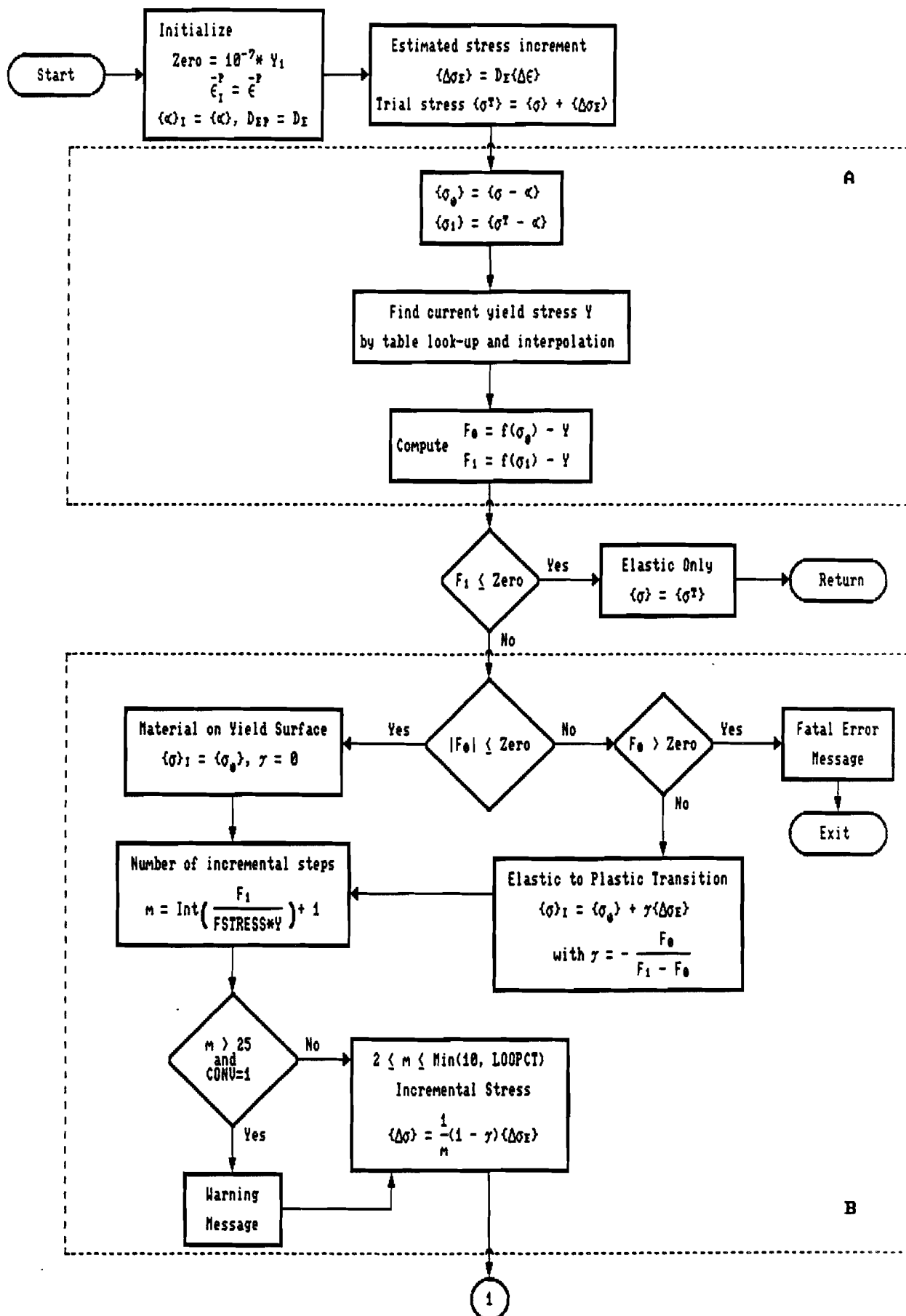


Figure 6.2.9a Flow Diagram for Elasto-Plastic Material (NMEP)

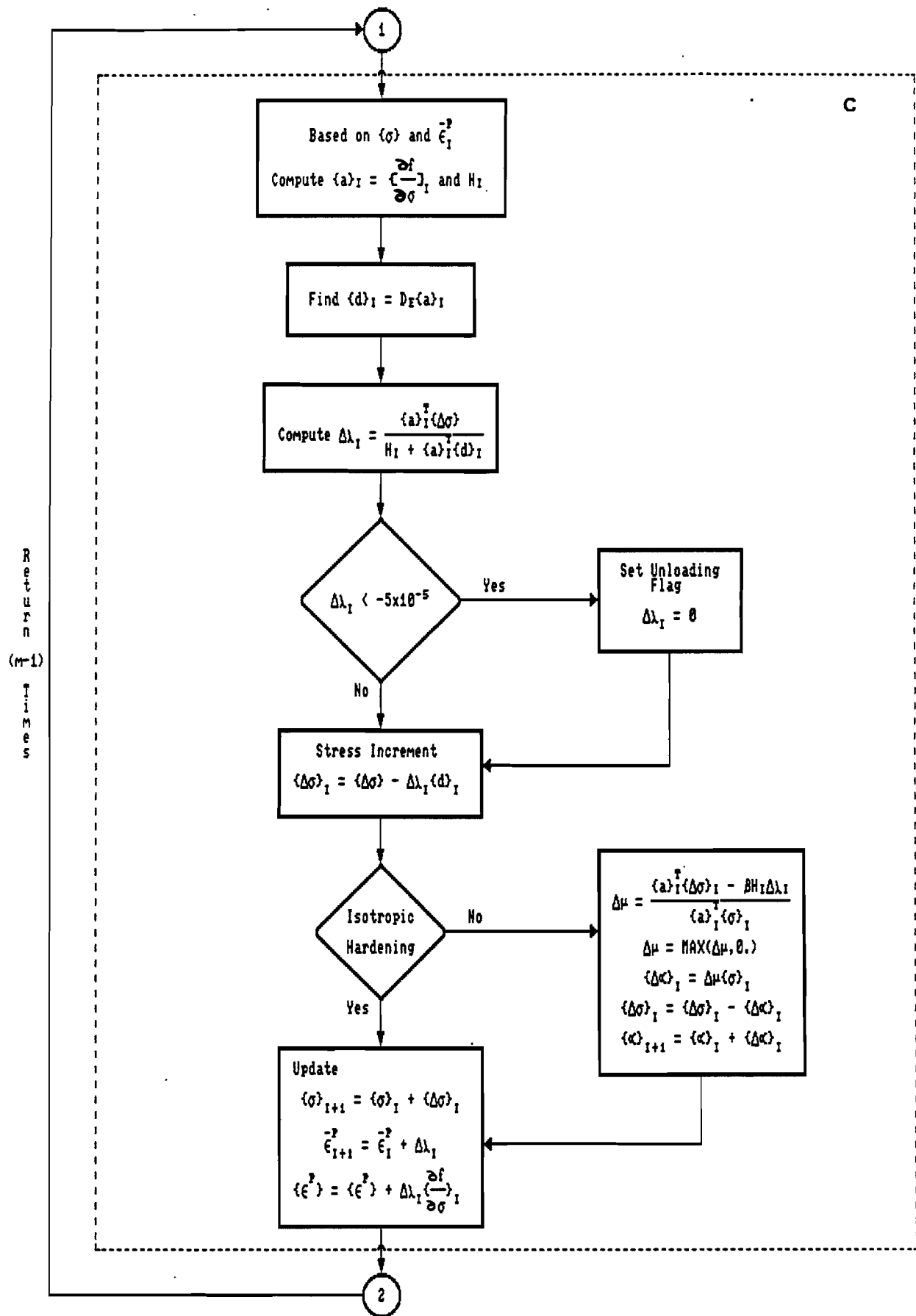


Figure 6.2.9b Flow Diagram for Elasto-Plastic Material (NMEP)

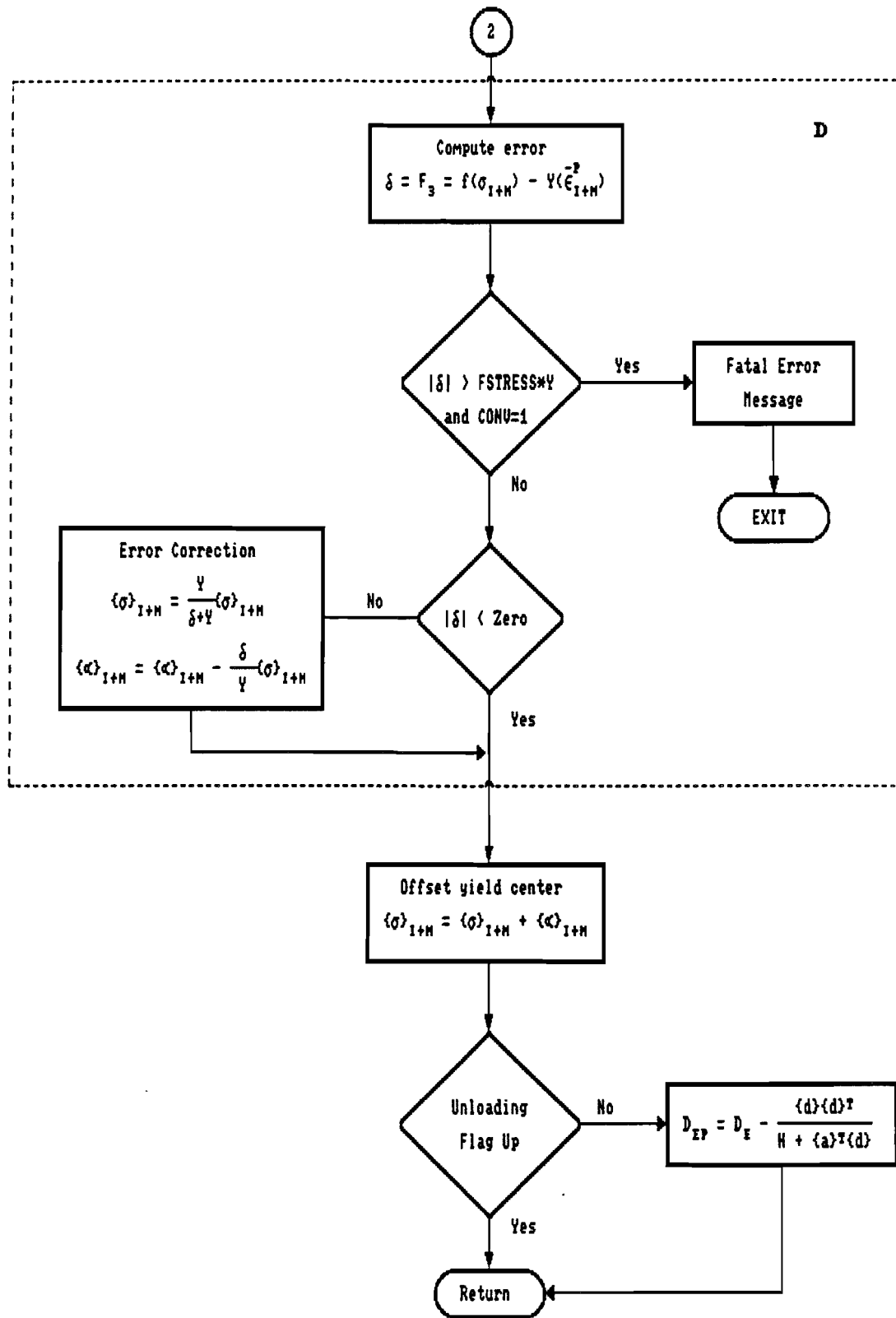


Figure 6.2.9c Flow Diagram for Elasto-Plastic Material (NMEP)

6.2.11 Verification and Validation

The elasto-plastic material capabilities were verified with respect to the algorithm, accuracy, and programming errors with a wide variety of problems. All types of elements (1-D, 2-D and 3-D) have been tested and known solutions have been reproduced [6.3, 6.4]. Some of the example problems of interest are illustrated below:

Uniaxial Loading and Apparent Poisson's Ratio

Experiments have shown that the volume of material does not change due to plastic strain. This incompressibility is signified by a Poisson's ratio of 0.5 in the Prandtl-Reuss stress-strain relations. The lateral dimension of the specimen shrinks during the simple tension test as shown in Fig. 6.2.10(a). The Poisson's ratio is defined as $\nu = -\varepsilon_w/\varepsilon_a$ in the elastic domain. If the Poisson's ratio is to be measured from the elastic-plastic deformation (termed apparent Poisson's ratio), a transition would be observed from an initially elastic (ν) to a smeared value with 0.5 due to plastic strain. The strain components in simple tension for the elastic-plastic deformation can be decomposed into elastic and plastic parts, i.e.,

$$\varepsilon_a = \varepsilon_a^e + \varepsilon_a^p \quad ; \text{ in tension direction}$$

and

$$\varepsilon_w = -\nu \varepsilon_a^e - 0.5 \varepsilon_a^p \quad ; \text{ in width direction}$$

The apparent Poisson's ratio then becomes

$$\nu^* = -\frac{\varepsilon_w}{\varepsilon_a} = \frac{\nu + 0.5\beta}{1 + \beta} = \nu + (0.5 - \nu)\frac{\beta}{1 + \beta}$$

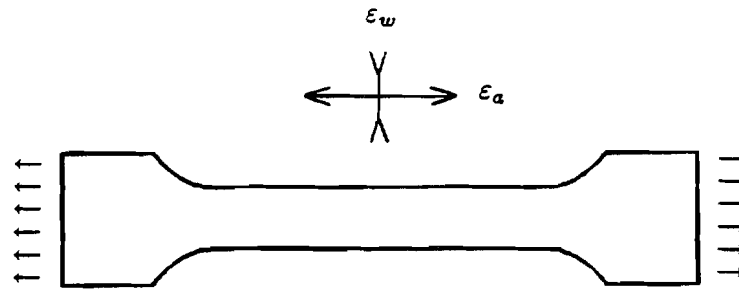
with

$$\beta = \frac{\varepsilon_a^p}{\varepsilon_a^e}$$

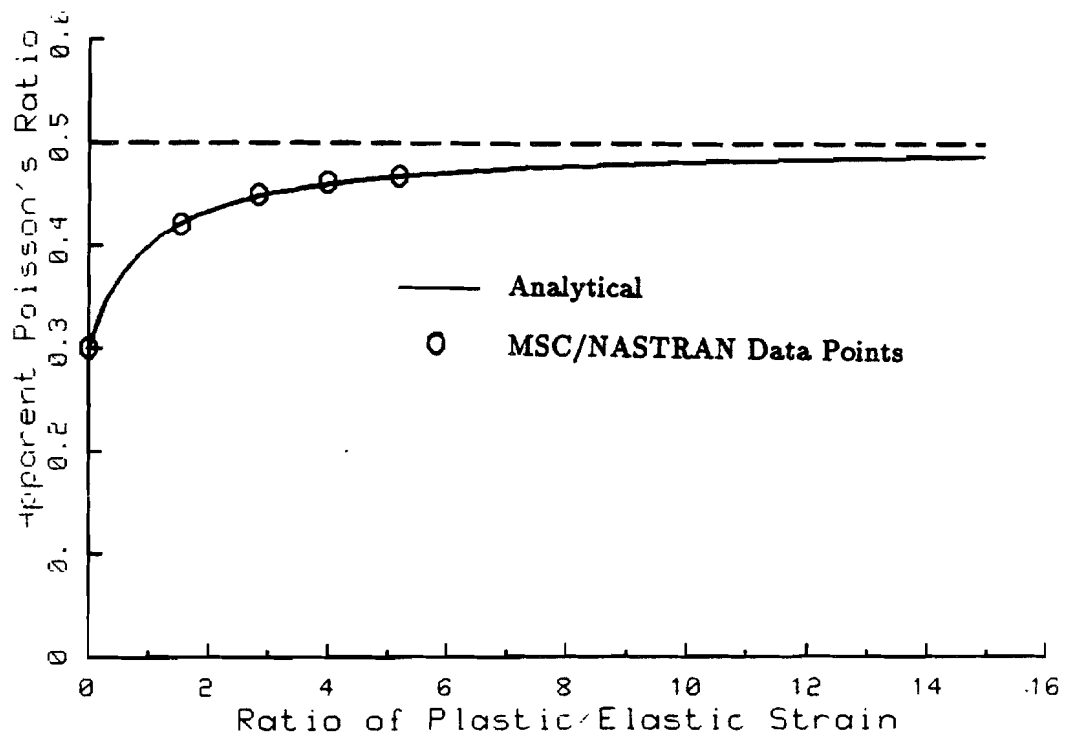
The variation of ν^* with respect to β is plotted in Fig. 6.2.10(b). It is shown that the apparent Poisson's ratio ν^* is a continuous and smooth function of β with a value of ν for $\beta = 0$ and asymptotic to 0.5 as β approaches infinity. However, β has a finite value limited by material properties (E and H), i.e.,

$$\beta_{max} = \lim_{\varepsilon^p \rightarrow \infty} \frac{E}{H(\varepsilon^p)}$$

A finite element simulation of a simple tension test resulted in data points circled in Fig. 6.2.10(b).



(a) Uniaxial Tension Specimen



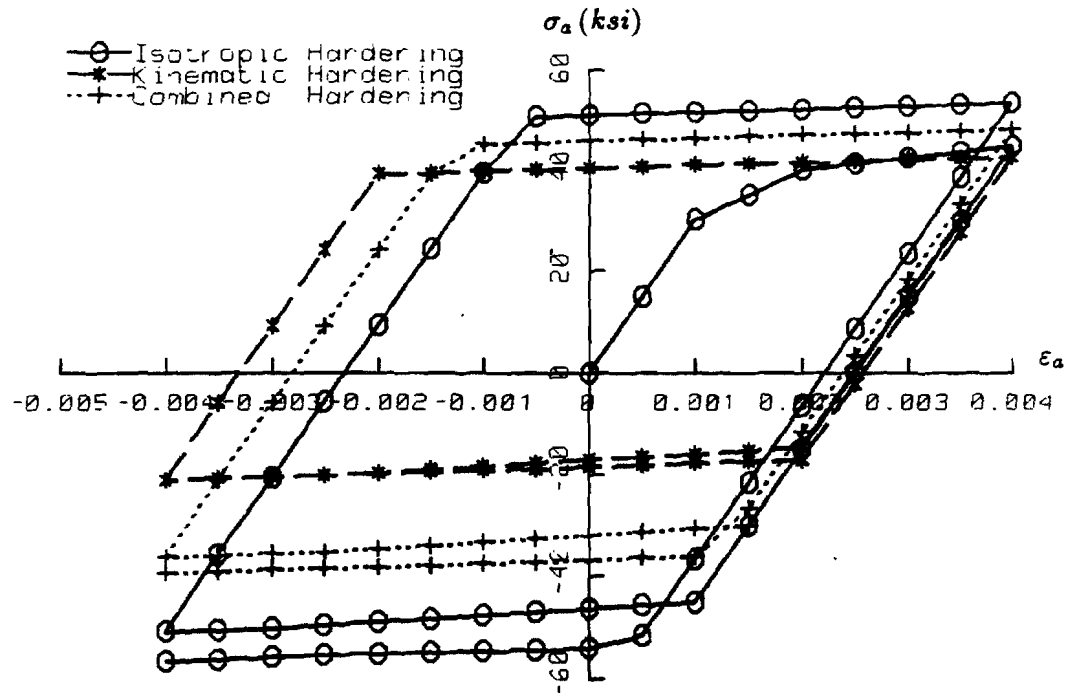
(b) Apparent Poisson's Ratio

Figure 6.2.10 Uniaxial Loading and Apparent Poisson's Ratio

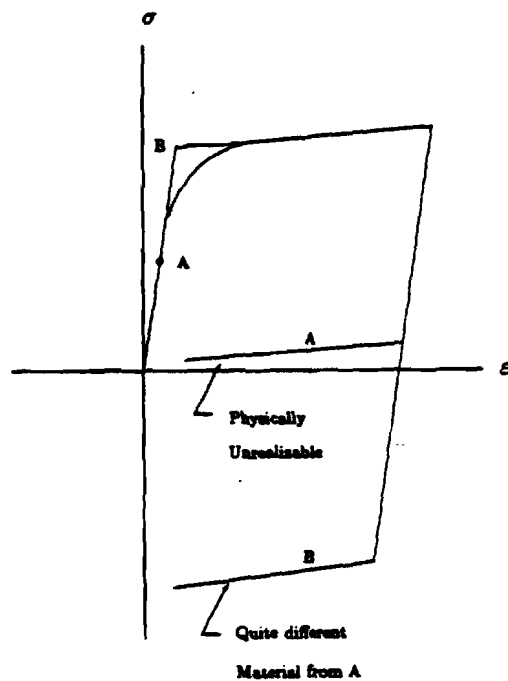
Cyclic Loading and Hysteresis

If the stress state is changed such that the stress value moves into the elastic region, the process is called unloading. The cyclic loading, involving successive loading, unloading and reloading, makes the problem more complicated due to a phenomenon known as the Bauschinger effect. A finite element simulation of the cyclic loading has been performed with a thin plate specimen subjected to repetitive uniaxial tension and compression. The loading cycle was controlled by strain using the enforced displacements. The stress-strain responses of the material obeying the von Mises yield criterion are shown in Fig. 6.2.11(a) using the isotropic, kinematic and combined hardening rules. During the first tensile loading phase shown in the first quadrant (O to A), the responses for the three hardening rules are identical to the original strain-strain curve specified in the input data. The kinematic hardening exhibits the Bauschinger effect upon reloading in the reverse direction, while the isotropic hardening shows a monotonic increase in the yield stress. The combined hardening curve falls in between the isotropic and kinematic hardening curves, as expected. The plasticity modulus is kept continuous upon reloading for all three hardening rules, i.e., an identical modulus is resumed upon reloading from the previously loaded point. Note that not all the flexure points are represented in the plot because of discrete solution points. The solution process progressed without any difficulty caused by unloading.

Kinematic hardening has an advantage over isotropic hardening for the Bauschinger effect. However, kinematic hardening only describes a perfect kinematic hardening material for which the size of the yield surface does not change. In reality, such a material does not exist. Suppose a material has a low initial yield stress and a high rate of hardening, as some materials may possess. If kinematic hardening is used to model such a material exactly, the reloading in compression could occur while the applied stress is still in tension as shown in Fig. 6.2.11(b). This behavior is physically unrealizable. In order to avoid such an anomaly, the initial yield point may have to be artificially elevated. The difficulty of such problems lies in the fact that the Bauschinger effect is known as a qualitative and not a quantitative description. The combined hardening rule is complementary for such cases.



(a) Stress Strain Hysteresis



(b) Anomaly in Kinematic Hardening

Figure 6.2.11 Strain-Controlled Cyclic Loading

6.3 NONLINEAR ELASTICITY

The nonlinear elastic capability was developed for Version 62 to predict the multi-axial stress-strain behavior for the nonlinear elastic material, for which only the simple tension stress-strain data are available. The theory and algorithm are adequate to trace the stress-strain curve accurately for the uniaxial loading cases. However, the theory is not based on the classical theory of finite elasticity [6.10]. Consequently, some of the constitutive relations may be violated in the multiaxial stress cases. Nevertheless, this capability can be justified by stating that the design is not intended for the large-strain deformation.

The current nonlinear elastic capability, modified in Version 65, can accommodate bilateral properties (uniaxial tension-compression) accurately for the uniaxial deformations. This design is considered adequate for the multiaxial deformation if the strain remains in the small range. Intrinsic deficiency due to oversimplification of the theory will introduce errors, but the error may not be noticeable unless large strains are applied in the multiaxial deformation. For large strain applications the generalized hyperelastic material based on the classical theory of finite elasticity is being planned to be implemented.

6.3.1 Theoretical Basis

The nonlinear elastic capability in the software was designed to satisfy the equivalence of the deformation work per unit volume in the simple tension to the strain energy per unit volume (conservation of energy), while the work done for deformation may be defined by a stress-strain curve in simple tension, i.e.,

$$\int \bar{\sigma} d\bar{\varepsilon} = \int \langle \sigma \rangle \{d\varepsilon\} \quad (6.3.1)$$

It was further assumed that the effective strain ($\bar{\varepsilon}$) may be defined by

$$\frac{1}{2} E \bar{\varepsilon}^2 = \frac{1}{2} \langle \varepsilon \rangle [D_e] \{\varepsilon\} \quad (6.3.2)$$

From the total differential of Eq. (6.3.2), we obtain

$$d\bar{\varepsilon} = \frac{1}{E\bar{\varepsilon}} \langle \varepsilon \rangle [D_e] \{d\varepsilon\} \quad (6.3.3)$$

Substituting Eq. (6.3.3) into Eq. (6.3.1), stresses may be expressed in terms of total strains, i.e.,

$$\{\sigma\} = \frac{\bar{\sigma}}{E\bar{\varepsilon}} [D_e] \{\varepsilon\} \quad (6.3.4)$$

The tangential matrix for such material may be obtained by differentiating Eq. (6.3.4), i.e.,

$$[D_{ne}] = \frac{\partial \{\sigma\}}{\partial \{\varepsilon\}} = \frac{\bar{\sigma}}{E\bar{\varepsilon}} [D_e] + \frac{1}{(E\bar{\varepsilon})^2} \left(\frac{\partial \bar{\sigma}}{\partial \bar{\varepsilon}} - \frac{\bar{\sigma}}{\bar{\varepsilon}} \right) [D_e] \{\varepsilon\} \langle \varepsilon \rangle [D_e]^T \quad (6.3.5)$$

6.3.2 Solution Algorithm

Users specify nonlinear stress-strain curve, $\bar{\sigma}(\bar{\epsilon})$, in the TABLES1 entry along with MAT1 and MATS1 entries. Young's modulus and the Poisson's ratio are available from MAT1 entry. The element routine calls the material routine driver NMATD, which calls the subroutine NMEVD for the nonlinear material. The material routine interfaces with element routines with the following data:

$$\begin{aligned}\text{Input} &: \{\sigma\}_{\text{old}}, \{\epsilon\}_{\text{old}}, \{\Delta\epsilon\}, E, \nu \\ \text{Output} &: \{\sigma\}_{\text{new}}, \{\epsilon\}_{\text{new}}, [D_{\text{ne}}]\end{aligned}$$

The computational procedure is described below:

Step 1. Upon entry to the subroutine NMEVD, the new strain state is computed by

$$\{\epsilon\}_{\text{new}} = \{\epsilon\}_{\text{old}} + \{\Delta\epsilon\} \quad (6.3.6)$$

Step 2. The effective strain ($\bar{\epsilon}$) is computed based on $\{\epsilon\}_{\text{new}}$ by

$$\bar{\epsilon}^2 = \frac{1}{E} \langle \epsilon \rangle \{\sigma_e\} \quad (6.3.7)$$

where

$$\{\sigma_e\} = [D_e]\{\epsilon\}$$

or

$$\begin{aligned}\bar{\epsilon}^2 &= \frac{1}{1-\nu^2} [\epsilon_x^2 + \epsilon_y^2 + 2\nu\epsilon_x\epsilon_y + \frac{1-\nu}{2}\gamma_{xy}^2] && \text{for plane stress} \\ \bar{\epsilon}^2 &= \frac{1}{(1-2\nu)(1+\nu)} [(1-\nu)(\epsilon_x^2 + \epsilon_y^2 + \epsilon_z^2) + 2\nu(\epsilon_x\epsilon_y + \epsilon_y\epsilon_z + \epsilon_z\epsilon_x) \\ &\quad + \frac{(1-2\nu)}{2}(\gamma_{xy}^2 + \gamma_{yz}^2 + \gamma_{zx}^2)] && \text{for 3-D Solid}\end{aligned}$$

Step 3. The effective stress ($\bar{\sigma}$) is determined by looking-up the user-specified stress-strain curve for $\bar{\epsilon}$.

Step 4. The new stress state is determined by

$$\{\sigma\}_{\text{new}} = \frac{\bar{\sigma}}{E\bar{\epsilon}} \{\sigma_e\} \quad (6.3.4a)$$

Step 5. The tangential matrix is determined by

$$[D_{\text{ne}}] = \frac{\bar{\sigma}}{E\bar{\epsilon}} [D_e] + \frac{1}{(E\bar{\epsilon})^2} \left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}} - \frac{\bar{\sigma}}{\bar{\epsilon}} \right) \{\sigma_e\} \{\sigma_e\}^T, \quad (6.3.5a)$$

for which $\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}$ is the slope at $\bar{\epsilon}$, i.e.,

$$\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}} = \frac{y_{k+1} - y_k}{x_{k+1} - x_k} \quad (6.3.8)$$

where (x_k, y_k) is the k-th data point in the TABLES1 entry and k is determined such that $x_k \leq \bar{\epsilon} < x_{k+1}$. It is noted that $k=1$ for $\bar{\epsilon} < x_1$ and $k = k_{max} - 1$ for $\bar{\epsilon} \geq x_{max}$.

6.3.3 Adaptation of Uniaxial Compression Stress-Strain Curve

Until Version 64, the effective stress-strain curve is determined only by the uniaxial tension data in two and three dimensional cases. The uniaxial compression data will be ignored if provided for two and three dimensional elements. Some materials, however, exhibit appreciably different behavior in compression from that in tension even in the small strain range. A typical case of such material is shown in Fig. 6.3.1.

For uniaxial loading, the magnitude of the strain in that direction becomes the effective strain, i.e.,

$$\begin{aligned} \bar{\epsilon} &= \epsilon_x & \text{for uniaxial tension in } x \\ \bar{\epsilon} &= -\epsilon_x & \text{for uniaxial compression in } x \end{aligned}$$

We need to find the effective stress ($\bar{\sigma}$) corresponding to $\bar{\epsilon}$. There are two known data points as shown in Fig. 6.3.2; namely the effective stress for uniaxial tension ($\bar{\sigma}_t$) and the effective stress for uniaxial compression ($\bar{\sigma}_c$). Some method of interpolation or extrapolation is required to predict the effective stress for the general stress state using two known data points.

The first stress invariant (I_1) is adopted for interpolation/extrapolation, i.e.,

$$I_1 = \sigma_x + \sigma_y + \sigma_z$$

Considering that the pure shear is in the midway between simple tension and simple compression, it seems appropriate to use the first stress invariant. Hydrostatic tension and compression cases will impose lower and upper bounds for extrapolation, i.e.,

$$\begin{aligned} I_1 &= \sigma_x & \text{for uniaxial tension/compression} \\ I_1 &= 0 & \text{for pure shear} \\ I_1 &= 3p & \text{for hydrostatic pressure} \end{aligned}$$

The instantaneous modulus ($\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}$) should be interpolated or extrapolated in the same manner.

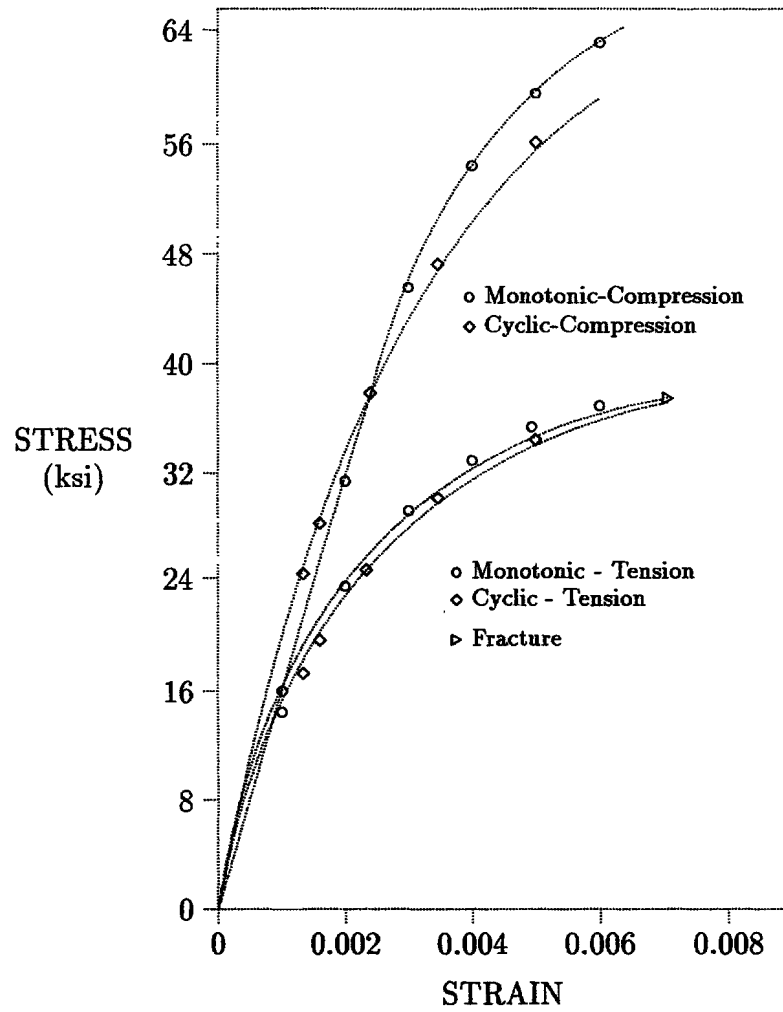


Figure 6.3.1. Stress-Strain Curves for Gray Cast Iron

6.3.4 Computational Procedure for Bilateral Stress-Strain Relations

The new stress state is proportional in magnitude to the effective stress ($\bar{\sigma}$), which should be determined as follows:

1. Compute the effective stress ($\bar{\sigma}_e$) based on $\{\sigma_e\}$, i.e.,

$$\{\sigma_e\} = [D_e]\{\varepsilon\}_{new}$$

$$\bar{\sigma} = \sqrt{\frac{1}{2}[(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2] + 3(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2)} \quad \text{for 3D}$$

$$\bar{\sigma} = \sqrt{\sigma_x^2 - \sigma_x\sigma_y + \sigma_y^2 + 3\tau_{xy}^2} \quad \text{for plane stress}$$

$$\bar{\sigma} = \sqrt{\frac{1}{2}[(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2] + 3\tau_{xy}^2} \quad \text{for plane strain}$$

2. Compute the first invariant of $\{\sigma_e\}$:

$$I_1 = \sigma_x + \sigma_y + \sigma_z$$

where $\sigma_z = 0$ for plane stress.

3. Determine the ratio (r) by normalizing I_1 by $\bar{\sigma}_e$, i.e.,

$$r = \frac{I_1}{\bar{\sigma}_e}$$

where r signifies the relative distance from the midpoint of $\bar{\sigma}_c$ and $\bar{\sigma}_t$ at $\bar{\epsilon}$, as shown in Fig 6.3.2. It would be implausible to process a large value of r (such is the case with a hydrostatic load). Therefore, r will be confined to a plausible range, $-1 \leq r \leq 1$. The value will be reset to the limit ($r = \pm 1$) if r lies outside the range.

4. Look up the user-specified stress-strain curve in the TABLES1 entry and determine $\bar{\sigma}_t$ and $\bar{\sigma}_c$, i.e.,

$$\begin{aligned} \bar{\sigma}_t &= \bar{\sigma}(\bar{\epsilon}) \\ \bar{\sigma}_c &= -\bar{\sigma}(-\bar{\epsilon}) \end{aligned}$$

5. Determine $\bar{\sigma}$ based on $\bar{\sigma}_t, \bar{\sigma}_c$ and r , i.e.,

$$\bar{\sigma} = \frac{\bar{\sigma}_t + \bar{\sigma}_c}{2} + r \frac{\bar{\sigma}_t - \bar{\sigma}_c}{2}$$

For the tangent matrix, the instantaneous modulus ($\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}$) should be determined using the same ratio (r) as follows:

1. Compute the instantaneous slope at $\bar{\epsilon}$ for tension, i.e.,

$$\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_t = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \quad \text{for } x_i \leq \bar{\epsilon} < x_{i+1}$$

where (x_i, y_i) is the i -th data point in the TABLES1 entry.

2. Compute the instantaneous slope at $(-\bar{\epsilon})$ for compression, i.e.,

$$\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_c = \frac{y_{j+1} - y_j}{x_{j+1} - x_j} \quad \text{for } x_j \leq -\bar{\epsilon} < x_{j+1}$$

3. Determine $\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)$ based on $\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_t$, $\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_c$, and r , i.e.,

$$\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}} = \frac{1}{2} \left[\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_t + \left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_c \right] + \frac{r}{2} \left[\left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_t - \left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}}\right)_c \right]$$

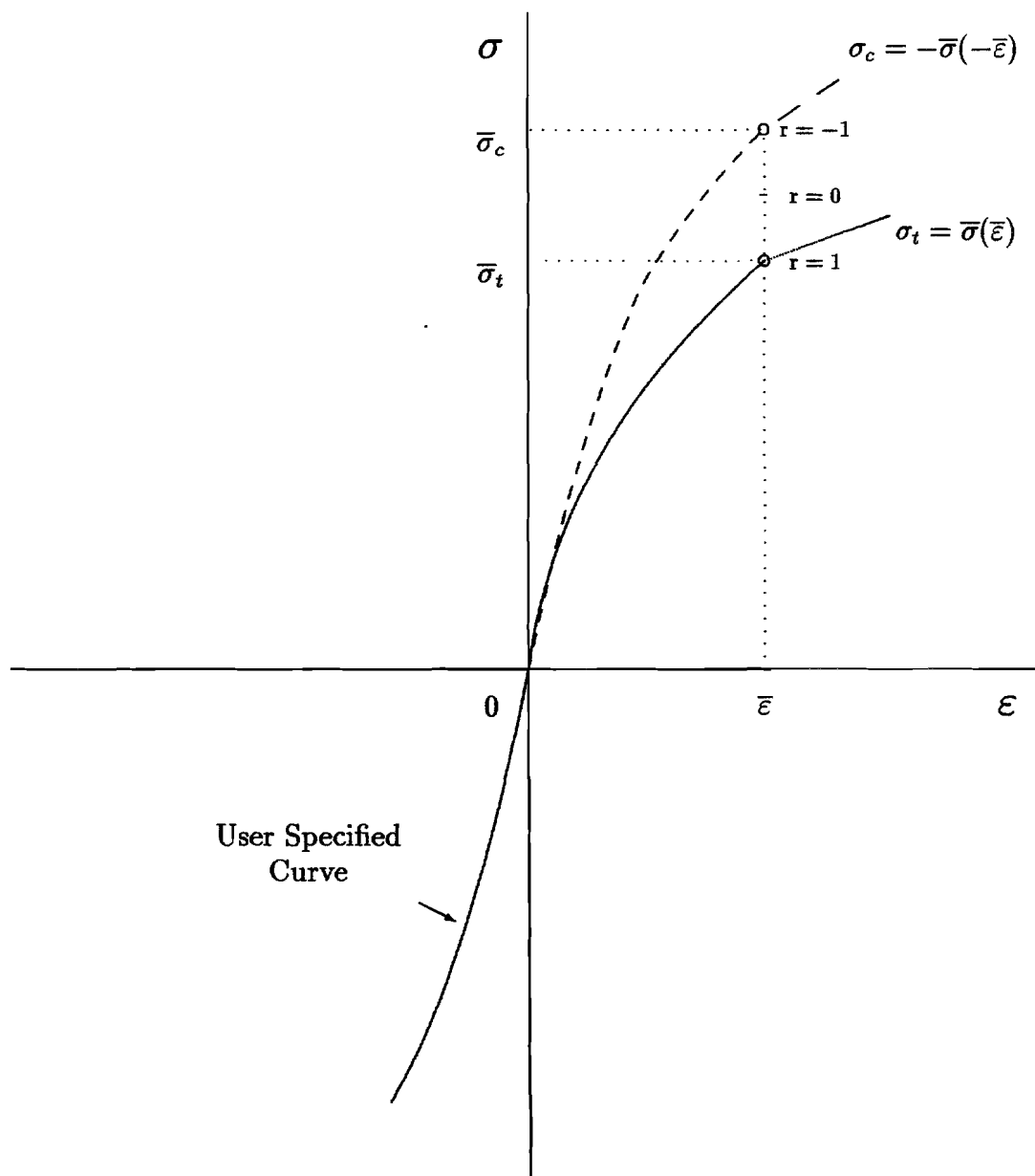


Figure 6.3.2. TABLES1 INPUT for Stress-Strain

6.4 CREEP AND VISCOELASTICITY

A creep analysis capability, using a generalized viscoelastic model [6.11], is introduced to process the creep behavior coupled with elastoplastic deformation. The formulation is based on the step-by-step time integration of the Kelvin-Maxwell rheological model with non-constant parameters.

The concept of a rheological model is extended to the multiaxial stresses by the Prandtl-Reuss stress-strain relationship, from which the tangential stiffness matrix is formed for Newton's iteration. If the plastic deformation is coupled with creep, the algorithm will seek a solution in two distinct steps. Various choices of empirical creep laws are available and small variations in temperature are allowed as implemented in the software.

6.4.1 Introduction

At the macroscopic level, the creep phenomenon is best observed in the uniaxial test under constant load and the relaxation test under constant strain at constant temperature. A specimen subjected to a constant uniaxial tension at an elevated temperature exhibits three distinct phases in a time frame: primary creep, secondary creep and the tertiary creep to rupture. If the specimen is unloaded after some creep deformation, the elastic strain is immediately recovered and a portion of the creep strain is gradually recovered. The recoverable portion of the creep deformation is called primary creep and the non-recoverable portion, secondary creep. The tertiary creep, similar to necking in plasticity, is considered as a localized instability phenomenon, which is beyond the scope of this handbook.

Elaborating on the Kelvin-Maxwell rheological creep model developed by Badani [6.12], the creep behavior is coupled with plastic deformation and the variable temperature effect is included. The Kelvin element represents the primary creep behavior and the Maxwell element describes the secondary creep. For a generalization of the viscoelastic material behavior, the rheological model parameters are treated as nonlinear functions of the effective stress and the temperature. The step-by-step integration is performed assuming that the strain-rate varies linearly during each time increment while the rheological model parameters remain constant for a short time interval.

A number of empirical creep laws, recommended by the Oak Ridge National Laboratory (ORNL), are provided along with options for general tabular input of the rheological model parameters as functions of effective stress. When the creep characteristics are specified in terms of empirical creep laws, the program converts the empirical formula to the corresponding rheological model.

6.4.2 Effects of Stress and Temperature

An analytical solution to the Kelvin-Maxwell model subjected to a constant stress (σ) is given by

$$\varepsilon_{total}^c = \frac{\sigma}{C_s} t + \varepsilon_{primary}^c \quad (6.4.1)$$

with

$$\varepsilon_{primary}^c = \frac{\sigma}{K_p} \left[1 - e^{-(K_p/C_p)t} \right]$$

where t denotes time and parameters K_p , C_p , and C_s are identified in Fig. 6.4.1. For a varying stress case, however, an instantaneous strain-rate should be employed to have the creep hardening effects properly accounted. The creep hardening (and softening) effects, assumed to be exhibited by the primary creep, are accounted for by using the creep hardening law, wherein creep strain rates are expressed in terms of primary creep strain rather than time, i.e.

$$\dot{\varepsilon}_{total}^c = \frac{\sigma}{C_s} + \dot{\varepsilon}_{primary}^c \quad (6.4.2)$$

where

$$\dot{\varepsilon}_{primary}^c = \frac{1}{C_p} (\sigma - K_p \varepsilon_{primary}^c).$$

Notice that the creep strain rate reverses the sign with proper effects of the cumulative primary creep when the sign of the stress reverses. Due to this virtue, the stress reversal effects are properly taken into account without any special provision in the algorithm.

The creep strain rate is, in general, a function of absolute temperature. Microscopically, the creep deformation is an integrated effect of dislocations of the crystal structure primarily due to thermal activation and stress. The effects of temperature at the microscopic level can be quantified as

$$\dot{\varepsilon}^c = A e^{-\Delta H/RT} \quad (6.4.3)$$

where ΔH is the energy of activation, R is the gas constant, T is the absolute temperature and A is in strain/unit time. On this basis, the creep model parameters measured at a reference temperature (T_0) can be used to compute the creep strain rate at the temperature T in the vicinity of T_0 , allowing small variations in the ambient temperature. The creep model parameters are corrected as

$$C_s(T) = \frac{C_s(T_0)}{F_c} \quad \text{and} \quad C_p(T) = \frac{C_p(T_0)}{F_c} \quad (6.4.4)$$

where

$$F_c = \frac{\dot{\varepsilon}^c}{\dot{\varepsilon}_0^c} = \left(e^{-\Delta H/RT_0} \right)^{\left(\frac{T_0}{T} - 1 \right)}.$$

6.4.3 Equilibrium of a Kelvin–Maxwell Model for a Stress Component

Suppose that the Kelvin-Maxwell model is applied to a typical component of a deviatoric stress-strain pair, denoted by $(\Delta s_1, e_1)$. Referring to Fig. 6.4.1, the state equilibrium equation of the model at any instant is expressed as:

$$[C] \{\Delta \dot{e}\} + [K] \{\Delta e\} = \{\Delta s\} \quad (6.4.5)$$

where

$$[C] = \begin{bmatrix} C_s & -C_s \\ -C_s & (C_p + C_s) \end{bmatrix},$$

$$[K] = \begin{bmatrix} 0 & 0 \\ 0 & K_p \end{bmatrix},$$

and

$$\{\Delta s\}^T = \langle \Delta s_1 \quad 0 \rangle.$$

Introducing a linearly varying strain-rate during the time interval, the strain-rate increment can be expressed as

$$\{\Delta \dot{e}\} = \frac{2}{\Delta t} [\Delta e(t) - \dot{e}(t) \Delta t], \quad (6.4.6)$$

by which Eq. (6.4.5) is reduced to

$$\left[\frac{2}{\Delta t} C + K \right] \{\Delta e\} = 2[C] \{\dot{e}\} + \{\Delta s\}. \quad (6.4.7)$$

Defining the stiffness of the primary and secondary creep elements by

$$k_1 = K_p + \frac{2C_p}{\Delta t} \quad \text{and} \quad k_2 = \frac{2C_s}{\Delta t}, \quad (6.4.8)$$

the equivalent creep stiffness (k_c) for a deviatoric stress-strain pair can be determined by

$$k_c = \frac{k_1 k_2}{k_1 + k_2}.$$

Solving Eq. (6.4.7) for Δe_1 gives

$$k_c \Delta e_1 = \Delta s' + \Delta s_1 \quad (6.4.9)$$

where $\Delta s'$ is a pseudo-incremental stress, determined by

$$\Delta s' = 2 \left[\frac{C_s}{k_2} (\dot{e}_1 - \dot{e}_2) + \frac{C_p}{k_1} \dot{e}_2 \right] k_c. \quad (6.4.10)$$

The pseudo-incremental stress represents the change in deviatoric stress component due to creep relaxation. This value of $\Delta s'$ can be converted to the equivalent psuedo-incremental

strain ($\Delta e'$), which represents the change in deviatoric strain component due to creep. By virtue of creep volume constancy, deviatoric strains ($\Delta e'$) and the strain rates (\dot{e}_1 and \dot{e}_2) in Eq. (6.4.10) are identical to the ordinary strain components. Hence the pseudo-incremental strain can be expressed as

$$\Delta e' = \frac{\Delta s'}{k_c} = 2 \left[\frac{C_s}{k_2} (\dot{e}_1 - \dot{e}_2) + \frac{C_p}{k_1} \dot{e}_2 \right] \quad (6.4.11)$$

in which the first and the second terms are contributions of the Maxwell and the Kelvin elements, respectively. Notice that as Δt approaches zero, Eq. (6.4.11) reduces to

$$\Delta e' \simeq \Delta t \dot{e}_1 = \Delta t \dot{e}_{total}^c$$

which makes it easier to comprehend Eq. (6.4.11).

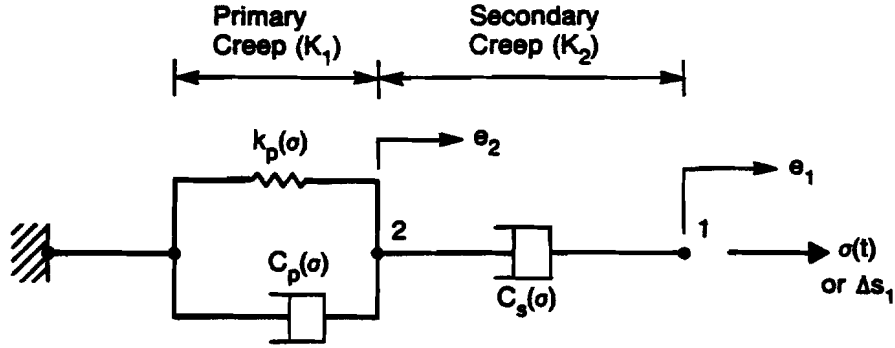


Figure 6.4.1. Rheological model

6.4.4 Adaptation of Rheological Model to Multiaxial Stress State

The concept of the rheological model for a stress-strain pair is extended to the general multiaxial stresses with the aid of effective stress and effective strain-rate by adopting the Prandtl-Reuss stress-strain relationship, i.e.,

$$\{\dot{\epsilon}^c\} = \frac{3}{2} \frac{\dot{\bar{\epsilon}}^c}{\bar{\sigma}} \{s\} \quad (6.4.12)$$

where creep strain rates

$$\{\dot{\epsilon}^c\}^T = \langle \dot{\epsilon}_x^c \quad \dot{\epsilon}_y^c \quad \dot{\epsilon}_z^c \quad \dot{\gamma}_{xy}^c \quad \dot{\gamma}_{yz}^c \quad \dot{\gamma}_{zx}^c \rangle,$$

deviatoric stresses

$$\{s\}^T = \langle \sigma'_x \quad \sigma'_y \quad \sigma'_z \quad 2\tau_{xy} \quad 2\tau_{yz} \quad 2\tau_{zx} \rangle,$$

and the effective stress ($\bar{\sigma}$) and the effective creep strain rate ($\dot{\bar{\epsilon}}^c$) are defined analogously to those in plasticity with von Mises yield criterion.

Implied by this relationship, there is a unique set of rheological parameters (\bar{K}_p , \bar{C}_p , and \bar{C}_s) based on the effective stress, which can be related to all the deviatoric stress components and the corresponding creep strain rates. In terms of effective creep model parameters, Eqs. (6.4.8) and (6.4.11) may be rewritten as follows:

$$k_1 = \frac{2}{3} \left(\bar{K}_p + \frac{2\bar{C}_p}{\Delta t} \right) \quad k_2 = \frac{4}{3} \frac{\bar{C}_s}{\Delta t} \quad (6.4.8a)$$

and

$$\{\Delta\epsilon'\} = \frac{4}{3} \left[\frac{\bar{C}_s}{k_2} \{\dot{\epsilon}_{total}^c - \dot{\epsilon}_{primary}^c\} + \frac{\bar{C}_p}{k_1} \{\dot{\epsilon}_{primary}^c\} \right] \quad (6.4.11a)$$

The creep strain rates in Eq. (6.4.2) may be rewritten likewise, i.e., in terms of deviatoric stresses

$$\{\dot{\epsilon}_{total}^c\} = \frac{3}{2\bar{C}_s} \{s\} + \{\dot{\epsilon}_{primary}^c\} \quad (6.4.2a)$$

where

$$\{\dot{\epsilon}_{primary}^c\} = \frac{3}{2\bar{C}_p} \{s\} - \frac{\bar{K}_p}{\bar{C}_p} \{\epsilon_{primary}^c\}.$$

Then the pseudo-incremental strain in Eq. (6.4.11a) is reduced to

$$\{\Delta\epsilon'\} = 2 \left(\frac{1}{k_1} + \frac{1}{k_2} \right) \{s\} - \frac{4}{3} \frac{\bar{K}_p}{k_1} \{\epsilon_{primary}^c\}. \quad (6.4.13)$$

In the absence of plastic deformation, the total strain increment (elastic and creep) may be expressed as

$$\{\Delta\epsilon^e + \Delta\epsilon^c\} = [D_e^{-1} + D_c^{-1}] \{\Delta\sigma\} \quad (6.4.14)$$

where D_e and D_c are material matrices for elasticity and creep, respectively. However, the total strain increment must be corrected with a pseudo-incremental strain $\{\Delta\epsilon'\}$, i.e.,

$$\{\Delta\epsilon^e + \Delta\epsilon^c\} = \{\Delta\epsilon - \Delta\epsilon'\} \quad (6.4.15)$$

Combining Eqs. (6.4.14) and (6.4.15), we obtain

$$\{\Delta\sigma\} = [D_{ec}] \{\Delta\epsilon - \Delta\epsilon'\} \quad (6.4.16)$$

where the elastic-creep tangent matrix may be conveniently obtained for an isotropic material by

$$[D_{ec}] = \begin{bmatrix} K + \frac{2}{3}k_{ec} & K - \frac{1}{3}k_{ec} & K - \frac{1}{3}k_{ec} & 0 & 0 & 0 \\ & K + \frac{2}{3}k_{ec} & K - \frac{1}{3}k_{ec} & 0 & 0 & 0 \\ & & K + \frac{2}{3}k_{ec} & 0 & 0 & 0 \\ & & & \frac{1}{2}k_{ec} & 0 & 0 \\ \text{SYM} & & & & \frac{1}{2}k_{ec} & 0 \\ & & & & & \frac{1}{2}k_{ec} \end{bmatrix} \quad (6.4.17)$$

with K being a bulk modulus ($K = \frac{E}{3(1-2\nu)}$), and k_{ec} the stiffness of the elastic-creep component defined by

$$\frac{1}{k_{ec}} = \frac{1}{2G} + \frac{1}{k_c}$$

where G denotes the shear modulus.

6.4.5 Coupling of Plasticity

When the plastic deformation is involved in the creep analysis, the plastic strain increment should be included in the total strain increment. The plastic strain increment may be obtained by

$$\{\Delta\epsilon^p\} = D_p^{-1} \{\Delta\sigma\} \quad (6.4.18)$$

where

$$D_p^{-1} = \frac{1}{H} \left\{ \frac{\partial f}{\partial \sigma} \right\} \left\{ \frac{\partial f}{\partial \sigma} \right\}^T$$

with plasticity modulus $H = \frac{d\bar{\sigma}}{d\bar{\epsilon}^p}$ and the function (f) defining effective stress ($\bar{\sigma}$). Introducing Eq. (6.4.18) into Eq. (6.4.14), the elasto-plastic-creep stress-strain relations are established as

$$[D_e^{-1} + D_c^{-1} + D_p^{-1}] \{\Delta\sigma\} = \{\Delta\epsilon - \Delta\epsilon'\} \quad (6.4.19)$$

from which the stress increment $\{\Delta\sigma\}$ can be obtained. However, Eq. (6.4.19) may be rearranged as

$$\{\Delta\sigma\} = [D_{ec}] \{\Delta\epsilon - \Delta\epsilon'\} - \Delta\lambda [D_{ec}] \left\{ \frac{\partial f}{\partial \sigma} \right\} \quad (6.4.20)$$

with

$$\Delta\lambda = \frac{\left\{ \frac{\partial f}{\partial \sigma} \right\}^T [D_{ec}] \{\Delta\epsilon - \Delta\epsilon'\}}{H + \left\{ \frac{\partial f}{\partial \sigma} \right\}^T [D_{ec}] \left\{ \frac{\partial f}{\partial \sigma} \right\}} \equiv \Delta\bar{\epsilon}^p > 0$$

where the effective plastic strain increment is denoted by $\Delta\bar{\epsilon}^p$.

The creep deformation tends to relax the stress gradient in the absence of further increments in external loads. In the creep-dominant process, therefore, plastic deformation can be induced only by creep deformation to alleviate stresses in the neighboring material. For this reason, the material routine employs a solution scheme which seeks a solution in two distinct steps when the plastic deformation is coupled with creep. First, it solves for the incremental stress components with an elastic-creep material as in Eq. (6.4.16). Then, if the new stress state ($\sigma = \sigma_{old} + \Delta\sigma$) exceeds the current yield stress, i.e.,

$$F(\sigma) = f(\sigma) - \bar{\sigma}(\bar{\epsilon}^p) > 0$$

with F being a yield function, a correction is made on a previously obtained incremental stress based on Eq. (6.4.20) and the plastic strains are computed. The algorithm for this process is depicted by a flow diagram in Fig. 6.4.2.

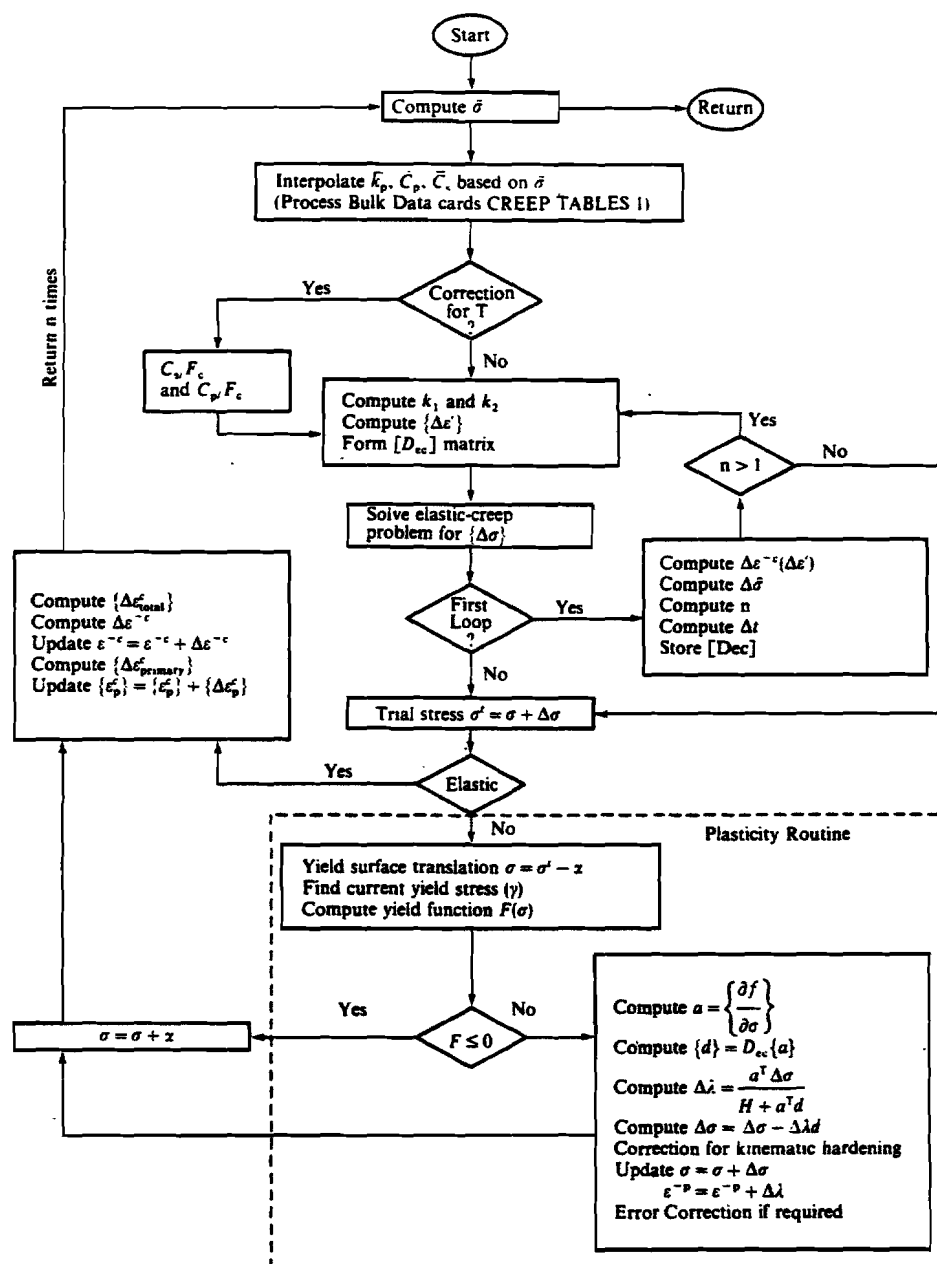


Figure 6.4.2. Flow Diagram for Creep Process

6.4.6 Implementation

The quasi-static nonlinear analysis capability for the generalized viscoelastic model coupled with the plastic deformation is implemented into the general-purpose nonlinear static analysis method in the software. The solution method employs an adaptive Newtons iteration with accelerated convergence schemes such as the BFGS updates and the line search process. A nonlinear static analysis can readily be converted to the quasi-static analysis and vice versa by adding or subtracting a few related data. All the potentially nonlinear elements (ROD, BEAM, QUAD4, TRIA3, HEXA, PENTA and TETRA) are applicable to creep analysis, but not all the nonlinear elements in the model need to be composed of creep material.

The material routines are implemented to adopt an adaptive subincremental scheme by which the computations in material routines are carried out in small increments within a global incremental step. The creep routine subdivides an incremental time step such that the effective creep strain increments and effective stress increments are bounded by a fraction of the total creep strain and effective stress, respectively. If the plastic deformation occurs during the creep process, the creep routine calls the plasticity routine. The plasticity routine again subdivides the given increment into smaller increments if the increment in the yield stress exceeds a certain fraction of the current yield stress. The fraction which defines the bounds is a user specified parameter with a default value of 0.2. By virtue of adaptive subincremental scheme within the material routines, the non-proportional loading or unloading does not present any difficulty. The non-proportional loading cases including neutral loading have been examined and validated by controlling either the stress or strain path in two-dimensional problems with the elasto-plastic material.

Users are allowed to either specify an empirical creep law or provide direct input of rheological parameter values as functions of the effective stress. If the creep behavior is prescribed by a creep law, creep law parameters are converted to rheological parameters whenever a new stress state is computed for the creep analysis. Rheological parameters are corrected for changes in temperature based on Eq. (6.4.4) if the operating temperature is different from the temperature at which creep behavior is characterized. The empirical creep laws provided in the software are in the following equation forms with user specified coefficients (a-g):

$$\epsilon^c(\sigma, t) = a\sigma^b t^d \quad (6.4.21)$$

and

$$\epsilon^c(\sigma, t) = A(\sigma)[1 - e^{-R(\sigma)t}] + K(\sigma)t \quad (6.4.22)$$

for all combinations of types 1 and 2 for A , R , and K , which are expressed as

Parameter	Type 1	Type 2
$A(\sigma)$	$a\sigma^b$	$a \text{ Exp}(b\sigma)$
$R(\sigma)$	$c \text{ Exp}(d\sigma)$	$c\sigma^d$
$K(\sigma)$	$e [\sinh(f\sigma)]^g$	$e \text{ Exp}(f\sigma)$

Conversion of creep laws in the form of Eq. (6.4.22) to the rheological model is performed by simply matching equivalent terms in Eq. (6.4.2). However, conversion of the popular and classical creep law in Eq. (6.4.21) to the rheological model is not so simple due to lack of similarity. In this case, equivalent rheological parameters are computed by matching the total creep strain and their first and second derivatives of Eqs. (6.4.1) and (6.4.21).

The material routine computes the elastic-creep tangential matrix for the formation of a global stiffness matrix, which is used for the iteration combined with the BFGS update and line searches. An equilibrium condition is achieved within the convergence tolerance at every time-increment. The numerical solution scheme employed in the creep formulation is unconditionally stable owing to the implicit method adopted for integration. However, a limit should be placed on the time increment to achieve an accurate solution. The time increment should be selected so that the strain and/or stress do not change excessively in a single step.

The total creep strain increment can be recovered by subtracting $\{\Delta\epsilon^e\}$ and $\{\Delta\epsilon^p\}$ from the total $\{\Delta\epsilon\}$ in Eq. (6.4.19), i.e.,

$$\{\Delta\epsilon_{total}^c\} = [D_c]^{-1} \{\Delta\sigma\} + \{\Delta\epsilon'\} = \frac{1}{k_c} \{\Delta s\} + \{\Delta\epsilon'\}. \quad (6.4.23)$$

The primary creep strain increment is saved in the database to take into account creep hardening/softening effects, i.e.,

$$\{\Delta\epsilon_{primary}^c\} = \frac{1}{k_1} \{\Delta s\} + \{\Delta\epsilon'_{primary}\} \quad (6.4.24)$$

where

$$\{\Delta\epsilon'_{primary}\} = \frac{2}{k_1} \{s\} - \frac{4}{3} \frac{\bar{K}_p}{k_1} \{\epsilon_{primary}^c\}.$$

6.4.7 Verification and Validation

All aspects of creep capabilities were verified with respect to the algorithm, accuracy, and programming errors with a wide variety of problems. All types of elements (1D, 2D and 3D) have been tested and known solutions have been reproduced. Various creep laws in the form of Eq. (6.4.22) and the direct input of rheological parameters reproduced analytical solutions accurately with errors less than 0.1% in the creep strain at the end of 70 steps. Creep laws in Eq. (6.4.21), however, produced a cumulative error of 15–19% in the creep strain at the end of 70 steps, because of the mismatch between the empirical formula and the rheological model.

The creep analysis capability under variable temperatures is verified using empirical creep laws recommended by ORNL. There are two different creep laws (types 111 and 121) for the same material (type 304 stainless steel) established at different temperatures (1100°F and 1200°F) as shown in Fig. 6.4.3. The creep law type 111, defining the creep behavior at 1100°F, is applied at the operating temperature of 1200°F with corrections for variable temperature.

The results are compared with the creep law type 121, which predicts proper creep behavior at 1200°F. The accuracy of the varying temperature case was found satisfactory only for a short period of time or for small temperature variations. Nevertheless this feature is considered useful and essential.

Coupling of the plastic deformation with creep is verified by reproducing the isochronous stress-strain curve for stainless steel, type 304, as shown in Fig. 6.4.4. The data points are obtained in 15 steps to the creep time of 100 hours.

The creep behavior is manifested in the relaxation process under constant strain. Figure 6.4.5 compares relaxation predictions by various methods. It is demonstrated that the current method is superior to the initial-strain method. Results from analysis fall between experimental data and the closed-form solution.

The creep response of various elements to the stress reversal was thoroughly investigated. The effects of the creep hardening/softening were exhibited properly. However, the convergence occasionally required a stiffness matrix update and a smaller time increment at the onset of stress reversal.

The solution to the creep behavior of an infinitely long thick-walled cylinder subjected to internal pressure was presented by Greenbaum and Rubinstein [6.13]. They employed an incompressible material with a simple empirical formula which accounts only for secondary creep, to which an analytical solution exists. The software reproduced the solution very accurately using nine HEXA elements with axisymmetric and plane strain boundary conditions.

The creep behavior of a thick-walled pressure vessel with a flat-end closure was also analyzed by Greenbaum and Rubinstein [6.13] under an internal pressure of 445 psi with a material obeying an empirical creep law in the form of

$$\epsilon^c = (19.8 \times 10^{-16}) (\sigma^{3.61}) (t^{1.06}) .$$

A finite element model of the same pressure vessel was analyzed by the software using 72 solid elements with 355 active DOFs. Fig. 6.4.6 shows stress contours at t=3 hours. For a model five times coarser, this solution represents an excellent agreement with that in Reference [6.13].

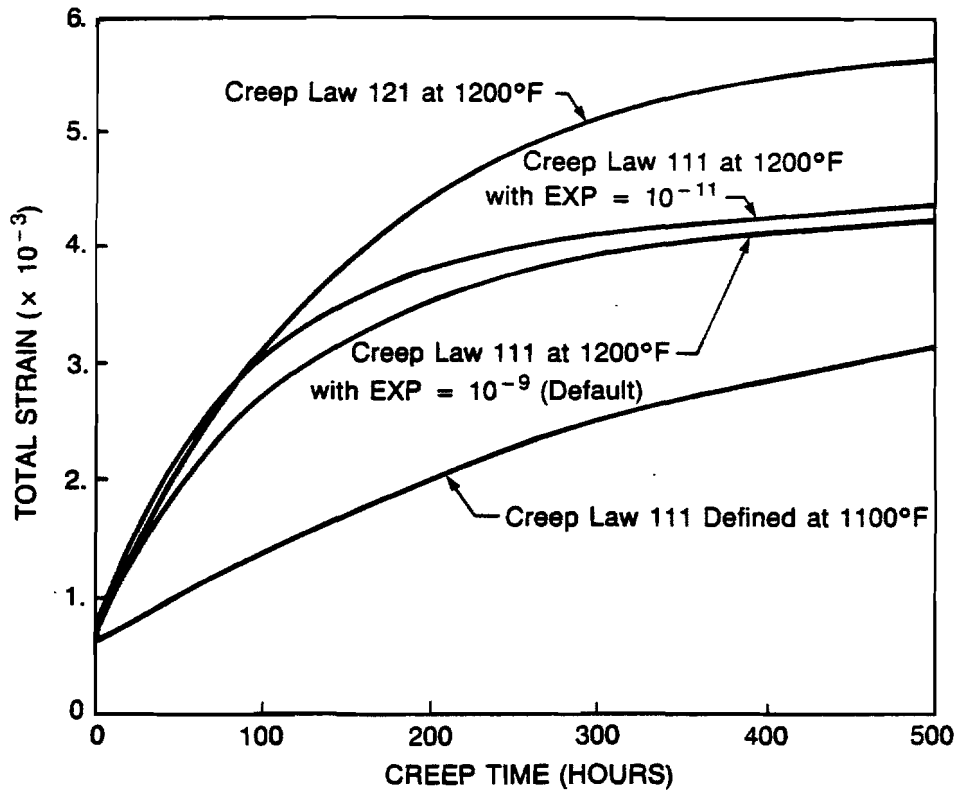


Figure 6.4.3. Creep Capability Verification for Variable Temperature

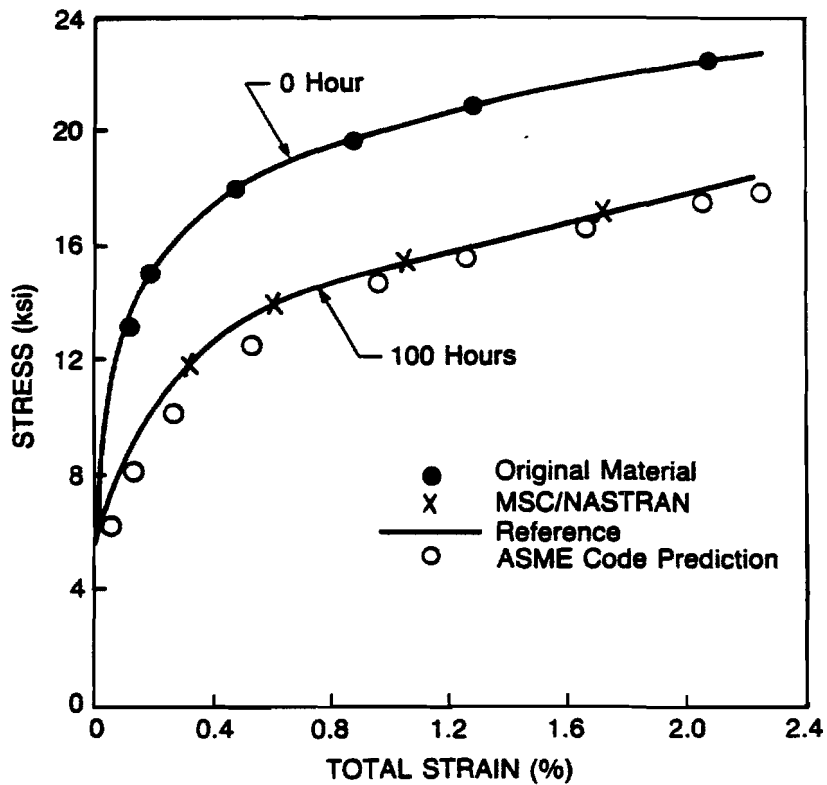


Figure 6.4.4. Isochronous Stress - Strain Curve

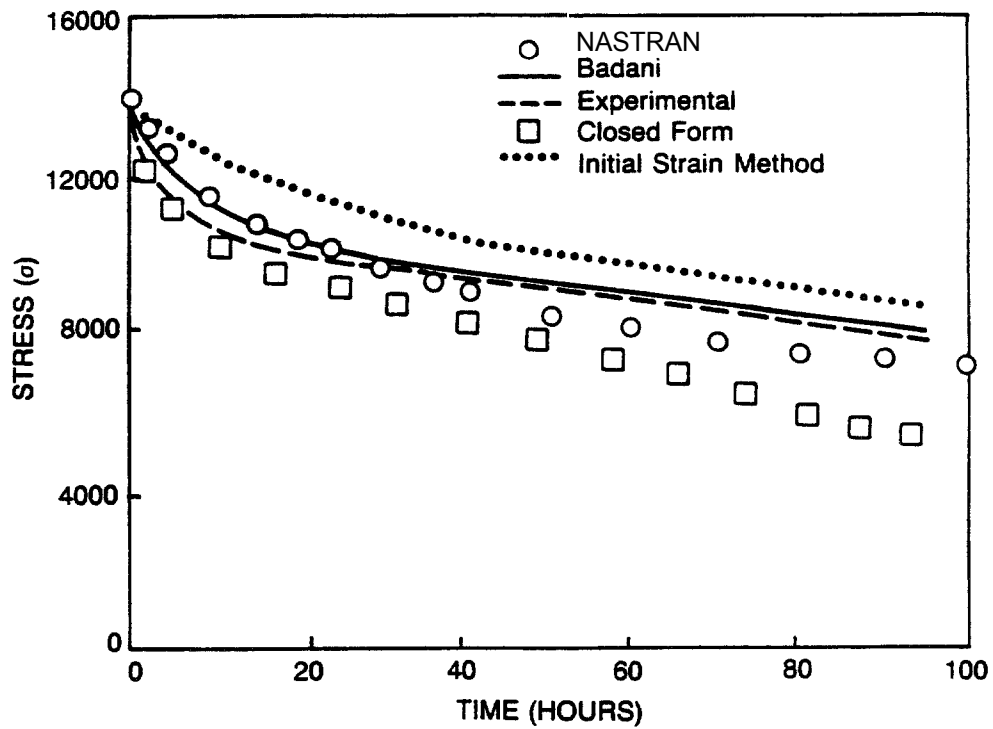


Figure 6.4.5. Comparison of Creep Relaxation Predictions

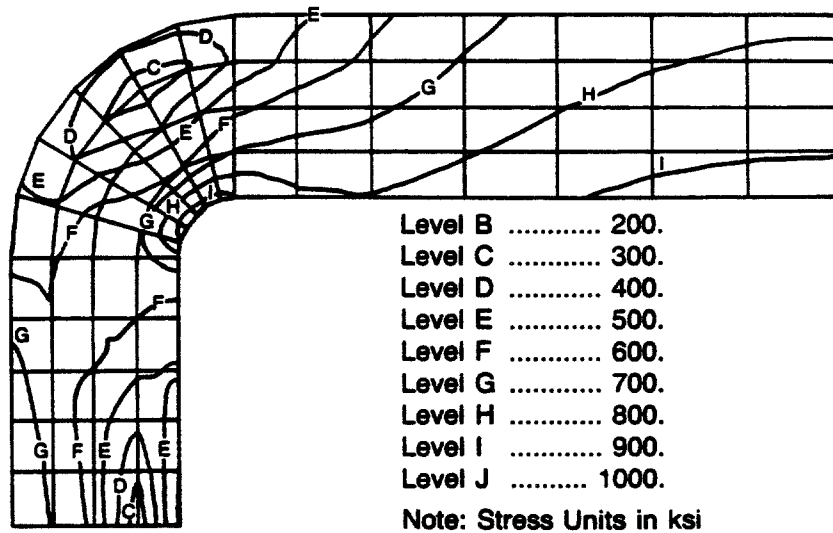


Figure 6.4.6. Effective Stress Distribution after 3 hours of Creep

6.4.8 Nozzle-to-Spherical Shell Attachment as Benchmark Problem

The High-Temperature Structural Design (HTSD) program of ORNL was instigated to develop design methodologies for breeder reactor structures. Under the ORNL program, a nozzle-to-spherical shell attachment model was presented as a benchmark problem with extensive experimental [6.14] and analytical [6.15, 6.16] results. The model represents a liquid-metal fast breeder reactor component, made of type 304 stainless steel, subjected to successive cycles of loading with internal pressure and end moment at 1100°F.

The model depicted in Fig. 6.4.7 represents this component using 1552 DOFs, which is analyzed by the software with elasto-plastic-creep material and geometric nonlinear effects. The creep characteristics are defined by a creep law in the form of Eq. (6.4.22) with ORNL recommended values for parameters as follows:

$$\begin{aligned}A(\sigma) &= 6.985 \times 10^{-6} \sigma^{2.444} \\R(\sigma) &= 7.032 \times 10^{-4} \text{Exp}(0.1072\sigma) \\K(\sigma) &= 6.73 \times 10^{-9} [\sinh(0.1479\sigma)]^{3.0}\end{aligned}$$

where the stress (σ) and the time (t) are measured in ksi and hour, respectively.

Typical results at selected points, where the plastic deformation is significant, are compared with data in References [6.14, 6.15, and 6.22] as shown in Figs. 6.4.8, 6.4.9(a) and 6.4.9(b). It is noted that the software used an isotropic, instead of a kinematic hardening rule as employed by Levy [6.15]. Figs. 6.4.10(a) and 6.4.10(b) show effective-stress contours upon **unloading** at $t=2546$ and $t=3400$ hours, respectively. These results demonstrate proficient capabilities of the software for a complex problem with combined nonlinearities.

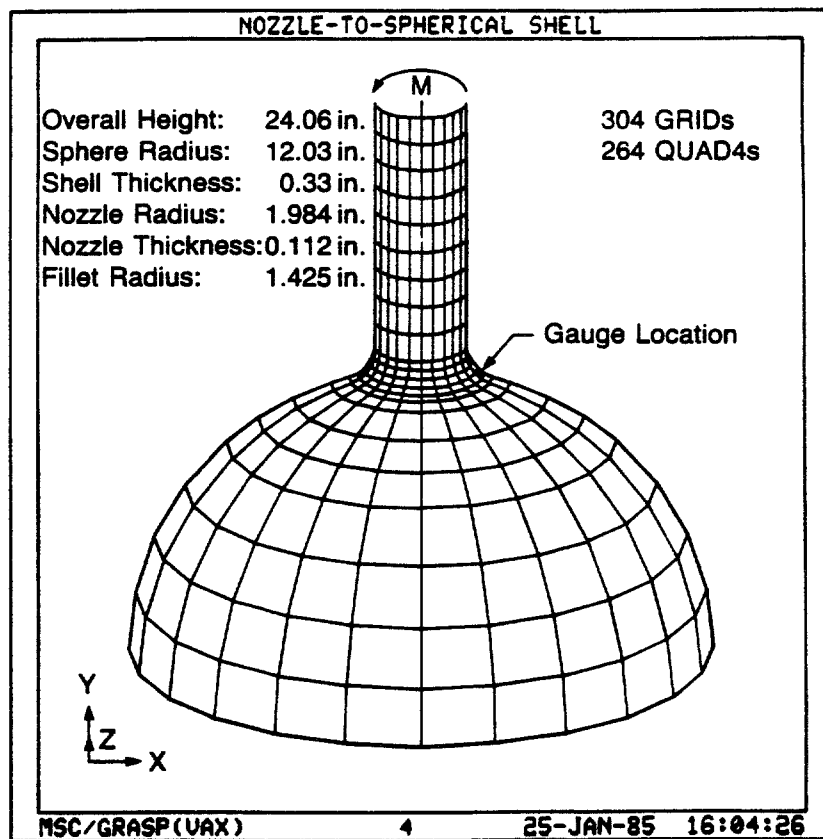


Figure 6.4.7 Finite element model

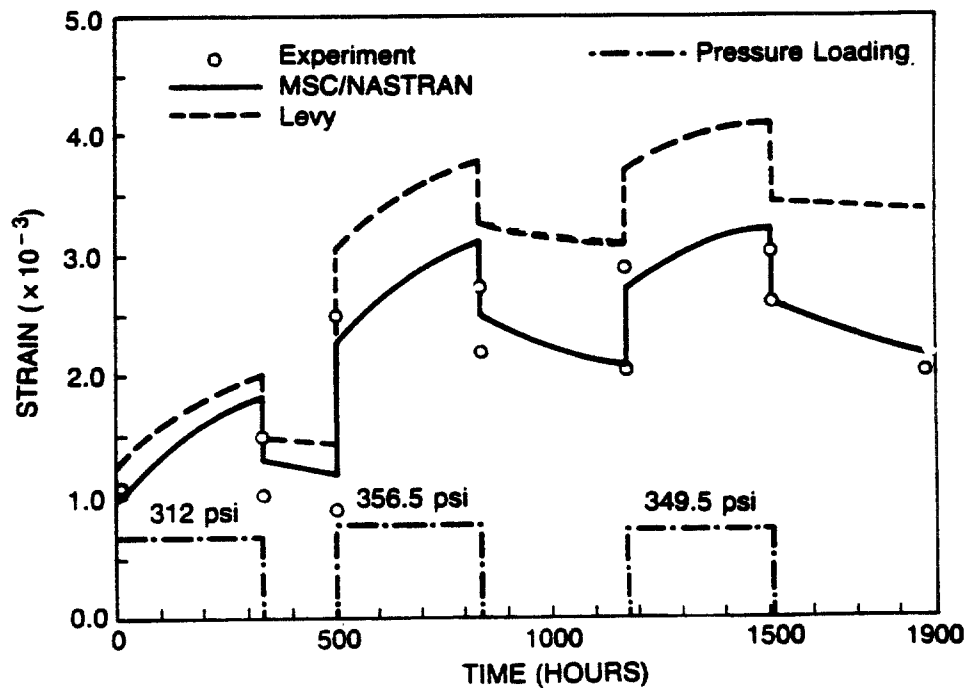


Figure 6.4.8 Total circumferential strain at gauge location during pressure loading

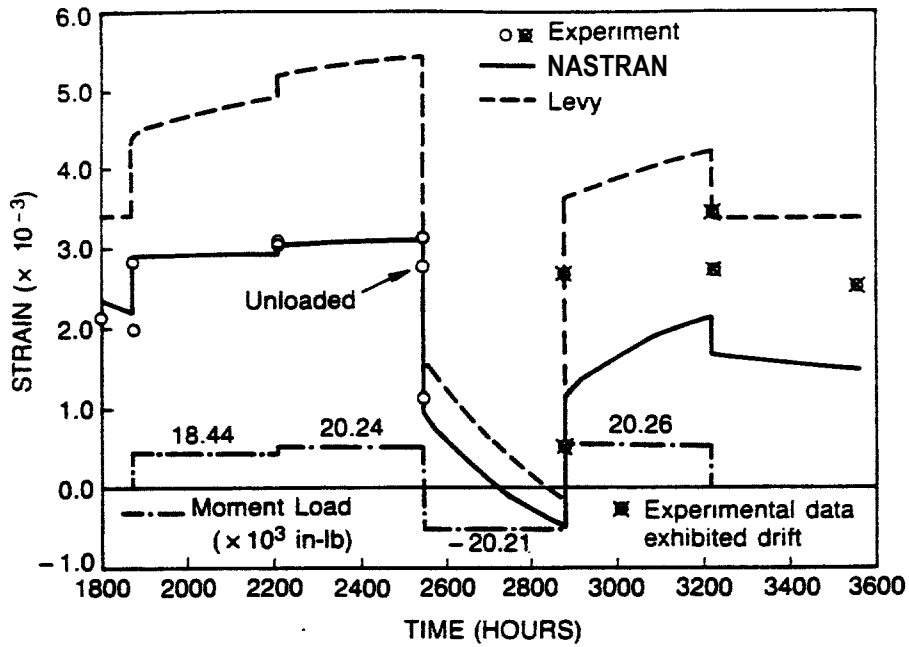


Figure 6.4.9(a) Total circumferential strain at gauge location (1) during moment loading

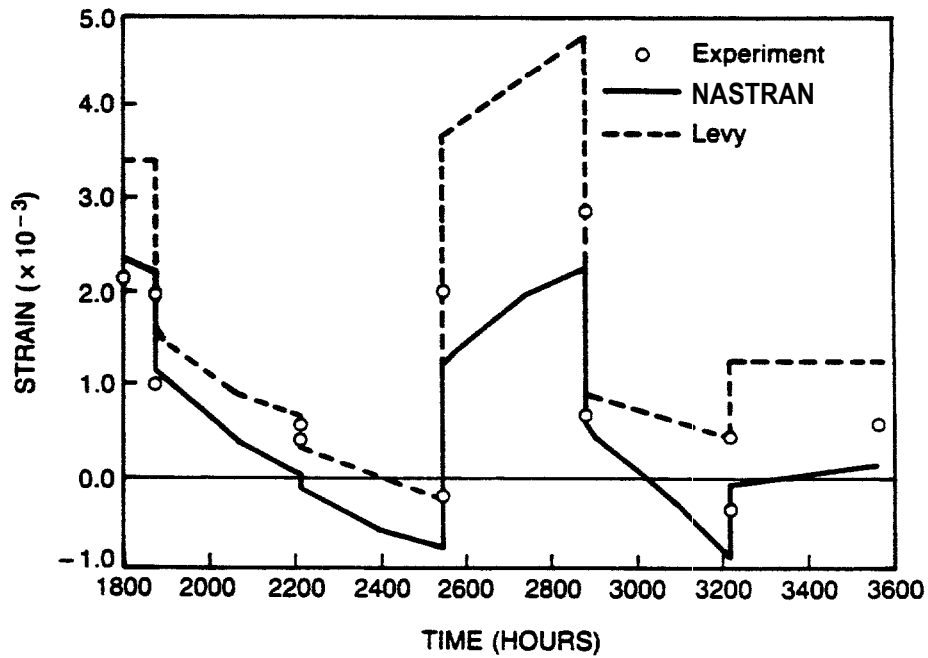


Figure 6.4.9(b) Total circumferential strain at gauge location (2) during moment loading

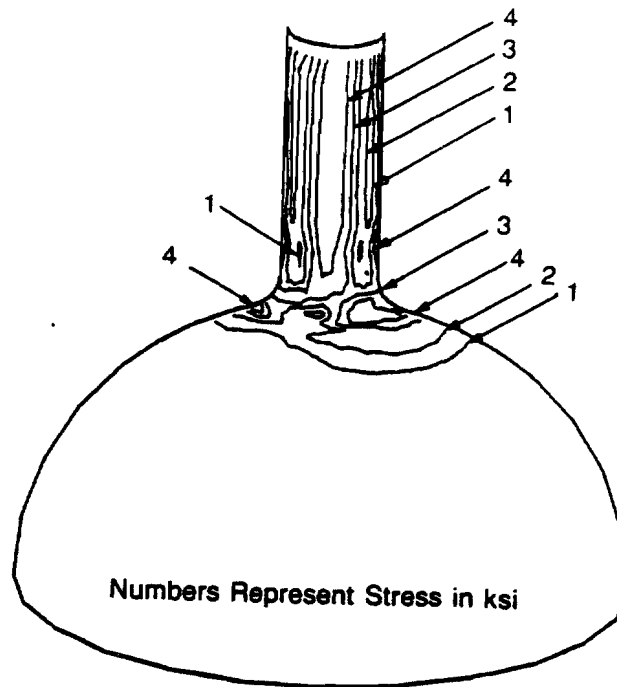


Figure 6.4.10(a) Effective stress contour plot at $t=2546$ hours upon loading

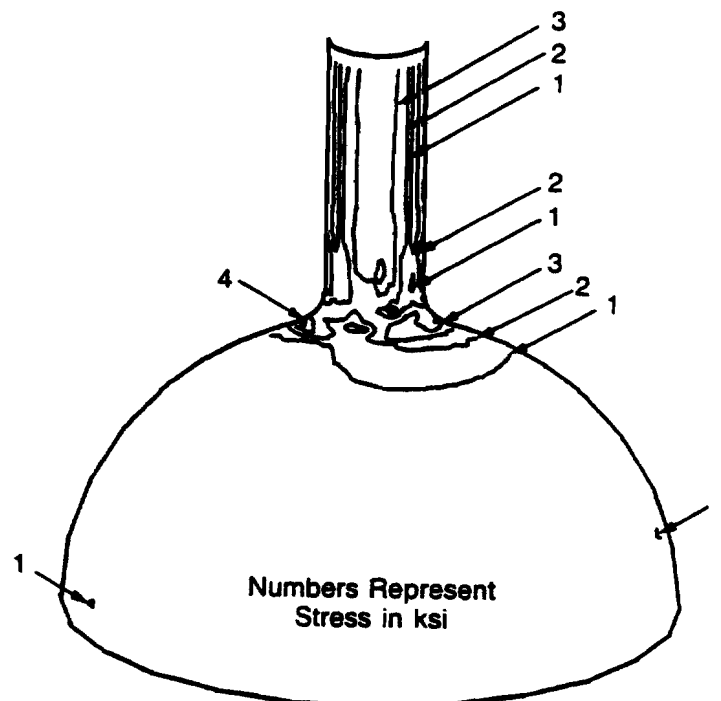


Figure 6.4.11(b) Effective stress contour plot at $t=3400$ hours upon loading

6.5 THERMO-ELASTICITY

Thermoelastic capability was implemented in SOL 66 (or SOL 106).

With this capability, the temperature dependent material properties for linear elastic and nonlinear elastic materials are updated as the temperature distribution in the structure changes.

6.5.1 Theoretical Basis

Thermo-elasticity is an extension of nonlinear elasticity as described in Section 6.3. The thermal strain is computed by

$$\epsilon^T = \alpha(T)(T - T_{REF}) - \alpha(T_0)(T_0 - T_{REF}) \quad (6.5.1)$$

where

$$\begin{aligned} T &= \text{current temperature} \\ T_{REF} &= \text{reference temperature at which coefficient thermal expansion is measured} \\ T_0 &= \text{stress free temperature (initial temperature)} \\ \alpha(T) &= \text{coefficient of thermal expansion} \end{aligned}$$

The total stress and total strain relations are modified from Eq. (6.3.4) as

$$\{\sigma\} = \frac{1}{E} \frac{\bar{\sigma}(\bar{\epsilon}, T)}{\bar{\epsilon}} [D_e(T)] \{\epsilon - \epsilon_T\}. \quad (6.5.2)$$

In the linear elastic case, Eq. (6.5.2) reduces to

$$\{\sigma\} = [D_e(T)] \{\epsilon - \epsilon_T\}. \quad (6.5.3)$$

By neglecting the rate of change of the instantaneous tangent to the uniaxial curve with respect to temperature, the tangent material stiffness matrix may be obtained by

$$[D_{ne}] = \frac{d\{\sigma\}}{d\{\epsilon\}} = \frac{1}{E} \frac{\bar{\sigma}}{\bar{\epsilon}} [D_e] + \frac{1}{(E\bar{\epsilon})^2} \left(\frac{\partial \bar{\sigma}}{\partial \bar{\epsilon}} \right) [D_e] \{\epsilon - \epsilon_T\} \{\epsilon - \epsilon_T\}^T [D_e]^T \quad (6.5.4)$$

6.5.2 User Interface

The thermal-stress free temperature distribution, T_0 , is used as a starting point for linear and nonlinear thermal elastic analyses. The set ID of this temperature distribution is specified above the Case Control section by the TEMP(INIT) command. The initial temperature, T_0 , is used in Eq. (3.5.1) and other parts of the computation, i.e.,

- To look up the material properties at temperature T_0 for linear and upstream superelements (these properties remain unchanged through the entire analysis),

- To be the default temperature distribution whenever TEMP(LOAD) is not specified for a subcase,
- To be the reference (stress-free) temperature when TEMP(LOAD) is specified.

From Version 67, there is no default value for T_0 and a fatal error message will be issued if TEMP(INIT) is not present. The set ID for the load temperature distribution, T , is specified in the Case Control by TEMP(LOAD). For nonlinear elements, the TEMP(LOAD) also specify the temperatures at which material properties are determined if Bulk Data entries MATTi are present. The material properties will conform to TEMP(LOAD) without requiring TEMP(MAT) or TEMP(BOTH). Each TEMP(INIT) or TEMP(LOAD) points to the Bulk Data TEMP, TEMPD, TEMPP1, TEMPRB on which the temperature distribution is specified. The coefficient of thermal expansion reference temperature, T_{REF} is input by the TREF field on the MATi, which is different from Version 65. Notice that the combination of TEMP(INIT) and TEMP(LOAD) is new user interface introduced in Version 66. They are recommended to be used for all types of thermo-elastic analyses available in SOL 66 and SOL 106 for Version 66 and beyond. Some old features such as TREF in MATi and TEMP(MAT) which were valid prior to Version 66 should not be used in versions following Version 65. For example, TEMP(MAT) without TEMP(INIT) would run through in Version 67. However, the material properties are determined at the very beginning of the analysis based on the temperature distribution given in the first subcase. But these properties will stay constant throughout the analysis. TEMP(MAT) may also be used in the nonlinear transient analysis (SOL 99 or SOL 129), but it can only be applied to the upstream superelements. A fatal message will be issued if TEMP(MAT) appears in a residual subcase.

Different options of thermo-elastic constitutive relations may be obtained by different combinations of input data:

- $\{\sigma\} = [D_e(T)]\{\epsilon\}$ (MAT1,MATT1)
- $\{\sigma\} = \frac{\bar{E}(\sigma,\epsilon)}{E}[D_e(T)]\{\epsilon\}$ (MAT1,MATT1,MATS1,TABLES1)
- $\{\sigma\} = \frac{\bar{E}(T,\sigma,\epsilon)}{E}[D_e(T)]\{\epsilon\}$ (MAT1, MATT1, MATS1, TABLEST, TABLES1)
- $\{\sigma\} = \frac{\bar{E}(T,\sigma,\epsilon)}{E}[D_e]\{\epsilon\}$ (MAT1,MATS1,TABLEST,TABLES1)

where $\{\sigma\}$ and $\{\epsilon\}$ are the stress and strain vectors, $[D_e(T)]$ the elasticity matrix from MATT1 data, \bar{E} the effective elasticity modulus $\frac{\bar{\sigma}}{\bar{\epsilon}}$ from TABLES1 data, and E the elasticity modulus from MAT1 data.

Temperature dependent properties for elasticity matrix, i.e. $[D_e(T)]$, are input by a combination of the MATi, MATTi, and TABLEMi entries. For the nonlinear elastic material, which is specified by the MATS1 and TABLES1, the TABLEST may be used to input the temperature dependent uniaxial stress-strain relations, i.e. $\frac{\bar{\sigma}(\epsilon,T)}{\bar{\epsilon}}$. The TABLEST is referenced by the MATS1 and contains the TID of TABLES1 cards for given temperatures. A fatal error message will be issued if both TABLES1 and TABLEST with the same TID are present. The TABLEST entry is shown below followed by brief description of the fields:

1	2	3	4	5	6	7	8	9	10
TID									
T1	I1	T2	I2	T3	I3	ENDT			

TID : Table identification number (integer > 0). This number must be identical to TID of MATS1 to be active.

Ti : Temperature values (real) which must be listed in ascending order.

Ii : Table identification number of TABLES1 entries (integer > 0). Ii must be unique with respect to all TABLES1 and TABLEST table identification numbers.

ENDT : The end-of-table flag.

Possible thermal stress analysis types and the corresponding required input cards are shown in the table below.

Types of Thermal Stress Analyses and Required Input Entries		
Type of analysis	Case Control	Bulk Data
thermal loads only, no temperature dependent material is present	TEMP(INIT) = m once TEMP(LOAD) = n each subcase	TEMP,TEMPD.. MATi
thermal loads + temperature dependent linear materials	same as above	TEMP,TEMPD.. MATi, MATTi TABLEMi
thermal loads + temperature dependent nonlinear elastic materials	same as above	TEMP,TEMPD.. MATi, MATS1 TABLEST TABLES1
no thermal load, temperature dependent material only	same as above	same as above but $\alpha=0.0$ on MATi

Notice that temperature dependent linear composite materials (MAT2 with MATT2 or MAT9 with MATT9) are available in the software, but *nonlinear elastic* composite materials (MATS1/NONLINEAR with MAT2 or MAT9) are not. Thermoplasticity is also not available in Version 67. Analysis capability for the creep (or viscoelastic) deformation under the variable temperature became available from Version 66. In addition, the creep characteristics and the elasticity properties both can be temperature dependent. The former

are specified by the CREEP entry (see Section 6.4 for more details), and the latter are input with the same procedure as described above.

6.5.3 Validation Problem

A 1-inch cube, as shown in Fig. 6.5.1, is subjected to a uniaxial tension of 10,000 lbs and a temperature change from 100°F to 200°F, assuming that the former is the stress-free temperature. The Young's modulus is 8.0×10^6 psi at 100°F and 4.0×10^6 psi at 200°F, and the Poisson's ratio is 0.3 for all temperatures. For simplicity, the thermal expansion coefficient is assumed to be 0. The structural response at 100°F and 200°F, respectively, is determined.

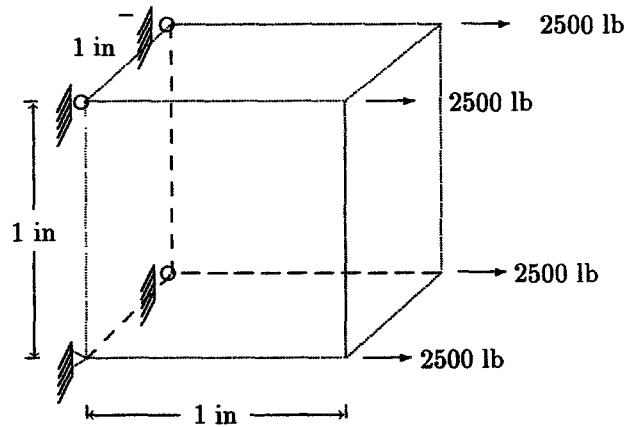


Figure 6.5.1 Cube Subjected to Uniaxial Tension

For verification purpose, two identical plate models were created and solved using different approaches as shown in the attached input data. In the first model, the reference material properties were defined on a MAT1 entry, and their variations with different temperatures were input through MAT1 and TABLE1 entries. In the second model, the reference properties were still input through a MAT1 entry, however, a set of temperature dependent stress-strain curves were specified using a combination of MATS1 and TABLES1 entries. The MAT1 entry present in the second model is a dummy, which is required in V66 and V66A to activate the thermal elastic capability. This requirement has been removed in V67 and beyond. The results given by these two models are identical as shown below.

Temperature	σ_X	σ_Y	ϵ_X	ϵ_Y
100°F	10,000 psi	0 psi	1.25×10^{-3}	-3.75×10^{-4}
200°F	10,000 psi	0 psi	2.50×10^{-3}	-7.50×10^{-4}

```

ID      THPLATE, V66A $ SSH 12/12/90
TIME    5          $ CPU MIN.
SOL      66        $ NONLINEAR STATIC SOLUTION SEQUENCE
DIAG     8,50      $ PRINT MATRIX TRAILERS AND TRACE N-L ITERATION
CEND
TITLE = 2 DISCRETE PLATE MODELS UNDER UNIAXIAL TENSION
ECHO = SORT
SEALL = ALL
  DISP      = ALL
  STRESS     = ALL
  ELFORCE    = ALL
  SPCFORCES  = ALL
  OLOAD      = ALL
TEMP(INIT)=100
SUBCASE 100
  NLPARM = 100
  LOAD   = 101
SUBCASE 200
  TEMP(LOAD) = 200
  NLPARM = 200
  LOAD   = 101
BEGIN BULK
$
$ TEMPERATURE DISTRIBUTION
$
TEMPD   100      100.    200    200.
$
$ ITERATION STRATEGY
$
NLPARM  100      1                      YES
NLPARM  200      5                      YES
$
$ FIRST MODEL
$
GRID    101      0      0.    0.    0.    123456
GRID    102      0      1.    0.    0.    3456
GRID    103      0      1.    1.    0.    3456
GRID    104      0      0.    1.    0.    13456
$
CQUAD4  101      100 101 102 103 104
$
PSHELL  100      120    1.0
$
MAT1    120      8.+6    3.0769+60.3
MATT1   120      151     152
$
TABLEM1 151
        0.      8.0000+6100.    8.0000+6200.    4.0000+6ENDT
TABLEM1 152

```

	0.	3.0769+6100.	3.0769+6200.	1.5385+6	ENDT	
\$						
FORCE	101	102	0	5.0E3	1.0	0.0 0.0
FORCE	101	103	0	5.0E3	1.0	0.0 0.0
\$						
\$	SECOND MODEL					
\$						
GRID	201	0	0.	0.	0.	123456
GRID	202	0	1.	0.	0.	3456
GRID	203	0	1.	1.	0.	3456
GRID	204	0	0.	1.	0.	13456
\$						
CQUAD4	201	200	201	202	203	204
\$						
PSHELL	200	220	1.0			
\$						
MAT1	220	8.+6	3.0769+60.3			
MATT1	220					
MATS1	220	221	NLELAST			
TABLEST	221					
+ST1	0.	251	100.	252	200.	253 ENDT
TABLES1	251					
+SS1	-1.	-8.+6	0.	0.	1.	8.+6 ENDT
TABLES1	252					
+SS2	-1.	-8.+6	0.	0.	1.	8.+6 ENDT
TABLES1	253					
+SS3	-1.	-4.+6	0.	0.	1.	4.+6 ENDT
\$						
FORCE	101	202	0	5.0E3	1.0	0.0 0.0
FORCE	101	203	0	5.0E3	1.0	0.0 0.0
ENDDATA						

6.6 INITIAL ANISOTROPY

This material capability became available in SOL 66 (or 106) and SOL 99 (or 129) since Version 66A. It allows the following combinations of material property entries:

Elastic-plastic: MAT2 or MAT9 combined with MATS1(TYPE=PLASTIC)

Thermo-elastic: MAT2 combined with MATT2 or MAT9 with MATT9

Viscoelastic: MAT2 or MAT9 combined with CREEP

Notice that MAT2 or MAT9 represent elastic anisotropy. When they are combined with plastic or creep material, they represent initial anisotropy because the plastic or creep (viscoelastic) deformation remains isotropic.

6.6.1 Theoretical Basis

The elasticity matrix $[D_e]$ appears in the incremental constitutive relations for elastic-plastic, thermoelastic, and viscoelastic materials as shown in the preceding sections.

In those constitutive relations, the elastic properties are assumed to be isotropic so that the elasticity matrix $[D_e]$ may be determined by two material constants, e.g. Young's modulus E and Poisson's ratio ν . This assumption was removed by simply replacing the isotropic $[D_e]$ by an anisotropic elasticity matrix specified by MAT2 or MAT9 Bulk Data entries. The orthotropic material (MAT8) is not applicable to this capability. This capability may be used to model nonlinear materials which exhibit significant initial anisotropy.

6.6.2 Validation Problems

A cube with dimensions of one unit was modeled with one QUAD4 element for testing the 2-D case and one HEXA element for testing the 3-D case. First, isotropic properties were input using MAT2 and MAT9 Bulk Data entries and the results were compared with the analytical solutions. Then, genuine anisotropic properties were used and the results from the plate and solid models were compared with each other.

The input data of three test decks, namely ANISOEP, ANISOC, and ANISOT, are listed at the end of this section as examples. Each deck contains two discrete models, a plate and a solid, which are subjected to a simple tension force of 10,000 lb, respectively. In-plane anisotropic elastic properties are assumed so that the plate and solid models are expected to give comparable results. ANISOEP is used to test the elastic-plastic analyses, in which the yield stress is assumed to be 5000 psi and the plastic hardening modulus to be 5×10^5 psi. In ANISOEC, two subcases are specified. The first subcase is for a static response analysis which is followed by the second subcase for a creep analysis of 100 hours. The material is assumed to remain linear elastic throughout the analysis. ANISOT is used to test the thermo-elastic

analyses. A tension force of 10,000 lb is applied in the first subcase. In the second subcase, the mechanical load remains unchanged but the temperature is increased, which reduces the value of each elasticity modulus by one half. The results of these three test problems are tabulated below:

	σ_x	ϵ_x	ϵ_y	γ_{xy}
ANISOEP :				
plate model	10,000	1.147×10^{-2}	-4.526×10^{-3}	-9.540×10^{-4}
solid model	10,000	1.147×10^{-2}	-4.526×10^{-3}	-9.550×10^{-4}
ANISOC :				
plate model	10,000	2.560×10^{-3}	-7.054×10^{-5}	-9.548×10^{-4}
solid model	10,000	2.560×10^{-3}	-7.050×10^{-5}	-9.551×10^{-4}
ANISOT :				
plate model	10,000	2.941×10^{-3}	9.486×10^{-4}	-1.910×10^{-3}
solid model	10,000	2.941×10^{-3}	9.487×10^{-4}	-1.910×10^{-3}

ID ANISOEP, V66 \$ SSH 3/25/91

TIME 5

SOL 66

CEND

\$

TITLE = MAT2 OR MAT9 COMBINED WITH MATS1(TYPE=PLASTIC).

SUBTITLE = 2 DISCRETE MODELS EACH CONTAINS 1 QUAD4 OR HEXA ELEMENT.

LABEL = UNIAXIAL TENSION CASE.

SEALL = ALL

DISP = ALL

STRESS = ALL

\$

NLPARM = 100

LOAD = 101

\$

BEGIN BULK

\$1-----2-----3-----4-----5-----6-----7-----8-----9-----10-----\$

\$

\$ ITERATION STRATEGY

\$

NLPARM 100 10 AUTO

\$

\$ PLATE MODEL

\$

GRID 201 0 0. 0. 0. 123456

GRID 202 0 1. 0. 0. 3456

GRID 203 0 1. 1. 0. 3456

GRID 204 0 0. 1. 0. 13456

CQUAD4 201 200 201 202 203 204

PSHELL 200 220 1.0

MAT2 220 7.9812E6-6.149E51.5136E64.9539E61.5136E63.0824E6

MATS1 220 PLASTIC 5.E5 5.E3

FORCE 101 202 0 5.0E3 1.0 0.0 0.0

FORCE 101 203 0 5.0E3 1.0 0.0 0.0

\$

\$ SOLID MODEL

\$

GRID 501 0 0.0 0.0 0.0 123456

GRID 502 0 1.0 0.0 0.0 3456

GRID 503 0 1.0 1.0 0.0 3456

GRID 504 0 0.0 1.0 0.0 13456

GRID 511 0 0.0 0.0 1.0 12456

GRID 512 0 1.0 0.0 1.0 456

GRID 513 0 1.0 1.0 1.0 456

GRID 514 0 0.0 1.0 1.0 1456

CHEXA 501 500 501 502 503 504 511 512 +E51

+E51 513 514

PSOLID 500 520

MAT9 520 7.9812E6-6.149E50. 1.5136E60. 0. 4.9539E6+M51

+M51	0.	1.5136E60.	0.	1.E6	0.	0.	0.	+M52
+M52	3.0824E60.	0.	3.E5	0.	3.E5			
MATS1	520		PLASTIC	5.E5			5.E3	
FORCE	101	502	0	5.0E3	0.5	0.0	0.0	
FORCE	101	503	0	5.0E3	0.5	0.0	0.0	
FORCE	101	512	0	5.0E3	0.5	0.0	0.0	
FORCE	101	513	0	5.0E3	0.5	0.0	0.0	
ENDDATA								

ID ANISOC, V66 \$ SSH 3/25/91

TIME 5

SOL 66

CEND

TITLE = MAT2 OR MAT9 COMBINED WITH CREEP

SUBTITLE = 2 DISCRETE MODELS, EACH CONTAINS ONE QUAD4 OR HEXA ELEMENT

LABEL = UNAXIAL TENSION CASE

SEALL = ALL

DISP = ALL

STRESS = ALL

\$

SUBCASE 10 \$ ELASTIC

LOAD = 1

NLPARM = 10

SUBCASE 11 \$ CREEP FOR 100 HOURS

LOAD = 1

NLPARM = 20

\$

BEGIN BULK

\$1-----2-----3-----4-----5-----6-----7-----8-----9-----10-----\$

\$

\$ ITERATION STRATEGY

\$

NLPARM 10 1 AUTO

NLPARM 20 5 20. AUTO

\$

\$ PLATE MODEL

\$

GRID 201 0 0. 0. 0. 123456

GRID 202 0 1. 0. 0. 3456

GRID 203 0 1. 1. 0. 3456

GRID 204 0 0. 1. 0. 13456

CQUAD4 201 200 201 202 203 204

PSHELL 200 220 1.0

MAT2 220 7.9812E6-6.149E51.5136E64.9539E61.5136E63.0824E6

CREEP 220 CRLAW

+CP21

+CP21 222 3.476-4 2.08-4 2.085-11 2.094 1.02-11 7.43-4

FORCE 1 202 0 0.5 1.+4 0.0 0.0

FORCE 1 203 0 0.5 1.+4 0.0 0.0

\$

\$ SOLID MODEL

\$

GRID 401 0 0. 0. 0. 123456

GRID 402 0 1. 0. 0. 3456

GRID 403 0 1. 1. 0. 3456

GRID 404 0 0. 1. 0. 13456

GRID 411 0 0. 0. 1. 12456

GRID 412 0 1. 0. 1. 456

GRID 413 0 1. 1. 1. 456

GRID	414	0	0.	1.	1.		1456		
CHEXA	401	400	401	402	403	404	411	412	+E41
+E41	413	414							
PSOLID	400	420							
MAT9	420	7.9812E6-6.149E50.			1.5136E60.		0.	4.9539E6+	M41
+M41	0.	1.5136E60.	0.		1.E6	0.	0.	0.	+M42
+M42	3.0824E60.	0.	3.E5	0.	3.E5				
CREEP	420			CRLAW					+CP41
+CP41	222	3.476-4 2.08-4	2.085-11	2.094	1.02-11	7.43-4			
FORCE	1	402	0	0.25	1.+4	0.0	0.0		
FORCE	1	403	0	0.25	1.+4	0.0	0.0		
FORCE	1	412	0	0.25	1.+4	0.0	0.0		
FORCE	1	413	0	0.25	1.+4	0.0	0.0		
ENDDATA									

```

ID  ANISOT, V66  $ SSH 3/25/91
TIME 5
SOL 66
CEND
$
TITLE      = MAT2 COMBINED WITH MATT2 AND MAT9 WITH MATT9
SUBTITLE   = 2 DISCRETE MODELS, EACH CONTAINS A QUAD4 OR HEXA ELEMENT
LABEL      = UNIAXIAL TENSION
ECHO       = SORT
SEALL      = ALL
DISP       = ALL
STRESS     = ALL
TEMP(INIT)=100
$
SUBCASE 100
  NLPARM = 100
  LOAD   = 101
SUBCASE 200
  TEMP(LOAD)=200
  NLPARM = 100
  LOAD   = 101
$
BEGIN BULK
$
TEMPD   100      100.    200    200.
NLPARM  100      1
$
$ PLATE MODEL
$
GRID    201      0       0.     0.     0.          123456
GRID    202      0       1.     0.     0.          3456
GRID    203      0       1.     1.     0.          3456
GRID    204      0       0.     1.     0.          13456
CQUAD4  201      200     201     202     203     204
PSHELL  200      220     1.0
MAT2    220      7.9812+6-6.149+51.5136+64.9539+61.5136+63.0824+6
MATT2   220      251     252     253     254     253     255
TABLEM1 255
          100.     3.0824+6200.    1.5412+6ENDT
TABLEM1 254
          100.     4.9539+6200.    2.4770+6ENDT
TABLEM1 251
          100.     7.9812+6200.    3.9906+6ENDT
TABLEM1 252
          100.    -6.149+5200.    -3.075+5ENDT
TABLEM1 253
          100.     1.5136+6200.    0.7568+6ENDT
FORCE   101      202      0       1.0E4    0.5     0.0     0.0
FORCE   101      203      0       1.0E4    0.5     0.0     0.0

```

```

$
$ SOLID MODEL
$
GRID 401 0 0.0 0.0 0.0 123456
GRID 402 0 1.0 0.0 0.0 3456
GRID 403 0 1.0 1.0 0.0 3456
GRID 404 0 0.0 1.0 0.0 13456
GRID 411 0 0.0 0.0 1.0 12456
GRID 412 0 1.0 0.0 1.0 456
GRID 413 0 1.0 1.0 1.0 456
GRID 414 0 0.0 1.0 1.0 1456
CHEXA 401 400 401 402 403 404 411 412 +E41
+E41 413 414
PSOLID 400 420
MAT9 420 7.9812E6-6.149E50. 1.5136E60. 0. 4.9539E6+M41
+M41 0. 1.5136E60. 0. 1.E6 0. 0. 0. +M42
+M42 3.0824E60. 0. 3.E5 0. 3.E5
MATT9 420 451 452 453 454 +MT41
+MT41 453 +MT42
+MT42 455
TABLEM1 455
100. 3.0824+6200. 1.5412+6ENDT
TABLEM1 454
100. 4.9539+6200. 2.4770+6ENDT
TABLEM1 451
100. 7.9812+6200. 3.9906+6ENDT
TABLEM1 452
100. -6.149+5200. -3.075+5ENDT
TABLEM1 453
100. 1.5136+6200. 0.7568+6ENDT
FORCE 101 402 0 1.0E4 0.25 0.0 0.0
FORCE 101 403 0 1.0E4 0.25 0.0 0.0
FORCE 101 412 0 1.0E4 0.25 0.0 0.0
FORCE 101 413 0 1.0E4 0.25 0.0 0.0
ENDDATA

```

Chapter 7

NONLINEAR STATIC ANALYSIS

7.1 USER INTERFACE

Nonlinear static analysis can be performed using SOLs 66 or 106. SOL 106 is a structured DMAP (utilizing the new executive system constructs) implemented in Version 67. The functionality of SOL 106 is identical to the old DMAP, SOL 66. Modeling options are compatible with linear analysis, i.e., A SOL 61 model can be converted to SOL 66 analysis with a few additional data entries relevant to the nonlinear analysis. The nonlinear properties and/or effects are defined by nonlinear material data (e.g., MATS1, CREEP and TABLES1), gap elements (GAP) for nonlinear interface, and PARAMeter LGDISP for geometric nonlinearity.

7.1.1 Case Control

Each subcase defines a set of loading and boundary conditions, which can be subdivided into a number of increments. Loading conditions are specified using the same loads as in linear static analysis, i.e., FORCEi, MOMENTi, PLOADi, GRAV, TEMP, etc. Constraints can be specified using SPCs and MPCs, which can be changed from subcase to subcase. The enforced motion can be applied using SPCD or non-zero value in SPC. The specified value for SPCD or SPC is a measure of displacement with reference to the undeformed position. The enforced motion is also subdivided in the subcase like the incremental loads. The unique data required for SOL 66 or 106 is supplied on the NLPARM data entry, which controls the incremental and iterative solution processes. The snap-through analysis or post-buckling analysis can be performed using arc-length methods which are controlled by parameters in the NLPCI Bulk Data entry.

The load step is labeled by the cumulative load factor. The load factor varies from 0 to 1 in each subcase. Then the load step will end with 1, 2, 3, etc. for the first, the second and the third subcase, respectively. There are advantages to divide the entire loading history into

many subcases so that each subcase does not have excessive number of increments, e.g., not to exceed 20 steps. The data blocks containing solutions can be generated at each increment or the end of each subcase depending on the intermediate output option specified in the INTOUT field of NLPARM data entry. Data blocks are stored in the database for output process and restarts. As such, the database size or the output quantities are better controlled with multiple subcases. Furthermore, the nonproportional loading can only be achieved by using multiple subcases.

Output requests for each subcase are processed independently. Requested output quantities for all the subcases are appended after the computational process for actual output operation. Available outputs are DISPLACEMENT, OLOAD, STRESS, ELFORCE, GRID point stresses and SPCFORCE. However, element force and GRID point stresses are available only for linear elements.

Restarts are controlled by parameters (LOOPID, SUBID and LOADINC) which can be specified either in the Case Control section or Bulk Data section. Some optional parameters (BUCKLE, LANGLE, MAXLP) are provided for additional control or capabilities. All the superelement model generation options and matrix reduction options are allowed for the linear portion of the structure.

7.1.2 Iteration Control: NLPARM Data

The most crucial data for successful nonlinear static solutions are contained in the Bulk Data NLPARM entry, which defines strategies for the incremental and iterative solution processes. It is difficult to choose the optimal combination of all the options for a specific problem. However, based on a considerable number of numerical experiments, the default option was intended to provide the best workable method for a general class of problems. Therefore, users with little insight or experience in a specific application should start with the default option. However, users with some experience and insight in a specific problem may change the default values based on the following observations:

- Computing cost for each line search is comparable to that of an iteration.
- The SEMI method usually provides better convergence than the AUTO method at the expense of higher computing cost.
- Default tolerances for the convergence criteria may be somewhat conservative. However, loose tolerances may cause difficulties in the subsequent steps.
- The quasi-Newton method is effective in most problems. However, it seems to have adverse effects in some problems, e.g., creep analysis.
- The line search method is effective to cope with difficulties in convergence in some problems. More extensive line searches may be exercised by a large value of MAXLS and/or a smaller value of LSTOL. On the other hand, line searches may have adverse effects in some problems, e.g., plane stress plasticity.

- The arc-length method should be used if the problem involves snap-through or post-buckling deformation. Then the Bulk Data entry NLPCI must be attached.

The NLPARM data format is shown below with default values wherever applicable:

1	2	3	4	5	6	7	8	9	10
NLPARM	ID	NINC	DT	KMETHOD	KSTEP	MAXITER	CONV	INTOUT	
NLPARM			0.	AUTO	5	25	PW	NO	+NP1

	EPSU	EPSP	EPSW	MAXDIV	MAXQN	MAXLS	FSTRESS	LSTOL	
+NP1	1.0E-3	1.0E-3	1.0E-7	3	MAXITER	4	0.2	0.5	+NP2

	MAXBIS				MAXR		RTOLB		
+NP2	5				20.		20.		

The ID field requires an integer to be referenced by a Case Control command NLPARM. The NINC field is an integer which specifies the number of increments to be processed in the subcase. The total load specified in the subcase minus the load specified in the preceding subcase is equally divided by this integer (NINC) to obtain the incremental load for the current subcase. Another subcase should be defined to change constraints or loading paths. However, multiple subcases may be required in the absence of any changes in constraints or loads to use a moderate value (e.g. not to exceed 20) for NINC. Use of a moderate value has advantages in controlling database size, output size, and restarts.

The DT field requires a real number specifying the time increment for each load step in the case of creep analysis. The time unit must be consistent with the unit used in the CREEP data to define the creep characteristics. The creep time increment should be determined properly for efficiency and accuracy based on the creep rate represented by the creep law to be used. The larger the creep rate, the smaller the time increment should be. No creep deformation is considered with the default value of 0. It is noted that a creep analysis should be preceded by a static analysis with at least one subcase, because the creep deformation requires stresses in the material.

Stiffness matrix update strategies are determined by a combination of the data specified in the two fields KMETHOD and KSTEP. Options for KMETHOD are AUTO, SEMI, or ITER. The KSTEP field, which is an auxiliary to the KMETHOD field, should have an integer greater than 1. With the AUTO option, the program automatically determines when to reevaluate the stiffness matrix based on the rate of convergence. At each iteration, the computing time for convergence without the stiffness matrix update is estimated and compared with the computing time for the matrix update in order to determine whether the update is more efficient. This

decision is deferred in the first two iterations after a new stiffness is obtained. If the solution tends to diverge, however, the update decision will be made effective immediately. The stiffness matrix will be updated upon convergence if the number of iterations required for convergence is greater than KSTEP. The SEMI option is identical to the AUTO option except for one additional stiffness update after the first iteration which always occurs unless the solution converges in a single iteration. With the ITER option, the stiffness matrix is updated at every KSTEP iterations. Thus, the full Newton-Raphson iteration is exercised if KSTEP is 1. The ITER method also allows a stiffness matrix update upon convergence (since Version 66) when $KSTEP \leq MAXITER$. If $KSTEP > MAXITER$, the stiffness will never be updated. It is noted that the original modified Newton's method is selected when $KSTEP = MAXITER$.

The MAXITER field is an integer representing the number of iterations allowed for each load increment. If the number of iterations reaches MAXITER without convergence, the load increment is bisected and the analysis is repeated. If the load increment cannot be bisected (i.e., MAXBIS is reached or MAXBIS=0) and MAXDIV is positive, the best attainable solution is computed and the analysis is continued to the next load increment. If MAXDIV is negative, the analysis is terminated.

The convergence test is performed at every iteration with the criteria specified in the CONV field. Any combination of U(for displacement), P(for load), and W(for work) may be specified. All the specified criteria must be satisfied to achieve convergence, except for an absolute convergence condition, under which the solution is converged regardless of criteria. This subject will be elaborated in the following section. The convergence tolerances are specified in the fields EPSU, EPSP, and EPSW for U, P, and W criteria, respectively.

The INTOUT field requires a specification of YES, NO or ALL. If YES is specified, the output requests (DISPLACEMENT, ELFORCE, STRESS, etc.), specified in the Case Control Data, will be processed for every computed load step in the subcase. If NO is specified, the output requests will be processed only for the last load step of the subcase. If ALL is specified, the output requests are processed for every computed and user specified load increment. For Newton's iteration methods (i.e., without NLPCI), the option ALL is equivalent to the option YES, since the computed load increment is equal to the user specified load increment. For arc-length methods (i.e., NLPCI command is specified), the computed load increment in general is not the same as the user specified load increment, and is not known in advance. The option ALL allows the user to obtain solutions at the desired intermediate load increments. The INTOUT data also affects the database if the PARAMeter SMALLDB (valid only in Version 65) has a value of 1. With a small database option, the database will retain only those data blocks corresponding to the incremental load steps for which output is requested by an INTOUT field. Then, the restarts will be restricted to those starting from these steps. However, if the job is terminated with an incomplete subcase for some reason (e.g. diverging, time expiration, etc.), data blocks for all the interim steps of that subcase will be stored in the database to allow flexible restarts.

The MAXDIV field requires an integer to specify a limit on the probable divergence conditions allowed for each iteration to continue. There are two classifications in the divergence

condition: probable and absolute. The absolute divergence is treated as two occurrences of the probable divergence. When the probable divergence occurs MAXDIV times or more and if MAXDIV is positive, the current solution base (displacements) is retracted and the stiffness matrix is updated in reference to the preceding iteration step. If the solution diverges again in the same load increment while MAXDIV is positive, the best attainable solution is computed and the analysis is continued to the next load increment. If MAXDIV is negative, the analysis is terminated on second divergence. Refer to Section 3.3.2 for a detailed description of divergence criteria.

The MAXQN field requires an integer to specify the maximum number of quasi-Newton vectors to be saved in the database. Quasi-Newton vectors are accumulated, if desired, until MAXQN is reached. If the number of accumulated QN vectors is greater than or close to MAXQN upon convergence, the stiffness matrix will be updated before the next increment is processed. All the QN vectors already accumulated are purged upon a stiffness update and the accumulation is re-initiated. The quasi-Newton update may be suppressed with a value of 0 for MAXQN. Details of the quasi-Newton method will be presented in the following section.

The line search is controlled by the fields MAXLS and LSTOL. The maximum number of line searches allowed for each iteration is specified in the MAXLS field. The line search process may be suppressed with a value of 0 for MAXLS. The LSTOL field requires a real number between 0.01 and 0.9 to specify a tolerance for the line search operation. The line search operation will be conducted if the error defining the divergence rate is greater than LSTOL. If the line search convergence is not achieved, the line search continues until the number of line searches reaches MAXLS.

The FSTRESS field requires a real number ($0.0 < \text{FSTRESS} < 1.0$), representing a fraction of the effective stress ($\bar{\sigma}$) used to limit the subincrement size in the material routines. The number of subincrements in the material routines for elasto-plastic or creep deformation processes is determined such that a subincrement in the effective stress is approximately $\text{FSTRESS} \cdot \bar{\sigma}$. This data is also used to establish an error tolerance in the yield function to alleviate accumulation and propagation of the error. If the error in the yield function exceeds $\text{FSTRESS} \cdot \bar{\sigma}$ at the converging state, the program will terminate the job with a fatal error message "ERROR EXCEEDED 20% OF CURRENT YIELD STRESS.." If the bisection option is selected, bisection will be activated under this condition.

The MAXBIS field requires an integer to specify the number of bisections allowed for a load or arc-length increment. When the solution diverges, different actions are taken depending on the sign of MAXBIS. If MAXBIS is positive, the stiffness matrix is updated on the first divergence and the load increment is bisected on the second divergence. If MAXBIS is negative, the load increment is bisected every time the solution diverges until the limit on bisection is reached. If the solution does not converge after $|\text{MAXBIS}|$ bisections, the analysis is continued or terminated depending on the sign of MAXDIV.

The MAXR field requires a real number to specify the bounds on the ratio of the adjusted arc-length increment relative to the initial value. In the adaptive load/arc-length increment,

the overall upper and lower bounds on the load/arc-length increment in the subcase are defined as

$$\frac{1}{\text{MAXR}} \leq \frac{\Delta l_n}{\Delta l_0} \leq \text{MAXR}$$

where Δl_n is the arc-length at step n and Δl_0 is the original arc-length. The arc-length method for load increments is selected by the NLPCI Bulk Data entry, which must have the same ID as the NLPARM Bulk Data entry.

The RTOLB field requires a real number to specify the incremental rotation (in degrees) allowed per iteration. The bisection is activated if the incremental rotation for any degree-of-freedom ($\Delta\theta_x$, $\Delta\theta_y$ or $\Delta\theta_z$) exceeds the value specified for RTOLB. This bisection strategy based on the incremental rotation is controlled by the MAXBIS field.

7.1.3 User Interface for Arc-Length Methods: NLPCI

Most of the parameters in the NLPARM Bulk Data entry are used to control the iteration strategy for arc-length methods. However, MAXLS is not applicable because the line search procedure is not coupled with arc-length methods currently. The parameters which are applicable only to the arc-length methods are specified in the Bulk Data entry NLPCI, in connection with the NLPARM data by the same ID. The NLPCI entry is shown below with default values, followed by brief description of the fields:

1	2	3	4	5	6	7	8	9	10
NLPCI	ID	TYPE	MINALR	MAXALR	SCALE		DESITER	MXINC	
NLPCI		CRIS	0.25	4.	0.		7	20	

ID : must be identical to ID of NLPARM to be active.

TYPE : Constraint type (CRIS for Crisfield, RIKS for Riks, or MRIKS for modified Riks method)

MINALR, MAXALR : For variable arc-length method, the arc-length is bounded by

$$\text{MINALR} \leq \frac{\Delta \ell_{\text{new}}}{\Delta \ell_{\text{old}}} \leq \text{MAXALR}$$

where $\text{MINALR} \leq 1$. and $\text{MAXALR} \geq 1$. The automatic adjustment is not activated if $\text{MAXALR}=\text{MINALR}=1$.

SCALE : Scaling factor for the magnitude of the load term in arc length. Real $\geq 0.$, Default = 0. The constraint equation has a disparity in the dimension by mixing the displacements with the load factor. The scaling factor (w) is introduced as user input so that the user can make constraint equation unit-independent by a proper scaling of the load factor, μ . As the value of w is increased, the constraint equation is gradually dominated by the load term. In the limiting case of infinite w , the arc-length method would be degenerated to the conventional Newton's method.

DESITER : Desired number of iterations for convergence to be used for adaptive arc-length adjustment.

$$\Delta \ell_{new} = \Delta \ell_{old} \sqrt{\frac{I_d}{I_{max}}}$$

MXINC : Maximum number of controlled increments allowed within the specified load step. There can be cases where the structure never snaps back and collapses to failure. In these cases, the analysis cannot be completed without bounding the maximum number of increments. For such cases, this value should be confined to a reasonable limit, e.g., less than 100, to prevent inadvertent depletion of the computing resources.

7.1.4 Iteration Related Output Data

At the end of every iteration, the relevant data from the iteration process are printed under the following heading:

ITERATION	: iteration count i
EUI	: relative error in terms of displacements (See Section 3.6.3)
EPI	: relative error in terms of loads (See Section 3.6.3)
EWI	: relative error in terms of work (See Section 3.6.3)
LAMBDA	: rate of convergence (defined in Section 3.3.1)
DLMAG	: absolute norm of the load error vector
FACTOR	: final value of the line search parameter (See Section 3.4)
E-FIRST	: divergence rate, initial error before line search (See Section 3.4.3)
E-FINAL	: error at the end of line search (See Section 3.4.3)
N-QNV	: number of quasi-Newton vectors appended (See Section 3.5.3)
N-LS	: number of line searches performed during the iteration
ENIC	: expected number of iterations for convergence (See Section 3.3.1)

- NDV** : number of occurrences of probable divergence during the iteration (See Section 3.3.2)
- MDV** : number of occurrences of bisection conditions in the material routine (excessive stress increment) or in the rotation angle (excessive rotation) during the iteration

7.2 ITERATION MODULE AND SOLUTION SEQUENCE

In the heart of the nonlinear static solution sequences 66 or 106, there is a nonlinear module which performs incremental and iterative processes. The solution sequence for nonlinear static analysis (SOL 66) is depicted by flow diagrams in Figs. 7.2.1 and 7.2.2. The module, NLITER, has evolved for a decade since Version 61. The DMAP interface, input/output data blocks and parameters as of Version 67 are described below.

7.2.1 DMAP Interface of NLITER Module (Version 67)

NLITER CASES, CNVTST, PLMAT, YSMAT, KAAL, ELDATA,
 KELMNL, LLLT, GMNL, MPT, DIT, MJJ, SLT1, CSTMS,
 BGPPTS, SILS, USET / UGVMO, FGNL, ESTNLH,
 CIDATA, QNV, FFGVH, *MUGV, MESTNL, DUGVMO* /
 S,N,LOADFAC/ S,N,CONV=0/ S,N,RSTEP/ S,N,NEWP=+1/
 S,N,NEWK=+1/ S,N,POUTF/ S,N,NSKIP/ V,Y,LGDISP/
 S,N,MPTSIZE/ S,N,ITERID/ S,N,KMATUP/ S,N,LSTEP/
 S,N,KTIME/ S,N,SOLCUR/ V,Y,TABS/ *S,N,KFLAG=+1/*
 S,N,NBIS=0/ S,N,NBISO=0/ S,N,CURMU/
 S,N,DELMU=1./ S,N,ARCLG=0./ S,N,ARCSIGN=1/
 S,N,TWODIV=0/ V,N,LANGLES \$

Note: New data in Version 67 are highlighted in *italics*.

7.2.2 Input Data Blocks

CASES Case Control data (Case Control subset for residual).

CNVTST Convergence test vectors : $\Delta P_g, P_g + Q_g$, diag(KLLT).

PLMAT Load vectors for subcase (initial and final), ℓ -size.

YSMAT Constraint displacement vectors for subcase (initial and final), s -size.

KAAL Reduced element stiffness matrix (linear element only).

ELDATA Element combination data, consisting of element thermal loads and element connectivity (used in SDRNL).

KELMNL Nonlinear element stiffness matrix.

LLLT Lower Triangular Factor for nonlinear elements (material and differential if positive definite, else only material stiffness unless the arc-length methods).

GMNL Multipoint constraint equation matrix.

MPT Material Properties Table (NLPARM and NLPCI data included).

DIT Direct Input Table.

MJJ Mass matrix (g-size).

SLT1 Static Load Table for follower forces.

CSTMS Coordinate System Transformation Table.

BGPDTS Basic Grid Point Definition Table.

SILS Scalar Index List.

USET Displacement Set Definition Table.

Note: The following data blocks may be purged – YSMAT (on first solution), GMNL, DIT, SLT1, CSTMS, BGPDTS.

7.2.3 Output Data Blocks

UGVMO Input/output, displacement vector in g-set at converged step.

FGNL Nonlinear element force from last iteration.

ESTNLH Input/output, updated table of ESTNL at converged step.

CIDATA Input/output, miscellaneous data for arc-length methods.

SECALG	Special root-selection flag for the false reversal case.
FULL	Constraint status, currently used only with modified Riks. 1 signifies that the solution is on the constraint; 2 that it is off the constraint.
CURDEL	The current value of the arc length.
ORGDEL	The value of the arc length at the beginning of the subcase.
NEWMU	The current value of μ .
DMUAST	If nonzero, the amount by which the projected value of μ exceeds the specified output value of μ .
DENOML	For the Riks method, the last denominator in the equation for the computation of $\Delta\mu$. For the other methods, DENOML is set to zero.

$\{\Delta u_p^1\}$ for the Riks method

$\{u^1 - u^0\}$, (normalized), used for the second Crisfield root-selection algorithm

No data store for the modified Riks method

QNV Quasi-Newton vector file.

FFGVH Follower force vector for OLOAD output.

MUGV Displacement vector for K-update.

MESTNL Input/output, nonlinear Element Summary Table for K-update.

DUGVMO Input/output, incremental displacement vector in g-set between converged steps ($U_n - U_{n-1}$), to be stored at the output steps.

7.2.4 Parameters

LOADFAC Input/output-complex. Load factor for current module iteration, fractional value between 0 to 1.

CONV Input/output-integer. On Input : Last iteration value (0 for initial iteration). On Output : -1 if solution did not converge; +1 if solution converged.

RSTEP Input/output-integer. The count of arc-length increments.

NEWP Input/output-integer. +1 for a new subcase, -1 otherwise.

NEWK Input/output-integer. On Output : -1 if old stiffness matrix to be retained; ≥ 1 if new stiffness matrix required.

POUTF Output-integer. On Output : -1 if stresses are not to be printed; +1 if stresses are to be output. (material nonlinear elements)

NSKIP Input/output-integer. The number of records to skip into the Case Control record (subcase ID). The value is set to -2 upon NOGO flag

LGDISP Input-integer. Problem nonlinear type flag. On Input : +1 for geometric nonlinear problem. -1 otherwise.

MPTSIZE Input/output-integer-default 0. (No longer used in Version 67).

ITERID Input/output-integer. Iteration count.

KMATUP Input/output-integer. The current count of the stiffness matrix updates within the increment.

LSTEP Input/output-integer. The value of the load step within a subcase. The value is assumed to be one at the start of a subcase and is incremented by one each time the module converges. When $LSTEP = NINC$ from the NLPARM card, NEWP is set +1.

KTIME Input/output-integer. The time-to-go value at the start of the stiffness matrix formulation loop within the DMAP. On output from the module, a negative value is stored, denoting the increment of time between stiffness computation and module start.

SOLCUR Input/output-integer. The DMAP loop number at the current iteration step.

TABS Input-real. Absolute temperature conversion constant for creep analysis.

KFLAG Input/output-integer. Flag to invoke the K update at last converged step. Initialized as +1 for the cold start and -1 for the restart in the DMAP. In the module it will be set to +1 when the iteration is converged and K is updated. Set to -1 when the iteration is converged but K is not updated, or K is updated with CONV=-1.

NBIS Input/output-integer. The current count of the bisection. The value is set to 0 at the start of each user specified load step.

NBISO Input/output-integer. The previous count of the bisection. The value is set to 0 at the start of each user specified load step and is set equal to NBIS when the solution converges.

CURMU Input/output-real. Current value of the load factor in the subcase. The value is set to 0 at the beginning of the subcase.

DELMU Input/output-real. The ratio of current load increment to the user specified load increment. The value is set to 1 at the start of each user specified load step.

ARCLG Input/output-real. The arc-length at converged step, to be stored in the database.

ARCSIGN Input/output-integer. The sign of $\{P\}^T \{\Delta P\}$ at the beginning of the subcase. This is used in restarts in the post-buckling region.

TWODIV Input/output-integer. Number of diverging solutions in the same load step. If this number reaches two, the solution process is terminated.

LANGLES Input-integer. Used to select the rotation vector approach to large rotations by a value of 2. The default (= 1) option is the gimbal angle approach.

7.2.5 Diagnostic Outputs

Diagnostic output is available in SOL 66 (or 106) if DIAG 50 or 51 is specified in the Executive Control Section. The data that may be produced are listed below. When DIAG 50 is requested, only the data marked with a plus (+) are printed. When DIAG 51 is selected, all the data are printed at each iteration, allowing the user to follow the nonlinear iteration calculations step by step. Because of the volume of output produced, the use of DIAG 51 is recommended only for small test problems. DIAG 51 is intended for debugging purposes, and users in general are cautioned against its use.

The data marked with an asterisk (*) are used only for the arc-length method whereas the data marked with a bullet (•) are used only for Newton's method.

For each entry into NLITER:

- + Subcase status data
- + NLPARM/NLPCI data

- Core statistics (ICORE, etc.)
- Problem statistics (g-size, etc.)
- File control block
- Input file status
- External load increment for subcase: ΔP
- * Displacement increment: Δu_l^P
- * + Initial arc-length
- Initial nonlinear force vector: F_g
- Initial sum of nonlinear forces including follower forces: F_l
- Initial displacement vector: u_l
- KFSNL*DELYS: $[k_{fs}]^T \{\Delta u_s\}$
- Initial unbalanced load: R_l
- * Cumulative displacement increment: $\{u_{n+1} - u_n\}$
- * Dot product determining sign: $\{\Delta u_l^P\}^T \{u_{n+1} - u_n\}$
- * Projected NEWMU: μ^{i+1}

For each iteration:

- Displacement increment: Δu_l
- Initial energy
- New displacement vector: u_g
- Nonlinear force vector: F_g
- Sum of nonlinear forces including follower forces: F_l
- New displacement vector: u_l
- New unbalanced load: R_l
- Denominator of EUI
- Denominator of EPI
- Contraction factor: q
- Remaining time
- * Displacement increment: Δu_l^R
- * Arc-length ratio
- * $d\mu$ calculations

For each quasi-Newton vector set:

Condition number: λ^2
quasi-Newton vector: δ
quasi-Newton vector: γ
Energy error: $z = \frac{1}{\delta_j^T \gamma_j}$

For each line search:

- Previous line search factor: α_k
- Previous error: E_k
- New line search factor: α_{k+1}

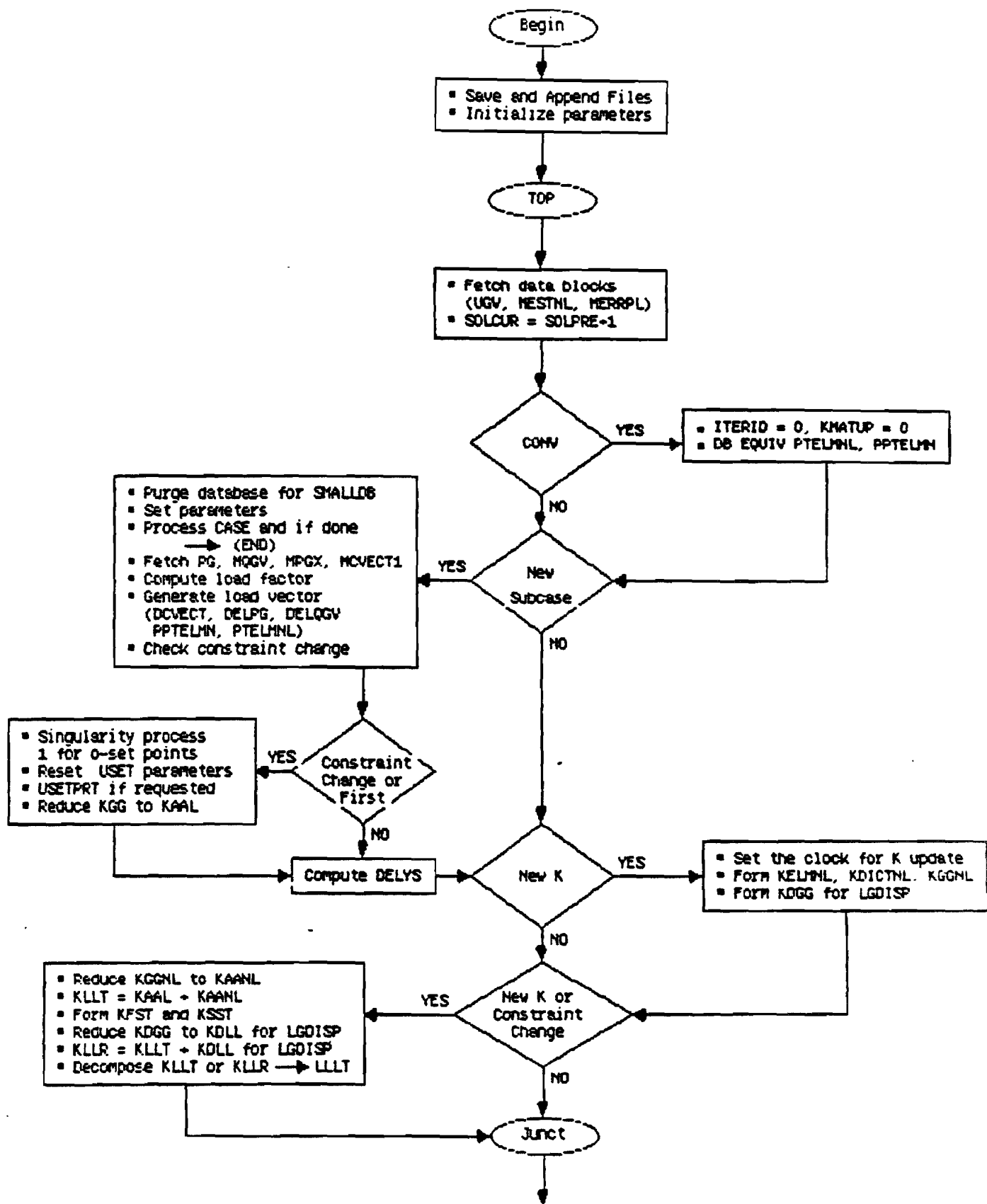


Figure 7.2.1a Flow Diagram for Nonlinear Process in SOL 66

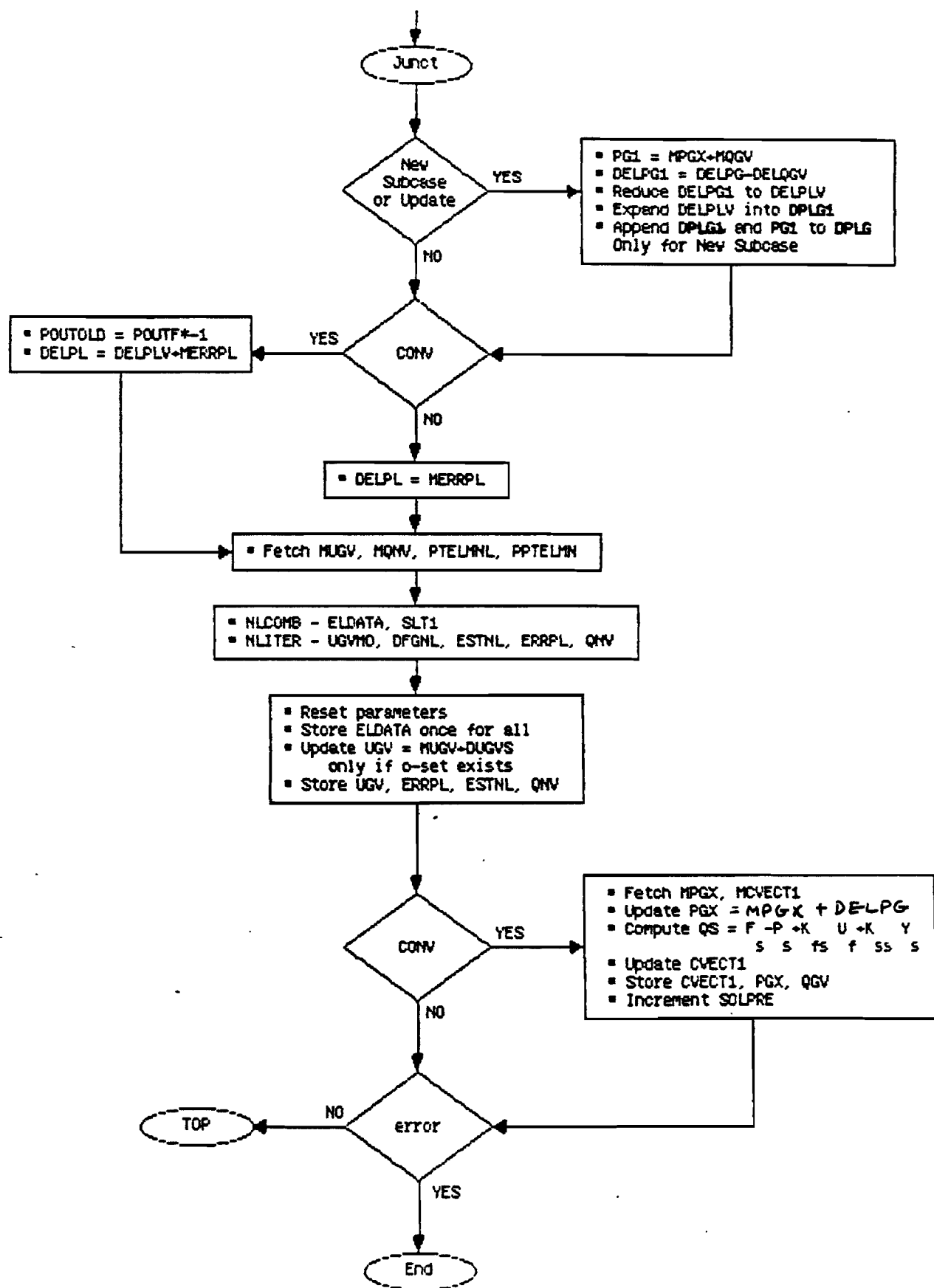


Figure 7.2.1b Flow Diagram for Nonlinear Process in SOL 66

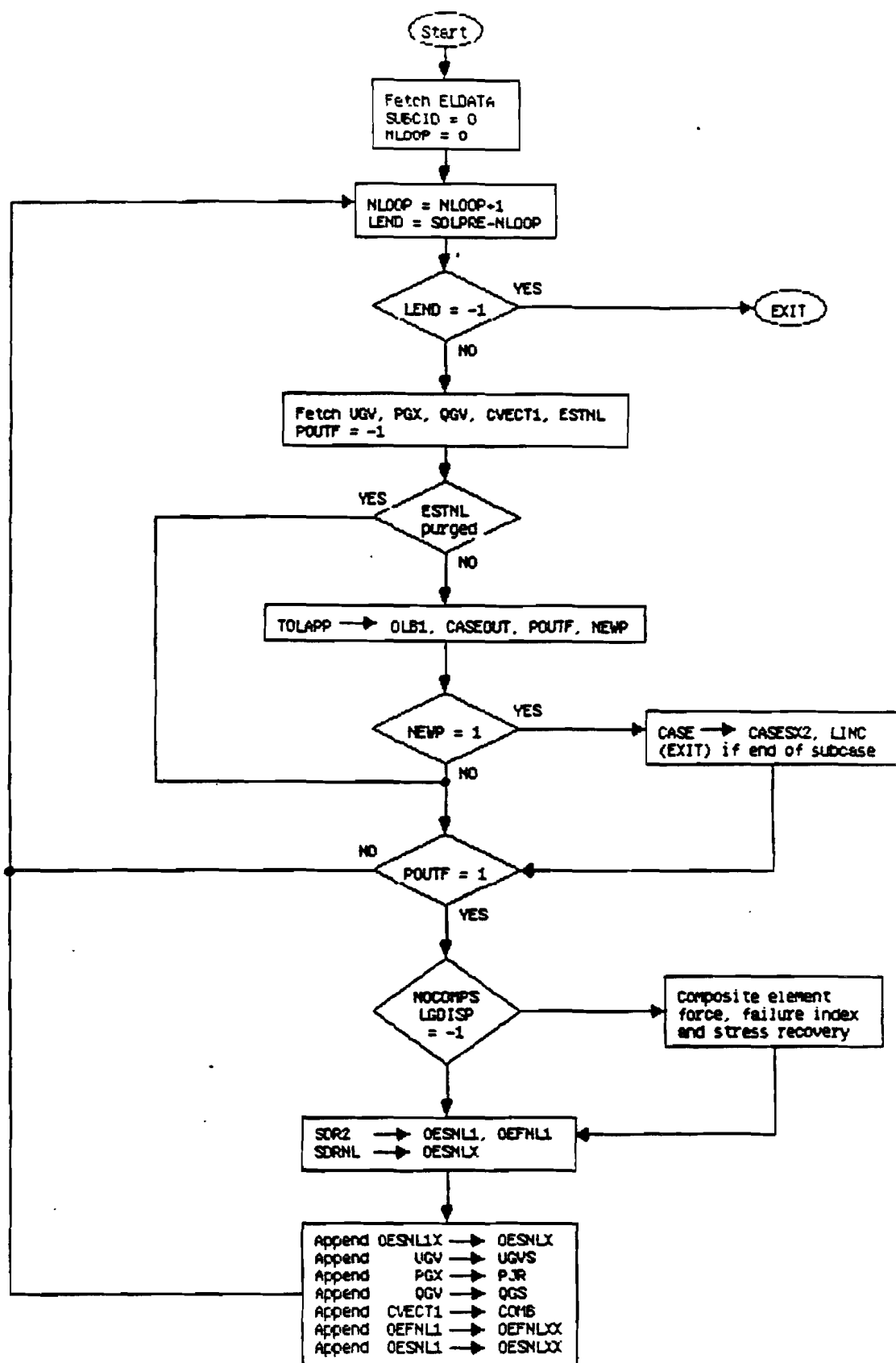


Figure 7.2.2 Flow Diagram for Appending Process in SOL 66

7.3 LOADS OVERVIEW

The static loads in nonlinear analysis consist of a subset of the total static load set. Most of the relevant loads data applicable to the linear static analysis are also applicable to nonlinear static analysis except for the Bulk Data entry DEFORM. The enforced displacements can be applied as loads using the Bulk Data entry SPCD, which defines an enforced motion of a grid in a specific fixed direction. Each grid point with an enforced displacement must also appear on an SPC or SPC1 entry.

The loads are selected in each subcase by a Case Control command LOAD. The load for a subcase is subdivided into the number of increments specified for the subcase. The solution strategy in nonlinear is to apply the loads in an incremental fashion until the desired load level is reached. The algorithms “remember” the loads from one subcase to the next. If the load reaches the desired level in a subcase and if the load description is left out in the subsequent subcase inadvertently, the nonlinear solution algorithm will begin in an incremental fashion to remove the load. The loads described below make up the valid subset applicable for nonlinear static analysis.

7.3.1 Concentrated Loads

The concentrated loads consist of two categories: those which are stationary in direction through out the analysis and those which follow the grid motion. The first category consists of the following:

FORCE, MOMENT, and SLOAD.

The second category consists of the following:

FORCE1, FORCE2, MOMENT1, and MOMENT2.

The three FORCE entries differ only in the manner in which the direction of the force is specified. FORCE uses the components of a vector. FORCE1 uses two grid points, not necessarily the same as the loaded grid points. FORCE2 defines the direction of the force as a vector which is the vector product of two other vectors. The distinctions between the three MOMENT entries are similar. The SLOAD entry is used to apply loads to scalar points.

7.3.2 Distributed Loads

The distributed loads consist of two categories: those which are stationary in direction through out the analysis and those which follow the element motion. The first category consists of the

PLOAD1.

The second category consists of the following:

PLOAD, PLOAD2, and PLOAD4.

Loads may be applied to the interiors of finite elements. The PLOAD1 entry defines concentrated and linearly distributed forces and moments to the BEAM element. However, the stress calculation for nonlinear BEAM element ignores the distributed loads, but lump them onto the nodal points. The PLOAD entry is exceptional in that it references three or four arbitrarily located grids rather than an element. The PLOAD2 entry defines a uniform static pressure load applied to the QUAD4 or TRIA3 elements. The PLOAD4 defines a pressure load on surfaces of HEXA, PENTA, TRIA3, and QUAD4 elements which need not be normal to the surface.

7.3.3 Mass Related Loads

The mass related loads are gravity and centrifugal loads, which require specification of mass data. The gravity loads which can be specified in the Bulk Data entry

GRAV

are stationary in direction through out the analysis. It is used to define the direction and magnitude of the gravity vector in some identified coordinate system. The components of the gravity vector are multiplied by the mass matrix to obtain the components of the gravity force at each grid point.

The centrifugal loads to be specified in the Bulk Data entry

RFORCE

is used to define the components of a spin vector which is used internally to compute centrifugal forces. Each component of the spin vector is multiplied by a scale factor.

7.3.4 Thermal Loads

Thermal loads are selected in each subcase by the Case Control command TEMP(LOAD). Thermal loading which is a follower loading is accomplished through the following entries:

TEMP, TEMPD, TEMPP1, TEMPP3, and TEMPRB

The TEMP and TEMPD provide for grid point temperature specification which are then interpolated to points within elements. For the nonlinear QUAD4 and TRIA3 elements the TEMPP1 provides for an average temperature and thermal gradient in the thickness direction and the TEMPP3 provides for temperature at points through the thickness. The TEMPRB provides for average temperature at ends A and B for the ROD and BEAM element. Aver-

age temperature specified directly for an element will take precedence over the temperature interpolated from the elements connected grid points.

There was a limitation until Version 65 which required specification of only one increment in the subcase where thermal load is first applied or removed. This limitation was removed in Version 66. This limitation was caused by the initial temperature which was specified in the material reference temperature field. In addition, the material temperature, TEMP(MAT), was distinguished from the loading temperature, TEMP(LOAD), until Version 65. Segregating these temperatures is not meaningful in the nonlinear analysis. In order to maintain a consistent temperature field, TEMP(MAT) was defaulted to TEMP(LOAD) in Version 66. In addition, a Case Control command TEMP(INIT) was created for the initial temperature distribution. For upward compatibility, Version 66 allowed TEMP(MAT) option in SOL 66, which caused a number of errors. For these reasons, TEMP(MAT) option was abandoned in Version 67 and TEMP(INIT) was made a required data in case any thermal load is present in the analysis.

The Case Control TEMP(INIT) is required above subcase level. The use of TEMP(MAT) or TEMP(BOTH) will cause a fatal error. Subcases that do not contain a TEMP(LOAD) will default to TEMP(INIT), which implies no thermal load. Thermal loads are often used to provide initial strains.

7.3.5 Loads Combination

Various load specifications can be combined using

LOAD and (CLOAD, LSEQ).

The LOAD provides for the linear combination of static load sets. The static loads (LOAD, FORCE, etc.) applied to the upstream superelements cannot be referenced by a Case Control command LOAD. The CLOAD entry is designed to apply static loads to upstream superelements by combining loads defined in LSEQ entries. The (CLOAD, LSEQ) combination allows the nonlinear algorithms to apply in an incremental fashion on the upstream superelement loads to the boundary of the residual structure.

The CLOAD defines a static load combination for superelement loads acting on residual boundary and the LSEQ defines a static load combination for superelement upstream loads. The LSEQ assigns load vectors to the superelements and numerically labels them via the DAREA field entry value. The LID and TID field entries point to the appropriate load and temperature distribution entries. The CLOAD defines a static load as a linear combination of previously calculated superelement loads defined by the LSEQ. The IDVi fields correspond directly to the numeric labels defined via the DAREA fields of the LSEQ entries.

The CLOAD must be selected in the residual solution subcase via the case control CLOAD entry. Any loads referenced by CLOAD should not be again referenced by LOAD until Ver-

sion 66, otherwise the load will be doubled: e.g., GRAV and TEMP. This deficiency has been corrected in Version 67. The LSEQ is selected by a LOADSET Case Control command above any residual subcase. Only one LOADSET may appear in Case Control. The hierarchy of the loads data is shown schematically in Fig. 7.3.1. An example for the Case Control set-up is shown below:

```

      :
SEALL=ALL
LOADSET = 1000 $ Selects LSEQ 1000 for upstream loads
SUPER = ALL $ Identify SEs to process
      :
      DISP = ALL
      :
      :
      :
$ NONLINEAR SOLUTIONS FOR RESIDUAL SUPERELEMENTS
SUBCASE 10
      CLOAD = 1001 $ Refers to CLOAD Bulk Data
      NLPARM = 12 $ Iteration control
      :
SUBCASE 20
      CLOAD = 1002
      LOAD = 10 $ Residual SE load
      NLPARM = 22
      :

```

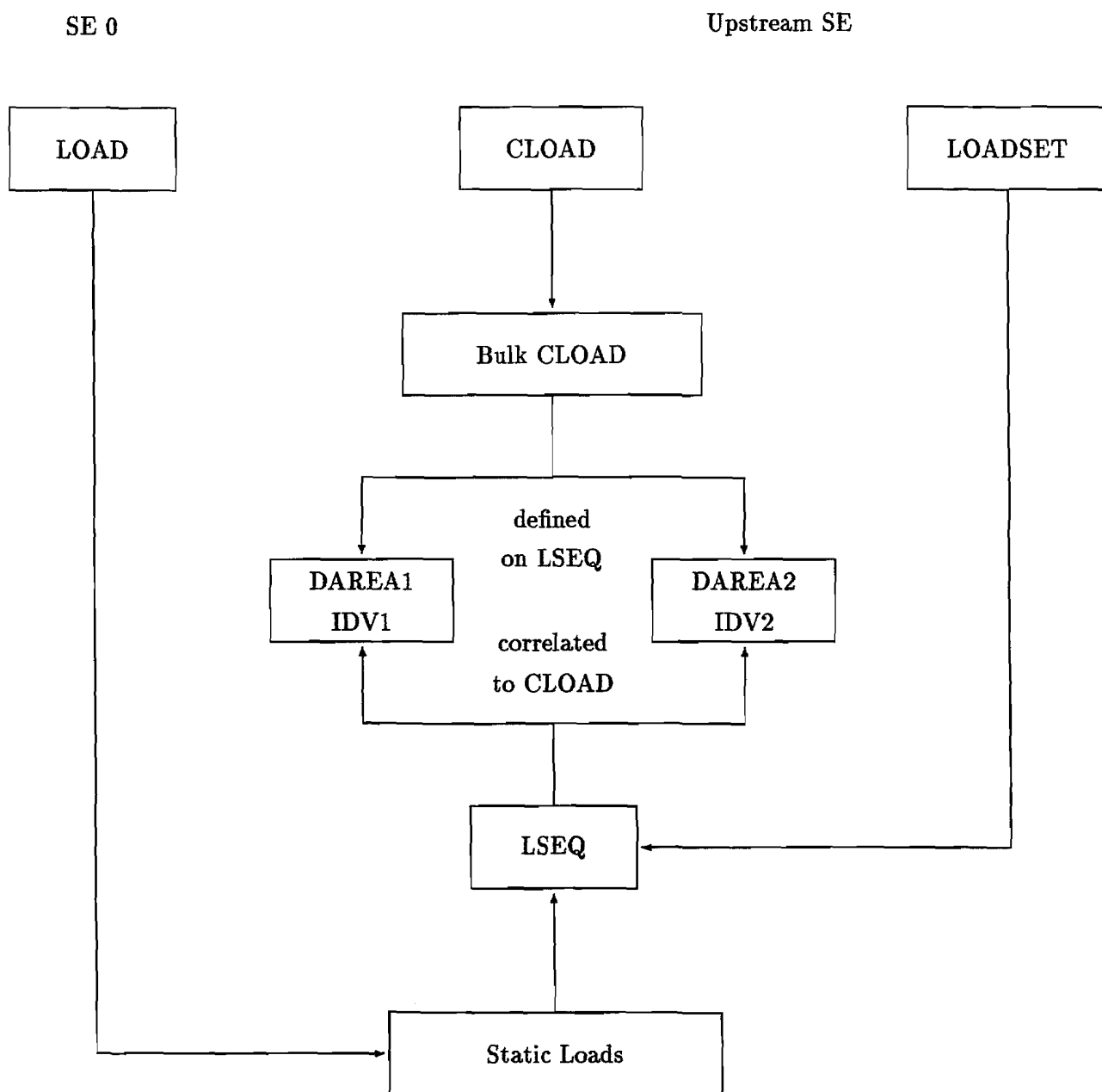


Figure 7.3.1 Hierarchy of Load Data

7.4 NONPROPORTIONAL LOADING

The loading condition under which all the stress components increase in the same proportion is called proportional loading, i.e.,

$$\frac{\Delta\sigma_x}{\sigma_x} = \frac{\Delta\sigma_y}{\sigma_y} = \frac{\Delta\sigma_z}{\sigma_z} = \frac{\Delta\tau_{xy}}{\tau_{xy}} = \frac{\Delta\tau_{yz}}{\tau_{yz}} = \frac{\Delta\tau_{zx}}{\tau_{zx}} = \lambda$$

The stress gradient vector $\{\frac{\partial f}{\partial \sigma}\}$ remains unchanged during proportional loading, and the stress state may be expressed in terms of any one stress component. It can be shown that the stress-strain relations reduce to stepwise-linear equations in the case of proportional loading. Proportional loading is in general difficult to achieve except for very simple and special cases such as uniaxial loading. Notice that the linear strain path does not always result in the proportional loading condition.

7.4.1 Validation of Nonproportional Loading Case

The general cases of nonproportional loading add more complexity to nonlinearity. Two classes of problems in this category can be contemplated: a strain path dependent problem controlled by a geometric (or essential) boundary condition and a stress path dependent problem controlled by a force (or natural) boundary condition. Both cases were validated by a simple model of a single HEXA element with eight Gauss quadrature points using the von Mises yield criterion and the isotropic hardening rule. In the stress path dependent problem, the loading was applied via the LOAD and PLOAD4 Bulk Data entries while in the strain path dependent problem, the SPC and SPC1 Bulk Data entries specified the loading conditions. The material properties were specified using the MAT1 and MATS1 Bulk Data entries and the solution strategy was provided by the NLPARM Bulk Data entries. The input data are shown in Tables 7.4.1 and 7.4.2 for the stress and strain path dependent problems, respectively.

First, the solution was obtained for a prescribed stress path (shown in Fig. 7.4.1(a)), and the strain path was plotted in Fig. 7.4.1(b). The solution was validated by obtaining a solution to an inverted problem, in which the stress path was measured while the strain path was traced. As illustrated in the figures, discrepancy between the two solutions is not discernible. Notice that points A and B in Fig. 7.4.1 signify the elastic limit and the turning point into the nonproportional loading, respectively.

By virtue of the subincremental scheme within the material routine, the nonproportional loading did not pose any difficulty. In order to examine the accuracy with respect to the number of increments in the nonproportional loading region, a parametric study was performed by taking a various number of global increments (2 to 10) between points B and D. Errors are measured at points C and D for five different cases as tabulated below:

No. of Inc.	Error in Stresses (%)			
	σ_x		σ_z	
	Point C	Point D	Point C	Point D
2	0.486	0.162	0.185	0.208
4	0.301	0.131	0.128	0.138
6	0.100	0.069	0.035	0.074
8	0.056	0.021	0.017	0.016
10	0.097	0.003	0.035	0.000

Note: Errors in $\bar{\sigma}$ are less than 0.0001 %.

It is observed that smaller increments tend to reduce the error. However, an acceptable solution was achieved with only two increments. Effectively, the entire loading path could have been analyzed in three steps (one in proportional and two in nonproportional loading) for engineering accuracy. The fact that the solution did not improve from eight to ten steps indicates that the number of subincrements is reduced when the global increment size is sufficiently small due to the adaptive subincremental scheme.

7.4.2 Biaxial Loading Case for Neutral Loading

Neutral loading, which is a special case of nonproportional loading, is verified by a biaxially loaded plate under the in-plane, plane-stress condition using the von Mises yield criterion and kinematic hardening. The finite element model and the material properties are shown in Figures 7.4.2(a) and 7.4.2(b). The plate is modeled by four QUAD4 elements with the material properties specified in the MAT1 and MATS1 Bulk Data entries. Appropriate boundary conditions are provided by the SPC1 and MPC Bulk Data entries. The loading is applied via the FORCE Bulk Data entries and the solution strategy is specified in the NLPARM Bulk Data entries. The input data are given in Table 7.4.3.

The loading path is shown in Fig. 7.4.2(c). The loading sequence starts with a uniaxial tension in the x-direction to 40.5 ksi, followed by a slight unloading to 40 ksi. The subsequent loading path is defined by an elliptic locus $\sigma_x^2 - \sigma_x\sigma_y + \sigma_y^2 = 40^2$. Although the locus seems to be a yield locus, it is slightly different because of the kinematic hardening rule employed in the analysis. Therefore, the second loading phase is nearly, but not exactly, neutral loading.

The analysis was performed in 36 increments (7 in radial loading and 29 in nonproportional loading), which ran to completion without any difficulty in convergence. The incremental process progressed with 11 stiffness matrix updates, 145 iterations and 36 line searches. It is noted that increments 12 and 13, where the incremental step size is larger than other steps, required more iterations (7 and 10, respectively) and line searches (7 and 18, respectively). The solution of the strain path (ϵ_x vs. ϵ_y) corresponding to the loading path is compared with the one presented by Chern [7.1], as shown in Fig. 7.4.2(d). The stress-strain response curves in x and y components are plotted in Fig. 7.4.2(e). The response is highly nonlinear during the second loading phase due to the nonproportional loading path.

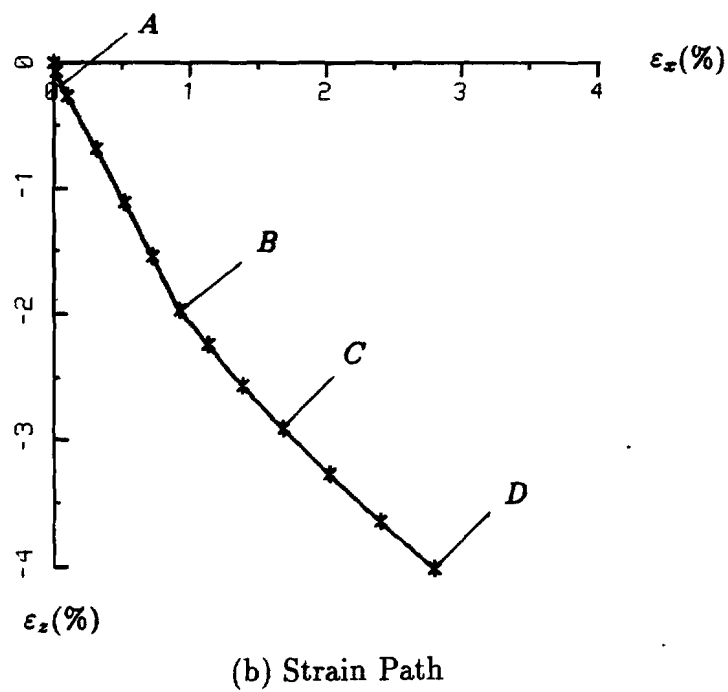
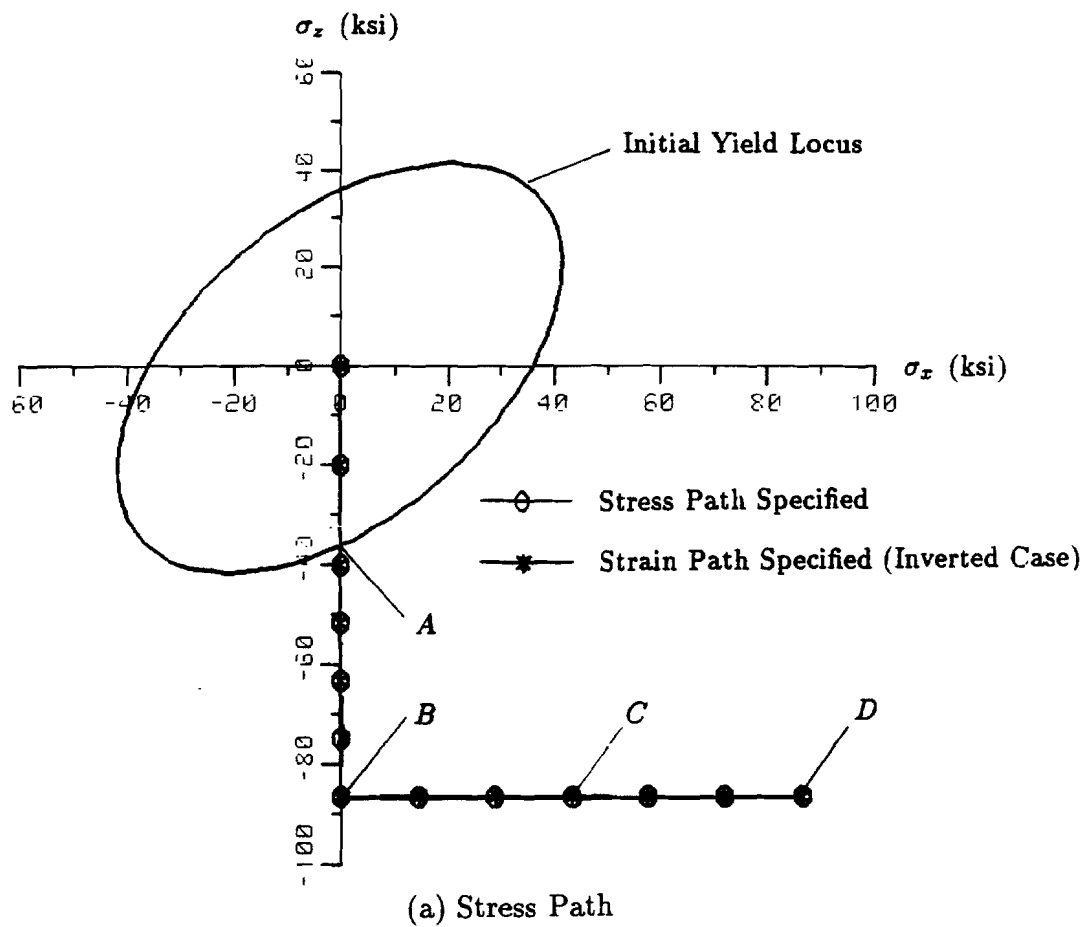


Figure 7.4.1 Validation of Nonproportional Loading Case

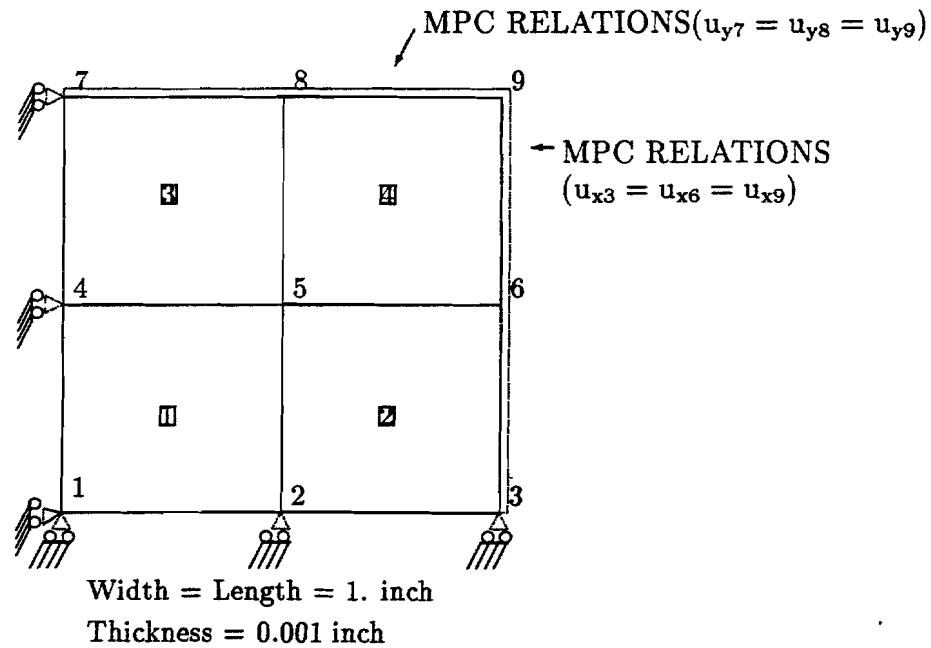
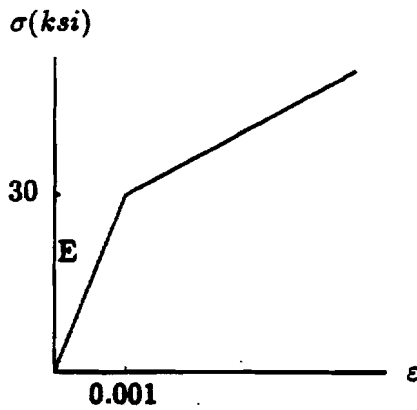


Figure 7.4.2(a) Finite Element Model for Biaxial Loading



$$\begin{aligned}
 E &= 30.0 \times 10^3 \text{ ksi} \\
 \nu &= 0.3 \\
 H &= 1.5 \times 10^3 \text{ ksi} \\
 &\text{Kinematic Hardening}
 \end{aligned}$$

7.4.2(b) Material Curve for Biaxial Loading

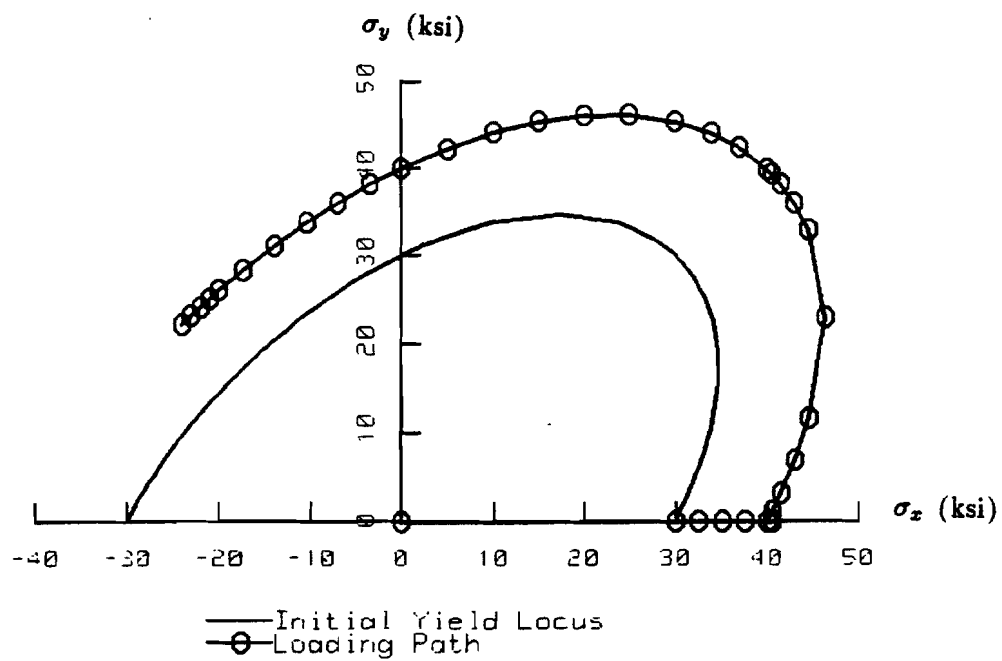


Figure 7.4.2(c) Biaxial Loading Path

STEP NO.	σ_x	σ_y
1	30000.	0.
2	32500.	0.
3	35000.	0.
4	37500.	0.
5	40000.	0.
6	40500.	0.
7	40000.	0.
8 to 36	$(\sigma_x^2 + \sigma_y^2 - \sigma_x \sigma_y)^{1/2} = 40000$	

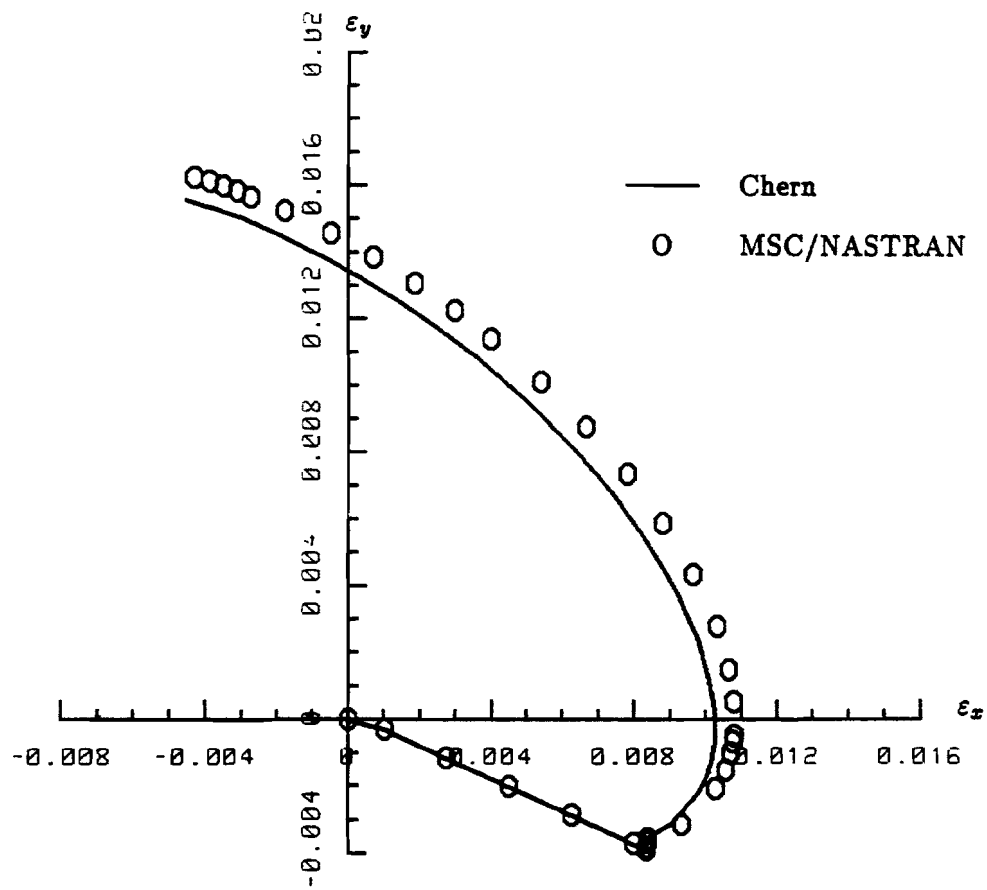


Figure 7.4.2(d) Corresponding Strain Path for Biaxial Loading

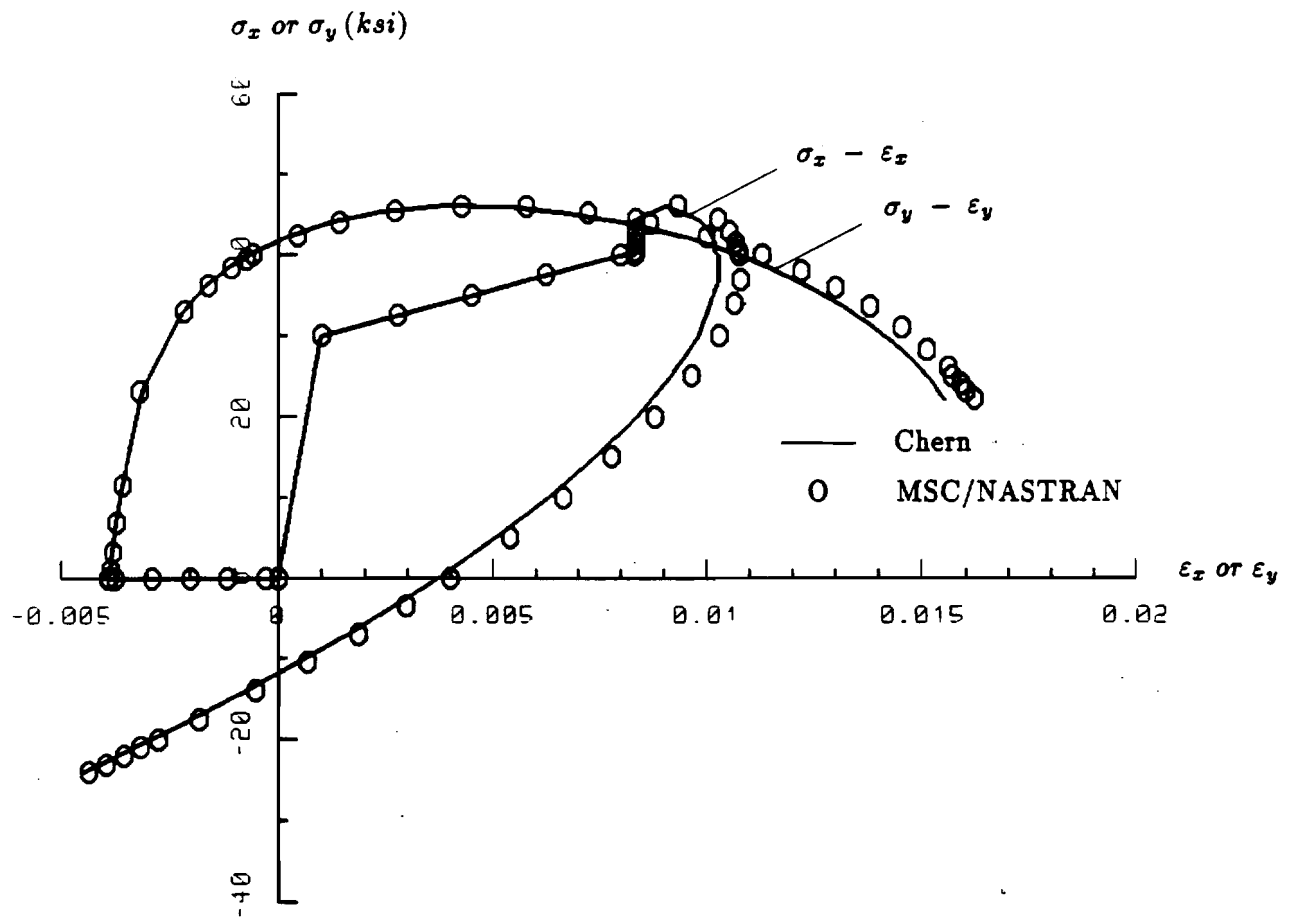


Figure 7.4.2(e) Stress-Strain Response Curves

Table 7.4.1 Input Data Listing for Stress Path Dependent Case

```

ID NLPATH2,V66  $  CJS 7-MAY-85,  SHL 1981
SOL      66      $  NONLINEAR STATIC ANALYSIS
DIAG     8,50    $  PRINT MATRIX TRAILERS AND ITERATIONS
TIME     5       $  CPU MIN FOR VAX
CEND
TITLE = CODE TEST FOR NONLINEAR ELEMENTS WITH STRESS PATH 2
ECHO = UNSORT
SEALL = ALL
  DISP = ALL
  OLOAD = ALL
  SPCF = ALL
  STRESS(PLOT) = ALL
SPC = 100
SUBCASE 1195
  LABEL = UNIAXIAL COMPRESSION
  LOAD  = 1195
  NLPARM = 10
SUBCASE 1196
  LABEL = PLANE STRAIN TENSION
  LOAD  = 1196
  NLPARM = 20
OUTPUT(PLOT)
  CSCALE = 1.3
  PLOTTER NAST
  SET 1 = ALL
  VIEW 30., 20., 0.
  PTITLE = ISOMETRIC VIEW
  FIND SCALE, ORIGIN 1, SET 1
PLOT  SET 1, ORIGIN 1, LABEL BOTH
PLOT  STATIC 0, SET 1, ORIGIN 1
  VIEW 0., 0., 0.
  PTITLE = FRONT VIEW
  FIND SCALE, ORIGIN 1, SET 1
PLOT  SET 1, ORIGIN 1
PLOT  STATIC 0, SET 1, ORIGIN 1
  AXES Z, Y, MX
  PTITLE = TOP VIEW
  FIND SCALE, ORIGIN 1, SET 1
PLOT  SET 1, ORIGIN 1
PLOT  STATIC 0, SET 1, ORIGIN 1
BEGIN BULK
$
NLPARM  10      2      AUTO      UPW      YES
NLPARM  20      4      SEMI      20      UPW      YES
$$ GEOMETRY
GRID    1000      0.      0.      0.
GRID    1100      1.      0.      0.
GRID    1110      1.      1.      0.

```


GRID	1010		0.	1.	0.				
GRID	1001		0.	0.	1.				
GRID	1101		1.	0.	1.				
GRID	1111		1.	1.	1.				
GRID	1011		0.	1.	1.				
GRDSET								456	
\$\$ CONNECTIVITY									
CHEXA	100	1	1000	1100	1110	1010	1001	1101	+HX100
+HX100	1111	1011							
\$									
CELAS4	9999	1.	9999						
SLOAD	100	9999	1.-5						
\$\$ MATERIAL AND GEOMETRIC PROPERTIES									
MAT1	10	30.+6	11.5+6	.3	7.332-4				
MATS1	10		PLASTIC	3.+6	1	1	36.+3		
PSOLID	1	10	0	. 0					
\$\$ BOUNDARY CONDITIONS									
SPC1	100	123456	1000						
SPC1	100	1	1010						
SPC1	100	2	1001						
SPC1	100	3	1100						
\$\$ LOADING CONDITIONS									
LOAD	1195	2.4	1.	123					
LOAD	1196	2.4	1.	123	-1.	121			
\$ UNIAXIAL TENSION IN X									
PLOAD4	121	100	36.+3				1100	1111	
PLOAD4	121	100	36.+3				1000	1011	
\$ UNIAXIAL TENSION IN Z									
PLOAD4	123	100	36.+3				1000	1110	
PLOAD4	123	100	36.+3				1001	1111	
ENDDATA									

Table 7.4.2 Input Data Listing for Strain Path Dependent Case

```

ID NLPATH3, V66 $ QYC 8/24/90
SOL      66      $ NONLINEAR STATIC ANALYSIS
DIAG     8,50    $ PRINT MATRIX TRAILERS AND ITERATIONS
TIME     5       $ FOR VAX
CEND
TITLE = CODE TEST FOR NONLINEAR ELEMENTS WITH STRAIN PATH 3
ECHO = UNSORT
SEALL = ALL
  DISP = ALL
  OLOAD = ALL
  SPCF = ALL
  STRES = ALL
SUBCASE 10
  LABEL = SUBCASE 10
  SPC = 10
  NLPARM = 10
SUBCASE 20
  LABEL = SUBCASE 20
  SPC = 20
  NLPARM = 20
SUBCASE 30
  LABEL = SUBCASE 30
  SPC = 30
  NLPARM = 20
SUBCASE 40
  LABEL = SUBCASE 40
  SPC = 40
  NLPARM = 20
SUBCASE 50
  LABEL = SUBCASE 50
  SPC = 50
  NLPARM = 20
SUBCASE 60
  LABEL = SUBCASE 60
  SPC = 60
  NLPARM = 20
SUBCASE 70
  LABEL = SUBCASE 70
  SPC = 70
  NLPARM = 20
SUBCASE 80
  LABEL = SUBCASE 80
  SPC = 80
  NLPARM = 20
SUBCASE 90
  LABEL = SUBCASE 90
  SPC = 90
  NLPARM = 20

```

```

SUBCASE 100
  LABEL = SUBCASE 100
  SPC = 100
  NLPARM = 20
SUBCASE 110
  LABEL = SUBCASE 110
  SPC = 110
  NLPARM = 20
SUBCASE 120
  LABEL = SUBCASE 120
  SPC = 120
  NLPARM = 20
OUTPUT(PLOT)
  CSCALE = 1.3
  PLOTTER NAST
  SET 1 = ALL
  VIEW 30., 20., 0.
  PTITLE = ISOMETRIC VIEW
  FIND SCALE, ORIGIN 1, SET 1
PLOT SET 1, ORIGIN 1, LABEL BOTH
PLOT STATIC 0, SET 1, ORIGIN 1
  VIEW 0., 0., 0.
  PTITLE = FRONT VIEW
  FIND SCALE, ORIGIN 1, SET 1
PLOT SET 1, ORIGIN 1
PLOT STATIC 0, SET 1, ORIGIN 1
  AXES Z, Y, MX
  PTITLE = TOP VIEW
  FIND SCALE, ORIGIN 1, SET 1
PLOT SET 1, ORIGIN 1
PLOT STATIC 0, SET 1, ORIGIN 1
BEGIN BULK
$
NLPARM 10      1          SEMI          UPW      YES
NLPARM 20      1          SEMI          20      UPW      YES
$$ GEOMETRY
GRID 1000      0.      0.      0.
GRID 1100      1.      0.      0.
GRID 1110      1.      1.      0.
GRID 1010      0.      1.      0.
GRID 1001      0.      0.      1.
GRID 1101      1.      0.      1.
GRID 1111      1.      1.      1.
GRID 1011      0.      1.      1.
GRDSET
456
$$ CONNECTIVITY
CHEXA 100      1      1000      1100      1110      1010      1001      1101      +HX100
+HX100 1111      1011
$

```

CELAS4 9999 1. 9999
 SLOAD 100 9999 1.-5
 \$\$ MATERIAL AND GEOMETRIC PROPERTIES
 MAT1 1 30.+6 11.5+6 .3 7.332-4
 MATS1 1 PLASTIC 3.+6 1 1 36.+3
 PSOLID 1 1 0 0
 \$\$ LOADING CONDITIONS
 SPC1 10 1 1000 1010 1001 1011
 SPC1 10 2 1000 1100 1001 1101
 SPC1 10 3 1000 1100 1010 1110
 SPC 10 1100 1 2.-4 1110 1 2.-4
 SPC 10 1101 1 2.-4 1111 1 2.-4
 SPC 10 1001 3 -6.67-4 1101 3 -6.67-4
 SPC 10 1011 3 -6.67-4 1111 3 -6.67-4
 \$
 SPC1 20 1 1000 1010 1001 1011
 SPC1 20 2 1000 1100 1001 1101
 SPC1 20 3 1000 1100 1010 1110
 SPC 20 1100 1 1.07-3 1110 1 1.07-3
 SPC 20 1101 1 1.07-3 1111 1 1.07-3
 SPC 20 1001 3 -2.67-3 1101 3 -2.67-3
 SPC 20 1011 3 -2.67-3 1111 3 -2.67-3
 \$
 SPC1 30 1 1000 1010 1001 1011
 SPC1 30 2 1000 1100 1001 1101
 SPC1 30 3 1000 1100 1010 1110
 SPC 30 1100 1 3.12-3 1110 1 3.12-3
 SPC 30 1101 1 3.12-3 1111 1 3.12-3
 SPC 30 1001 3 -6.92-3 1101 3 -6.92-3
 SPC 30 1011 3 -6.92-3 1111 3 -6.92-3
 \$
 SPC1 40 1 1000 1010 1001 1011
 SPC1 40 2 1000 1100 1001 1101
 SPC1 40 3 1000 1100 1010 1110
 SPC 40 1100 1 5.17-3 1110 1 5.17-3
 SPC 40 1101 1 5.17-3 1111 1 5.17-3
 SPC 40 1001 3 -1.12-2 1101 3 -1.12-2
 SPC 40 1011 3 -1.12-2 1111 3 -1.12-2
 \$
 SPC1 50 1 1000 1010 1001 1011
 SPC1 50 2 1000 1100 1001 1101
 SPC1 50 3 1000 1100 1010 1110
 SPC 50 1100 1 7.21-3 1110 1 7.21-3
 SPC 50 1101 1 7.21-3 1111 1 7.21-3
 SPC 50 1001 3 -1.54-2 1101 3 -1.54-2
 SPC 50 1011 3 -1.54-2 1111 3 -1.54-2
 \$
 SPC1 60 1 1000 1010 1001 1011
 SPC1 60 2 1000 1100 1001 1101

SPC1	60	3	1000	1100	1010	1110	
SPC	60	1100	1	9.26-3	1110	1	9.26-3
SPC	60	1101	1	9.26-3	1111	1	9.26-3
SPC	60	1001	3	-1.97-2	1101	3	-1.97-2
SPC	60	1011	3	-1.97-2	1111	3	-1.97-2
\$							
SPC1	70	1	1000	1010	1001	1011	
SPC1	70	2	1000	1100	1001	1101	
SPC1	70	3	1000	1100	1010	1110	
SPC	70	1100	1	1.124-2	1110	1	1.124-2
SPC	70	1101	1	1.124-2	1111	1	1.124-2
SPC	70	1001	3	-2.25-2	1101	3	-2.25-2
SPC	70	1011	3	-2.25-2	1111	3	-2.25-2
\$							
SPC1	80	1	1000	1010	1001	1011	
SPC1	80	2	1000	1100	1001	1101	
SPC1	80	3	1000	1100	1010	1110	
SPC	80	1100	1	1.378-2	1110	1	1.378-2
SPC	80	1101	1	1.378-2	1111	1	1.378-2
SPC	80	1001	3	-2.57-2	1101	3	-2.57-2
SPC	80	1011	3	-2.57-2	1111	3	-2.57-2
\$							
SPC1	90	1	1000	1010	1001	1011	
SPC1	90	2	1000	1100	1001	1101	
SPC1	90	3	1000	1100	1010	1110	
SPC	90	1100	1	1.68-2	1110	1	1.68-2
SPC	90	1101	1	1.68-2	1111	1	1.68-2
SPC	90	1001	3	-2.92-2	1101	3	-2.92-2
SPC	90	1011	3	-2.92-2	1111	3	-2.92-2
\$							
SPC1	100	1	1000	1010	1001	1011	
SPC1	100	2	1000	1100	1001	1101	
SPC1	100	3	1000	1100	1010	1110	
SPC	100	1100	1	2.021-2	1110	1	2.021-2
SPC	100	1101	1	2.021-2	1111	1	2.021-2
SPC	100	1001	3	-3.28-2	1101	3	-3.28-2
SPC	100	1011	3	-3.28-2	1111	3	-3.28-2
\$							
SPC1	110	1	1000	1010	1001	1011	
SPC1	110	2	1000	1100	1001	1101	
SPC1	110	3	1000	1100	1010	1110	
SPC	110	1100	1	2.394-2	1110	1	2.394-2
SPC	110	1101	1	2.394-2	1111	1	2.394-2
SPC	110	1001	3	-3.65-2	1101	3	-3.65-2
SPC	110	1011	3	-3.65-2	1111	3	-3.65-2
\$							
SPC1	120	1	1000	1010	1001	1011	
SPC1	120	2	1000	1100	1001	1101	
SPC1	120	3	1000	1100	1010	1110	

SPC	120	1100	1	2.79-2	1110	1	2.79-2
SPC	120	1101	1	2.79-2	1111	1	2.79-2
SPC	120	1001	3	-4.03-2	1101	3	-4.03-2
SPC	120	1011	3	-4.03-2	1111	3	-4.03-2

ENDDATA

Table 7.4.3 Input Data Listing for Biaxial Loading Case

```

ID BIAxIAL, V66 $ FEB84
SOL      66      $ NONLINEAR STATIC ANALYSIS
DIAG     8,50    $ PRINT MATRIX TRAILERS AND ITERATIONS
TIME     5       $ CPU MIN FOR VAX
CEND
TITLE = QUAD4 - PLASTIC ANALYSIS CHECK (M. WASYLczAK, MCAUTO)
SUBTITLE = BIAxIAL LOADING CONDITION
LABEL = REF.: FOSTER WHEELER REPORT FWR-27, MARCH 1972, P. A-50
SEALL = ALL
  STRESS = ALL
  SPC = 10
  MPC = 10
SUBCASE 1
  LOAD   = 1
  NLPARM = 10
SUBCASE 2
  LOAD   = 2
  NLPARM = 10
SUBCASE 3
  LOAD   = 3
  NLPARM = 10
SUBCASE 4
  LOAD   = 4
  NLPARM = 10
SUBCASE 5
  LOAD   = 5
  NLPARM = 10
SUBCASE 6
  LOAD   = 6
  NLPARM = 10
SUBCASE 7
  LOAD   = 7
  NLPARM = 10
SUBCASE 8
  LOAD   = 8
  NLPARM = 10
SUBCASE 9
  LOAD   = 9
  NLPARM = 10
SUBCASE 10
  LOAD   = 10
  NLPARM = 10
SUBCASE 11
  LOAD   = 11
  NLPARM = 10
SUBCASE 12
  LOAD   = 12
  NLPARM = 12

```

SUBCASE 13
LOAD = 13
NLPARM = 13
SUBCASE 14
LOAD = 14
NLPARM = 14
SUBCASE 15
LOAD = 15
NLPARM = 10
SUBCASE 16
LOAD = 16
NLPARM = 10
SUBCASE 17
LOAD = 17
NLPARM = 10
SUBCASE 18
LOAD = 18
NLPARM = 10
SUBCASE 19
LOAD = 19
NLPARM = 10
SUBCASE 20
LOAD = 20
NLPARM = 10
SUBCASE 21
LOAD = 21
NLPARM = 10
SUBCASE 22
LOAD = 22
NLPARM = 10
SUBCASE 23
LOAD = 23
NLPARM = 10
SUBCASE 24
LOAD = 24
NLPARM = 10
SUBCASE 25
LOAD = 25
NLPARM = 10
SUBCASE 26
LOAD = 26
NLPARM = 10
SUBCASE 27
LOAD = 27
NLPARM = 10
SUBCASE 28
LOAD = 28
NLPARM = 10
SUBCASE 29


```

LOAD = 29
NLPARM = 10
SUBCASE 30
LOAD = 30
NLPARM = 10
SUBCASE 31
LOAD = 31
NLPARM = 10
SUBCASE 32
LOAD = 32
NLPARM = 10
SUBCASE 33
LOAD = 33
NLPARM = 10
SUBCASE 34
LOAD = 34
NLPARM = 10
SUBCASE 35
LOAD = 35
NLPARM = 10
SUBCASE 36
LOAD = 36
NLPARM = 10
BEGIN BULK
$$ PARAMETERS
PARAM NLAYERS 1
$
NLPARM 10 1 AUTO
NLPARM 12 1 AUTO 2 30
NLPARM 13 1 AUTO 2 30 +NLP13
+NLP13 3.5E-3
NLPARM 14 1 AUTO 2 30
$$ GEOMETRY
GRID 1 0.0 0.0 0.0
GRID 2 0.5 0.0 0.0
GRID 3 1.0 0.0 0.0
GRID 4 0.0 0.5 0.0
GRID 5 0.5 0.5 0.0
GRID 6 1.0 0.5 0.0
GRID 7 0.0 1.0 0.0
GRID 8 0.5 1.0 0.0
GRID 9 1.0 1.0 0.0
$$ CONNECTIVITY
CQUAD4 1 1 1 2 5 4
CQUAD4 2 1 2 3 6 5
CQUAD4 3 1 4 5 8 7
CQUAD4 4 1 5 6 9 8
$$ PROPERTIES
PSHELL 1 1 .001

```

MAT1	1	30.+6		.3			
MATS1	1		PLASTIC	1.5+6	1	2	30000.
\$\$ CONSTRAINTS							
MPC	1	6	1	1.	3	1	-1.
MPC	1	9	1	1.	3	1	-1.
MPC	1	7	2	1.	9	2	-1.
MPC	1	8	2	1.	9	2	-1.
SPC1	1	3456	1	THRU	9		
SPC1	1	2	1	2	3		
SPC1	1	1	1	4	7		
\$\$ LOADING							
FORCE	1	3		30.	1.	0.	0.
FORCE	2	3		32.5	1.	0.	0.
FORCE	3	3		35.	1.	0.	0.
FORCE	4	3		37.5	1.	0.	0.
FORCE	5	3		40.	1.	0.	0.
FORCE	6	3		40.5	1.	0.	0.
FORCE	7	3		40.	1.	0.	0.
FORCE	8	3		40.5	1.	0.	0.
FORCE	8	9		1.02	0.	1.	0.
FORCE	9	3		41.5	1.	0.	0.
FORCE	9	9		3.19	0.	1.	0.
FORCE	10	3		43.	1.	0.	0.
FORCE	10	9		6.89	0.	1.	0.
FORCE	11	3		44.5	1.	0.	0.
FORCE	11	9		11.53	0.	1.	0.
FORCE	12	3		46.188	1.	0.	0.
FORCE	12	9		23.05	0.	1.	0.
FORCE	13	3		44.5	1.	0.	0.
FORCE	13	9		32.965	0.	1.	0.
FORCE	14	3		43.	1.	0.	0.
FORCE	14	9		36.1	0.	1.	0.
FORCE	15	3		41.5	1.	0.	0.
FORCE	15	9		38.3	0.	1.	0.
FORCE	16	3		40.5	1.	0.	0.
FORCE	16	9		39.48	0.	1.	0.
FORCE	17	3		40.	1.	0.	0.
FORCE	17	9		40.	0.	1.	0.
FORCE	18	3		37.	1.	0.	0.
FORCE	18	9		42.44	0.	1.	0.
FORCE	19	3		34.	1.	0.	0.
FORCE	19	9		44.07	0.	1.	0.
FORCE	20	3		30.	1.	0.	0.
FORCE	20	9		45.41	0.	1.	0.
FORCE	21	3		25.	1.	0.	0.
FORCE	21	9		46.13	0.	1.	0.
FORCE	22	3		20.	1.	0.	0.
FORCE	22	9		46.05	0.	1.	0.
FORCE	23	3		15.	1.	0.	0.

FORCE	23	9	45.33	0.	1.	0.
FORCE	24	3	10.	1.	0.	0.
FORCE	24	9	44.05	0.	1.	0.
FORCE	25	3	5.	1.	0.	0.
FORCE	25	9	42.26	0.	1.	0.
FORCE	26	3	0.	1.	0.	0.
FORCE	26	9	40.	0.	1.	0.
FORCE	27	3	-3.5	1.	0.	0.
FORCE	27	9	38.13	0.	1.	0.
FORCE	28	3	-7.	1.	0.	0.
FORCE	28	9	36.04	0.	1.	0.
FORCE	29	3	-10.5	1.	0.	0.
FORCE	29	9	33.7	0.	1.	0.
FORCE	30	3	-14.	1.	0.	0.
FORCE	30	9	31.12	0.	1.	0.
FORCE	31	3	-17.5	1.	0.	0.
FORCE	31	9	28.27	0.	1.	0.
FORCE	32	3	-20.	1.	0.	0.
FORCE	32	9	26.06	0.	1.	0.
FORCE	33	3	-21.	1.	0.	0.
FORCE	33	9	25.13	0.	1.	0.
FORCE	34	3	-22.	1.	0.	0.
FORCE	34	9	24.17	0.	1.	0.
FORCE	35	3	-23.	1.	0.	0.
FORCE	35	9	23.19	0.	1.	0.
FORCE	36	3	-24.	1.	0.	0.
FORCE	36	9	22.18	0.	1.	0.

\$

ENDDATA

7.5 CONSTRAINTS AND ENFORCED MOTION

In addition to the loading conditions, SOL 66 (or SOL 106) allows the user to change the boundary conditions or enforce the displacements at the specified grid or scalar points.

7.5.1 Boundary Conditions

In the Case Control section, an SPC entry is used to select a single point constraint set (s-set) which will be applied to the structural model. The specified set identification must be identical to the SID field of an SPC, SPC1 or SPCADD Bulk Data entry. Notice that SPCADD entries take precedence over SPC or SPC1 entries. If both have the same SID, only the SPCADD entry will be used.

A significant application of SPC is the imposition of boundary conditions. The PS field in the GRID entry is also able to specify single-point constraints associated with a grid point. But these constraints are so-called permanent constraints which can not be changed during the analysis. An advantage of using SPC to specify boundary conditions is that these boundary conditions can be changed from subcase to subcase by simply selecting a different SPC set inside each subcase.

7.5.2 Multipoint Constraint

While SPC is to constrain the motion by specifying a known value, MPC is to tie the motion of a degree of freedom to other degrees of freedom. An MPC entry in the Case Control section is used to select a multipoint constraint set to be applied to the structural model. The specified MPC set identification must appear at least in one MPC or MPCADD Bulk Data entry. Each MPC Bulk Data entry may be used to define a constraint equation involving a group of degrees of freedom in which the first degree of freedom is assumed to be the dependent degree of freedom and included in the m-set. All the degree of freedom in m-set will be condensed out prior to the matrix operations. Their response will be directly recovered from those of the independent degrees of freedom according to the specified constraint equation. Similar to SPC, MPC can also be changed from subcase to subcase by specifying a different MPC set ID inside each subcase in static analysis (SOL 66 or SOL 106).

7.5.3 Enforced Motion

The enforced displacements may be specified in the Bulk Data section using SPC or SPCD entries. In the former case, the enforced displacement values may be input directly in the D field of the SPC entry. Each SPC entry may define enforced displacements for up to two grid or scalar points. Several SPC entries which reference the same SID may be used if enforced

displacements for more than two grid or scalar points are desired. The only disadvantage of this method is that the entire s-set must be redefined if the enforced displacement conditions vary among subcases.

One way to avoid this condition is to define the enforced displacements using SPCD entries. In this case, all the enforced degrees-of-freedom must be included in the s-set by SPC or SPC1 entries. The SPCD entry is requested as a LOAD, and therefore a LOAD set must also be selected in the Case Control section. Since only the LOAD set needs to be changed instead of the entire s-set, this method of applying the enforced displacements is more efficient than the SPC entry method when more than one enforced displacement condition is applied. If the enforced displacements are specified on both the SPCD and SPC entries, then the former will override the latter. Other Bulk Data load entries such as FORCE, FORCE1, PLOAD, PLOADi, etc. may also be combined with SPCD. However, it should be noted that the Bulk Data LOAD combination entry will not combine an SPCD load.

The table below summarizes the data entries required for the enforced motions or the boundary condition changes in MSC/NASTRAN.

Option	Case Control Entries	Bulk Data Entries
1	SUBCASE 1 SPC = 100 SUBCASE 2 SPC = 200	SPC 100 SPC 200
2	SUBCASE 1 SPC = 100 LOAD = 10 SUBCASE 2 SPC = 100 LOAD = 20	SPC(or SPC1) 100 SPCD 10 SPCD 20

7.5.4 Example

Fig. 7.5.1 shows a steel clip which will be pushed into a cavity whose height is 5.0 inches. It is assumed that there is no friction force at the contact surface. The clip is modeled with 24 elastic BEAM elements. Since a part of the clip bottom might lift off the ground during the process, 7 GAP elements are provided at the clip-ground interface. The goal of the analysis is to predict the final configuration of the clip.

It is obvious from Fig. 7.5.1 that GRID 18 will be the first grid point to come in contact with the ceiling of the cavity. Since the height of the clip at GRID 18 is 5.5 inches and that of the cavity is 5.0 inches, an enforced displacement of -0.5 inches in the y direction is applied at GRID 18 in the cold start run. The result shows that the height of the deformed clip at GRID 19 is 5.264 inches, which is still greater than the height of the cavity. Therefore, in the first restart run the enforced displacement at GRID 18 is released and another enforced

displacement of -1.0 inches (i.e., $5.0 - 6.0 = -1.0$) is applied in the y direction at GRID 19. Notice that the specified enforced displacements should always refer to the undeformed configuration and the software will automatically compute the increments. The analysis from this run predicts that GRID 20 will attain a height of 5:156 inches, which is still greater than the cavity height. A second restart is performed to release the constraint at GRID 19 and enforce the displacement at GRID 20 by -1.5 inches in the y direction. At this time, the analysis results indicate that no grid point in the model has a height over 5.0 inches, which means that the entire clip is inside the cavity. The final deformed shape of the clip is shown in Fig. 7.5.1. The predicted heights for some of the grid points are tabulated below at different stages with the values greater than 5.0 in italics.

	Grid 17	Grid 18	Grid 19	Grid 20	Grid 21
Original	4.750	<i>5.500</i>	<i>6.000</i>	<i>6.500</i>	<i>7.000</i>
Cold Start	4.493	5.000	<i>5.264</i>	<i>5.529</i>	<i>5.721</i>
1st Restart	4.439	4.842	5.000	<i>5.156</i>	<i>5.211</i>
2nd Restart	4.465	4.812	4.911	5.000	4.969

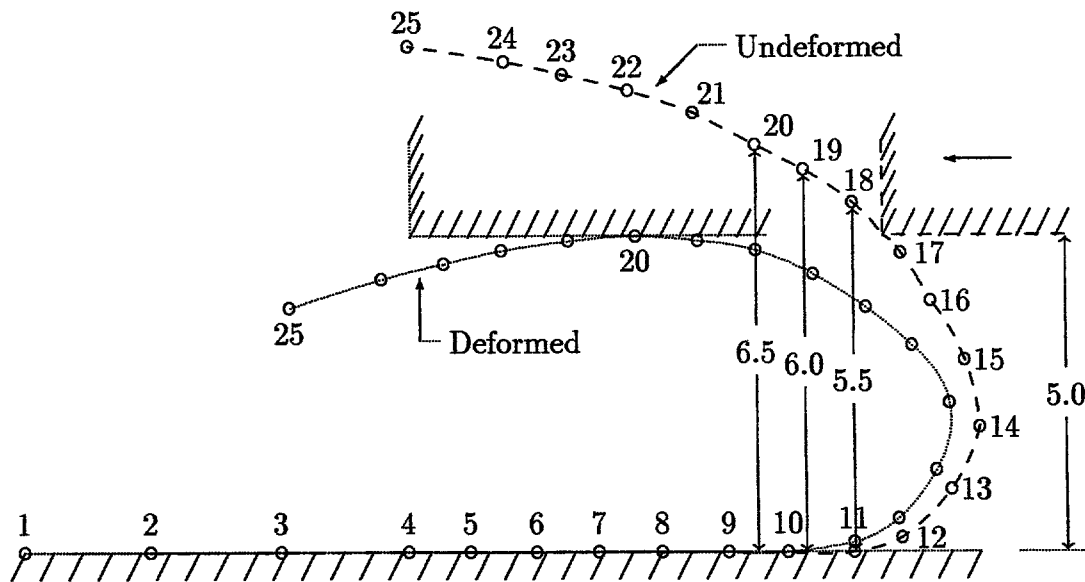


Figure 7.5.1. Elastic Clip Pushed into a Cavity

Table 7.5.1. Input Data Listing for Elastic Clip

```

$RESTART          $ added for the restart runs
ID      CLIPB, V67 $ SSH 12/12/90
TIME    5          $
SOL      66        $ NONLINEAR STATIC ANALYSIS
DIAG     8,50      $
CEND
TITLE = INSERTING A CLIP INTO A CAVITY (BEAM MODEL)
SEALL = ALL
ECHO = BOTH
    DISP = ALL
    SPCF = ALL
$-----
$PARAM,SUBID,2    $ used in the 1st restart
$PARAM,LOOPID,2   $
$-----
$PARAM,SUBID,3    $ used in the 2nd restart
$PARAM,LOOPID,7   $
$-----
SUBCASE 10
    SPC    = 10
    LOAD   = 100
    NLPARM = 1
$-----
$SUBCASE 20      $
$   SPC=20      $ added for the 1st restart
$   LOAD=200    $
$   NLPARM = 2  $
$-----
$SUBCASE 30      $
$   SPC=30      $ added for the 2nd restart
$   LOAD=300    $
$   NLPARM = 3  $
$-----
BEGIN BULK
PARAM  LGDISP  1
$$ SOLUTION CONTROL
NLPARM  1      2      0.      SEMI      1
NLPARM  2      5      0.      AUTO
NLPARM  3      4      0.      AUTO
$
$$ GEOMETRY
$
GRID    1      0.      0.      0.      123456
GRID    2      2.      0.      0.      2345
GRID    3      4.      0.      0.      2345

```

GRID	4	6.	0.	0.	2345
GRID	5	7.	0.	0.	2345
GRID	6	8.	0.	0.	345
GRID	7	6.	0.	0.	345
GRID	8	10.	0.	0.	345
GRID	9	11.	0.	0.	345
GRID	10	12.	0.	0.	345
GRID	11	13.	0.	0.	345
GRID	12	13.875	0.125	0.	345
GRID	13	14.5	1.0	0.	345
GRID	14	15.0	2.0	0.	345
GRID	15	14.75	3.0	0.	345
GRID	16	14.25	4.0	0.	345
GRID	17	13.75	4.75	0.	345
GRID	18	13.0	5.5	0.	345
GRID	19	12.25	6.0	0.	345
GRID	20	11.5	6.5	0.	345
GRID	21	10.5	7.0	0.	345
GRID	22	9.5	7.375	0.	345
GRID	23	8.5	7.6	0.	345
GRID	24	7.5	7.8	0.	345
GRID	25	6.0	8.0	0.	345
GRID	106	8.	0.	0.	123456
GRID	107	6.	0.	0.	123456
GRID	108	10.	0.	0.	123456
GRID	109	11.	0.	0.	123456
GRID	110	12.	0.	0.	123456
GRID	111	13.	0.	0.	123456
GRID	112	13.875	0.125	0.	123456

\$

\$\$ ELEMENT DEFINITION

\$

CBEAM	101	100	1	2	0.	0.	1.
CBEAM	102	100	2	3	0.	0.	1.
CBEAM	103	100	3	4	0.	0.	1.
CBEAM	104	100	4	5	0.	0.	1.
CBEAM	105	100	5	6	0.	0.	1.
CBEAM	106	100	6	7	0.	0.	1.
CBEAM	107	100	7	8	0.	0.	1.
CBEAM	108	100	8	9	0.	0.	1.
CBEAM	109	100	9	10	0.	0.	1.
CBEAM	110	100	10	11	0.	0.	1.
CBEAM	111	100	11	12	0.	0.	1.
CBEAM	112	100	12	13	0.	0.	1.
CBEAM	113	100	13	14	0.	0.	1.
CBEAM	114	100	14	15	0.	0.	1.

CBEAM	115	100	15	16	0.	0.	1.	
CBEAM	116	100	16	17	0.	0.	1.	
CBEAM	117	100	17	18	0.	0.	1.	
CBEAM	118	100	18	19	0.	0.	1.	
CBEAM	119	100	19	20	0.	0.	1.	
CBEAM	120	100	20	21	0.	0.	1.	
CBEAM	121	100	21	22	0.	0.	1.	
CBEAM	122	100	22	23	0.	0.	1.	
CBEAM	123	100	23	24	0.	0.	1.	
CBEAM	124	100	24	25	0.	0.	1.	
CORD2R	1		0.	0.	0.	0.	0.	-1. +COR1
+COR1	0.	1.	-1.					
CGAP	206	200	106	6	0.	0.	1.	1
CGAP	207	200	107	7	0.	0.	1.	1
CGAP	208	200	108	8	0.	0.	1.	1
CGAP	209	200	109	9	0.	0.	1.	1
CGAP	210	200	110	10	0.	0.	1.	1
CGAP	211	200	111	11	0.	0.	1.	1
CGAP	212	210	112	12	0.	0.	1.	1
\$								
\$\$ PROPERTIES								
\$								
PBEAM	100	100	.15	2.8125-42.8125-4				
MAT1	100	2.+7	1.+7	0.				
PGAP	200	0.	0.	1.+4	1.	0.		
PGAP	210	0.125	0.	1.+4	1.	0.		
\$								
\$\$ LOADING								
\$								
SPC1	10	2	18					
SPC1	20	2	19					
SPC1	30	2	20					
SPCD	100	18	2	-0.5				
SPCD	200	19	2	-1.				
SPCD	300	20	2	-1.5				
\$								
ENDDATA								

7.6 RESTARTS

Restarts may be performed for various purposes. The most common reason is to extend a normally or abnormally (due to divergence or CPU time expiration, etc.) terminated analysis. Restarts may also be used only for obtaining additional output. The present section will concentrate on these two types of restarts. In addition, one may use SOL 66 to preload a nonlinear structure and then restart into either SOL 66 for a nonlinear buckling analysis, SOL 63 for a normal mode analysis, or SOL 99 for a nonlinear transient analysis.

7.6.1 Preliminaries

In Version 65 the data blocks corresponding to every converged load step in SOL 66 are saved for restarts by default. If PARAM,SMALLDB is specified as 1, however, solution data blocks corresponding to the intermediate load steps not requested for output (by INTOUT field in NLPARM) are deleted from the database after a subcase is completed. In this case, the restarts are restricted to those starting from the solution with the corresponding data blocks stored in the database, i.e., any converged load step in the current subcase or from the end of any previous subcase. Note that the small database option (PARAM,SMALLDB,1) is only applicable in Version 65.

In Version 66, the DMAP was modified to store only the data blocks for the output load steps without requiring parameter SMALLDB. Therefore, the solution can only be restarted from the end of a subcase if the default (INTOUT=NO) is used.

Further improvements were made in Version 67 so that the data blocks for the last converged load step will be saved if the solution is terminated in the middle of a subcase due to divergence or consumption of the specified CPU time. Consequently, the solution can always be restarted from the termination point regardless of the value of INTOUT in the NLPARM entry. In addition, restarts from the unstable region for the arc-length methods were also made available in Version 67.

7.6.2 Restart Procedure

Restarts in SOL 66 are controlled by three user parameters which may be specified either in the Case Control section prior to the subcases or in the Bulk Data section:

PARAM, LOOPID, ℓ

PARAM, SUBID, m

PARAM, LOADINC, n (only applicable in V65 and V66)

The integer ℓ identifies data blocks for a specific converged solution stored in the data base, which may be used as initial condition for a subsequent restart. The value of LOOPID as well as SUBID and LOADINC for each solution step is printed with the iteration information in the output. PARAMeters SUBID and LOADINC refer to the initial load step for the restart. SUBID selects the subcase in which the restart analysis is initiated and LOADINC selects the load step number in that subcase. The default values for LOADINC and SUBID are 1.

The simplest way to perform a restart analysis is to specify a new value for SUBID which is greater than the SUBID value appearing in the last converged solution step of the previous run (e.g., new SUBID = last SUBID + 1) and to use the default for LOADINC (i.e., LOADINC = 1). This means that the solution will restart into the very first load step of a new subcase. If a subcase with the new SUBID value is added to the Case Control section of the restart data deck, the solution will continue with the new LOAD and NLPARM data specified in that subcase. If the new subcase is not added to the restart data deck, only the data recovery (without any computation) will be processed in the restart analysis.

For example, a load P is applied in one subcase with 5 increments, i.e., $0.2P$ for each increment. Assume that the solution diverges and the third load increment (assume its LOOPID = 3) is the last solution step saved for the restart. In the restart data deck, one may set the new parameter values LOOPID=3, SUBID=2, LOADINC=1 and add a second subcase in the Case Control section. Assuming that the second subcase specifies the same LOAD and NLPARM entries as the first subcase (i.e., load P with 5 increments), the solution will continue with a load increment of $(P - 3 \times 0.2P)/5 = 0.08P$. If the second subcase is not added, only the data recovery will be processed.

In Versions 65 and 66, a value greater than 1 may be specified for LOADINC if the new SUBID value is the same as that which appeared in the previous run. Using the example described in the preceding paragraph, one may also set LOOPID=3, SUBID=1, LOADINC=4 and leave out the second subcase. In this case, the solution will restart into the fourth load increment to complete the remaining part of the first subcase. Since this feature has only limited applications and is very difficult to use with the adaptive load/arc-length method, it is removed from SOL 66 in Version 67.

7.6.3 Example

Fig. 7.6.1 shows a nearly-horizontal simple rod whose left end is hinged and right end is simply supported. An elastic spring of 3 lbs/in. is connected in the vertical direction to the right end of the rod, where a vertical force is applied. This is a snap-through problem with a single degree-of-freedom. The force-displacement curve of the rod is shown in Fig. 7.6.2.

In the cold start run whose input data is given in Table 7.6.1, the loading is increased from 0 to 2 lbs in two load increments and the analysis executed to completion. A new subcase is added in the restart run to increase the loading to 4 lbs using two load increments. The input data for this run is given in Table 7.6.2. This analysis diverges in the second load increment,

i.e., LOOPID 3 (LOAD STEP 1.5) is the last converged solution step which is saved for the restart (except Version 66). The solution is restarted again using the arc-length method, whose input data is given in Table 7.6.3. In this run, MAXINC is intentionally set to 2 in the NLPCI entry so that the analysis would terminate with a unstable solution. Finally, the solution is restarted a third time to reach the loading of 4 lbs. The input data for the third restart is given in Table 7.6.4.

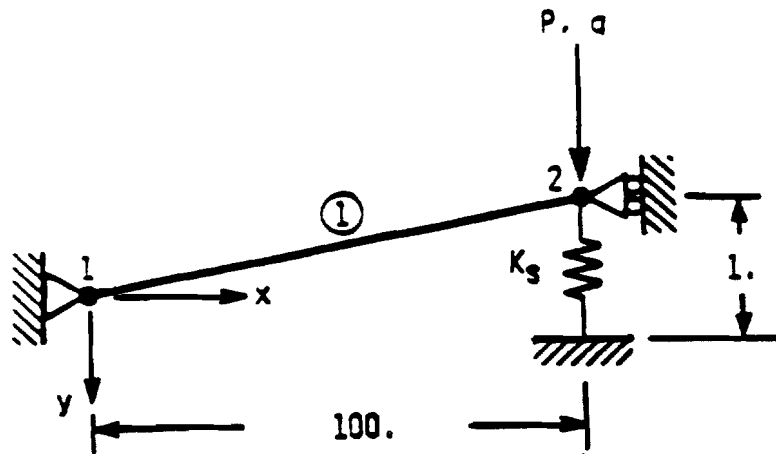


Figure 7.6.1 Model of Rod with Spring

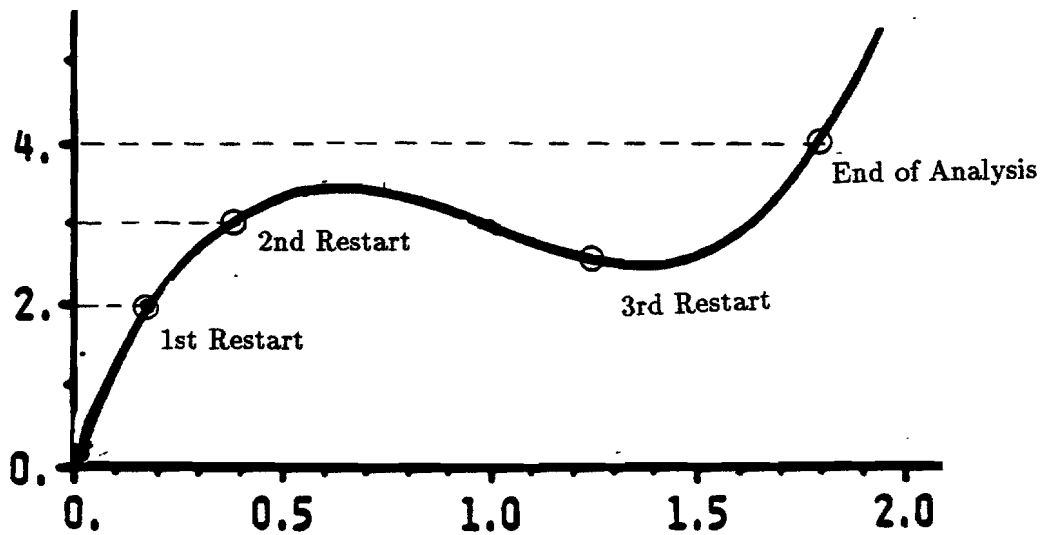


Figure 7.6.2 Force-Displacement Curve

Table 7.6.1 Input Data Listing for Cold Start

```

ID SPROD,V67 $ SSH 6/17/91
TIME 5      $
SOL 66      $
DIAG 8,50   $
CEND
TITLE=SIMPLE ONE DOF GEOMETRIC NONLINEAR PROBLEM
SUBTITLE= COLD START RUN
LABEL=REF: STRICKLIN AND HAISLER; COMP. AND STRUCT.; 7:125-136 (1977)
ECHO=UNSORT
SEALL=ALL
  OLOAD=ALL
  DISP=ALL
SUBCASE 10 $ LOAD = 2 LB.
  LOAD=1
  NLPARM=1
BEGIN BULK
$ GEOMETRY
GRID    1          0.      0.      0.          123456
GRID    2          100.    -1.      0.          13456
$ CONNECTIVITY
CROD    10         10      1        2
CELAS1  20         20      2        2      0      0
$ PROPERTIES
PROD    10         1        .1
PELAS   20         3.
MAT1    1          10.E7
$ LOADING
FORCE   1          2          2.      0.      1.      0.
$ PARAMETERS
PARAM   LGDISP    +1
NLPARM  1          2                      +NLP21
$
ENDDATA

```

Table 7.6.2 Input Data Listing for First Restart

```

RESTART
ID SPRODR1,V67 $ SSH 6/17/91
TIME 5 $
SOL 66 $
DIAG 8,50 $
CEND
TITLE=SIMPLE ONE DOF GEOMETRIC NONLINEAR PROBLEM
SUBTITLE= FIRST RESTART (INCREASES THE LOAD UP TO 4 LBS)
LABEL=REF: STRICKLIN AND HAISLER; COMP. AND STRUCT.; 7:125-136 (1977)
$
ECHO=UNSORT
SEALL=ALL
OLOAD = ALL
DISP = ALL
PARAM, SUBID, 2
PARAM, LOOPID, 2
SUBCASE 10 $ LOAD = 2 LB.
    LOAD=1
    NLPARM=1
SUBCASE 20 $ LOAD = 4 LB.
    LOAD=2
    NLPARM=2
BEGIN BULK
$ GEOMETRY
GRID    1           0.      0.      0.      123456
GRID    2          100.    -1.      0.      13456
$ CONNECTIVITY
CROD    10         10      1      2
CELAS1  20         20      2      2      0      0
$ PROPERTIES
PROD    10         1      .1
PELAS   20         3.
MAT1    1          10.E7
$ LOADING
FORCE   1          2           2.      0.      1.      0.
FORCE   2          2           4.      0.      1.      0.
$ PARAMETERS
PARAM   LGDISP    +1
NLPARM  1          2
NLPARM  2          2
+NLP21              -2
+NLP22  0
$
ENDDATA

```

Table 7.6.3 Input Data Listing for Second Restart

```

RESTART
ID SPRODR2,V67 $ SSH 6/17/91
TIME 5 $
SOL 66 $
DIAG 8,50 $
CEND
TITLE=SIMPLE ONE DOF GEOMETRIC NONLINEAR PROBLEM
SUBTITLE= 2ND RESTART (SWITCHES TO THE ARC-LENGTH METHOD)
LABEL=REF: STRICKLIN AND HAISLER; COMP. AND STRUCT.; 7:125-136 (1977)
ECHO=UNSORT
SEALL=ALL
OLOAD = ALL
DISP = ALL
PARAM, SUBID, 3
PARAM, LOOPID, 3
SUBCASE 10 $ LOAD = 2 LB.
    LOAD=1
    NLPARM=1
SUBCASE 20 $ LOAD = 4 LB.
    LOAD=2
    NLPARM=2
SUBCASE 30 $ LOAD = 4 LB., ARC-LENGTH METHOD
    LOAD=2
    NLPARM=3
BEGIN BULK
$ GEOMETRY
GRID 1 0. 0. 0. 123456
GRID 2 100. -1. 0. 13456
$ CONNECTIVITY
CROD 10 10 1 2
CELAS1 20 20 2 2 0 0
$ PROPERTIES
PROD 10 1 .1
PELAS 20 3.
MAT1 1 10.E7
$ LOADING
FORCE 1 2 2. 0. 1. 0.
FORCE 2 2 4. 0. 1. 0.
$ PARAMETERS
PARAM LGDISP +1
NLPARM 1 2 +NLP11
NLPARM 2 2 +NLP21
+NLP21 -2 +NLP22
+NLP22 0
NLPARM 3 1 +NLP31
NLPCI 3 2
$
ENDDATA

```

Table 7.6.4 Input Data Listing for Third Restart

```

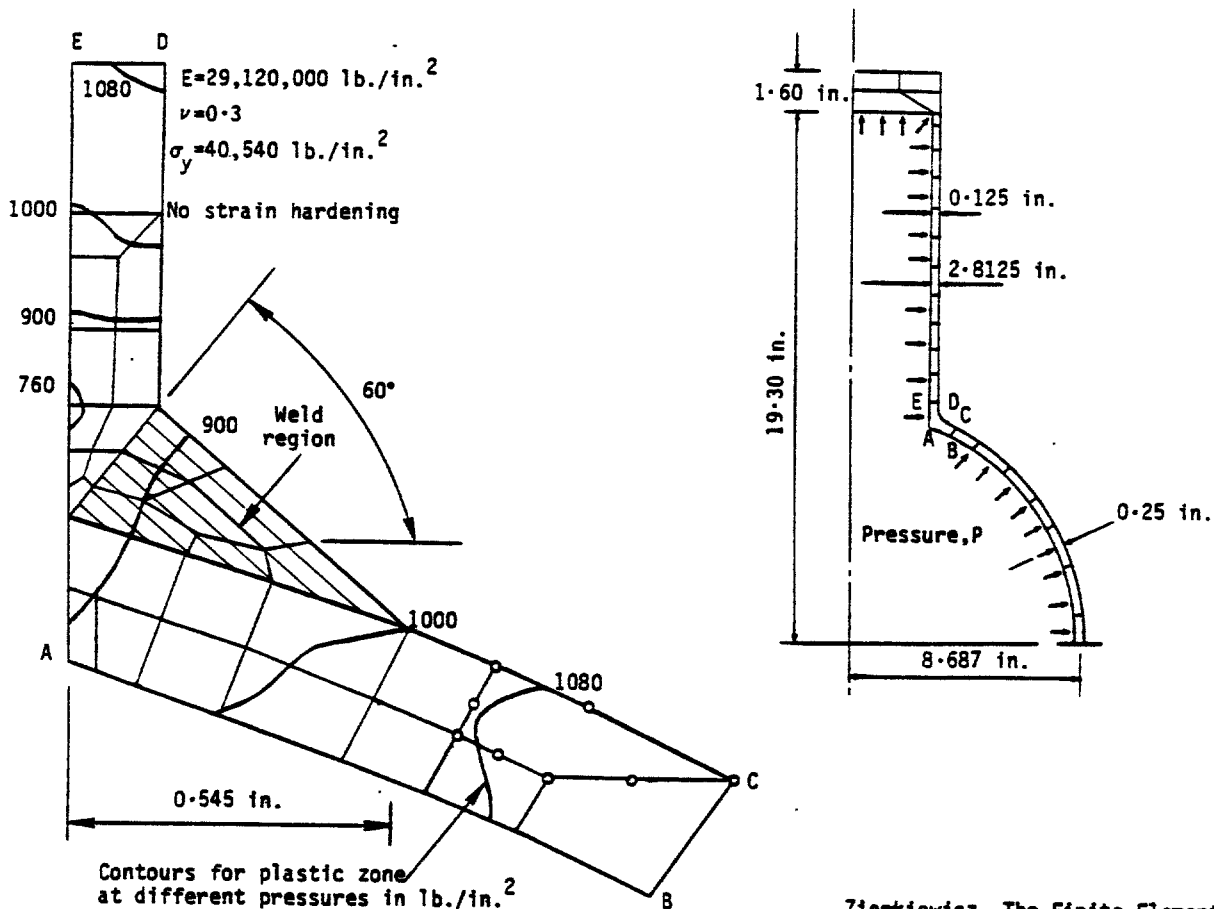
RESTART
ID SPRODR3,V67 $ SSH 6/17/91
TIME 5      $
SOL 66      $
DIAG 8,50   $
CEND
TITLE=SIMPLE ONE DOF GEOMETRIC NONLINEAR PROBLEM
SUBTITLE= 3RD RESTART (RESTARTS IN THE UNSTABLE REGION TO COMPLETION)
LABEL=REF: STRICKLIN AND HAISLER; COMP. AND STRUCT.; 7:125-136 (1977)
$
ECHO=UNSORT
SEALL=ALL
OLOAD = ALL
DISP = ALL
PARAM, SUBID, 4
PARAM, LOOPID, 5
SUBCASE 10 $ LOAD = 2 LB.  COLD START
    LOAD=1
    NLPARM=1
SUBCASE 20 $ LOAD = 4 LB.  NORMAL RESTART
    LOAD=2
    NLPARM=2
SUBCASE 30 $ LOAD = 4 LB., RESTART WITH THE ARC-LENGTH METHOD
    LOAD=2
    NLPARM=3
SUBCASE 40 $ LOAD = 4 LB., RESTART IN UNSTABLE REGION
    LOAD=2
    NLPARM=4
BEGIN BULK
$ GEOMETRY
GRID    1           0.      0.      0.           123456
GRID    2           100.    -1.      0.           13456
$ CONNECTIVITY
CROD    10          10      1        2
CELAS1  20          20      2        2        0        0
$ PROPERTIES
PROD    10          1        .1
PELAS   20          3.
MAT1    1           10.E7
$ LOADING
FORCE   1           2           2.      0.      1.      0.
FORCE   2           2           4.      0.      1.      0.
$ PARAMETERS
PARAM   LGDISP  +1
NLPARM  1        2
NLPARM  2        2
+NLP21
-2
+NLP22

```


+NLP22	0				
NLPARM	3	1			+NLP31
NLPCI	3			2	
NLPARM	4	1			+NLP41
NLPCI	4				
\$					
ENDDATA					

7.7 VERIFICATION PROBLEM: ANALYSIS of a PRESSURE VESSEL

This problem is used to illustrate the nonlinear static analysis of pressure vessel using QUAD4 elements. The pressure vessel is illustrated in Fig. 7.7.1. The pressure vessel consists of a spherical shell joined by a thin cylindrical shell at the top. A 10° sector of the steel pressure vessel is modeled using QUAD4 and PENTA elements as shown in Fig. 7.7.2. The intersection of the spherical and cylindrical shells of the vessel is difficult to model using plate elements. This difficulty is emphasized here because changes in the plate thicknesses at the junction result in large differences in the final answers. The results agree reasonably well with the experimental results of Dinno and Gill [7.3] as shown in Fig. 7.7.3. The input data for the model is listed in Table 7.7.1.



Zienkiewicz, *The Finite Element Method*
McGraw-Hill, 1971, Figure 18.10, P. 474

Figure 7.7.1. Steel Pressure Vessel.

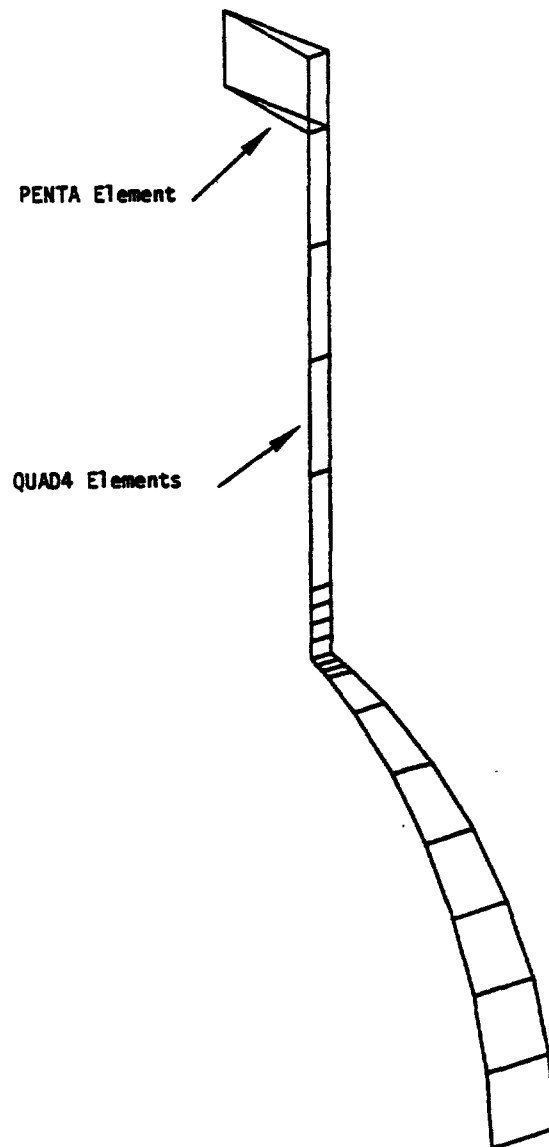


Figure 7.7.2. Model (10% Sector of the Steel Pressure Vessel)

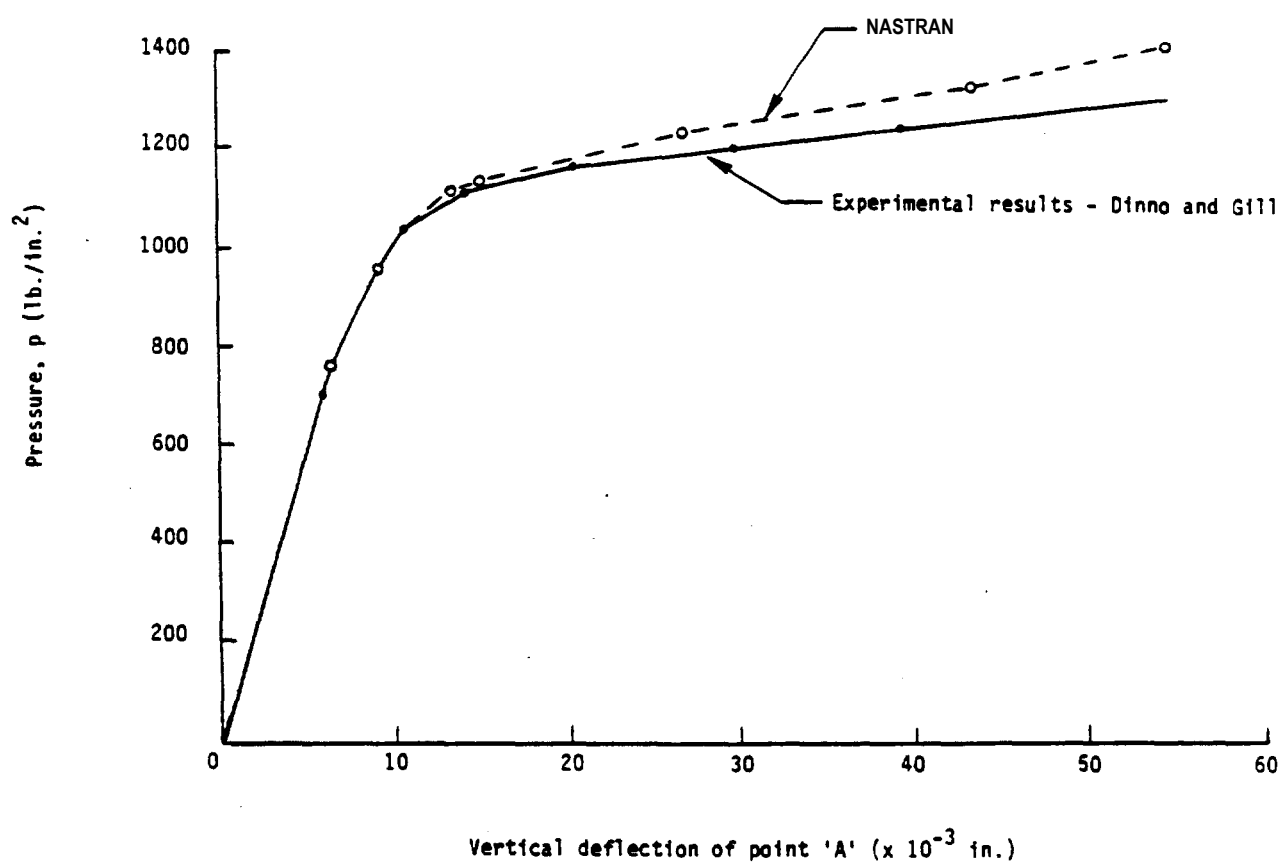


Figure 7.7.3. Experimental Results vs. NASTRAN

Table 7.7.1 Input Data Listing for Pressure Vessel Model

```

ID VESSEL,V65 $ CJS 27-SEP-1985, 15-MAY-1984
SOL 66
DIAG 8,50
TIME 30
CEND
TITLE = PRESSURE VESSEL
SUBTITLE = MATERIAL NONLINEAR ANALYSIS
  ECHO = UNSORT
  SEALL = ALL
  DISP = ALL
  ELST = ALL
SUBCASE 1
LABEL = PRESSURE LOAD OF 749.913 PSI
  LOAD = 10
  NLPARM = 10
SUBCASE 2
LABEL = PRESSURE LOAD OF 1080. PSI
  LOAD = 20
  NLPARM = 20
SUBCASE 3
LABEL = PRESSURE LOAD OF 1200. PSI
  LOAD = 30
  NLPARM = 30
SUBCASE 4
LABEL = PRESSURE LOAD OF 1400. PSI
  LOAD = 40
  NLPARM = 30
OUTPUT(PLOT)
  CSCALE 1.3
  SET 1 ALL
  MAXI DEFO 1.0
  VIEW 34.27, 23.17, 0.00
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1
  AXES MY, MZ, X
  VIEW 0., 5., 0.
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1
BEGIN BULK
$ PARAMETERS
PARAM  AUTOSPC YES
$ PARAMETERS FOR NONLINEAR ITERATION
NLPARM  10      1      AUTO      W      YES
NLPARM  20      3      AUTO      W      YES
NLPARM  30      5      AUTO      W      YES
$ DEFINE GRID LOCATIONS
CORD2C  10      0.      0.      0.      0.      0.      1.      +C2C1
+C2C1   1.      0.      0.

```

CORD2S	20		0.	0.	0.	0.	0.	1.	+C2S2
+C2S2	1.	0.	0.						
GRID	1	20	8.687	90.	0.	20	123456		
GRID	2	20	8.687	79.8415	0.	20	345		
GRID	3	20	8.687	69.683	0.	20	345		
GRID	4	20	8.687	59.524	0.	20	345		
GRID	6	20	8.687	39.2075	0.	20	345		
GRID	5	20	8.687	49.366	0.	20	345		
GRID	7	20	8.687	29.049	0.	20	345		
GRID	8	20	8.687	22.7367	0.	10	246		
GRID	9	20	8.687	21.443560.		10	246		
GRID	10	20	8.687	20.161780.		10	246		
GRID	11	10	2.8125	0.	8.2191	10	246		
GRID	12	10	2.8125	0.	8.58932510		246		
GRID	13	10	2.8125	0.	8.95955	10	246		
GRID	14	10	2.8125	0.	9.32977510		246		
GRID	15	10	2.8125	0.	9.7	10	246		
GRID	16	10	2.8125	0.	12.1	10	246		
GRID	17	10	2.8125	0.	14.5	10	246		
GRID	18	10	2.8125	0.	16.9	10	246		
GRID	19	10	2.8125	0.	19.3	10	246		
GRID	20	10	2.8125	0.	20.9	10	2456		
GRID	101	20	8.687	90.	10.	20	123456		
GRID	102	20	8.687	79.8415	10.	20	345		
GRID	103	20	8.687	69.683	10.	20	345		
GRID	104	20	8.687	59.524	10.	20	345		
GRID	105	20	8.687	49.366	10.	20	345		
GRID	106	20	8.687	39.2075	10.	20	345		
GRID	107	20	8.687	29.049	10.	20	345		
GRID	108	20	8.687	22.7367	10.	10	246		
GRID	109	20	8.687	21.4435610.		10	246		
GRID	110	20	8.687	20.1617810.		10	246		
GRID	111	10	2.8125	10.	8.2191	10	246		
GRID	112	10	2.8125	10.	8.58932510		246		
GRID	113	10	2.8125	10.	8.95955	10	246		
GRID	114	10	2.8125	10.	9.32977510		246		
GRID	115	10	2.8125	10.	9.7	10	246		
GRID	116	10	2.8125	10.	12.1	10	246		
GRID	117	10	2.8125	10.	14.5	10	246		
GRID	118	10	2.8125	10.	16.9	10	246		
GRID	119	10	2.8125	10.	19.3	10	246		
GRID	120	10	2.8125	10.	20.9	10	2456		
GRID	201		0.	0.	19.3		12456		
GRID	202		0.	0.	20.9		12456		
\$ QUAD4 ELEMENTS									
CQUAD4	1	1	1	101	102	2			
CQUAD4	2	1	2	102	103	3			
CQUAD4	3	1	3	103	104	4			
CQUAD4	4	1	4	104	105	5			

CQUAD4	5	1	5	105	106	6	
CQUAD4	6	1	6	106	107	7	
CQUAD4	7	1	7	107	108	8	
CQUAD4	8	1	8	108	109	9	
CQUAD4	9	4	9	109	110	10	+CQ0
+CQ0			0.25	0.25	0.28125	0.28125	
CQUAD4	10	3	10	110	111	11	+CQ1
+CQ1			0.28125	0.28125	0.3125	0.3125	
CQUAD4	11	3	11	111	112	12	+CQ2
+CQ2			0.3125	0.3125	0.15625	0.15625	
CQUAD4	12	5	12	112	113	13	+CQ3
+CQ3			0.15625	0.15625	0.125	0.125	
CQUAD4	13	2	13	113	114	14	
CQUAD4	14	2	14	114	115	15	
CQUAD4	15	2	15	115	116	16	
CQUAD4	16	2	16	116	117	17	
CQUAD4	17	2	17	117	118	18	
CQUAD4	18	2	18	118	119	19	
\$ QUAD4 CHARACTERISTICS							
PSHELL	1	1	0.25	1			
PSHELL	2	1	0.125	1			
PSHELL	3	1	0.3125	1			
PSHELL	4	1	0.28125	1			
PSHELL	5	1	0.15625	1			
MAT1	1	2.912+7	0.3				
MATS1	1		PLASTIC	1	1	4.054+4	
\$ PENTA ELEMENT							
CPENTA	101	4	201	19	119	202	20 120
\$ PENTA CHARACTERISTICS							
PSOLID	4	2	0				
MAT1	2	2.912+7	0.3				
\$ DEFINE LOADING CONDITIONS							
PLOAD2	10	749.913	1	THRU	18		
PLOAD2	20	1080.	1	THRU	18		
PLOAD2	30	1200.	1	THRU	18		
PLOAD2	40	1400.	1	THRU	18		
PLOAD4	10	101	749.913				201
PLOAD4	20	101	1080.				201
PLOAD4	30	101	1200.				201
PLOAD4	40	101	1400.				201
ENDDATA							

Chapter 8

DIRECT TIME INTEGRATION

8.1 OVERVIEW

If the effects of inertia and damping forces are to be included in the nonlinear analysis, the nonlinear transient response is analyzed by a step-by-step integration of an equation of motion, i.e.,

$$M \ddot{u} + C \dot{u} + F(u) = P(t). \quad (8.1.1)$$

By direct numerical integration, the equilibrium of Eq. (8.1.1) is satisfied at discrete time steps with an interval of Δt . The equilibrium is ensured by iterations until the solution converges to the preset error tolerance.

Since Version 62, several methods of improving the convergence of nonlinear problems have been adopted in the software [8.1]. These procedures, such as the quasi-Newton update and line search process, provide users with additional options to adjust incremental and iterative processes in search of more efficient and effective strategy for obtaining solutions. The goal of the general-purpose program is to provide users with tools that are easy to use. As a step toward this goal, an adaptive time stepping algorithm has been implemented for the nonlinear transient analysis.

A self-adaptive time stepping method has been implemented in Version 66 for the transient analysis of nonlinear structures. The goal is to provide an automatic method of implicit time integration, which renders accurate solutions efficiently to any nonlinear transient response problem. This goal was achieved by combining the automatic time step adjustment with the bisection process in a coordinated manner.

Newmark's direct time integration method is implemented using the two-point recurrence (or one-step) formula as a foundation of this self-adaptive time stepping algorithm [8.2]. The optimal time step size, which is required for accuracy and efficiency, changes continuously in the transient dynamic environment. The primary concept of automatic time step adjustment

is that the proper size of the time step can be predicted based on the dominant frequency in the incremental deformation pattern at the previous time step. This concept presents a deficiency of time lag involved in the prediction process. Furthermore, changes in nonlinearity cannot be predicted from the deformation pattern at the previous time step. This deficiency is overcome by the bisection process, which is activated when any difficulties arise in terms of convergence during the iteration.

Frequent decomposition of the dynamic stiffness matrix is inevitable in adaptive time stepping using the implicit integration method. Efforts have been made to minimize the thrashing in the algorithm (e.g., too frequent or repetitive adjustment of the time step size) and yet to provide a solution reliably to any type of problem. The iteration process for the equilibrium employs expeditious methods such as BFGS updates and line searches as well as an adaptive stiffness matrix update strategy. The present self-adaptive time stepping algorithm for the nonlinear transient analysis is proving to be a robust and practical method, aimed at efficiency, effectiveness, and user-friendliness.

The transient response analysis capability is provided in the self-contained solution sequences, SOLs 99 and 129. SOL 99, which is depicted by flow diagrams in Figs. 8.1.1 and 8.1.2, has evolved since Version 62. SOL 129 is a structured DMAP with the same functionality as SOL 99, which was implemented in Version 67 to utilize the new executive system constructs. Although these solution sequences are developed for nonlinear transient response analysis, linear transient response analysis can be performed in these solution sequences in order to take advantage of the automatic time stepping and restart capabilities. The core part of SOL 99 are the nonlinear modules (NLTRD and NLTRD2) which perform the incremental and iterative processes for implicit direct time integration.

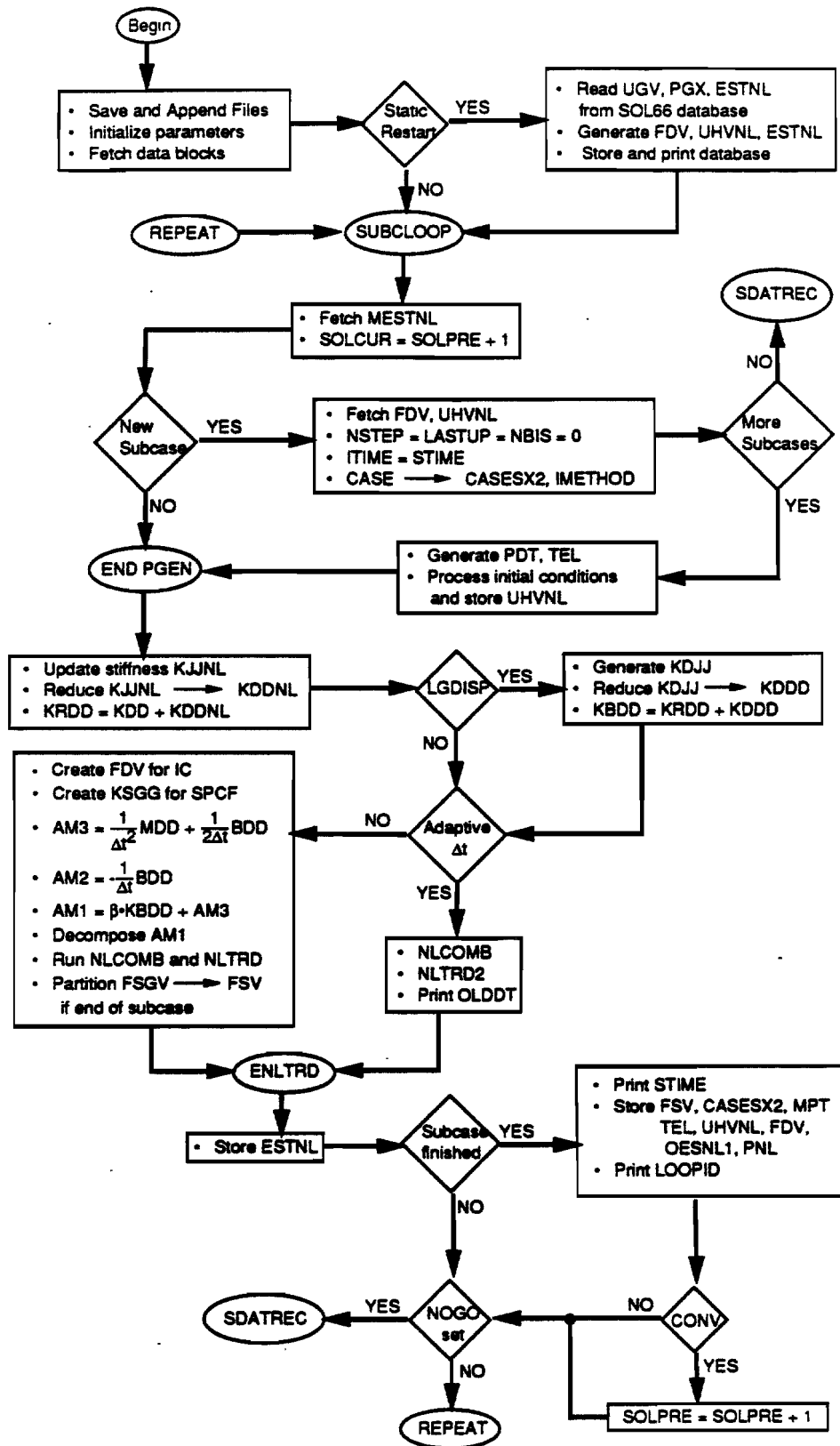


Figure 8.1.1 Flow Diagram for Nonlinear Process in SOL 99

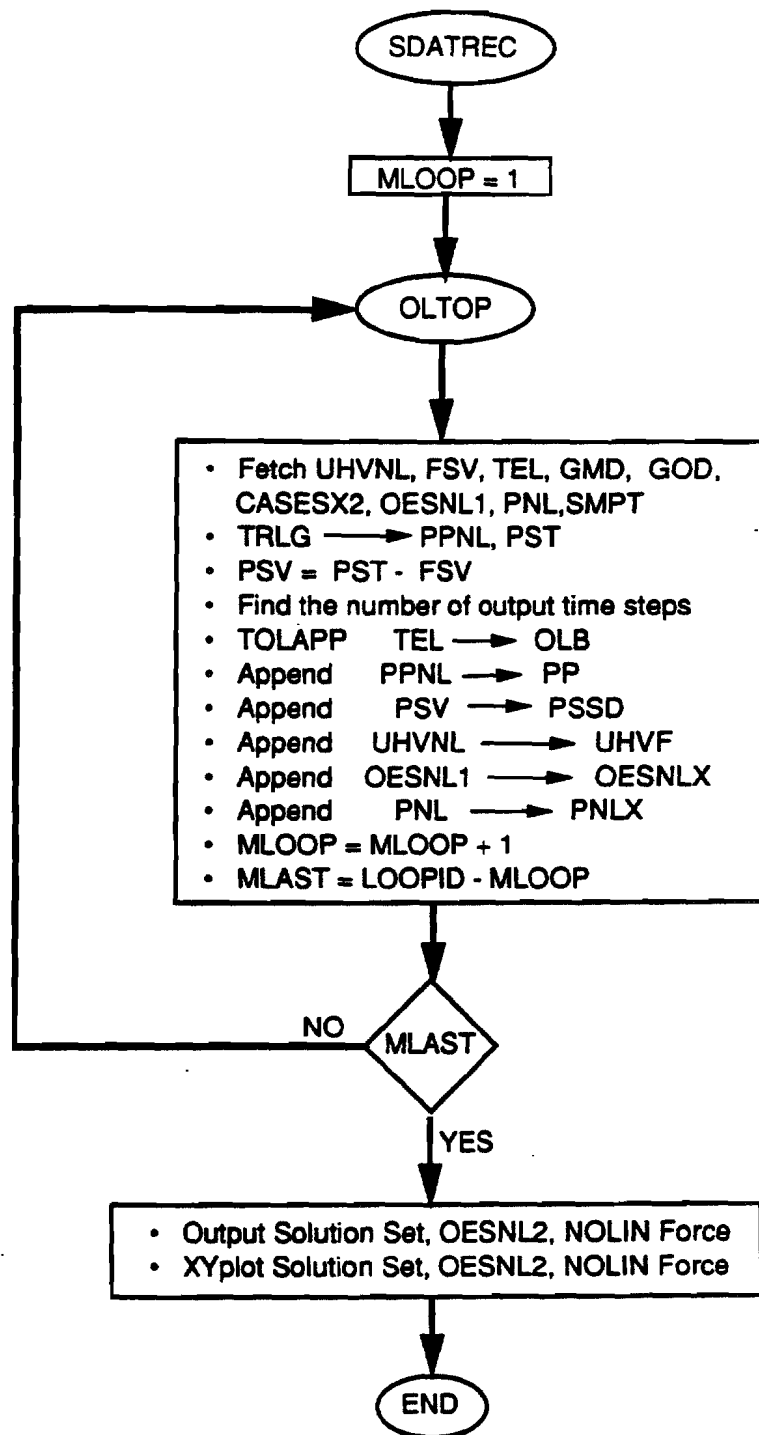


Figure 8.1.2 Flow Diagram for Appending Process in SOL 99

8.2 THREE-POINT METHOD (NLTRD MODULE)

The three-point method is chosen to be compatible with the software's linear transient integration method. The Newmark Beta method based on the two-step integration is combined with the modifications to Newton's method for nonlinear solutions [8.1]. The additional iteration steps provide equilibrium solutions at each time step, thereby guaranteeing stability and accuracy for arbitrary time step size. This method was implemented in the NLTRD module in Version 62. This method can be used by selecting the AUTO or TSTEP method in the TSTEPNL Bulk Data entry.

8.2.1 Basic Equations

Adding dynamic loads to the basic equation given previously, we obtain a load equilibrium error vector, $\{R_n\}$ at time step n by the equation:

$$\{R_n\} = \{\bar{P}_n - M\ddot{u}_n - B\dot{u}_n - \bar{F}_n\} \quad (8.2.1)$$

where $\{\bar{P}_n\}$ = Average load over the time period ($t_{n-1} < t_n < t_{n+1}$)
 $\{\ddot{u}_n\}, \{\dot{u}_n\}$ = Corresponding acceleration and velocity vectors.
 $\{\bar{F}_n\}$ = Average elasto-plastic element total force vector.
 ($\bar{F} = K\bar{u}$ for linear problems.)

The above equation is solved at the reduced (u_d) displacement vector size. The approximation errors due to dynamic reduction methods are not included in the error vector $\{R_n\}$. Applying Newmark's averaging method over a finite time period, $t_{n-1} < t < t_{n+1}$, the static forces are

$$\{\bar{F}\} = \{\beta F(u_{n+1}) + (1 - 2\beta)F(u_n) + \beta F(u_{n-1})\} \quad (8.2.2)$$

where β = Newmark Beta operator and $F(u_n)$ is the nonlinear force due to a generalized displacement vector $\{u_n\}$. An identical definition occurs for $\{\bar{P}_n\}$ from the applied loads at each time step.

From central finite differences, the acceleration and velocity vectors are:

$$\{\ddot{u}_n\} = \frac{1}{\Delta t^2} \{\Delta u_{n+1} - \Delta u_n\} = \frac{1}{\Delta t^2} \{d_n\} \quad (8.2.3)$$

$$\{\dot{u}_n\} = \frac{1}{2\Delta t} \{\Delta u_{n+1} + \Delta u_n\} \quad (8.2.4)$$

where $\{\Delta u_{n+1}\} = \{u_{n+1} - u_n\}$ for small angles and $\Delta t = t_{n+1} - t_n$ is the time step size.

For large angle changes we require that $\{\ddot{u}\}, \{\dot{u}\}$, and $\{\Delta u\}$ are vectors in the global coordinate direction. The vector $\{u\}$ contains gimbals angles requiring a transformation $[R(u_n)]$ such that

$$\{u_{n+1}\} = \{u_n\} + [R_n]\{\Delta u_{n+1}\} \quad (8.2.5)$$

Substituting Eqs. (8.2.2), (8.2.3) and (8.2.4) into Eqs. (8.2.1) defines the load error in terms of the current vectors estimates:

$$\{R_n\} = \{\bar{P}_n\} - \left[\frac{1}{\Delta t^2} M + \frac{1}{2\Delta t} B \right] \{d_n\} - \left[\frac{1}{\Delta t} B \right] \{\Delta u_n\} - \{\beta F_{n+1} + (1-2\beta)F_n + \beta F_{n-1}\} \quad (8.2.6)$$

and

$$\{\Delta u_{n+1}\} = \{\Delta u_n + d_n\} \quad (8.2.7)$$

At any time step the vectors $\{\bar{P}_n\}$, $\{\Delta u_n\}$, $\{F_n\}$, and $\{F_{n-1}\}$ are known. In general the vectors $\{d_n\}$, $\{u_{n+1}\}$, and $\{F_{n+1}\} = \{F(u_{n+1})\}$ must be found either by approximating $\{F_{n+1}\}$ or by an iterative search.

Note that for a linear solution $\{F_n\}$ equals $[K]\{u_n\}$, where $[K]$ is the reduced stiffness matrix. In this case Eqs. (8.2.6) and (8.2.7) may be used to solve for $\{u_{n+1}\}$ directly by setting $\{R_n\} = \{0\}$. With $\beta = 1/3$ this will produce the linear SOL 27 results.

8.2.2 Nonlinear Iterations

For a nonlinear solution, Eqs. (8.2.6) and (8.2.7) may be solved with Newton's method (or a modified version). Using $\{d_n\}$ as the primary solution variable provides the following iteration algorithm.

First assume a linear approximation:

$$\{R_n^{i+1}\} \sim \{R_n\} + \left[\frac{\partial R_n}{\partial d_n} \right] \{d_n^{i+1} - d_n^i\} = 0 \quad (8.2.8)$$

Let

$$[A_1] = - \left[\frac{\partial R_n}{\partial d_n} \right] \quad (8.2.9)$$

Then, the estimated displacement change is

$$\{d_n^{i+1} - d_n^i\} = [A_1]^{-1} \{R_n(d_n^i)\} \quad (8.2.10)$$

The algorithm is identical to the static case except that instead of a tangent stiffness matrix, the left hand side matrix, obtained from Eqs. (8.2.6) and (8.2.8) is

$$[A_1] = \left[\frac{1}{\Delta t^2} M + \frac{1}{2\Delta t} B + \beta \tilde{K} \right] \quad (8.2.11)$$

where $[\tilde{K}] = \frac{\partial F}{\partial u}$ is the current tangent stiffness matrix.

At a new time step the load iterations may be started by assuming that $\{\Delta u_{n+1}^0\} = \{\Delta u_n\}$ and therefore:

$$\{d_n^0\} = \{0\} \quad (8.2.12)$$

However, to be consistent with displacements, we must extrapolate the nonlinear forces:

$$\{F_{n+1}^0\} \cong \{F_n + (F_n - F_{n-1})\} = \{2F_n - F_{n-1}\} \quad (8.2.13)$$

Substituting Eqs. (8.2.12) and (8.2.13) into Eq. (8.2.6) for $i = 0$ results in the first estimate:

$$\{R_n^0\} = \{\bar{P}_n\} + [A_2]\{\Delta u_n\} - \{F_n\} \quad (8.2.14)$$

where

$$[A_2] = [-\frac{1}{\Delta t}B] \quad (8.2.15)$$

Here $[A_2]$ is a precalculated matrix. $\{\Delta u_n\}$ and $\{F_n\}$ are previous results.

For subsequent iterations, ($i > 0$) Eq. (8.2.6) becomes:

$$\{R_n^i\} = \{R_n^{i-1}\} - [A_3]\{d_{n+1}^i\} - \beta\{F_{n+1}^i - F_{n+1}^0\} \quad (8.2.16)$$

where

$$[A_3] = [\frac{1}{\Delta t^2}M + \frac{1}{2\Delta t}B] \quad (8.2.17)$$

At each time step the code will iterate on Eq. (8.2.10) and Eq. (8.2.16) until $\{R_n^i\}$ passes the convergence tests or the number of passes reaches an iteration limit. With a single step, $i = 1$, calculating only R_0 , the results will be identical to the existing NOLIN_i results in the software. For faster convergence the iterations may continue, the matrices may be updated, and/or the time step size may be reduced.

8.2.3 Stability Analysis

In transient analysis, two types of instability could occur. The first is the familiar nonlinear load iteration divergence which also occurs in static analysis. The second is the divergence which grows with time in the transient integration. Both instabilities are caused by uncorrected nonlinear equilibrium errors.

A convenient method, from von Neumann, for analyzing the stability limits is to assume that the nonlinear forces are nearly linear and the error vector has a constant convergence. It is assumed that the nonlinearity has a first order approximation:

$$\{F(u)\} \sim [K^r + \Delta K^{NL}]\{u\} \quad (8.2.18)$$

where ΔK^{NL} is the difference between the tangent stiffness matrix and its approximation K^r . The error vectors, $\{R\}$, are assumed to grow at the rate λ , defined as:

$$\{R^{i+1}\} = \lambda\{R^i\} \quad (8.2.19)$$

Note that if $|\lambda| > 1$ the system will be defined as unstable.

After lengthy calculations we may summarize the various criteria, assuming that the matrices are reduced to equivalent scalar modal quantities, the criterion for stable solutions are:

- For time steps with converged static iterations:

$$1/2 > \beta \geq 1/4 \quad (8.2.20)$$

This is the same criteria as linear analysis.

- For static load iterations:

$$\beta \Delta K^{NL} \leq \beta K^r + \frac{1}{2\Delta t} B + \frac{1}{\Delta t^2} M \quad (8.2.21)$$

This states that the mass and damping add to the effective linear stiffness $[K^r]$ and improve the stability.

- For time step integration with no intermediate static iterations, as in the standard Newmark Beta method:

$$\Delta K^{NL} \leq (4\beta - 1)K^r + \frac{4}{\Delta t^2} M \quad (8.2.22)$$

This restriction is more severe than criteria (8.2.21) above, proving that the internal iterations are more stable than the Newmark integration.

In summary, the method will have fewer divergence problems than either the static nonlinear solution or the single step transient nonlinear methods. The better stability of the method and the capability to use larger time steps outweighs the cost of a few internal iterations on the static element forces.

8.2.4 DMAP Interface of NLTRD Module (Version 66)

```
NLTRD      CASESX2,MESTNL,PDT,YS,KRDD,ELDATA,
            KELMNL,AM1LL,GM,MPTS,DIT,AM1UU,DLT1,CSTMS,
            BGPPTS,SILS,SETD,AM2,AM3,NLFT,KSGG/
            UHVNLFSGV,ESTNL,FDV,OESNL1,PNL,TEL/
            V,Y,BETA/S,N,CONV/S,N,RSTIME/S,N,NEWP/S,N,NEWK//
            S,N,NSTEP/V,Y,LGDISP/S,N,MPTSIZE/S,N,ITERIDS//
            S,N,KTIME/S,N,LASTUPD/S,N,NOGONL//V,Y,MAXLP $      ++V66
```

Input Data Blocks:

CASESX2 Case control data.

MESTNL Nonlinear element summary table

PDT Applied loads, one column vector for each time step.

YS Enforced displacement.

KRDD Reduced stiffness in d-set

ELDATA Element data used to generate OESNL file.

KELMNL Nonlinear element stiffness matrix required for nonlinear force (F) computation.

AM1LL Lower triangular factor for nonlinear elements

GM Transformation matrix for multipoint constraint reduction.

MPTS Material properties table containing TSTEPNL data.

DIT Direct input tables.

AM1UU Upper triangular matrix for nonlinear elements.

DLT1 Dynamic loads table for follower forces.

CSTMS Coordinate system transformation table.

BGPPTS Basic GRID point definition table.

SILS Scalar index list.

USETD Displacement set definition table.

AM2 Damping matrix in d-set divided by $-\Delta t$.

AM3 Matrix consisting of mass and damping terms: $\frac{1}{\Delta t^2} M + \frac{1}{2\Delta t} B$.

NLFT Nonlinear transient loads for NOLIN.

KSGG Partition of stiffness matrix for forces of constraints.

Output Data Blocks:

UHVNL Input/output, displacement and velocity vectors at converged time steps during the direct integration, UHVNL consists of output displacement, velocity, and acceleration vectors at output time steps upon completion of the subcase.

FSGV Nonlinear element forces at constrained points, equivalent to $[K_{ss}]\{Y_s\}$ at output time steps.

ESTNL Input/output, nonlinear Element Summary Table at the last converged step.

FDV Input/output, Two column vectors $[F_n, \dot{F}_n]$ at the end of subcases. Element forces and the current and the preceding steps (F_{n-1}, F_n) for matrix update.

OESNL1 Output stress data for nonlinear elements.

PNL Nonlinear load vector appended at output time steps.

TEL Appended output time step list. Purged at the beginning of subcases.

Parameters:

BETA Input-real, integration parameter, default = 1/3.

CONV Input/output-integer, value from the last iteration for an input (0 for the initial iteration), set to -1 if not converged, +1 if converged.

STIME Input/output-real, starting time or accumulated time used for restarts by user input.

NEWP Input/output-integer, set to +1 for new subcase, -1 for no subcase switch.

NEWK Input/output-integer, set to +1 or +3 for stiffness update, -1 otherwise.

NSTEP Input/output-integer, current time step position for subcase, set to 0 at the beginning of the subcase.

LGDISP Input-integer, set to +1 for geometric nonlinearity, -1 otherwise.

MPTSIZE Input/output-integer, the amount of open-core words required to hold the material data, initialized to 0 in MPL and retained during the problem solution.

ITERIDS Input/output-integer, iteration count within a time step.

KTIME Input/output-integer, time-to-go value at the beginning of K update in the DMAP loop, on output from the module, a negative value is stored, denoting the increment of time between the K computation and the start of the module.

LASTUPD Input/output-integer, the time step number of the last stiffness update. The value 0 is used to indicate if the stiffness update is performed due to the GAP element during the iteration.

NOGONL Output-integer, set to +1 to continue, -1 to terminate.

MAXLP Input-integer, maximum limit allowed for element relaxation iteration and the material subincrement processes.

8.3 TWO-POINT METHOD (NLTRD2 MODULE)

The multistep implicit integration method [8.1] is not suitable for the adaptive method. In addition to the extra memory requirement, the multistep integration scheme also introduces difficulties in the starting algorithm, ending time step, and restarting. These difficulties become more complicated and cumbersome when the adaptive time stepping is employed due to the unequal time intervals involved. Therefore, the two-point integration method [8.2], or one-step integration, is introduced in Version 66 with module NLTRD2. This method can be selected by specifying the ADAPT method in the TSTEPNL Bulk Data entry.

8.3.1 Newmark Integration

For the adaptive scheme, Newmark's method [8.3] is employed with the two-point recurrence formula for one-step integration, i.e.,

$$\{U_{n+1}\} = \{U_n\} + \Delta t \{\dot{U}_n\} + \frac{1}{2} \Delta t^2 \{\ddot{U}_n\} + \beta \Delta t^2 \{\ddot{U}_{n+1} - \ddot{U}_n\} \quad (8.3.1)$$

and

$$\{\dot{U}_{n+1}\} = \{\dot{U}_n\} + \Delta t \{\ddot{U}_n\} + \gamma \Delta t \{\ddot{U}_{n+1} - \ddot{U}_n\} \quad (8.3.2)$$

where $\{U\}$, $\{\dot{U}\}$, $\{\ddot{U}\}$ and Δt denote displacement, velocity, acceleration and the time step increment, respectively. The subscript n designates the time step and the parameters (β and γ) are to be determined. An equilibrium equation to be satisfied at time step ($n+1$) is:

$$M \{\ddot{U}_{n+1}\} + C \{\dot{U}_{n+1}\} + \{F(U_{n+1})\} = \{P_{n+1}\} \quad (8.3.3)$$

where M and C denote mass and damping matrices, and $\{F\}$ and $\{P_{n+1}\}$ denote internal and external forces, respectively. An incremental/iterative form of Eq. (8.3.3) can be recast for the i -th iteration (denoted by superscripts) as:

$$M \{\ddot{U}_{n+1}^{i+1}\} + C \{\dot{U}_{n+1}^{i+1}\} + K_{n+1}^i \{\Delta U^{i+1}\} = \{P_{n+1} - F_{n+1}^i\} \quad (8.3.3a)$$

satisfying

$$\{F_{n+1}^{i+1}\} \simeq \{F_{n+1}^i\} + K_{n+1}^i \{U_{n+1}^{i+1} - U_{n+1}^i\}$$

where

$$K_{n+1}^i = \left[\frac{\partial F}{\partial U} \right]_{n+1}^i$$

By solving Eqs. (8.3.1) and (8.3.2) for the velocity and the acceleration vectors, these variables at the i -th iteration of the time step ($n+1$) can be expressed in terms of displacements as shown below:

$$\{\dot{U}_{n+1}^{i+1}\} = \frac{\gamma}{\beta \Delta t} \{U_{n+1}^{i+1} - U_n\} + \left(1 - \frac{\gamma}{\beta}\right) \{\dot{U}_n\} + \left(1 - \frac{\gamma}{2\beta}\right) \Delta t \{\ddot{U}_n\} \quad (8.3.4)$$

$$\{\ddot{U}_{n+1}^{i+1}\} = \frac{1}{\beta \Delta t^2} \{U_{n+1}^{i+1} - U_n\} - \frac{1}{\beta \Delta t} \{\dot{U}_n\} - \left(\frac{1}{2\beta} - 1\right) \{\ddot{U}_n\} \quad (8.3.5)$$

where

$$\{U_{n+1}^{i+1}\} = \{U_{n+1}^i\} + \{\Delta U^{i+1}\}$$

The governing equation for the Newton-Raphson iteration is obtained by substituting Eqs. (8.3.4) and (8.3.5) into Eq. (8.3.3a) to obtain:

$$\left[\frac{1}{\beta \Delta t^2} M + \frac{\gamma}{\beta \Delta t} C + K_{n+1}^i \right] \{\Delta U^{i+1}\} = \{R_{n+1}^i\} \quad (8.3.6)$$

where the residual load vector $\{R_{n+1}^i\}$ is expressed as

$$\begin{aligned} \{R_{n+1}^i\} = & \{P_{n+1} - F_{n+1}^i\} - \frac{M}{\beta \Delta t^2} \{U_{n+1}^i - U_n - \Delta t \dot{U}_n\} + \left(\frac{1}{2\beta} - 1\right) M \{\ddot{U}_n\} \\ & - \frac{\gamma}{\beta \Delta t} C \{U_{n+1}^i - U_n\} + \left(\frac{\gamma}{\beta} - 1\right) C \{\dot{U}_n\} - \left(1 - \frac{\gamma}{2\beta}\right) \Delta t C \{\ddot{U}_n\} \end{aligned} \quad (8.3.7)$$

Notice that the tangential stiffness matrix, K_{n+1}^i , in Eq. (8.3.6) may be replaced by K_n (modified Newton's iteration) or \tilde{K} which is a stiffness matrix evaluated at any preceding time step.

8.3.2 Adaptation of Newmark's Method

An alternative expression for the load vector can be derived for $\gamma=0.5$ by introducing

$$M \{\ddot{U}_n\} + C \{\dot{U}_n\} = \{P_n - F_n\} \quad (8.3.8)$$

i.e.,

$$\begin{aligned} \{R_{n+1}^i\} = & \{P_{n+1} - F_{n+1}^i\} - \frac{M}{\beta \Delta t^2} \{U_{n+1}^i - U_n - \dot{U}_n \Delta t\} - \frac{C}{2\beta \Delta t} \{U_{n+1}^i - U_n\} \\ & + \left(\frac{1}{2\beta} - 1\right) \{P_n - F_n\} - \left(1 - \frac{1}{4\beta}\right) \Delta t C \{\ddot{U}_n\} \end{aligned} \quad (8.3.7a)$$

By virtue of Eq. (8.3.8) the residual load error at each time step is effectively carried over to the next step and the error propagation is reduced, while the computation is significantly simplified. Notice that the acceleration term in Eq. (8.3.7a) conveniently disappears when the parameter β has the recommended value of 0.25, i.e.,

$$\begin{aligned} \{R_{n+1}^i\} = & \{P_{n+1} - F_{n+1}^i\} + \frac{4}{\Delta t} M \{\dot{U}_n\} + \{P_n - F_n\} \\ & - \left[\frac{4}{\Delta t^2} M + \frac{2}{\Delta t} C \right] \{U_{n+1}^i - U_n\} \end{aligned} \quad (8.3.7b)$$

If follower forces (i.e., applied forces change as a function of displacements) are involved, however, the external load $\{P_{n+1}\}$ should be updated at every iteration by

$$\{P_{n+1}^i(U)\} = \{P_{n+1}(0)\} + \{f_F^i(U, t_{n+1})\} + \{N^i(U, t_{n+1})\} \quad (8.3.9)$$

where N^i represents NOLINi forces at the i -th iteration, and $f_F^i(U, t_{n+1})$ is a correction for follower forces as expressed below:

$$f_F^i(U, t_{n+1}) = g(t_{n+1})\{p^i(U) - p(0)\}$$

where $g(t)$ is a time dependent forcing function specified in the TLOADi entries and $p^i(U)$ denotes static follower forces. It is noted that the residual vector in Eq. (8.3.7b) can be computed without computing velocities and accelerations at every iteration.

8.3.3 Start-up Algorithm

The first iteration is performed with initial conditions

$$\{U_{n+1}^0\} = \{U_n\} \quad \text{and} \quad \{F_{n+1}^0\} = \{F_n\} \quad (8.3.10)$$

which satisfy

$$\{F_{n+1}^0\} \simeq \{F_n\} + K_n\{U_{n+1}^0 - U_n\}$$

It is noted that the initial equilibrium is satisfied by virtue of Eq. (8.3.8) without the need for computing initial acceleration \ddot{U}_0 , i.e.,

$$M\{\ddot{U}_0\} + C\{\dot{U}_0\} = \{P_0 - F_0\} \quad (8.3.8a)$$

In this method of direct integration (based on the average acceleration scheme), the Eqs. (8.3.4), (8.3.5) and (8.3.6) are reduced to:

$$\{\dot{U}_{n+1}\} = \frac{2}{\Delta t} \{U_{n+1} - U_n\} - \{\dot{U}_n\} \quad (8.3.4a)$$

$$\{\ddot{U}_{n+1}\} = \frac{4}{\Delta t^2} \{U_{n+1} - U_n\} - \frac{4}{\Delta t} \{\dot{U}_n\} - \{\ddot{U}_n\} \quad (8.3.5a)$$

$$\left[\frac{4}{\Delta t^2} M + \frac{2}{\Delta t} C + \tilde{K} \right] \{\Delta U^{i+1}\} = \{R_{n+1}^i\} \quad (8.3.6a)$$

Since the acceleration vector does not appear in Eq. (8.3.7b), $\{\ddot{U}\}$ needs to be computed only for output purposes. However, Eq. (8.3.5a) requires an initial value, $\{\ddot{U}_0\}$. After some numerical experiments, it has been determined that accelerations are most effectively evaluated by the central difference method using velocities at three consecutive time steps at the end of the integration process. For irregular time step intervals, the modified expression for the acceleration is

$$\{\ddot{U}_n\} = \frac{1}{\Delta t_n + \Delta t_{n+1}} \left[\frac{\Delta t_n}{\Delta t_{n+1}} \{\dot{U}_{n+1}\} + \left(\frac{\Delta t_{n+1}}{\Delta t_n} - \frac{\Delta t_n}{\Delta t_{n+1}} \right) \{\dot{U}_n\} - \frac{\Delta t_{n+1}}{\Delta t_n} \{\dot{U}_{n-1}\} \right] \quad (8.3.5b)$$

for $0 < n < N$, with N being the last time step. The initial and the final accelerations, $\{\ddot{U}_0\}$ and $\{\ddot{U}_N\}$, are evaluated by an extrapolation using Eq. (8.3.5a).

8.3.4 DMAP Interface of NLTRD2 Module (Version 67)

NLTRD2 CASESX2, PDT, YS, ELDATA,
KELMNL, KDD, GM, MPT, DIT, KBDD, DLT1, CSTMS,
BGPPTS, SILS, USETD, BDD, MDD, NLFT/ UHVN,
FSV, ESTNL, FDV, OESNL1, PNL, TEL, MUHVNL, MESTNL/
V,N,KRATIO=(1.,0.)/ S,N,CONV/ S,Y,STIME/
S,N,NEWP/ S,N,NEWDT=0/ S,N,OLDDT/ S,N,NSTEP/
V,Y,LGDISP/ S,N,MPTSIZE/ S,N,ITERIDS/ V,N,ITIME/
S,N,KTIME/ S,N,LASTUPD/ S,N,NOGONL/ S,N,NBIS/
V,Y,MAXLP=5/ V,Y,TSTATIC=-1/ V,N,LANGLES/ V,Y,NDAMP \$

Input Data Blocks:

CASESX2 Case control data.

PDT Applied loads, one column vector for each time step.

YS Enforced displacement.

ELDATA Element data used to generate OSENL file.

KELMNL Nonlinear element stiffness matrix required for nonlinear force (F) computation.

KDD Stiffness matrix for linear portion in d-set.

GM Transformation matrix for multipoint constraint reduction.

MPT Material properties table containing TSTEPNL data.

DIT Direct input tables.

KBDD Tangential stiffness of residual structure in d-set.

DLT1 Dynamic load table for follower forces.

CSTMS Coordinate system transformation table.

BGPPTS Basic GRID point definition table.

SILS Scalar index list.

USETD Displacement set definition table.

BDD Damping matrix in d-set.

MDD Mass matrix in d-set.

NLFT Nonlinear transient loads for NOLIN.

Output Data Blocks:

UHVNL Input/output, displacement and velocity vectors at converged time steps during the direct integration, UHVNL consists of output displacement, velocity, and acceleration vectors at output time steps upon completion of the subcase.

FSV Nonlinear element forces at constrained points, equivalent to $[K_{ss}]\{Y_s\}$ at output time steps.

ESTNL Input/output, nonlinear Element Summary Table at the last converged step.

FDV Input/output, Two column vectors $[F_n, P_n]$ at the last converged step.

OESNL1 Output stress data for nonlinear elements.

PNL Nonlinear load vector appended at output time steps.

TEL Appended output time step list. Purged at the beginning of subcases.

MUHVNL Input/output, displacement and velocity vectors (2 columns) for K update. It is deleted at the beginning of a subcase ($MUHVNL \equiv UHVNL$).

MESTNL Input/output, nonlinear Element Summary Table for K update.

Parameters:

KRATIO Input/output-complex, stiffness ratio to be used for time step adjustment, initialized to (1.,0.).

CONV Input/output-integer, value from the last iteration for an input (0 for the initial iteration), set to -1 if not converged, +1 if converged.

STIME Input/output-real, starting time or accumulated time used for restarts by user input.

NEWP Input/output-integer, set to +1 for new subcase, -1 for no subcase switch.

NEWDT Input/output-integer, used to detect if the time step is adjusted in the previous subcase. The value is set at the end of each subcase: 1 if ADJUST>0 and 0 if ADJUST=0.

OLDDT Input/output-real, time step increment used in the previous iteration or time step; to be used after the matrix update or subcase switch.

NSTEP Input/output-integer, current time step position for subcase, set to 0 at the beginning of the subcase.

LGDISP Input-integer, set to +1 for geometric nonlinearity, -1 otherwise.

MPTSIZE Input/output-integer, the amount of open-core words required to hold the material data, initialized to 0 in MPL and retained during the problem solution.

ITERIDS Input/output-integer, iteration count within a time step.

ITIME Input-real, initial time at the beginning of a subcase.

KTIME Input/output-integer, time-to-go value at the beginning of K update in the DMAP loop, on output from the module, a negative value is stored, denoting the increment of time between the K computation and the start of the module.

LASTUPD Input/output-integer, the time step number of the last stiffness update. The value 0 is used to indicate if the stiffness update is performed due to the GAP element during the iteration.

NOGONL Output-integer, set to +1 to continue, -1 to terminate.

NBIS Input/output-integer, current count of bisection.

MAXLP Input-integer, maximum limit allowed for element relaxation iteration and the material subincrement processes.

TSTATIC Input-integer, default=-1, +1 allows static analysis by ignoring inertia and damping forces.

LANGLES Input-integer, used to select the rotation vector approach to large rotations by a value of 2. The default (= 1) option is the gimbal angle approach.

NDAMP Input-real, default= 0.025. Introduces numerical damping with a value greater than zero, which is used in

$$\{\dot{u}_{n+1}\} = \frac{2 - 2\eta}{\Delta t} \{u_{n+1} - u_n\} - (1 - 2\eta) \{\dot{u}_n\}$$

where η represents NDAMP.

8.4 ADAPTIVE TIME STEPPING

8.4.1 Introduction

It is desirable to have a fully automated method of time integration which renders effective and accurate solutions of nonlinear problems. This premise is wishful thinking, because such a method is yet to be found. Bergan et al. [8.4] have shone some light on this goal, but their method has inherent flaws with regard to nonlinear problems. The dynamic response characteristics of the structure may change due to the nonlinearity (geometric or material) or the mode of excitation. When the mode of response changes, the time step size should be adjusted, requiring a decomposition. However, the stiffness matrix does not have to be reformulated unless it is so required due to nonlinearity. The authors [8.4] did not distinguish between these two cases, which are intrinsically different and therefore require distinct treatment as discussed below. Nevertheless, their idea is salient and can be applied to nonlinear problems with some modification to cope with the difficulties in adjusting the time step size which fluctuates with the deformation pattern. The deficiency of this method stems from the time lag, which is coped with by combining it with the bisection method in a coordinated manner. Divergence activates the bisection process which is valid within the time interval in which the process is activated. After the solution is achieved, the usual procedure will be resumed for the next time step.

In the present implementation of the automatic time stepping for nonlinear applications, the adjustment is designed to be adaptive to the severity of the nonlinearity in the problem by monitoring the changes in the stiffness. However, difficulties in the automatic time stepping have been discovered, when plasticity or GAP elements are involved. In such cases an undesired time step adjustment can be caused by a drastic change in the stiffness, which may be a temporary situation. A filtering scheme has been devised to suppress the effects of the spurious mass or stiffness in the automatic time step adjustments. In addition, the bisection method is chosen to be activated only at the time when divergence occurs. The bisection process is coordinated with the stiffness matrix update so that changes in nonlinearities are properly reflected while the bisection is in progress.

8.4.2 Automatic Time Stepping Algorithm

The proper time step size, required for accuracy and efficiency, changes continuously in the transient dynamic environment. This size can be predicted from the dominant frequency, ω_n , in the incremental deformation pattern based on the Rayleigh's method [8.4]. The dominant frequency is estimated at every converged time step using the displacement increment (as an assumed vibration shape) and internal forces by

$$\omega_n^2 = \frac{\{\Delta U_n\}^T K \{\Delta U_n\}}{\{\Delta U_n\}^T M \{\Delta U_n\}} \simeq \frac{\{\Delta U_n\}^T \{F_n - F_{n-1}\}}{\{\Delta U_n\}^T M \{\Delta U_n\}} \quad (8.4.1)$$

where $\{\Delta U_n\} = \{U_n - U_{n-1}\}$, and $\{F_n - F_{n-1}\}$ represents a change in the internal forces which are readily available in memory. A provision should be made to skip the time step adjustment if $\{\Delta U_n\}$ approaches zero, which is effectively detected by

$$\frac{\|\dot{U}_n\|}{\|\dot{U}\|_{max}} < \text{UTOL (defaulted to 0.1)}$$

where $\|\dot{U}\|_{max}$ is the maximum value of the norms computed throughout the preceding steps.

The next time step can be predicted by multiplying the current time step by a scaling factor r , i.e.,

$$r = \frac{\Delta t_{n+1}}{\Delta t_n} = \frac{1}{m} \frac{2\pi}{\omega_n} \frac{1}{\Delta t_n} \quad (8.4.2)$$

where it is assumed that the dominant period is marched through with m steps. However, continuous adjustment of the step size at every time step is not a viable option because it requires too frequent matrix decomposition. To prevent thrashing, a stepping function $f(r)$ is introduced such that

$$\Delta t_{n+1} = f(r) \Delta t_n \quad (8.4.3)$$

where

$$\begin{array}{lll} f = 0.25 & \text{for} & r < 0.5R_b \\ f = 0.5 & \text{for} & 0.5R_b \leq r < R_b \\ f = 1.0 & \text{for} & R_b \leq r < 2 \\ f = 2.0 & \text{for} & 2 \leq r < 3/R_b \\ f = 4.0 & \text{for} & r \geq 3/R_b \end{array}$$

with R_b being a user specified parameter defaulted to 0.75.

Since Δt is allowed to be quadrupled or quadrisected at each time step, the time step will be quickly adapted if the original Δt is not pertinent. While the adjustment algorithm is based on the mode of response, however, Δt may have to be bounded to represent the external loads properly, e.g., the impulsive loading case. Upper and lower bounds of Δt will be defined as

$$\Delta t_{max} = \text{MAXR} * \text{DT} \quad (8.4.4)$$

and

$$\Delta t_{min} = \text{MIN} (\text{DT}/2^{**}\text{MAXBIS}, \text{DT}/\text{MAXR})$$

where DT is the user specified step size, and MAXR and MAXBIS are additional parameters to be specified in the TSTEPNL card. If the predicted time step size falls outside the bounds, the time step will be set to the limit.

When the automatic time stepping option is selected, the user-specified DT will be used only as an initial time step size and the automatic time stepping scheme will be activated. Since the automatic time stepping scheme ignores user specified DT in time marching, a termination criterion is required to wrap up the subcase. The last step may be determined by terminating

the normal time stepping scheme when the currently processed step is close to the last step, i.e., no further step will be taken if

$$\sum_n^N \Delta t_n + \frac{\Delta t_N}{2} \geq \text{DT*NDT} \quad (8.4.5)$$

where the right hand side represents the total duration for a subcase. The adjusted time step will remain effective across the subcases. Notice that matrix decomposition is required whenever the time step is adjusted.

It has been observed that the criterion in Eq. (8.4.1) is dictated by spurious mass or stiffness terms when artificially low or high masses or stiffnesses are used. Large mass or large spring approach to the base motion is an example. It could be caused by massless point or GAP elements as well. This undesired effect should be eliminated, which is accomplished by checking the “apparent frequency” of each degree-of-freedom by comparing components of $\{F_n - F_{n-1}\}$ and $\{M \Delta U_n\}$, i.e.,

$$\omega_i^2 = \text{ABS} \left[\frac{\{F_n - F_{n-1}\}_i}{\{M \Delta U\}} \right]$$

where ω_i is the apparent angular frequency of the i-th DOF. The central idea is to eliminate the effects of the DOF that has extremely small or large apparent frequency compared with the dominant frequency. The bounds are determined based on the reference frequency as follows:

$$\Delta u_i = 0 \quad \text{if} \quad \omega_i^2 < 10^{-3} * \omega_{ref}^2 \quad \text{or} \quad \omega_i^2 > 10^3 * \omega_{ref}^2$$

where ω_{ref} is the dominant frequency previously computed. The dominant frequency should then be computed by

$$\omega_n^2 = \frac{\{\Delta U_n^*\}^T \{F_n - F_{n-1}\}}{\{\Delta U_n^*\}^T M \{\Delta U_n\}} \quad (8.4.1a)$$

where $\{\Delta U_n^*\}$ is a modified vector with zeros inserted for the undesired DOF. Undesired time step adjustment due to the drastic change in stiffness of the plastic deformation is also properly suppressed by this method. This is because the plastic deformation casts implications similar to the GAP in the automatic time stepping algorithm.

It has also been observed that m of 20, in Eq. (8.4.2), is adequate if the deformation remains linear but should be greater for nonlinear problems. Severity of the nonlinearity may be represented by the stiffness ratio defined as:

$$\lambda = \frac{\Delta U^T \{F_n - F_{n-1}\}}{\Delta U^T \tilde{K} \Delta U} \quad (8.4.6)$$

where \tilde{K} represents the last updated stiffness matrix. The value of m can be a function of the stiffness ratio. This is implemented as a default option in the following manner:

$$\begin{aligned} \lambda^* &= \sqrt{\lambda^2} & \text{if} & \quad \lambda^2 \geq 1 \\ \lambda^* &= \frac{1}{\sqrt{\lambda^2}} & \text{if} & \quad \lambda^2 < 1 \end{aligned}$$

then

$m = 20$	for	$\lambda^* < 5$
$m = 40$	for	$5 \leq \lambda^* < 1000$
No Adjust	for	$1000 \leq \lambda^*$

This option will be superseded if the user specifies a value for m (MSTEP).

8.4.3 Bisection Algorithm

The automatic time stepping option is implemented to adapt the time step size primarily to the dynamic environment based on the response pattern. This method, however, is not likely to alleviate difficulties due to the change in nonlinearity or divergence. Such difficulties are resolved by using the bisection method. The material processing routine can detect a change in nonlinearity. The bisection is activated by a material routine if an incremental stress ($\Delta\bar{\sigma}$) exceeds the specified tolerance (e.g., 20 % of the effective stress). The bisection could also be initiated when a divergence is detected or anticipated. In addition, the bisection is activated when the maximum number of iterations allowed for an increment is reached without convergence. The divergence criterion is defined as

$$\text{NDIV} > \text{MAXDIV} \quad (8.4.7)$$

where MAXDIV is a user-specified parameter (defaulted to 2) and NDIV is an internal variable which is initialized to zero at the beginning of a new time step and incremented as follows:

$$\text{NDIV} = \text{NDIV} + 2 \quad \text{if} \quad E_1^i > 1 \quad \text{or} \quad E_1^i < -10^{+12}$$

and

$$\text{NDIV} = \text{NDIV} + 1 \quad \text{if} \quad -10^{12} < E_1^i < -1 \quad \text{or if} \quad |E_2^i| > 1$$

with

$$E_1^i = \frac{\{\Delta U^i\}^T \{R^i\}}{\{\Delta U^i\}^T \{R^{i-1}\}} \quad \text{and} \quad E_2^i = \frac{E_p^i}{E_p^{i-1}}. \quad (8.4.8)$$

In addition, reaching MAXITER without convergence will be treated as a diverging case.

The stiffness will be updated at the last converged step when the first divergence is encountered. Then, an iteration is tried without bisection. Bisection normally begins upon the second detection of divergence within a Δt , i.e.,

$$\Delta t_1 = \frac{1}{2} \Delta t.$$

Bisection continues until the solution converges, i.e.,

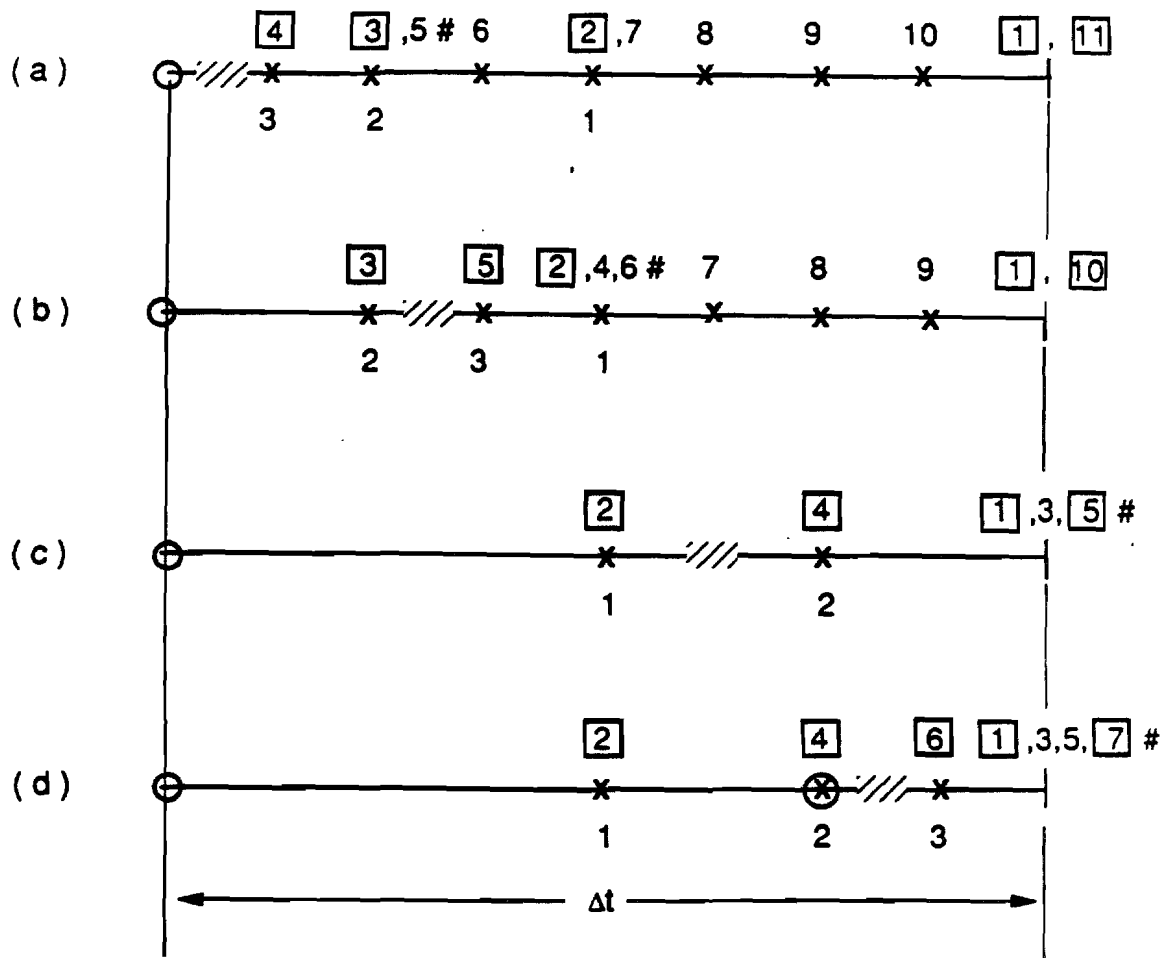
$$\Delta t_k = \frac{1}{2} \Delta t_{k-1} = \frac{1}{2^k} \Delta t$$

where k is a bisection count. Once the bisection is successful (rendering a converged solution), the integration proceeds to the next time step with the same increment (Δt_k). Further bisection may be required after achieving converged solution at the intermediate time steps as shown in Fig. 8.4.1.

If no further bisection is required and the solutions converge without any difficulty, some recovery process such as doubling time step may be activated. However, any adjustment to the time step would require additional matrix decomposition which offsets the advantages of adjusting time step size. Therefore, it has been determined to continue stepping through the specified time interval to completion. The normal process should be resumed when the user-specified time step at which bisection is activated is fully processed.

When the automatic time stepping is combined with the bisection method and if the bisection is activated, the automatic adjustment procedure will be deferred until the divergence is trapped and the solution process is stabilized. The solution process is considered stabilized when two successive solutions have converged without requiring further bisection. This is the time when the automatic time step adjustment is resumed as illustrated in Fig. 8.4.1.

The maximum number of bisections is limited by a user-specified parameter MAXBIS (defaulted to 5). The bisection process is activated on an as-needed basis. Users also have an option to suppress bisection by specifying MAXBIS=0. If the bisection is required more than MAXBIS times, the solution process will continue without bisection by activating the reiteration procedure, by which the same iteration process is repeated to find the best attainable solution. The best attainable solution is a solution obtained from the iteration that yields the least average error. The average error is defined by averaging relative errors in displacements, loads and energy. If the i -th iteration yields the least average error, the reiteration procedure will end at the i -th iteration and the normal time stepping procedure will be resumed.



Notes:

1. The baseline represents a duration of a time step before bisection starts.
2. Numbers above the baseline denote a sequence of trial solutions.
3. X denotes converged solutions during the time step Δt .
4. \square denotes the step which triggers decomposition.
5. O denotes reference point for k update.
6. $///$ denotes critical period.
7. Automatic time stepping resumes at # if selected.
8. Numbers below the baseline denote a sequence of bisections.

Figure 8.4.1 Illustration of the Bisection Strategy

8.5 EXPEDIENT ITERATION STRATEGIES

The dynamic tangent matrix may change for the following reasons:

- Stiffness (K) changes due to geometric, material and/or kinematic nonlinearity.
- Bisection or time-step adjustment causes changes in the effective stiffness.

While the second case only requires decomposition, the first case requires reformulation of the new stiffness matrix as well. Examples of various cases for the update relative to the bisection strategy are illustrated in Fig. 8.4.1(a-d). The adaptive method is based on the following matrix update strategies:

- Stiffness is updated at the previously converged position when the divergence is detected for the first time at a given time step.
- Stiffness matrix update is allowed only once within the same time step.
- If the bisection is in progress, additional K-updates are allowed within a given Δt at every KSTEP-th converged bisection, where KSTEP is a user-specified parameter in the TSTEPNL entry. See Figure 8.4.1(d) for an example.
- Decomposition is performed within the module at every bisection or time step adjustment.
- Whenever the decomposition is performed, the iteration starts from the previously converged position.
- The stiffness matrix is updated at the beginning of each subcase.

The BFGS update [8.5] and the line search [8.6] processes are fully operational during the iteration. Quasi-Newton (QN) vectors are continuously accumulated up to MAXQN (user-specified parameter) pairs until the new stiffness matrix is evaluated. Once the number of QN vector pairs reaches MAXQN (defaulted to 10), the QN vectors will be updated selectively based on the condition number of the QN update matrix. As the condition number approaches unity, the QN update matrix approaches identity matrix, which causes little effect on the matrix to be modified. Therefore, the QN vector pair, which renders the condition number closest to unity, is replaced or eliminated when a new candidate QN vector pair becomes available. As a requirement for the reiteration procedure, the QN vectors to be accumulated are stored on a temporary basis until a converged solution is obtained. All the accumulated QN vectors are purged if the stiffness matrix update or the decomposition (due to the change in Δt) is scheduled.

The iterative process for time step (n+1) can be summarized as follows:

1. Given at the beginning of the time step are $\Delta t_n, \omega_{ref}, [M], [C], \{P_n\}, \{\Delta F_n\}, \{F_n\}, \{U_n\}, \{\dot{U}_n\}$, and the triangularized matrices (LDL^T) for $[A] = \left[\frac{4}{\Delta t^2} M + \frac{2}{\Delta t} C + \tilde{K} \right]$.
2. Adjust the time step size, if applicable.

3. Initialize $\{U_{n+1}^0\}$, $\{F_{n+1}^0\}$ and $\{P_{n+1}^0\}$.
4. Decompose, $[A] = LDL^T$, if $\Delta t_{n+1} \neq \Delta t_n$. Then, purge the QN vector file.
5. Compute $\{R_{n+1}^i\}$.
6. Solve for $\{\Delta U^{i+1}\}$ by forward and backward substitution, using the BFGS update if applicable.
7. Compute the line search error.
8. If divergence occurs, go to step 12.
9. Proceed to the next step if the line search error is less than the tolerance. Otherwise, perform the line search process and go back to step 7.
10. Compute global error functions and check convergence.
11. If the solution has converged, go to step 16. Otherwise, save the QN vector on a temporary basis (if applicable) and go to the next iteration in step 5.
12. If it is the first divergence or the divergence after the KSTEP-th converged bisection step, update the stiffness matrix, and go back to step 4. Otherwise, proceed to the next step.
13. Bisect Δt and go back to step 3. However, if the maximum number of bisections allowed for the time step is reached, proceed to the next step.
14. Go to step 5 to continue iteration. However, if the maximum number of iterations allowed for an increment is reached, proceed to the next step.
15. Perform the reiteration procedure to find the best attainable solution.
16. Accept the solution (U_{n+1} and F_{n+1}) and append temporary QN vectors to the permanent QN vector file.
17. Compute the velocity \dot{U}_{n+1} .
18. Advance to the next time step with $n = n + 1$.

The adaptive time stepping method is implemented as an optional feature in SOL 99 with a new module NLTRD2 for the nonlinear transient response analysis [8.7]. Among numerous enhancements from the DMAP standpoint, the efficiency of the adaptive time stepping method stems from skipping about 50 statements by virtue of the internal decomposition within NLTRD2 module. Another aspect to consider in the DMAP is that the output time steps cannot be predetermined in the variable time stepping. Some efforts were required to bring together all the data blocks in compatible size and format.

The automatic time stepping and the bisection algorithms are flow-charted in Figs. 8.5.1 and 8.5.2, respectively. These algorithms are combined into the module NLTRD2 as shown in

Fig. 8.5.3. Control parameters for these algorithms to be specified by the user are gathered in the TSTEPNL Bulk Data entry.

Efforts have been made to minimize the thrashing in the algorithm and yet to provide a reliable solution efficiently to nonlinear problems. Thrashing of the decomposition is effectively prevented by adopting a stepping function in the adjustment criterion. However, some combination of user parameters could cause thrashing such as excessive bisection, or repeated cycle of time-step adjustment and bisection. Default values are tuned to near optimal condition to the best of our experience. If necessary, the user should make a judicious selection of the parameter values to prevent any thrashing.

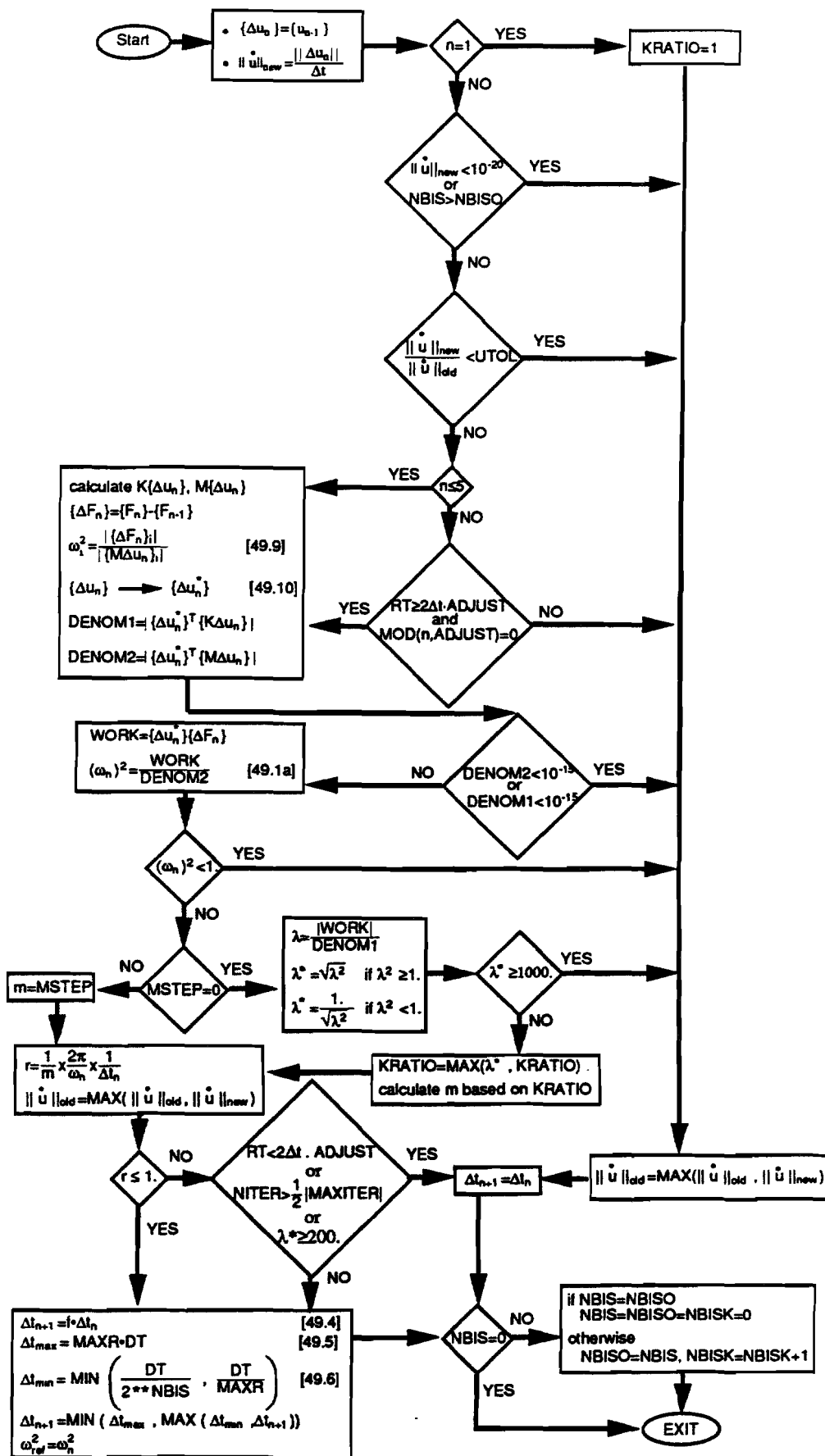


Figure 8.5.1 Flow Diagram for Automatic Time Step Adjustment

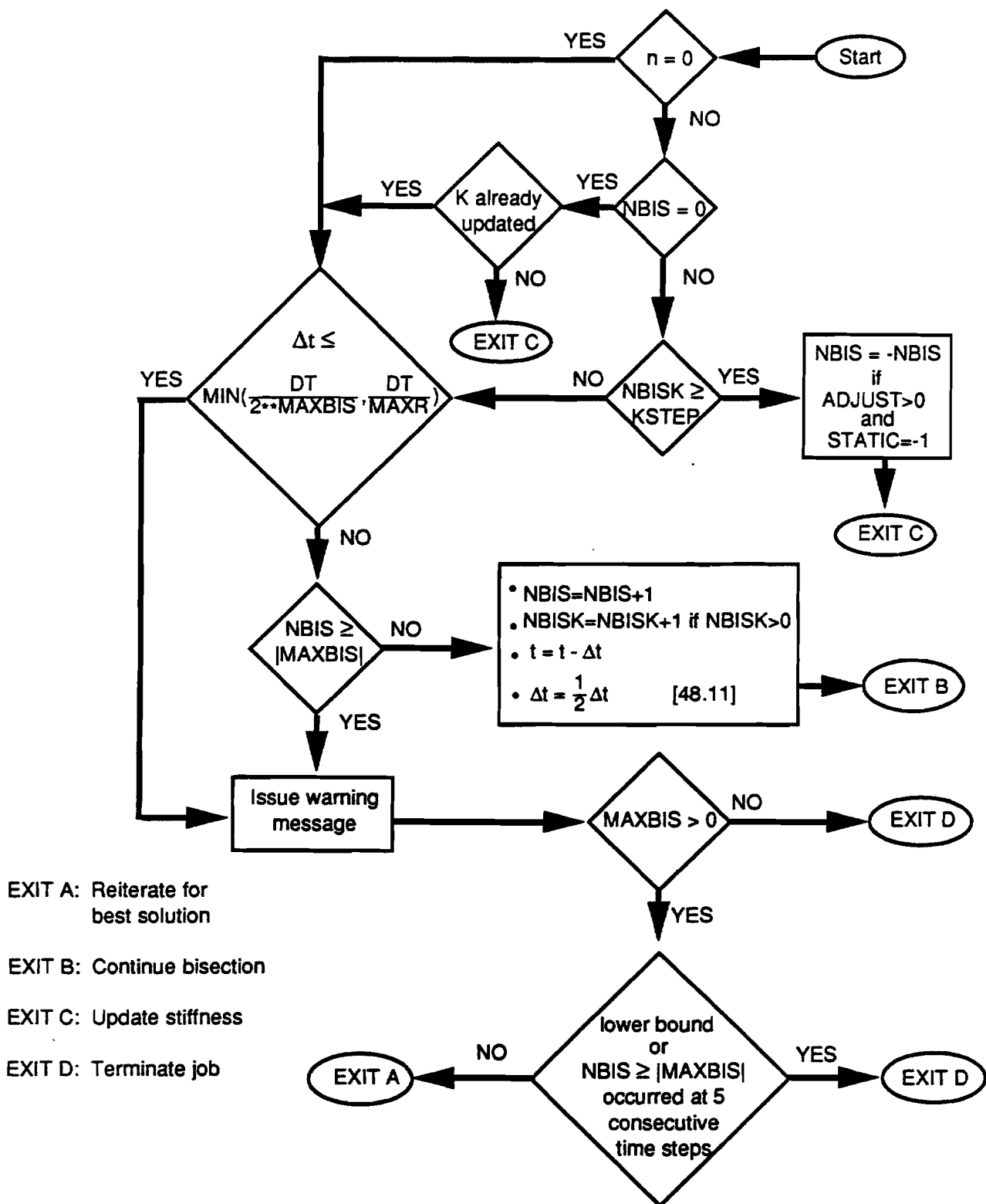


Figure 8.5.2 Flow Diagram for the Bisection Algorithm

8.6 VERIFICATION PROBLEMS

Five example problems are presented in this section to validate the effectiveness of the algorithm and to verify the program's performance.

8.6.1 Elastic-Plastic Rod with an End Mass Subjected to a Step Loading

Consider a single-degree system (mass-rod model), in which the rod is made of an elastic-perfectly plastic material. Fig. 8.6.1(a) shows the model subjected to a step loading with four different magnitudes: namely, $0.5F_y$, $0.66F_y$, $0.88F_y$ and F_y , respectively, in which F_y is the axial force corresponding to the initial yield of the rod. As a result, the mass responds with different patterns. All four cases are included in a single run by four disjoint models to test the effectiveness of the automatic time step adjustment. The input data for this problem is given in Table 8.6.1.

The theoretical solution can be considered in three different regimes: the elastic regime, plastic, and rebounding (unloading) regime. In the elastic regime the solution can be obtained simply by $u = \frac{P}{k}(1 - \cos \omega t)$ which is valid throughout the time history if $P \leq \frac{1}{2}F_y$. For $\frac{1}{2}F_y < P \leq F_y$, there is a point in time where the stress in the rod reaches the yield point. During the plastic deformation, the solution may be obtained from $m\ddot{u} + F_y = P$ using proper initial conditions. When the displacement reaches the maximum value, the system is governed by an elastic equation for rebound. If $P > F_y$, no rebound occurs and the mass will break away from the rod.

Two analyses have been made: one with an initial time step of 0.001 second and the other with 0.01 second. The displacement responses of the two analyses are almost identical and agree with the closed-form solution [8.8] as shown by A, B, C and D in Fig. 8.6.1(b). Also shown in Fig. 8.6.1(b) is the variation of the time step increment. In the first analysis, the program detected that the time step is too small and automatically quadrupled it to 0.004 second. In the second analysis when the initial time step was 0.01 second, the program reduced the time step to 0.005 second. No unnecessary time step adjustments were performed during the elastic-plastic vibration. Bisection did not occur in this analysis.

8.6.2 Impulsively Loaded Clamped Plate

An elastic-plastic plate clamped at two ends is subjected to an impulsive load over the center portion as shown in Fig. 8.6.2(a). A quarter section of the plate is modeled by five QUAD4 elements with symmetric boundary conditions using this software. The impulsive load was simulated by initial velocities applied at the nodal points as shown in Fig. 8.6.2(a). The material is elastic-perfectly plastic and the large displacement effect is also included in the response analysis for the duration of one *msec*. The input data for this problem is given in

Table 8.6.2.

Four different initial time step sizes (namely, 0.1 msec, 0.05 msec, 0.02 msec and 0.0025 msec) were used in the analysis. Predicted time histories of the mid-span displacement of the four cases are nearly identical as shown in Fig. 8.6.2(b). Also shown in Fig. 8.6.2(b) are the experimental data obtained by Balmer and Witmer [8.9] and the solution from the explicit integration obtained by Belytschko et al. [8.10]. This figure demonstrates that the software solution agrees more closely with the test results. Fig. 8.6.2(b) also shows the variation of time increments due to the automatic time step adjustment. In all of the cases the time step is adjusted to approximately 0.02 msec, from either the high or low value of a specified initial increment size. Bisection was not activated in this analysis.

A question arose if a coarse mesh resulted in a more accurate solution by beneficially filtering the high frequency response. In order to study this point, the analysis was repeated with a 10-QUAD4 model for comparison with the 10-element model of Belytschko et al. [8.10]. The result is even more favorable as shown in Fig. 8.6.2(b). It is noted, however, that the time step variation is quite different from the 5-element model. Except for the case of 0.0025 msec, bisection processes were activated at the beginning to reduce Δt below 0.0125 msec. This step size was maintained until approximately 0.4 msec, when Δt was doubled and remained unchanged to completion in all four cases.

8.6.3 Simply Supported Beam with a Restrained Motion

A simply supported elastic beam is subjected to a single cycle of a sinusoidal forcing function at the quarter span as illustrated in Fig. 8.6.3(a). The beam is modeled using 20 linear BEAM elements. A stopper (snubber) is present underneath the center of the beam with a clearance of 0.02 inch. This gap is simulated by a displacement dependent force which is active only when the beam is in contact with the snubber. The problem is linear except for the effect of the stopper. The input data for this problem is given in Table 8.6.3.

To demonstrate the effectiveness of the automatic time step adjustment, the same model was analyzed with two different initial time step sizes: 0.0002 sec and 0.002 sec. Each case was analyzed twice: once with and once without using the automatic time step adjustment. The displacement responses at the loading point for the four cases are compared in Fig. 8.6.3(b), in which the time steps are indicated by the symbol x in the figure. It is demonstrated that the time step size is adjusted to an equitable value (fluctuating between 0.0004 sec and 0.001 sec) regardless of the initial value. When the automatic time step adjustment is not activated, the solution process either wastes computing time by using time increments which are too small or renders inaccurate solutions by using time step sizes which are too coarse. All the solutions converged without activating the bisection process.

The model is modified for the performance comparison to include 200 beam elements. The duration of the analysis time span is also extended to 0.1 sec. Five different time step sizes (0.0001 sec, 0.0002 sec, 0.0005 sec, 0.001 sec, and 0.002 sec) have been used to investigate the

efficiency and effectiveness of the adaptive method. All computer runs were made on the VAX 8700 system at its off-peak time. The results are tabulated below:

CPU Time Comparison

User-specified $\Delta t(\text{sec})$	0.002	0.001	0.0005	0.0002	0.0001
Analysis with fixed time step	56	86	140	327	661
Analysis with adaptive time step algorithm	135	145	144	150	158

Note : Accuracy of the CPU time is estimated to have a 5% deviation.

It is noted that the CPU time is increasing almost proportionally to the decrease in the time step size if the time step size is fixed. However, in the case of the adaptive time stepping method the CPU time remains approximately constant regardless of the initial time step size. The adaptive time stepping method provides the definite advantage which is the avoidance of the trial-and-error process. Furthermore, it is likely to be more efficient in terms of computing time than the fixed time stepping, because the proper time step size is not only difficult to predict in advance but it also changes throughout the analysis history.

8.6.4 Bouncing Weight on an Elastic Platform

A 4-lb weight is resting on a spring-supported platform which is initially compressed 4 inches in the position as shown in Fig. 8.6.4(a). The platform is then released, and after moving 2 inches, is halted by a stopper. The finite element model consists of one ROD element representing the spring, a mass for the weight block, and two GAP elements simulating contacts between the mass-platform and the platform-stopper interfaces as shown in Fig. 8.6.4(a). The gravity field is applied and a small mass is assigned to the platform to suppress its jittering. The stopper is represented by a fixed nodal point and the initial deformation in the spring by an equivalent step force. The input data for this problem is given in Table 8.6.4.

A closed-form solution is obtained for the massless platform. While the weight is riding on the platform, the motion is governed by $m\ddot{x} + kx = -mg$ with initial conditions $x_0 = -4$ and $\dot{x}_0 = 0$. The velocity of the weight block at the time of its departure from the platform can be obtained by the conservation of energy. This velocity becomes the initial condition for the equation $h = v_0t - \frac{1}{2}gt^2$, which governs the motion after the weight departs from the platform.

The response predicted by the software is compared with the theoretical solution as shown in Fig. 8.6.4(b). The time steps are indicated by the symbol x in the figure. The initial time step size was specified as 0.0025 sec, which was automatically adjusted to 0.01 sec at the initial adjustment phase as shown in Fig. 8.6.4(b). In this analysis, bisection was activated when the block was departing from the platform and changing the status of the GAP,

resulting in 0.005 sec. This analysis proves the effectiveness of the current adaptive method for the problems involving the gaps and/or drastic changes of the structural stiffness.

8.6.5 Shallow Spherical Cap under a Concentrated Apex Load

A thin shell in the shape of a shallow dome is clamped at the periphery and subjected to a step loading of a concentrated force at the apex as shown in Fig. 8.6.5(a). The geometric nonlinear effects are included, but the material properties are assumed to remain elastic throughout the analysis.

Considering the symmetry, only one quarter of the cap is modeled using 27 QUAD4 elements with symmetric boundary conditions as illustrated in Fig. 8.6.5(a). The input data for this problem is given in Table 8.6.5. This problem is chosen to demonstrate the performance of the automatic time step adjustment and the bisection method combined. Hence, the analysis was deliberately started with a relatively larger time step size, 20 μ sec. This time step size is 10 times larger than the size adopted in the previous study by Mondkar and Powell using an axisymmetric model with ten 8-noded quadrilateral elements [8.11].

As expected, the solution diverged at the very first time step and the algorithm activated the bisection process, resulting in $\Delta t = 10 \mu$ sec. After obtaining converged solutions at two consecutive time steps, the program detected that the time step size was still too large and adjusted it to a smaller size, 2.5 μ sec. The analysis continued without any adjustment until the displacement approached the peak, where Δt was adjusted to 2 μ sec (which was the lower bound allowed for adjustment). The step size was again adjusted to 4 μ sec past the peak response, which was later adjusted back to 2 μ sec before completion. No further bisection was required to complete the analysis. The displacement response at the apex is compared with the results by Mondkar and Powell [8.11] in Fig. 8.6.5(b). Fig. 8.6.5(b) also shows the variation of time step size due to the adaptive time step adjustment.

When the same model was analyzed again with a smaller Δt (i.e., 2 μ sec), the adjusted time step size was reduced to as low as 0.5 μ sec during the analysis. In this analysis, however, the bisection did not occur. The resulting response agrees with that of the previous case shown in Fig. 8.6.5(b) except for the minor details of spikes.

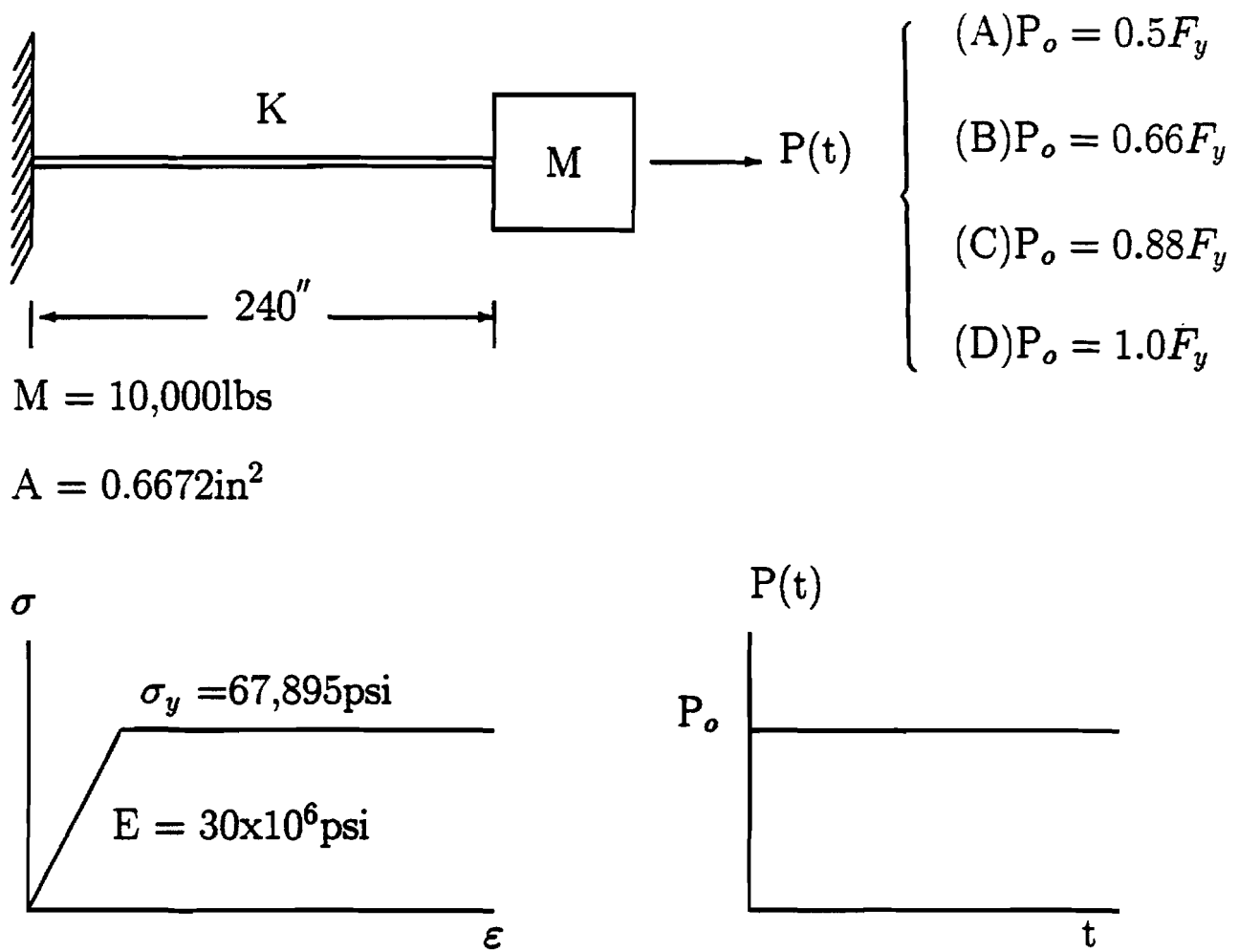


Figure 8.6.1(a). Elastic-Plastic Rod Subjected to a Step Loading

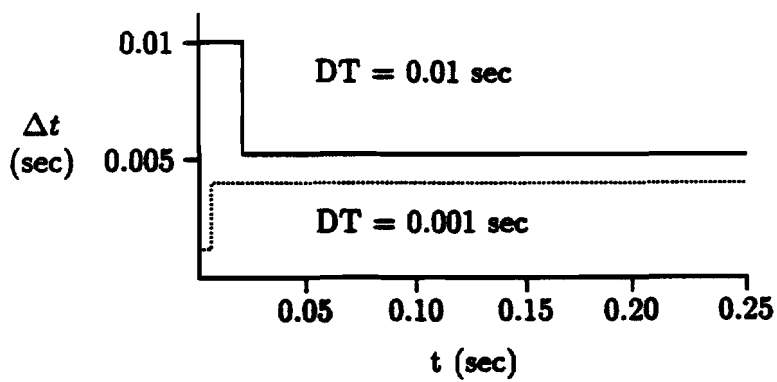
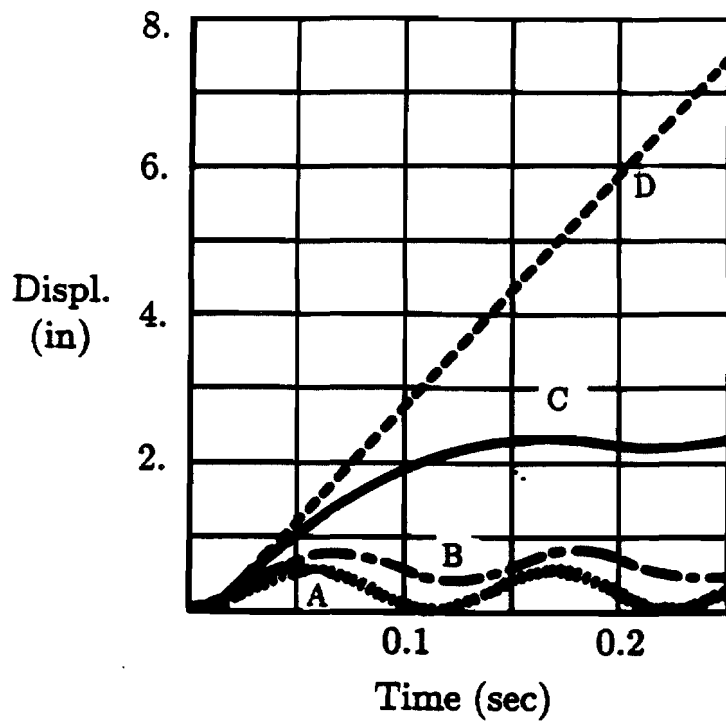
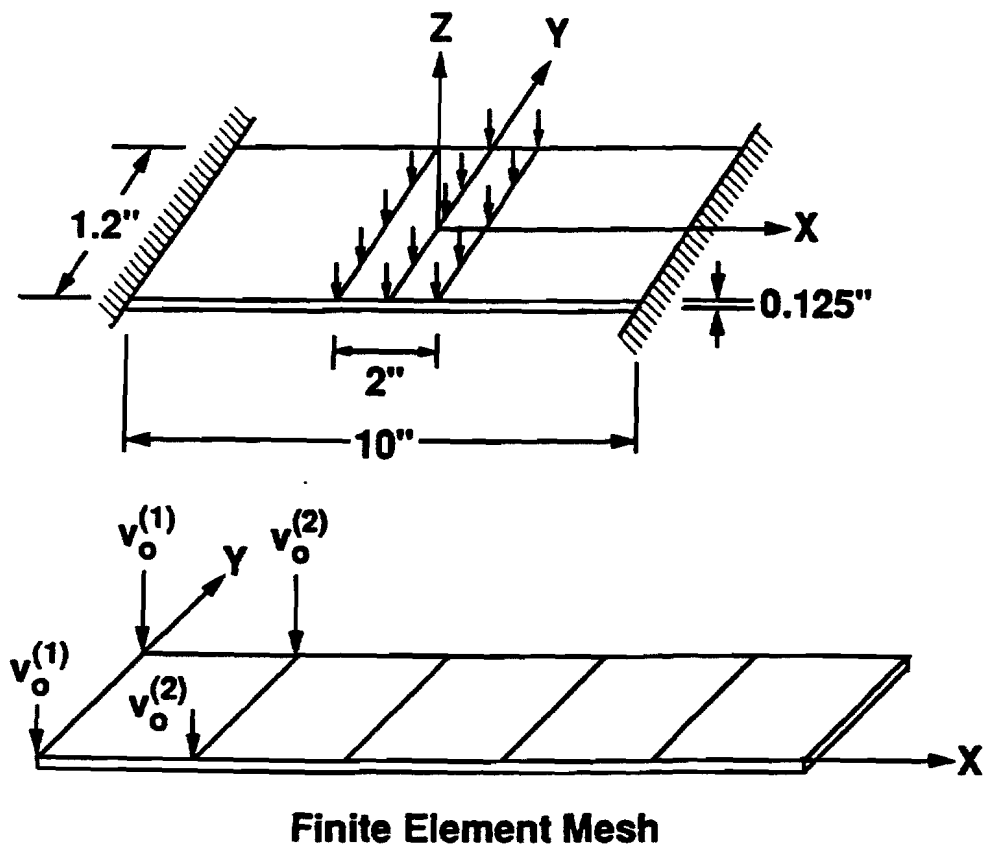


Figure 8.6.1(b). Elastic-Plastic Rod: Displacement Response and Time Step Variation



$$\begin{aligned}
 \rho &= 2.61 \times 10^{-4} \text{ lb-sec}^2/\text{in}^4 \\
 E &= 10.4 \times 10^6 \text{ psi} \\
 \sigma_y &= 41400 \text{ psi, yield stress} \\
 v_o^{(1)} &= 5000 \text{ in/sec.} \\
 v_o^{(2)} &= 3904 \text{ in/sec.}
 \end{aligned}$$

Figure 8.6.2(a). Impulsively Loaded Clamped Plate

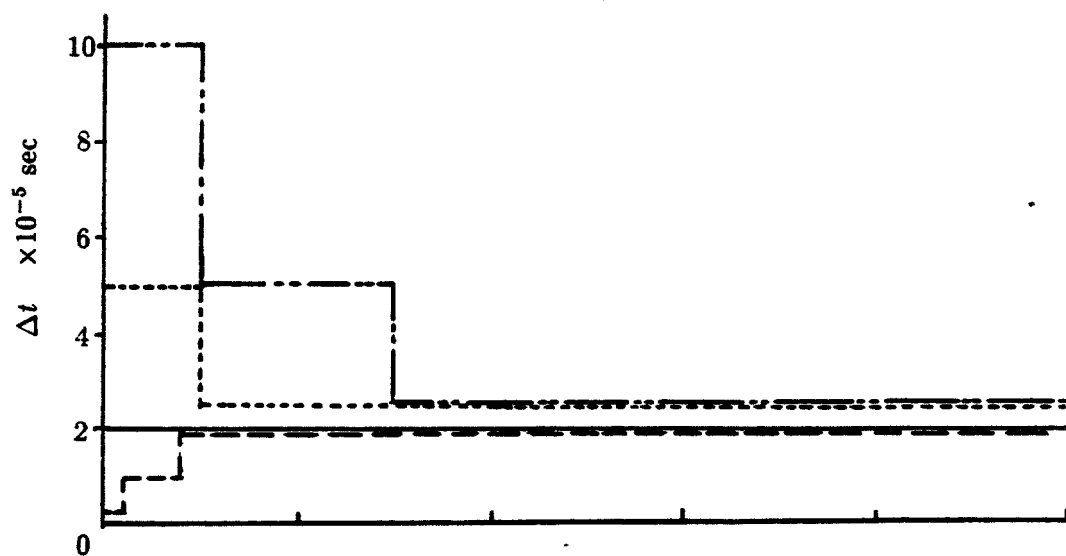
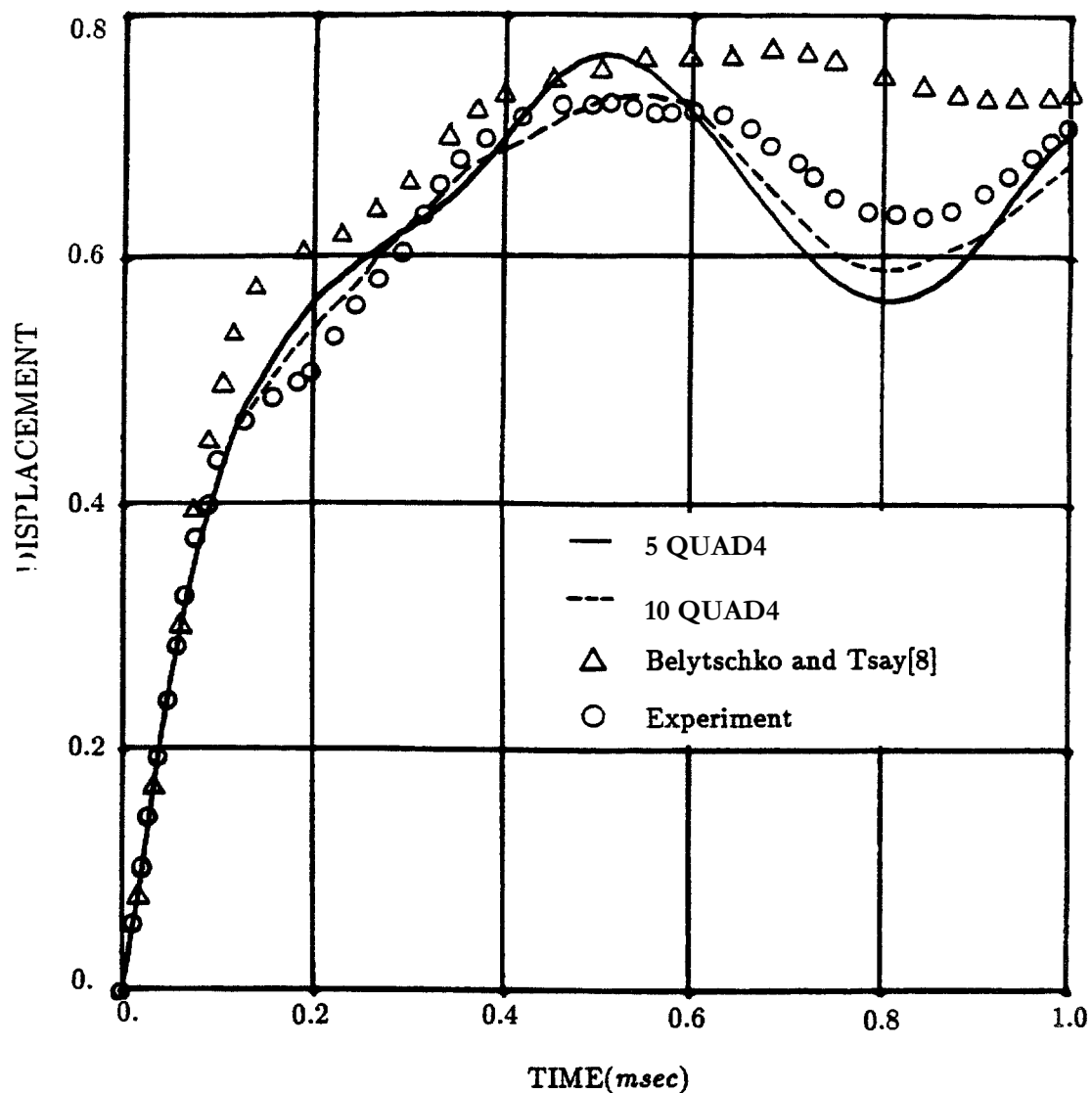


Figure 8.6.2(b). Clamped Plate: Central Displacement Response and Time Step Variation

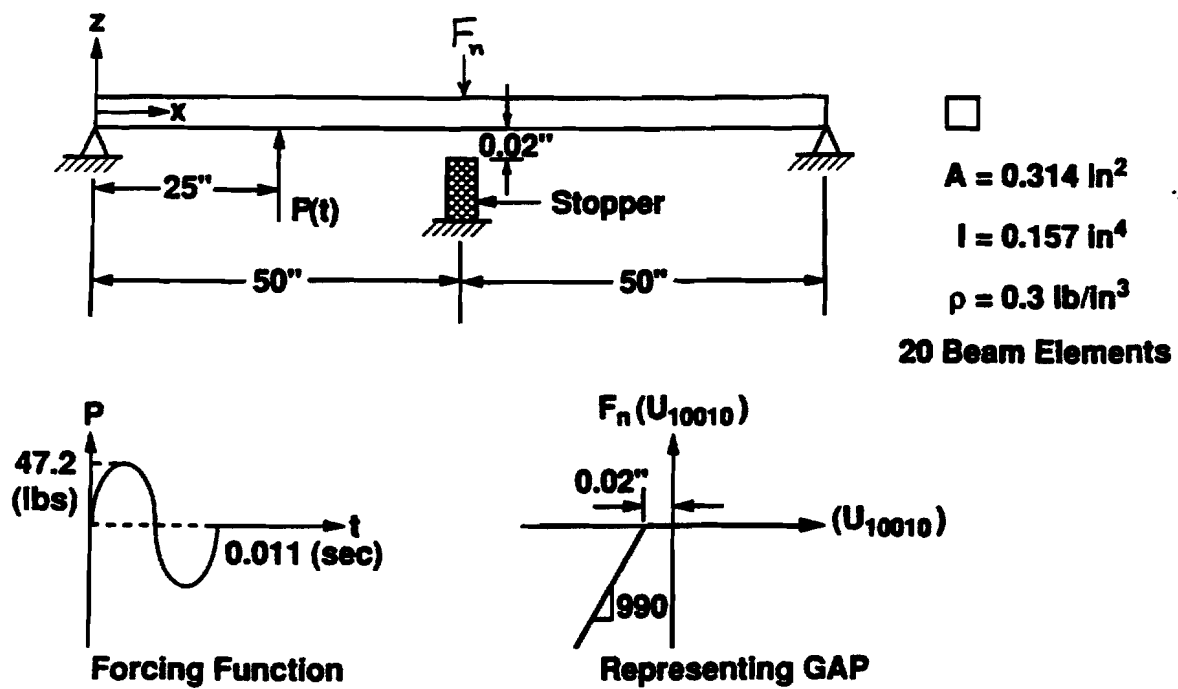
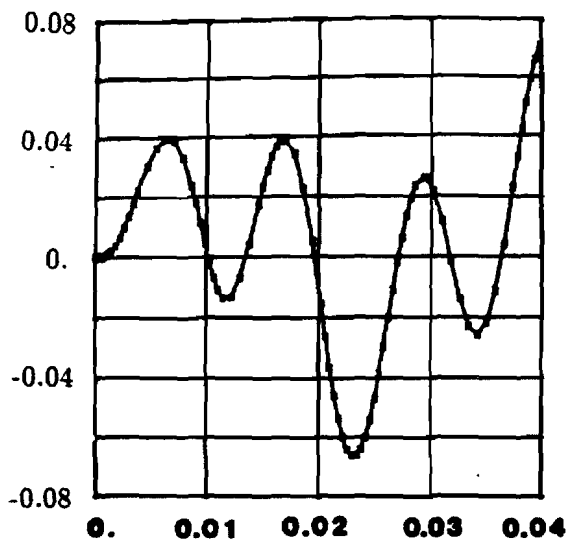
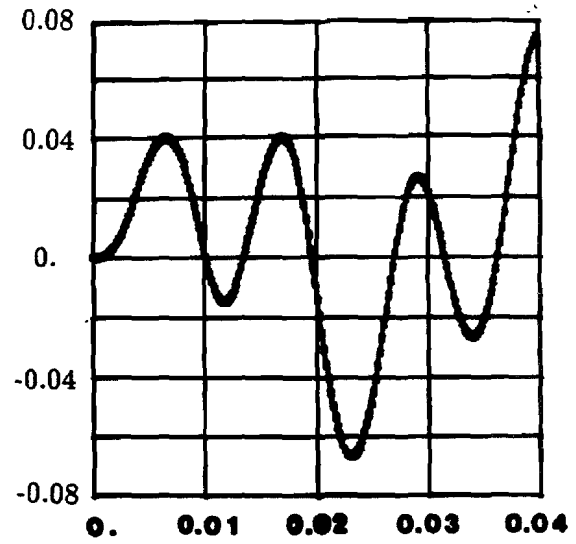


Figure 8.6.3(a). Simply Supported Beam with a Restrained Motion



Time(sec.)

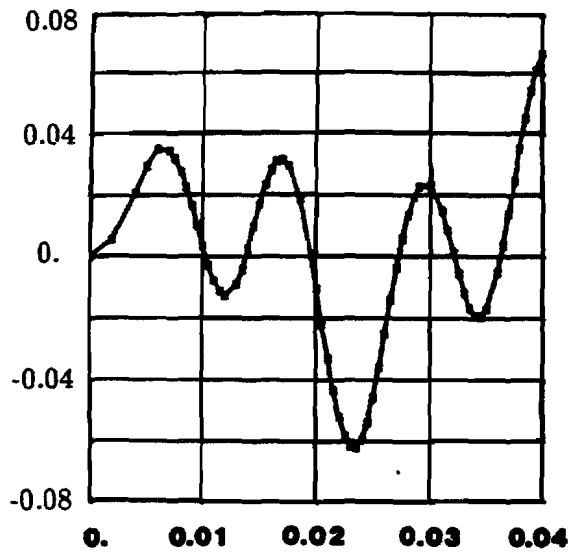
with adjustment



Time(sec.)

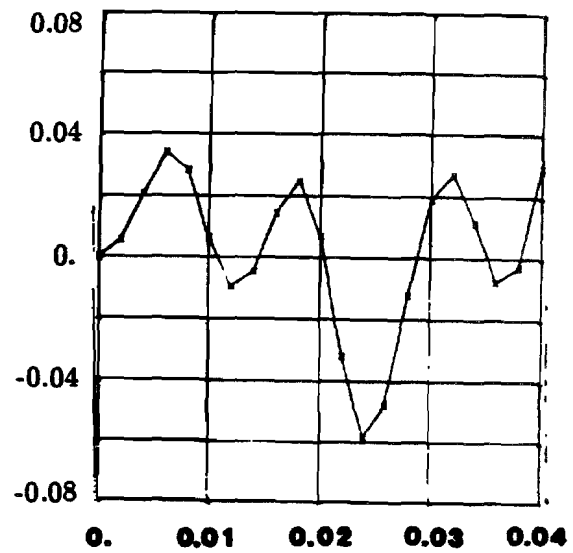
without adjustment

User-specified $\Delta t = 0.0002$ Sec.



Time(sec.)

with adjustment

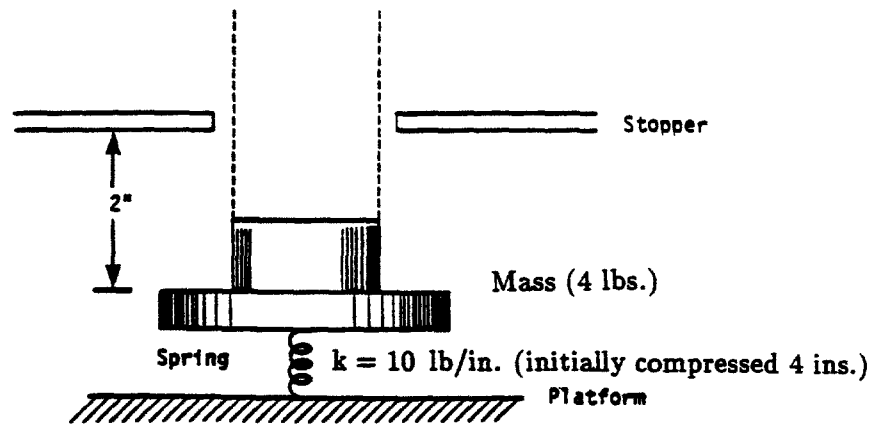


Time(sec.)

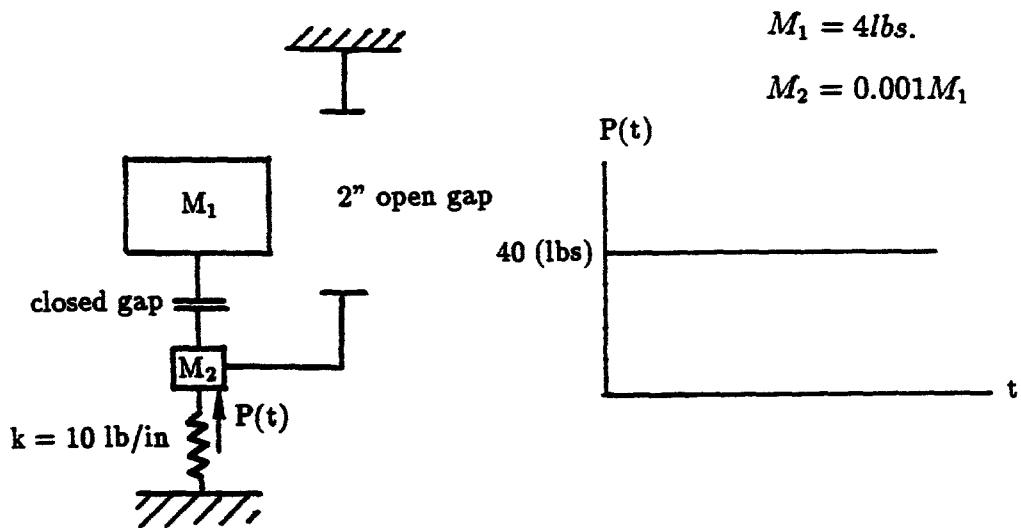
without adjustment

User-specified $\Delta t = 0.002$ Sec.

Figure 8.6.3(b). Simply Supported Beam: Loading Point Displacement Time History



Structural Configuration



Software Model

Figure 8.6.4(a). Bouncing Weight on an Elastic Platform

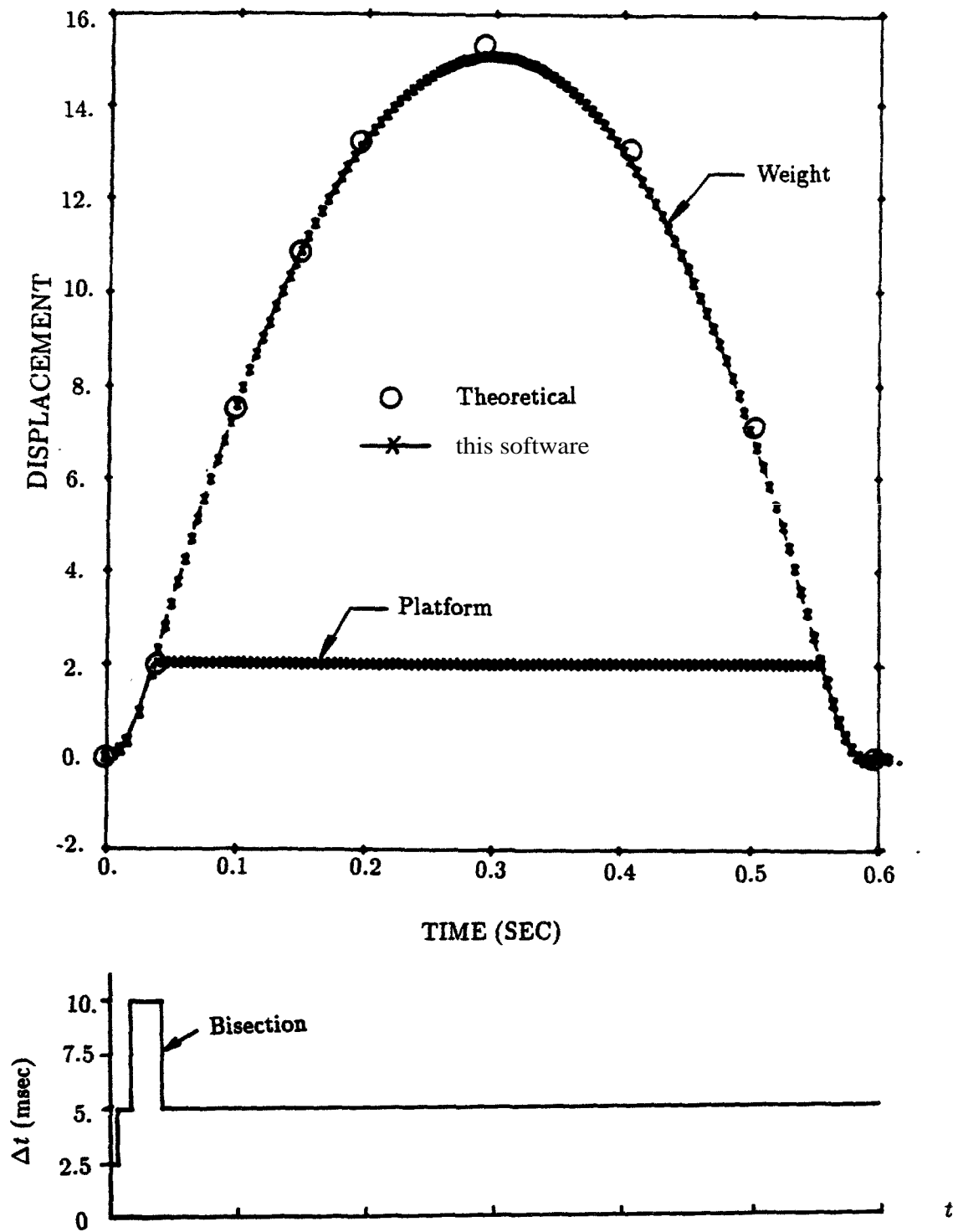
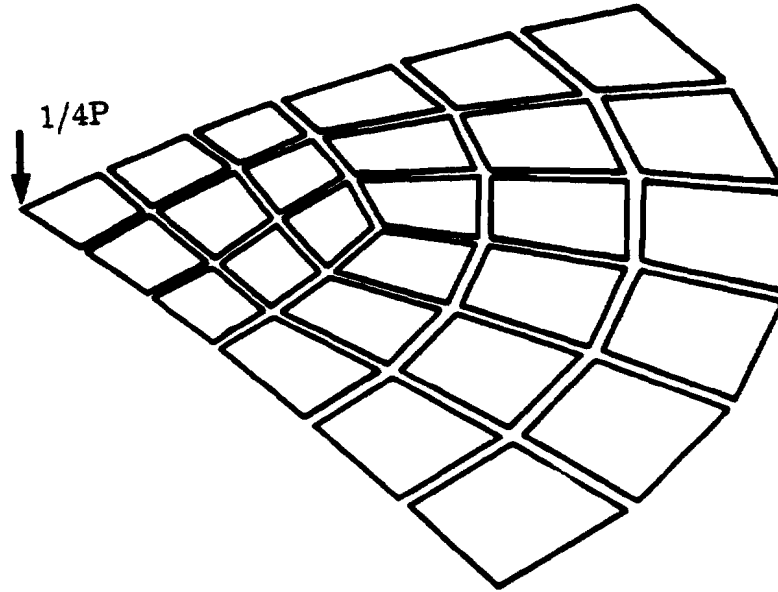
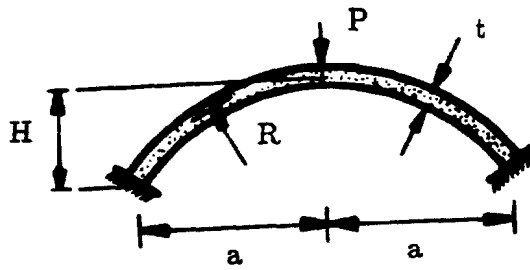


Figure 8.6.4(b). Bouncing Weight: Displacement Time History



$R = 4.75$ in.
 $a = 0.90$ in.
 $H = 0.08589$ in
 $t = 0.01576$ in.
 $E = 10000$ ksi, Young's modulus
 $\nu = 0.3$, Poisson's ratio
 $\rho = 2.45 \times 10^{-4}$ lb-sec²/in⁴
 $P = 100$ lbs., constant applied force

Figure 8.6.5(a). Shallow Spherical Cap under a Concentrated Apex Load

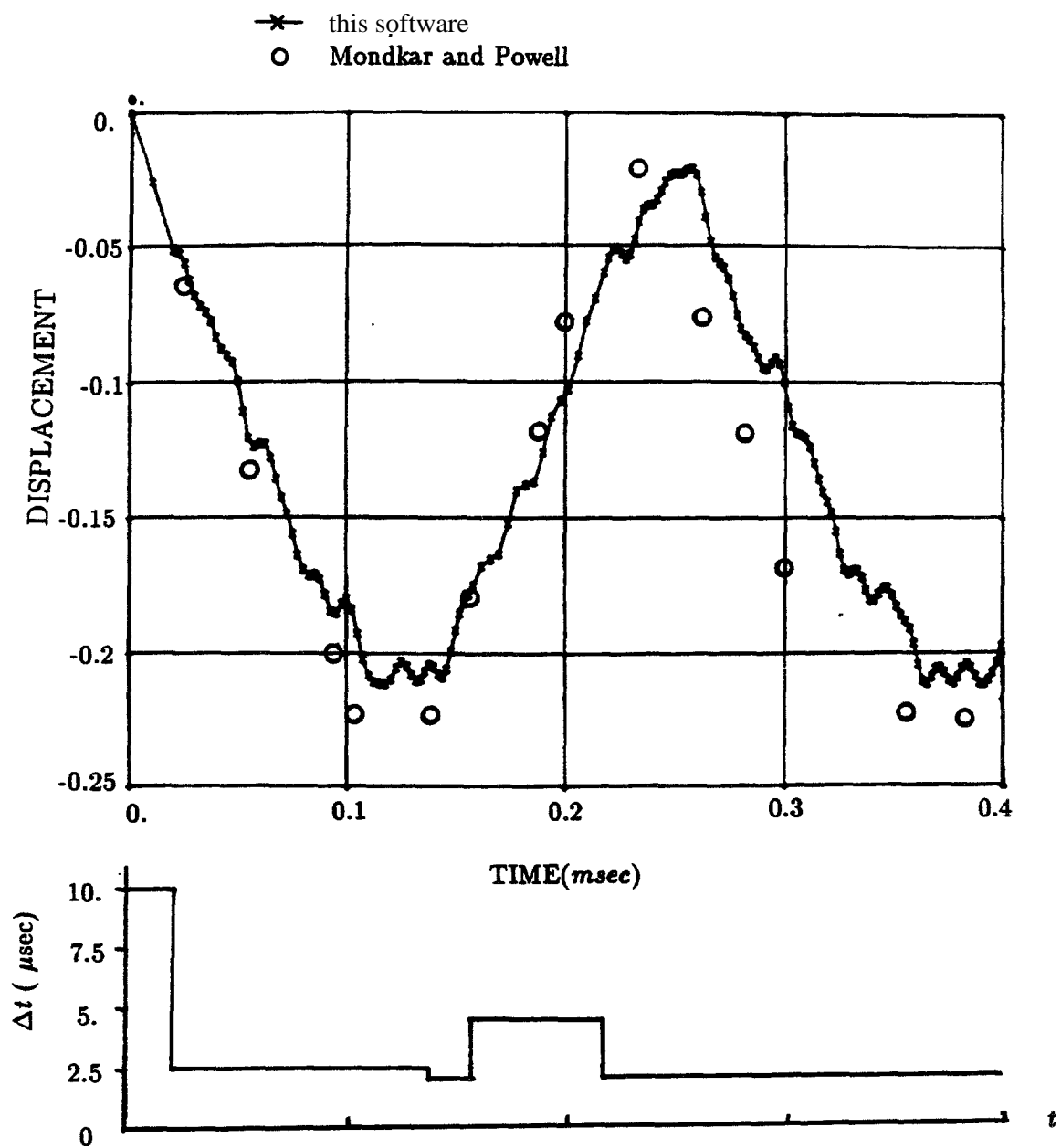


Figure 8.6.5(b). Spherical Cap: Apex Displacement Response

Table 8.6.1 Input Data Listing for Elastic-Plastic Rod

```

ID TBIGGS4,V66 $ SSH 10/22/87
DIAG 8,50      $ PRINT MATRIX TRAILERS AND ITERATIONS
TIME 5         $ VAX TETRA
SOL 99         $
CEND
TITLE=ELASTO-PLASTIC VIBRATION PROBLEM (LOAD = .5/.66/.88/1. FY)
SUBTITLE=NONLINEAR TRANSIENT ANALYSIS FOR 0.25 SEC.
LABEL=J. M. BIGGS, SECTION 2.7, P. 69-71
ECHO=BOTH
  SET 1 = 11,21,31,41
    DISP=1
    ACCE=1
    VELO=1
    STRESS=1
SEALL=ALL
SUBCASE 1
  DLOAD=100
  TSTEPNL=100
OUTPUT(XYPLOT)
  CSCALE = 1.5
  XTITLE = TIME IN SECS
  XGRID LINES = YES
  YGRID LINES = YES
  YTITLE = DISPLACEMENT GRID 1
  XYPLOT DISP RESP/11(T2),21(T2),31(T2),41(T2)
  YTITLE = VELOCITY GRID 1
  XYPLOT VELO RESP/11(T2),21(T2),31(T2),41(T2)
  YTITLE = ACCELERATION GRID 1
  XYPLOT ACCE RESP/11(T2),21(T2),31(T2),41(T2)
  YTITLE = STRESS IN RODS
  XYPLOT STRESS RESP/11(2),21(2),31(2),41(2)
  YTITLE = STRESS IN ROD
  XYPLOT STRESS RESP /11(2),21(2),31(2),41(2)
BEGIN BULK
$ GEOMETRY AND CONNECTIVITY
GRID 11 13456
GRID 12 240. 123456
CROD 11 2 12 11
CMASS2 11 10000. 11 2
GRID 21 13456
GRID 22 240. 123456
CROD 21 2 22 21
CMASS2 21 10000. 21 2
GRID 31 13456
GRID 32 240. 123456
CROD 31 2 32 31

```

CMASS2	31	10000.	31	2					
GRID	41						13456		
GRID	42			240.			123456		
CROD	41	2	42	41					
CMASS2	41	10000.	41	2					
\$ PROPERTIES									
PROD	2	2	.6672						
MAT1	2	30.+06							
MATS1	2		PLASTIC 0.0	1	1	67895.68			
\$ LOADING									
DLOAD	100	1.	1.	101	1.	102	1.	103	+DL1
+DL1	1.	104							
DAREA	201	11	2	22650.					
DAREA	202	21	2	30000.					
DAREA	203	31	2	40000.					
DAREA	204	41	2	45300.					
TLOAD1	101	201		0	50				
TLOAD1	102	202		0	50				
TLOAD1	103	203		0	50				
TLOAD1	104	204		0	50				
TABLED1	50								+ABC
+ABC	0.	1.	10.	1.	ENDT				
\$ PARAMETERS									
PARAM	WTMASS	.0025907							
TSTEPNL	100	50	.0050	1	ADAPT				+TS21
+TS21									+TS22
+TS22									
ENDDATA									

Table 8.6.2 Input Data Listing for Impulsively Loaded Plate

```

ID GMNL08, V66 $ SHL 5/21/87 $IMPULSIVE LOAD
TIME 5          $ VAX8700(TETRA) CPU MIN.
SOL 99          $ NL-TRANSIENT SCR=YES
DIAG 8,50       $ PRINT MATRIX TRAILERS AND ITERATION
CEND
TITLE = NONLINEAR IMPULSIVELY LOADED CLAMPED PLATE (K6=1.+4) 99IQ4
  SEALL = ALL
  SPC   = 200
  IC    = 300
  TSTEP = 400
  STRE(PLOT) = ALL
$
  SET 1 = 1
  DISP = 1
  VELO = 1
  ACCE = 1
OUTPUT (XYOUT)
  CSCALE 1.3
  XGRID LINES = YES
  YGRID LINES = YES
  XTIT =
  XMAX = 1.0-3
$
  YTITLE = D I S P L 1
  XYPLOT DISP / 1(T3)
  YTITLE = V E L O C 1
  XYPLOT VELO / 1(T3)
  YTITLE = A C C E L 1
  XYPLOT ACCE / 1(T3)
  YTITLE = D I S P L 3 1
  XYPLOT DISP / 31(T3)
  YTITLE = V E L O C 3 1
  XYPLOT VELO / 31(T3)
  YTITLE = A C C E L 3 1
  XYPLOT ACCE / 31(T3)
  YTITLE = S T R E S S 1 -
  XYPLOT ELST / 1(3)
BEGIN BULK
TSTEPNL 400      200      5.0-6      1      ADAPT      +TS1.
GRID 1           0.      0.0
GRID 2           0.      +.6
GRID 11          1.      0.0
GRID 12          1.      +.6
GRID 21          2.      0.0
GRID 22          2.      +.6
GRID 31          3.      0.0

```

GRID	32		3.	+ .6				
GRID	41		4.	0.0				
GRID	42		4.	+ .6				
GRID	51		5.	0.0				
GRID	52		5.	+ .6				
CQUAD4	1	100	1	11	12	2		
CQUAD4	2	100	11	21	22	12		
CQUAD4	3	100	21	31	32	22		
CQUAD4	4	100	31	41	42	32		
CQUAD4	5	100	41	51	52	42		
PSHELL	100	100	1.25-1	100		100		+PS
+PS	-.05	+.05						
MAT1	100	10.4+6		.3	2.61-4			.02
MATS1	100		PLASTIC	0.0	1	1	41400.	
PARAM	COUPMASS1							
PARAM	K6ROT	1.0+4						
PARAM	NLAYERS	5						
PARAM	LGDISP	1						
PARAM	W4	1.2E4						
SPC1	200	246	1	11	21	31	41	51
SPC1	200	123456	51	52				
SPC1	200	156	1	2				
TIC	300	1	3		5000.			
TIC	300	2	3		5000.			
TIC	300	11	3		3903.88			
TIC	300	12	3		3903.88			
ENDDATA								

Table 8.6.3 Input Data Listing for Beam with a Restrained Motion

```

ID TNOLIN, V66 $ SHL 5/19/87 $ MSC, D9918 $ KDB 12JUN84
TIME 5          $ VAX8700(TETRA) CPU MIN.
SOL 99          $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50       $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE= DYNAMICS HANDBOOK DEMONSTRATION PROBLEM           D9918
SUBTITLE= DIRECT TRANSIENT RESPONSE, NONLINEAR FORCE
  LABEL= NOLIN IN SOL 99
SEALL=ALL
  ECHO=UNSORT
  SPC=1002
SET 1=10005
SET 2=10010
SET 3=10005,10010
  DISPL=3
$ ACCEL=2
  VELO=3
  OLOAD=1
  NLLOAD=2
SUBCASE 1 $
  DLOAD=30
  TSTEPNL=20
  NONLINEAR=13 $ SELECT NONLINEAR FORCE
OUTPUT(XYPLT)
  CSCALE=1.8
  PLOTTER NAST
  CURVELINESYMBOL = 1
  XGRID=YES
  YGRID=YES
  XTITLE=TIME
  YTITLE=DISP 10005 T3
XYPLT DISP /10005(T3)
  YTITLE=DISP 10010 T3
XYPLT DISP /10010(T3)
  YTITLE=APPLIED LOAD 10005 T3
XYPLT OLOAD RESPONSE/10005(T3) $ PLOT THE INPUT
  YTITLE=NONLINEAR LOAD 10010 T3
XYPLT NONLINEAR RESPONSE/10010(T3)
BEGIN BULK
$. . . . .2. . . . .3. . . . .4. . . . .5. . . . .6. . . . .7. . . . .8. . . . .9. . . . .10. . . .
$ MODELING INFORMATION FOR BEAM ONLY
CBAR 101 100 10000 10001 0.0 0.0 1.
CBAR 102 100 10001 10002 0.0 0.0 1.
CBAR 103 100 10002 10003 0.0 0.0 1.
CBAR 104 100 10003 10004 0.0 0.0 1.
CBAR 105 100 10004 10005 0.0 0.0 1.

```

CBAR	106	100	10005	10006	0.0	0.0	1.
CBAR	107	100	10006	10007	0.0	0.0	1.
CBAR	108	100	10007	10008	0.0	0.0	1.
CBAR	109	100	10008	10009	0.0	0.0	1.
CBAR	110	100	10009	10010	0.0	0.0	1.
CBAR	111	100	10010	10011	0.0	0.0	1.
CBAR	112	100	10011	10012	0.0	0.0	1.
CBAR	113	100	10012	10013	0.0	0.0	1.
CBAR	114	100	10013	10014	0.0	0.0	1.
CBAR	115	100	10014	10015	0.0	0.0	1.
CBAR	116	100	10015	10016	0.0	0.0	1.
CBAR	117	100	10016	10017	0.0	0.0	1.
CBAR	118	100	10017	10018	0.0	0.0	1.
CBAR	119	100	10018	10019	0.0	0.0	1.
CBAR	120	100	10019	10020	0.0	0.0	1.
CONM2	12	10010		.1			
GRID	10		50.		-1.		
GRID	10000		0.0	0.0	0.0		1246
GRID	10001		5.	0.0	0.0		1246
GRID	10002		10.	0.0	0.0		1246
GRID	10003		15.	0.0	0.0		1246
GRID	10004		20.	0.0	0.0		1246
GRID	10005		25.	0.0	0.0		1246
GRID	10006		30.	0.0	0.0		1246
GRID	10007		35.	0.0	0.0		1246
GRID	10008		40.	0.0	0.0		1246
GRID	10009		45.	0.0	0.0		1246
GRID	10010		50.	0.0	0.0		1246
GRID	10011		55.	0.0	0.0		1246
GRID	10012		60.	0.0	0.0		1246
GRID	10013		65.	0.0	0.0		1246
GRID	10014		70.	0.0	0.0		1246
GRID	10015		75.	0.0	0.0		1246
GRID	10016		80.	0.0	0.0		1246
GRID	10017		85.	0.0	0.0		1246
GRID	10018		90.	0.0	0.0		1246
GRID	10019		95.	0.0	0.0		1246
GRID	10020		100.	0.0	0.0		1246
MAT1	1000	3.+7		.3	.3		
PARAM	GRDPNT	10010					
PARAM	WTHASS	.002588					
PBAR	100	1000	0.31416	0.15708			+000001
++0000011.		0.					+000002
SPC	1002	10	123456				
SPC	1002	10020	3		10000	3	
\$							
\$	MODELING INFORMATION FOR CENTER SPRING						
CROD	10	10	10	10010			
MAT1	10	10.		0.			

```

PROD      10      10      1.
MATS1,10,,PLASTIC,0.0,1,1,3.0E8
$ LOADING AND SOLUTION INFORMATION
TLOAD2  30      33      0.0      .11451-187.33      -90.
DAREA   33      10005   3      47.2
TSTEPNL 20      199      .0002   1      ADAPT      +TS1
+TS1      +TS2
+TS2
$
$ MODELING INFORMATION FOR NONLINEAR SPRING
$NOLIN1 SID      GI      CI      S      GJ      CJ      T
NOLIN1  13      10010   3      1.0      10010   3      13
$TABLED1 ID      +ABC
$+ABC   X1      Y1      X2      Y2      X3      Y3
TABLED1 13      +TAB
+TAB     -2.5E-2 4.95      -2.0E-2 0.      0.      0.      ENDT
ENDDATA

```


Table 8.6.4 Input Data Listing for Bouncing Weight Model

```

ID VGAPM,V66 $ SHL 12/11/82, SSH 1/13/88
TIME 5 $ FOR VAX
DIAG 8,50 $ PRINT MATRIX TRAILERS AND ITERATIONS
SOL 99
CEND
TITLE=GAP VERIFICATION
SUBTITLE=NONLINEAR TRANSIENT ANALYSIS
ECHO=UNSORT
SET 1=201,202
SET 2=200 $,203
SET 3=22 ,23
SET 4=20 $ CONROD
    DISP=1
$ OLOAD=1
$ SPCF=2
    ELFORCE=4
    STRESS=3 $ FOR GAP FORCES
SEALL=ALL
    LOADSET=20
SUBCASE 1
    DLOAD=210
    TSTEPNL=21
SUBCASE 2
    DLOAD=210
    TSTEPNL=21
OUTPUT(XYPLOT)
    CSCALE=1.3
    PLOTTER=NAST
    XTITLE=TIME IN SEC.
    YTITLE=DISPL. IN INCH
$ YMIN=0.
$ YMAX=15.
XYPLOT DISP RESP/201(T1),202(T1)
BEGIN BULK
TSTEPNL 21      200      0.0025  1      ADAPT      PW      +TSP1
+TSP1
+TSP2          0
$ PROPERTIES
PARAM  W4      31.
$PARAM W3      1000.
$PARAM G       0.01
MAT1    1      100.      0.3      0.01
PGAP    3      1.E+4      +PG3
+PG3    5.-3
PGAP    4      2.      1.E+4      +PG4
+PG4    5.-3

```

\$ LOADING							
FORCE	201	201		40.	1.	0.	0.
GRAV	200		386.	-1.	0.	0.	
LOAD	202	1.	1.	200	1.	201	
LSEQ	20	301	202				
\$TLOAD2	210	301		0	0.	100.	
TLOAD1	210	301		0	220		
TABLED1 220							
+TAB1	0.	0.	0.001	1.	100.	1.	ENDT
\$ BOUNDARY CONDITIONS							
GRDSET							23456
\$ GEOMETRY AND CONNECTIVITY							
GRID	200		-6.	0.	0.		123456
GRID	201		4.	0.	0.		
GRID	202		4.1	0.	0.		
GRID	203		6.	0.	0.		123456
CONROD	20	200	201	1	1.		
CONM2	24	201		1.-5			
CONM2	21	202		0.01036			
CGAP	22	3	201	202	0.	1.	0.
CGAP	23	4	201	203	0.	1.	0.
ENDDATA							

+TAB1

Table 8.6.5 Input Data Listing for Shallow Spherical Cap

```

ID SCAP,V66    $KI-DOO KIM $ MAY 1987,  GAD MSC, SAAB4
SOL 99        $ NONLINEAR TRANSIENT RESPONSE
DIAG 50       $ TRACE ITERATION PROCESS
TIME 60       $ CPU TIME IN MINUTES
CEND
TITLE   = TEST PROBLEM 1.4
SUBTITLE= SHALLOW SPHERICAL CAP WITH CONCENTRATED APEX LOAD
LABEL   = REF: MONDKAR AND POWELL, IJNME 11:499-520 (1977)
SEALL   = ALL
SET 130 = 1 $ THRU 30
  DISP  = 130
SET 14  = 1 $ THRU 4
  OLOAD = 14
  SPC   = 1234
SUBCASE 1
  DLOAD      = 100
  TSTEPNL   = 250
OUTPUT(PLOT)
  CSCALE 1.5
  PLOTTER NAST
  PAPER SIZE 29. X 21.
  SET 1 = ALL
  SET 2 = 1 THRU 6, 10, 11, 17, 18, 24, 25, 31, 32
  AXES X, Y, Z
  VIEW 34.27, 23.17, 0.
  FIND SCALE ORIGIN 1 SET 1
  PLOT SET 1 LABEL BOTH SHRINK
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 SET 1
  AXES Y, MX, Z
  VIEW 0., 0., 0.
  FIND SCALE ORIGIN 1 SET 2
  PLOT STATIC 0 SET 2
OUTPUT(XYLOT)
  CSCALE 1.5
  PLOTTER NAST
  XPAPER = 29.
  YPAPER = 21.
  CURVELINESYMBOL = 2
  XGRID = YES
  YGRID = YES
  XTITLE = TIME
  YTITLE = VERTICAL DISPLACEMENT AT GRID 1
  XYLOT DISP /1(T3)
BEGIN BULK
TSTEPNL 250      80      5.-6      1      ADAPT      10      PW      +TS1

```

+TS1									+TS2
+TS2									
PARAM	LGDISP	+1							
CORD2S	10		0.	0.	0.	0.	0.	1.	+1
+1	1.	0.	1.						
GRID	1	10	4.76	0.0	0.0				
GRID	2	10	4.76	1.5	0.0				
GRID	3	10	4.76	1.9	45.0				
GRID	4	10	4.76	1.5	90.0				
GRID	5	10	4.76	3.0	0.0				
GRID	6	10	4.76	3.2	22.5				
GRID	7	10	4.76	3.7	45.0				
GRID	8	10	4.76	3.2	67.5				
GRID	9	10	4.76	3.0	90.0				
GRID	10	10	4.76	4.5	0.0				
GRID	11	10	4.76	4.5	15.0				
GRID	12	10	4.76	4.7	30.0				
GRID	13	10	4.76	5.1	45.0				
GRID	14	10	4.76	4.7	60.0				
GRID	15	10	4.76	4.5	75.0				
GRID	16	10	4.76	4.5	90.0				
GRID	17	10	4.76	6.6	0.0				
GRID	18	10	4.76	6.6	15.0				
GRID	19	10	4.76	6.6	30.0				
GRID	20	10	4.76	6.6	45.0				
GRID	21	10	4.76	6.6	60.0				
GRID	22	10	4.76	6.6	75.0				
GRID	23	10	4.76	6.6	90.0				
GRID	24	10	4.76	8.7	0.0				
GRID	25	10	4.76	8.7	15.0				
GRID	26	10	4.76	8.7	30.0				
GRID	27	10	4.76	8.7	45.0				
GRID	28	10	4.76	8.7	60.0				
GRID	29	10	4.76	8.7	75.0				
GRID	30	10	4.76	8.7	90.0				
GRID	31	10	4.76	10.9	0.0				
GRID	32	10	4.76	10.9	15.0				
GRID	33	10	4.76	10.9	30.0				
GRID	34	10	4.76	10.9	45.0				
GRID	35	10	4.76	10.9	60.0				
GRID	36	10	4.76	10.9	75.0				
GRID	37	10	4.76	10.9	90.0				
\$									
CQUAD4	1	1	1	2	3	4			
CQUAD4	2	1	2	5	6	3			
CQUAD4	3	1	3	6	7	8			
CQUAD4	4	1	4	3	8	9			
CQUAD4	5	1	5	10	11	6			
CQUAD4	6	1	6	11	12	7			

CQUAD4	7	1	7	12	13	14	
CQUAD4	8	1	8	7	14	15	
CQUAD4	9	1	9	8	15	16	
CQUAD4	10	1	10	17	18	11	
CQUAD4	11	1	11	18	19	12	
CQUAD4	12	1	12	19	20	13	
CQUAD4	13	1	13	20	21	14	
CQUAD4	14	1	14	21	22	15	
CQUAD4	15	1	15	22	23	16	
CQUAD4	16	1	17	24	25	18	
CQUAD4	17	1	18	25	26	19	
CQUAD4	18	1	19	26	27	20	
CQUAD4	19	1	20	27	28	21	
CQUAD4	20	1	21	28	29	22	
CQUAD4	21	1	22	29	30	23	
CQUAD4	22	1	24	31	32	25	
CQUAD4	23	1	25	32	33	26	
CQUAD4	24	1	26	33	34	27	
CQUAD4	25	1	27	34	35	28	
CQUAD4	26	1	28	35	36	29	
CQUAD4	27	1	29	36	37	30	
\$							
PSHELL	1	1	0.01576	1			
MAT1	1	10.0+6		0.3	2.45-4		
\$							
SPC1	1	12456	1				
SPC1	2	246	2	5	10	17	24
SPC1	2	156	4	9	16	23	30
SPC1	3	123456	31	THRU	37		
SPC1	4	6	3	6	7	8	
SPC1	4	6	11	THRU	15		
SPC1	4	6	18	THRU	22		
SPC1	4	6	25	THRU	29		
SPCADD	1234	1	2	3	4		
\$							
DAREA	10	1	3	-25.0			
TLOAD1	100	10			2		
TABLED1	2						+TBD1
+TBD1	0.0	1.0	500.0-6	1.0	ENDT		
ENDDATA							

Chapter 9

NONLINEAR TRANSIENT RESPONSE ANALYSIS

9.1 USER INTERFACE

The input data required for SOL 99 or SOL 129 is a combination of direct time integration control data, similar to SOL 69 (for direct linear transient with superelements), and nonlinear modeling data similar to SOL 66 (for nonlinear statics). The nonlinear properties and/or effects are defined by nonlinear material data (MATS1 and TABLES1), gap elements (GAP) for nonlinear interface, and PARAMeter LGDISP for geometric nonlinearity. The transient effects are produced by time-dependent loading functions (TLOADi, DAREA, LSEQ, etc.), damping (parameters, elements and material data), and mass properties. Massless degree of freedom should be avoided for numerical stability and the small damping is also recommended.

The unique data required for SOL 99 or 129 is supplied on the TSTEPNL Bulk Data entry. The TSTEPNL entry in itself is a combination of the TSTEP entry for direct time integration and the NLPARM entry for nonlinear iteration control. Restarts are controlled by parameters (LOOPID, STIME, SLOOPID and SDATA) which can be specified either in the Case Control section or Bulk Data section. Some optional parameters (TSTATIC, NDAMP) are provided for additional control or capabilities.

9.1.1 Case Control

Each subcase defines a time interval starting from the last time step of the previous subcase and the interval is subdivided into small time steps. The output time is labeled by the cumulative time, including all previous subcases. There are advantages to divide the total duration of analysis into many subcases so that each subcase does not have excessive number of time

steps, e.g., not to exceed 200 steps. The data blocks containing solutions are generated at the end of each subcase to store in the database for output process and restarts. As such, converged solutions are apt to be saved at many intermediate steps in case of divergence and more flexible control becomes possible with multiple subcases.

The input loading functions may be changed for each subcase or continued by repeating the same DLOAD request. However, it is recommended to use the same TLOAD Bulk Data for all the subcases in order to maintain the continuity between subcases, because TLOADi data defines the loading history as a function of cumulative time. Static loads (PLOADi, FORCEi, MOMENTi) may be associated with time-dependent functions by the Bulk Data LSEQ which can be selected by a Case Control command LOADSET. However, thermal loads or enforced displacements (SPCD) are not allowed in the nonlinear transient analysis. Nonlinear forces as function of displacements or velocities (NOLINI) may be selected and printed by Case Control commands NONLINEAR and NLLOAD, respectively. Each subcase may have a different time step size, time interval, and iteration control selected by the TSTEPNL request. The Case Control requests which may not be changed after the first subcase are: SPC, MPC, DMIG, and TF.

Output requests for each subcase are processed independently. Requested output quantities for all the subcases are appended after the computational process for actual output operation. Available outputs are DISPLACEMENT, VELOCITY, ACCELERATION, OLOAD, STRESS, FORCE, SDISPLACEMENT, SVELOCITY, SACCELERATION, NONLINEAR (NLLOAD) and SPCFORCE. However, element force output and GRID point stresses are not available for nonlinear elements.

Initial conditions (displacement or velocity) can be specified by the Bulk Data TIC selectable by the Case Control command IC. If initial conditions were given, all of the nonlinear element forces and stresses must be computed to satisfy equilibrium with the prescribed initial conditions. On the other hand, initial conditions could be a part of the nonlinear analysis by applying static analysis for the preload using PARAM, TSTATIC in the first subcase. Then the transient analysis can be performed in the ensuing subcases. Associated with the adaptive time stepping method, the PARAMETER NDAMP is used to control the stability in the ADAPT method. The parameter NDAMP represents the numerical damping (recommended value for usual case is 0.01) which often required to improve the stability and convergence in the contact problems.

All the superelement model generation options and matrix reduction options are allowed for the linear portion of the structure. The generalized dynamic reduction, component mode synthesis, and Guyan reduction may be performed for upstream superelements. The residual superelement may contain scalar degree of freedom representing linear modal formulations.

9.1.2 Implicit Integration Control: TSTEPNL Data

The input fields of the TSTEPNL Bulk Data entry specify the time step size, the number of steps, and the output interval as well as the nonlinear iteration options. The TSTEPNL Bulk Data is selectable by the Case Control command TSTEPNL. Although the same TSTEPNL Bulk Data may be selected by more than one subcase, it is recommended to select different TSTEPNL entry for each subcase in preparation for changes in the restarts.

The choice of time step size is determined primarily by the frequency content of the input load and the highest frequency mode-of-interest. A general guideline is that 7 or more steps per cycle be provided for reasonable accuracy. Modes with shorter periods (higher frequency) will be attenuated by the numerical process. Highly nonlinear problems may require smaller step size. However, the adaptive time stepping capability is the vital part of SOL 99 or 129 and is recommended for any problem (linear or nonlinear). Nevertheless, the initial time step size should be estimated by the user according to the aforementioned principles.

A caution is necessary in using the automatic time step adjustment if the forcing function is a short duration pulse. Since the automatic time step adjustment algorithm does not consider the loading history, short pulses could be skipped if the time step is automatically adjusted to a larger value than the pulse duration. It is advised that a drastic change in the time step size between subcases be avoided. The drastic change, e.g., ratio exceeding 1000, could cause a loss of accuracy upon subcase switch. In such case, an intermediate subcase should be provided for a transition period of short interval to reduce the ratio.

The parameters for controlling the accuracy and stability of the incremental and iterative process can be specified in the TSTEPNL Bulk Data entry. The controls are applicable to the automatic time step adjustment and bisection process in addition to stiffness matrix updates, BFGS updates and line searches similar to those on the NLPARM Bulk Data entry. Since default values are resulted from numerous test runs, the analysis should be started with the default setting and changed if necessary. The TSTEPNL data format (Version 67) is shown below with default values:

Format:

1	2	3	4	5	6	7	8	9	10
TSTEPNL	ID	NDT	DT	NO	METHOD	KSTEP	MAXITER	CONV	+TNL1
+TNL1	EPSU	EPSP	EPSW	MAXDIV	MAXQN	MAXLS	FSTRESS		+TNL2
+TNL2	MAXBIS	ADJUST	MSTEP	RB	MAXR	UTOL	RTOLB		

Example with defaults:

1	2	3	4	5	6	7	8	9	10
TSTEPNL	250			1	ADAPT	2	10	PW	+TNL1
+TNL1	1.E-2	1.E-3	1.E-6	2	10	2	0.2	+TNL2	
+TNL2	5	5	0	0.75	16.0	0.1	20.		

The TSTEPNL Bulk Data entry is selected using ID by the Case Control command TSTEPNL. Each subcase (residual superelement solutions only) requires a TSTEPNL entry. Multiple subcases are assumed to occur sequentially in time. Therefore, the initial conditions of each subcase are defined by the end conditions of the previous subcase.

The NDT field specifies the number of time steps with DT as the size of each time step. The total duration for the subcase can be assessed by multiplying NDT with DT (i.e., $NDT \cdot DT$). The time increment (Δt) remains constant during the analysis in AUTO and TSTEP options, and is equal to DT. However, the time increment (Δt) changes during the analysis in the ADAPT option and the actual number of time steps will not be equal to NDT. In ADAPT option DT is used as an initial value for Δt .

The NO field specifies the time step interval for output, i.e., every NO^{th} step solution is saved for output. The data will be output at steps 0, NO, 2NO, ... etc., and the last converged step for printing and plotting purposes. The Case Control command OTIME may also be used to control the output points.

The METHOD field selects an option for direct time integration and the stiffness matrix update strategies among ADAPT, AUTO and TSTEP. If the AUTO option is selected, the program automatically updates the stiffness matrix to improve convergence while KSTEP value is ignored. If the TSTEP option is selected, the program updates the stiffness matrix every $KSTEP^{th}$ increment of time. If the ADAPT option is selected, the program automatically adjusts the incremental time and uses the bisection algorithm in case of divergence. During the bisection process in ADAPT option, stiffness is updated at every $KSTEP^{th}$ successful bisection. The ADAPT method allows linear transient analysis, but AUTO or TSTEP will abort the run if the model does not have any data representing nonlinearity. The stiffness matrix is always updated for a new subcase or restart, irrespective of the option selected.

The number of iterations for a time step is limited to MAXITER. If the solution does not converge in MAXITER iterations, the process is treated as a divergent process, i.e., either a bisection or stiffness matrix update takes place based on the value of MAXBIS. The sign of MAXITER provides a control over the ultimate recourse (reiteration) in case of failure in convergence or bisection. If the MAXITER is negative, the analysis is terminated when the divergence condition is encountered twice during the same time step or the solution diverges for five consecutive time steps. If MAXITER is positive, the program computes the best attainable solution and continues the analysis on second divergence.

The convergence test is controlled by convergence test flags (U for displacement error test, P for load equilibrium error test, W for work error test) and the error tolerances (EPSU, EPSP and EPSW) which define the convergence criteria. All requested criteria (combination of U, P, and/or W) are satisfied upon convergence. It should be noted that at least two iterations are necessary to check the displacement convergence criterion.

The MAXDIV field provides control over diverging solutions. Depending on the rate of divergence, the number of diverging solutions (NDIV) is incremented by 1 or 2. The solution is assumed to be divergent when NDIV reaches MAXDIV during the iteration. If the bisection option is used with the ADAPT method, the time step is bisected upon divergence. Otherwise, the solution for the time step is repeated with a new stiffness based on the converged state at the beginning of the time step. If NDIV reaches MAXDIV twice within the same time step, the analysis is terminated with a fatal message.

The BFGS quasi-Newton updates and the line search process work in the way as in static analysis except for the default setting. The MAXQN field defines the maximum number of quasi-Newton vectors to be saved on the database and the MAXLS defines the number of line searches allowed per iteration. Non-zero values of MAXQN and MAXLS activate the quasi-Newton update and the line search process, respectively.

The FSTRESS field defines a fraction of the effective stress ($\bar{\sigma}$) which is used to limit the subincrement size in the material routine. The number of subincrements in the material routines is determined such that the subincrement size is approximately $FSTRESS * \bar{\sigma}$ (equivalent stress). FSTRESS is also used to establish a tolerance for error correction in the elasto-plastic material, i.e.,

$$Error \text{ in yield function} < FSTRESS * \bar{\sigma}$$

If the limit is exceeded at the converging state, the program will exit with a fatal error message. Otherwise, the stress state is adjusted to the current yield surface.

MAXBIS is the maximum number of bisections allowed for each time step ($-9 \leq MAXBIS \leq 9$). The bisection process is activated when divergence occurs and $MAXBIS \neq 0$. The number of bisections for a time increment is limited to $|MAXBIS|$. If MAXBIS is positive and the solution does not converge after MAXBIS bisections, the best solution is computed and the analysis is continued to the next time step. If MAXBIS is negative and the solution does not converge in $|MAXBIS|$ bisection, the analysis is terminated.

The parameter ADJUST allows the user to control the automatic time stepping in the ADAPT option. A value of zero for ADJUST turns off the automatic adjustment completely. If ADJUST is positive, the time increment is continually adjusted for the first few steps until a good value of Δt is obtained. After this initial adjustment, the time increment is adjusted every ADJUST time steps only. A value of ADJUST an order greater than NDT will turn off adjustments after the initial adjustment. Since the automatic time step adjustment is based on the mode of response and not on the loading pattern, it may be necessary to limit the adjustable step size when the period of the forcing function is much shorter than the period of dominant response frequency of the structure. It is user's responsibility to ensure that the loading history is properly traced with ADJUST option. The ADJUST option should be suppressed for the duration of short pulse loading. If unsure, start with DT which is much smaller than the pulse duration in order to properly represent the loading pattern.

MSTEP defines the desired number of time steps to obtain the dominant period response accurately ($10 \leq \text{Integer} \leq 200$): RB defines bounds for maintaining the same time step for the stepping function in the automatic time step adjustment method ($0.1 \leq \text{Real} \leq 1.0$). Parameters MSTEP and RB are used to adjust the time increment during the analysis in the ADAPT option. The adjustment is based on the number of time steps desired to capture the dominant frequency response accurately. The time increment is adjusted as follows:

$$\Delta t_{n+1} = f(r) \Delta t_n$$

where

$$r = \frac{1}{\text{MSTEP}} \left(\frac{2\pi}{\omega_n} \right) \left(\frac{1}{\Delta t_n} \right)$$

with

$$\begin{aligned} f &= 0.25 \text{ for } r < 0.5 * \text{RB} \\ f &= 0.5 \text{ for } 0.5 * \text{RB} \leq r < \text{RB} \\ f &= 1.0 \text{ for } \text{RB} \leq r < 2 \\ f &= 2.0 \text{ for } 2. \leq r < 3./\text{RB} \\ f &= 4.0 \text{ for } r \geq 3./\text{RB} \end{aligned}$$

The recommended value of MSTEP for nearly linear problems is 20. A larger value (e.g., 40) is required for highly nonlinear problems. In the default option, the program automatically computes the value of MSTEP based on the changes in the stiffness.

The MAXR field defines the maximum ratio for the adjusted incremental time relative to DT allowed for time step adjustment ($1.0 \leq \text{Real} \leq 32.0$). MAXR is used to define the upper and lower bounds for adjusted time step size, i.e.,

$$\text{MIN} \left(\frac{\text{DT}}{2\text{MAXR}}, \frac{\text{DT}}{\text{MAXR}} \right) \leq \Delta t \leq \text{MAXR} * \text{DT}$$

The UTOL field defines the tolerance on displacement increment below which there is no time step adjustment ($0.001 < \text{Real} \leq 1.0$). UTOL is used to filter undesirable time step

adjustment, i.e., no time step adjustment is performed if

$$\frac{\|\dot{U}_n\|}{\|\dot{U}\|_{max}} < \text{UTOL}$$

The RTOLB field defines the maximum value of incremental rotation (in degrees) allowed per iteration to activate bisection (Real > 2.0). The bisection is activated if the incremental rotation for any degree-of-freedom ($\Delta\theta_x$, $\Delta\theta_y$ or $\Delta\theta_z$) exceeds the value specified for RTOLB. This bisection strategy based on the incremental rotation is controlled by MAXBIS field.

9.1.3 Iteration Related Output Data

During the incremental and iterative computation, the process information consisting of iteration data is printed at the end of each iteration or time step. They are printed under the following heading:

TIME: Cumulative time for the duration of the analysis

ITER: Iteration count for each time step

DISP: Relative error in terms of displacements defined as

$$E_u^i = \frac{\lambda^i \|u^i - u^{i-1}\|}{(1 - \lambda^i) u_{max}} \quad \text{for ADAPT method}$$

$$E_u^i = \frac{\|u^i - u^{i-1}\|}{(1 - \lambda^i) u_{max}} \quad \text{for AUTO or TSTEP}$$

where $u_{max} = \max(\|u_1\|, \|u_2\|, \dots, \|u_n\|)$

LOAD: Relative error in terms of load vectors defined as

$$E_p^i = \frac{\|R^i\|}{\max(\|F_n\|, \|P_n\|)} \quad \text{for ADAPT method}$$

$$E_p^i = \frac{\|R^i\|}{\max(\|\bar{F}\|, \|\bar{P}\|)} \quad \text{for AUTO or TSTEP}$$

where \bar{F} and \bar{P} are three-point average values for internal and external forces, respectively.

WORK: Relative error in terms of work defined as

$$E_w^i = \frac{\{u^i - u^{i-1}\}^T R^i}{\max(\{u_n\}^T \{F_n\}, \{u_n\}^T \{P_n\})}$$

LAMBDA(I): Rate of Convergence in iteration where

$$\lambda^i = \frac{E_p^i}{E_p^{i-1}}$$

LAMBDA(T): Ratio of the load error for two consecutive time steps computed only for AUTO or TSTEP method

$$\lambda_t^n = \frac{E_p^n}{E_p^{n-1}}$$

LAMBDA-BAR: Average of LAMBDA(T) over the last 3 steps, computed only for AUTO or TSTEP method

DLMAG: Absolute norm of the load vector, $\| R \|$. The absolute convergence is defined using DLMAG by $\| R \| < 10^{-12}$.

FACTOR: Final value of the line search parameter

E-FIRST: Divergence rate, error before line search

E-FINAL: Error at the end of line search

NQNV: Number of quasi-Newton vectors appended

NLS: Number of line searches performed during the iteration

ITR DIV: Number of occurrences of divergence detected during the adaptive iteration by the iteration module NLTRD2

MAT DIV: Number of occurrences of bisection conditions in the material routine (excessive stress increment) or in the rotation angle (excessive rotation) during the iteration using the ADAPT method

NO. BIS: Number of bisections executed for the current time interval

ADJUST: Ratio of time step adjustment relative to DT within a subcase

9.2 RESTARTS

Since SOL 66 (or 106) and SOL 99 (or 129) share the same database storage for nonlinear tables and matrices, the restart system for transient analysis can use either a previous static or transient nonlinear analysis as its initial conditions.

9.2.1 Restarting from SOL 66 into SOL 99

The options for a restart from SOL 66 into SOL 99 are static to static and static to dynamic analysis. For a restart from a previous static analysis, only the first subcase is affected. Simply provide a database created in SOL 66 and specify the parameter

. PARAM, SLOOPID, N

where N is the printed value of LOOPID for the desired static solution. The initial transient load should be identical to static loads at the restart state. Constraint sets, direct input matrices, mass, and damping may be changed.

It is noted that SOL 99 destroyed (until Version 66) the database for any subsequent static analysis once a SOL 66 database is used for restart. It is recommended for the user to restart from a copy of the input SOL 66 database. This deficiency is corrected in Version 67.

9.2.2 Restarting within SOL 99

Restarting within SOL 99 allows static to static, static to dynamic, and dynamic to dynamic analysis. Restarts from a previous nonlinear transient execution are available for a number of cases. If the same model is to be reexecuted, only the residual superelement needs to be reassembled (SEMA, SELA = 0). If the final results from the previous transient run are to be used as the initial conditions at $t=0$, add N dummy SUBCASE commands to start the residual Case Control execution and STIME = 0.

The normal restart for a transient run is to be continued from the last step of a previous subcase with different loads and/or TSTEPNL data. For the normal restart provide the following parameters:

LOOPID = N : Start from the N-th subcase
STIME = t : Start from time t

Note that constraint sets should not be changed to avoid incompatible matrix sizes. The values of LOOPID and STIME, which are printed with the iteration information for each subcase, can be directly read from the printout of the previous run. If the adaptive time stepping algorithm

is not activated, the value of t may also be calculated by the following equation:

$$t = \sum_{i=1}^N \text{NDT}_i * \text{DT}_i;$$

where NDT_i and DT_i are the number of time steps and the time increment of the i -th subcase, respectively.

If a SOL 99 run is terminated abnormally in the middle of a subcase, it may or may not be restartable depending upon the cause of the abnormal exit. If the job is stopped due to a diverging solution, it can be restarted either from the end of a previous subcase or from the last saved solution step. The restart procedure for the former is identical to that for the normal restart as described in the preceding paragraph. The latter case also requires parameters LOOPID and STIME, however, the input value for STIME differs depending on the value of METHOD specified on the TSTEPNL entry. If METHOD = AUTO or TSTEP (the NLTRD module), STIME is the time corresponding to the last output step which may be calculated based on the output skip factor (i.e., the NO on the TSTEPNL entry). If METHOD = ADAPT (NLTRD2 module), which is a new feature implemented in Version 66, the last converged solution is treated as an output step and is always saved for the restart so that STIME can be the time of the last converged step. The values of STIME and LOOPID can also be found in the printout, if the ADAPT method is used. Once STIME and LOOPID are known, determine the number of remaining time steps in the subcase and create a new TSTEPNL entry for the remaining time. Insert a new subcase which references the new TSTEPNL entry prior to the remaining subcases in the Case Control section.

A solution may be terminated in the middle of a subcase due to insufficient CPU time: (1) the CPU time specified in the Executive Control section is insufficient so the run is forced to exit by the program, or (2) the CPU time exceeds the limit specified in the computer system which leads to a sudden job abortion by the system. In the first case, MSC/NASTRAN is able to detect the specified CPU time in the Executive Control section and automatically activate the wrapping-up procedure before time expiration. When completed, the solution can be restarted from the termination point as in the solution diverging case. In the second case the solution can only be restarted from the end of a subcase.

Restarts may also be performed solely for data recovery by providing the following parameters:

SDATA = -1 : Recover data without running the solution module
LOOPID = N : from the 1st through the N-th subcases.

Note that solution sets DISP, VELO, ACCE, OLOAD, SPCF (printout and plotting) and NLLOAD (plotting only) are recoverable while STRESS, SDISP, SVELO and SACCE sets are not.

9.2.3 Example

A simply supported elastic beam is subjected to a single cycle of a sinusoidal forcing function at the quarter span as illustrated in Fig. 9.2.1. The beam is modeled using 20 linear BEAM elements. A stopper (snubber) is present underneath the center of the beam with a clearance of 0.02 inches. This gap is simulated by a displacement dependent force which is active only when the beam is in contact with the snubber. The problem is linear except for the effect of the stopper. The input data for this problem is given in Table 9.2.1.

One subcase with 100 time steps and an initial time step size of 0.1 msec was analyzed in the cold start run. Since the adaptive time stepping algorithm was activated, the job ran to completion at 9.7 msec instead of the specified time span, $100 \times 0.1 = 10$ msec. A new subcase with 600 time steps was added in the first restart run to extend the time span for another 30 msec. The adaptive time stepping method was turned off and the execution time was specified to be 0.8 min. As expected, the run was aborted at 36 msec due to insufficient time. In the second restart run, the execution time was changed to 5.0 min. and another subcase with 164 time steps and a time increment of 0.1 msec was added to extend the analysis time span up to 40 msec. The third restart was simply to recover the DISP, VELO, ACCE, OLOAD, NLLOAD and SPC output. The response at the loading point of the beam is shown in Fig. 9.2.2.

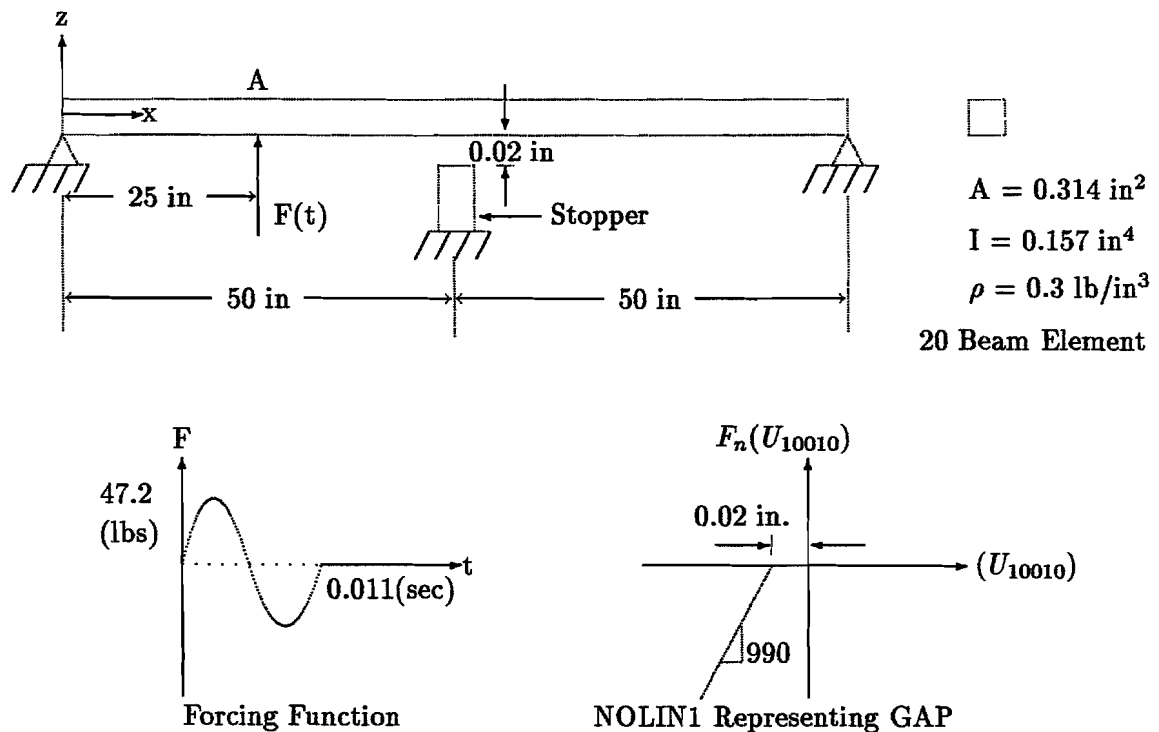


Figure 9.2.1. Simply Supported Beam With a Restrained Motion

VERT.
DISPL.
AT
POINT A

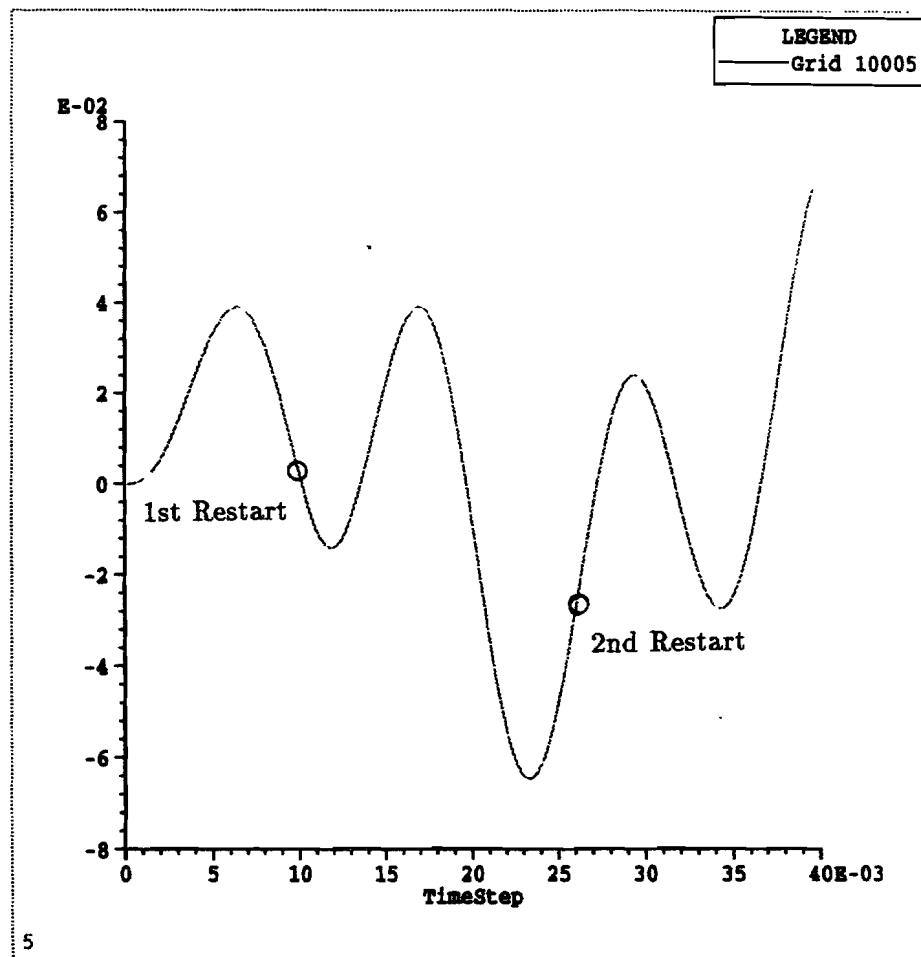


Figure 9.2.2. Simply Supported Beam: Loading Point Response

Table 9.2.1. Input Data Listing for Simply Supported Beam

```

$RESTART          $ ***added for the restart runs
ID TNOLIN, V66 $ SSH 7/10/91
TIME 5            $
SOL 99            $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50         $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = DYNAMICS HANDBOOK DEMONSTRATION PROBLEM           D9918
SUBTITLE = DIRECT TRANSIENT RESPONSE, NONLINEAR FORCE
LABEL = NOLIN IN SOL 99
SEALL = ALL
ECHO = UNSORT
  SET 1 = 10005
  SET 2 = 10010
  SET 3 = 10000
SPC = 1002
$-----
$PARAM,LOOPID,1      *** used for the 1st restart
$PARAM,STIME,0.0093
$-----
$PARAM,LOOPID,2      *** used for the 2nd restart
$PARAM,STIME,0.0236
$-----
$ DISPL=2
$ VELO=2             *** used for the 3rd restart
$ ACCEL=2
$ OLOAD=1
$ NLLOAD=2
$ SPCF=3
$PARAM,LOOPID,3
$PARAM,SDATA,-1
$-----
SUBCASE 1
  DLOAD = 30
  TSTEPNL = 10
  NONLINEAR = 13 $ SELECT NONLINEAR FORCE
SUBCASE 2
  DLOAD = 30
  TSTEPNL = 20
  NONLINEAR = 13 $ SELECT NONLINEAR FORCE
SUBCASE 3
  DLOAD = 30
  TSTEPNL = 30
  NONLINEAR = 13 $ SELECT NONLINEAR FORCE
BEGIN BULK
$ MODELING INFORMATION FOR BEAM ONLY
CBAR  101    100    10000  10001  0.    0.    1.
CBAR  102    100    10001  10002  0.    0.    1.

```

CBAR	103	100	10002	10003	0.	0.	1.
CBAR	104	100	10003	10004	0.	0.	1.
CBAR	105	100	10004	10005	0.	0.	1.
CBAR	106	100	10005	10006	0.	0.	1.
CBAR	107	100	10006	10007	0.	0.	1.
CBAR	108	100	10007	10008	0.	0.	1.
CBAR	109	100	10008	10009	0.	0.	1.
CBAR	110	100	10009	10010	0.	0.	1.
CBAR	111	100	10010	10011	0.	0.	1.
CBAR	112	100	10011	10012	0.	0.	1.
CBAR	113	100	10012	10013	0.	0.	1.
CBAR	114	100	10013	10014	0.	0.	1.
CBAR	115	100	10014	10015	0.	0.	1.
CBAR	116	100	10015	10016	0.	0.	1.
CBAR	117	100	10016	10017	0.	0.	1.
CBAR	118	100	10017	10018	0.	0.	1.
CBAR	119	100	10018	10019	0.	0.	1.
CBAR	120	100	10019	10020	0.	0.	1.
CONM2	12	10010		.1			
GRID	10		50.		-1.		
GRID	10000		0.	0.	0.		1246
GRID	10001		5.	0.	0.		1246
GRID	10002		10.	0.	0.		1246
GRID	10003		15.	0.	0.		1246
GRID	10004		20.	0.	0.		1246
GRID	10005		25.	0.	0.		1246
GRID	10006		30.	0.	0.		1246
GRID	10007		35.	0.	0.		1246
GRID	10008		40.	0.	0.		1246
GRID	10009		45.	0.	0.		1246
GRID	10010		50.	0.	0.		1246
GRID	10011		55.	0.	0.		1246
GRID	10012		60.	0.	0.		1246
GRID	10013		65.	0.	0.		1246
GRID	10014		70.	0.	0.		1246
GRID	10015		75.	0.	0.		1246
GRID	10016		80.	0.	0.		1246
GRID	10017		85.	0.	0.		1246
GRID	10018		90.	0.	0.		1246
GRID	10019		95.	0.	0.		1246
GRID	10020		100.	0.	0.		1246
MAT1	1000	3.+7		.3	.3		
PARAM	GRDPNT	10010					
PARAM	WTMASS	.002588					
PBAR	100	1000	.31416	.15708			+PB1
+PB1	1.	0.					
SPC	1002	10	123456				
SPC	1002	10020	3		10000	3	
\$							

\$ MODELING INFORMATION FOR CENTER SPRING

CROD	10	10	10	10010			
MAT1	10	10.		0.			
PROD	10	10	1.				
MATS1	10		PLASTIC	0.	1	1	3.+8

\$ LOADING AND SOLUTION INFORMATION

TLOAD2	30	33			0.	.011451	87.33	-90.	
DAREA	33	10005	3	47.2					
TSTEPNL	10	100	.0001	1	ADAPT				
TSTEPNL	20	600	.00005	1	ADAPT				+TS21
+TS21									+TS22
+TS22		0							
TSTEPNL	30	164	.00010	1	ADAPT				+TS31
+TS31									+TS32
+TS32		0							

\$

\$ MODELING INFORMATION FOR NONLINEAR SPRING

NOLIN1	13	10010	3	1.	10010	3	13	
TABLED1	13							+TAB
+TAB	-2.5-2	4.95	-2.0-2	0.	0.	0.	ENDT	

\$

ENDDATA

9.3 DYNAMIC EFFECTS: MASS AND DAMPING

9.3.1 Mass Modeling

Mass properties of the finite element model may be specified in several different ways, including mass elements (CONM1, CONM2, CMASSi), parameters (WTMASS, COUPMASS, GRDPNT) and fields in material or property Bulk Data entries. They are described below:

- The density (mass per unit volume) is specified in the RHO field of a MATi Bulk Data entry and is used to automatically compute the mass for all structural elements which reference the MATi entry.
- Nonstructural mass distributed over elements (mass per unit length for line elements or mass per unit area for surface elements) is specified in the NSM field of property Bulk Data entries. Applications of this feature are floor loads, electrical cables or thermal insulating materials.
- The mass elements have inertia properties directly specified by the user.
 - The CONM1 Bulk Data entry provides a general description of a concentrated mass at a grid point of the structural model by defining a 6x6 symmetric matrix of mass coefficients at the grid point. Applications of CONM1 are rare.
 - The CONM2 Bulk Data entry provides a more convenient definition of a concentrated mass at a grid point. The inputs to CONM2 are the mass, the offset of the center of mass from the grid point, and the moments and products of inertia about the center of mass. As an option, the center of mass may be measured from the origin of the basic coordinate system rather than as an offset from the grid point.
 - The CMASSi Bulk Data entries provide mass coupling between any two degrees of freedom. The form of the relationship is

$$\begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = \begin{bmatrix} M & -M \\ -M & M \end{bmatrix} \begin{Bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \end{Bmatrix}$$

where f_1 and f_2 are the inertia forces acting at degrees of freedom 1 and 2, respectively, and M is the mass coefficient (specified on the CMASSi entry or on the PMASS entry if $i = 2$ or 4). In most applications, the second degree of freedom is not specified. If this is the case, the inertia force $f_i = -M\ddot{u}_1$ is generated, or in other words, M is added to the mass matrix in the diagonal position corresponding to u_1 . An important application of the CMASSi entry occurs in the recommended method for specifying enforced motion at grid points. Another application is for modeling an inertia term in relative coordinates. If the u_1 and u_2 terms are both defined on the CMASSi entry, then inertia forces will result only when \ddot{u}_1 is not equal to \ddot{u}_2 .

- The three parameters which relate to the specification of mass properties are
 - PARAM, WTMASS, V1 specifies a factor by which to multiply the mass input data to obtain dimensionally correct mass. For example, if the in-lb-sec system is used and the mass input is specified in weight pounds, then $V1 = \frac{1}{386} = 0.00259$.
 - PARAM, COUPMASS, 1 requests that the coupled mass option be used rather than the lumped mass option (default) for all elements which support the coupled mass option. In the lumped mass option, the distributed mass of an element is simply divided up and the parts are located at the connected grid points. In the coupled mass option, the distributed mass is replaced by a nondiagonal mass matrix which couples the connected grid points. The coupled mass option is generally more accurate, but uses more computer time. The lumped mass option is therefore preferred for computational speed.
 - PARAM, GRDPNT, V1 causes the Grid Point Weight Generator to be executed. The value of V1 identifies a grid point at which the rigid body mass properties of the structure will be computed. This capability can be used as a check on mass and geometric input data.

9.3.2 Mass Matrix

For transient response analysis, the mass matrix is formulated as

$$[M] = [M_1] + [M_2] \quad (9.3.1)$$

where M is the mass matrix, M_1 contains mass terms from the following sources

- A 6x6 matrix of mass coefficients at a grid point defined on a CONM1 entry
- A concentrated mass element defined on a CONM2 entry
- Structural mass for all elements, except plate elements without membrane stiffness, using the mass density on the MATi entry
- Nonstructural mass for all elements specifying a value on the property entry
- Scalar masses defined on CMASSi entries

and M_2 corresponds to the direct input matrices which are generated by transfer functions specified by TF Bulk Data entries or supplied directly via DMIG or DMIAX Bulk Data entries.

Unlike the stiffness matrix, the mass matrix is generated only once and is not updated during the analysis. This situation should not cause much concern. The mass matrix for the deformed structure is the same for the undeformed structure in material nonlinear analysis. During large displacement analysis, precautions can be taken so that the mass matrix for the deformed structure is very similar to the one for the undeformed structure.

The mass matrix is defined in the global coordinate system using the initial undeformed geometry. During large displacement the structure deforms and rotates relative to the global coordinate system. This causes the element mass distribution and rotational inertias to change. The element mass distribution due to element deformation is usually minimal and will in most cases cause little error. The changes in the rotational inertia due to element rotation and elongation are more severe and may cause large errors. The following guidelines will help avoid most problems when using the large displacement option.

1. Do not use the coupled mass option (PARAM,COUPMASS,1). The coupled mass option generates rotational inertia terms for many of the elements. These terms are defined in the global coordinate system. As the elements rotate, the rotational inertia terms defined in the global coordinate system change. This change, however, is not reflected in the mass matrix since it is formulated only once and is not updated. By using the lumped mass option (default), only translational terms are generated. Translational terms are unaffected by element distortions.
2. If concentrated masses are used, then specify the same mass on all three translational degrees of freedom. This is required to keep the inertial forces consistent with the elastic forces of the structure.

Consider the following example: a cantilevered beam with a mass at the free end defined only in the y-direction. Initially the elastic bending and shear forces are in-line with the inertial forces of the mass. As the beam is deformed by an end load the elastic forces follow the deformation (stiffness matrix is updated). The mass is only defined in the y-direction and will resist only the y component of the elastic force. The x component of the elastic force will not be resisted. In the extreme case the elastic force is unresisted.

3. As with element rotational inertias, the same situation exists for concentrated rotational inertias. Rotational inertias about the three directions of the global coordinate are usually different. As the actual structure rotates at the concentrated mass point, the concentrated mass will also rotate with respect to the global coordinate system. As it rotates, the inertias about the directions of the global system change. This change in rotational inertia is not taken into account by the mass matrix. For this reason, concentrated rotational inertia terms should not be used.
4. If rotational inertia terms are necessary then a rigid structure with concentrated translational masses should be used. This rigid structure should emulate the mass properties of the concentrated mass. For example, a user wishes to simulate a rigid mass with the following properties:

- Total mass = 10 kg
- $I_{xx} = 10 \text{ kg} \cdot \text{cm}^2$
- $I_{yy} = 20 \text{ kg} \cdot \text{cm}^2$
- $I_{zz} = 30 \text{ kg} \cdot \text{cm}^2$

where the principal axes of the mass align with the global coordinate system. The rotational inertia will be modeled by lumped masses (two mass points of equal magnitude in each direction: m_x , m_y and m_z) at the ends of rigid bars (stiff CBEAMs). The bars will extend an equal distance in the positive and negative directions from the grid along the global coordinate system axes. The bar lengths and concentrated masses must satisfy the following requirements:

- $2m_x + 2m_y + 2m_z = \text{total mass (10 kg)}$
- $2m_y l_y^2 + 2m_z l_z^2 = I_{xx} \text{ (10 kg} \cdot \text{cm}^2\text{)}$
- $2m_x l_x^2 + 2m_z l_z^2 = I_{yy} \text{ (20 kg} \cdot \text{cm}^2\text{)}$
- $2m_x l_x^2 + 2m_y l_y^2 = I_{zz} \text{ (30 kg} \cdot \text{cm}^2\text{)}$

where l_x , l_y and l_z are lengths of the rigid bars in the x, y and z-axes, respectively. There are more unknowns than equations. In this example $m_x = 1 \text{ kg}$, $m_y = 1 \text{ kg}$, and $m_z = 3 \text{ kg}$ were selected. Solving for l_x , l_y , and l_z results in $l_x = \sqrt{10} \text{ cm}$, $l_y = \sqrt{5} \text{ cm}$, and $l_z = 0 \text{ cm}$. This rigid structure will simulate the translational and rotational properties of the concentrated mass as the structure rotates because the motion of the rigid bars is taken into account during the stiffness matrix updating procedure.

9.3.3 Damping Modeling

Damping represents the energy dissipation observed in the structure and results from many sources including

- Viscous effects (dashpot, shock absorber)
- Internal friction (characteristic of material type, i.e., hysteresis)
- External friction (slippage in structural joints)
- Structural nonlinearities (plasticity)

In MSC/NASTRAN, damping is divided into two types: viscous and structural. The two types of damping perform different functions. Viscous damping is used to simulate actual structural damping elements whereas structural damping is used to simulate the inherent damping properties of materials. The two damping types can be combined to model the damping characteristics of a structure.

Viscous damping is input using discrete damping elements (CVISC and CDAMPi). Because stable damping situations are difficult to model using discrete elements, the CVISC and CDAMPi elements are not used to add stability to a structure during transient response analysis. They are used mainly to model actual damping components in a structure, such as hydraulic dampers and viscous interface material. When using the discrete elements in this situation, the major relative motion is assumed to be between the coupled degrees of freedom or along the initial axis connecting the grid points.

Structural damping is a global damping proportional to the stiffness. The matrix is formed from the input in the GE field of MATi or via PARAM G, PARAM W3, and PARAM W4. Structural damping is included in a model in an attempt to simulate the intrinsic material damping. The user should choose a realistic value (0.5% to 4%), dependent upon the material in the model. A light-weight metal structure may only have a 0.5% damping value, whereas a composite material may have a 4% damping value. A change in the damping value may have a significant effect on the calculation of the response. A minimum value of $0.03 \times W4 \times \Delta t$ is recommended for stability, regardless of the material.

The viscous damping force is proportional to velocity, i.e.,

$$m\ddot{u} + c\dot{u} + ku = p(t)$$

and the structural force is proportional to displacement, i.e.,

$$m\ddot{u} + (1 + ig)ku = 0$$

Viscous and structural damping are related by

$$c = g\omega_n m = \frac{g}{\omega_n} k$$

or

$$g = \frac{c\omega}{k} = 2\xi \frac{\omega}{\omega_n}$$

with

$$\xi = \frac{c}{2m\omega_n}$$

which results in an equivalence at resonance with

$$g = 2\xi .$$

Fig. 9.3.1 compares the viscous and structural damping forces as a function of frequency. Damping due to internal material friction and damping due to bolted or riveted joints behave more like structural damping than viscous damping.

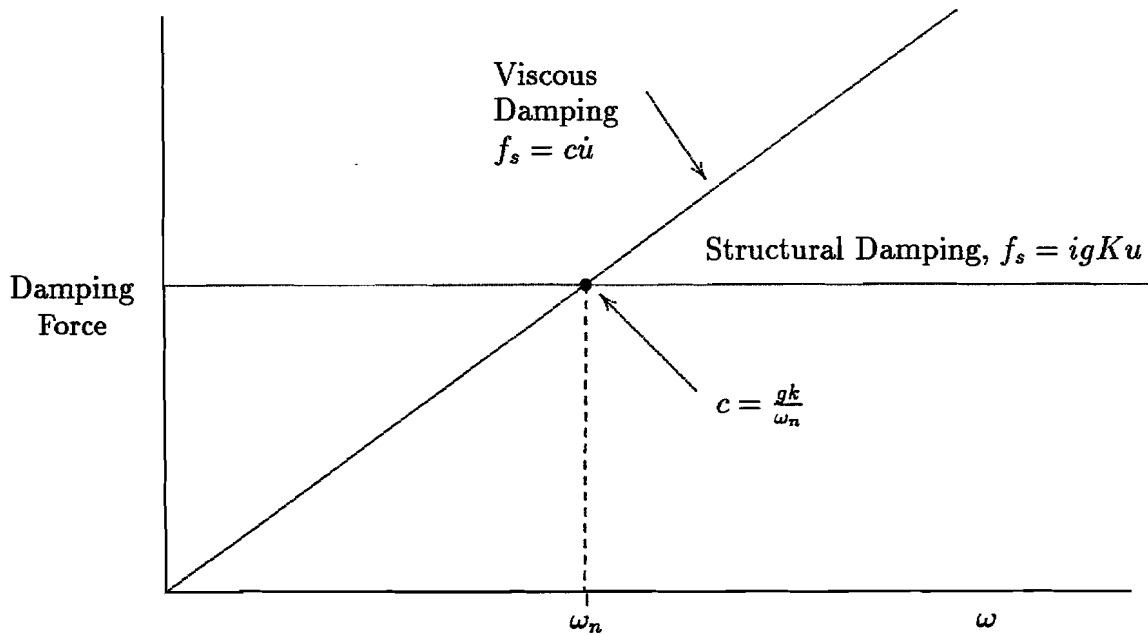


Figure 9.3.1 Comparison of Viscous and Structural Damping

Except for the overall structural damping parameter (G), all of the linear damping options may be used in nonlinear transient analysis. The damping input options in the finite element model are described below:

- The CVISC Bulk Data entry defines a viscous damping element that may be connected between any two grid points. It acts like a rod element with viscous extensional and torsional properties rather than elastic properties.
- The CDAMPi Bulk Data entries define a scalar damping element using a 2x2 viscous damping matrix between any two degrees of freedom. The form of the relationship is

$$\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} B & -B \\ -B & B \end{bmatrix} \begin{Bmatrix} \dot{u}_1 \\ \dot{u}_2 \end{Bmatrix}$$

where f_1 and f_2 are the damping forces acting at degrees of freedom 1 and 2, respectively, and B is the damping coefficient (specified on the CDAMPi entry or on the PDAMP entry if $i = 2$ or 4). Two degrees of freedom can be in any direction and may not act like a rod element. If the second degree of freedom is not specified, then $f_1 = B\dot{u}_1$, i.e., B is added to the damping matrix in the diagonal position corresponding to u_1 .

- Structural damping is specified in the GE field of a MATi Bulk Data entry. The effect of GE is to multiply the stiffness matrix for an element which references the MATi entry by the scalar factor $(1 + ig)$ where g is the value of the GE field and i is the operator for imaginary number.

- PARAM, G specifies an overall structural damping factor (default = 0.0) for the linear part of the stiffness matrix, in nonlinear transient analysis.
- PARAM, W3 selects the frequency (default = 0.0) to convert the overall structural damping on the linear elements to viscous damping.
- PARAM, W4 selects the frequency (default = 0.0) to convert the element structural damping (including nonlinear elements) to viscous damping.
- PARAM, NDAMP provides numerical damping in the ADAPT method of SOL 99 for numerical stability (implemented in Version 67). The numerical damping increases as the value of NDAMP increases, with zero being no numerical damping (default is 0.025).

9.3.4 Damping Matrix

For transient response analysis, the damping matrix is formulated as

$$B = B^1 + B^2 + \frac{G}{W_3} K^1 + \frac{1}{W_4} \sum_e G_e K_e \quad (9.3.2)$$

where

B¹ : contains terms from viscous damping elements (CVISC, CDAMP)

B² : direct input matrices generated by transfer functions (TF entries) or supplied by direct matrix input (DMIG, DMIAX)

G : Overall structural damping coefficient, specified by PARAM G

W₃ : frequency of interest, specified by PARAM W3

K¹ : global linear stiffness matrix

G_e : structural damping constant, specified on MATi

W₄ : frequency of interest, specified by PARAM W4

K_e : initial element stiffness matrix, generated for t=0

Like the mass matrix, the viscous damping matrix is generated only once and is not updated. For material nonlinear analysis, for instance, the viscous damping matrix for the deformed structure remains constant as for the undeformed structure based on the elastic material. For the geometric nonlinear analysis, the viscous damping matrix will be inaccurate for the deformed structure with large rotation because the direction has been changed and the damping matrix is not rotated. In both cases caution must be exercised when using discrete damping elements in transient analysis.

9.4 TRANSIENT LOADS AND INITIAL CONDITIONS

The methods employed to define loads in nonlinear transient analysis are similar to those used in the linear solutions. A single degree of freedom or a set of GRID points may be loaded with force pattern that varies with time. Functions may be tabular such as an earthquake or a booster liftoff, or they may be simple analytic functions such as a sine wave. Simple static load sets generated in both upstream superelements and the default residual may be used to create the dynamic loads. They may be scaled and combined with other loads to simulate complex loading problems.

For special problems involving simple scalar nonlinearities, the traditional nonlinear loads (NOLINi option) may be used in both types of transient solutions. These are useful for simulating nonlinear damping mechanisms and rotational coupling not provided by the GAP elements.

9.4.1 Transient Loads

Transient loads define the loadings as functions of time and the location. They can be a load applied at a particular degree of freedom, pressure over the surface area, or the body force simulating an acceleration. The time history is provided by TLOADi Bulk Data and the static loads are converted to dynamic loads using LSEQ option. The transient dynamic loads are selected in the Case Control section, controlled by the following commands:

DLOAD: Required to select TLOADi or DLOAD Bulk Data loads. Each stage of the transient solution may be defined with a separate SUBCASE, each with its own DLOAD request. The DLOAD data is restricted to act only in the time defined in its subcase.

LOADSET: Required for LSEQ processing. Must be placed above the solution subcases and/or in the Superelement Subcases.

NOLINEAR: Used to select a family of NOLINi functions. May not be changed between solution subcases.

NLOAD: Required for NOLINi output request.

Input Bulk Data associated with the transient loads in SOL 99 (or 129) are briefly described below:

TLOAD1, TLOAD2: Required data defining the coupling between Load factors (DAREA Sets) and functions of time (TABLED, DELAY, or coefficients). Specifies the time functions, $F_i(t)$, for dynamic load i . TLOAD1 defines the load history in TABLED format while TLOAD2 defines the load history by an analytic expression. Note that each TLOADi input must have a unique identification number.

DAREA: Optional Bulk Data defining simple load scale factors, A_{ij} , for each degree of freedom, u_j requested in set i . The DAREA factors may also be defined with static load

data with the LSEQ Bulk Data. The name was chosen by a dynamicist who viewed the problem in terms of time-dependent pressures and location-dependent “areas”.

LSEQ: Generates transient load history for static loads and controls the assembly of static load vectors to be used in dynamic analysis. Each set is equivalent to a Subcase in Case Control. It will cause a static load vector A_i to be generated and will label it with the DAREA identification. This is useful for modeling problems with distributed transient loading such as GRAV loads or pressurized areas.

DLOAD: Combines different TLOADi functions into a single set and scales each by a factor, C_{ki} . Performs the same function as the static LOAD Bulk Data input. Unlike statics, this is the only method to combine loads in dynamic analysis.

TABLEDi, i=1,2,3,4 : Used with TLOAD1 input to specify general time varying tabular functions. Since extrapolation is used beyond the range of data the user is cautioned to be careful with the end points.

DELAY: Specifies the delay time for applying the forcing function defined in TLOADi to each GRID point. The Bulk Data DELAY is associated with the degree of freedom and not the forcing function in each TLOADi. This option is useful for defining loads that travel across a structure such as an oblique wave along a dam or a bump in the road exciting the wheels of a traveling vehicle. This saves the analyst the effort of defining multiple TLOADi inputs for simple time lags.

NOLINi, i=1,2,3,4 : Define specific nonlinear scalar loads as functions of velocity and/or displacement. Basic analog nonlinear transfer functions are defined. However, these functions do not generate tangent matrix terms and the Newton iteration method may experience difficulties in convergence. Line search and BFGS solution methods are recommended for these models.

NOLIN1 : Nonlinear transient load as a tabular function

NOLIN2 : Nonlinear transient load as products of two variables

NOLIN3 : Nonlinear transient load as a positive variable raised to a power

NOLIN4 : Nonlinear transient load as a negative variable raised to a power

A diagram of input data linkages is shown in Fig. 9.4.1. The general form of the transient load, P_{kj} , for each degree of freedom, u_j , is:

$$P_{kj} = C_{ki} * A_j * F_i(t)$$

where the factors are defined by various Bulk Data inputs as shown in Fig. 9.4.2.

Case Control Section

Bulk Data Section

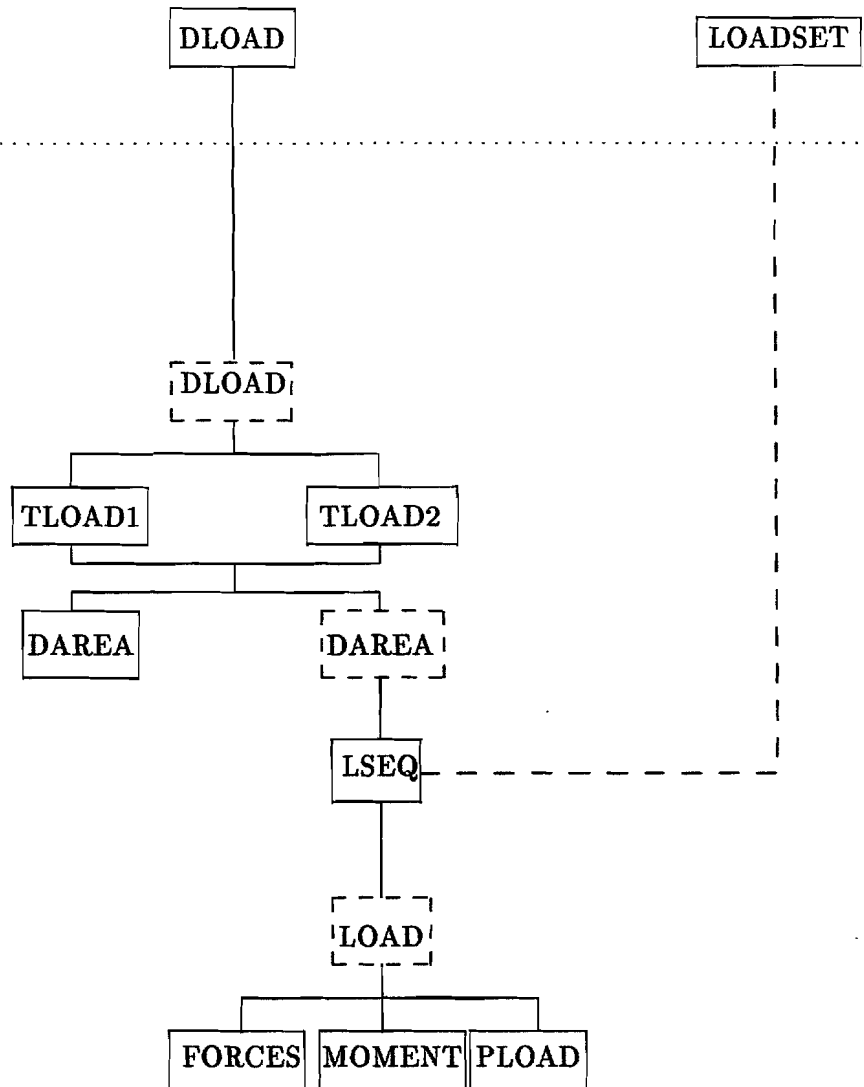
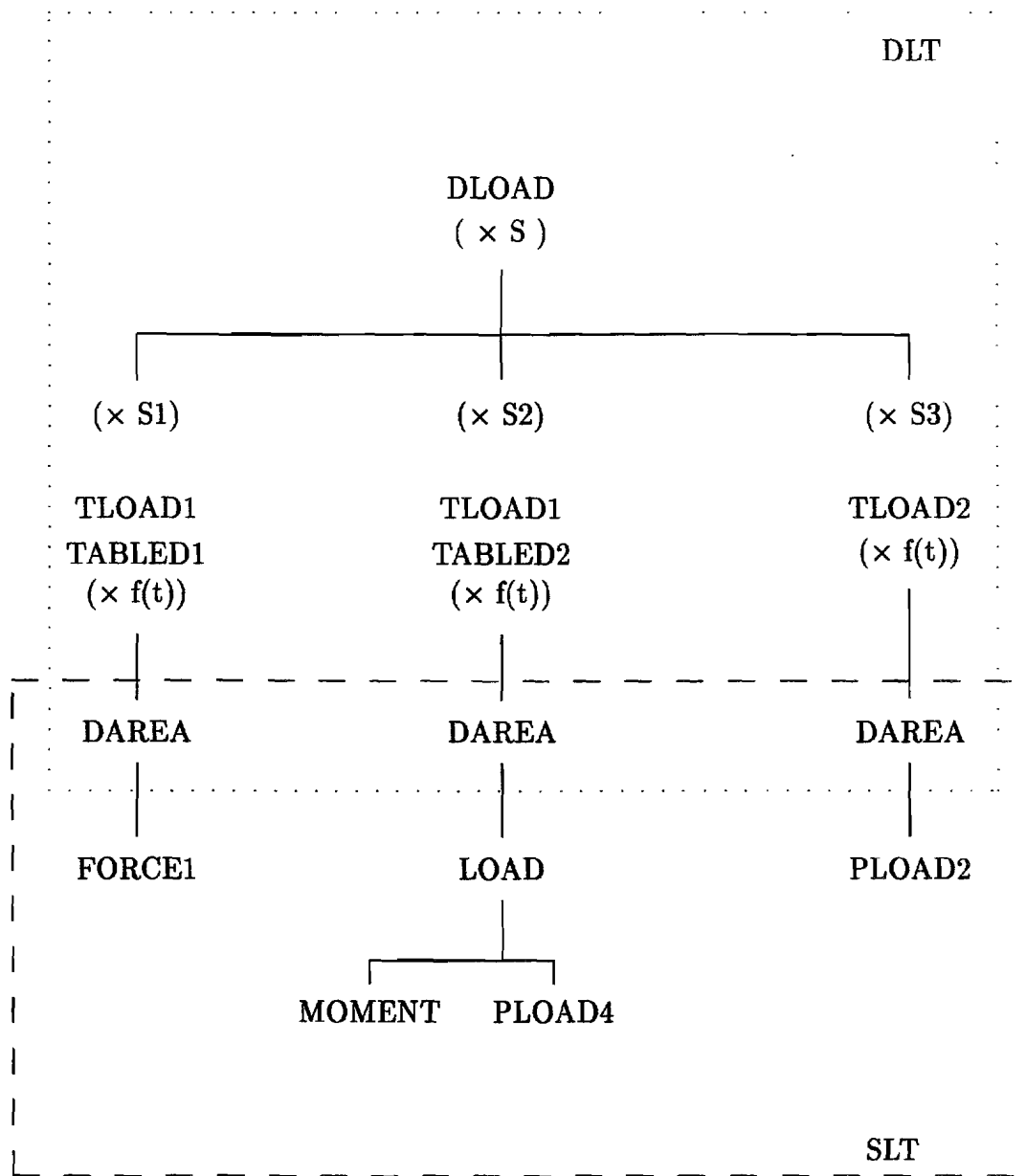


Figure 9.4.1 Dynamic Loads for Transient Analysis



Note : Each load is scaled by $[S \times S_i \times f(t)]$ at each time step in dynamic loads.

Figure 9.4.2 Example of Combining Loads

9.4.2 Enforced Motion

Excitation of the base of a structure with a time-dependent motion (displacement, velocity, or acceleration) is provided by the software indirectly via the "large mass method". In effect the user places a large mass element on the degrees of freedom representing the base and then provides a large force proportional to the desired accelerations. If a very large mass (M_b) is connected to a degree of freedom on an unconstrained structure, the load of magnitude P_b will produce an approximate base acceleration:

$$\ddot{u}_b \simeq \frac{P_b}{M_b}.$$

The accuracy of this approximation increases as M_b is made larger in comparison to the mass of the structure. However, numerical difficulties arise if M_b is too large. It is recommended that the value of M_b be approximately 1000 times the mass of the structure. The factor 1000 is a safe limit that will produce three digits of numerical accuracy.

When the large mass method is used, the function \ddot{u}_b is normally specified as input loads, and the value of M_b is input on DAREA or DLOAD Bulk Data entry, whichever is more convenient. CMASSi or CONM2 entry should be used to input the large mass. CMASSi entries allow different masses for different directions. The TLOADi inputs allow the specification of displacements and velocities, which in turn are converted to acceleration by the program to compute the inertia force as depicted in Fig. 9.4.3.

A large stiffness method could be used in lieu of the large mass approach. In this case, the applied force would be computed by

$$P_b = K_b u_b$$

where K_b is the stiff stiffness and u_b is the enforced displacement. The large stiffness method has the advantage in case of enforced displacement, because it avoids the roundoff error of numerical differentiation which occurs while computing accelerations from displacements in the large mass approach.

For Nonlinear problems, the "large mass or large stiffness approach" requires special caution in choosing convergence error tolerances. The actual errors will be normalized by terms proportional to the large base forces. It may be necessary to reduce the error tolerances on the TSTEPNL inputs depending on the size of the large base mass.

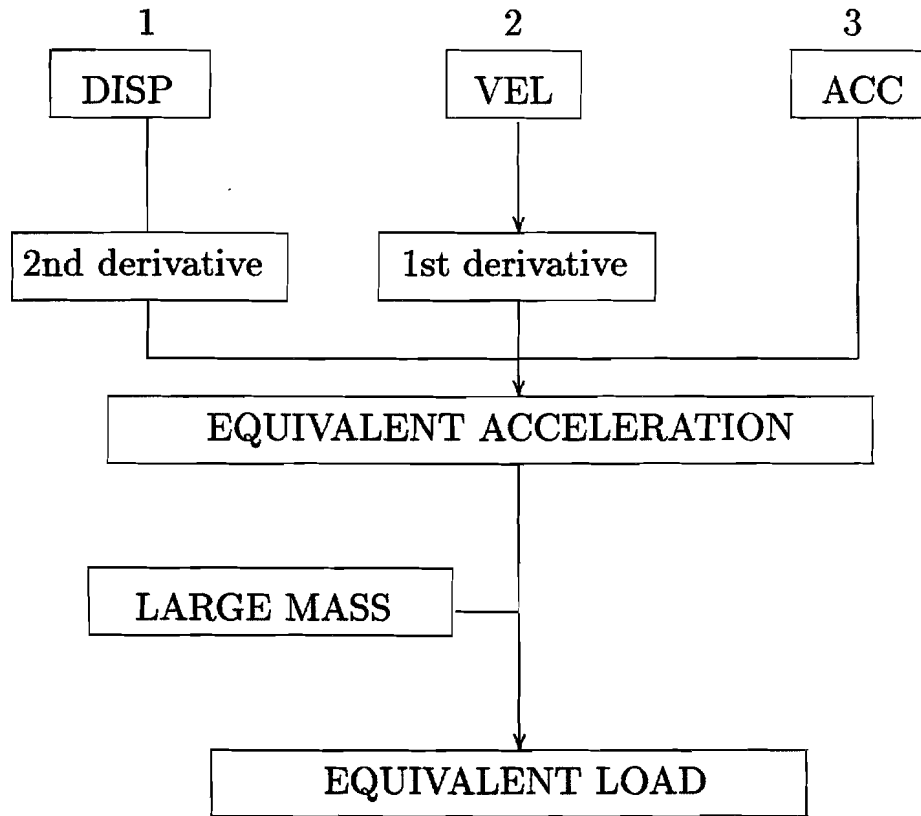


Figure 9.4.3 Options in TLOADi (Types 1, 2 and 3)

9.4.3 Initial Conditions

Initial velocities or displacements can be imposed by using the Bulk Data TIC which are selected by a Case Control command IC. Alternatively, three methods are available for providing a set of initial displacements and/or velocities for Nonlinear Transient Analysis:

1. TIC Bulk Data are used to specify point-by-point initial displacements and velocities. As in linear analysis, the system also calculates internal equivalent loads to provide equilibrium loads at the start. If u_0 and v_0 are the initial displacement and velocity vectors, respectively, the equivalent loads are:

$$\{P_0\} = [K]\{u_0\} + [C]\{v_0\}$$

where $[K]$ and $[C]$ are the linear stiffness and damping matrices. Note that if the initial state is nonlinear, the initial loads on the structure will not be in equilibrium. This will

require iterations and may cause difficulty for the convergence. To be safe, use the TIC data only for initial velocities and on linear sections of the structure.

2. Restarts from static analysis (SOL 66 or SOL 99 with TSTATIC) are highly recommended for initially displaced and loaded structures. The initial static equilibrium will be preserved if the initially supplied transient loads are identical to the static loads. The static solution (in SOL 66 or 106) may use enforced displacements (SPCD) and different boundaries to obtain the desired initial shape.
3. An Initial “dummy” Subcase may be used to generate complicated initial conditions without a restart. An extra subcase is added in front of the normal transient subcases with loading functions designed to produce the desired initial condition by matching the momentum. For instance, an initial velocity may be generated by a large impulsive load applied for 5-10 small time steps. An initial deformation may be produced by a subcase with large time steps (larger than the lowest mode’s period) which eliminate the transient behavior and produces a static solution if it converges. This method requires that the other loads be shifted in time and will also cause a shift in the output response.

The user is also cautioned against supplying large initial loads in the structure that may produce initial nonlinearities. The job may not be able to complete the first time step. Solutions should be obtained as follows:

- Restart from a Nonlinear static solution using the initial load.
- Start with a smaller load and increase it over a small time interval.
- Reduce the size of the time steps by a few orders of magnitude for the first subcase.

9.5 Diagnostic Outputs

Diagnostic output is available in SOL 99 (or 129) if DIAG 50 or 51 is specified in the Executive Control section. When DIAG 50 is requested, only the data marked with a plus (+) are printed. When DIAG 51 is selected, all the data are printed at each iteration, allowing the user to follow the nonlinear iteration calculations step by step. Because of the volume of output produced, the use of DIAG 51 is recommended only for small test problems. DIAG 51 is intended for debugging purposes, and users in general are cautioned against its use.

9.5.1 NLTRD Module for AUTO or TSTEP Method

For each entry into NLTRD:

- + Subcase status data
- + TSTEPNL data
 - Core statistics (ICORE, etc.)
 - Problem statistics (g-size, etc.)
 - File control block
 - Input file status

For each iteration:

- Initial energy for line search
- Nonlinear internal force: F_g
- Sum of nonlinear forces including follower forces: F_d
- NOLINi vector: N_d
- Displacement vector: u_d
- Total internal force: F_d
- Loading error vector: R_d
- + Iteration summary (Convergence factors, line search data, etc.)

For each quasi-Newton vector set:

- Condition number: λ^2
- Quasi-Newton vector: δ
- Quasi-Newton vector: γ
- Energy error: $z = \frac{1}{\delta_j^T \gamma_j}$

For each line search:

- Previous line search factor: α_k
- Previous error: E_k
- New line search factor: α_{k+1}

9.5.2 NLTRD2 Module for ADAPT Method

For each entry into NLTRD2:

- + Subcase status data
- + TSTEPNL data
 - Core statistics (ICORE, etc.)
 - Problem statistics (g-size, etc.)
 - File control block
 - Input file status

For each time step:

- NOLINi vector: N_d
- External load vector: P_d
- Load vector including follower forces and NOLINs: P_d
- Constant portion of residual vector: R'_d
- Total internal force: F_d
- Initial loading error vector: R_d

For each iteration:

- Initial energy for line search
- Nonlinear internal force: F_g
- Displacement vector: u_d
- Nonlinear internal force: F_d
- Total internal force: F_d
- NOLINi vector: N_d
- Load vector including follower forces and NOLINs: P_d
- Loading error vector: R_d
- + Iteration summary (Convergence factors, line search data, etc.)

For each quasi-Newton vector set:

- Condition number: λ^2
- Quasi-Newton vector: δ
- Quasi-Newton vector: γ
- Energy error: $z = \frac{1}{\delta_j^T \gamma_j}$

For each line search:

- Previous line search factor: α_k
- Previous error: E_k
- New line search factor: α_{k+1}

For each converged time step:

Velocity vector: v_d

For each time step adjustment:

Magnitude of the old velocity vector: \dot{u}_n

Magnitude of the new velocity vector: \dot{u}_{n+1}

Generalized stiffness: $\text{DENOM1} = \Delta u_n^* K \Delta u_n$

Generalized mass: $\text{DENOM2} = \Delta u_n^* M \Delta u_n$

Work = $\Delta u_n^* \Delta F_n$

Square of dominant frequency: ω_n^2

Stiffness ratio

Number of steps for the period of dominant frequency: MSTEP

Controlling ratio for time step adjustment: r

Chapter 10

SPECIAL APPLICATIONS

10.1 NONLINEAR BUCKLING ANALYSIS

10.1.1 Introduction

A simple approach to nonlinear buckling analysis was introduced in SOL 66 (or 106) by using a quadratic extrapolation [10.1]. A similar approach to the limit point calculations was published by Fujikake [10.2]. The distinction is that this software uses two converged solutions in the vicinity of the buckling point to form a differential stiffness $[\Delta K]$, whereas Fujikake uses a converged point and an estimated data point beyond the buckling point. To find an instability point within a small range of nonlinear domain, two methods of idealization can be contemplated:

1. The tangent stiffness matrix is proportional to the external loads, which implies that the critical load may be linearly interpolated, i.e., $P_{cr} = P_n + \lambda \Delta P$.
2. The tangent stiffness matrix is proportional to the displacement increments, which implies that the critical displacements may be obtained by extrapolating from the current state, i.e., $U_{cr} = U_n + \lambda \Delta U$.

It appears that Fujikake has chosen the first approach. However, he did not show how he obtains the critical displacements. MSC/NASTRAN has chosen the second approach. Since the tangent matrix is assumed to change linearly, the internal loads are quadratic function of displacements. The distinction between the two approaches disappears in the linear bifurcation problem.

Restart is required for a nonlinear buckling analysis after the non-positive definite stiffness matrix is detected. Since the modified Newton's method is employed with quasi-Newton updates in the general purpose program, tangential stiffness matrices in the two successive increments are not immediately available in general. Details of the method are given below.

10.1.2 Formulation of Nonlinear Buckling Analysis

It can be shown that the eigenvalue problem for a nonlinear buckling analysis may be approximated by

$$[K_n + \lambda \Delta K] \{\phi\} = \{0\} \quad (10.1.1)$$

with

$$\Delta K = K_n - K_{n-1}$$

where K_n and K_{n-1} are the stiffness matrices evaluated at the known solution points in the vicinity of the instability.

The critical displacements upon instability may be estimated as

$$\{U_{cr}\} = \{U_n\} + \lambda \{\Delta U\} \quad (10.1.2)$$

with

$$\{\Delta U\} = \{U_n\} - \{U_{n-1}\}$$

Based on the virtual work principle,

$$\{\Delta U\}^T \{P_{cr}\} = \{\Delta U\}^T \{F_{cr}\} \quad (10.1.3)$$

where

$$F_{cr} = F_n + \int_{U_n}^{U_{cr}} K du = F_n + \int_0^\lambda K(\lambda) \Delta U d\lambda \simeq F_n + \lambda [K_n + \frac{1}{2} \lambda \Delta K] \Delta U.$$

The critical buckling load may be estimated as

$$\{P_{cr}\} = \{P_n\} + \alpha \{\Delta P\} \quad (10.1.4)$$

with

$$\{\Delta P\} = \{P_n\} - \{P_{n-1}\}$$

where

$$\alpha = \frac{\lambda \{\Delta U\}^T [K_n + \frac{1}{2} \lambda \Delta K] \{\Delta U\}}{\{\Delta U\}^T \{\Delta P\}} \quad (10.1.5)$$

This procedure was first implemented with DMAP ALTER in Version 64 of the software by a simple DMAP (Direct Matrix Abstraction Program) language. This capability has been available with a control parameter BUCKLE in SOL 66 since Version 65.

10.1.3 Analysis Procedure

Nonlinear buckling analysis capability is provided in SOL 66 (or 106) and activated by a user-specified parameter BUCKLE with a value of +1. The operational steps are

1. Run SOL 66 for static analysis and create a database for restarts. It is recommended to continue the nonlinear static analysis with small steps until a negative determinant of the stiffness matrix is encountered.
2. Make a restart run as follows:
 - (a) Select PARAM entries SUBID, LOADINC and LOOPID for restart.
 - (b) Provide two small loading steps (below buckling). This may be provided by adding a new subcase, skipping the rest of the increments of the subcase in which the stiffness matrix becomes singular.
 - (c) Use "KSTEP=1" in the NLPARM entry for two load steps to be solved so that the stiffness matrix is updated for each solution.
 - (d) Include EIGB Bulk Data entry with a METHOD command in the Case Control section for the eigenvalue analysis. SINV method is recommended for the eigenvalue extraction.
 - (e) Provide mode shape PLOT data if desired.
 - (f) Include PARAM, BUCKLE, 1.

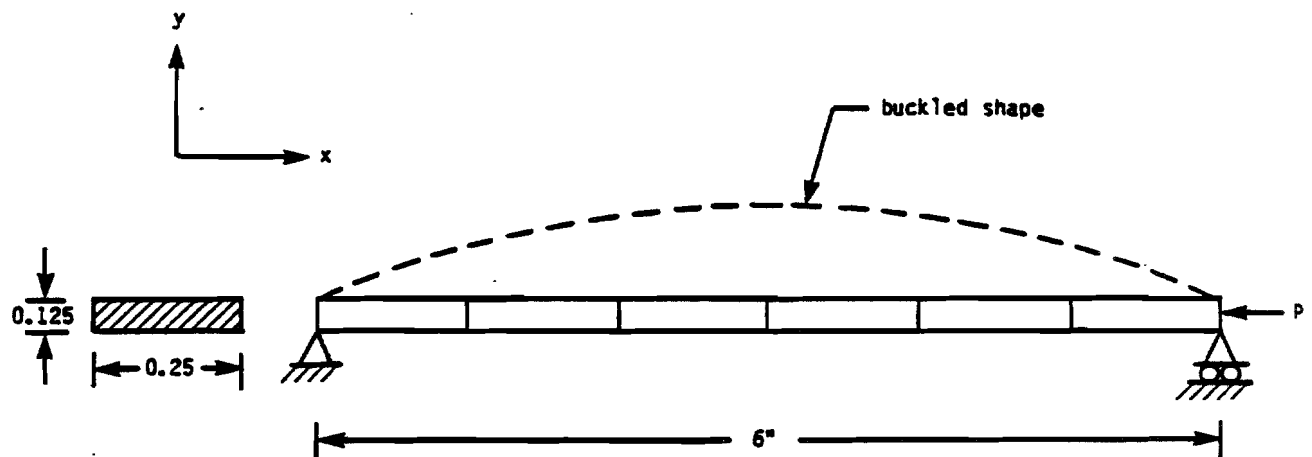
10.1.4 Verification Problem: Euler Column Buckling

This method should be applicable to linear problems because linear buckling is a special case of nonlinear buckling. For verification, an Euler buckling problem of a pin-ended column is illustrated. The column with dimensions shown in Fig. 10.1.1 is modeled with 6 CBEAM elements and subjected to a compressive axial load. The input data listing is shown in Table 10.1.1.

The buckling analysis was performed by two independent methods as follows:

1. Using SOL 65 (linear buckling analysis), restarted from SOL 61 (linear static analysis) database.
2. Using SOL 66 with PARAM, BUCKLE, restarted from SOL 66 database.

Both procedures produced identical solutions with the critical buckling load (P_{cr}) of 120.38 lbs.



$$P_{cr} = \frac{n^2 \pi^2 E I}{L^2}$$

where

- $n = 1$
- $E = 10.8 \times 10^6$ psi
- $I = 4.069 \times 10^{-5}$ in⁴
- $L = 6$ in
- $P_{cr} = 120.48$ lbs

Figure 10.1.1 Euler Buckling for Verification

Table 10.1.1 Verification Problem Input Data

```

ID BBUCKL,V65    $ CJS, 7-MAY-85, SHL 5/2/84
SOL 66           $ NONLINEAR STATIC ANALYSIS
DIAG 8,50        $ PRINT MATRIX TRAILERS AND ITERATIONS
TIME 5           $ CPU MINUTES
CEND
TITLE = EULER BUCKLING OF BEAM-COLUMN
SUBTITLE = AXIAL FORCE, PIN-ENDED BOUNDARY
ECHO = UNSORT
SEALL = ALL
  DISP = ALL
  OLOAD = ALL
  SPCF = ALL
  STRESS = ALL
  METHOD = 30
$-----*
$PARAM BUCKLE 1 *
$PARAM SUBID 2 * CHANGES REQUIRED
$PARAM LOADING 1 * FOR BUCKLING
$PARAM LOOPID 1 *
$-----*
SUBCASE 1
  LOAD = 10
  NLPARM = 10
SUBCASE 2
  LOAD = 20
  NLPARM = 20
OUTPUT(PLOT)
  CSCALE = 1.3
  PLOTTER NAST
  SET 1 = ALL EXCEPT PLOTTEL
  MAXI DEFO .05
  AXES Z, X, Y
  VIEW 0.,0.,0.
  PTITLE = SIDE VIEW
  FIND SCALE ORIGIN 1 SET 1
  PLOT SET 1 ORIGIN 1 LABEL BOTH
  PLOT STATIC 0 MAXI DEFO .05 SET 1 ORIGIN 1
  PLOT MODAL 0 SET 1 ORIGIN 1
BEGIN BULK
$ PARAMETERS
PARAM LGDISP 1
$ SOLUTION CONTROL
NLPARM 10 1 AUTO UPW
NLPARM 20 4 AUTO 1 PW YES
$ EIGENVALUE ANALYSIS
EIGB 30 INV 0. 3. 20 2 2 +EIGB
+EIGB MAX

```

\$ MATERIAL AND GEOMETRIC PROPERTIES

MAT1	1	10.8+6	.3						+MAT1
+MAT1	7.8+4	7.8+4	4.5+4						
MATS1	1		PLASTIC	1.225+6	1	2	7.8+4		
PBEAM	3	1	3.125-2	4.069-5	1.628-4				+PB1
+PB1	-.0625	-.125	-.0625	.125	.0625	.125	.0625	-.125	

\$ LOADING

FORCE	10	16	100.	-1.	0.	0.
FORCE	20	16	140.	-1.	0.	0.

\$ GEOMETRY

GRID	1000	0.	1.	0.	123456
GRDSET					345
GRID	10	0.	0.	0.	12345
GRID	11	1.	0.	0.	
GRID	12	2.	0.	0.	
GRID	13	3.	0.	0.	
GRID	14	4.	0.	0.	
GRID	15	5.	0.	0.	
GRID	16	6.	0.	0.	2345

\$ CONNECTIVITY

CBEAM	1	3	10	11	1000
CBEAM	2	3	11	12	1000
CBEAM	3	3	12	13	1000
CBEAM	4	3	13	14	1000
CBEAM	5	3	14	15	1000
CBEAM	6	3	15	16	1000

\$

ENDDATA

10.1.5 Example of Nonlinear Buckling

An elastic-plastic buckling analysis of a thin spherical shell is presented [10.1]. With a model consisting of 16 shell elements (10 QUAD4s and one TRIA3), the nonlinear static analysis determined that buckling would occur between 3500 psi and 3600 psi, where a negative determinant of $[K]$ was first detected. The results are summarized in Fig. 10.1.2 and Table 10.1.2.

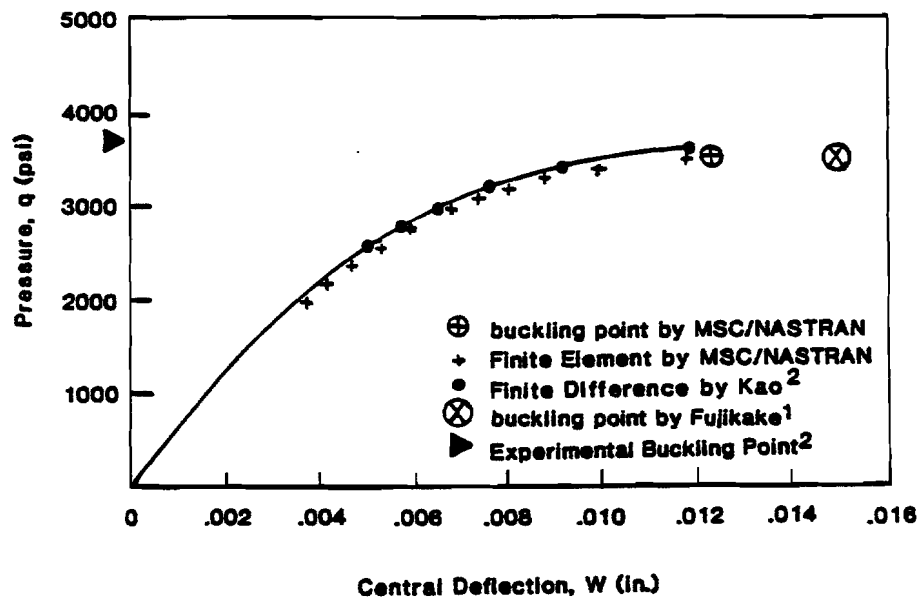


Figure 10.1.2 Elastic-Plastic Buckling of a Clamped Spherical Cap with Flat Spot (Load vs. Central Deflection)

Table 10.1.2 Output Summary

Subcase		Load Factor	Load (psi)	No. of Iter.	-u100(in)	max $\bar{\sigma}_{10}$ (psi)
1	C o S	0.50	1000	5	1.739-3	3.532+4
		1.0	2000	7	3.719-3	7.555+4
2	l t d a r t	0.20	2200	7	4.154-3	7.894+4****
		0.40	2400	8	4.640-3	8.061+4
		0.60	2600	10*	5.217-3	8.279+4
		0.80	2800	10*	5.927-3	8.564+4
		1.0	3000	20	6.846-3	8.944+4
	R	0.125	3100	7-K	7.418-3	9.179+4
3	e s t a r t	0.25	3200	7	8.091-3	9.460+4
		0.375	3300	9	8.918-3	T9.795+4
		0.500	3400	17	10.032-3	1.024+5
		0.625	3500	8-K	11.942-3	1.098+5
		0.75	3600	2***	—	—

* Max limit is reached.

** In this stage of the cold start run, the number of iterations reached the prescribed maximum limit of 10 with $e_p = 3.3437E-2$. The iteration limit was changed to 20 and the convergence tolerances were relaxed before the job restarted.

*** A negative term on factor diagonal in [K] detected. The execution was terminated after seven iterations due to two consecutive diverging solutions with stiffness updating.

**** Plastic deformation initiated. Reference 1 reports that the yielding initiated at 2600 psi.

-K [K] update during iteration.

As shown in Table 10.1.3, a restart run was made with PARAM, BUCKLE starting from the solution at $P=3300$ psi. Solutions at $P=3400$ psi and 3500 psi were repeated and a buckling analysis was performed using the differential stiffness between those two loading steps. The results of this eigenvalue analysis are

$$\lambda = 0.775 \quad \text{and} \quad \alpha = 0.466$$

from which the critical buckling load is calculated by

$$P_{cr} = P_n + \alpha \Delta P = 3500 + 0.466 \times 100 = 3546.6 \text{ psi}$$

and the maximum deflection at the crown by

$$U_{cr} = U_n + \lambda \Delta U = 0.01224 \text{ in.}$$

The loading history to buckling point was obtained as shown in Fig. 10.1.2. Notice that Fujikake [10.2] overestimated the maximum deflection at buckling. Fig. 10.1.3 shows the deformed shape on buckling.

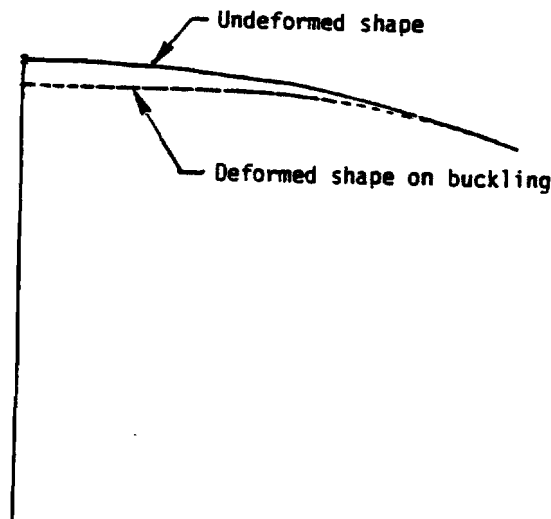


Figure 10.1.3 Deformed Shape on Buckling

This structure was analyzed again with a solid-element model. The buckling point is somewhat underestimated with solid model, resulting in

$$P_{cr} = 3520.8 \text{ psi} \quad \text{and} \quad U_{cr} = 0.0122 \text{ in.}$$

at the crown. Fig. 10.1.4 shows the deformed shape on buckling. It seems that the deformed shape at buckling is as significant and useful as the buckling mode shape in the large-displacement problem. Note that the mode shapes for bifurcation buckling may be quite different.

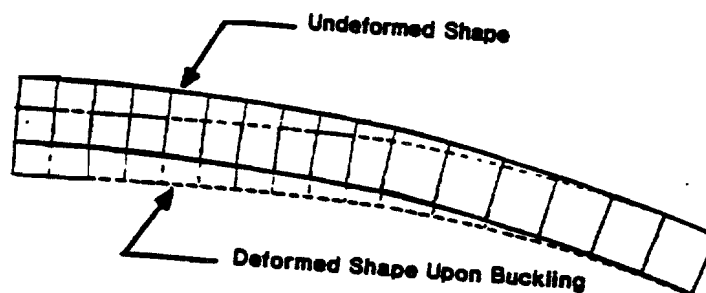


Figure 10.1.4 Deformed Shape Upon Buckling with a Solid Model

A few more analyses were attempted to predict buckling point from the solutions at the earlier stage of loading. When the solution points are farther removed from the actual buckling point, the prediction becomes less reliable. Thus the user is cautioned to be suspicious when the α is greater than unity, in which the predicted buckling point is not close to the instability point detected by the static solution. In such cases, the solution may be easily validated by repeating the process with new trial loads closer to the estimated buckling load.

Table 10.1.3 Example Problem Input Data

```

ID SBUCKL,V65      $ SHL 5/7/85 N6609
SOL 66             $ NONLINEAR STATIC ANALYSIS
DIAG 8,50          $ PRINT MATRIX TRAILERS AND ITERATIONS
TIME 15            $ CPU MINUTES
CEND
TITLE = ELASTIC-PLASTIC BUCKLING OF IMPERFECT SPHERICAL SHELL A1185G
SUBTITLE = HYDROSTATIC PRESSURE APPLIED, PERIPHERY CLAMPED
ECHO = UNSORT
SEALL = ALL
  DISP  = ALL
  OLOAD = ALL
  SPCF  = ALL
  STRESS = ALL
  SPC   = 10
  METHOD = 30
$-----*
$PARAM  BUCKLE  1      *
$PARAM  SUBID   4      *  CHANGES REQUIRED
$PARAM  LOADING 1      *  FOR BUCKLING
$PARAM  LOOPID 10     *
$-----*
SUBCASE 1
  LOAD   = 10
  NLPARM = 10
SUBCASE 2
  LOAD   = 20
  NLPARM = 20
SUBCASE 3
  LOAD   = 30
  NLPARM = 30
SUBCASE 4      $ ADDED FOR BUCKLING ANALYSIS
  LOAD   = 40
  NLPARM = 40
OUTPUT(PLOT)
  CSCALE = 1.3
  PLOTTER NAST
  SET 1 = ALL EXCEPT PLOTTEL
  SET 2 = 1000
  SET 3 = ALL
  VIEW 30.,20.,0.
  MAXI DEFO .05
  FIND SCALE ORIGIN 1 SET 3
  PTITLE = ISOMETRIC VIEW
PLOT  SET 1 ORIGIN 1 SET 2 SYMBOL 3
  VIEW 90.,0.,0.
  FIND SCALE ORIGIN 1 SET 1
  PTITLE = SIDE VIEW

```



```

PLOT   SET 1 ORIGIN 1 SET 2 SYMBOL 3
PLOT   STATIC 0 MAXI DEFO .05 SET 1 ORIGIN 1 SET 2 SYMBOL 3
PLOT   MODAL 0                SET 1 ORIGIN 1 SET 2 SYMBOL 3
BEGIN BULK
$$ PARAMETERS
PARAM  SMALLDB 1
PARAM  LGDISP 1
$$ SOLUTION CONTROL
NLPARM 10      2      AUTO      PW
NLPARM 20      5      AUTO      PW      NO
NLPARM 30      8      AUTO      YES
NLPARM 40      2      AUTO      1      YES
$$ EIGENVALUE ANALYSIS
EIGB   30      SINV   0.      2.      20      2      2      +EIGB
+EIGB  MAX
$$ PROPERTIES
MAT1   1      10.8+6      .3
MATS1  1      PLASTIC 1.225+6 1      2      7.8+4
PSHELL 2      1      .0251 1
$$ BOUNDARY AND LOADING
SPC1   10      123456 131      132
PLOAD2 10      -2000. 10      THRU      25
PLOAD2 20      -3000. 10      THRU      25
PLOAD2 30      -3800. 10      THRU      25
PLOAD2 40      -3500. 10      THRU      25
$$ GEOMETRY AND CONNECTIVITY
CORD2S 100      0.      0.      0.      0.      0.      1.      +C2S1
+C2S1  1.      0.      1.
CORD2S 200      0.      0.      -.32908 0.      0.      1.      +C2S2
+C2S2  1.      0.      1.
GRDSET                                100      345
GRID   1000      0.      0.      0.      123456
GRID   100      200      1.1506 0.      0.      0      12456
GRID   101      200      1.1506 .715      -5.
GRID   102      200      1.1506 .715      5.
GRID   103      200      1.1506 1.43      -5.
GRID   104      200      1.1506 1.43      5.
GRID   105      200      1.1506 2.145      -5.
GRID   106      200      1.1506 2.145      5.
GRID   107      200      1.1506 2.86      -5.
GRID   108      200      1.1506 2.86      5.
GRID   109      200      1.1506 3.575      -5.
GRID   110      200      1.1506 3.575      5.
GRID   111      200      1.1506 4.29      -5.
GRID   112      200      1.1506 4.29      5.
GRID   113      200      1.1506 5.005      -5.
GRID   114      200      1.1506 5.005      5.
GRID   115      200      1.1506 5.72      -5.
GRID   116      200      1.1506 5.72      5.

```

GRID	117	200	1.1506	6.435	-5.	
GRID	118	200	1.1506	6.435	5.	
GRID	119	100	.8251	10.	-5.	
GRID	120	100	.8251	10.	5.	
GRID	121	100	.8251	11.48	-5.	
GRID	122	100	.8251	11.48	5.	
GRID	123	100	.8251	12.96	-5.	
GRID	124	100	.8251	12.96	5.	
GRID	125	100	.8251	14.44	-5.	
GRID	126	100	.8251	14.44	5.	
GRID	127	100	.8251	15.92	-5.	
GRID	128	100	.8251	15.92	5.	
GRID	129	100	.8251	17.40	-5.	
GRID	130	100	.8251	17.40	5.	
GRID	131	100	.8251	18.8806	-5.	
GRID	132	100	.8251	18.8806	5.	
PLOTEL	1000	1000	100			
CTRIA3	10	2	100	101	102	
CQUAD4	11	2	101	103	104	102
CQUAD4	12	2	103	105	106	104
CQUAD4	13	2	105	107	108	106
CQUAD4	14	2	107	109	110	108
CQUAD4	15	2	109	111	112	110
CQUAD4	16	2	111	113	114	112
CQUAD4	17	2	113	115	116	114
CQUAD4	18	2	115	117	118	116
CQUAD4	19	2	117	119	120	118
CQUAD4	20	2	119	121	122	120
CQUAD4	21	2	121	123	124	122
CQUAD4	22	2	123	125	126	124
CQUAD4	23	2	125	127	128	126
CQUAD4	24	2	127	129	130	128
CQUAD4	25	2	129	131	132	130
ENDDATA						

10.2 NONLINEAR MODAL ANALYSIS

The modal analysis DMAP, SOL 63, has been modified for nonlinear modal analysis with the nonlinear stiffness matrices obtained from the SOL 66 database. The rigid format DMAP alter for SOL 63, RF63D89, is shown in Table 10.2.1. This rigid format alter can be used in Version 66. The DMAP alter for Version 65 is listed in the Application Note, "Nonlinear Normal Mode Analysis," dated January 1986. This is an alternate method to the SOL 64 - SOL 63 sequence for nonlinear normal mode analysis, demonstrated in the Application Note of May 1981. The main advantage of using SOL 66 instead of SOL 64 is that both geometric and material nonlinearities can be included. In Version 67, this capability is built in SOL 106 with PARAM, NMLOOP.

This capability is demonstrated by four problems: (a) vibration of a linear material beam, (b) vibration of a nonlinear material beam, (c) vibration of a plate, and (d) vibration of a nonlinear stepped beam.

10.2.1 Analysis Procedure

In nonlinear static analysis using SOL 106, normal modes can be computed with the updated stiffness if PARAM, NMLOOP, n is specified where n is the desired LOOPID. This option also requires the presence of the METHOD command in the Case Control section and EIGR or EIGRL entry in the Bulk Data section. If the restart procedure (from SOL 66 to SOL 63) is used with RF63D89, data blocks (ESTNL and UGV) are fetched for the nonlinear stiffness from the SOL 66 run. LOOPID is used as a parameter to choose the step at which the modal analysis is desired. If LOOPID is not specified in the SOL 63 run or ESTNL is not found in the data base, the linear normal modes will be computed.

The parameter LGDISP needs to be set to 1 in both SOLs 66 and 63 to obtain the differential stiffness matrix to account for the prestressed condition. Since the superelement analysis does not have the nonlinear capability, the nonlinear matrix generations for superelements will be skipped. However, SEALL = n for the residual structure or SEALL = ALL should be used to initiate the nonlinear stiffness matrix generation.

The stiffness matrix must be updated after the load step where the normal modes are desired. In order to force the update, the value of KSTEP in the NLPARM entry should be set to 1 with the AUTO, SEMI or ITER method.

The structure of the matrices used in SOL 66 and SOL 63 should be consistent. For example, when dynamic reduction is to be applied in SOL 63, the same scalar points should be defined in SOL 66, even though these points are not actually used. See the examples and the comments in Table 10.2.1 for other input items required for SOL 63.

10.2.2 Vibration of a Linear Material Beam

Fig. 10.2.1 shows a simply supported beam under axial force that is used to test this capability. The total length of the beam is 1000 mm and the width and height of its cross section is 100 mm and 50 mm, respectively. The Young's modulus is 2.0684×10^5 MPa and the mass density per unit length is 3.9167×10^{-5} Nsec²/mm² (ton/mm). An axial force of 1.0×10^7 N is applied and the beam remains linear. The bending vibration frequencies will increase because of the axial force. The beam is modeled using five BEAM elements.

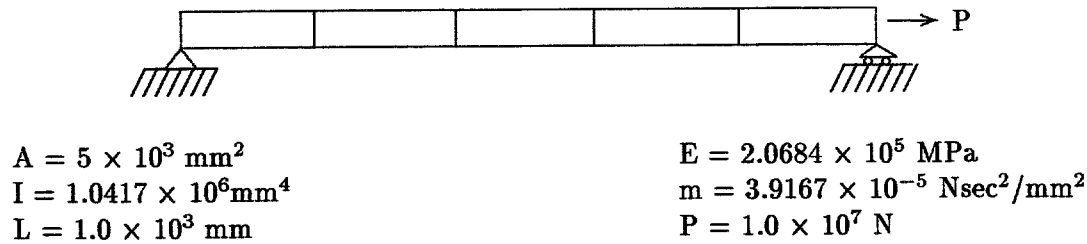


Figure 10.2.1 Simply Supported Beam.

To check the accuracy of the finite element beam model, first the theoretical natural frequencies of a simply supported beam without axial force are compared with those obtained from the software. The theoretical values are obtained by

$$\omega_n = (n\pi)^2 \sqrt{\frac{EI}{mL^4}} \quad (10.2.1)$$

The comparison is shown below:

Beam Vibration with $P = 0$.

Mode	ω (Theory)	ω (NASTRAN)	Mode
1	7.3208×10^2	7.3020×10^2	1st bending
2	2.9281×10^3	2.9043×10^3	2nd bending
3	6.5883×10^3	6.5079×10^3	3rd bending
4	— — —	8.0385×10^3	1st longitudinal
5	1.1712×10^4	1.1589×10^4	4th bending

The above table shows that the software values compare very well with the theoretical values. Therefore, the model is quite accurate.

The natural frequencies of the simply supported beam subjected to constant axial force are compared with the theoretical values obtained by

$$\omega_n = (n\pi)^2 \sqrt{\frac{EI}{mL^4} \left(1 + \frac{1}{(n\pi)^2} \frac{PL^2}{EI} \right)} \quad (10.2.2)$$

The comparison is shown below:

Linear Beam with $P = 1. \times 10^7$ N.

$P = 1. \times 10^7$ N. , $E = 2.0684 \times 10^5$ MPa.

Mode	ω (Theory)	ω (NASTRAN)	Mode
1	1.7481×10^3	1.7339×10^3	1st bending
2	4.3189×10^3	4.2828×10^3	2nd bending
3	— — —	8.0385×10^3	1st longitudinal
4	8.1292×10^3	8.0453×10^3	3rd bending
5	1.3323×10^4	1.3207×10^4	4th bending

The above tables show that the axial load affects the lower modes more than the higher modes.

10.2.3 Vibration of a Nonlinear Material Beam

This example calculates the natural frequencies of a simply supported beam (Fig. 10.2.1) subjected to an axial load of 1.0×10^7 N, and with the nonlinear elastic stress-strain relationship shown in Fig. 10.2.2. The input data for this model are shown in Tables 10.2.2 and 10.2.3.

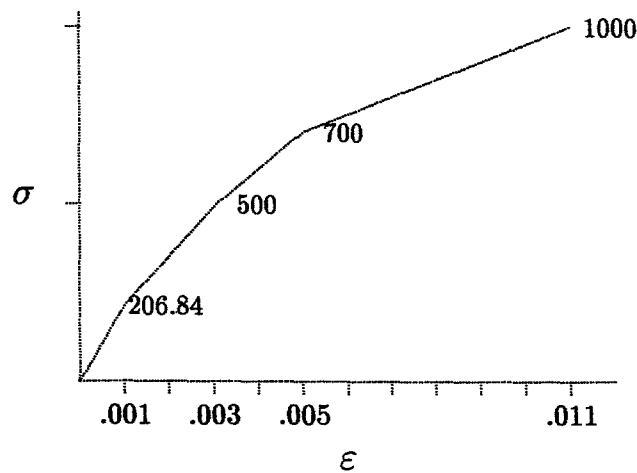


Figure 10.2.2 Nonlinear Stress-Strain Relationship for Simply Supported Beam

Analysis results are compared with the theoretical solution as tabulated below. The theoretical values are calculated by using the tangent modulus at the final solution ($\varepsilon = 0.031$).

Nonlinear Beam with $P = 1. \times 10^7$ N.

$P = 1. \times 10^7$ N. , $E_0 = 2.0684 \times 10^5$ MPa. , $E = 5. \times 10^4$ MPa.

Mode	ω (Theory)	ω (NASTRAN)	Mode
1	1.6277×10^3	1.6768×10^3	1st bending
2	3.4859×10^3	3.7978×10^3	2nd bending
3	— — —	6.0180×10^3	1st longitudinal
4	5.7593×10^3	6.5456×10^3	3rd bending
5	8.5717×10^3	9.9099×10^3	4th bending

The frequencies of the lower modes have increased while those of the higher modes decreased. This is due to the fact that the softening material effect is larger on the higher modes than the longitudinal tension.

10.2.4 Plate Vibration

This example demonstrates geometric nonlinear modal analysis for a rectangular plate. The plate has a thickness of 1.0 cm and the dimension is 2.0 m \times 4.0 m. The Young's modulus is 2.0×10^{11} N/m² and the mass density is 8.0×10^3 kg/m³. The plate is clamped on all sides and is subjected to a pressure loading of 2.0×10^5 Pa normal to its plane.

Due to symmetry, only one quarter of the plate is modeled using 50 QUAD4 elements as shown in Fig. 10.2.3. The clamped boundary condition is applied to sides BC and CD by constraining all six DOFs. The axisymmetric boundary condition is imposed by constraining components 2, 4, and 6 on side AB and components 1, 5, and 6 on side AD. The input data for this model are listed in Tables 10.2.4 and 10.2.5 for nonlinear static analysis and modal analysis, respectively.

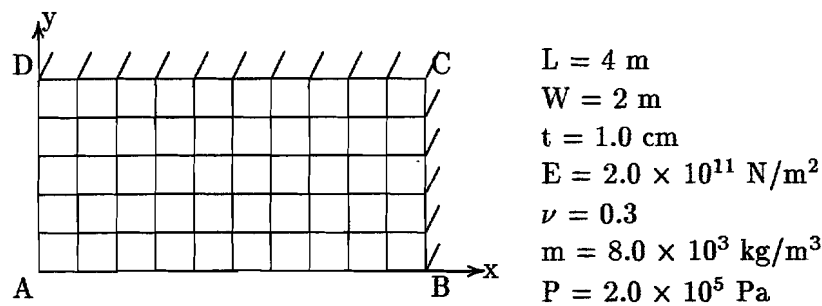


Figure 10.2.3 Clamped Plate (Quarter Only, Due to Symmetry)

At first, SOL 66 provides information about the nonlinear stiffness matrix. In the SOL 63 restart, dynamic reduction is used.

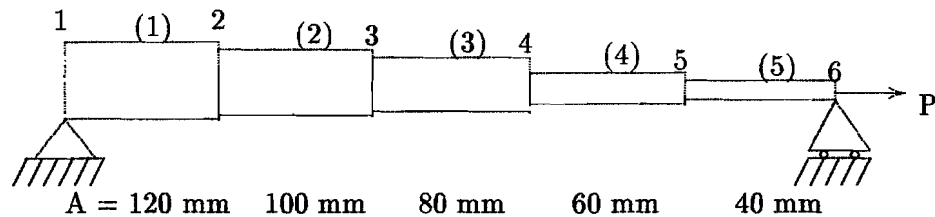
The results are tabulated along with frequencies of the linear system without the pressure load. The lower modes are affected more by the pressure load than the higher modes, which indicates the same tendency as in the beam problem.

Angular Frequencies of Plate.

Mode	ω (Linear)	ω (Nonlinear)	Mode
1	9.2595×10^1	4.7648×10^2	1st bending
2	1.6710×10^2	5.5690×10^2	2nd bending
3	3.2677×10^2	7.1118×10^2	3rd bending
4	4.6600×10^2	9.0174×10^2	4th bending
5	5.3634×10^2	9.4842×10^2	5th bending

10.2.5 Vibration of a Stepped Beam

This example demonstrates the geometric and material nonlinear modal analysis for a simply supported stepped beam, in Fig. 10.2.4. The material is nonlinear elastic with the properties shown in Fig. 10.2.2. The total length of the beam is 1000 mm and its cross sectional areas vary from 120 mm to 40 mm. The Young's modulus is $2.0684 \times 10^5 \text{ MPa}$ and the mass density per unit length is $3.9167 \times 10^{-5} \text{ Nsec}^2/\text{mm}^2$ (ton/mm). An axial force of $1.5 \times 10^6 \text{ N}$ is applied, for which elements 1 and 2 remain linear. The stepped beam is modeled using five BEAM elements. The input data for this model are listed in Tables 10.2.6 and 10.2.7.



$$\begin{aligned}
 L &= 1.0 \times 10^3 \text{ mm} \\
 E &= 2.0684 \times 10^5 \text{ MPa} \\
 m &= 3.9167 \times 10^{-5} \text{ Nsec}^2/\text{mm}^2 \\
 P &= 1.5 \times 10^6 \text{ N}
 \end{aligned}$$

Figure 10.2.4. Stepped Beam.

The bending vibration frequencies are tabulated below. The first column represents the linear system without the axial load. The second column shows the nonlinear system with the axial load of $1.5 \times 10^6 \text{ N}$.

Angular Frequencies of Stepped Beam.

$$P = 1.5 \times 10^6 \text{ N.}, E = 2.0684 \times 10^5 \text{ MPa.}$$

Mode	ω (Linear)	ω (Nonlinear)	Mode
1	1.0112×10^3	1.1142×10^3	1st bending
2	5.1089×10^3	4.9748×10^3	2nd bending
3	1.1614×10^4	1.0806×10^4	3rd bending
4	1.2437×10^4	1.2179×10^4	1st longitudinal
5	1.9544×10^4	1.8154×10^4	4th bending

The nonlinear frequencies, except the first, have decreased. This is because the softening material effect is larger than the longitudinal tension. This result is consistent with the previous analyses (Section 10.2.2 and 10.2.3).

Table 10.2.1 DMAP Alters for SOL 63 (Version 66)

```

$ BEGINNING OF RF ALTER 63D89
$ + + + + +          22-DEC-1988          + + + + +
$
$ RF63D89 -- ALTER FOR NORMAL MODES WITH NONLINEAR STIFFNESS
$
$ JCL
$      ATTACH DATABASE FROM SOL 66 COLD START
$
$ EXECUTIVE DECK INPUT
$      NO SPECIAL CARDS REQUIRED
$
$ CASE CONTROL DECK INPUT
$      SEALL = ALL
$      SUPER = N
$
$ BULK DATA DECK INPUT
$      PARAM,LGDISP,1 - SELECT LARGE DISPLACEMENT EFFECTS (DEFAULT=-1)
$      PARAM,LOOPID,1 - SELECTS NONLINEAR STIFFNESS (DEFAULT=0)
$
ALTER 483 $ JUST AFTER LABEL JMPK4GG
$ ACTUALLY JUST AFTER OLD DBSTORE K4JJ
$ NOTE KELM AND KDICT IF IN HERE NEVER STORED
TYPE DB,ESTNL,UGV $ PATH = SSNL, LOCATION = DBUP
COND      LNSTIF, ACON $ SKIP THE ALTERS FOR SUPERELEMENTS
PARAM      //'LT'/S,N,NOLOOP/V,Y,LOOPID=0/1 $ READ LOOPID CARD
COND      LNSTIF, NOLOOP $ NO LOOPID, GO TO LINEAR STIFFNESS
$DBFETCH  /UGV,ESTNL,,,/SOLID/LOOPID//DBSET3 $ DATA FROM SOL 66 RUN
DBVIEW UGV66 =UGV (WHERE VERSION=RESTART AND SOLFETCH=LOOPID) $ +V66NX
DBVIEW ESTNL66=ESTNL (WHERE VERSION=RESTART AND SOLFETCH=LOOPID) $ +V66NX
EQUIV UGV66,UGV/ALWAYS/ESTNL66,ESTNL/ALWAYS $ +V66NX
PARAML     ESTNL// 'PRES'////S,N,NONLK $ CHECK PRESENCE OF ESTNL
COND      LNSTIF, NONLK $ NO ESTNL, GO TO LINEAR STIFFNESS
TA1,      MPT,ECTS,EPT,BGPDTS,SILS,ETT,CSTMS,DIT/
           EST,DESTNL,GEI,GPECT,ESTL/V,N,LUSETS/S,N,NOESTL/
           S,N,NP/2/S,N,NOGENL/SEID/S,Y,LGDISP=-1/
           V,N,NLAYERS=5 $ GENERATE TABLE FOR LINEAR ELEMENT
COND      JMPKGG,NOKGGX $
EMG      ESTL,CSTMS,MPT,DIT,GEOM2S,,,/KELM,KDICT,,,/
           S,N,NOKGXX/0/0/0//////////K6ROT $ STIFFNESS FOR LINEAR ELEMENTS
EMA      GPECT,KDICT,KELM,BGPDTS,SILS,CSTMS/KBJJZ,/ $
EMG      ESTL,CSTMS,MPT,DIT,,UGV,ETT,EDT/KDLEL,KDLDI,,,/
           S,Y,NOD=1/0/0//NP $ DIFFERENTIAL STIFFNESS FOR LINEAR ELEMENTS
EMA      GPECT,KDLDI,KDLEL,BGPDTS,SILS,CSTMS/KDLGG,-1/$
ADD      KBJJZ,KDLGG/KLTOT/ $
EMG      ESTNL,CSTMS,MPT,DIT,GEOM2S,,,/KELMNL,KDICTNL,,,/
           1/0/0//////////V,Y,K6ROT $ STIFFNESS FOR NONLINEAR ELEMENTS
EMA      GPECT,KDICTNL,KELMNL,BGPDTS,SILS,CSTMS/KBJJZNL,/ $
ADD      KLTOT,KBJJZNL/KB1/ $

```

```

EMG      ESTNL,CSTMS,MPT,DIT,,UGV,ETT,EDT/KDELM,KDDICT,,,/
          1/0/0//NP/   $ DIFFERENTIAL STIFFNESS FOR NONLINEAR ELEMENTS
EMA      GPECT,KDDICT,KDELM,BGPDTS,SILS,CSTMS/KBDJJ,/ -1/$
DELETE   /KJJZ,KELM,KDICT,,/ $                                ++V66NX
ADD      KB1,KBDJJ/KJJZ/ $
JUMP     JMPKGG $ SKIP  LINEAR STIFFNESS GENERATION
LABEL    LNSTIF $ LABEL FOR LINEAR STIFFNESS GENERATION
$ END OF RF ALTER 63D89

```

Table 10.2.2 SOL 66 Input Data Listing for Uniform Nonlinear Beam

ID BEAMWT, V66 \$ SHL 2/13/90 \$ A0186H, CSR6347 \$ JFC 13-DEC-1989

SOL 66

TIME 5

CEND

\$ GENERATE NONLINEAR STIFFNESS TERMS FOR SOL 63 RUN

TITLE=EXTENSION OF S-S BEAM

SUBTITLE=FIVE BEAM ELEMENTS

SET 5 = 5

SET 6 = 6

DISP = 6

STRESS = 5

SEALL = ALL

SUBCASE 1

LOAD=100

NLPARM=10

OUTPUT(PLOT)

CSCALE 1.3

PLOTTER NAST

SET 1 ALL

VIEW 0. , 0. , 0.

AXES MY , X , Z

FIND SCALE ORIGIN 1 SET 1

PTITLE=SIDE VIEW

PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID

BEGIN BULK

\$ PARAMETERS

PARAM LGDISP 1

\$

NLPARM	10	1	AUTO	1	PW	YES
--------	----	---	------	---	----	-----

\$ GEOMETRY

GRID	1	0.	0.	0.	12346
GRID	2	200.	0.	0.	246
GRID	3	400.	0.	0.	246
GRID	4	600.	0.	0.	246
GRID	5	800.	0.	0.	246
GRID	6	1000.	0.	0.	2346

\$ CONNECTIVITY

CBEAM	1	1	1	2	0.	1.	0.
CBEAM	2	1	2	3	0.	1.	0.
CBEAM	3	1	3	4	0.	1.	0.
CBEAM	4	1	4	5	0.	1.	0.
CBEAM	5	1	5	6	0.	1.	0.

\$ BEAM PROPERTIES

PBEAM	1	1	5.+3	4.1667+61.0417+6
-------	---	---	------	------------------

\$ MATERIAL PROPERTIES

MAT1	1	2.0684+5	0.3	7.8334-9
------	---	----------	-----	----------

MATS1	1	101	NLELAST
-------	---	-----	---------

TABLES1 101

+TB1

+TB1	0.	0.	.001	206.84	.003	500.	.005	700.	+TB2
+TB2	.011	1000.	ENDT						
\$ LOADING									
FORCE	100	6		1.+7	1.				
\$									
ENDDATA									

Table 10.2.3 SOL 63 Input Data Listing for Uniform Nonlinear Beam

```

RESTART
ID BEAMWT63, V66 $ SHL 2/13/90 $ A0186I, CSR6347R $ JFC 13-DEC-1989
DIAG 8
TIME 5
SOL 63 $ NORMAL MODES
COMPILE SOL63,SOUIN=MSCSOU,LIST,REF
$ RFALTER RF63D89 IN V66 AND V66A NEEDS THE FOLLOWING LINE INSERTED IN
$ THE ALTER.
$ FILE EST=OVRWRT/GEI=OVRWRT/GPECT=OVRWRT/KELM=OVRWRT/KDICT=OVRWRT $
ALTER 483 $
FILE EST=OVRWRT/GEI=OVRWRT/GPECT=OVRWRT/KELM=OVRWRT/KDICT=OVRWRT $
RFALTER RF63D89
CEND
TITLE=NONLINEAR VIBRATION OF S-S BEAM
SUBTITLE=FIVE BEAM ELEMENTS
DISP = ALL
SEALL = ALL
METHOD = 10
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 ALL
  VIEW 0. , 0. , 0.
  AXES MY , X , Z
  FIND SCALE ORIGIN 1 SET 1
  PTITLE=SIDE VIEW
  PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$ PARAMETERS
PARAM LGDISP 1
PARAM LOOPID 1
PARAM COUPMASS1
$ GEOMETRY
GRID 1 0. 0. 0. 12346
GRID 2 200. 0. 0. 246
GRID 3 400. 0. 0. 246
GRID 4 600. 0. 0. 246
GRID 5 800. 0. 0. 246
GRID 6 1000. 0. 0. 2346
$ CONNECTIVITY
CBEAM 1 1 1 2 0. 1. 0.
CBEAM 2 1 2 3 0. 1. 0.
CBEAM 3 1 3 4 0. 1. 0.
CBEAM 4 1 4 5 0. 1. 0.
CBEAM 5 1 5 6 0. 1. 0.
$ BEAM PROPERTIES
PBEAM 1 1 5.+3 4.1667+61.0417+6
$ MATERIAL PROPERTIES

```

```
MAT1      1      2.0684+5      0.3      7.8334-9
$ EIGENVALUE EXTRACTION
EIGR      10      SINV      0.      4000.      8      8
$
ENDDATA
```

Table 10.2.4. SOL 66 Input Data Listing for Plate

```

NASTRAN MESH $ PREFORT=2
ID PLATEWP, V66 $ SHL 2/13/90 $ A0186J $ KOK 11/85
SOL 66
TIME 5
CEND
$ GENERATE NONLINEAR STIFFNESS MATRIX FOR SOL 63 RESTART
TITLE= GEOMETRIC NONLINEAR ANALYSIS OF A FLAT PLATE
SUBTITLE= PRESSURE LOAD WITH CLAMPED EDGES
LABEL= REF. APP. NOTE, MAY 1981
    SEALL = ALL
    DISP = ALL
    STRESS = ALL
    SPC = 10
SUBCASE 1
    LOAD=1
    NLPARM=10
SUBCASE 2
    LOAD=2
    NLPARM=20
OUTPUT(PLOT)
    CSCALE 1.3
    PLOTTER NAST
    SET 1 ALL
    VIEW 30., 20., 0.
    FIND SCALE ORIGIN 1 SET 1
    PTITLE=ISOMETRIC VIEW
    PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1
    VIEW 0., 0., 0.
    AXES Y, X, Z
    FIND SCALE ORIGIN 1 SET 1
    PTITLE=SIDE VIEW
    PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
    AXES X, Y, Z
    FIND SCALE ORIGIN 1 SET 1
    PTITLE=FRONT VIEW
    PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$ PARAMETERS
PARAM LGDISP 1
PARAM K6ROT 100.
$
NLPARM 10 2 AUTO PW YES
NLPARM 20 2 AUTO 1 PW YES
$ VERTEX GRID POINTS
EGRID 1
EGRID 2 2.
EGRID 3 2. 1.
EGRID 4 1.

```

```

$ GENERATE GRID POINT ARRAY
GRIDG  1              10      -1      -2      -3              GRIDG1
+RIDG1  5             -4
$ GENERATE QUAD4 ELEMENTS
CGEN    QUAD4    201      200      1
$ PROPERTIES
PSHELL  200      1      .01      1
MAT1    1      2.+11      .3      8000.
$ CONSTRAINTS
SPCG    2      1      156      AD
SPCG    2      1      246      AB
SPCG    1      1      123456 BC      CD
SPCADD  10      1      2      3
$ APPLIED LOADS
PLOAD4  1      201      1.+5              THRU      250
PLOAD4  2      201      2.+5              THRU      250
$ THESE POINTS ARE FOR SOL 63
SPOINT  20001    THRU      20012
SPC1    3      0      20001    THRU      20012
$
ENDDATA

```


Table 10.2.5 SOL 63 Input Data Listing for Plate

```

NASTRAN MESH $ PREFORT=2
RESTART
ID PLATEWP63, V66 $ SHL 2/13/90 $ A0581B $ JAJ 8 DEC 82
TIME 5
SOL 63
COMPILE SOL63,SOUIN=MSCSOU,LIST,REF
ALTER 483 $
FILE EST=OVRWRT/GEI=OVRWRT/GPECT=OVRWRT/KELM=OVRWRT/KDICT=OVRWRT $
RFALTER RF63D89
CEND
TITLE=SYMMETRIC MODES
SUBTITLE=PRELOAD OF UNIFORM PRESSURE-CLAMPED EDGES
LABEL = REF. APP. NOTE OF MAY 1981
    SEALL = ALL
    SPC = 15
    SET 1 = 0
    DISP = ALL
    METHOD = 1
    DYNRED = 1
OUTPUT(PLOT)
    CSCALE 1.3
    PLOTTER NAST
    SET 1 ALL
    VIEW 30., 20., 0.
    FIND SCALE ORIGIN 1 SET 1
    PTITLE=ISOMETRIC VIEW
    PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1
    VIEW 0., 0., 0.
    AXES Y , X , Z
    FIND SCALE ORIGIN 1 SET 1
    PTITLE=SIDE VIEW
    PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
    AXES X , Y , Z
    FIND SCALE ORIGIN 1 SET 1
    PTITLE=FRONT VIEW
    PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$ PARAMETERS
PARAM LOOPID 4
PARAM LGDISP 1
PARAM K6ROT 100.
$ VERTEX GRID POINTS
EGRID 1
EGRID 2 2.
EGRID 3 2. 1.
EGRID 4 1.
$ GENERATE GRID POINT ARRAY
GRIDG 1 10 -1 -2 -3 GRIDG1

```

```

+RIDG1 5      -4
$ GENERATE QUAD4 ELEMENTS
CGEN  QUAD4  201  200  1
$ PROPERTIES
PSHELL 200    1      .01  1
MAT1    1      2.+11    .3  8000.
$ CONSTRAINTS
SPCG  2      1      156  AD
SPCG  2      1      246  AB
SPCG  1      1      123456 BC  CD
$ SYMMETRIC BOUNDARY CONDITION FOR SOL 63
SPCADD 15    1      2
$ DATA FOR GENERALIZED DYNAMIC REDUCTION AND EIGENVALUE EXTRACTION - SOL 63
SPOINT 20001  THRU  20012
DYNRED  1      160.    8
EIGR    1      MGIV  0.    160.      EIGR1
+IGR1  MASS
ASET1   0      20001  THRU  20012
QSET1   0      20001  THRU  20012
$
ENDDATA

```

Table 10.2.6. SOL 66 Input Data Listing for Stepped Beam

```

ID SBEAMWT, V66 $ SHL 1/13/90 $ A0186L KOK 12/85
TIME 5
SOL 66
CEND
$ GENERATE NONLINEAR STIFFNESS TERMS FOR SOL 63 RUN
TITLE=EXTENSION OF S-S STEPPED BEAM
SUBTITLE=FIVE BEAM ELEMENTS
  DISP = ALL
  STRESS = ALL
  SEALL = ALL
SUBCASE 1
  LOAD=100
  NLPARM=10
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 ALL
  VIEW 0. , 0. , 0.
  AXES MY , X , Z
  FIND SCALE ORIGIN 1 SET 1
  PTITLE=SIDE VIEW
  PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$ PARAMETERS
PARAM LGDISP 1
$
NLPARM 10 1 AUTO 1 PW YES
$ GEOMETRY
GRID 1 0. 0. 0. 12346
GRID 2 200. 0. 0. 246
GRID 3 400. 0. 0. 246
GRID 4 600. 0. 0. 246
GRID 5 800. 0. 0. 246
GRID 6 1000. 0. 0. 2346
$ CONNECTIVITY
CBEAM 1 11 1 2 0. 1. 0.
CBEAM 2 12 2 3 0. 1. 0.
CBEAM 3 13 3 4 0. 1. 0.
CBEAM 4 14 4 5 0. 1. 0.
CBEAM 5 15 5 6 0. 1. 0.
$ BEAM PROPERTIES
PBEAM 11 1 1.44+4 1.728+7 1.728+7
PBEAM 12 1 1.00+4 8.333+6 8.333+6
PBEAM 13 1 6.40+3 3.413+6 3.413+6
PBEAM 14 1 3.60+3 1.080+6 1.080+6
PBEAM 15 1 1.60+3 2.133+6 2.133+6
$ MATERIAL PROPERTIES
MAT1 1 2.0684+5 0.3 7.8334-9

```

```

MATS1  1      101      NLELAST
TABLES1 101
+TB1    0.      0.      .001    206.84  .003    500.    .005    700.    +TB1
+TB2    .011    1000.    ENDT
$ LOADING
FORCE  100      6      1.5+6    1.
$
ENDDATA

```

Table 10.2.7. SOL 63 Input Data Listing for Stepped Beam

```

RESTART
ID SBEMWT63, V66 $ SHL 2/13/90 $ A0186M $ KOK 12/85
DIAG 8
TIME 5
SOL 63
COMPILE SOL63,SOUIN=MSCSOV,LIST,REF
ALTER 483 $
FILE EST=OVRWRT/GEI=OVRWRT/GPECT=OVRWRT/KELM=OVRWRT/KDICT=OVRWRT $
RFALTER RF63D89
CEND
TITLE=VIBRATION OF S-S STEPPED BEAM
SUBTITLE=FIVE BEAM ELEMENTS
SEALL = ALL
DISP = ALL
METHOD = 10
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 ALL
  VIEW 0. , 0. , 0.
  AXES MY , X , Z
  FIND SCALE ORIGIN 1 SET 1
  PTITLE=SIDE VIEW
  PLOT MODAL 0 SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$ PARAMETERS
PARAM LOOPID 1
PARAM LGDISP 1
PARAM COUPMASS1
$ GEOMETRY
GRID 1 0. 0. 0. 12346
GRID 2 200. 0. 0. 246
GRID 3 400. 0. 0. 246
GRID 4 600. 0. 0. 246
GRID 5 800. 0. 0. 246
GRID 6 1000. 0. 0. 2346
$ CONNECTIVITY
CBEAM 1 11 1 2 0. 1. 0.
CBEAM 2 12 2 3 0. 1. 0.
CBEAM 3 13 3 4 0. 1. 0.
CBEAM 4 14 4 5 0. 1. 0.
CBEAM 5 15 5 6 0. 1. 0.
$ BEAM PROPERTIES
PBEAM 11 1 1.44+4 1.728+7 1.728+7
PBEAM 12 1 1.00+4 8.333+6 8.333+6
PBEAM 13 1 6.40+3 3.413+6 3.413+6
PBEAM 14 1 3.60+3 1.080+6 1.080+6
PBEAM 15 1 1.60+3 2.133+6 2.133+6

```

```
$ MATERIAL PROPERTIES
MAT1    1      2.0684+5    0.3    7.8334-9
$ EIGENVALUE EXTRACTION
EIGR    10      SINV    0.    4000.    8      8
$
ENDDATA
```

10.3 STATIC ANALYSIS USING SOL 99

Static analysis can be performed using SOL 99 (or 129), the nonlinear transient solution sequence. Since the effects of inertia and damping forces are ignored in the static analysis, the automatic time stepping algorithm is not applicable. However, the bisection algorithm is active with the adaptive stiffness matrix update strategy and other expedient iteration schemes such as the BFGS updates and the line search process. The TSTATIC option was implemented in Version 66 when the bisection was not available in SOL 66. The bisection algorithm is implemented in SOL 66 (or 106) in Version 67.

While the static capability in SOL 99 provides convenience for transient response analysis of a preloaded structure, it lacks some features provided by SOL 66. SOL 99 static capability is confined to what SOL 99 can perform and abides by its limitations. Restrictions on the SOL 99 input data include the following:

- No constraint changes (SPC, MPC) after the first subcase, including restarts
- No enforced displacements (SPCD) allowed
- No CREEP or thermal loads allowed

The SOL 99 static analysis is compared with SOL 66 for two example problems. In the first example, material nonlinearity is combined with geometric nonlinear effects in the analysis of a Z-shaped beam with a static load. The second example combines frictional contact with geometric nonlinear effects in the rotation of a beam with a GAP element, and illustrates SOL 99 static and dynamic analysis using the same input data.

10.3.1 Input Data for SOL 99 Static Analysis

The important input data for static analysis using SOL 99 are described below:

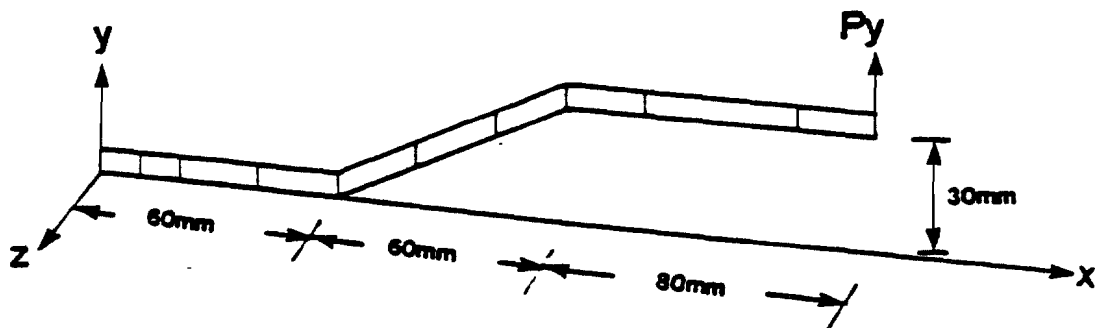
- The parameter TSTATIC with a value of 1 must be included in the Case Control or Bulk Data section in order to activate the static analysis. TSTATIC with a value of -1 (default) activates the dynamic analysis. The value of TSTATIC can be varied from subcase to subcase, thus allowing a static and dynamic analysis in a single run from the same input data.
- The solution strategy is controlled by the TSTEPNL Bulk Data entry, in which the ADAPT method (NLTRD2 module) must be specified for static analysis. A minimum of 4 time steps is required in the NDT field for each subcase. The MAXITER and MAXLS fields should be assigned a larger value than the defaults (e.g., 25 and 4) in order to allow more iterations and line searches, which is the general strategy adopted in the NLPARM for static analysis.
- The Bulk Data LSEQ, selected by LOADSET in the Case Control section, references static load entries (i.e., FORCE, MOMENT, PLOAD, etc.) and provides static to dy-

dynamic load conversion. LSEQ also selects DAREA which is internally generated to define the points of dynamic load application and is referenced by TLOADi. The time-dependent function for the dynamic load is specified by a TLOADi entry which is defined by a tabular function via TABLED input or by an analytic function. The static loading history should be defined following the same convention using TLOADi, i.e., time implies the static load step in the subcase with TSTATIC=1. One advantage in SOL 99 static analysis is that the loading path does not have to be increased linearly within a subcase as in SOL 66.

- The parameter MAXLP determines the maximum number of internal subincrements in the plasticity process. If a plastic material is used in the finite element model, MAXLP with a value of 10 should be included in the input data to conform with the static default. The default value for dynamic analysis is 5.

10.3.2 A Z-Shaped Beam Loaded with a Static Force

A Z-shaped cantilever beam, shown in Fig. 10.3.1, was proposed as a benchmark problem for testing nonlinear finite element analysis codes [10.4]. The Z-shaped beam is loaded statically with a vertical force of 2000 N at one end and clamped at the other end. The material is elastic-plastic obeying the von Mises yield criterion and the isotropic hardening rule. The problem is geometrically nonlinear, involving almost 90 degree rotation, but the follower force effect is absent.



$$\begin{aligned}
 A &= 68 \text{ mm}^2; \text{ cross-sectional area} \\
 I_{zz} &= 65.5067 \text{ mm}^4; \text{ moment of inertia} \\
 E &= 2 \times 10^5 \text{ N/mm}^2; \text{ Young's modulus} \\
 \nu &= 0.3; \text{ Poisson's ratio} \\
 \sigma_y &= 450 \text{ N/mm}^2; \text{ yield stress}
 \end{aligned}$$

Figure 10.3.1 Z-Shaped Cantilever Beam

The beam is modeled by ten BEAM elements of various lengths as seen in Fig. 10.3.1. Smaller elements are used near the fixed end because of the large plastic deformation anticipated in this area. Material properties are given by the MAT1, MATS1, and TABLES1 Bulk Data entries representing the elasto-plastic material as shown in Fig. 10.3.2. The clamped boundary condition at the end of the beam is imposed by constraining all six components of GRID 1. The load applied at the free end of the beam is specified by FORCE and LOAD Bulk Data entries in SOL 66 and by FORCE, LSEQ, TLOAD1, and TABLED1 Bulk Data entries in SOL 99. The NLPARM and TSTEPNL Bulk Data entries provide the solution strategy in SOL 66 and SOL 99, respectively. The input data are given in Table 10.3.1 and 10.3.2.

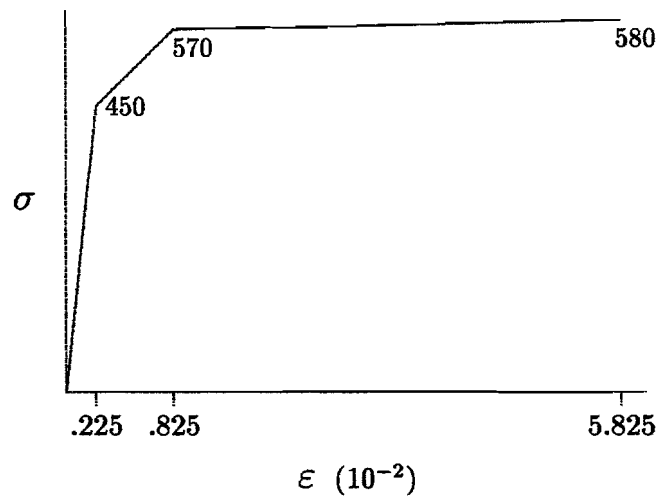


Figure 10.3.2 Nonlinear Stress-Strain Relationship for Z-Shaped Beam

The analysis is challenging because the substantial loading results in very large rotation and severe plastic deformation at the fixed end. The loading history, which is identical for SOL 66 and SOL 99, is divided into 20 subcases. Each subcase specifies 10 increments, except for subcase 2 and 3 which contain 25 increments to cope with the severe rotation and plastic yielding in this domain.

The solution produced by SOL 66 and SOL 99 was identical. Fig. 10.3.3 shows the deformed shapes of the beam at progressively increasing loading stages. The load-deflection curve at the free end is shown in Fig. 10.3.4. The numerical performance of each subcase, comparing SOL 66 and SOL 99, is tabulated below.

Performance Comparison for Z-Shaped Beam

Subcase ID	SOL 66			SOL 99		
	No. of Iter	No. of K Update	No. of Bis	No. of Iter	No. of K Update	No. of Bis
1	70	10	0	145	3	0
2	391	59	20	576	13	17
3	249	50	1	701	20	12
4	126	19	2	361	8	10
5	70	12	0	141	3	0
6	78	11	0	345	11	20
7	63	3	0	115	3	0
8	57	2	0	68	1	0
9	48	2	0	56	1	0
10	45	2	0	51	1	0
11	42	1	0	47	1	0
12	37	2	0	39	1	0
13	34	1	0	37	1	0
14	34	1	0	31	1	0
15	31	1	0	29	1	0
16	28	1	0	29	1	0
17	25	0	0	27	1	0
18	26	1	0	32	1	0
19	21	1	0	26	1	0
20	20	0	0	24	1	0
TOTAL	1495	179	23	2880	74	59

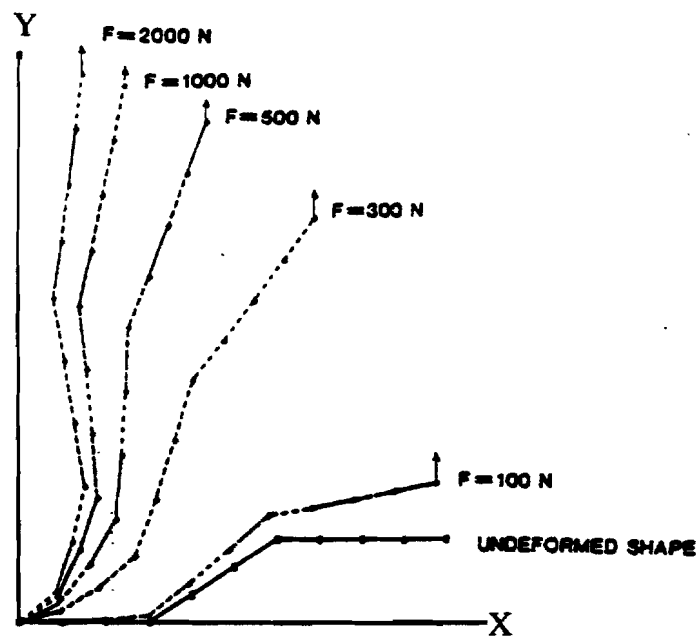


Figure 10.3.3 Deformed Shapes of Z-Shaped Beam

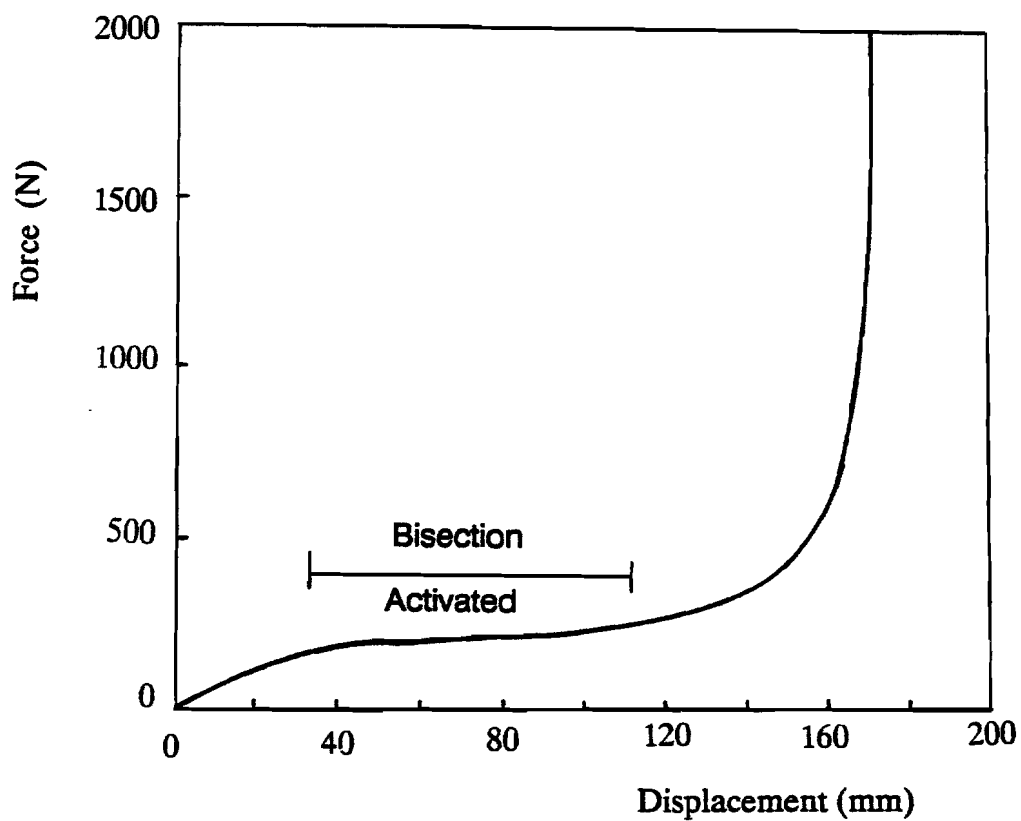


Figure 10.3.4 Load-Deflection Curve for Z-Shaped Beam

Table 10.3.1 SOL 66 Input Data Listing for Z-Shaped Beam

```

ID      ZBEAM, V67      $ KOK 5-APR-85, SHL 9-MAY-85, CYP 19-OCT-87
TIME    30              $ CPU MINUTES FOR VAX
SOL     66              $ NONLINEAR STATIC ANALYSIS
DIAG    8, 50           $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = Z-BEAM ANALYSIS WITH 10 BEAM ELEMENTS
SUBTITLE = COLOGNE CHALLENGE, AUG. 1984
SEALL = ALL
DISP = ALL
SUBCASE 1
  LOAD = 10
  NLPARM = 10
SUBCASE 2
  LOAD = 20
  NLPARM = 25
SUBCASE 3
  LOAD = 30
  NLPARM = 25
SUBCASE 4
  LOAD = 40
  NLPARM = 10
SUBCASE 5
  LOAD = 50
  NLPARM = 10
SUBCASE 6
  LOAD = 60
  NLPARM = 10
SUBCASE 7
  LOAD = 70
  NLPARM = 10
SUBCASE 8
  LOAD = 80
  NLPARM = 10
SUBCASE 9
  LOAD = 90
  NLPARM = 10
SUBCASE 10
  LOAD = 100
  NLPARM = 10
SUBCASE 11
  LOAD = 110
  NLPARM = 10
SUBCASE 12
  LOAD = 120
  NLPARM = 10
SUBCASE 13
  LOAD = 130
  NLPARM = 10

```

```

SUBCASE 14
  LOAD   = 140
  NLPARM = 10
SUBCASE 15
  LOAD   = 150
  NLPARM = 10
SUBCASE 16
  LOAD   = 160
  NLPARM = 10
SUBCASE 17
  LOAD   = 170
  NLPARM = 10
SUBCASE 18
  LOAD   = 180
  NLPARM = 10
SUBCASE 19
  LOAD   = 190
  NLPARM = 10
SUBCASE 20
  LOAD   = 200
  NLPARM = 10
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  MAXI DEFO 1.
  SET 1 = ALL
  VIEW 0., 0., 0.
  AXES Z, X, Y
  PTITLE = TOP VIEW
  FIND SCALE ORIGIN 1 SET 1
  PLOT STATIC 0 MAXI DEFO 1. SET 1 ORIGIN 1
OUTPUT(XYPLT)
  CSCALE 1.3
  PLOTTER NAST
  XTITLE = FORCE (N)
  YTITLE = TIP DISPLACEMENT (MM) AT NODES 5, 8, 11
  XYPLT = DISP RESP /5(T2),8(T2),11(T2)
BEGIN BULK
PARAM  LGDISP 1
$$ SOLUTION CONTROL
NLPARM 10      10      AUTO      7
NLPARM 25      25      AUTO      7
$$ GEOMETRY
GRID   1      0.      0.      0.      123456
GRID   2      10.     0.      0.      345
GRID   3      20.     0.      0.      345
GRID   4      40.     0.      0.      345
GRID   5      60.     0.      0.      345
GRID   6      80.     10.     0.      345

```

GRID	7		100.	20.	0.		345
GRID	8		120.	30.	0.		345
GRID	9		140.	30.	0.		345
GRID	10		180.	30.	0.		345
GRID	11		200.	30.	0.		345
GRID	99		0.	200.	0.		123456
\$\$ CONNECTIVITY							
CBEAM	100	100	1	2	99		
CBEAM	110	100	2	3	99		
CBEAM	120	100	3	4	99		
CBEAM	130	100	4	5	99		
CBEAM	140	100	5	6	99		
CBEAM	150	100	6	7	99		
CBEAM	160	100	7	8	99		
CBEAM	170	100	8	9	99		
CBEAM	180	100	9	10	99		
CBEAM	190	100	10	11	99		
PLOTEL	200	1	99				
\$\$ PROPERTIES							
PBEAM	100	110	68.	65.5067	2266.667		
MAT1	110	2.+5		.3			
MATS1	110	111	PLASTIC			450.	
TABLES1	111						+TBS1
+TBS1	0.	0.	2.25-3	450.	8.25-3	570.	5.825-2 580. +TBS2
+TBS2	ENDT						
\$\$ LOADING							
FORCE	1000	11		2000.	0.	1.	0.
LOAD	10	1.	.05	1000			
LOAD	20	1.	.1	1000			
LOAD	30	1.	.15	1000			
LOAD	40	1.	.2	1000			
LOAD	50	1.	.25	1000			
LOAD	60	1.	.3	1000			
LOAD	70	1.	.35	1000			
LOAD	80	1.	.4	1000			
LOAD	90	1.	.45	1000			
LOAD	100	1.	.5	1000			
LOAD	110	1.	.55	1000			
LOAD	120	1.	.6	1000			
LOAD	130	1.	.65	1000			
LOAD	140	1.	.7	1000			
LOAD	150	1.	.75	1000			
LOAD	160	1.	.8	1000			
LOAD	170	1.	.85	1000			
LOAD	180	1.	.9	1000			
LOAD	190	1.	.95	1000			
LOAD	200	1.	1.	1000			
ENDDATA							

Table 10.3.2 SOL 99 Input Data Listing for Z-Shaped Beam

```

ID      ZBEAMD,V67      $ SSH 3-DEC-87, SHL 9-MAY-85, CYP 19-OCT-87
TIME    30              $ CPU MINUTES FOR VAX
SOL     99              $ NONLINEAR TRANSIENT ANALYSIS
DIAG    8,50            $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND

TITLE = Z-BEAM ANALYSIS WITH 10 BEAM ELEMENTS
SUBTITLE = COLOGNE CHALLENGE, AUG. 1984
SEALL = ALL
DISP = ALL
$-----$
LOADSET = 1000 $
$-----$
SUBCASE 1
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 2
  DLOAD = 100
  TSTEPNL = 25
SUBCASE 3
  DLOAD = 100
  TSTEPNL = 25
SUBCASE 4
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 5
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 6
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 7
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 8
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 9
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 10
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 11
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 12
  DLOAD = 100
  TSTEPNL = 10

```

```

SUBCASE 13
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 14
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 15
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 16
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 17
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 18
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 19
  DLOAD = 100
  TSTEPNL = 10
SUBCASE 20
  DLOAD = 100
  TSTEPNL = 10
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  MAXI DEFO 1.
  SET 1 = ALL
  VIEW 0., 0., 0.
  AXES Z, X, Y
  FIND SCALE ORIGIN 1 SET 1
  PLOT TRANS 0 MAXI DEFO 1. SET 1 ORIGIN 1
OUTPUT(XYPLT)
  CSCALE 1.3
  PLOTTER NAST
  XTITLE = FORCE (N)
  YTITLE = TIP DISPLACEMENT (MM) AT NODES 5, 8, 11
  XYPLT DISP RESP /5(T2),8(T2),11(T2)
BEGIN BULK
$-----$
PARAM TSTATIC 1
$-----$
PARAM LGDISP 1
PARAM MAXLP 10
$$ SOLUTION CONTROL
TSTEPNL 10 10 .1 ADAPT 25 +TS1
+TS1 4
TSTEPNL 25 25 .04 ADAPT 25 +TS2

```


+TS2

4

\$\$ GEOMETRY

GRID	1	0.	0.	0.	123456
GRID	2	10.	0.	0.	345
GRID	3	20.	0.	0.	345
GRID	4	40.	0.	0.	345
GRID	5	60.	0.	0.	345
GRID	6	80.	10.	0.	345
GRID	7	100.	20.	0.	345
GRID	8	120.	30.	0.	345
GRID	9	140.	30.	0.	345
GRID	10	180.	30.	0.	345
GRID	11	200.	30.	0.	345
GRID	99	0.	200.	0.	123456

\$\$ CONNECTIVITY

CBEAM	100	100	1	2	99
CBEAM	110	100	2	3	99
CBEAM	120	100	3	4	99
CBEAM	130	100	4	5	99
CBEAM	140	100	5	6	99
CBEAM	150	100	6	7	99
CBEAM	160	100	7	8	99
CBEAM	170	100	8	9	99
CBEAM	180	100	9	10	99
CBEAM	190	100	10	11	99
PLOTEL	200	1	99		

\$\$ PROPERTIES

PBEAM	100	110	68.	65.5067	2266.667			
MAT1	110	2.+5		.3				
MATS1	110	111	PLASTIC			450.		
TABLES1	111							+TBS1
+TBS1	0.	0.	2.25-3	450.	8.25-3	570.	5.825-2	580. +TBS2
+TBS2	ENDT							

\$-----\$

\$\$ LOADING

LSEQ	1000	910	110					
FORCE	110	11		2000.	0.	1.	0.	
TLOAD1	100	910		0	200			
TABED1	200							+TBD1
+TBD1	0.	0.	1.	.05	20.	1.	ENDT	

\$-----\$

ENDDATA

10.3.3 Rotation of a Beam with Friction

A uniform beam has a length of 12 inches and a cross-sectional area of 0.15 in^2 . One end of the beam is clamped and the other end is attached to a GAP element as shown in Fig. 10.3.5. A force of 50 lb acting in the $-z$ direction is applied as a preload providing a friction force at the end attached to the GAP. In the first phase, the beam is statically loaded by a 1000 in-lb moment which is applied in the $+z$ direction. In the second phase, the moment is removed while the applied force remains. The unloading phase was analyzed statically (SOL 66) and dynamically (SOL 99), and the beam responses were observed.

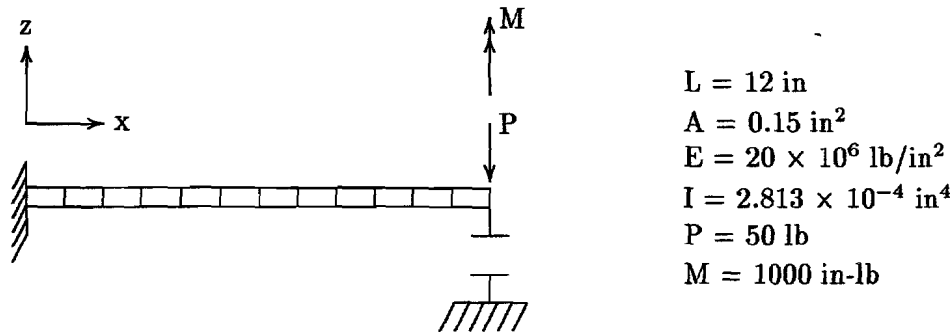


Figure 10.3.5 Uniform Beam with a Gap

The beam is modeled by twelve BEAM elements of equal length with linear elastic material. Friction at one end of the beam is simulated by a GAP element, connected at GRID 112 and 212, which is initially closed. The coefficient of static friction (μ_s) is 0.5, the coefficient of kinetic friction (μ_k) is 0.3, the closed axial stiffness (K_a) is 10^4 lb/in , and the transverse shear stiffness (K_t) is $5.0 \times 10^3 \text{ lb/in}$ (default value). The clamped boundary condition at the end of the beam is imposed by constraining all six components of GRID 100. The applied load (at GRID 112) is specified using FORCE, MOMENT and LOAD Bulk Data entries in SOL 66, while in SOL 99 the Bulk Data entries LSEQ, DLOAD, TLOAD1 and TABLED1 are used in addition to FORCE and MOMENT entries. The solution strategy is provided by the NLPARM Bulk Data entry in SOL 66 and by the TSTEPNL Bulk Data entry in SOL 99. The input data for SOL 66 and SOL 99 are given in Table 10.3.3 and 10.3.4, respectively.

The analysis is performed in three subcases. The first subcase applies the preload (force of 50 lb) in 4 increments. The second subcase statically loads the beam with a 1000 in-lb moment applied in 10 increments, resulting in a total rotation of 115.8° at the end. Notice that in SOL 99 the first and second subcase use PARAM,TSTATIC,1 for the static analysis. The same loading history is specified in the first and second subcase of the SOL 66 and SOL 99 analysis. The deformed shapes of the beam during static loading are identical for SOL 66 and SOL 99 and are shown in Fig. 10.3.6. The deformed shapes are labeled from 1 to 6, corresponding to

the tabulated applied moment and θ_{end} , the rotation of the end of the beam about the z-axis. The first deformed shape does not exhibit translation of the end point because the lateral force induced by the applied moment is less than the friction force. When the induced lateral force exceeds the frictional force, the end point starts to move as evidenced by the subsequent deformed shapes in the figure.

In SOL 66 the third subcase statically unloads the beam by removing the applied moment while retaining the 50 lb preload. The resulting deformed shapes are shown in Fig. 10.3.7. The third subcase in SOL 99 uses PARAM,TSTATIC,-1 for dynamic unloading of the beam after the moment is released, maintaining the preload on the GAP element. Fig. 10.3.8 shows the rotation of the end of the beam in radians about the z-axis during dynamic unloading. The angle (θ_{end}) starts from 115.8° and diminishes as the beam vibrates due to friction.

The numerical performance of the second subcase in terms of the number of iterations, stiffness updates, and bisections is tabulated below for SOL 66 and SOL 99. Notice that the bisection algorithm was implemented in SOL 66 in Version 67 and was activated in the SOL 66 analysis.

Performance Comparison During Static Loading (Subcase 2)

	No. of iterations	No. of K updates	No. of bisections
SOL 66, V67	246	66	2
SOL 99, V67	1533	63	25

The table shows a variation in the efficiency of the two solutions. Different stiffness update strategies are provided in SOL 66 and SOL 99, which have significant effects on the efficiency of the iterative process. SOL 99 limits the number of stiffness updates within an increment and updates in reference to the last converged step except for GAP induced update. SOL 66 has a more liberal stiffness update strategy, allowing numerous intermediate stiffness updates as required for convergence based on the global divergence criteria in addition to the GAP induced updates. This signifies the effects of timely stiffness updates on the efficiency of the iterative process.

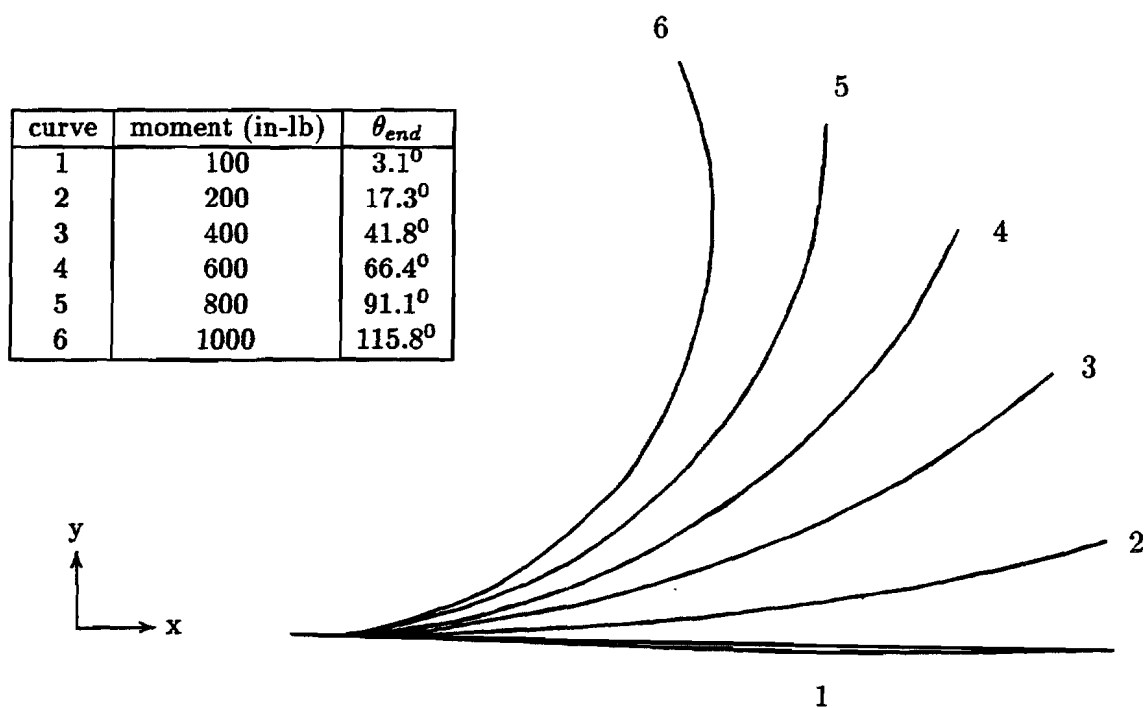


Figure 10.3.6 Deformed Shapes of Beam During Static Loading (SOL 66 and SOL 99)

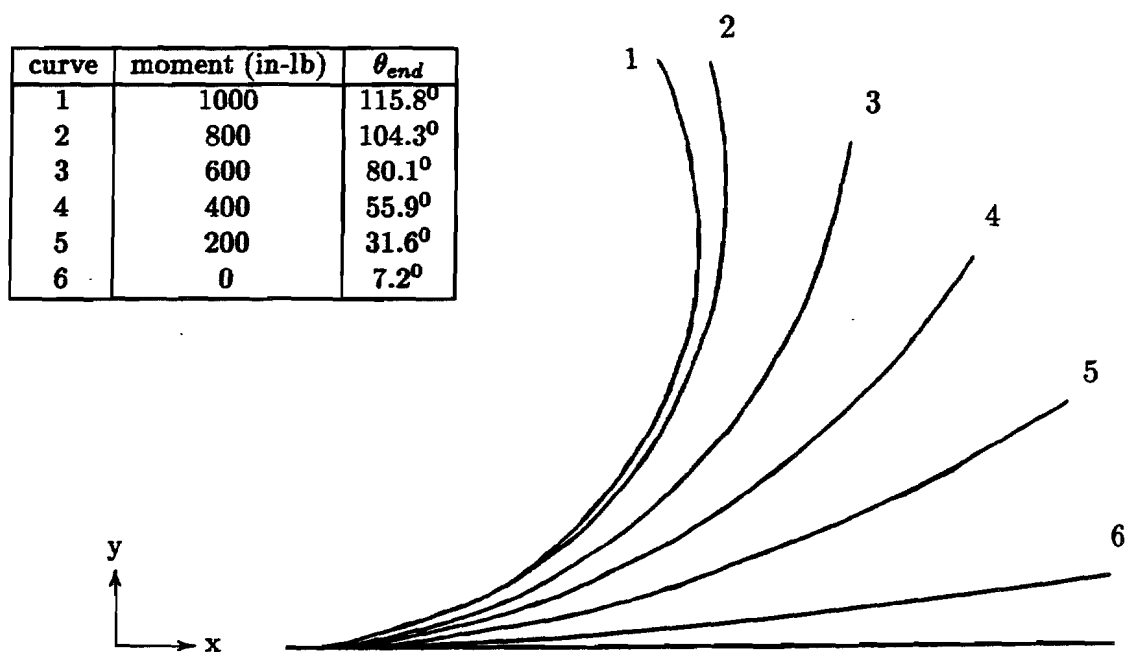
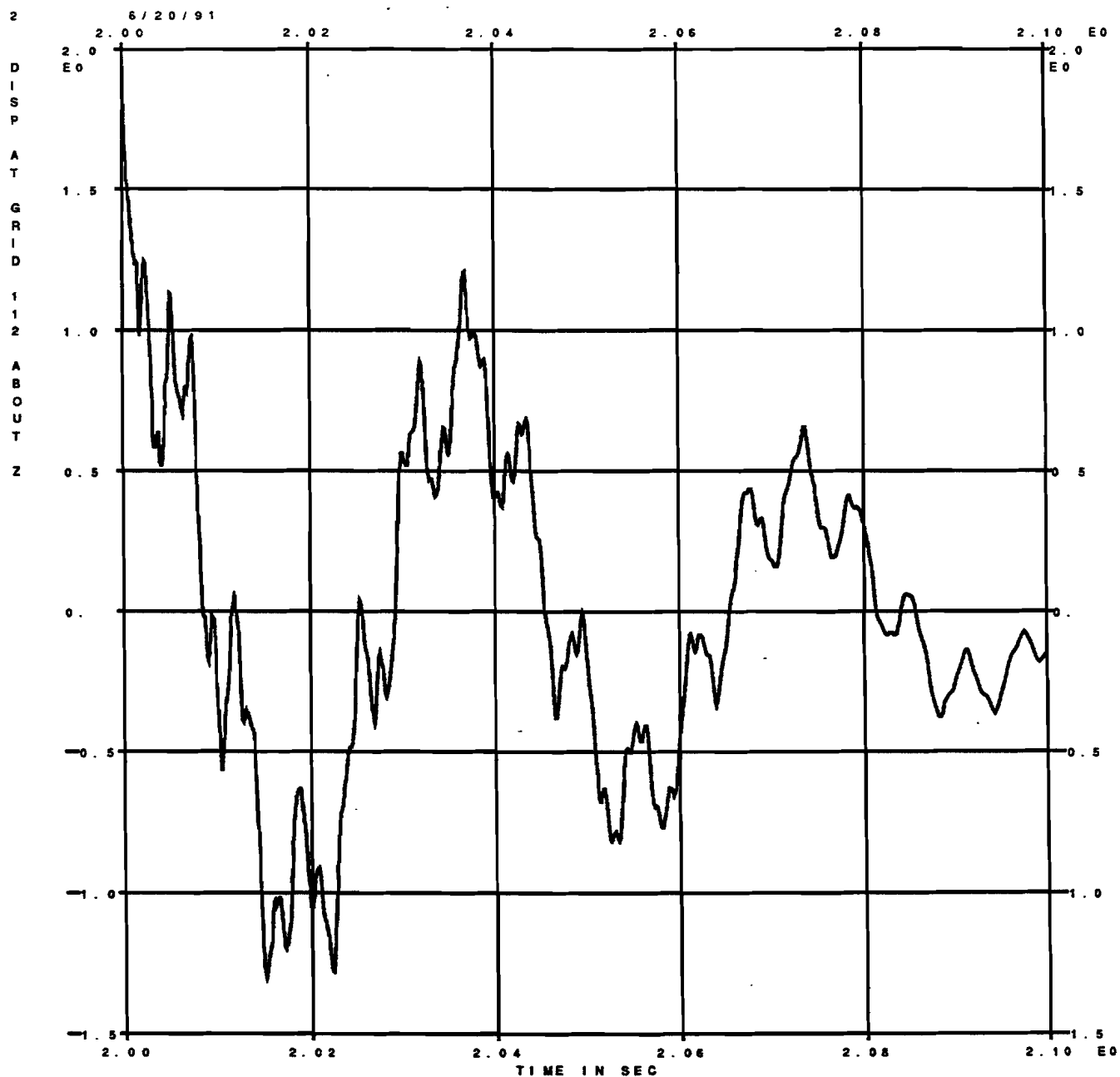


Figure 10.3.7 Deformed Shapes of Beam During Static Unloading (SOL 66)



FIXED/FREE BEAM, WITH A GAP AND APPLIED MOMENT AT FREE END
MOMENT IS 1000 IN-LB

Figure 10.3.8 Rotation (θ_z) at End of Beam During Dynamic Unloading

Table 10.3.3 SOL 66 Input Data Listing for Beam with Friction

```

ID      FRIBeam66, V67  $ APR 15-AUG-90
TIME    30              $ CPU MINUTES
SOL      66             $ NONLINEAR STATIC ANALYSIS
DIAG     8,50           $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND

TITLE = FIXED/FREE BEAM, WITH A GAP AND APPLIED MOMENT AT FREE END
SUBTITLE = MOMENT IS 1000 IN-LB
ECHO = BOTH
SEALL = ALL
  SET 1 = 101 THRU 112
  SET 2 = 100
  SET 3 = 200
  DISP = 1
  SPCF = 2
  STRES = 3
SUBCASE 1      $ FORCE = 50 LB
  NLPARM = 10
  LOAD = 101
SUBCASE 2      $ MOMENT = 1000 IN-LB
  NLPARM = 20
  LOAD = 201
SUBCASE 3      $ UNLOADING
  NLPARM = 30
  LOAD = 101
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  MAXI DEFO .001
  SET 1 = ALL EXCEPT GAP
  SET 2 = ALL EXCEPT PLOTTEL, EXCLUDE GRID POINTS 1000 THRU 1003
  VIEW 0., 0., 0.
  AXES MZ , X , Y
  FIND SCALE ORIGIN 1 SET 1
  PTITLE = TOP VIEW
  PLOT  STATIC 0 MAXI DEFO .001 SET 1 ORIGIN 1
BEGIN BULK
PARAM  LGDISP 1
$$ SOLUTION CONTROL
NLPARM 10      4      AUTO      YES
NLPARM 20      10     AUTO      40     YES
NLPARM 30      10     AUTO      40     YES
$$ GEOMETRY
GRID   100      0.     0.     0.     123456
GRID   101      1.     0.     0.     45
GRID   102      2.     0.     0.     45
GRID   103      3.     0.     0.     45
GRID   104      4.     0.     0.     45
GRID   105      5.     0.     0.     45

```

GRID	106		6.	0.	0.		45
GRID	107		7.	0.	0.		45
GRID	108		8.	0.	0.		45
GRID	109		9.	0.	0.		45
GRID	110		10.	0.	0.		45
GRID	111		11.	0.	0.		45
GRID	112		12.	0.	0.		45
GRID	212		12.	0.	-.01		123456
\$\$ CONNECTIVITY							
CBEAM	1	10	100	101	0.	0.	1.
CBEAM	2	10	101	102	0.	0.	1.
CBEAM	3	10	102	103	0.	0.	1.
CBEAM	4	10	103	104	0.	0.	1.
CBEAM	5	10	104	105	0.	0.	1.
CBEAM	6	10	105	106	0.	0.	1.
CBEAM	7	10	106	107	0.	0.	1.
CBEAM	8	10	107	108	0.	0.	1.
CBEAM	9	10	108	109	0.	0.	1.
CBEAM	10	10	109	110	0.	0.	1.
CBEAM	11	10	110	111	0.	0.	1.
CBEAM	12	10	111	112	0.	0.	1.
CGAP	200	20	212	112	1.	0.	0.
\$\$ PROPERTIES							
PBEAM	10	11	.15	2.813-4	2.813-4		
MAT1	11	2.E+7		.3	7.25-4		
PGAP	20	0.	0.	1.E+4	5.E+3	.5	.3 +PG1
+PG1	.02						
\$\$ LOADING							
LOAD	101	1.	1.	100			
LOAD	201	1.	1.	100	1.	200	
FORCE	100	112		50.	0.	0.	-1.
MOMENT	200	112		1000.	0.	0.	1.
\$\$ FRAME FOR PLOT							
GRID	1000		-2.	-2.			123456
GRID	1001		+13.	-2.			123456
GRID	1002		+13.	10.			123456
GRID	1003		-2.	10.			123456
PLOTEL	200	1000	1001				
PLOTEL	201	1001	1002				
PLOTEL	202	1002	1003				
PLOTEL	203	1003	1000				
\$							
ENDDATA							

Table 10.3.4 SOL 99 Input Data Listing for Beam with Friction

```

ID      FRIBeam99, V67  $ APR 15-AUG-90
TIME    90      $ CPU MINUTES
SOL      99      $ NONLINEAR STATIC ANALYSIS
DIAG     8,50    $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = FIXED/FREE BEAM, WITH A GAP AND APPLIED MOMENT AT FREE END
SUBTITLE = MOMENT IS 1000 IN-LB
ECHO = BOTH
SEALL = ALL
      SET 1 = 101 THRU 112
      SET 2 = 100
      SET 3 = 200
      DISP = 1
      SPCF = 2
      STRES = 3
$-----$
LOADSET = 1000 $
$-----$
SUBCASE 1      $ FORCE = 50 LB
$-----$
      PARAM TSTATIC 1 $
$-----$
      TSTEPNL = 10
      DLOAD = 101
SUBCASE 2      $ MOMENT = 1000 IN-LB
$-----$
      PARAM TSTATIC 1 $
$-----$
      TSTEPNL = 20
      DLOAD = 201
SUBCASE 3      $ DYNAMIC UNLOADING
$-----$
      PARAM TSTATIC -1 $
$-----$
      TSTEPNL = 30
      DLOAD = 101
OUTPUT(PLOT)
      CSCALE 1.3
      PLOTTER NAST
      MAXI DEFO .001
      SET 1 = ALL EXCEPT GAP
      SET 2 = ALL EXCEPT PLOTTEL, EXCLUDE GRID POINTS 1000 THRU 1003
      VIEW 0., 0., 0.
      AXES MZ , X , Y
      FIND SCALE ORIGIN 1 SET 1
      PTITLE = TOP VIEW
      PLOT TRANS 0 TIME 0.0 2.0 MAXI DEFO .001 SET 1 ORIGIN 1
OUTPUT(XY PLOT)

```



```

CSCALE 1.3
PLOTTER NAST
XMIN = 2.0
XGRID LINES = YES
YGRID LINES = YES
XTITLE =
TIME IN SEC
YTITLE = DISP AT GRID 112 ALONG Y
XYPLOT DISP RESP/112(T2)
YTITLE = DISP AT GRID 112 ABOUT Z
XYPLOT DISP RESP/112(R3)
BEGIN BULK
PARAM LGDISP 1
$$ SOLUTION CONTROL
TSTEPNL 10      4      .25      ADAPT      25      +TS1
+TS1
TSTEPNL 20      10      .1      ADAPT      40      +TS2
+TS2
TSTEPNL 30      50      .002     ADAPT
$$ GEOMETRY
GRID 100      0.      0.      0.      123456
GRID 101      1.      0.      0.      45
GRID 102      2.      0.      0.      45
GRID 103      3.      0.      0.      45
GRID 104      4.      0.      0.      45
GRID 105      5.      0.      0.      45
GRID 106      6.      0.      0.      45
GRID 107      7.      0.      0.      45
GRID 108      8.      0.      0.      45
GRID 109      9.      0.      0.      45
GRID 110     10.      0.      0.      45
GRID 111     11.      0.      0.      45
GRID 112     12.      0.      0.      45
GRID 212     12.      0.      -.01     123456
$$ CONNECTIVITY
CBEAM 1      10      100      101      0.      0.      1.
CBEAM 2      10      101      102      0.      0.      1.
CBEAM 3      10      102      103      0.      0.      1.
CBEAM 4      10      103      104      0.      0.      1.
CBEAM 5      10      104      105      0.      0.      1.
CBEAM 6      10      105      106      0.      0.      1.
CBEAM 7      10      106      107      0.      0.      1.
CBEAM 8      10      107      108      0.      0.      1.
CBEAM 9      10      108      109      0.      0.      1.
CBEAM 10     10      109      110      0.      0.      1.
CBEAM 11     10      110      111      0.      0.      1.
CBEAM 12     10      111      112      0.      0.      1.
CGAP 200     20      212      112      1.      0.      0.
$$ PROPERTIES
PBEAM 10     11      .15      2.813-4 2.813-4

```

MAT1	11	2.E+7		.3	7.25-4				
PGAP	20	0.	0.	1.E+4		5.E+3	.5	.3	+PG1
+PG1	.02								

-----\$

\$\$ LOADING (SUBCASE 1 AND 3)

LSEQ	1000	910	100						
FORCE	100	112		50.	0.	0.	-1.		
TLOAD1	101	910			111				
TABLED1	111								+TB1
+TB1	0.	0.	1.	1.	10.	1.	ENDT		

\$\$ LOADING (SUBCASE 2)

LSEQ	1000	920	200						
MOMENT	200	112		1000.	0.	0.	1.		
DLOAD	201	1.	1.	101	1.	2001			
TLOAD1	2001	920			211				
TABLED1	211								+TB2
+TB2	0.	0.	1.	0.	2.	1.	ENDT		

-----\$

\$\$ FRAME FOR PLOT

GRID	1000		-2.	-2.		123456
GRID	1001		+13.	-2.		123456
GRID	1002		+13.	10.		123456
GRID	1003		-2.	10.		123456
PLOTEL	200	1000	1001			
PLOTEL	201	1001	1002			
PLOTEL	202	1002	1003			
PLOTEL	203	1003	1000			

\$

ENDDATA

Chapter 11

EXAMPLE PROBLEMS

11.1 SNAP-THROUGH ANALYSIS OF A SPHERICAL DIAPHRAGM BY NEWTON'S ITERATION

11.1.1 Problem Description

A spherical diaphragm resting on a rigid backing plate is subjected to a uniform pressure as shown in Fig. 11.1.1. It is desired to find the critical pressure loads for snap-through and snap-back. The diaphragm has a thickness of 0.15 in. and the gap at the center between the diaphragm and the backing plate is 0.757 in. Material properties of the diaphragm are given as follows:

Young's modulus:	$E = 2.0 \times 10^5 \text{ psi}$
Poisson's ratio:	$\nu = 0.28$

The objective of this analysis is to study the effects of NLPARM parameters on the convergence characteristics when difficulties arise due to snap-through. The geometry and the loadings are axisymmetric. For simplicity, it is further assumed that the deformation pattern is confined to axisymmetric shapes.

Three cases are analyzed in this study according to the boundary conditions of the diaphragm edge, i.e.,

- Clamped boundary
- Simply supported boundary (and release of constraints)
- Sliding and simply supported boundary

11.1.2 Finite Element Modeling and Input Data

The main features of this problem are geometric nonlinearities due to large displacements and changes in constraints due to the backing plate. A 10-degree sector of the diaphragm is modeled using shell elements with axisymmetric boundary conditions. The contact process between the diaphragm and the backing plate is simulated by gap elements as shown in Fig. 11.1.2. The flange portion is omitted in the model. Various boundary conditions are imposed on the periphery for the three different cases to be investigated. The typical input data for the clamped and the simply supported cases are listed in Tables 11.1.1 and 11.1.2, respectively.

Executive Control Section

SOL 66 (in Version 65) is used for the static analysis of the snap-through phenomena.

Case Control Section

Subcases are used to change the loading path and/or boundary conditions. Within each subcase, the loads are subdivided into equal increments and each load increment requires iterations for convergence. As shown in Tables 11.1.1 and 11.1.2, Case Control data include loading and unloading phases. In the case of the simply supported boundary in Table 11.1.2, Subcase 7 is supplemented to remove the constraints at the periphery after full unloading.

Bulk Data Section

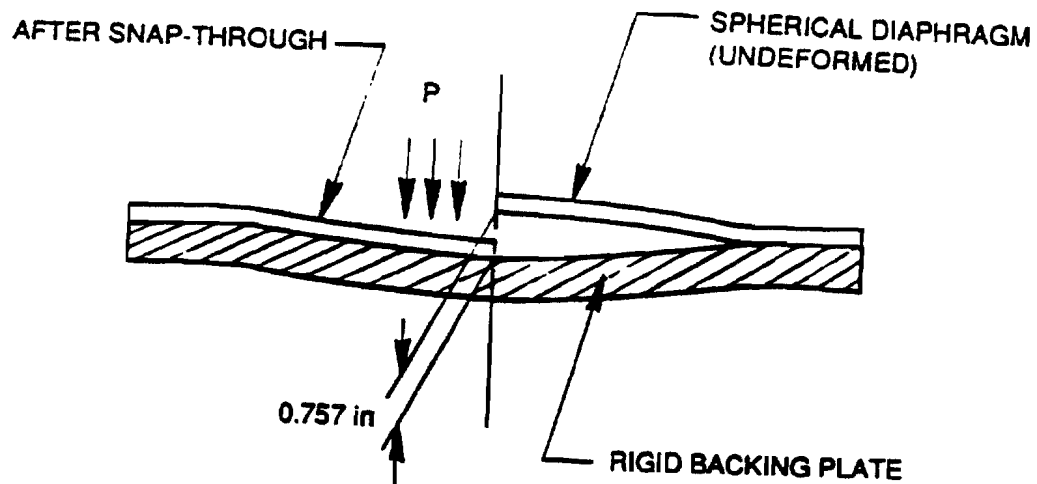
A 10-degree sector of the diaphragm is modeled by one TRIA3 and six QUAD4 elements with linear elastic material. The contact process between the diaphragm and the backing plate is simulated by 13 GAP elements as shown in Fig. 11.1.2. A total of 28 GRID points are used in the model.

The geometry and the output data are described in the cylindrical coordinate system. The axisymmetric boundary condition is imposed by constraining components 2, 4, and 6 of the grid points on the planes of symmetry. The apex point is allowed to move only in the axis of symmetry. Various boundary conditions are applied to the GRID points on the periphery (GRIDs 170 and 171) as follows:

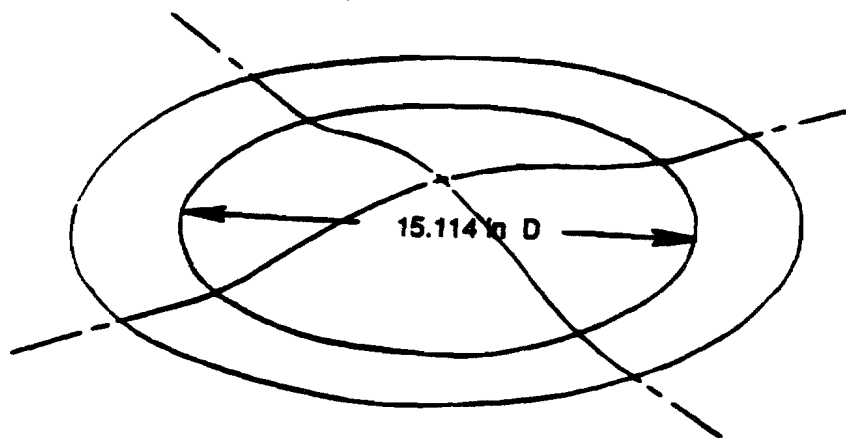
- Clamped case: all components (1-6) fixed
- Simply supported case: components 1, 2, 3, 4, and 6 fixed
- Sliding and simply supported case: components 2, 3, 4, and 6 fixed

All grid points (GRIDs 200 - 261) simulating the backing plate are fixed. The number of total DOFs of the problem is 168, and the number of active DOFs is 37 for the clamped case.

Effects of geometric nonlinearity are included in the analysis by specifying PARAM, LGDISP, +1. The external pressure applied to the diaphragm surface in each subcase is specified using PLOAD2. Smaller load increments are used around the loads where snap-through and snap-back are anticipated to occur. The solution strategy is provided by the NLPARM Bulk Data, which are varied from subcase to subcase depending on the convergence characteristics.



Side View



Isometric View

Figure 11.1.1. Spherical Diaphragm Subjected to a Uniform Pressure
(a) Side View (b) Isometric View

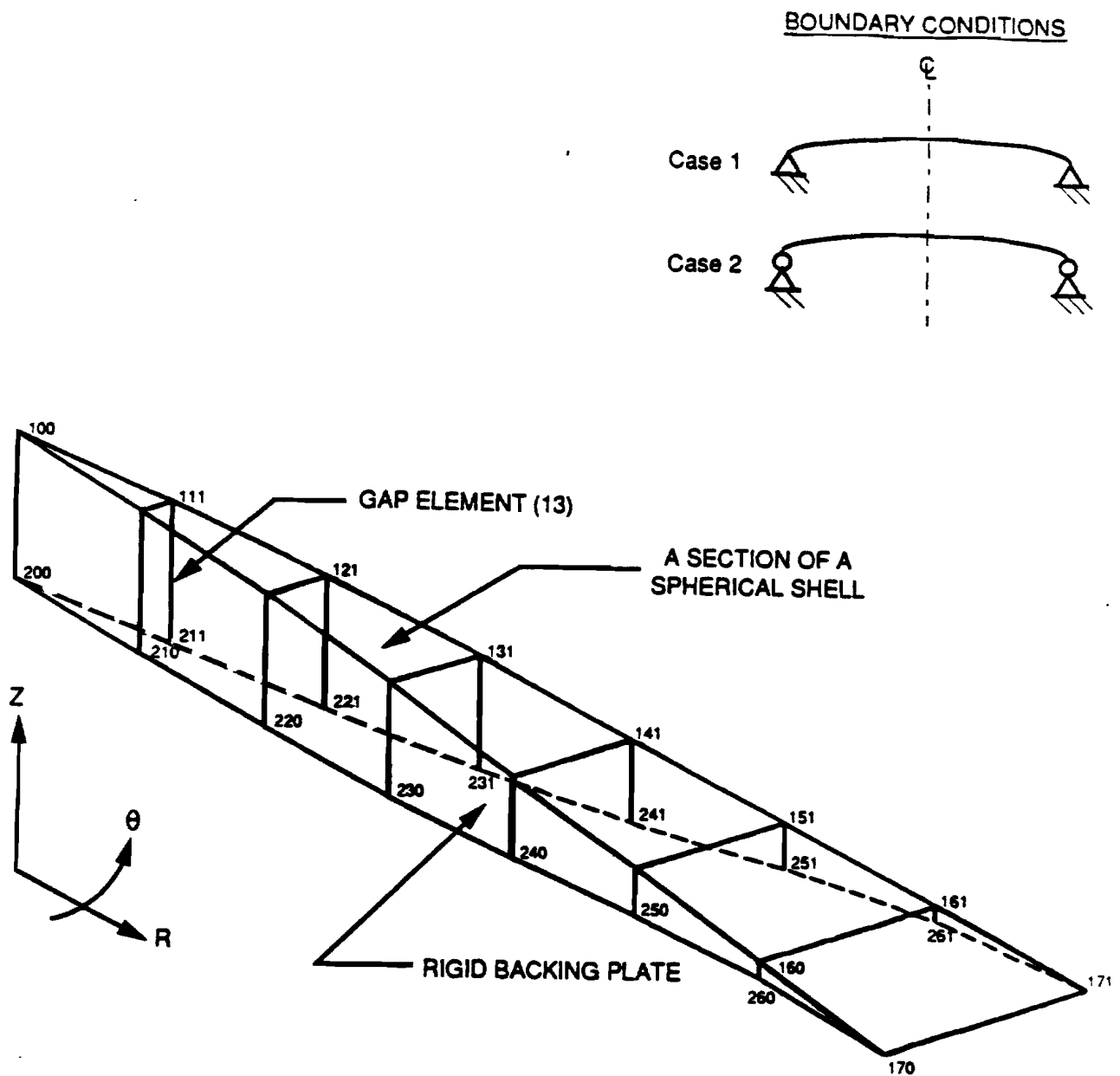


Figure 11.1.2. Spherical Diaphragm Model with Axisymmetric Boundary Condition and Constraints by Gap Elements

Table 11.1.1 Input Data for the Clamped Case

```

ID SNAPC,V66 $ SHL 5/2/85, GBL 11/22/86
TIME 30
SOL      66
DIAG 8,50
CEND
TITLE    = SNAP-THROUGH ANALYSIS OF A SPHERICAL DOME
SUBTI    = EDGE CLAMPED
ECHO     = UNSORT
SEALL    = ALL
SPC      = 200
  SET 1 = 100 110 120 130 140 150 160
  SET 2 = 1 THRU 7
  SET 3 = 170,200
  DISPL = 1
  SPCFO = 3
SUBCASE 1
  SUBTI = 0.90 PSI
  NLPAR = 1
  LOAD  = 90
SUBCASE 2
  SUBTI = 1.0 PSI
  NLPAR = 2
  LOAD  = 100
SUBCASE 3
  SUBTI = 1.5 PSI
  NLPAR = 3
  LOAD  = 150
SUBCASE 4
  SUBTI = 0.6 PSI
  NLPAR = 4
  LOAD  = 60
SUBCASE 5
  SUBTI = 0.35 PSI
  NLPAR = 5
  LOAD  = 35
SUBCASE 6
  SUBTI = 0.0 PSI
  NLPAR = 6
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  PAPER SIZE 29. X 21.
  SET 1 ALL
  SET 2 ALL EXCEPT GAP
  MAXI DEFO .001
  FIND SCALE ORIGIN 1 SET 1
  PTITLE=ISOMETRIC VIEW
  PLOT SET 1, ORIGIN 1, LABEL
  VIEW 0. , 0. , 0.
  AXES MY , X , Z
  FIND SCALE ORIGIN 1 SET 2

```



```

PTITLE=ORTHOGRAPHIC VIEW
PLOT STATIC 0 MAXI DEFO .001 SET 2 SYMBOL 1
BEGIN BULK
$ PARAMETERS
PARAM K6ROT 1.0
PARAM LGDISP 1
NLPARM 1 6 AUTO YES +NL1
NLPARM 2 5 SEMI 2 YES +NL2
+NL2 5
NLPARM 3 2 AUTO YES +NL3
+NL3 5
NLPARM 4 3 AUTO YES +NL4
NLPARM 5 5 AUTO YES +NL5
+NL5 5
NLPARM 6 1 AUTO YES +NL6
$ BOUNDARY CONDITIONS
SPC1 200 12456 100
SPC1 200 246 110 120 130 140 150 160
SPC1 200 246 111 121 131 141 151 161
SPC1 200 123456 170 171
$ LOADING
PLOAD2 35 .35 1 THRU 7
PLOAD2 60 .60 1 THRU 7
PLOAD2 90 .90 1 THRU 7
PLOAD2 100 1.0 1 THRU 7
PLOAD2 150 1.5 1 THRU 7
$
$ GEOMETRY
$
CORD2C 1 0.0 0.0 0.0 0.0 0.0 1.0 +C1
+C1 1.0 0.0 0.0
$ GRID POINTS
GRID 100 1 0.0 0.0 .503 1 6
GRID 110 1 1.082 0.0 .486 1 6
GRID 111 1 1.082 10. .486 1 6
GRID 120 1 2.163 0.0 .445 1 6
GRID 121 1 2.163 10. .445 1 6
GRID 130 1 3.243 0.0 .393 1 6
GRID 131 1 3.243 10. .393 1 6
GRID 140 1 4.322 0.0 .315 1 6
GRID 141 1 4.322 10. .315 1 6
GRID 150 1 5.401 0.0 .238 1 6
GRID 151 1 5.401 10. .238 1 6
GRID 160 1 6.479 0.0 .153 1 6
GRID 161 1 6.479 10. .153 1 6
GRID 170 1 7.557 0.0 .064 1 6
GRID 171 1 7.557 10. .064 1 6
$ ELEMENTS
CTRIA3 1 1 100 111 110
CQUAD4 2 1 110 111 121 120
CQUAD4 3 1 120 121 131 130
CQUAD4 4 1 130 131 141 140
CQUAD4 5 1 140 141 151 150
CQUAD4 6 1 150 151 161 160
CQUAD4 7 1 160 161 171 170

```

\$ BACKUP OF GAPS

GRID	200	1	0.0	0.0	-.254	1	123456
GRID	210	1	1.082	0.0	-.2533	1	123456
GRID	211	1	1.082	10.	-.2533	1	123456
GRID	220	1	2.163	0.0	-.2354	1	123456
GRID	221	1	2.163	10.	-.2354	1	123456
GRID	230	1	3.243	0.0	-.1862	1	123456
GRID	231	1	3.243	10.	-.1862	1	123456
GRID	240	1	4.322	0.0	-.1062	1	123456
GRID	241	1	4.322	10.	-.1062	1	123456
GRID	250	1	5.401	0.0	-.1800-21		123456
GRID	251	1	5.401	10.	-.1800-21		123456
GRID	260	1	6.479	0.0	.0697	1	123456
GRID	261	1	6.479	10.	.0697	1	123456

\$ GAPS TO REPRESENT BACKING PLATE

CGAP	100	200	200	100	1.
CGAP	110	210	210	110	1.
CGAP	111	210	211	111	1.
CGAP	120	220	220	120	1.
CGAP	121	220	221	121	1.
CGAP	130	230	230	130	1.
CGAP	131	230	231	131	1.
CGAP	140	240	240	140	1.
CGAP	141	240	241	141	1.
CGAP	150	250	250	150	1.
CGAP	151	250	251	151	1.
CGAP	160	260	260	160	1.
CGAP	161	260	261	161	1.

\$ PROPERTIES

MAT1	1	2.+5		.28	7.835-9	
PSHELL	1	1	0.15	1	1.2	1 0.833
PGAP	200	.7570		1.+6	1.-6	
PGAP	210	.7393		1.+6	1.-6	
PGAP	211	.7393		1.+6	1.-6	
PGAP	220	.6804		1.+6	1.-6	
PGAP	221	.6804		1.+6	1.-6	
PGAP	230	.5792		1.+6	1.-6	
PGAP	231	.5792		1.+6	1.-6	
PGAP	240	.4212		1.+6	1.-6	
PGAP	241	.4212		1.+6	1.-6	
PGAP	250	.2398		1.+6	1.-6	
PGAP	251	.2398		1.+6	1.-6	
PGAP	260	.0833		1.+6	1.-6	
PGAP	261	.0833		1.+6	1.-6	

\$ PLOT DATA

PLOTEL	11	200	210
PLOTEL	12	210	220
PLOTEL	13	220	230
PLOTEL	14	230	240
PLOTEL	15	240	250
PLOTEL	16	250	260
PLOTEL	17	260	170

ENDDATA

Table 11.1.2 Partial Listing of the Input Data for the Simply Supported Case

```

ID SNAPS,V65 $ SHL 5/2/85, GBL 12/10/86
TIME 30
SOL 66
DIAG 8,50
CEND
TITLE = SNAP-THROUGH ANALYSIS OF A SPHERICAL DOME
SUBTI = EDGE SIMPLY SUPPORTED
SEALL = ALL
ECHO = UNSORT
SPC = 200
SET 1 = 100 110 120 130 140 150 160
SET 2 = 1 THRU 7
SET 3 = 170,200
DISPL = 1
SPCFO = 3
OLOAD = 3
SUBCASE 1
SUBTI = 0.5 PSI
NLPAR = 1
LOAD = 50
SUBCASE 2
SUBTI = 0.6 PSI, first snap-through
NLPAR = 2
LOAD = 60
SUBCASE 3
SUBTI = 0.7 PSI
NLPAR = 3
LOAD = 70
SUBCASE 4
SUBTI = 0.72 PSI, second snap-through
NLPAR = 4
LOAD = 72
SUBCASE 5
SUBTI = 0.8 PSI
NLPAR = 5
LOAD = 80
SUBCASE 6
SUBTI = 0.0 PSI, unloading
NLPAR = 6
SUBCASE 7
SUBTI = 0.0 PSI (EDGE CONSTRAINT REMOVED)
SET 5 = 100 110 120 130 140 150 160 170
DISPL = 5
SPC = 300
NLPAR = 7
OUTPUT(PLOT)
CSCALE 1.3
PLOTTER NAST
PAPER SIZE 29. X 21.

```

```

SET 1 ALL
SET 2 ALL EXCEPT GAP
MAXI DEFO .001
FIND SCALE ORIGIN 1 SET 1
  PTITLE=ISOMETRIC VIEW
PLOT SET 1, ORIGIN 1, LABEL
VIEW 0. , 0. , 0.
AXES MY , X , Z
FIND SCALE ORIGIN 1 SET 2
  PTITLE=ORTHOGRAPHIC VIEW
PLOT STATIC 0 MAXI DEFO .001 SET 2 SYMBOL 1
BEGIN BULK
$ PARAMETERS
PARAM K6ROT 1.0
PARAM LGDISP 1
NLPARM 1 5 AUTO YES +NL1
NLPARM 2 5 AUTO 2 YES +NL2
+NL2 7 2
NLPARM 3 4 AUTO YES +NL3
NLPARM 4 4 SEMI 2 YES +NL4
+NL4 5 10
NLPARM 5 4 AUTO YES +NL5
NLPARM 6 8 AUTO YES +NL6
NLPARM 7 10 AUTO YES +NL7
$ BOUNDARY CONDITIONS
SPC1 200 12456 100
SPC1 200 246 110 120 130 140 150 160
SPC1 200 246 111 121 131 141 151 161
SPC1 200 12346 170 171
SPC1 300 12456 100
SPC1 300 246 110 120 130 140 150 160
SPC1 300 246 111 121 131 141 151 161
SPC1 300 2346 170 171
$ LOADING
PLOAD2 50 .50 1 THRU 7
PLOAD2 60 .60 1 THRU 7
PLOAD2 70 .70 1 THRU 7
PLOAD2 72 .72 1 THRU 7
PLOAD2 80 .80 1 THRU 7
$
$ GEOMETRY
$
*** The rest is identical to Table 11.1.1 ***

```

11.1.3 Analysis Results

Clamped Case

The load-deflection curve at the center point is shown in Fig. 11.1.3. The curve is constructed by combining loading and unloading paths. The snap-through occurs between 0.96 psi and 0.98 psi, and the snap-back between 0.45 psi and 0.40 psi. Since the center point comes into contact with the backing plate at $p = 0.98$ psi, the point does not translate beyond the gap distance (0.757 in.) despite the increasing load. The load history between two points A and B, where the snap-through and the snap-back begin to occur, is conjectured as shown by the dashed line in Fig. 11.1.3.

Fig. 11.1.4 shows the deformed shapes of the diaphragm for various loading and unloading levels. It is shown that the curvatures of the diaphragm surface are reversed upon snap-through or snap-back at the critical loads.

Simply Supported Case

In the simply supported case, snap-through occurs twice as shown in Fig. 11.1.5. The first snap-through occurs between 0.54 psi and 0.56 psi, and the second occurs between 0.700 psi and 0.705 psi. If it is unloaded before the second snap-through, the diaphragm snaps back at a pressure between 0.50 psi and 0.48 psi. However, the snap-back does not occur when the pressure is removed after the second snap-through as shown.

As shown in Fig. 11.1.6(a), two steps of snap-through assume distinct shapes. The first snap-through can be considered as an interim snap-through by flipping into a shape of double curvature. Closer examination of the analysis results reveals that the diaphragm comes into contact with the backing plate near the periphery after the first snap-through. In fact, such contacts stabilized the interim snap-through mode, which would not have existed otherwise. The central portion of the diaphragm comes into contact with the backing plate in addition to the peripheral area upon the second snap-through.

The central area remains in contact with the backing plate during unloading until the pressure is reduced to 0.1 psi. When the pressure is removed entirely, the central area is lifted off, leaving the peripheral area of the diaphragm (point A in Fig. 11.1.6(b)) in contact with the backing plate. Based on these observations, the load history between points A and B can be conjectured by the dashed line in Fig. 11.1.5.

It is interesting that the snap-back does not occur upon unloading in the elastic problem. This is because the spring-back force is trapped by SPC forces at the periphery. This SPC force (Q) upon full unloading is computed as 19.9 lbs on each node at the periphery, which is equivalent to a reaction force of 15.088 lbs/in. in the radial direction. It is attempted to release this reaction force by removing radial constraints on the periphery. The SPC force of

19.9 lbs is released in ten steps by Subcase 7. As shown in Fig. 11.1.6(b), the diaphragm snaps back as the reaction force is released. When the reaction force is removed entirely at the end, the diaphragm resumes the original shape as shown in the figure.

Fig. 11.1.7 shows the vertical displacement at the center and the radial displacement at the periphery as functions of the SPC force. The snap-back occurs when the SPC force is between 15.9 lbs and 13.9 lbs. Upon snap-back, the central displacement overshoots due to the radial compression force remaining at the periphery. The periphery also slides back and forth radially as the SPC force is diminished.

Sliding and Simply Supported Case

As the pressure is increased, the diaphragm undergoes a slight snap-through at a pressure between 0.188 psi and 0.190 psi. Fig. 11.1.8 shows the load-deflection curve at the center point. As shown, the snap-back occurs at a pressure between 0.184 psi and 0.182 psi.

11.1.4 Solution Progression

The convergence of the solution is the most important issue for the iterative and incremental solution processes of nonlinear problems. The convergence characteristics are mainly influenced by the increment size and the iteration strategy. These crucial data for controlling the incremental and iterative processes are specified in the Bulk Data NLPARM.

The general strategy is to start with default options and reasonable increments of the applied load. Nonlinear analysis requires restarts in general. If the solution has not converged at a certain step, restarts are attempted from that step with different options in the NLPARM. Major options that can be changed for better convergence are increment size and parameters KMETHOD, MAXDIV, and MAXLS. It is recommended that the value of MAXDIV be increased (5 to 7) for the subcase where snap-through or snap-back is anticipated. The increment size was also reduced in the vicinity of the snap-through.

It was discovered in this study that the effect of the line search process is rather erratic and unpredictable around the critical loads. The line search was helpful to jump over the unstable region in most cases, but on some occasions, the solution was more easily converged by suppressing the line search. This behavior was scrutinized and the following observations were contemplated:

- The convergence characteristics around the critical load (snap-through or snap-back) are very sensitive to the timing of the stiffness matrix update.
- The timing and the line search affect the base vector (displacements) for the matrix update.

- A slight difference in the base vector for the update could lead the solution to either side of divergence or convergence in the vicinity of the critical loads without due explanation.

Clamped Case

Table 11.1.3 shows the solution strategy and the performance of each subcase when the solutions are obtained successfully. Subcases 1, 2, and 3 represent the loading phase, and Subcases 4, 5, and 6 represent the unloading phase. The snap-through occurs in Subcase 2, and the snap-back occurs in Subcase 5. The solution has not converged with default options in the vicinity of the critical loads. Restart runs were made with smaller load increments and different iteration strategies until converged solutions were achieved. It is noted that more stiffness updates, iterations, and/or line searches were required in these subcases to obtain the solution than in other subcases without an unstable region. In subcases without an unstable region, solutions were easily achieved by default options.

Table 11.1.4 summarizes trial-and-errors attempted for Subcase 2/increment 4 ($p = 0.96$ psi - 0.98 psi), in which snap-through occurs. In the first trial, the program detects a negative term on the factor diagonal of $[K]$, and the problem stops due to TWO DIVERGING SOLUTIONS after 11 iterations with 3 stiffness updates. In the second and third trials, the problem does not converge due to the same causes. In the fourth trial, the problem stops before convergence due to the maximum iteration limit of 20. Finally, when KMETHOD is set to SEMI, MAXDIV to 5, and KSTEP to 2, the problem converges after 16 iterations with 6 stiffness updates. It can be noticed that frequent stiffness updates at converged states (due to KSTEP=2) expedite convergence.

Table 11.1.3 Solution Strategy and Performance for Clamped Case

Subcase ID	Load (psi)	No. of INC	Solution Strategy	No. of K Update	No. of Iter	No. of L.S.
1	0.90	6	Default	1	29	0
2*	1.00	5	SEMI, KSTEP=2 MAXDIV= 5	12	29	41
3	1.5	2	MAXDIV= 5	1	11	16
4	0.60	3	Default	2	18	1
5*	0.35	5	MAXDIV= 5	4	35	16
6	0.00	1	Default	0	6	0

Note: Snap occurs at subcases denoted by *

**Table 11.1.4 Effects of NLPARM Parameters on Convergence
for Subcase 2/Increment 4 of Clamped Case**

Trial Case	Parameters				No. of K Update	No. of Iter	No. of L.S.
	KMETHOD	KSTEP	MAXDIV	MAXLS			
1	(AUTO)	(7)	(2)	(5)	3	11	36
2	(AUTO)	(7)	5	(5)	5	19	48
3	SEMI	(7)	5	(5)	2	6	6
4	SEMI	(7)	7	(5)	5	20	45
5	SEMI	2	5	(5)	6	16	41

Note: 1. () denotes default option
2. Successful solution is achieved at the final trial

Simply Supported Case

Table 11.1.5 shows the solution strategy and the performance of each subcase upon successful solutions. Subcases 1 through 5 represent the loading phase and subcase 6 represents the unloading phase. After the pressure is removed entirely, the radial constraint of the diaphragm is released, which is analyzed by Subcase 7. Another set, Subcase 3, 4, and 5 denoted by +, analyzed the unloading path from 0.60 psi after the first snap-through occurs. The solution requires strategies different from the default options for NLPARM in the vicinity of the critical loads.

Tables 11.1.6 and 11.1.7 show the trial-and-errors to obtain solutions for the first (Subcase 2/increment 3) and the second snap-through (Subcase 4/increment 1), respectively. The first 3 trials in Table 11.1.6 were terminated due to TWO DIVERGING SOLUTIONS.

Table 11.1.7 shows two trials terminated due to the maximum iteration limit of 20, where the program detects a negative term on the factor diagonal of [K] as well as the GIMBAL ANGLE exceeding 90 degrees. It is noted that variations of MAXDIV and MAXLS have significant effects on the performance while jumping over the unstable region for the snap-through analysis.

Table 11.1.5 Solution Strategy and Performance for Simply Supported Case

Subcase ID	Load (psi)	No. of INC	Solution Strategy	No. of K Update	No. of Iter	No. of L.S.
1	0.50	5	Default	1	24	0
2*	0.60	5	KSTEP=2 MAXDIV=7 MAXLS=2	7	25	8
3	0.70	4	Default	0	17	0
4*	0.72	4	SEMI,KSTEP=2 MAXDIV=5 MAXLS=10	10	25	87
5	0.80	4	Default	0	8	0
6	0.00	8	Default	2	30	3
7*	0.00	10	Default	7	75	18
3+	0.50	4	Default	0	12	0
4+*	0.40	5	KSTEP=2 MAXDIV=7	8	27	11
5+	0.00	5	Default	1	24	0

Notes: 1. Snap occurs at subcases denoted by *
2. + denotes restart for snap-back during unloading

Table 11.1.6 Effects of NLPARM Parameters on Convergence for Subcase 2/Increment 3 of Simply Supported Case

Trial Case	Parameters				No. of K Update	No. of Iter	No. of L.S.
	KMETHOD	KSTEP	MAXDIV	MAXLS			
1	(AUTO)	(7)	(2)	(5)	1	5	5
2	(AUTO)	2	5	(5)	2	8	24
3	(AUTO)	2	5	2	2	8	12
4	(AUTO)	2	7	2	3	11	8

Notes: 1. () denotes default option
2. Successful solution is achieved at the final trial

**Table 11.1.7 Effects of NLPARM Parameters on Convergence
for Subcase 4/Increment 1 of Simply Supported Case**

Trial Case	Parameters				No. of K Update	No. of Iter	No. of L.S.
	KMETHOD	KSTEP	MAXDIV	MAXLS			
1	(AUTO)	(7)	(2)	(5)	3	12	12
2	(AUTO)	2	7	2	6	20	20
3	(AUTO)	2	7	0	6	20	0
4	SEMI	2	5	10	7	19	87

Note: 1. () denotes default option
2. Successful solution is achieved at the final trial

Sliding and Simply Supported Case

As shown in Table 11.1.8, nine subcases are used for this analysis. Subcases 1 through 5 represent the loading phase, and Subcases 6 through 9 the unloading phase. The snap-through occurs in Subcase 3, and the snap-back occurs in Subcase 8. Since the snap-through and the snap-back in this case are moderate and shallow as shown in Fig. 11.1.8, the load is incremented in very small steps around the critical load to obtain a more descriptive load-deflection curve. In fact, the coarser load increment rendered solutions by default options without giving details of snap-through around the critical load.

**Table 11.1.8 Solution Strategy and Performance for
Sliding and Simply Supported Case**

Subcase ID	Load (psi)	No. of INC	Solution Strategy	No. of K Update	No. of Iter	No. of L.S.
1	0.16	4	Default	1	24	0
2	0.18	1	Default	0	5	0
3*	0.19	5	MAXDIV=5	4	32	18
4	0.20	1	Default	0	4	0
5	0.60	4	Default	2	25	1
6	0.35	1	Default	0	6	0
7	0.19	1	Default	1	9	2
8*	0.18	5	Default	3	29	14
9	0.00	2	Default	2	8	16

Note: Snap occurs at subcases denoted by *

11.1.5 Concluding Remarks

While default options in NLPARM are effective as a general strategy for ordinary problems, some variations are required to overcome difficulties if divergence is encountered. It is discovered that such variations are required to jump over the unstable region of the snap-through and snap-back problems.

In general, more conservative methods are more reliable but also expensive. Helpful variations for snap-through problems are

- SEMI rather than AUTO
- Smaller value for KSTEP to allow more frequent stiffness updates at convergence
- Larger value for MAXDIV to bypass the divergence criteria
- Larger value for MAXLS to exercise more line searches

It is also observed that the solutions to the snap-back are easier to achieve than to the snap-through.

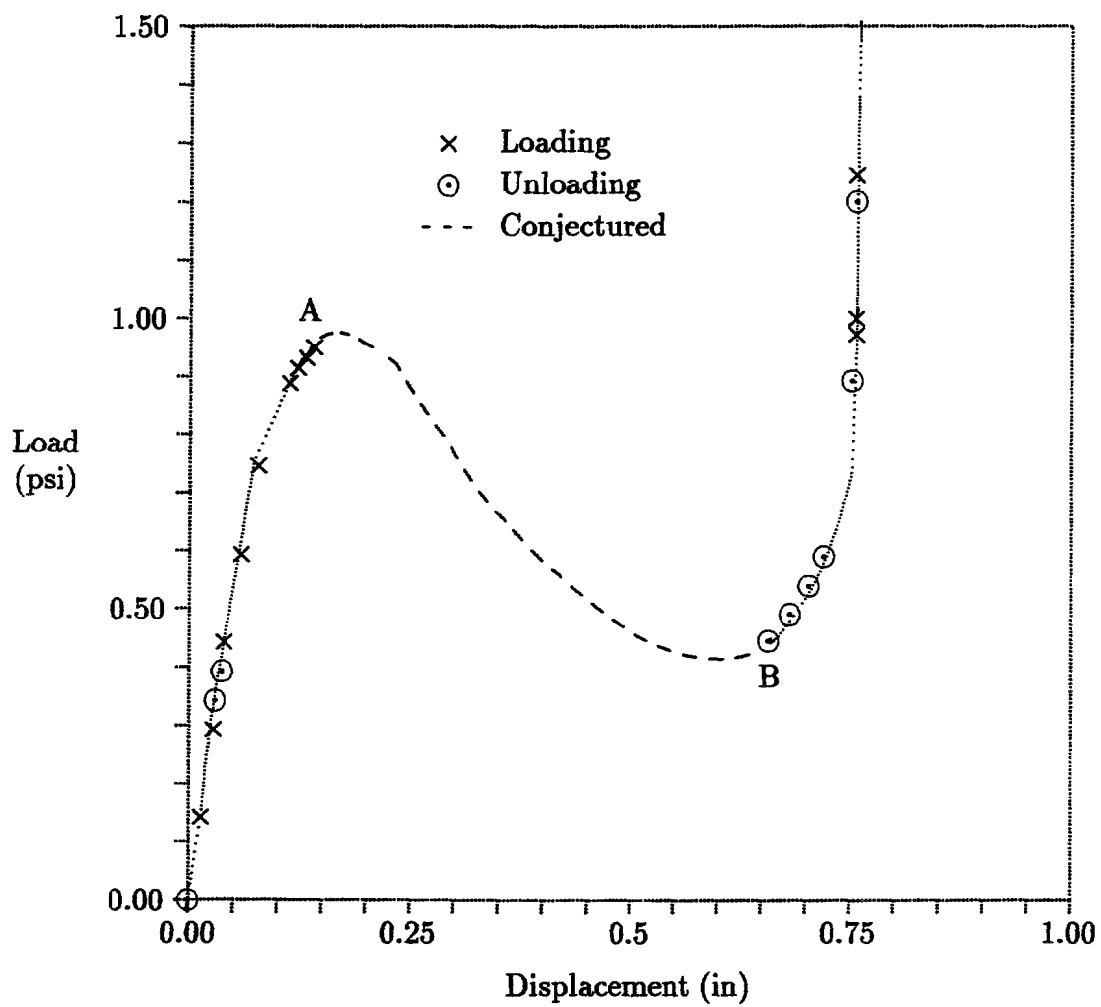
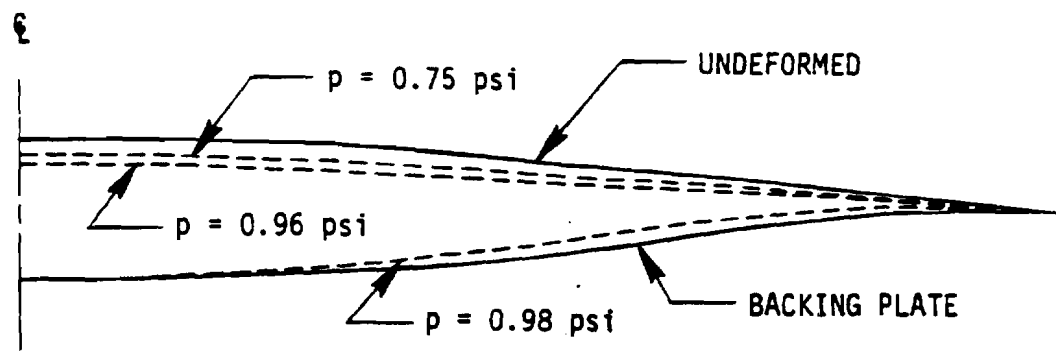
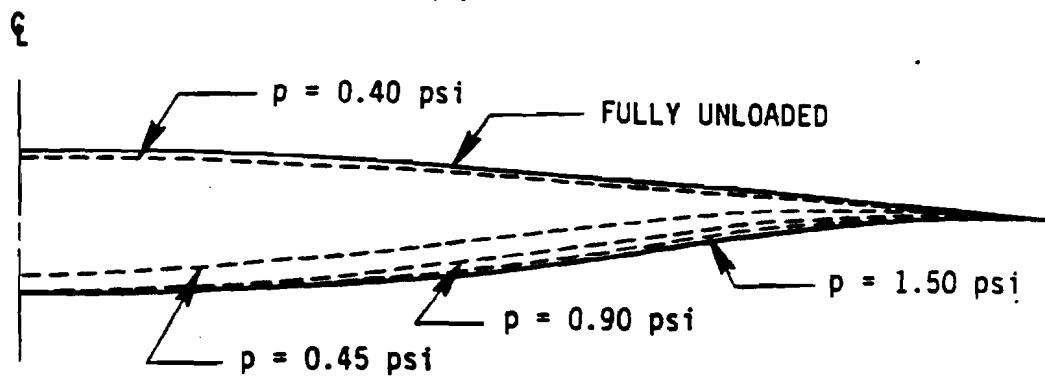


Figure 11.1.3 Load vs. Vertical Displacement at the Center (Clamped Case).



(a)



(b)

Figure 11.1.4. Deformed Shapes of the Diaphragm (Clamped Case)
(a) Loading (b) Unloading

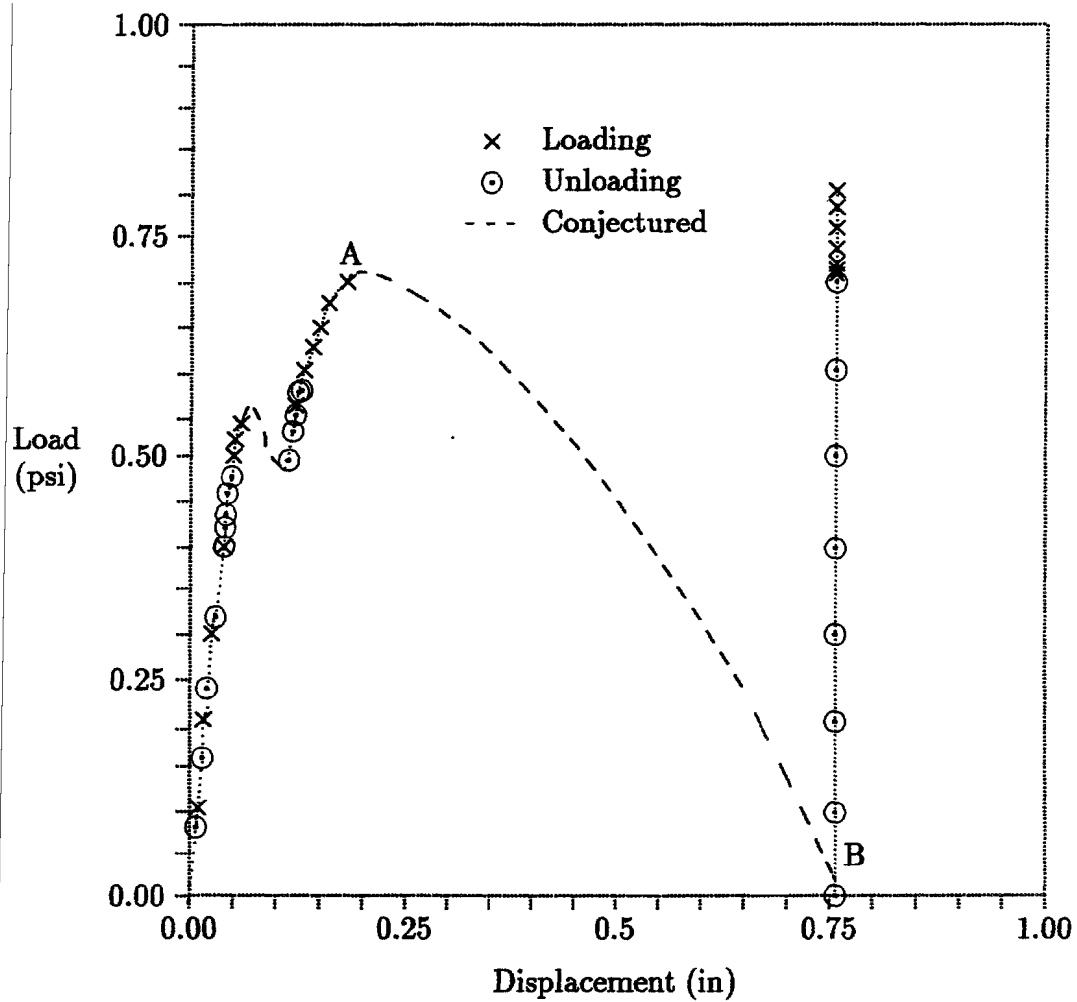
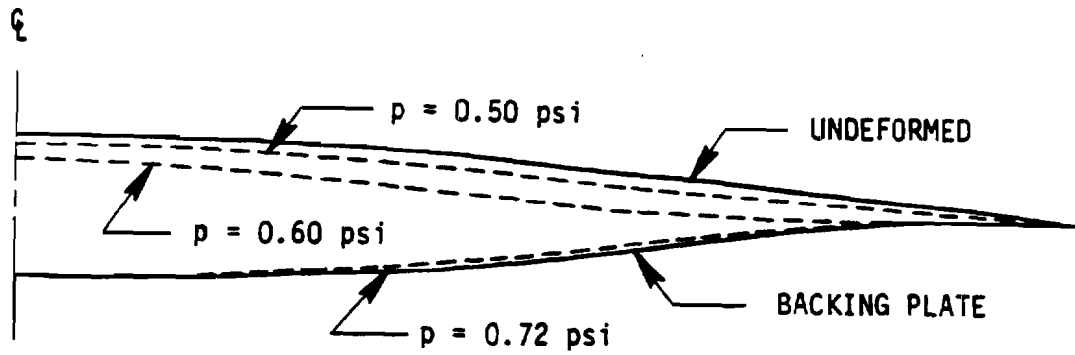
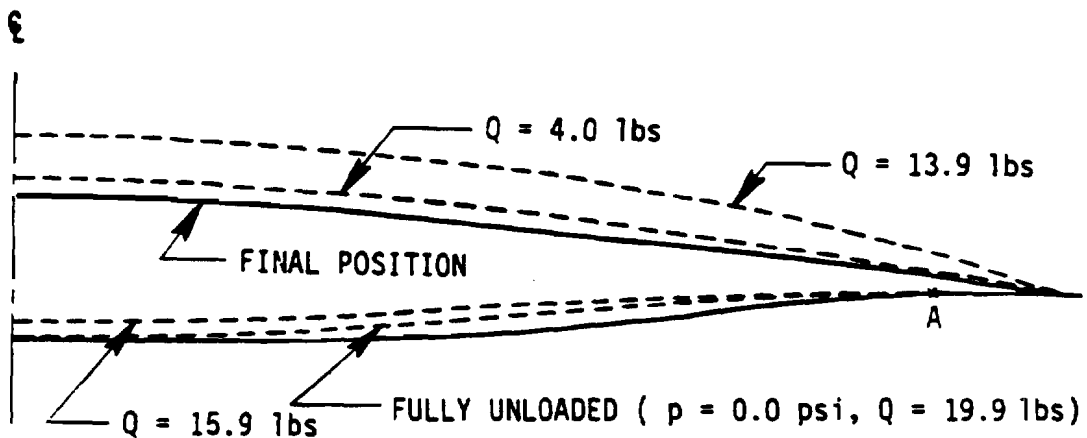


Figure 11.1.5 Load vs. Vertical Displacement at the Center
(Simply Supported Case).



(a)



(b)

Figure 11.1.6. Deformed Shapes of the Diaphragm (Simply Supported Case) (a) Loading (b) Edge Constraint Removed after Unloading

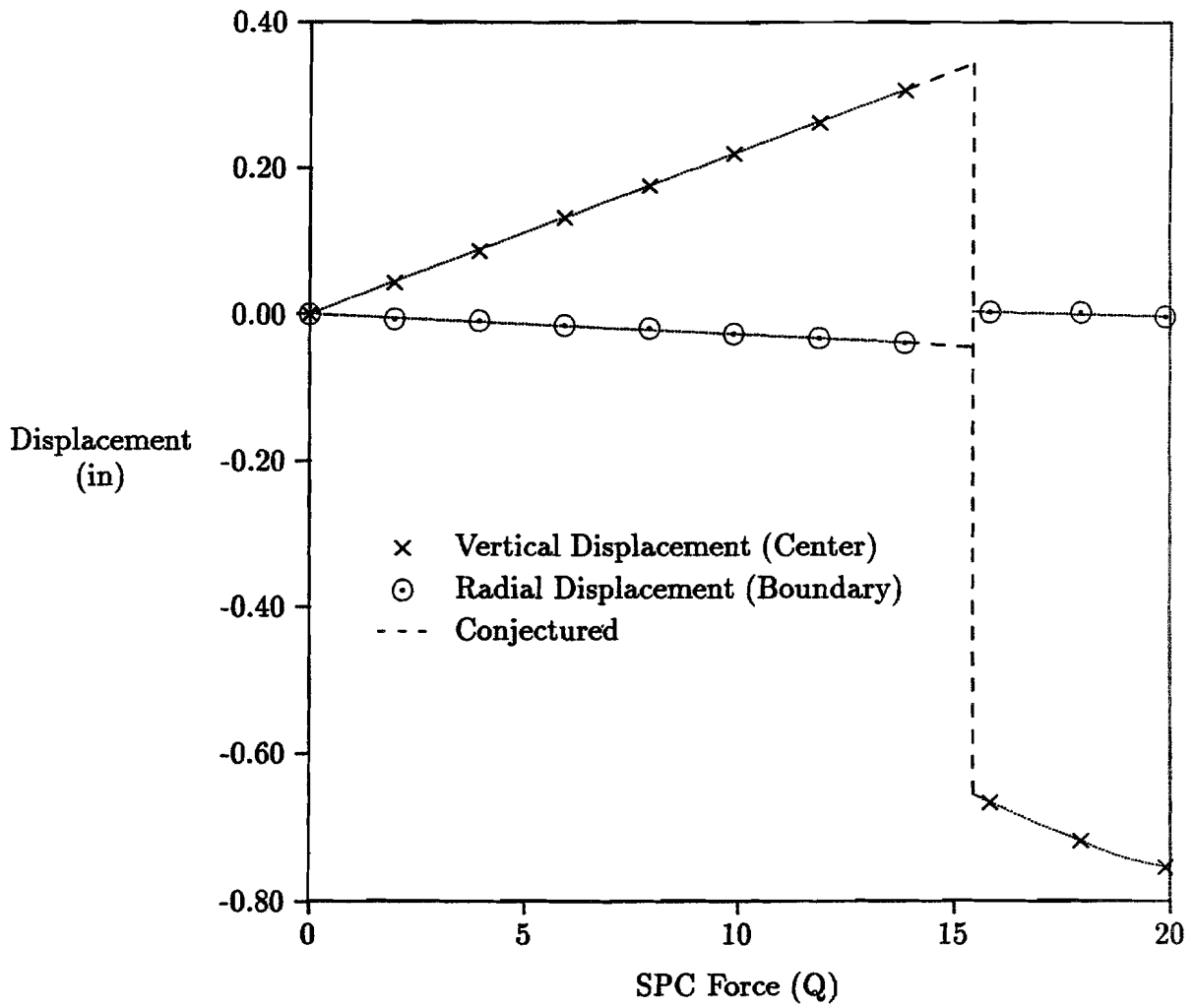


Figure 11.1.7 Vertical and Radial Displacements at the Center and Boundary Points vs. SPC Forces at the Periphery.

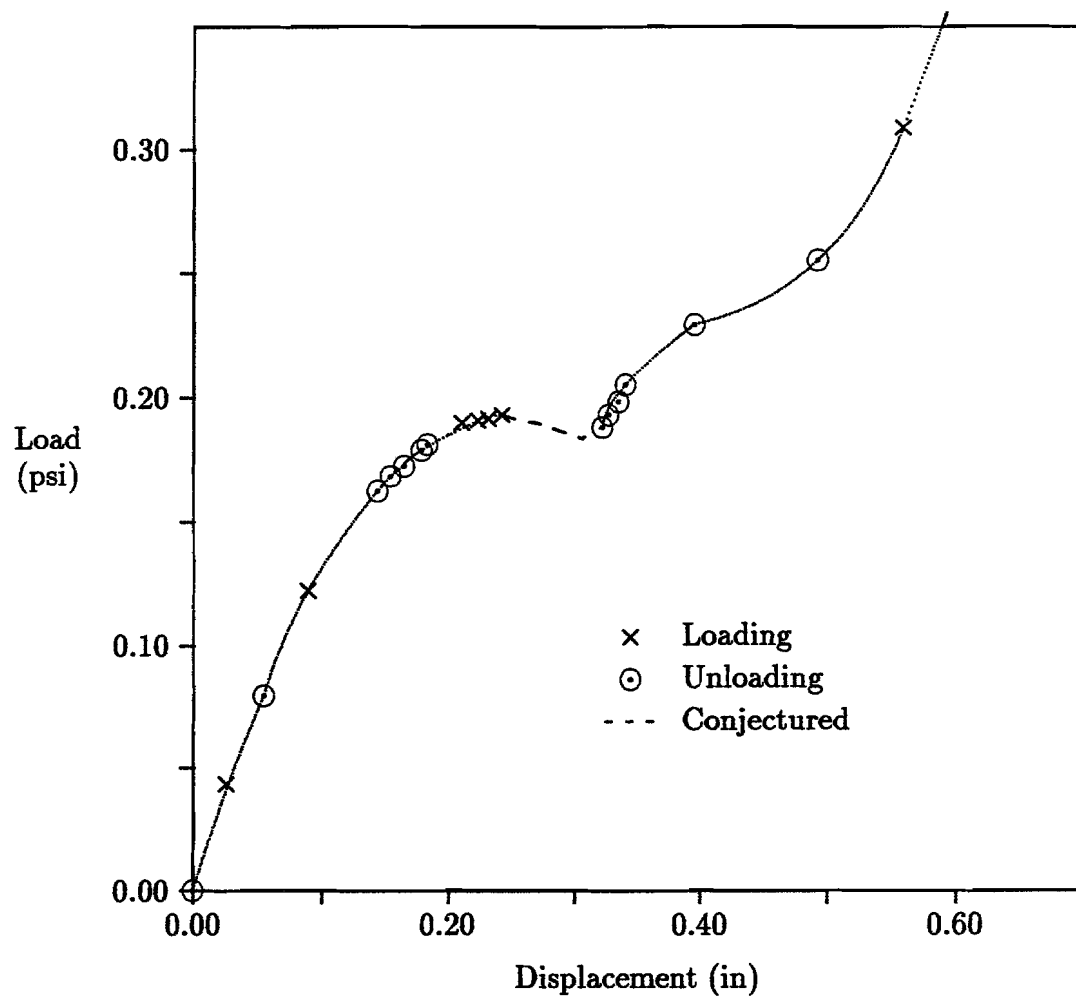


Figure 11.1.8 Load vs. Vertical Displacement at the Center
(Sliding and Simply Supported Case).

11.2 PARAMETRIC STUDY ON BFGS UPDATES AND LINE SEARCH METHOD IN SOL 66

11.2.1 Introduction

The quasi-Newton update and the line search methods are employed as options to accelerate the convergence and improve the overall effectiveness of the modified Newton's iteration. When the BFGS method was implemented and tested in the software, our experience showed that the implementation scheme and adjustment of parameters had a significant impact on the program's performance [11.1]. The line search method is an expedient, but not an expeditious method that could improve the computational efficiency. The line search method is modified and implemented to optimize its usefulness for a general class of problems in conjunction with quasi-Newton updates [11.2].

Parametric studies are performed to validate the effectiveness and efficiency of the algorithm. The study shows that the line search method is essential to achieve convergence in some highly nonlinear problems, but has little impact on the efficiency.

Various problems are examined to verify the algorithm's ability to handle nonlinear problems involving softening/hardening and buckling behavior with material/geometric nonlinearities. The effectiveness of the solution process is studied for different values of the fields MAXQN, MAXLS, and LSTOL in the NLPARM Bulk Data entry. Three example problems are presented to illustrate the convergence and efficiency characteristics of the quasi-Newton update and the line search process. The analyses are performed using Version 65. The iteration strategy is based on the **hybrid Newtons method, by which the stiffness is updated only when it is necessary for efficiency and effectiveness** [11.3]. Two cases are considered for each problem. In the first case, fields MAXLS and LSTOL are kept constant (equal to 5 and 0.5, respectively, unless otherwise stated), and field MAXQN is varied. In the second case, field MAXQN is kept constant (equal to 30), and fields MAXLS and LSTOL are varied.

11.2.2 The Cologne Challenge: a Z-Shaped Cantilever Beam

A Z-shaped cantilever beam as shown in Fig. 11.2.1 was proposed as a benchmark problem for testing nonlinear capabilities of FEM codes [11.4]. The beam, when subjected to the vertical load at the end, undergoes very large displacement and plastic deformation with von Mises yield criterion and isotropic hardening [11.5]. Two different models are used in the present analysis: a plate element model (ZPLATE) and a beam element model (ZBEAM). The analysis is difficult because of the highly nonlinear behavior resulting from the large rotation at the plastic hinge as shown in Fig. 11.2.2. The load-deflection curve at the end point is shown in Fig. 11.2.3. Notice that the nonlinearity of the system in the loading phase of 0–1000 N is much more severe than that of 1000–2000 N. This makes the problem suitable for examining the performance of nonlinear iteration algorithms in both highly and slightly nonlinear regions.

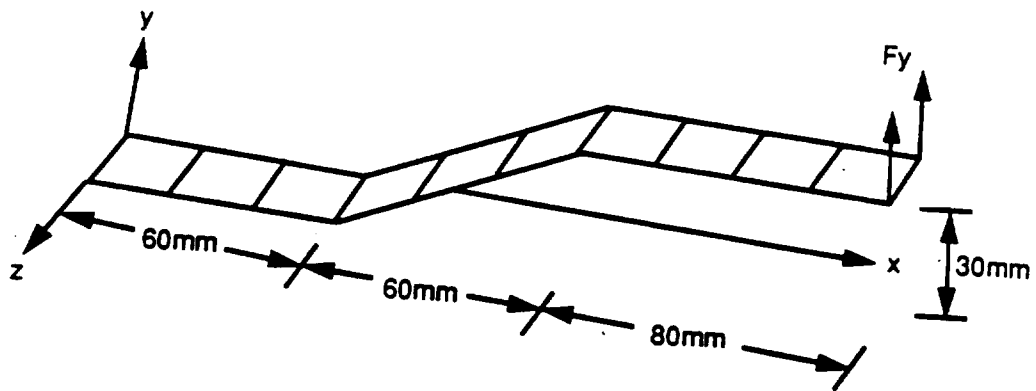


plate thickness: 3.4 mm
plate width : 20 mm
tip load : 2000 N

Figure 11.2.1. Cantilevered Z-Shaped Beam (QUAD4 Model).

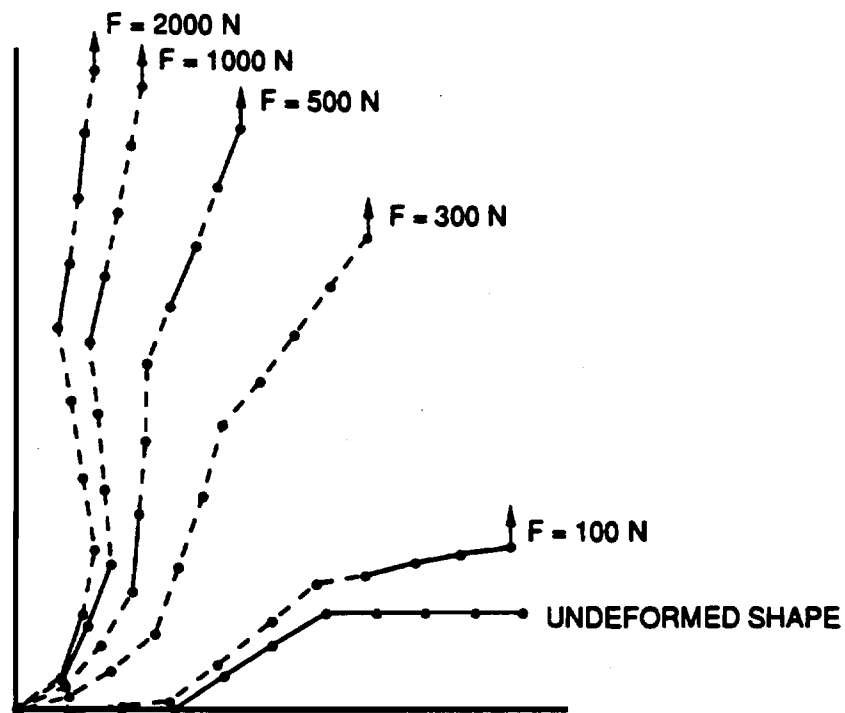


Figure 11.2.2. Deformed Shapes (QUAD4 model).

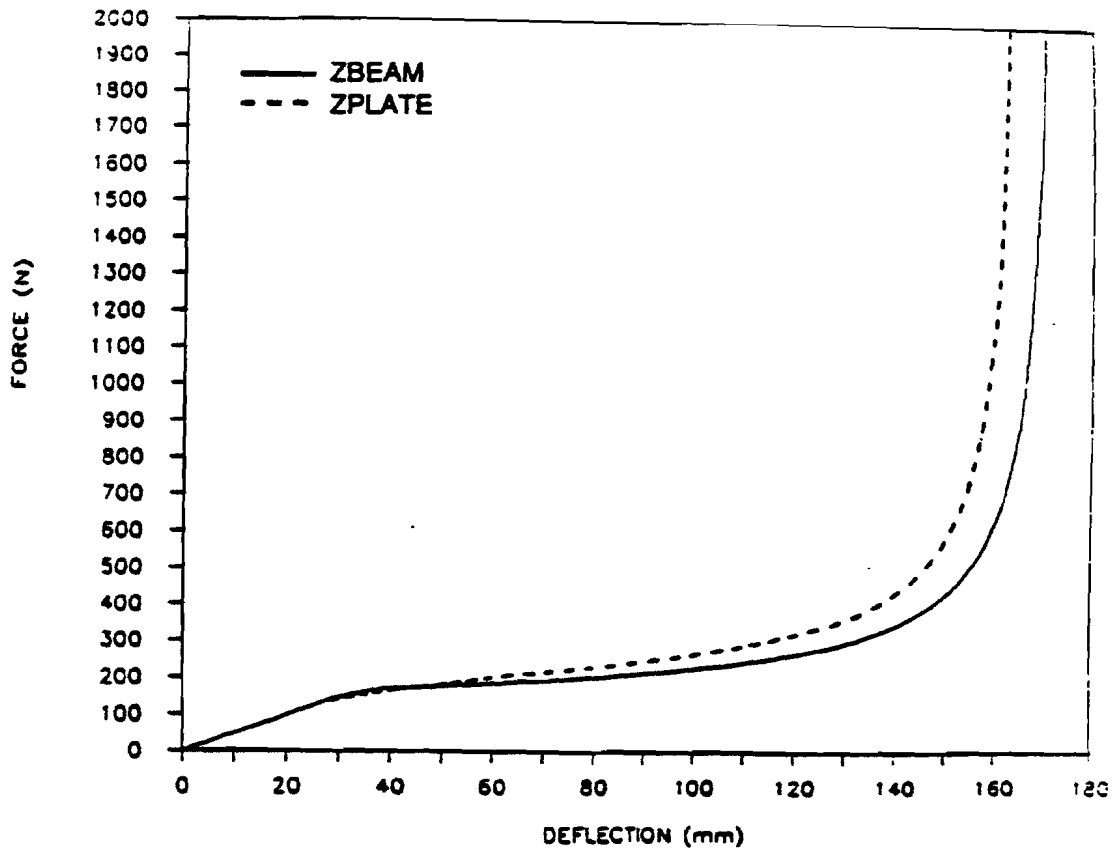


Figure 11.2.3. Load-Deflection Curve.

Case 1: Tables 11.2.1(a) and 11.2.2(a) summarize the numerical performance of the BFGS updates for ZPLATE and ZBEAM, respectively, by varying MAXQN from 0 to 30 with an interval of 5. The number of QN updates reflects the cumulative QN updates counting all the quasi-Newton vector pairs involved in every iteration. By increasing the value of MAXQN, more QN updates are performed. When the QN update is suppressed (MAXQN=0), the beam model fails to converge, whereas the plate model requires an excessive number of line searches, stiffness updates and iterations for convergence. In terms of efficiency, both models seemed to show most favorable results with a MAXQN of 20.

Table 11.2.1(a). Numerical Performance as a function of MAXQN in ZPLATE

LOAD (N)	COMPUTATION EFFICIENCY	MAXQN						
		0	5	10	15	20	25	30
0 – 1000 (130 increments) severe nonlinearity	No. of QN updates	0	4191	4255	4995 ⁺	4526	4476	4910
	No. of line searches	503	327	260	256	259	252	258
	No. of iterations	1412	1326	1196	1195	1212	1158	1196
	No. of K updates	225	148	147	140	138	141	140
1000 – 2000 (100 increments) mild nonlinearity	No. of QN updates	0	2329	1462	2288	3882	4971	5537
	No. of line searches	8	0	0	1	5	9	9
	No. of iterations	578	565	433	436	463	507	514
	No. of K updates	29	19	40	23	15	12	14
total CPU time*, normalized		1.303	1.079	1.034	1.031	1.000	1.013	1.023

Notes:

* VAX 11/780 with VAX/VMS V4.5, not accurate as performance index

+ 15 QN vectors dragging at 870N for 21 iterations

Table 11.2.2(a). Numerical Performance as a function of MAXQN in ZBEAM

LOAD (N)	COMPUTATION EFFICIENCY	MAXQN						
		0	5	10	15	20	25	30
0 – 1000 (130 increments) severe nonlinearity	No. of QN updates	+	–	3010	3676	4112	4746	4632
	No. of line searches	+	–	333	341	347	362	353
	No. of iterations	+	–	1077	1094	1108	1111	1103
	No. of K updates	+	–	173	162	155	155	154
1000 – 2000 (100 increments) mild nonlinearity	No. of QN updates	+	–	1078	1926	2750	3673	4632
	No. of line searches	+	–	0	0	0	2	2
	No. of iterations	+	–	337	368	388	403	411
	No. of K updates	+	–	32	20	13	11	10
total CPU time*, normalized		+	–	1.050	1.054	1.001	1.000	1.010

Notes:

* VAX 11/780 with VAX/VMS V4.5, not accurate as performance index

+ Diverging solution at load step 100–200 N.

– Reaching the iteration limit of 30 at load step 300–400 N.

Case 2: Tables 11.2.1(b) and 11.2.2(b) summarize the numerical performance of the line search algorithm for ZPLATE and ZBEAM, respectively, by varying MAXLS from 0 to 6 with LSTOL set at 0.5. It is clear from the tables that the effectiveness of the line search varies with the extent of nonlinearity. Extensive line searches are performed in the highly nonlinear region: the number of line searches over the number of iterations is around 22% in the plate model and 29–33% in the beam model. In the slightly nonlinear region, the line search is hardly required: the ratio of the number of line searches to iterations is around 1% in the plate model and zero in the beam model. By increasing the value of MAXLS, more line search operations are performed in the highly nonlinear region with more favorable results for MAXLS of 4 and 5. However, little difference is observed in the efficiency for MAXLS between 3 and 6. In the less nonlinear region on the other hand, MAXLS of 2 or 3 shows more efficient results as illustrated in Table 11.2.1(b). When the line search is suppressed (MAXLS=0), both the plate and beam models fail to converge at the onset of plastic deformation around the load of 200 N.

Such failures illustrate the necessity of combining the line search operation in the nonlinear iteration algorithm.

Table 11.2.1(b). Numerical Performance as a function of MAXLS in ZPLATE

LOAD (N)	COMPUTATION EFFICIENCY	MAXLS						
		0	1	2	3	4	5	6
0 - 1000 (130 increments) severe nonlinearity	No. of line searches	+	223	259	259	260	244	247
	No. of iterations	+	1197	1166	1212	1159	1149	1164
	No. of K updates	+	157	146	138	137	137	138
1000-2000 (100 increments) mild nonlinearity	No. of line searches	+	6	4	5	13	7	5
	No. of iterations	+	471	441	463	469	469	459
	No. of K updates	+	15	17	15	15	15	15
total CPU time* (normalized)		+	1.081	1.035	1.010	1.000	1.006	1.013

Notes:

* VAX 11/780 with VAX/VMS V4.5 (not accurate as performance index)

+ Diverging solution at load step 200-300 N.

Table 11.2.2(b). Numerical Performance as a function of MAXLS in ZBEAM

LOAD (N)	COMPUTATION EFFICIENCY	MAXLS						
		0	1	2	3	4	5	6
0 - 1000 (130 increments) severe nonlinearity	No. of line searches	+	-	329	347	321	359	359
	No. of iterations	+	-	1109	1108	1094	1083	1090
	No. of K updates	+	-	162	155	149	150	145
1000-2000 (100 increments) mild nonlinearity	No. of line searches	+	-	0	0	0	0	0
	No. of iterations	+	-	388	388	388	388	388
	No. of K updates	+	-	12	12	12	12	12
total CPU time* (normalized)		+	-	1.031	1.001	1.000	1.002	1.010

Notes:

* VAX 11/780 with VAX/VMS V4.5 (not accurate as performance index)

+ Diverging solution at load step 100-200 N.

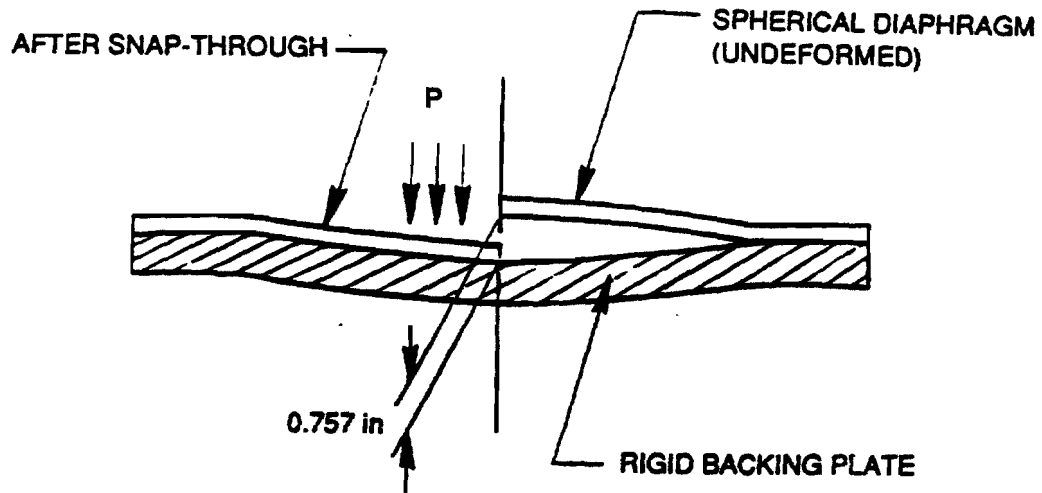
- Reaching the iteration limit of 30 at load step 500-600 N.

11.2.3 Snap-Through and Snap-Back: a Spherical Shell with Backing Plate

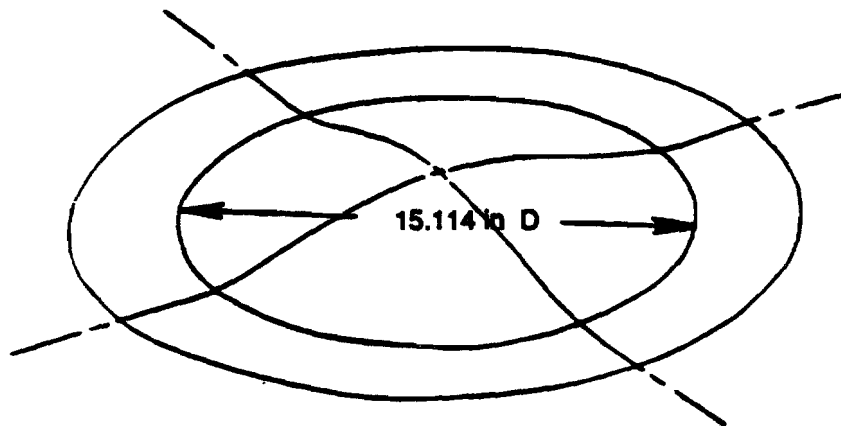
A spherical diaphragm resting on the rigid backing plate is subjected to a uniform pressure as shown in Fig. 11.2.4. The analysis objective is to find the critical pressure for snap-through and snap-back under different boundary conditions: clamped, simply supported, and sliding/simple supported cases. Additional constraints are applied after the snap-through by the backing plate. The material is linear elastic, and the deformation pattern is assumed to be axisymmetric because the diaphragm is connected to a plunger which prevents the apex from rotating.

A 10-degree sector of the diaphragm is modeled using shell elements with axisymmetric boundary conditions. The contact process between the diaphragm and the backing plate is simulated by gap elements as shown in Fig. 11.2.5. The flange portion is omitted in the model.

The main features of these problems are geometric nonlinearities due to large displacements, follower forces, and changes in constraints due to the backing plate. The convergence characteristics are influenced by the load increment and the iteration strategy. The increment size is reduced in the vicinity of the snap-through and snap-back. The convergence behavior is rather erratic and unpredictable around the critical loads.



Side View



Isometric View

Figure 11.2.4. Spherical Diaphragm Subjected to a Uniform Pressure.

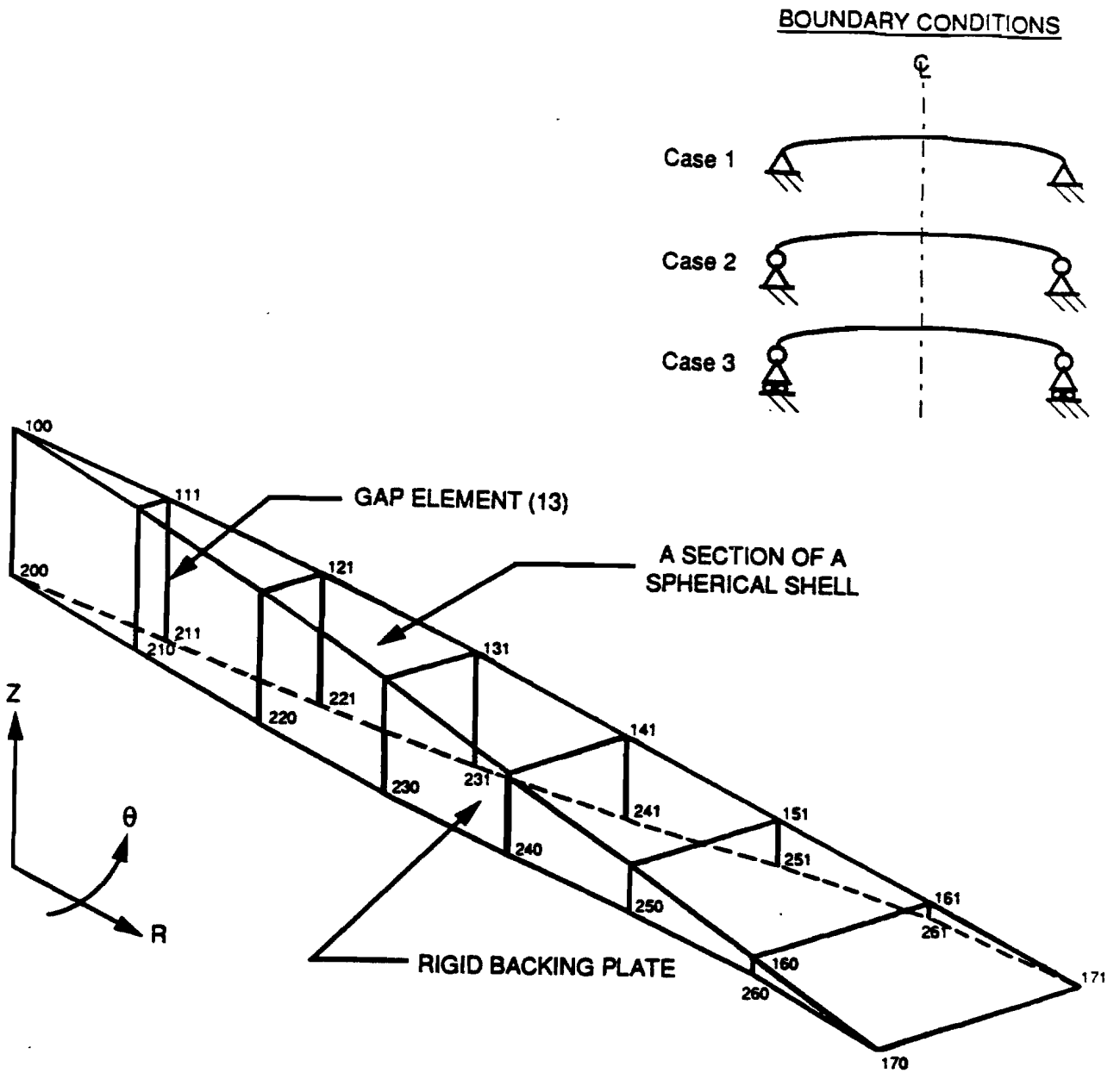


Figure 11.2.5. Spherical Diaphragm Model with Axisymmetric Boundary Condition and Constraints by Gap Elements.

i) Clamped Case

The load-deflection curve at the apex point for the clamped boundary conditions, constructed by combining loading and unloading paths, is shown in Fig. 11.2.6. The snap-through occurs between 0.96–0.98 psi, and the snap-back between 0.45–0.40 psi. Since the apex point comes into contact with the backing plate at 0.98 psi, the point does not translate beyond the gap distance (0.757 in.) despite the increasing pressure. The load history between two points A and B where the snap-through and the snap-back begin to occur is conjectured as shown by the dashed line.

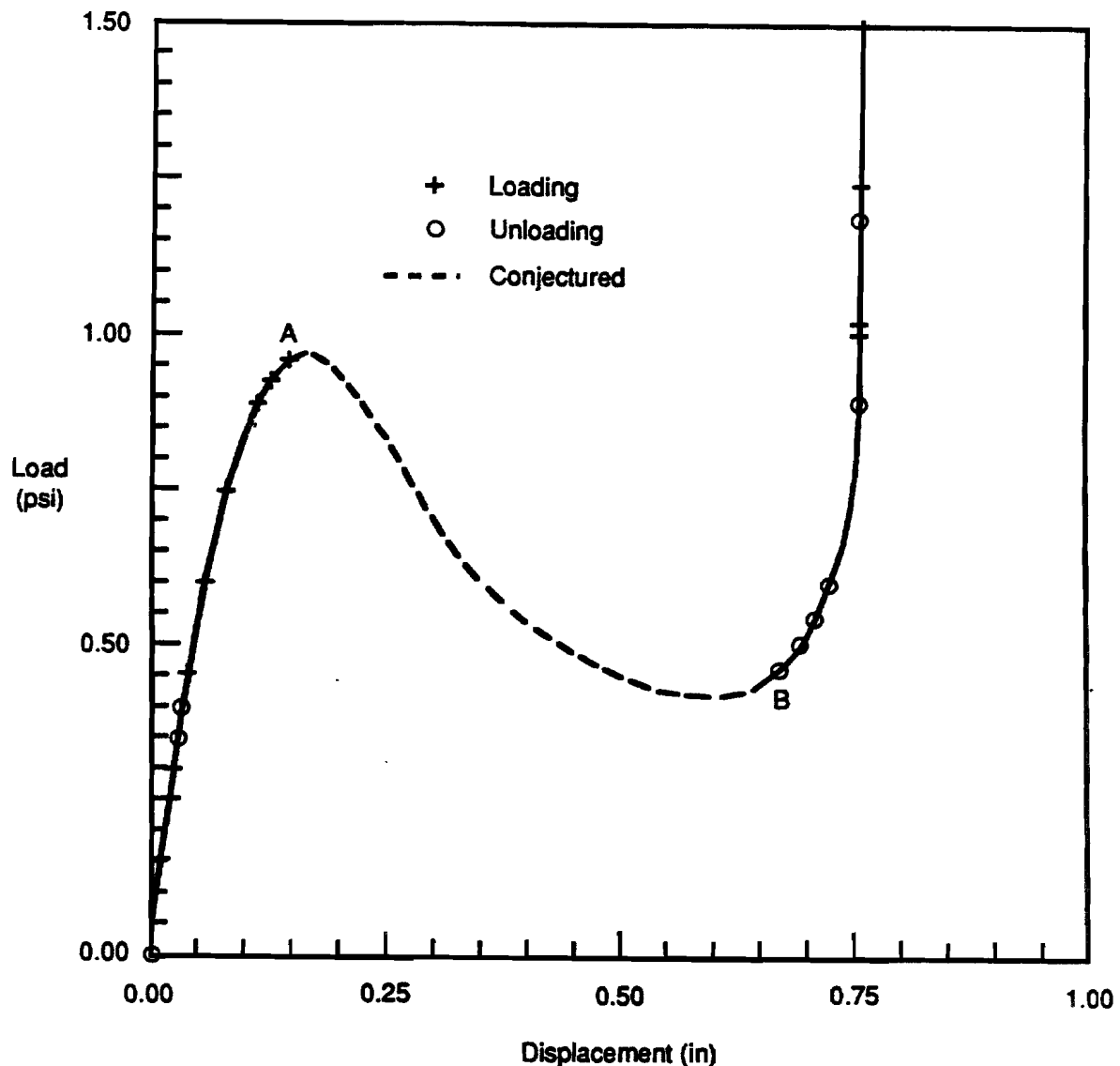


Figure 11.2.6. Load vs. Vertical Displacement at the Center (Clamped Case).

Case 1: The parametric study is performed by varying MAXQN from 0 to 30 with an interval of 5. The results are summarized in Table 11.2.3(a) for loading and unloading phases separately. Convergence near the snap-through point occurs only at a MAXQN of 10 and 20, which indicated the erratic nature of the problem. The unloading phase is analyzed by restarting from the converged solution with a MAXQN of 20. The fact that the unloading is easier than the loading for convergence is intriguing.

Table 11.2.3(a). Numerical Performance as a function of MAXQN in the spherical diaphragm with clamped boundary

LOAD (psi)	COMPUTATION EFFICIENCY	MAXQN						
		0	5	10	15	20	25	30
0 - 1.5 (12 increments) loading (snap-through)	No. of QN updates	+	+	172	+	306	+	+
	No. of line searches	+	+	61	+	57	+	+
	No. of iterations	+	+	67	+	70	+	+
	No. of K updates	+	+	13	+	9	+	+
	CPU time*	+	+	1.062	+	1.000	+	+
1.5 - 0 (9 increments) unloading (snap-back)	No. of QN updates	0	100	104	104	104	98	98
	No. of line searches	23	13	15	15	15	13	13
	No. of iterations	66	54	54	54	54	53	53
	No. of K updates	16	12	12	12	12	12	12
	CPU time*	1.147	1.011	1.024	1.005	1.021	1.022	1.000

Notes:

* VAX 8700 with VAX/VMS V4.5, CPU time normalized

+ Diverging solution at snap-through (0.96-0.98 psi)

Case 2: The parametric study is performed by varying MAXLS from 0 to 8. The results are summarized in Table 11.2.3(b) for loading and unloading phases separately. As indicated in the table, the convergence near the snap-through point requires MAXLS to be greater than 4. Computational efficiency appears to be better with MAXLS of 5 or 7 than 6, which cannot be rationalized. Again, the convergence is easier for unloading than the loading.

Table 11.2.3(b). Numerical Performance as a function of MAXLS in the spherical diaphragm with clamped boundary

LOAD (psi)	COMPUTATION EFFICIENCY	MAXLS								
		0	1	2	3	4	5	6	7	8
0 - 1.5 (13 increments) loading (snap-through)	No. of line searches	+	+	-	-	-	100	116	95	82
	No. of iterations	+	+	-	-	-	80	85	76	82
	No. of K updates	+	+	-	-	-	12	12	11	11
1.5 - 0 (9 increments) unloading (snap-back)	No. of line searches	+	+	-	-	-	17	76	9	70
	No. of iterations	+	+	-	-	-	51	69	51	69
	No. of K updates	+	+	-	-	-	10	17	12	16
total CPU time* (normalized)		+	+	-	-	-	1.070	1.210	1.000	1.177

Notes:

* VAX 8700 with VAX/VMS V4.5

+ Diverging solution at the pressure 0.96-0.98 psi

- Reaching the iteration limit of 30 in the vicinity of the snap-through pressure

ii) Simply Supported Case

In the simply supported case, the snap-through occurs twice as shown in Fig. 11.2.7. The first snap-through occurs at 0.54–0.56 psi, and the second occurs at 0.70–0.705 psi. The first snap-through brings the diaphragm into an interim stable mode by flipping it into a shape of double curvature. The diaphragm comes into contact with the backing plate near the periphery immediately after the first snap-through, stabilizing the interim position. The central area of the diaphragm comes into contact with the backing plate upon the second snap-through. If the pressure is unloaded before the second snap-through, the diaphragm snaps back at 0.50–0.48 psi. Once the second snap-through occurs, however, the snap-back does not occur upon unloading because the spring-back force is trapped by the constraint forces at the periphery. Instead, the central area remains in contact with the backing plate during unloading until the pressure is reduced to less than 0.1 psi (around point B). When the pressure is removed entirely, the central area is lifted off, leaving the peripheral area in contact with the backing plate. Based on these observations, the load history between points A and B can be conjectured by the dashed line in Fig. 11.2.7.

Case 1: Table 11.2.4 summarizes the numerical performance as a function of MAXQN for the first and second snap-through phases separately. The convergence behavior dictates that the maximum number of line searches allowed for each iteration be set to 2 and 10 for the first and the second snap-through, respectively. The first snap-through requires MAXQN to be greater than 0 for convergence. The solution to the second snap-through phase can only be obtained with a MAXQN of 15 and 20. Computational efficiency appears to be better with a MAXQN of 20.

Table 11.2.4. Numerical Performance as a function of MAXQN in the spherical diaphragm with simply supported boundary

LOAD (psi)	COMPUTATION EFFICIENCY	MAXQN						
		0	5	10	15	20	25	30
0 – 0.6 (10 increments) first snap-through	No. of QN updates	+	170	113	145	252	296	296
	No. of line searches	+	8	8	8	8	9	9
	No. of iterations	+	60	51	51	49	57	57
	No. of K updates	+	8	9	8	8	7	7
0.6 – 0.8 (12 increments) second snap-through	No. of QN updates	+	+	–	153	138	–	–
	No. of line searches	+	+	–	163	88	–	–
	No. of iterations	+	+	–	61	50	–	–
	No. of K updates	+	+	–	14	10	–	–
total CPU time*, normalized		+	+	–	1.167	1.000	–	–

Notes:

* VAX 8700 with VAX/VMS V4.5

+ Diverging solution at first and second snap-through

– Reaching the iteration limit of 30 at the second snap-through

Case 2: The first snap-through requires MAXLS to be greater than 1 for convergence and the second snap-through converges only with MAXLS of 6 and 10 for the MAXQN range of 0 to 10. The parametric study is not fruitful for this case due to the erratic numerical behavior.

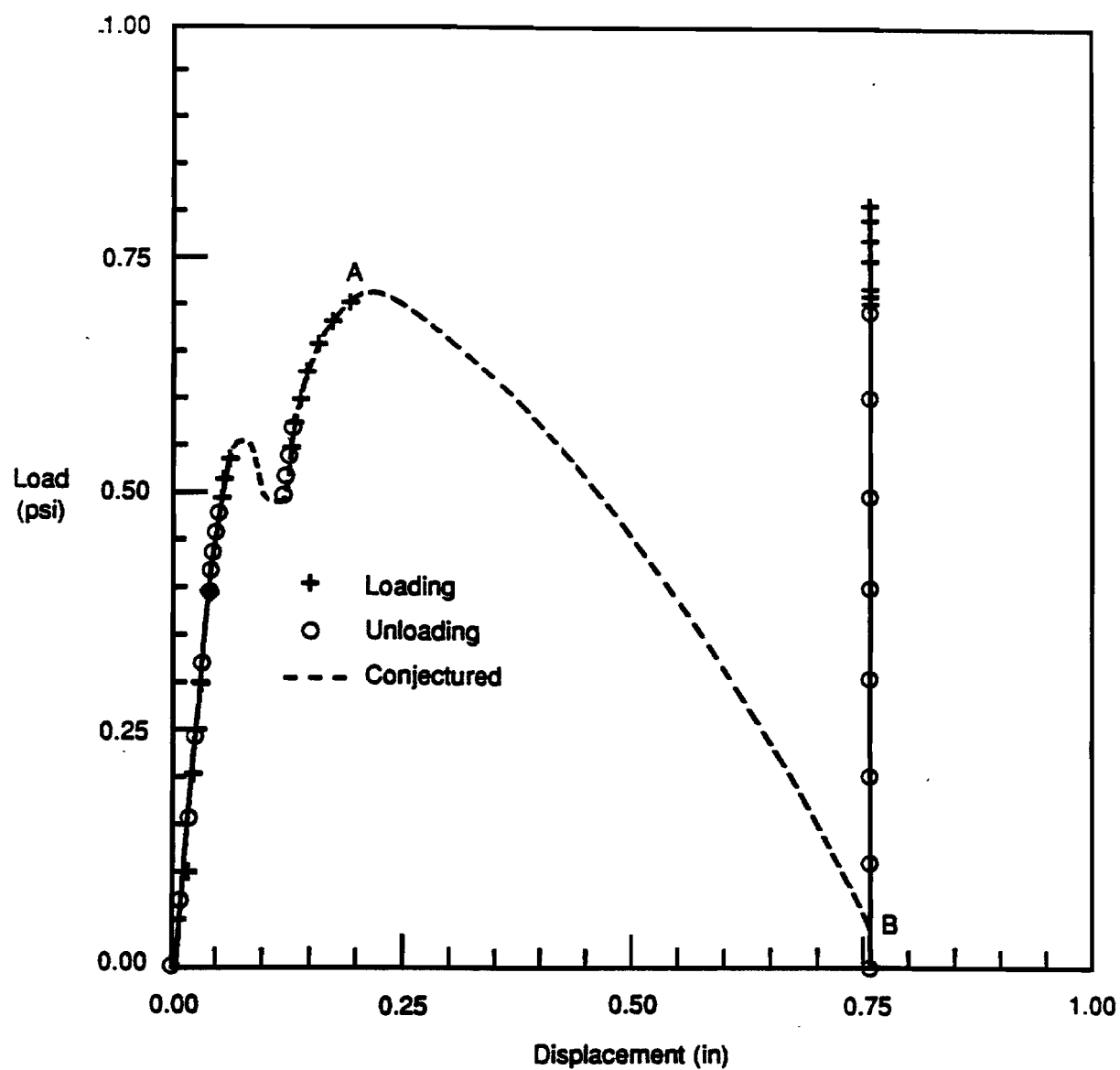


Figure 11.2.7. Load vs. Vertical Displacement at the Center (Simply Supported Case).

iii) Simply Supported and Sliding Case

In the case where the periphery is allowed to slide in the radial direction while it is simply supported, the diaphragm undergoes a slight snap-through at a pressure between 0.188–0.190 psi. The snap-back occurs at a pressure between 0.184–0.182 psi as shown in Fig.11.2.8. Since the snap-through in this case is moderate and shallow, as shown in the figure, the pressure is incremented in very small steps around the critical load to obtain a more descriptive load-deflection curve.

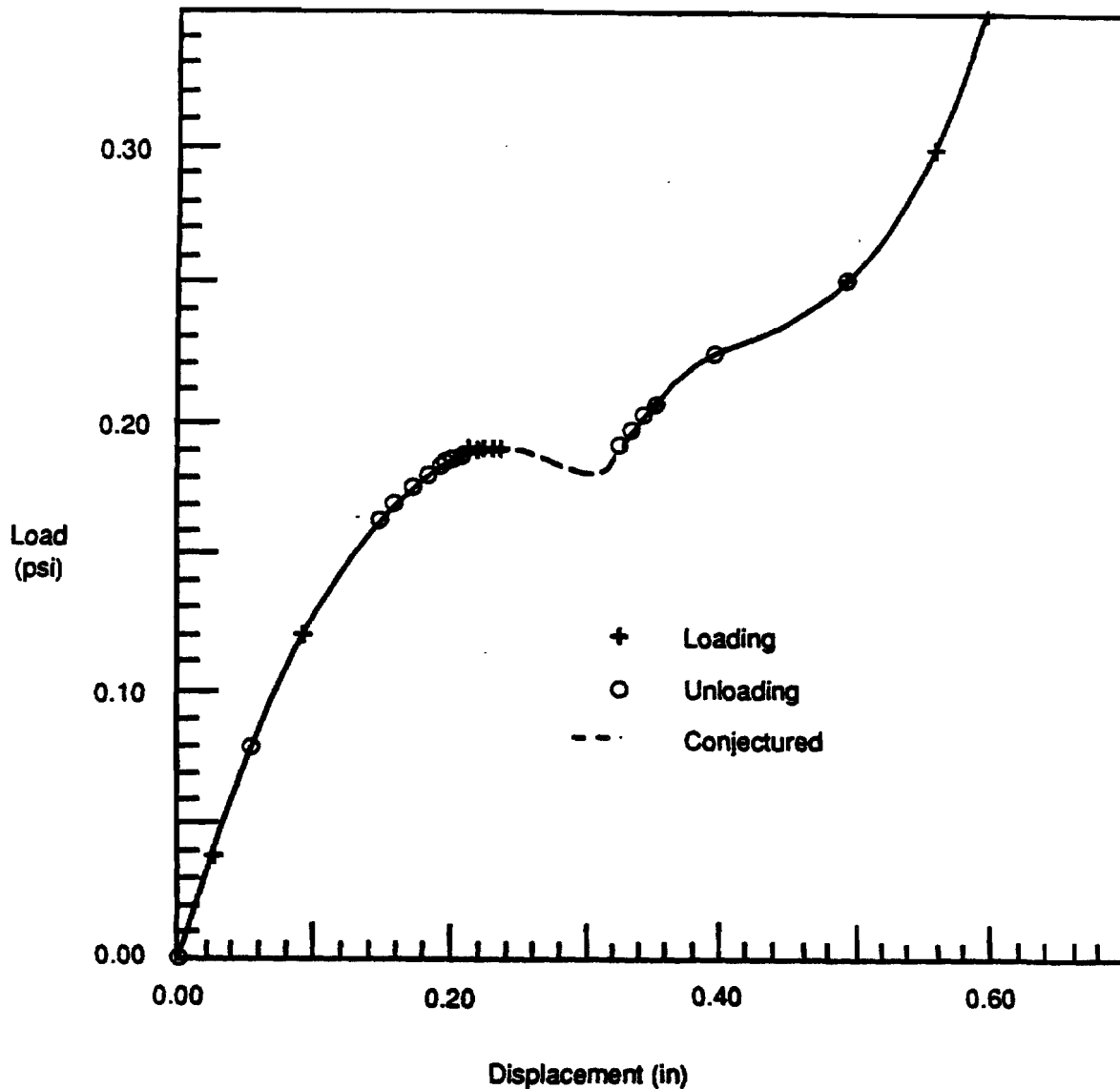


Figure 11.2.8. Load vs. Vertical Displacement at the Center (Sliding and Simply Supported Case).

Case 1: Table 11.2.5(a) summarizes the numerical performance as a function of MAXQN for the loading and the unloading phases separately. The table shows that the solution diverges around the snap-through or snap-back point for a MAXQN less than 10. This indicates that BFGS update operations do help to overcome the difficulties near the singularity in the stiffness. The computational efficiency improves slightly as MAXQN increases; the MAXQN values of 25 and 30 seem to be more efficient.

Table 11.2.5(a). Numerical Performance as a function of MAXQN in the spherical diaphragm with sliding/simply supported boundary

LOAD (psi)	COMPUTATION EFFICIENCY	MAXQN						
		0	5	10	15	20	25	30
0 - 0.6 (15 increments) loading (snap-through)	No. of QN updates	+	+	311	367	608	603	653
	No. of line searches	+	+	20	21	19	19	19
	No. of iterations	+	+	88	91	90	90	90
	No. of K updates	+	+	10	7	6	5	5
0.6 - 0 (9 increments) unloading (snap-back)	No. of QN updates	+	+	193	231	183	245	171
	No. of line searches	+	+	20	23	26	21	21
	No. of iterations	+	+	61	61	59	59	55
	No. of K updates	+	+	7	6	7	5	5
total CPU time*, normalized		+	+	1.108	1.115	1.074	1.000	1.027

Notes:

* VAX 8700 with VAX/VMS V4.5

+ Diverging solution at snap-through or snap-back point

Case 2: Table 11.2.5(b) summarizes the numerical performance as a function of MAXLS while LSTOL is fixed at 0.5. The table shows that the solution diverges around the snap-through or snap-back for MAXLS less than 3. This indicates that line search operations do help to overcome the difficulties near the singularity in the stiffness. The significance demonstrated in Table 11.2.5(b) is the effectiveness of the line search, and the efficiency is not sensitive to the value of MAXLS from 3 to 6. The parametric study on the line search tolerance (LSTOL) with MAXLS of 5 is summarized in Table 11.2.5(c). Notice that when LSTOL is small (0.1-0.3), the algorithm keeps thrashing on line search operations until the solution diverges at the snap-through or snap-back point. It also indicates that the algorithm becomes inefficient if LSTOL has too large a value, e.g., 0.7. The choice of 0.5 for LSTOL seems to be a good compromise for convergence and efficiency of the line search method.

Table 11.2.5(b). Numerical Performance as a function of MAXLS in the spherical diaphragm with sliding/simply supported boundary

LOAD (psi)	COMPUTATION EFFICIENCY	MAXLS						
		0	1	2	3	4	5	6
0 - 0.6 (15 increments) loading (snap-through)	No. of line searches	0	+	+	16	18	19	21
	No. of iterations	87	+	+	91	91	90	90
	No. of K updates	4	+	+	5	5	5	5
0.6 - 0 (9 increments) unloading (snap-back)	No. of line searches	+	+	+	22	22	22	22
	No. of iterations	+	+	+	59	62	62	62
	No. of K updates	+	+	+	5	5	5	5
total CPU time* (normalized)		+	+	+	1.010	1.003	1.008	1.000

Notes:

* VAX 11/780 with VAX/VMS V4.5 (not accurate as performance index)

+ Diverging solution at snap-through or snap-back point

Table 11.2.5(c). Numerical Performance as a function of LSTOL in the spherical diaphragm with sliding/simply supported boundary

LOAD (psi)	COMPUTATION EFFICIENCY	LSTOL						
		0.1	0.2	0.3	0.4	0.5	0.6	0.7
0 - 0.6 (15 increments) loading (snap-through)	No. of line searches	+	+	+	22	19	17	35
	No. of iterations	+	+	+	95	90	90	106
	No. of K updates	+	+	+	5	5	5	8
0.6 - 0 (9 increments) unloading (snap-back)	No. of line searches	+	+	+	22	22	23	22
	No. of iterations	+	+	+	61	62	65	65
	No. of K updates	+	+	+	5	5	5	6
total CPU time* (normalized)		+	+	+	1.006	1.000	1.012	1.122

Notes:

* VAX 11/780 with VAX/VMS V4.5 (not accurate as performance index)

+ Diverging solution at snap-through or snap-back point

11.2.4 Pre-Buckling Behavior: an Imperfect Spherical Cap

A spherical shell with an initial imperfection is analyzed [11.6]. The shell is subjected to an external uniform pressure, while the periphery is clamped. The problem is assumed to remain axisymmetric in geometry and loading throughout the deformation. The details of the geometry and the material are shown in Fig. 11.2.9. The initial imperfection is introduced by making the radius of curvature near the apex greater than the shell radius. The material is elastoplastic with von Mises yield criterion and kinematic hardening, and the large displacement is assumed in the analysis. The external pressure is gradually increased until the shell collapses [11.7], as shown in Fig. 11.2.10. The finite element model shown in Fig. 11.2.11 represents a 10-degree sector in solid elements with axisymmetric boundary conditions.

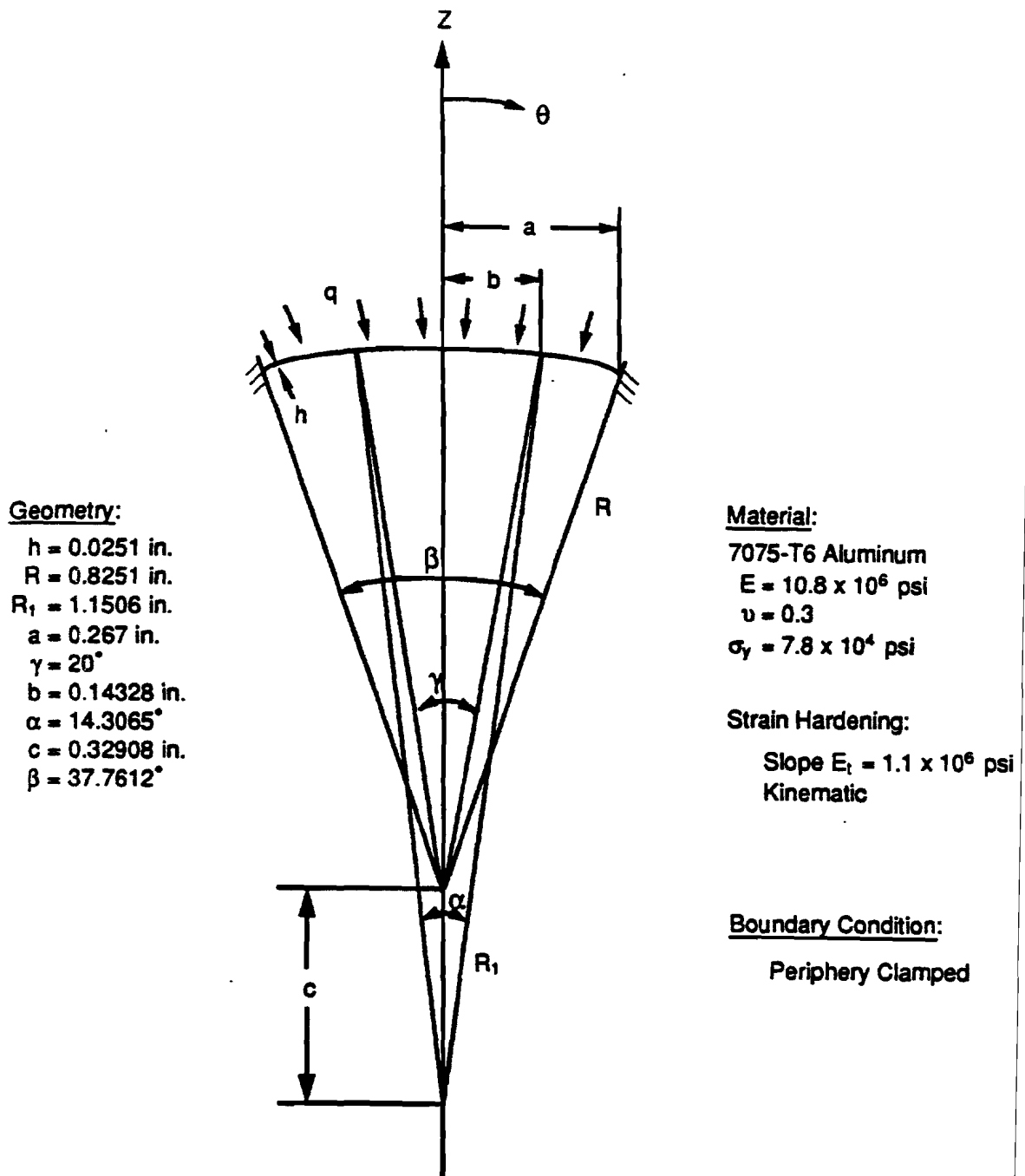


Figure 11.2.9. Imperfect Spherical Shell.

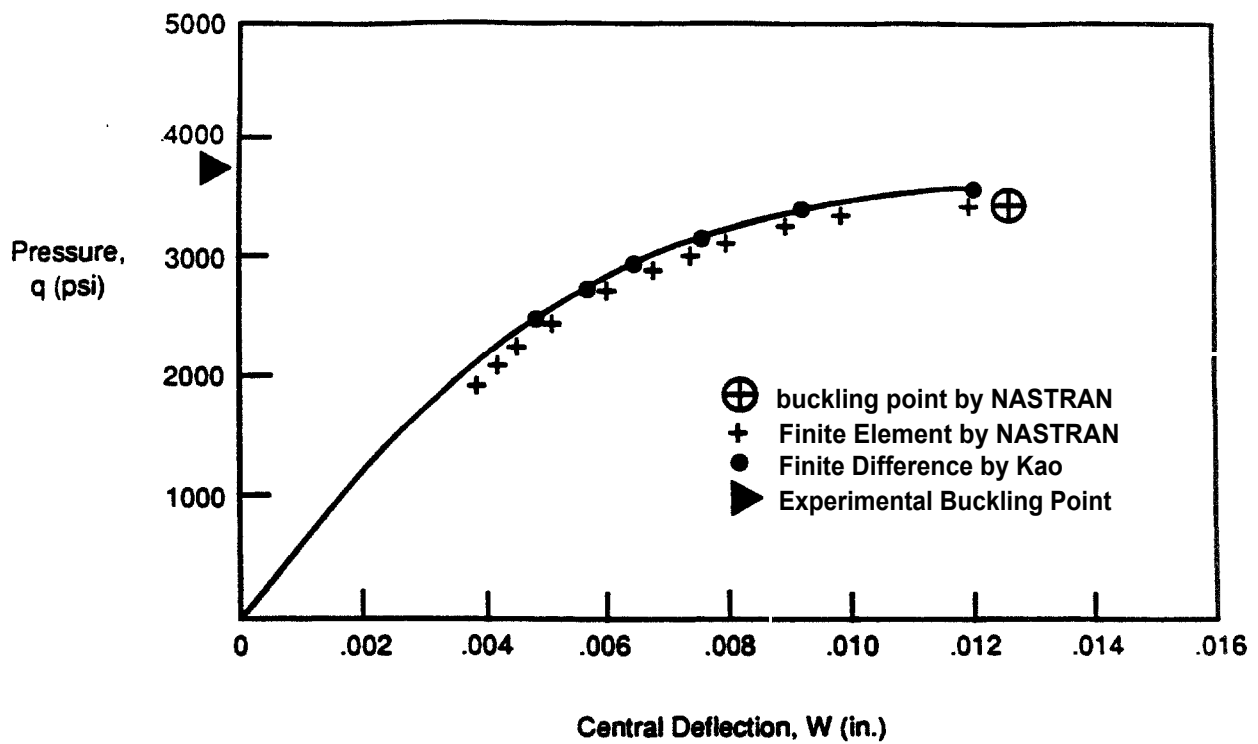


Figure 11.2.10. Elastic-Plastic Buckling of a Clamped Spherical Cap with Flat Spot (Load vs. Central Deflection).

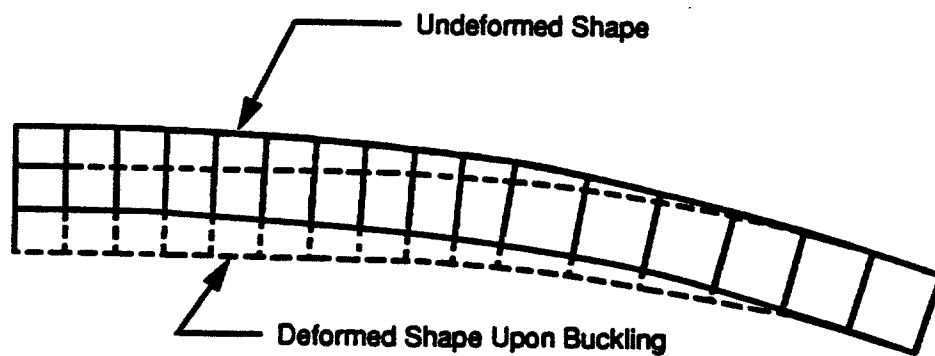


Figure 11.2.11. Deformed Shape upon Buckling with a Solid Model.

For the parametric studies, numerous analyses are repeated for various values of MAXQN, MAXLS and LSTOL with the pressure increased up to 3500 psi. In this problem, the loading can be divided into two stages: the plastic deformation developing stage (0–3000 psi) and the instability initiating stage (3000–3500 psi). Nonlinearity is fairly mild in both stages compared to other examples. In the second stage, all the elements undergo plastic deformation.

Case 1: The results are summarized in Table 11.2.6(a), which illustrates the numerical performance of the program as a function of MAXQN. Note that in Table 11.2.6(a) convergence is achieved for all values of MAXQN except for the second loading stage and MAXQN=0, where the geometric nonlinear effects become more prominent. The computational efficiency is most favorable with a MAXQN of 20.

Table 11.2.6(a). Numerical Performance as a function of MAXQN in the imperfect spherical cap

LOAD (psi)	COMPUTATION EFFICIENCY	MAXQN						
		0	5	10	15	20	25	30
0 – 3000 (7 increments) plasticity developing	No. of QN updates	0	154	165	161	200	368	368
	No. of line searches	28	8	4	2	2	1	1
	No. of iterations	79	43	41	38	37	38	38
	No. of K updates	8	2	4	3	2	1	1
3000 – 3500 (5 increments) instability developing	No. of QN updates	+	158	133	188	229	224	224
	No. of line searches	+	4	6	5	3	5	5
	No. of iterations	+	41	32	33	31	34	34
	No. of K updates	+	3	3	2	2	3	3
total CPU time*, normalized		+	1.217	1.202	1.086	1.020	1.005	1.000

Notes:

* VAX 8700 with VAX/VMS V4.5, not accurate as performance index

+ Diverging solution at load step 3400–3500 psi

Case 2: The results are summarized in Tables 11.2.6(b) and 11.2.6(c), which illustrate the numerical performance of the program as functions of MAXLS and LSTOL. Note that in Table 11.2.6(b) the convergence is not only achieved when the line search is suppressed but also achieved most efficiently for MAXLS=0. When the line search operation is allowed (MAXLS > 0, LSTOL=0.5), more line searches are performed in the second stage where the geometric nonlinear effects become more prominent. The parametric study performed on LSTOL with MAXLS=5 is summarized in Table 11.2.6(c). A quite natural consequence of unnecessary line searches is exemplified by decreasing efficiency with smaller LSTOL. However, Table 11.2.6(c) does demonstrate that the accurate line search does not expedite convergence but rather deteriorates efficiency. This problem illustrates that the line search is unnecessary for moderately nonlinear problems where the divergence is not an issue.

Table 11.2.6(b). Numerical Performance as a function of MAXLS
in the imperfect spherical cap

LOAD (psi)	COMPUTATION EFFICIENCY	MAXLS						
		0	1	2	3	4	5	6
0 - 3000 (7 increments) plasticity developing	No. of line searches	0	3	2	2	2	2	2
	No. of iterations	37	38	37	37	37	37	37
	No. of K updates	2	2	2	2	2	2	2
3000 - 3500 (5 increments) instability developing	No. of line searches	0	13	19	18	18	19	19
	No. of iterations	50	49	49	49	49	49	49
	No. of K updates	7	7	7	7	7	7	7
total CPU time* (normalized)		1.000	1.039	1.072	1.077	1.083	1.086	1.080

Notes:

* VAX 8700 with VAX/VMS V4.5

Table 11.2.6(c). Numerical Performance as a function of LSTOL
in the imperfect spherical cap

LOAD (psi)	COMPUTATION EFFICIENCY	LSTOL						
		0.1	0.2	0.3	0.4	0.5	0.6	0.7
0 - 3000 (7 increments) plasticity developing	No. of line searches	23	9	2	2	2	1	1
	No. of iterations	36	36	37	37	37	37	37
	No. of K updates	2	2	2	2	2	2	2
3000 - 3500 (5 increments) instability developing	No. of line searches	56	32	23	23	19	19	17
	No. of iterations	48	48	49	49	49	49	49
	No. of K updates	7	7	7	7	7	7	7
total CPU time* (normalized)		1.214	1.083	1.046	1.056	1.020	1.014	1.000

Notes:

* VAX 11/780 with VAX/VMS V4.5

11.2.5 Concluding Remarks

The BFGS update method is an expedient and effective procedure to prevent divergence in complex nonlinear problems. It is also an expeditious method that improves computational efficiency. If divergence is not an issue, the solution requires more line searches, iterations and stiffness updates in the absence of BFGS updates, as demonstrated in the detailed parametric studies. Preliminary tests showed that the current BFGS update scheme works well on the negative-definite matrices, but did not work on the unsymmetric matrix.

Although the line search method appears theoretically attractive, the computational intensity involved in the search process often prohibits indiscriminate use of this procedure in nonlinear finite element analysis. The line search method is an effective procedure to prevent divergence in complex nonlinear problems. It is particularly useful to cope with difficulties when the stiffness approaches singularity. The line search process can help to converge the solution dramatically in some cases, such as bypassing the unstable regime of the snap-through

problem without the aid of a displacement control method or when the structure stiffens drastically as in the closing gap phenomenon. However, indiscriminate use of line searches causes adverse effects on efficiency. In fact, the line search tends to deteriorate the efficiency in moderately nonlinear problems where divergence is not encountered. Even when the line search is valid, the numerical behavior associated with the line search is rather erratic. Moreover, the line search diminishes the effectiveness and efficiency of the full Newton-Raphson iteration.

Optimal values of parameters and tolerances are extremely difficult to achieve. They are not only problem dependent but also vary within a problem during the course of an incremental process. Tolerances and parameter values in the software were tuned based on 50 nonlinear test problems [11.8]. Default values are intended to provide the best workable method for the wide spectrum of nonlinear problems without any insight or experience, but they are by no means optimal.

11.3 CREEP ANALYSIS OF THICK-WALLED PRESSURE VESSEL

11.3.1 Introduction

The purpose of this example is to illustrate the creep solution of a typically encountered structure. The structure contains a complex stress pattern requiring an axisymmetric three-dimensional solution. The material has the properties of an engineering metal subjected to high pressure.

11.3.2 Problem Description

The creep behavior of a thick-walled pressure vessel with a flat-end closure [11.9] is analyzed under an internal pressure of 445 psi. The geometry of the pressure vessel, as shown in Fig. 11.3.1, is defined as follows:

Overall length	0.89 inch
Outside radius	0.25 inch
Inside radius	0.159 inch
Wall thickness (uniform)	0.091 inch
Fillet radius	0.03 inch

The material has the following elastic properties:

Young's Modulus	20×10^6 psi
Poisson's Ratio	0.3

and is assumed to obey an empirical creep law in the form of

$$\epsilon^c = (19.8 \times 10^{-16})(\sigma^{3.61})(t^{1.06})$$

where σ and t are measured in psi and hours, respectively.

11.3.3 Model Description

The modeling is simplified by axisymmetry and symmetry about the mid-plane. A finite element model is constructed using 72 solid elements (4 PENTAs and 68 HEXAs) with 355 active degree-of-freedom. Appropriate boundary conditions are imposed for symmetry (using SPC1, SPCADD and GRDSET entries) in the cylindrical coordinate system (CORDIC). A local coordinate system (CORD2C) is used to specify the geometry of a toroidal section.

The internal pressure is applied by PLOAD4 entries. Although the applied load does not produce plastic deformation, all the elements are treated as being made of elasto-plastic-creep material by attaching MATS1 and CREEP entries to the MAT1 Bulk Data entry.

Fig. 11.3.1 shows an isometric view of the model. Also shown is a model in the Reference [11.9] for comparison. Table 11.3.1 shows a listing of the input data.

11.3.4 Solution Procedure

The creep analysis should be performed using SOL 66 of Version 63. The internal pressure of 445 psi is applied in the first subcase for the initial static solution. Creep analyses are performed in the subsequent subcases by applying a nonzero time increment while the applied pressure is kept constant.

Six subcases (only two subcases are activated on the listing shown in Table 11.3.1) are set up as follows:

Subcase ID	KMETHOD	Increments	Δt (Hour)	Ending Creep Time (Hour)
100	AUTO	1	0.	0.
200	SEMIQN	5	0.02	0.1
300	AUTOQN	9	0.1	1.0
400	AUTOQN	10	0.1	2.0
500	AUTOQN	5	0.2	3.0
600	AUTOQN	5	0.4	5.0

The computer run was stopped after every subcase to obtain the stress contour plot. Then the analysis was continued by restarting from the last converged solution. Notice that the restart run requires three PARAMeter entries (i.e., SUBID, LOADINC and LOOPID) to designate the solution from which the restart run starts.

11.3.5 Results and Discussion

The entire analysis took six runs (five restarts) to obtain solutions up to five hours of creep. Each creep solution took three to four iterations to converge (average 10 CPU minutes on the VAX 11/780). The results of the analysis are presented in Fig. 11.3.2 through 11.3.5 in comparison with those of the Reference.

Fig. 11.3.2 and 11.3.3 show stress contours at $t = 0$ and 3, respectively. The stress history at two selected points (junction of the cylinder and end closure) are shown in Fig. 11.3.4. It can be seen that the steady-state stress is reached after approximately three hours of creep. Notice that the software shows a higher stress level at the inside surface (where the stress concentration occurs) throughout the analysis. Since the initial stress is similarly higher, this

error is actually a consequence of a coarse mesh. Also, notice that the point with higher stress relaxes a little faster than might be expected.

It is not possible to plot creep strain contours with the current version of the software. But the effective creep strains are printed with nonlinear stresses. Effective creep strains at five sampling points at $t = 3$ hours are shown on the creep strain contour plot in the Reference. The results are in good agreement with the Reference. More accurate solutions should be obtained with a more refined mesh around the fillet area.

11.3.6 Input Data Listing for the Pressure Vessel

```
ID TWCAS,V65 $ SHL 6/21/83
SOL 66 $
DIAG 8, 50 $,20, 51
TIME 40 $ FOR VAX
CEND
TITLE = THICK WALLED CYLINDER SUBJECT TO INTERNAL PRESSURE
SUBTITLE = AXISYMMETRIC
ECHO = UNSORT
  DISP = ALL
  STRESS(PLOT) = ALL
  GPSTRESS=ALL
  STRFIELD=ALL
SEALL = ALL
  SPC = 100
SUBCASE 100
  LABEL = STATIC ANALYSIS
  LOAD = 100
  NLPARM = 10
SUBCASE 200
  LABEL = CREEP 0.1 HOUR
  LOAD = 100
  NLPARM = 20
SUBCASE 300
  LABEL = CREEP 1 HOUR
  LOAD = 100
  NLPARM = 30
SUBCASE 400
  LABEL = CREEP 2 HOURS
  LOAD = 100
  NLPARM = 40
SUBCASE 500
  LABEL = CREEP 3 HOURS
```

```

        LOAD = 100
        NLPARM = 50
SUBCASE 600
        LABEL = CREEP 5 HOURS
        LOAD = 100
        NLPARM = 60
OUTPUT(PLOT)
        CSCALE 1.3
        PLOTTER NAST
        SET 1 ALL
        AXES Z,X,Y
        VIEW 30.,20.,0.
        FIND SCALE ORIGIN 1 SET 1
        PTTITLE=ISOMETRIC VIEW
        PLOT SET 1 ORIGIN 1 LABEL POINT
        VIEW 0. , 0. , 0.
        PTTITLE=SIDE VIEW
        FIND SCALE ORIGIN 1 SET 1
        PLOT SET 1 ORIGIN 1 LABEL ELEMENT
        PLOT STATIC SET 1 ORIGIN 1
OUTPUT(POST)
        SET 10 = ALL
        VOLUME 10 SET 10 SYSTEM CORD 1
BEGIN BULK
$ GENERAL
PARAM  POST    0
NLPARM 10      1          AUTO
NLPARM 20      5          0.02
NLPARM 30      9          0.1
NLPARM 40     10          0.1
NLPARM 50      5          0.2
NLPARM 60      5          0.4
CORD1C  1      1000      1001      1002
GRID    1000    0         0.         0.         0.         0         123456
GRID    1001    0         0.         0.5        0.         0         123456
GRID    1002    0         0.5        0.         0.         0         123456
PLOTTEL 1001    1000      1001
PLOTTEL 1002    1000      1002
CORD2C  2         0.129    0.324    0.         0.129    0.324    -1.         +COR2
+COR2   0.129    0.354    0.
$ PROPERTIES
MAT1    1         20.0+6          0.3
MAT1    1         PLASTIC 2.0+6    1         1         16.+3

```


CREEP	1			CRLAW					+CR1
+CR1	300	19.8-16	3.61		1.06				
PSOLID	3	1	1	0					
\$ BOUNDARY CONDITIONS									
SPCADD	100	200	300						
SPC1	200	23456	1301	1302	1321	1322	1341	1342	+SPC
+SPC	1361	1362	1381	1382					
SPC1	300	12456	1100	1120	1140	1160	1180		
\$ LOADING CONDITIONS									
LOAD	100	1.	445.	200					
PLOAD4	200	104	1.				1180		+PL104
PLOAD4	200	108	1.				1181	1184	
PLOAD4	200	112	1.				1183	1186	
PLOAD4	200	116	1.				1185	1188	
PLOAD4	200	120	1.				1187	1190	
PLOAD4	200	204	1.				1189	1282	
PLOAD4	200	208	1.				1281	1284	
PLOAD4	200	212	1.				1283	1286	
PLOAD4	200	216	1.				1285	1288	
PLOAD4	200	220	1.				1287	1290	
PLOAD4	200	224	1.				1289	1396	
PLOAD4	200	304	1.				1381	1384	
PLOAD4	200	308	1.				1383	1386	
PLOAD4	200	312	1.				1385	1388	
PLOAD4	200	316	1.				1387	1390	
PLOAD4	200	320	1.				1389	1392	
PLOAD4	200	324	1.				1391	1394	
PLOAD4	200	328	1.				1393	1396	
\$ GRIDS									
GRDSET		1	R		Z				
GRID	1100		0.	0.	0.445				
GRID	1101		0.025	2.	0.445				

***** OTHER GRID CARDS ARE OMITTED. *****

GRID	1201	2	0.121	15.	5.595-3				
GRID	1396		0.159	-2.	0.324				
\$ ELEMENTS									
CPENTA	101	3	1100	1101	1102	1120	1121	1122	

\$***** OTHER CONNECTIVITY CARDS ARE OMITTED. *****

CHEXA	328	3	1373	1374	1376	1375	1393	1394	+H328
+H328	1396	1395							
ENDDATA									

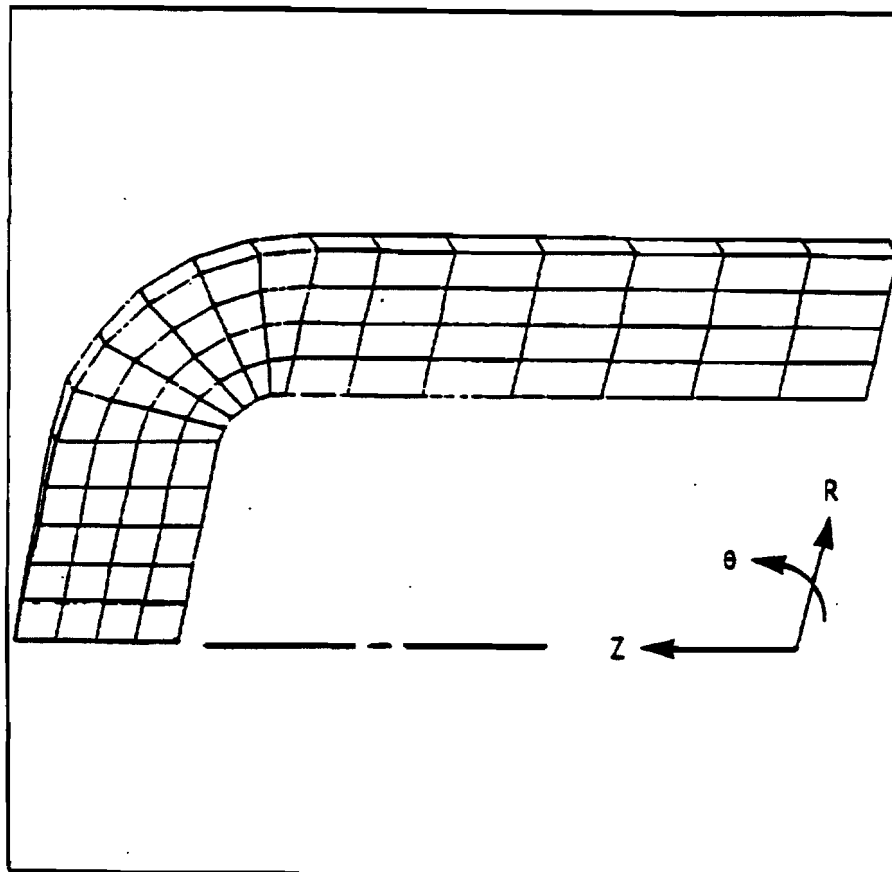
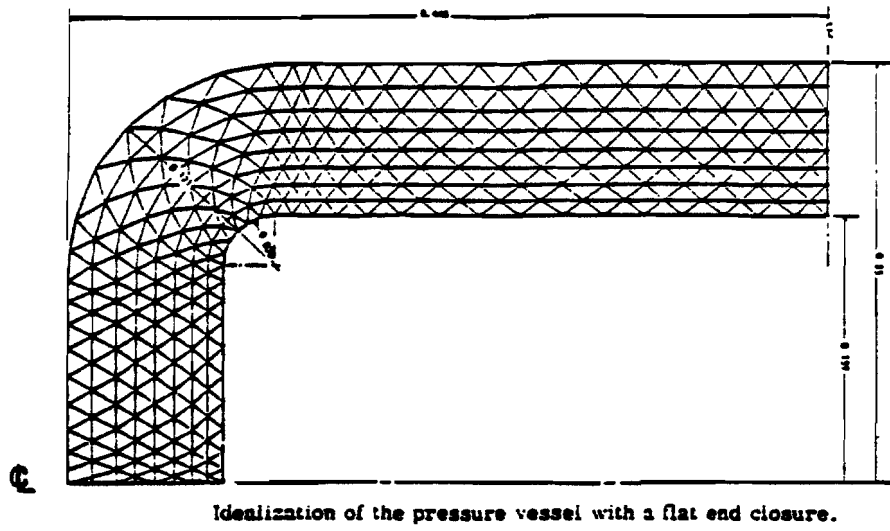
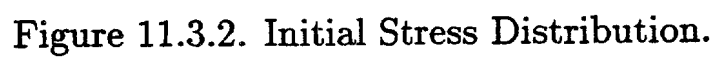
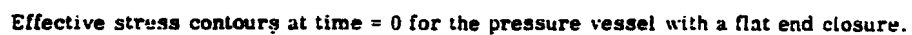
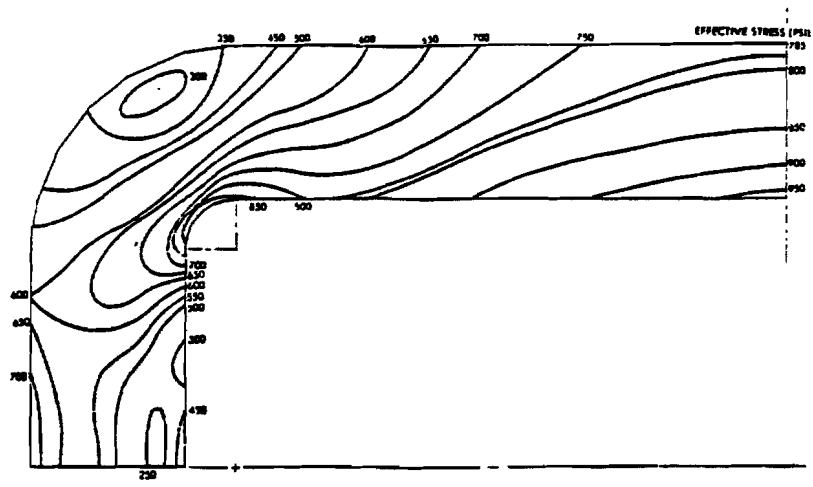


Figure 11.3.1. Finite Element Model of Pressure Vessel.





Effective stress contours at time = 3.0 hours for the pressure vessel with a flat end closure.

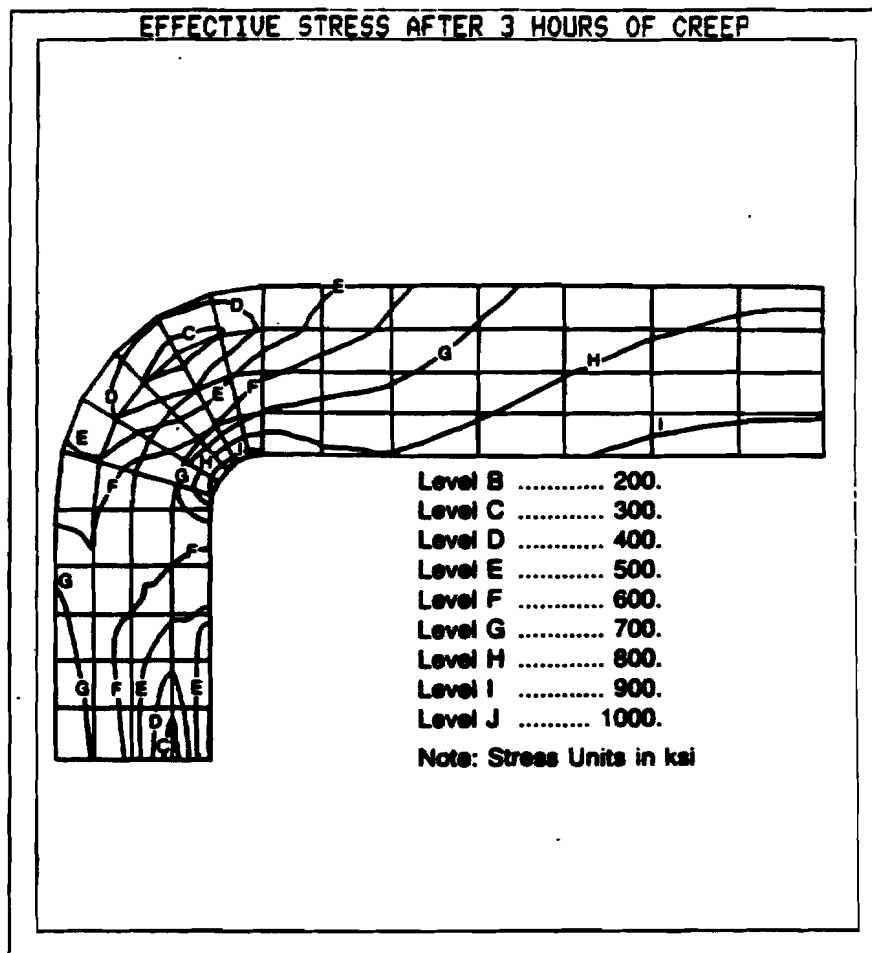


Figure 11.3.3. Effective Stress Distribution after 3 hours of Creep

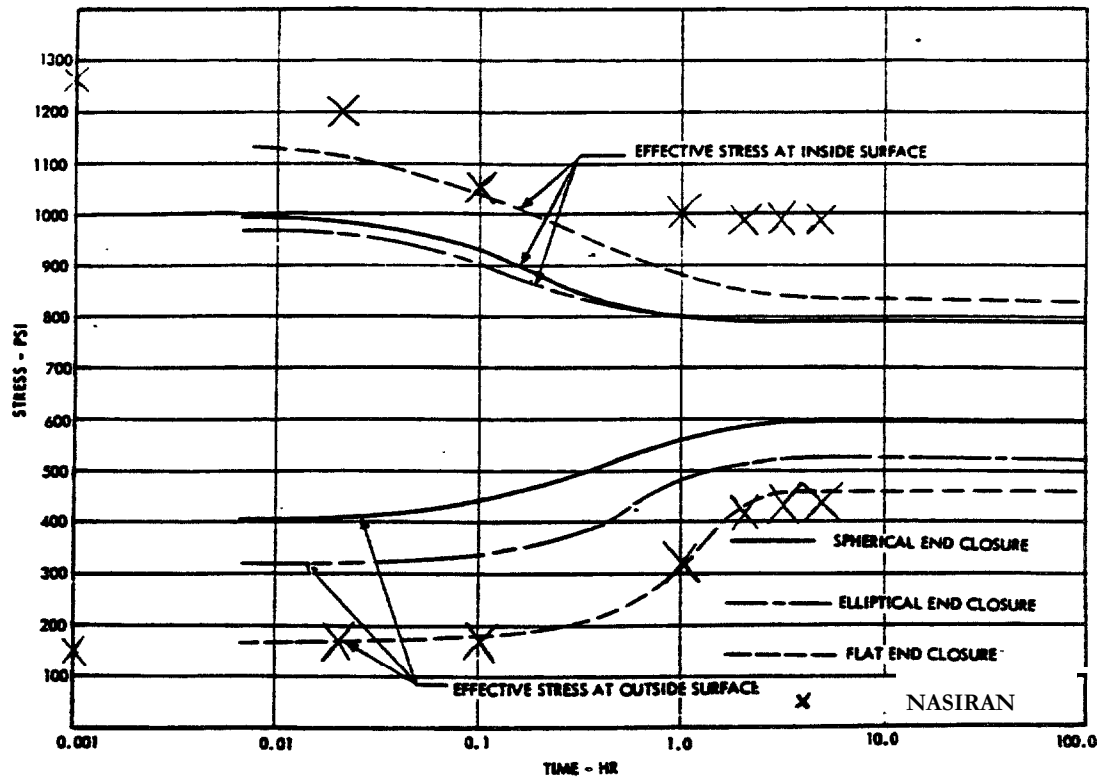


Figure 11.3.4. Stress Relaxation at the Junction of the Cylinder and End Closure.

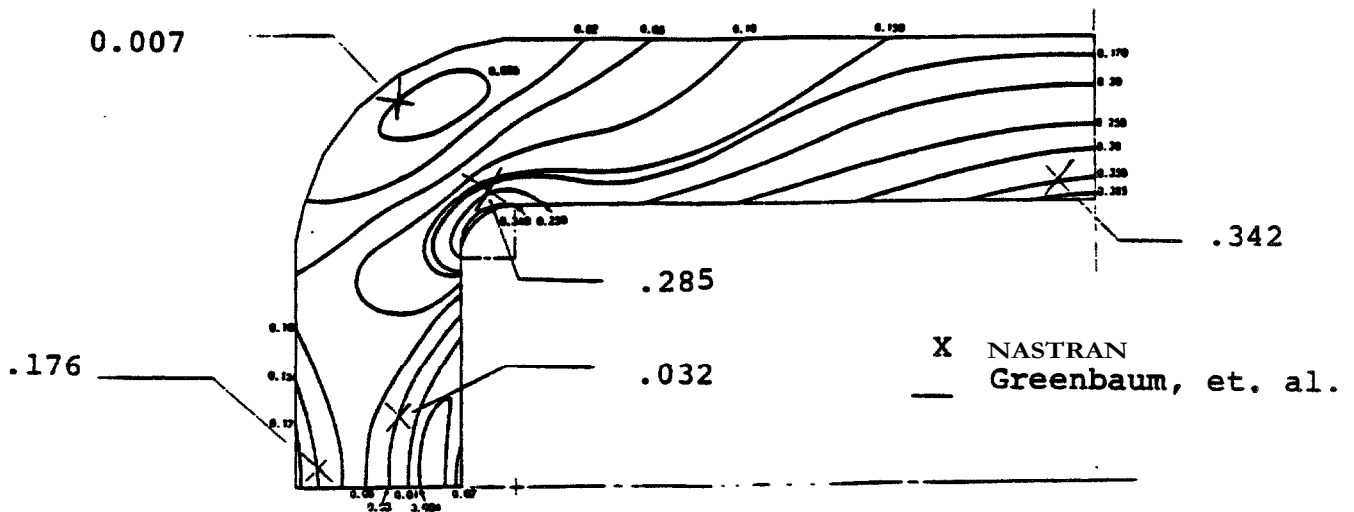


Figure 11.3.5. Effective Creep Strain ($\times 10^{-3}$) after 3 hours of Creep

11.4 TRANSIENT RESPONSE OF AN IMPULSIVELY LOADED CYLINDRICAL PANEL

11.4.1 Problem Description

A self-adaptive algorithm has been developed and implemented for the implicit time integration of the nonlinear transient response analysis (SOL 99 or 129) in the software. In this algorithm, the time increment for the next time step is estimated based on the deformation pattern at the preceding step. The iteration process for equilibrium employs expeditious methods such as quasi-Newton updates and line searches, as well as an adaptive stiffness matrix update strategy for efficiency. Convergence difficulties induced by inadequate prediction of the time step size or the change in nonlinearities are tackled by the bisection method based on the divergence criterion.

The divergence criterion can detect and overcome dynamic buckling problems. For instance, dynamic buckling or snap-through may induce a singular or indefinite dynamic stiffness matrix. The iteration is bound to diverge with a nearly-singular dynamic stiffness matrix, which is detected by the divergence criterion. Then the bisection is activated and the dynamic stiffness is formed again. The dynamic stiffness is likely to be stable at this point and the difficulty in the snap-through is coped with. Otherwise, the bisection will continue until the iteration process is stabilized and a converged solution is found.

A 120° cylindrical panel is 12.56 inches long, with an inside radius of 2.9375 inches and a thickness of 0.125 inches, as shown in Fig. 11.4.1(a). The ends of the panel are simply-supported and the boundaries at the sides are fixed. The panel is impulsively loaded by applying an initial velocity of $v_0 = 5650$ in./sec normal to the region whose length and width are 10.205 inches and 60° in angle, respectively. The objective of this analysis is to illustrate the applicability and effectiveness of the self-adaptive algorithm to a nonlinear problem by examining an impulsively loaded cylindrical panel.

11.4.2 Finite Element Model

This problem features material nonlinearity along with geometric nonlinearity due to large displacements. Due to symmetry, only one half of the cylindrical panel is modeled using shell elements. Three different meshes are used to solve this problem:

- 6x16 mesh
 - 6 circumferential elements, one element every 10 degrees
 - 16 elements along the panel's length
- 8x16 mesh
 - 8 circumferential elements, one element every 7.5 degrees

- 16 elements along the panel's length
- 16x32 mesh
 - 16 circumferential elements, one element every 3.75 degrees
 - 32 elements along the panel's length

The undeformed shape of the 8x16 model is shown in Fig. 11.4.1(b), in which points A and B (at $z=6.28$ in and $z=9.42$ in) indicate the nodal points where displacement responses are measured. The input data for the 8x16 model is listed in Section 11.4.4. SOL 99 (in Version 67) is used for the nonlinear transient analysis of the impulsively loaded cylindrical panel. The analysis is performed in five subcases of equal duration ($200 \mu\text{sec}$). Subcase 1 contains the loading conditions (IC for initial velocity), specified by TIC in the Bulk Data. The remaining four subcases calculate the response to the impulsive loading.

The 6x16, 8x16, and 16x32 meshes used to model the cylindrical panel contain 96, 128, and 512 QUAD4 elements, respectively. The geometry is described in the cylindrical coordinate system, indicated by CORD1C. Boundary conditions, specified using SPC1 entries, include the constraint of all six degrees-of-freedom (DOF) for each GRID point at the fixed side of the panel. Translational DOFs, namely components 1, 2, and 3, for the GRID points at the simply-supported ends of the panel are also constrained. Symmetric boundary conditions are imposed by constraining components 2, 4, and 6 of the GRID points along the side of the panel on the symmetric plane.

Material nonlinearity is signified by the inclusion of MATS1. The material is assumed to be elastic-perfectly plastic and is modeled using the von Mises yield criterion. The geometric nonlinearity is also taken into account by specifying PARAM,LGDISP,1. The solution strategy is controlled by the TSTEPNL data entry, which specifies the parameter ADAPT indicating the adaptive strategy.

11.4.3 Analysis Results and Discussion

The analysis was performed for a duration of 1 msec with an initial time step size of $2 \mu\text{sec}$. During the analysis of the 6x16 model, the automatic time step adjustment was activated three times. The initial step size was quadrupled to $8 \mu\text{sec}$ at the end of 0.04 msec, reduced to $4 \mu\text{sec}$ at 0.22 msec, then reduced to the original size at 0.31 msec. During the analysis of the 8x16 model, the time step size was automatically adjusted nine times. The initial step size was quadrupled at the end of $10 \mu\text{sec}$, then reduced to $4 \mu\text{sec}$ in the next time step when the bisection was activated. The adjustment was activated again, with the time step sizes ranging from $8 \mu\text{sec}$ to $1 \mu\text{sec}$ for the remainder of the analysis. The automatic time step adjustment occurred eleven times in the analysis of the 16x32 model. The bisection was activated at $3 \mu\text{sec}$. Thereafter, Δt was adjusted to step sizes ranging from $4 \mu\text{sec}$ to $0.5 \mu\text{sec}$, having the smallest value during peak displacements.

The deformed shape of the 8x16 model at the peak displacement is shown in Fig. 11.4.1(c). There is a significant improvement in the results as the mesh is refined. Notably, the 16x32 mesh solution is in very good agreement with the experimental results. Refinement of the mesh causes the cylindrical panel model to become more flexible, thereby producing a more realistic displacement response curve. The table below lists the maximum displacement of the midspan normalized to the experimental results, and are compared with a solution using explicit integration [11.12].

Normalized Maximum Displacement at Midspan

mesh size	NASTRAN	Kennedy et al. [11.12]
6x16	0.69	0.78
8x16	0.87	0.86
16x32	0.95	0.93

Kennedy et al. [11.12] stated that the increased number of quadrature points through the thickness resulted in a stiffer behavior in the panel, compared with the solutions for the same problem in Reference [11.13]. In order to examine this point, solutions were obtained using three Gauss points through the thickness by including PARAM,NLAYERS,3 in the Bulk Data. Fig. 11.4.3 compares the displacement response of the 8x16 model at points A and B using three and five Gauss points. Contrary to the observations in [11.12], the number of Gauss points did not have a significant effect on the displacement response. This is attributed to the fact that the dominant reaction force stems from membrane action and the material is perfectly-plastic. When the panel undergoes snap-through, the stresses in the crown line reach the yield point and plastic yielding occurs in compression **throughout** the thickness. Varying the number of Gauss points, therefore, should not greatly affect the results of this problem.

11.4.4 Input Data Listing for the 8x16 Model

```
ID CYROOF, V66 $ SHL, 5/27/87, DSNLTR
SOL 99          $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50       $ PRINT MATRIX TRAILERS AND ITERATIONS
TIME 300        $ FOR VAX
CEND
TITLE=CYLINDRICAL PANEL UNDER IMPACT
ECHO=BOTH
SET 1=108,112
  DISP=1
  SEALL=ALL
  SPC=100
SUBCASE 1
  LABEL=LINEAR ELASTIC REGION
  IC=1
  TSTEPNL=10
SUBCASE 2
  LABEL=RESPONSE TO IMPACT
  TSTEPNL=10
SUBCASE 3
  TSTEPNL=10
SUBCASE 4
  TSTEPNL=10
SUBCASE 5
  TSTEPNL=10
OUTPUT(PLOT)
  CSCALE=1.3
  PLOTTER NAST
  SET 1=ALL
  AXES X,MZ,Y
  VIEW 30.,20.,0.
  MAXI DEFO 1.
  PTITLE=ISOMETR_AIC VIEW
  FIND
PLOT LABEL GRID_A
PLOT TRANSIENT DEFORMATION 1 TIME 4.E-5
OUTPUT(XYLOT)
  XTITLE=                TIME IN SEC.
  XGRID LINES=YES
  YGRID LINES=YES
YTITLE=D I S P.   A T   1 0 8   &   1 1 2   ( T 1 )
XYLOT DISP RESP/108(T1),112(T1)
```

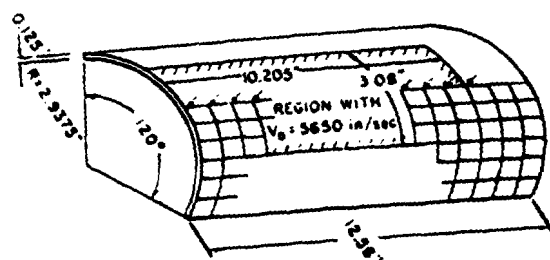
```

BEGIN BULK
PARAM  K6ROT  100.
PARAM  LGDISP 1
TSTEPNL 10      100      2.-6      1      ADAPT
$ MATERIAL AND GEOMETRIC PROPERTIES
MAT1    1      10.5+6      0.33      2.5-4
MATS1   1      PLASTIC      1      1      44000.      +MS1
PSHELL  10      1      0.125      1      1
$ BOUNDARY CONDITIONS
SPC1    100      123456      900      THRU      916
SPC1    100      246      100      THRU      116
SPC1    100      123      100      200      300      400      500      600      +SPC
+SPC    700      800      900      116      216      316      416      516      +SPC1
+SPC1   616      716      816      916
$ LOADING CONDITIONS
TIC     1      101      1      -5650.
=       =      *1      == $
=11 $
TIC     1      201      1      -5650.
=       =      *1      == $
=11 $
TIC     1      301      1      -5650.
=       =      *1      == $
=11 $
TIC     1      401      1      -5650.
=       =      *1      == $
=11 $
TIC     1      501      1      -5650.
=       =      *1      == $
=11 $
$ GEOMETRY AND CONNECTIVITY
CORD1C  1      1000      1001      1010
PLOTTEL 1000      1000      1001
PLOTTEL 1001      1000      1010
GRDSET, , , , , 1 $
GRID    1000      0.      0.      0.      123456
GRID    1001      0.      0.      -1.      123456
GRID    1010      0.      1.      0.      123456
GRID    100      1      3.0      0.      0.
GRID    101      1      3.0      0.      0.785
GRID    102      1      3.0      0.      1.57
GRID    103      1      3.0      0.      2.355
GRID    104      1      3.0      0.      3.14

```

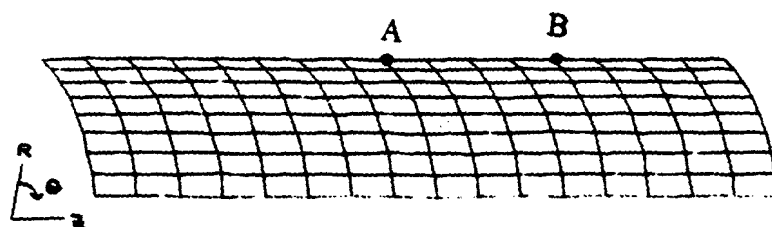
GRID	105	1	3.0	0.	3.925	
GRID	106	1	3.0	0.	4.71	
GRID	107	1	3.0	0.	5.495	
GRID	108	1	3.0	0.	6.28	
GRID	109	1	3.0	0.	7.065	
GRID	110	1	3.0	0.	7.85	
GRID	111	1	3.0	0.	8.635	
GRID	112	1	3.0	0.	9.42	
GRID	113	1	3.0	0.	10.205	
GRID	114	1	3.0	0.	10.99	
GRID	115	1	3.0	0.	11.775	
GRID	116	1	3.0	0.	12.56	
GRID	200	1	3.0	7.5	0.	
=	*1	=	=	=	*0.785	
=15						
GRID	300	1	3.0	15.	0.	
=	*1	=	=	=	*0.785	
=15						
GRID	400	1	3.0	22.5	0.	
=	*1	=	=	=	*0.785	
GRID	500	1	3.0	30.	0.	
=	*1	=	=	=	*0.785	
=15						
GRID	600	1	3.0	37.5	0.	
=	*1	=	=	=	*0.785	
=15						
GRID	700	1	3.0	45.	0.	
=	*1	=	=	=	*0.785	
=15						
GRID	800	1	3.0	52.5	0.	
=	*1	=	=	=	*0.785	
=15						
GRID	900	1	3.0	60.	0.	
=	*1	=	=	=	*0.785	
CQUAD4	101	10	100	200	201	101
CQUAD4	102	10	101	201	202	102
CQUAD4	103	10	102	202	203	103
CQUAD4	104	10	103	203	204	104
CQUAD4	105	10	104	204	205	105
CQUAD4	106	10	105	205	206	106
CQUAD4	107	10	106	206	207	107
CQUAD4	108	10	107	207	208	108
CQUAD4	109	10	108	208	209	109

CQUAD4	110	10	109	209	210	110
CQUAD4	111	10	110	210	211	111
CQUAD4	112	10	111	211	212	112
CQUAD4	113	10	112	212	213	113
CQUAD4	114	10	113	213	214	114
CQUAD4	115	10	114	214	215	115
CQUAD4	116	10	115	215	216	116
CQUAD4	201	10	200	300	301	201
=	*1	=	*1	*1	*1	*1
=14						
CQUAD4	301	10	300	400	401	301
=	*1	=	*1	*1	*1	*1
=14						
CQUAD4	401	10	400	500	501	401
=	*1	=	*1	*1	*1	*1
=14						
CQUAD4	501	10	500	600	601	501
=	*1	=	*1	*1	*1	*1
=14						
CQUAD4	601	10	600	700	701	601
=	*1	=	*1	*1	*1	*1
=14						
CQUAD4	701	10	700	800	801	701
=	*1	=	*1	*1	*1	*1
=14						
CQUAD4	801	10	800	900	901	801
=	*1	=	*1	*1	*1	*1
=14						
ENDDATA						

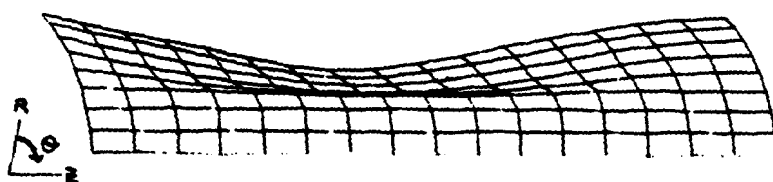


$\rho = 2.5 \times 10^{-4} \text{ lb} - \text{sec}^2/\text{in}^4$
 $E = 10.5 \times 10^6 \text{ psi}$; Young's Modulus
 $\nu = 0.33$; Poisson's Ratio
 $\sigma_y = 4.4 \times 10^4 \text{ psi}$; Yield Stress
 $E_p = 0$; Plastic Modulus
 $v_0 = 5650 \text{ in/sec}$; Initial Velocity

(a) Problem Description (with 6×16 mesh)



(b) 8×16 Model of Half the Panel



(c) 8×16 Deformed Shape at Time = 0.349 msec

Figure 11.4.1 Impulsively Loaded Cylindrical Panel

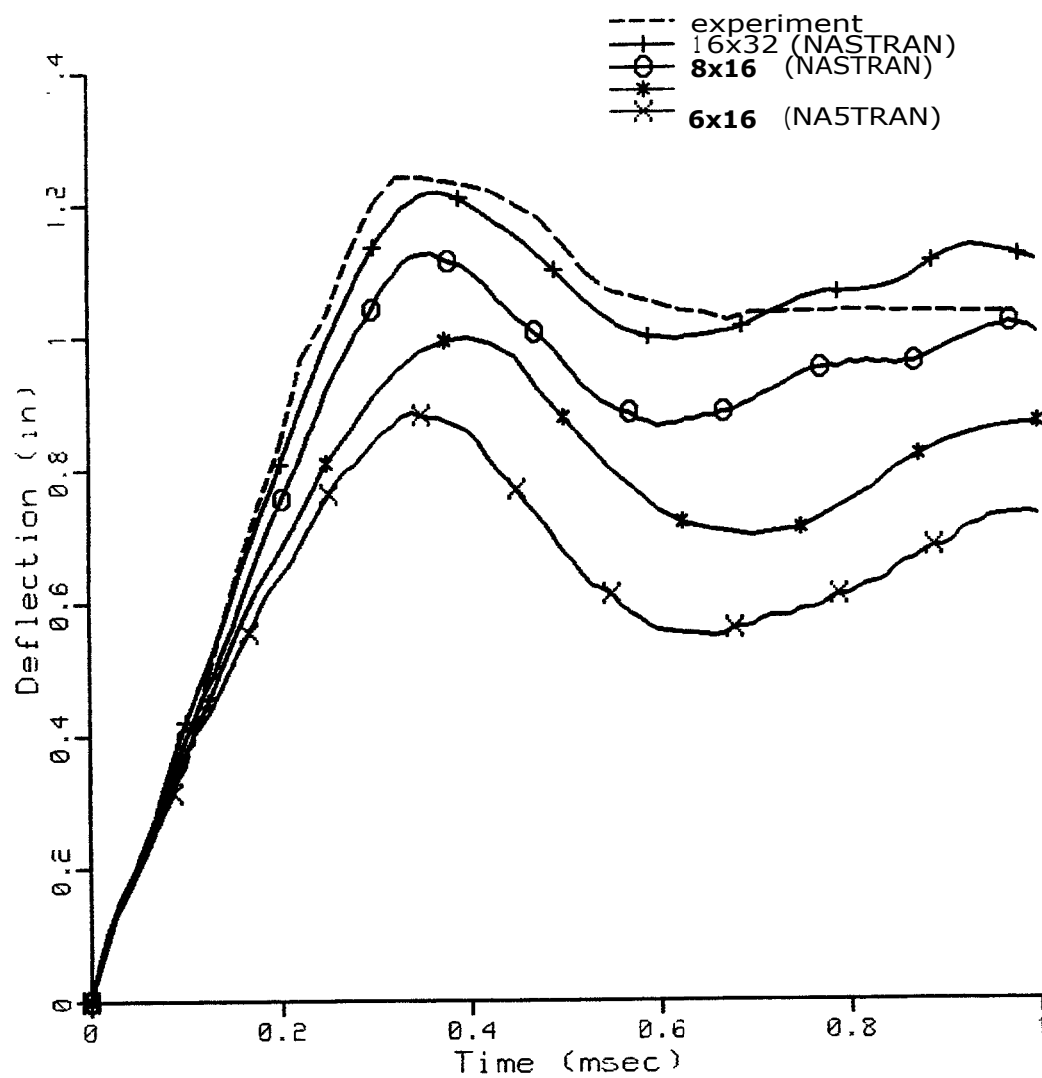


Figure 11.4.2 Displacement Response at Midspan ($z=6.28$ in)

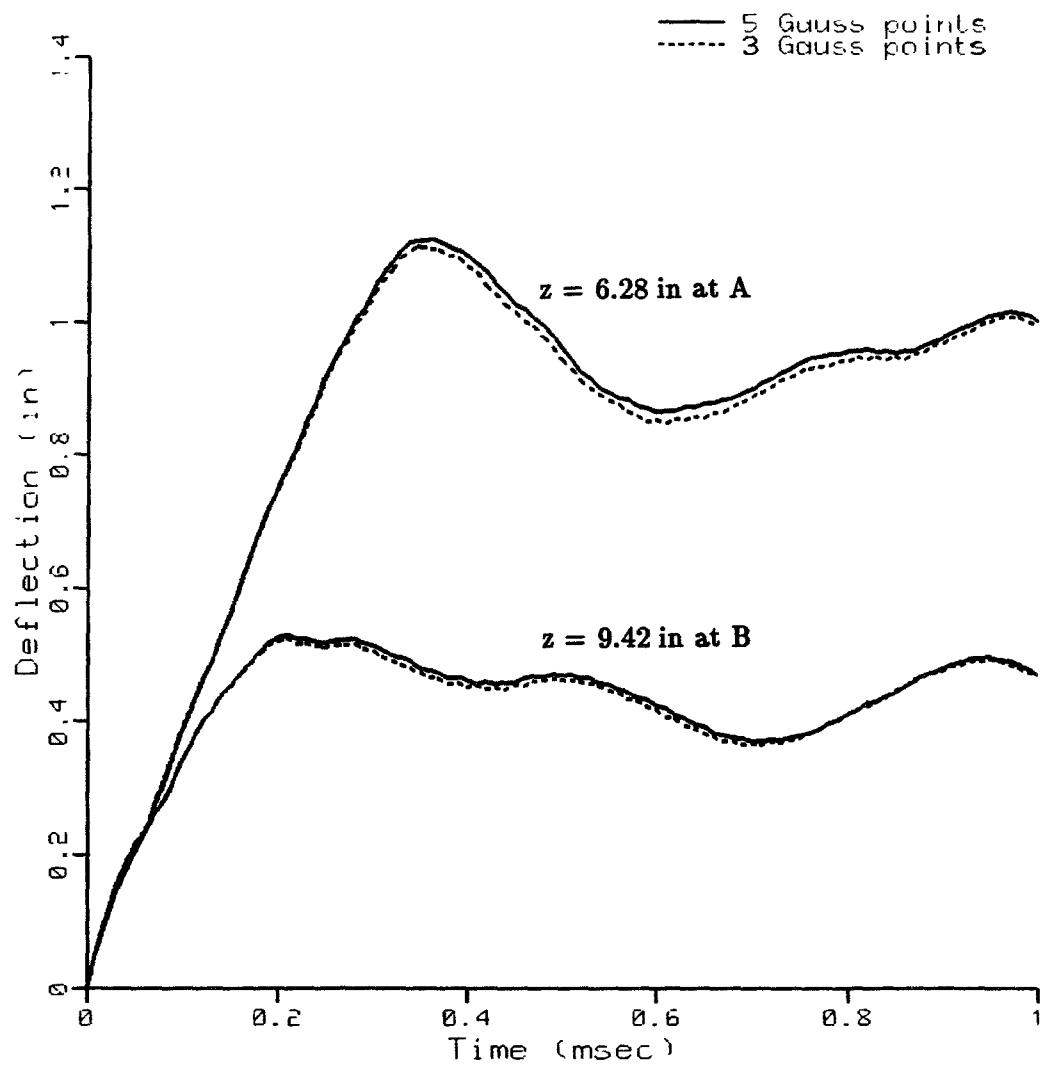


Figure 11.4.3 Comparison of Gauss Integration Schemes
Using 8x16 Model

11.5 CONTACT ANALYSIS FOR THE SHAFT-JOURNAL INTERFACE IN SOL 66

11.5.1 Introduction

The GAP element is used to model the contact between surfaces. It is a nonlinear element available in both the nonlinear static and nonlinear transient solution sequences, SOL 66 and SOL 99, respectively. Difficulties encountered with the use of the GAP include the selection of the GAP properties such as satisfactory open and closed stiffness values, selection of the stiffness matrix update strategy, and efficient reanalysis (restarting). This example examines these issues in SOL 66, using Version 65. All analyses were performed on a VAX 8700. Although the same analysis can be performed more efficiently and effectively with much less user-intervention by using the adaptive GAP elements in Version 67, guidelines and principles suggested in this section is still valid for determining initial penalty values and the preload for the adaptive GAP element.

The GAP is intended to model point-to-point contact (with a bilinear stiffness and a friction) and is applied to the surfaces which may come into contact. When the GAP is open it has an axial stiffness, K_b , which is typically very small and may be 0.0, provided that this does not make the global stiffness matrix singular. While the GAP is open there is no lateral stiffness and no frictional forces. However, there is a large axial stiffness, K_a , when the GAP is closed and significant compressive and shear forces may be present. When compressive force exists in the GAP, it can carry any transverse shear load which is less than the coefficient of friction times the normal load. When the shear load is greater than the friction force, slip occurs.

For the GAP to close, the structure must deform so that the GRIDs attached to the GAP deflect toward each other a distance U_0 . U_0 is the initial separation distance between the surfaces as defined by the PGAP entry, not defined by the distance between the GRID points attached to the GAP element. Currently, the GAP element does not take large displacements into account, i.e., the element coordinate system is not updated. This implies that the orientation of the contact plane remains unchanged during deflection, which is a limitation to be considered before using GAP elements.

11.5.2 Problem Description

The model consists of a hollow steel shaft and a steel journal. The shaft is 20.0 inches long with a 2.0 inch inside diameter and is 0.25 inches thick. It is completely fixed at one end and free at the other end. The 9.0 inch long journal is 0.10 inches thick with 0.5 inch thick flanges at each end. The journal is located 1.0 inch from the fixed end of the shaft and is free to slide along the shaft (this rigid body motion is restrained with a spring in the finite element model). There is a 0.001 inch clearance between the shaft and the journal. A 20000 lb point-load on the

shaft acts radially inward and is located 5.0 inches from the free end. Analyses are performed to determine the forces placed on the journal as the shaft deflects into the journal.

The three fundamental components of the finite element model are the shaft, the journal and the gaps defining the contact surface. By taking advantage of the plane of symmetry present in the structure, only half of it is modeled. A cylindrical coordinate system is used to define 182 GRIDs for the 36 HEXA elements used to model the shaft and the 30 HEXA elements for the journal. One element every 30 degrees is used in the circumferential direction. This is a very coarse finite element mesh but adequate for demonstration purposes.

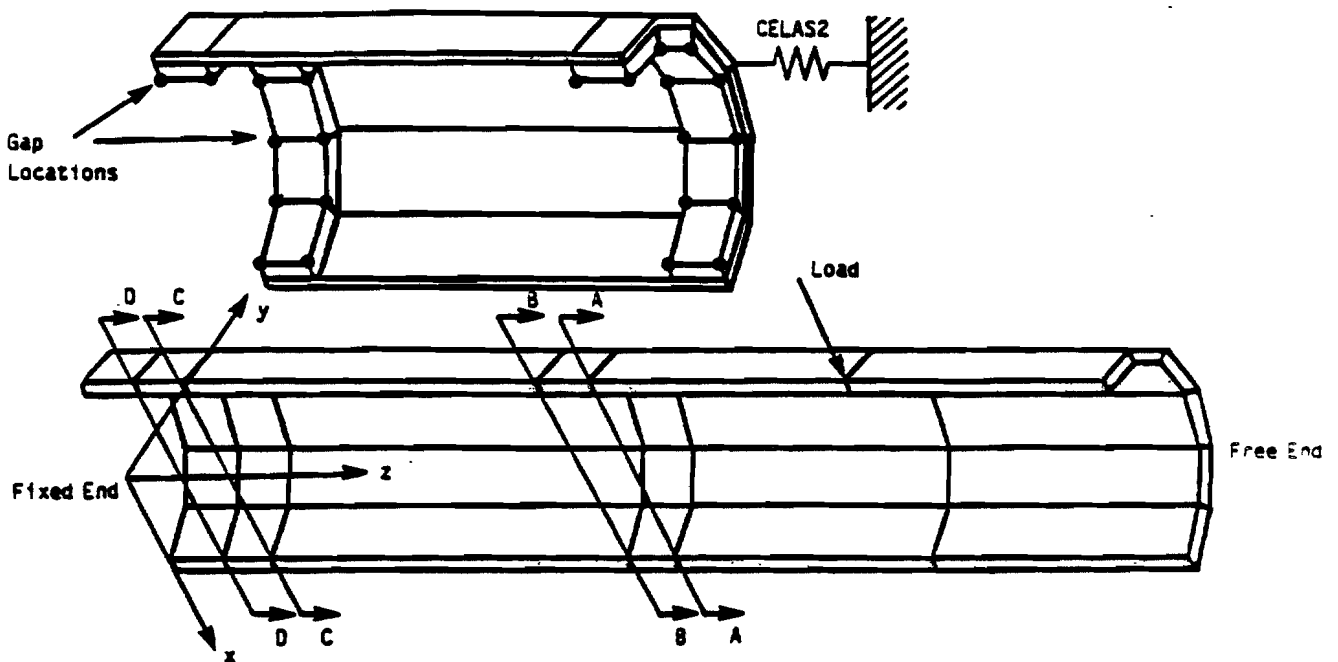


Figure 11.5.1 Finite Element Model Exposing GAP Locations

The boundary conditions include constraining all 6 degrees of freedom (DOF) of each GRID point at the fixed end of the shaft. Symmetric boundary conditions are specified along the edges on the symmetric plane for both the shaft and journal. Rotational DOFs attached to the HEXA elements are also removed via SPCs.

Twenty-eight GAP elements connect the shaft to the journal, one at each GRID location around the journal's flanges. The cylindrical coordinate system is used to define the x-axes of the GAPS to be the T1 or radial direction. The space between the shaft and the journal is 0.001 inches. The initial gap opening, U_0 , is set to 0.001.

With the GAPS open, the stiffness between the shaft and journal is theoretically zero. The constraints and boundary conditions are such that the journal is not restrained from the rigid body motion in three degrees of freedom. These rigid body motions lead to singularities in the stiffness matrix. The rigid body motions are rotation about the y-axis and translation along the x and z-axes. To avoid singularities, a soft spring is attached to the edge of the journal

at GRID 30321 in the z-direction. This CELAS2 scalar spring element elastically restrains translation in the z-direction. The spring constant should be as small as possible to avoid the singularity. A value of 10.0 lb/in. fails, however, an increase in magnitude to 100.0 lb/in. is satisfactory. When the GAPS closed and the stiffness matrix is updated with the larger stiffness value, K_a , the other two rigid body motions are taken care of. However, they remain singular until the contact occurs, thus a small nonzero K_b must be specified.

Due to the symmetry condition, half of the 20000 lb force is applied to GRID 10015 via a FORCE entry. Because there are no material or geometric nonlinearities in the model, the whole load is applied in 1 increment (INC = 1 on the NLPARM entry) in a single SUBCASE. Once the load is applied, the GAPS either open or close and applying the load in multiple increments has no affect on helping the solution to converge nor on the validity of the final solution. To improve convergence, the preload, F_0 on the PGAP entry, is set to 0.10 ($F_0 = U_0 \times K_b$).

When superelements are used in SOL 66, the CLOAD and LSEQ entries must be used to apply the loads to upstream superelements.

11.5.3 Selection of GAP Stiffness

K_b must be large enough to prevent singularities in the initial stiffness matrix and small enough to present realistic answers. Caution is also required in choosing the closed GAP stiffness, K_a . A value that is too large may lead to numerical ill-conditioning, slower convergence or divergence. A K_a value which is too small leads to wrong answers. Experience shows that good choices for K_a and K_b are:

$$\begin{aligned} \text{Closed stiffness } K_a &> 10^3 \times \max[K(\text{journal}), K(\text{shaft})] \\ \text{Open stiffness } K_b &< 10^{-3} \times \min[K(\text{journal}), K(\text{shaft})] \end{aligned} \tag{11.5.1}$$

where $K(\text{journal})$ and $K(\text{shaft})$ are the local stiffnesses of the journal and shaft in the direction of the GAP axis at the GAP location.

Using Roark and Young [11.14], the formulas for circular rings are used to estimate the local structural stiffness. The radial stiffness of the cylinder subject to the radial load can be estimated by

$$K_{local} = \frac{2EI}{0.149R_c^3} \tag{11.5.2}$$

where E is the elastic modulus, I is the area moment of inertia and R_c is the approximate radius to the centroid.

Based on Eq. 11.5.2, the stiffness of the flange portion of the journal in Fig. 11.5.2 is estimated to be 2.68×10^8 lb/in. and for the shaft 1.37×10^5 lb/in. In estimating the local

stiffness of the shaft, the effective length of the shaft is assumed to extend 3 times the thickness of the shaft beyond each edge of the flange.

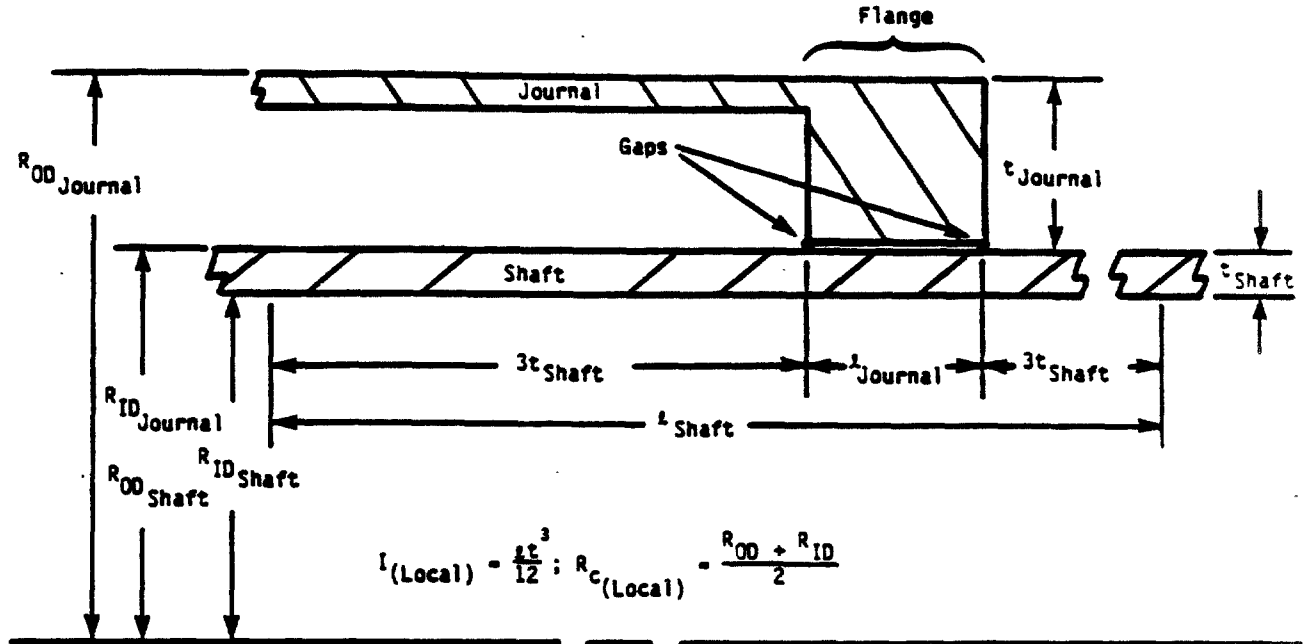


Figure 11.5.2 Detail of Contact Region for Determining Local Stiffness Values

Since two parallel rows of GAPS cover the contact region, half of the stiffness is attributed to each row of GAPS and the estimates for the local stiffness values are divided by 2. The estimate for the local journal stiffness is 1.34×10^8 and for the shaft is $.7 \times 10^5$ lb/in. Hence, from Eq. 11.5.1

$$K_a = 1.0 \times 10^8$$

$$K_b = 1.0 \times 10^2$$

while K_a and K_b should be halved at the plane of symmetry.

11.5.4 Superelement and ASET

To minimize computer time and turn around time for each analysis, the use of superelements (SEs) and ASET entries are implemented. Superelements are used for substructuring. ASET entries are used for Guyan reduction.

The time to perform the analysis is decreased by reducing the size of the stiffness matrix which is updated in SOL 66. A substantial reduction is made using static condensation via ASET entries. By using ASET entries for nonlinear DOFs, only stiffness terms from nonlinear

elements will be retained for nonlinear iteration. DOFs to be retained in the analysis set are placed on ASET entries and all others are automatically omitted. The complement of the ASET entry is the OMIT entry. The use of the ASET or OMIT entry is based upon convenience. The GAP's axial DOFs are the only nonlinear variables in this problem. The T1 DOF from each GRID connected to each GAP are placed on ASET entries. The nonlinear stiffness matrix is thereby reduced from 452 DOF to 56 DOF, reducing substantially the time required for each iteration and stiffness matrix update.

The benefit from using superelements is realized in reanalysis, specifically, when upstream superelements remain unchanged. By placing the nonlinear elements in the residual structure (SE 0) and the linear elements in an upstream superelement (SE 10), only SE 0 requires reanalysis, not upstream SE 10. (The use of either the SESET entry and/or field 9 of the GRID entry to place GRID points in superelements is a matter of convenience.) When restarts are performed with changes to the GAP properties, only the residual structure is updated. For instance, to restart with a new K_a value requires only to change the value of K_a on the PGAP entry and to perform an SEALL operation on SE 0, the residual structure.

Table 11.5.1 compares run times of analyses done with and without static condensation via ASETs and with and without superelements. In order to emphasize the benefit that comes from reducing the size of the nonlinear stiffness matrix and using superelements, two models are compared, a 66-element model and a 900-element model. In the smaller model the ratio of nonlinear DOFs (A-set) to linear DOFs (O-set) is about 1:7. In the big model the ratio of A-set to O-set DOFs is about 1:153.

Table 11.5.1. Normalized values of elapsed CPU time for analyses with and without ASETs and/or SUPERELEMENTS. (Results for each model normalized to 1.)

	All DOFs in ASET by default	Nonlinear GAP DOFs (T1)in ASET	All DOFs in ASET by default	Nonlinear GAP DOFs (T1)in ASET
No SEs	1.00	.88	1.00	.36
GAP GRIDs in SE 0	.87	.89	.55	.53
GAP GRIDs in SE 0 (restart)	.59	.61	.08	.06
No. Elements	66		900	

11.5.5 Analysis Procedure

Several analyses are performed to demonstrate how results vary depending on the value of K_a used. Comparisons are done for two models: a 66-element model with one element every 30 degrees (a very coarse mesh), a 132-element model with one element every 15 degrees. The results are compared in Fig. 11.5.3. It is considered that the best possible solution is achieved when the GAP forces become stationary with respect to K_a .

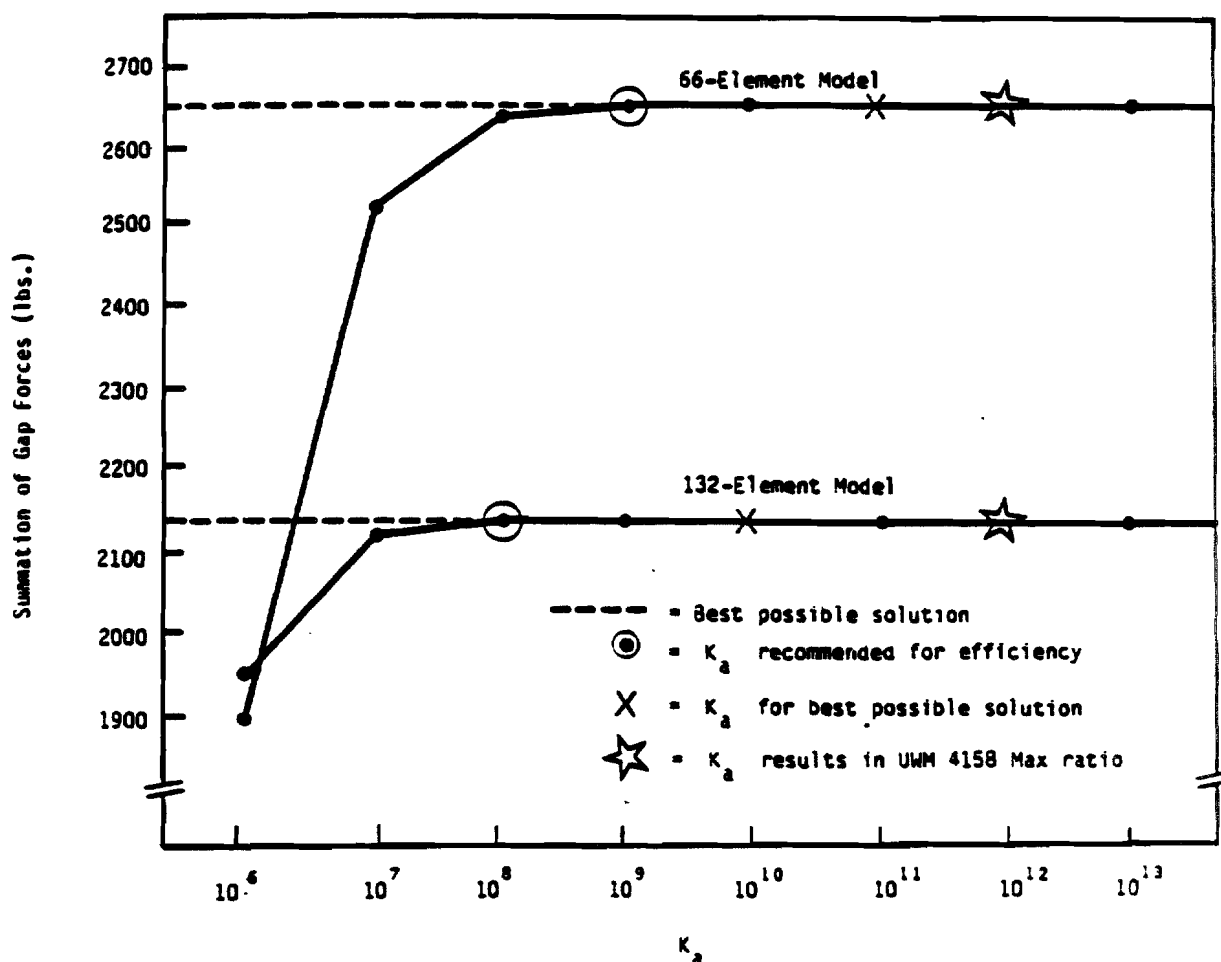


Figure 11.5.3. K_a vs. axial GAP forces for 66-element and 132-element models

The NLPARM entry provides for three types of nonlinear solution methods (ITER, SEMI and AUTO), which control the stiffness matrix update strategy. The ITER method which updates the stiffness matrix after each iteration (KSTEP=1) is referred to as full Newton's method. Variations of all three methods are used in several analyses and are tabulated in Table 11.5.2, which summarizes results from restarts of the residual structure. Only axial GAP DOFs are in the nonlinear stiffness matrix via ASET entries. All cases use $K_a = 10^9$ and $K_b = 100$. The severe nonlinear effects (rapidly changing stiffnesses) make it difficult to obtain a converged solution. Inspection of the convergence factors and line search data from

the nonlinear iteration output is required to make a judicious selection of NLPARM parameters for better convergence.

Table 11.5.2 Variation of Iteration Strategy in Version 65

(Numbers in parenthesis or blank indicate default value.)

Note that the performance is significantly improved in Version 67.

Case	KMETHOD	KSTEP	MAXDIV	MAXITER	MAXQN	MAXLS	SOLUTION
1	AUTO	(7)	(2)	(20)	(20)	(5)	stopped due to 2 diverging sols
2	AUTO					10	stopped due to 2 diverging sols
3	AUTO		7				stopped due to 2 diverging sols
4	AUTO		7			10	stopped due to MAXITER limit
5	AUTO		7	30	30	15	stopped due to 2 diverging sols
6	AUTO		10	30	30	20	converges in 82. sec
7	SEMI		1				stopped due to 2 diverging sols
8	SEMI		7				stopped due to MAXITER limit
9	SEMI		7	30	30		stopped due to MAXITER limit
10	SEMI		7	30	30	10	stopped due to MAXITER limit
11	SEMI		7	30	30	15	converges in 80. sec
12	ITER						stopped due to 2 diverging sols
13	ITER	2					stopped due to MAXITER limit
14	ITER	2				0	stopped due to MAXITER limit
15	ITER	1					converges in 63. sec
16	ITER	1			0	0	converges in 43. sec

11.5.6 Summary of Results

The load deforms the shaft into an elliptical shape near the free end which then contacts the journal over a small area at a few locations shown in Fig. 11.5.4. At these locations the GAPS close and the stiffness changes from K_b to K_a .

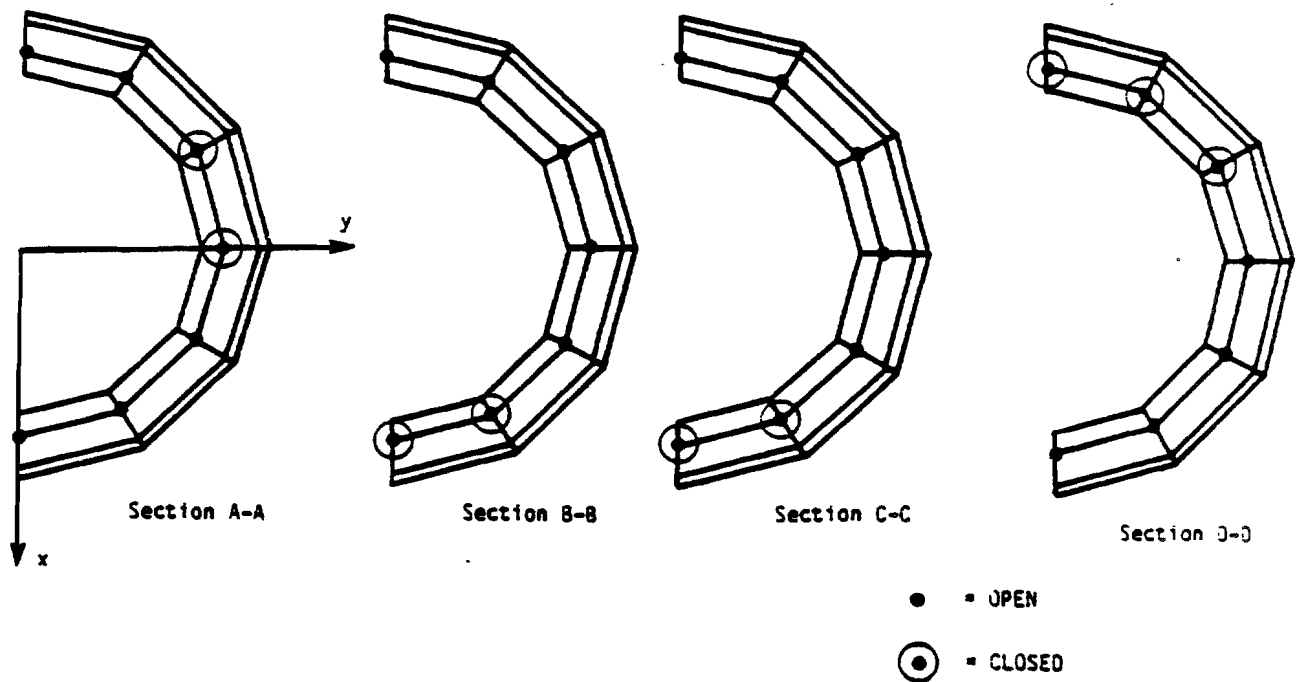


Figure 11.5.4 Sections from Fig. 11.5.1 Indicating Which GAPS Close

Fig. 11.5.3 shows that the 66-element model converges to the best possible solution with $K_a = 10^{11}$. However, acceptable results are achieved with $K_a = 10^8$ and the answers are less than 5% off when K_a is as small as 10^7 . $K_a = 10^8$ is considered the smallest value which provides a good solution, hence recommended for efficiency. The best possible solution for the 132-element model is achieved over a larger range of K_a values. A value of K_a spanning six orders of magnitude provides results that differ by less than 1%. The best possible solution is achieved with $K_a = 10^{10}$ while $K_a = 10^7$ gives a solution that is less than 1% different from the best possible solution. For the 132-element model, $K_a = 10^7$ is recommended for efficiency. The range of acceptable K_a values increases as the mesh is refined. This trend continues until the "best" mesh is used, which in this case is about 1 element every 10 degrees in the circumferential direction.

All three of the iteration methods determined identical solutions as shown in Table 11.5.2. However, the default values on the NLPARM entry do not yield a converged solution. The full Newton's method found the solution most efficiently (which is not the case in Version 67). The time to reach a converged solution was further minimized by restricting the number of line searches done. Optimal iteration strategy varies with each individual problem. Nevertheless, the full Newton's method is considered the most conservative and reliable method.

The use of ASETs to reduce the problem size for the nonlinear iterations is strongly recommended if the model has high ratios of linear to nonlinear elements. The large 900-element model in Table 11.5.1 uses 64% less CPU time when the nonlinear stiffness matrix is reduced via ASET entries to include only the essential nonlinear terms.

Superelements are especially effective when restarts are performed. Without superelements, the entire analysis must be repeated. However, when superelements are combined with ASETs for the large model, the restart uses only 6% of the CPU as in the original analysis. These numbers will vary depending on factors such as: number of stiffness matrix updates, iterations, ratio of linear to nonlinear elements.

11.5.7 Friction in the GAP

The effects of friction in the GAPs are considered in a separate analysis. The analysis is of the same model but with two loading conditions. The initial load closes the GAPs (SUBCASE 1), the second one loads the closed GAPs in shear (SUBCASE 2 and 3). When the shear force is greater than the friction force, slip occurs. Before the full slip condition, the shear force in each GAP equals K_t (transverse stiffness from PGAP entry) times the shear displacement. K_t is an elastic stiffness (ideally rigid) used to simulate the stationary condition under the critical load at which slip occurs. The data is nearly identical to the data in Table 11.5.3 but with modifications to the PGAP and ASET entries, additional load cases and NLPARM entries.

Friction coefficients of 0.25 are used for the GAP y and z-directions. With friction included, the GAP's shear DOFs (y and z) become nonlinear DOFs and, as such, need to be included in the analysis set. DOFs 2 and 3 for each GRID attached to a GAP are placed on ASET entries (except where SPCed).

The CELAS spring used to restrain the journal is not a structural member, however its stiffness (100 lb/in) provides an adequate stability to the model. The transverse stiffness, K_t , connected in series with the CELAS should be the predominant stiffness to resist the transverse loading. As such, the value of K_t should be orders of magnitude larger than 100. A parametric study using K_t equal to 10^3 , 10^4 , and 10^5 found the minimum load at which full slip occurs. With K_t equal to 10^4 and 10^5 , the same load produces slip. At K_t equal to 10^3 , a slightly higher load is required for slip. At very large K_t , e.g. 10^5 or higher, convergence is difficult to achieve and caution is needed in selecting the loads and iteration parameters. For best efficiency, K_t which is two orders of magnitude larger than the spring stiffness is proven sufficient (i.e., $K_t = 10^4$).

The load in SUBCASE 1 is the concentrated force. The second load, SUBCASEs 2 and 3, is a line load along the edge of the journal acting in the +z-direction. A 90 lb force is applied to each of the seven GRIDs along the edge of the journal. All but one of the GAPs (no. 43) slip. In SUBCASE 3 the load is increased to 97 lbs in 1 lb increments. The journal slips completely at 95 lbs with a total shear force equal to about 1311 lbs. As the remaining load increments are applied, the journal does not experience large displacements as would be expected in the full slip condition. This is due to the spring attached along the z-axis.

Table 11.5.3 Input Data Listing for Case 16.

```

ID GAP, APPNOTE $ BY BOB LOUWERS
SOL 66
TIME 10
$ CONCENTRIC CYLINDERS CONNECTED BY GAPS.
CEND
TITLE = KA = 1.E+9; KB = 100.; INITIAL GAP OPENING = 0.001; SE MODEL.
SUBTITLE = METHOD = ITER WITH 1 INC; K UPDATE AFTER EACH ITERATION
LABEL = ASETS AND SUPERELEMENTS
$SET 111 = 0    $ FOR RESTARTS
$SEALL = 111    $ FOR RESTARTS
SEALL = ALL
SUPER = ALL
ECHO = BOTH
SPC = 100
LOADSET = 950
$STRESS = ALL          $ ADDITIONAL OUTPUT REQUESTS
$DISPLA = ALL          $ FOR UPSTREAM DATA RECOVERY
$ELFORC = ALL
SET 10 = 10 THRU 50    $ FORCES IN GAP ELEMENTS
ELFORCE = 10
SUBCASE 1
CLOAD = 901
NLPARM = 2001
$
$      Remove "$" from the following Case Control entries to include
$      shear loads when friction effects are included in the Bulk Data
$      as discussed in Section 7.0.
$
$SUBCASE 2
$ CLOAD = 902          $ SHEAR LOADING FOR GAP ELEMENTS
$ NLPARM = 2002
$SUBCASE 3
$ CLOAD = 903          $ SHEAR LOADING FOR GAP ELEMENTS
$ NLPARM = 2003
OUTPUT(PLOT)
*
*
*
BEGIN BULK
$
$ USED TO DEFINE SHAFT, JOURNAL AND THE RADIAL DIRECTION OF GAP AXES.
$

```

CORD2C 1 0 0. 0. 0. 0. 0. 1. +CORD
 +CORD 1. 0. 0.
 \$
 \$ GRID POINTS WHICH DEFINE SHAFT AND JOURNAL.
 \$
 GRID 10000 1 2. 0.0 0.0 1 10
 *
 *
 *
 GRID 30621 1 2.75 180. 10. 1 10
 \$
 \$ CHEXA ELEMENTS WHICH DEFINE SHAFT AND JOURNAL.
 \$
 CHEXA 100 12 10000 10001 10011 10010 10100 10101 +M00000
 +M00000 10111 10110
 *
 *
 *
 CHEXA 405 12 20511 30510 30520 20521 20611 30610 +M00065
 +M00065 30620 20621
 \$
 MAT1 13 3.+7 .3
 PSOLID 12 13
 \$
 \$ CGAPS DEFINE CONNECTIVITY - USE COORD SYSTEM 1 TO DEFINE GAP X AXES.
 \$
 CGAP 10 15 10011 20000 1
 *
 *
 *
 CGAP 46 15 10614 30601 1
 \$
 \$
 \$ U0 F0 KA KB
 PGAP 15 0.001 .10 1.E9 1.E2
 \$
 \$ SYMMETRIC BOUNDARY CONDITIONS.
 \$
 SPC1 100 246 10000 10001 10002 10003 10004 10005 +M00068
 +M00068 10006 10010 10011 10012 10013 10014 10015 10016
 *
 *
 *
 SPC1 100 246 30600 30601 30610 30611 30620 30621

```

$
$ CONSTRAIN ROTATIONAL DOFS ON ALL SOLID HEXA ELEMENTS.
$

SPC1    100      456      10000   THRU    40000
$
$ BOUNDARY CONDITIONS FOR FIXED END OF SHAFT.
$
SPC1    100      123456  10000   10010   10100   10110   10200   10210   +M00066
+M00066 10300   10310   10400   10410   10500   10510   10600   10610
$
$ PLACE AXIAL DOFS CONNECTED TO GAPS IN THE ANALYSIS SET.
$
ASET1   1        10011   10012   10013   10014
                *
                *
                *
ASET1   1        21100   21101   31100   31101
$
$ PLACE GRIDS ATTACHED TO GAPS IN THE RESIDUAL STRUCTURE.
$
SESET   0        10011   10111   10211   10311   10411   10511   10611
                *
                *
                *
SESET   0        30001   30101   30201   30301   30401   30501   30601
$
$ SOFT SPRING TO PREVENT RIGID BODY MOTION OF JOURNAL IN T3 DIRECTION.
$
CELAS2  76      100.0   30321   3
$
$ CLOAD, LSEQ REQUIRED FOR LOADING WHEN SUPERELEMENTS ARE USED.
$
CLOAD   901      1.      1.      975
LSEQ    950      975     1000
$
FORCE   1000     10015           10000.0 -1.      0.      0.
$
$ NLPARM PROVIDES PARAMETERS FOR NONLINEAR ANALYSIS ITERATION STRATEGY.
$
$          INC          KMETHOD KSTEP
NLPARM  2001      1          ITER    1          +NLP
$          MAXQN  MAXLS

```

```

+NLP
$
$ The following Bulk Data entries defining the shear loads and NLPARM entries
$ for SUBCASE 2 and 3 are used when SUBCASE 2 and 3 are included.
$
$ SUBCASE 2 LOADS AND NLPARM CARD
$
CLOAD 902 1. 1. 976
LSEQ 950 976 1002
FORCE 1002 10015 10000.0 -1. 0. 0.
FORCE 1002 20010 90. 0. 0. 1.0
*
*
*
FORCE 1002 20610 90. 0. 0. 1.0
NLPARM 2002 4 ITER 1 +NLP2

+NLP2
$
$ SUBCASE 3 LOADS AND NLPARM CARD
$
CLOAD 903 1. 1. 977
LSEQ 950 977 1003
FORCE 1003 10015 10000.0 -1. 0. 0.
FORCE 1003 20010 97. 0. 0. 1.0
*
*
*
FORCE 1003 20610 97. 0. 0. 1.0
NLPARM 2003 7 ITER 1 YES +NLP3
+NLP3
$
$ Remove the "$" to include the following required shear DOFs when
$ friction effects are used in SUBCASE 2 and 3.
$
$ASET1 3 10011 10012 10013 10014
$ASET1 23 10111 10112 10113 10114
*
*
*
$ASET1 23 20500 20501 30500 30501
$ASET1 3 20600 20601 30600 30601

```

```

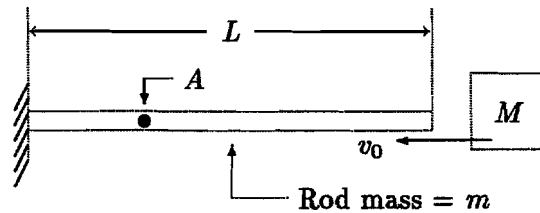
$
$   To include transverse stiffness and friction effects for SUBCASE 2
$   and 3, replace the above PGAP card with the following PGAP card.
$
$           Uo      Fo      Ka      Kb      Kt      mu-y      mu-z
$PGAP      15      0.001   .10     1.E9     1.E2     1.E4     0.25     0.25
ENDDATA

```

11.6 IMPACT ANALYSIS IN SOL 99: STRESS WAVE PROPAGATION IN AN ELASTIC ROD

11.6.1 Problem Description

A rod with a fixed end is struck by a moving mass at the other end. Let v_0 be the initial velocity of the mass which is considered rigid. The velocity of particles at the end of the rod at the instant of impact ($t = 0$) is v_0 . Determine the time history of stress and displacement at various cross sections of the rod during the impact. Find the duration of the impact and the maximum stress in the rod for various values of mass ratio, M/m .



$L = 100.0$ m, rod length
 $A = 1.0$ m², cross-sectional area of the rod
 $E = 10^3$ N/m², Young's Modulus
 $\nu = 0.3$, Poisson's ratio
 $\rho = 0.1$ Kg/m³, mass density
 $m = \rho AL = 10.0$ Kg, rod mass

11.6.2 Theoretical Solution

Let M be the mass of the moving body and m be the total mass of the rod. Denoting the stress at the free end of the rod by σ and the velocity of the body by v , we have

$$\sigma = \rho cv = E \frac{v}{c} \quad \text{with} \quad c = \sqrt{\frac{E}{\rho}} \quad (11.6.1)$$

where $c = \sqrt{E/\rho}$ is the velocity of sound in rod material. As such, the stress in the wave is determined by the ratio of the two velocities. The initial compressive stress in the rod is

$$\sigma_0 = v_0 \sqrt{E\rho} \quad (11.6.2)$$

Considering the force balance at the free end of the rod during the impact, the equilibrium equation can be written as

$$M \frac{dv}{dt} + \sigma A = 0 \quad (11.6.3)$$

where A is the cross-sectional area of the rod.

Let the time interval (T) for the stress wave, traveling a complete cycle in the rod, be

$$T = \frac{2L}{c} \quad (11.6.4)$$

Solving the equilibrium equation for stress at the end struck in various time intervals, we obtain [11.15]

$$\begin{aligned} \sigma &= S_0(t) && \text{for } 0 \leq t < T \\ \sigma &= S_1(t) + S_0(t - T) && \text{for } T \leq t < 2T \\ \sigma &= S_2(t) + S_1(t - T) && \text{for } 2T \leq t < 3T \\ &\cdot && \\ &\cdot && \\ &\cdot && \\ &\text{etc.} && \end{aligned} \quad (11.6.5)$$

where

$$\begin{aligned} S_0(t) &= \sigma_0 e^{-2(m/M)(t/T)} \\ S_1(t) &= S_0(t) + \sigma_0 e^{-2(m/M)(t/T-1)} \left[1 + 4 \frac{m}{M} \left(1 - \frac{t}{T} \right) \right] \\ S_2(t) &= S_1(t) + \sigma_0 e^{-2(m/M)(t/T-2)} \left[1 + 8 \frac{m}{M} \left(2 - \frac{t}{T} \right) + 8 \left(\frac{m}{M} \right)^2 \left(2 - \frac{t}{T} \right)^2 \right] \end{aligned}$$

Since the stress wave is reflected when it arrives at the fixed end of the rod, the stress value is twice the intensity of the incident wave. The general expression for the stress at the fixed

end during any time interval $(\frac{1}{2} + n)T \leq t < (\frac{3}{2} + n)T$ is therefore

$$\sigma(t) = 2S_n(t - \frac{T}{2}) \quad (11.6.6)$$

11.6.3 Software Solution

The rod is modeled with twenty CONRODs and a GAP element. One end of the rod is constrained completely and the other is left free axially. One CONM2 is used to represent the rigid mass, and the mass value is varied for various ratios of m to M . A GAP element connects the CONM2 and the free end of the rod with a zero initial opening. An initial velocity of $v_0 = -0.1m/sec$ is specified for both the CONM2 and the free end nodal point using TIC card. A time step is chosen so that twenty vibration modes of the rod model can be adequately represented. A small damping (0.4% at resonance) is introduced in the rod elements to smooth out the undesired oscillations. The consistent mass matrix option (PARAM, COUPMASS) is recommended for more stable responses.

11.6.4 Comparison of Results

The instant when the stress at the free end (or force in the GAP element) vanishes signifies the end of the impact. The durations of impact for four distinct values of M/m are determined from the software analyses and compared with the calculations of Saint-Venant in the following table.

Table 11.6.1 Duration of Impact (sec)

M/m	1	2	4	6
Theory	3.068	4.708	5.900	7.419
NASTRAN	3.065	4.708	6.700*	7.420
% Difference	0.1 %	0.0 %	13.56 %*	0.01 %

* The discrepancy is explained below.

The results given in the above table are obtained using the prescribed model of 20 CONRODs. They match the theoretical values closely except for $M/m = 4$. It was discovered, however, that this appreciable discrepancy is caused by a limitation of the theoretical solution. The theoretical solution is valid only if the mass body stays in contact with the rod. With a more refined model, the software reveals that there is a momentary opening near the theoretical solution (5.86 seconds in the software) while the real departure of the mass body from the rod occurs at 6.7 seconds. Thus, we may conclude that the software predicts the true end of impact.

Time histories of displacement at various grid points and stresses at various elements are obtained from analyses for several different values of M/m . Illustrations in Fig. 11.6.1 and Fig. 11.6.2 are typical examples of the transient responses associated with the free end of the rod. A plot of displacement vs. time in Fig. 11.6.1 for the mass point and the free end point clearly shows a good departure of the mass body, consistent with the theory. In Fig. 11.6.2, a stress vs. time plot for the element at the free end is given in comparison to the theoretical one for the end point. It should be noted that due to the stress gradients in both time and space, the element stress output can not reproduce the theoretical stress curve exactly at a grid point.

The maximum stress during impact always occurs at the fixed end, but in different time intervals, for different values of M/m . This shifting in time interval causes the discontinuity of the slope of σ_{\max}/σ_0 vs. M/m curve shown in Fig. 11.6.3. In general, element stresses are lower than the theoretical values due to discretization error as expected. For each continuous and smooth curve segment, the result improves in agreement with the theoretical solution as it approaches the end of the segment. It is shown that a refined model improves the results slightly.

11.6.5 Input Data Listing

```

NASTRAN BUFFPOOL=37
NASTRAN SYSTEM(66)=112
INIT MASTER(S)
INIT SCRATCH(MEM=150)
ID HAMMER,V66 $ SSH 6/22/87, SHL 5/28/87 $ TKT 5-10-1985, SHL 6/4/85
TIME 30          $ FOR VAX
SOL 99           $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50        $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE= TRANSIENT RESPONSE OF SHOCK WAVE IN BAR -- IMPACT
SUBTITLE= BAR WITH FIXED END STRUCK BY A MOVING MASS AT THE OTHER END
  SET 1=21,99
  SET 2=101,120,899
  SET 3=99,21
  SEALL=ALL
  DISPLACEMENT=1
  VELOCITY=3
  STRESS=2
SUBCASE 1 $ UP TO 3 SEC.
  IC=1
  TSTEPNL=10
SUBCASE 2 $ UP TO 5 SEC.
  TSTEPNL=20

```

```

$SUBCASE 3
$ TSTEPNL=30
OUTPUT(XY PLOT)
  CSCALE=1.3
  XAXIS=YES
  YAXIS=YES
  XTITLE=TIME
  YTITLE=DISPLACEMENT
  TCURVE= DISP.(T1) AT MASS PT.(GP99),FREE END(GP21)
XY PLOT DISP /99(T1),21(T1)
  YTITLE=STRESS
  TCURVE= STRESS AT FREE END (ELEMENT 120)
XY PLOT STRESS /120(2)
  TCURVE= STRESS AT FIXED END (ELEMENT 101)
XY PLOT STRESS /101(2)
  TCURVE= FORCE IN THE GAP (ELEMENT 899)
  YMAX=2.
XY PLOT STRESS /899(2)
BEGIN BULK

```

	\$.....	2.....	3.....	4.....	5.....	6.....	7.....	8.....	9.....	10.....
GRID	1			0.0	0.0	0.0				123456
GRID	2			5.	0.0	0.0				23456
GRID	3			10.	0.0	0.0				23456
GRID	4			15.	0.0	0.0				23456
GRID	5			20.	0.0	0.0				23456
GRID	6			25.	0.0	0.0				23456
GRID	7			30.	0.0	0.0				23456
GRID	8			35.	0.0	0.0				23456
GRID	9			40.	0.0	0.0				23456
GRID	10			45.	0.0	0.0				23456
GRID	11			50.	0.0	0.0				23456
GRID	12			55.	0.0	0.0				23456
GRID	13			60.	0.0	0.0				23456
GRID	14			65.	0.0	0.0				23456
GRID	15			70.	0.0	0.0				23456
GRID	16			75.	0.0	0.0				23456
GRID	17			80.	0.0	0.0				23456
GRID	18			85.	0.0	0.0				23456
GRID	19			90.	0.0	0.0				23456
GRID	20			95.	0.0	0.0				23456
GRID	21			100.	0.0	0.0				23456
GRID	99			100.	0.0	0.0				23456

\$ ROD ELEMENTS

CONROD	101	1	2	100	1.0		
CONROD	102	2	3	100	1.0		
CONROD	103	3	4	100	1.0		
CONROD	104	4	5	100	1.0		
CONROD	105	5	6	100	1.0		
CONROD	106	6	7	100	1.0		
CONROD	107	7	8	100	1.0		
CONROD	108	8	9	100	1.0		
CONROD	109	9	10	100	1.0		
CONROD	110	10	11	100	1.0		
CONROD	111	11	12	100	1.0		
CONROD	112	12	13	100	1.0		
CONROD	113	13	14	100	1.0		
CONROD	114	14	15	100	1.0		
CONROD	115	15	16	100	1.0		
CONROD	116	16	17	100	1.0		
CONROD	117	17	18	100	1.0		
CONROD	118	18	19	100	1.0		
CONROD	119	19	20	100	1.0		
CONROD	120	20	21	100	1.0		
\$ GAP ELEMENT							
CGAP	899	90	21	99	0.	1.	0.
PGAP	90			1.E+5			
\$ MASS ELEMENT							
CONM2	999	99		10.			
\$ INITIAL CONDITION							
TIC	1	99	1		-0.1		
TIC	1	21	1		-0.1		
\$ MATERIAL PROPERTY OF ROD							
MAT1	100	1.E+3		.3	0.1		.002
PARAM	W4	1.570796					
PARAM	COUPMASS1						
\$ TIME STEP DEFINITION							
TSTEPNL	10	120	0.025	1	AUTO		10
TSTEPNL	20	80	0.025	1	AUTO		10
TSTEPNL	30	75	0.010	2	AUTO		10
ENDDATA							

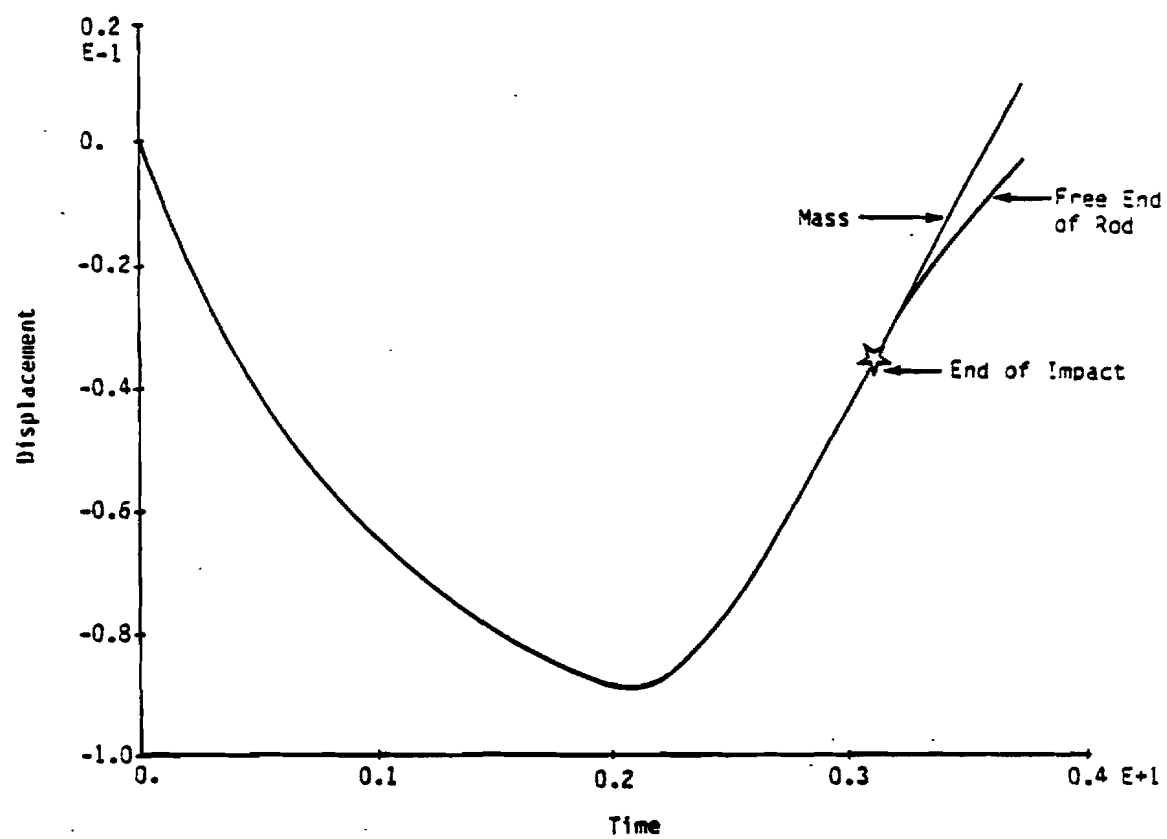


Figure 11.6.1. Displacement vs. Time for $M/m = 1$.

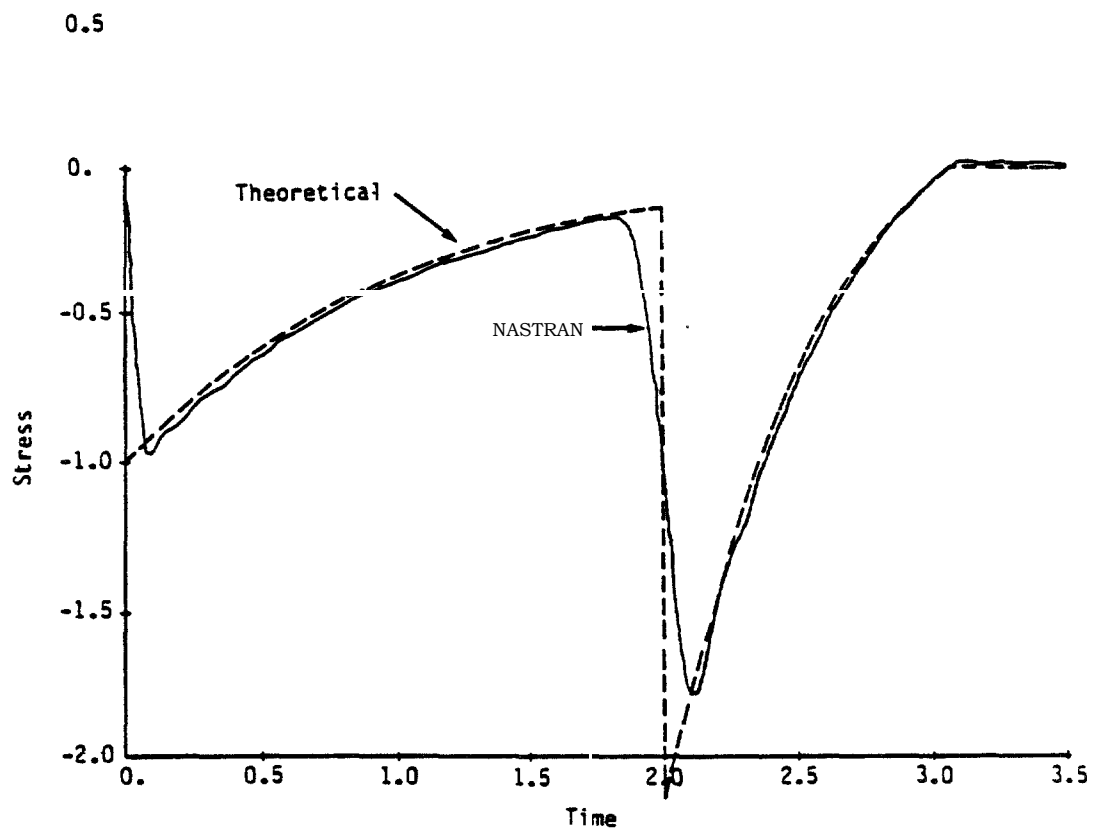


Figure 11.6.2 Comparison of Free End Stress
(Time History for $M/m = 1$)

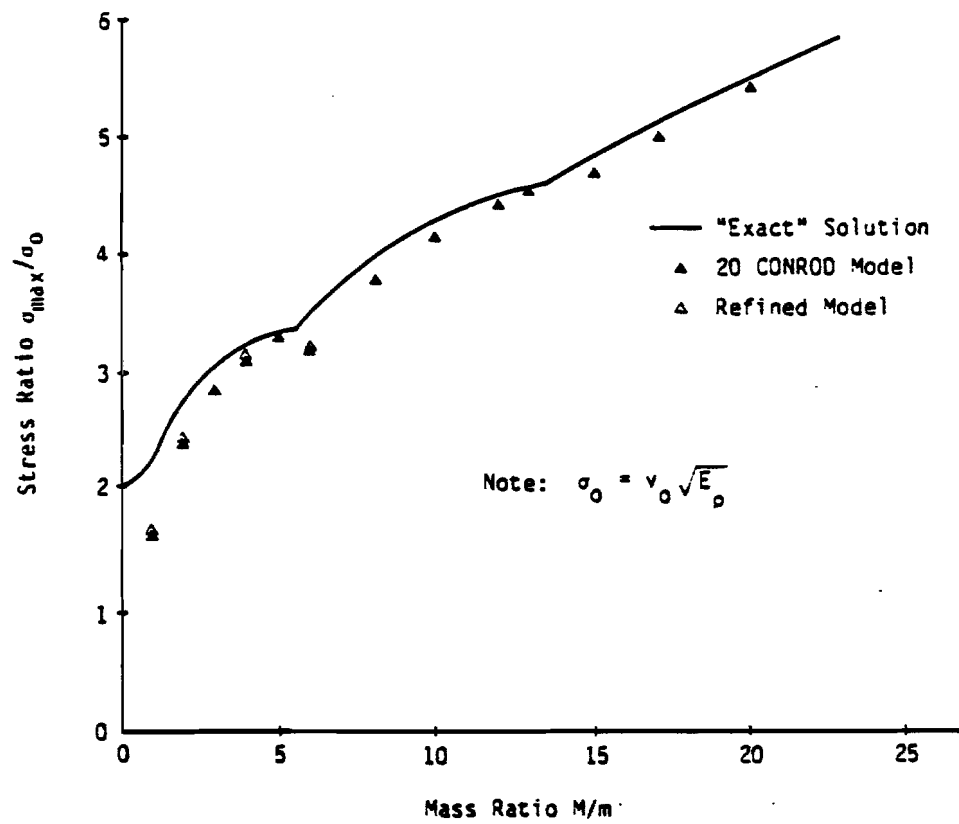


Figure 11.6.3. Comparison of Impact Stresses.

11.7 EFFECTS OF K6ROT ON GEOMETRIC STIFFENING OF THE THIN SHELL MODEL

11.7.1 Theoretical Basis for Parameter K6ROT

Stiffness is not defined for the rotation normal to the surface of shell elements (QUAD4 and TRIA3). Unlike linear analysis, this degree of freedom can not be constrained in the geometric nonlinear analysis. It is noted that QUADR and TRIAR elements have stiffness in this direction, but they are not equipped with nonlinear features. In order to remove the singularity in the out-of-plane rotation (the sixth DOF), an artificial stiffness K_θ , proportional to the user-specified parameter K6ROT, is provided for the normal rotation in the QUAD4 and TRIA3 shell elements [11.16]. The pseudo stiffness K_θ is added by the parameter K6ROT to the relative rotation of the element to grid rotation as follows:

$$K_\theta \text{ for } (\Omega_z - \theta_z) = 10^{-6} * G * t * |J| * \text{K6ROT}$$

with

θ_z = rotation of a GRID from global displacement

$$\Omega_z = \frac{1}{2} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right); \text{ rotation measured in the element}$$

where G is the shear modulus, t is the thickness, and J is the Jacobian, representing the area of the element. The requirements for K_θ are

- The unit should be in moments.
- Pass the constant strain patch test.
- No effect on the rigid body rotation.
- Insensitive to the mesh size.
- The value should be small enough to maintain accuracy.

11.7.2 Difficulties in the Thin Shell Model

The thin shell structure is flexible in the lateral direction at the beginning of the analysis. The differential stiffness develops as the deformation proceeds, and the shell structure becomes progressively stiffer due to the nonlinear geometric effect. Geometric stiffening of thin shells often poses difficulties in convergence during the initial loading stage. The difficulties arise from ill-conditioned stiffness due to a very large ratio between bending and membrane stiffnesses. Consider the stiffness of a $(L \times L \times t)$ quadrilateral element:

$$k_b \propto \frac{Et^3}{L^2} \quad \text{and} \quad k_m \propto Et$$

Thus the membrane to bending stiffness ratio (k_m/k_b) is dictated by $(L/t)^2$. As such, the lateral pressure on the thin shell causes excessively large rotations due to bending of the initially soft stiffness. The large rotation could easily involve a larger displacement than 1.57 radian (equivalent to 90°) for rotation, which causes singularity in the gimbal angle calculation.

The gimbal angle for large rotations is applicable to the QUAD4 and TRIA3 elements. At each iteration, the gimbal angle is accumulated as

$$\begin{aligned}\delta\theta_x &= (\Delta\theta_y \sin \theta_z + \Delta\theta_x \cos \theta_z) / \cos \theta_y \\ \delta\theta_y &= \Delta\theta_y \cos \theta_z - \Delta\theta_x \sin \theta_z \\ \delta\theta_z &= \Delta\theta_z + [(\Delta\theta_y \sin \theta_z + \Delta\theta_x \cos \theta_z) / \cos \theta_y] * \sin \theta_y\end{aligned}$$

where $\Delta\theta$ is the incremental rotation in the global coordinate system and $\delta\theta$ is the net incremental gimbal angle. A mathematical singularity occurs when the incremental gimbal angle component $\theta_y = \pm 90^\circ$. If $|\theta_y| \geq 90^\circ$, the user warning message "Gimbal Angle Exceeds 90 Degrees" is issued. This is usually caused by numerical ill-conditioning during the nonlinear iterations rather than by physical rotation exceeding 90° . The problem is most likely aggravated beyond the occurrence of $|\theta_y| \geq 90^\circ$ and the solution diverges. This limitation has been eliminated in Version 67 by introducing a set of auxiliary angles to avoid division by $\cos 90^\circ$.

The divergent process can be contained by using a proper value of K6ROT. This is because excessive θ_y rotation of the thin shell structure propagates to the sixth DOF, even though there is no normal rotation in the physical problem, and causes excessive normal rotation. K6ROT, which provides a stiffness for the normal rotation, can favorably affect the iterative process. Recommendations to alleviate the convergence difficulties when the gimbal angle exceeds 90 degrees are:

- If the physical problem specifies the rotation of θ_y beyond 90° , the coordinate system should be changed so that $\theta_y \leq 90^\circ$ (this limitation is removed in Version 67).
- If the problem diverges due to numerical condition, a smaller load increment with a proper value of K6ROT should be used.

11.7.3 Parametric Studies

Four problems are examined to study the effect of the parameter K6ROT on the iterative process of thin shell structures subjected to geometric stiffening. The problems are

- QUAD4 model of rectangular plate with line load
- TRIA3 model of rectangular plate with line load
- QUAD4 model of square plate with uniform pressure
- QUAD4 model of square plate with concentrated load

SOL 66 in Version 66 is used in the analysis of the problems. The main feature of these problems is geometric nonlinearity due to geometric stiffening. The effects of geometric nonlinearity are included in the analysis by specifying PARAM, LGDISP,1.

The convergence and efficiency characteristics of the solution process are observed in order to determine the optimal range of values for K6ROT. To obtain an even distribution of converging and diverging solutions for each problem, the value of K6ROT is varied from 1 to 10^9 , incremented by a factor of 10 in each run. The performance (i.e., the number of stiffness updates, iterations, and line searches) of the successful executions is presented for the corresponding K6ROT value.

11.7.4 QUAD4 Model of Rectangular Plate with Line Load

A rectangular plate has dimensions of 200 mm \times 400 mm and a thickness of 0.5 mm. The plate is simply supported at the 200 mm sides and is subjected to a line load of 400 N/mm acting in the -z direction as shown in Fig. 11.7.1.

The plate is modeled by 8 QUAD4 elements with linear elastic material. The simply supported boundary condition is imposed by constraining components 1, 2, and 3 of the GRID points at the 200 mm sides of the plate. The line load at GRID 2, 5, and 8 is specified using FORCE and LOAD Bulk Data entries. The input data are given in Table 11.7.1 for K6ROT = 10^4 .

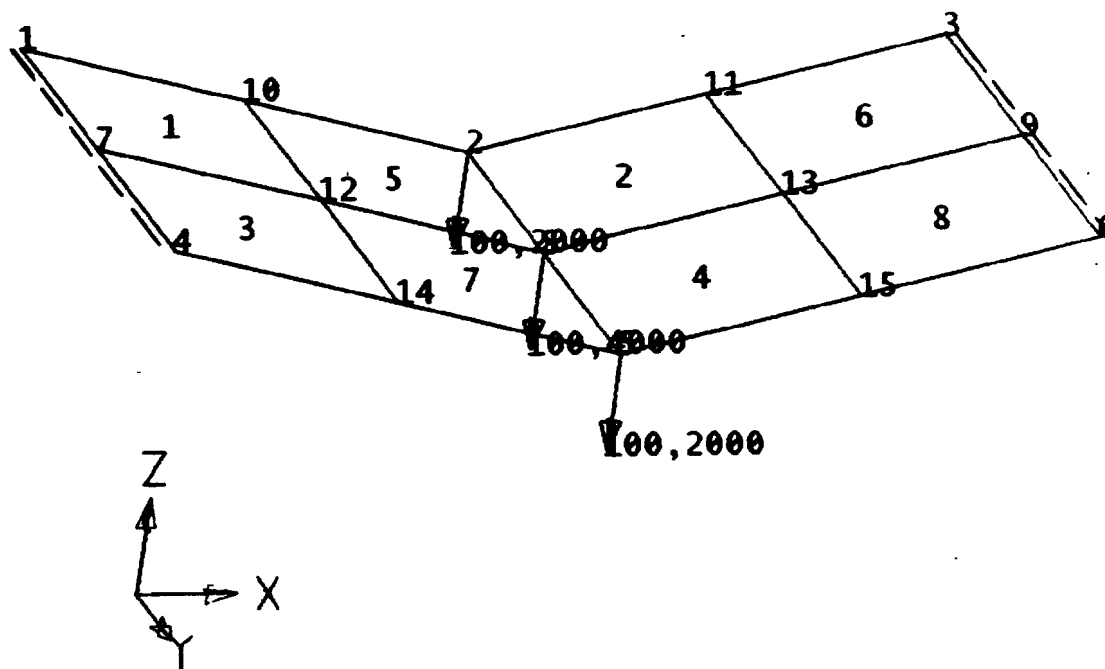


Figure 11.7.1 QUAD4 Model of Rectangular Plate with Line Load.

Table 11.7.1 Input Data Listing for QUAD4 Model with Line Load

```

ID TK6ROT1, V66 $ SHL 3/16/90, TLL 4/21/89
TIME 5 $ CPU MINUTES
SOL 66 $ NONLINEAR STATIC ANALYSIS
DIAG 50
CEND
TITLE = S.S. RECTANGULAR PLATE WITH 8 QUAD4 ELEMENTS AND LINE LOAD
SEALL = ALL
DISP = ALL
OLOAD = ALL
SUBCASE 1
NLPARM = 10
LOAD = 100
SUBCASE 2
NLPARM = 20
LOAD = 200
OUTPUT(PLOT)
CSCALE 1.3
PLOTTER NAST
SET 1 = ALL
MAXI DEFO = 1.
VIEW 30., 20., 0.
FIND SCALE ORIGIN 1 SET 1
PTITLE = ISOMETRIC VIEW
PLOT SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
PLOT STATIC 0 MAXI DEFO 1. SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$$ PARAMETERS
PARAM K6ROT 1.E+4
PARAM LGDISP 1
$
NLPARM 10 4 AUTO 1 -50 YES
NLPARM 20 5 AUTO 1 -50 YES
$$ GEOMETRY
GRID 1 0. 0. 0. 123
GRID 2 200. 0. 0.
GRID 3 400. 0. 0. 123
GRID 4 0. 200. 0. 123
GRID 5 200. 200. 0.
GRID 6 400. 200. 0. 123
GRID 7 0. 100. 0. 123
GRID 8 200. 100. 0.
GRID 9 400. 100. 0. 123
GRID 10 100. 0. 0.
GRID 11 300. 0. 0.
GRID 12 100. 100. 0.
GRID 13 300. 100. 0.
GRID 14 100. 200. 0.
GRID 15 300. 200. 0.

```

\$\$ CONNECTIVITY

CQUAD4	1	1	1	10	12	7
CQUAD4	2	1	2	11	13	8
CQUAD4	3	1	4	14	12	7
CQUAD4	4	1	5	15	13	8
CQUAD4	5	1	10	2	8	12
CQUAD4	6	1	11	3	9	13
CQUAD4	7	1	14	5	8	12
CQUAD4	8	1	15	6	9	13

\$\$ PROPERTIES

MAT1	10	2.07E+5	.3
PSHELL	1	10	.5

\$\$ LOADING

FORCE	100	2	2000.	0.	0.	-1.
FORCE	100	5	2000.	0.	0.	-1.
FORCE	100	8	4000.	0.	0.	-1.
LOAD	200	1.	10.	100		

\$

ENDDATA

Table 11.7.2 summarizes the results of the K6ROT parametric study. The solution converges only for $K6ROT = 10^4$. For $K6ROT \geq 10^5$, the solution diverges at the first load increment. The analysis stops for $K6ROT = 10^2$ because the maximum number of iterations is reached in the first increment of the second subcase. For the remaining K6ROT values of 1, 10, and 10^3 , the analysis stops due to a diverging solution. For this problem, the optimum value of K6ROT is 10^4 . The load-deflection curve of the plate at the center (GRID 8) is given in Fig. 11.7.2.

Table 11.7.2 Results of Parametric Study for K6ROT:
QUAD4 Model with Line Load
(K Updates/Iterations/Line Searches)

INC	LOAD	K6ROT					U_8
		1	10	10^2	10^3	10^4^{**}	
1-1	10	diverging	4/32/19	5/39/21	5/32/22	4/33/18	9.115
1-2	20		1/11/3	1/11/3	1/11/3	1/11/3	11.496
1-3	30		1/13/3	3/14/11	3/13/10	3/13/10	13.167
1-4	40		diverging	1/12/6*	1/12/3	1/11/7*	14.497
2-1	112			MAXITER	1/13/9	1/14/6	20.468
2-2	184				2/16/14	2/9/5	24.178
2-3	256				2/17/13	3/17/11	27.017
2-4	328				diverging	2/14/14	29.369
2-5	400					1/9/4	31.401

- Notes :
1. Discrepancies in U_8 with different values of K6ROT is less than 0.0003% at *.
 2. Solution diverges at the first increment beyond $K6ROT=10^5$.
 3. ** indicates the most effective case.

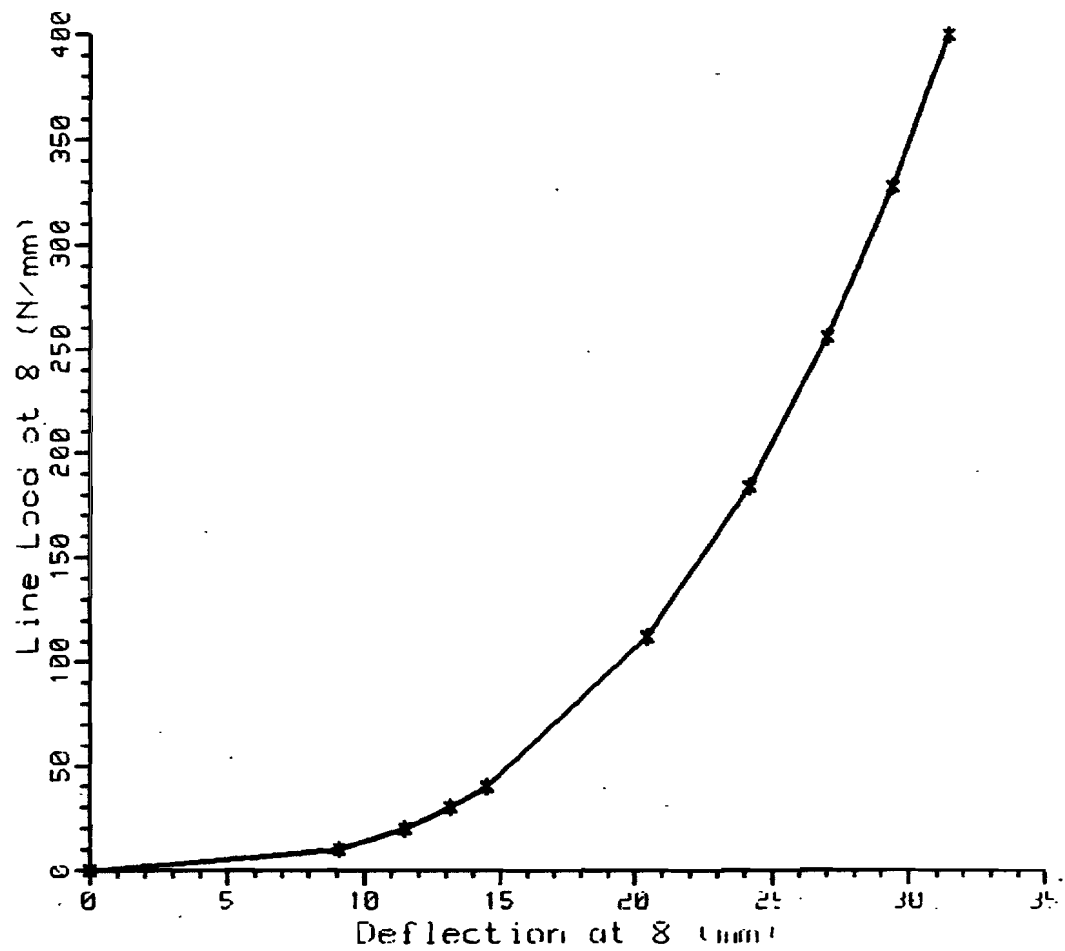


Figure 11.7.2 Load-Deflection at Center of QUAD4 Model with Line Load.

11.7.5 TRIA3 Model of Rectangular Plate with Line Load

This problem was chosen to be identical to the previous one, but with TRIA3 instead of QUAD4 elements in the model. The loading had to be reduced, however, because the solution did not converge with the original loading of 400 N/mm. Fig. 11.7.3 shows the rectangular plate with dimensions of 200 mm \times 400 mm and a thickness of 0.5 mm. The plate is simply supported at the 200 mm sides and is subjected to a line load of 140 N/mm acting in the -z direction.

The plate is modeled by 16 TRIA3 elements with linear elastic material. The simply supported boundary condition at the 200 mm sides of the plate is imposed by constraining components 1, 2, and 3 of the GRID points. The line load at GRID 2, 5, and 8 is specified using FORCE and LOAD Bulk Data entries. Table 11.7.3 shows the input data for K6ROT = 10^2 .

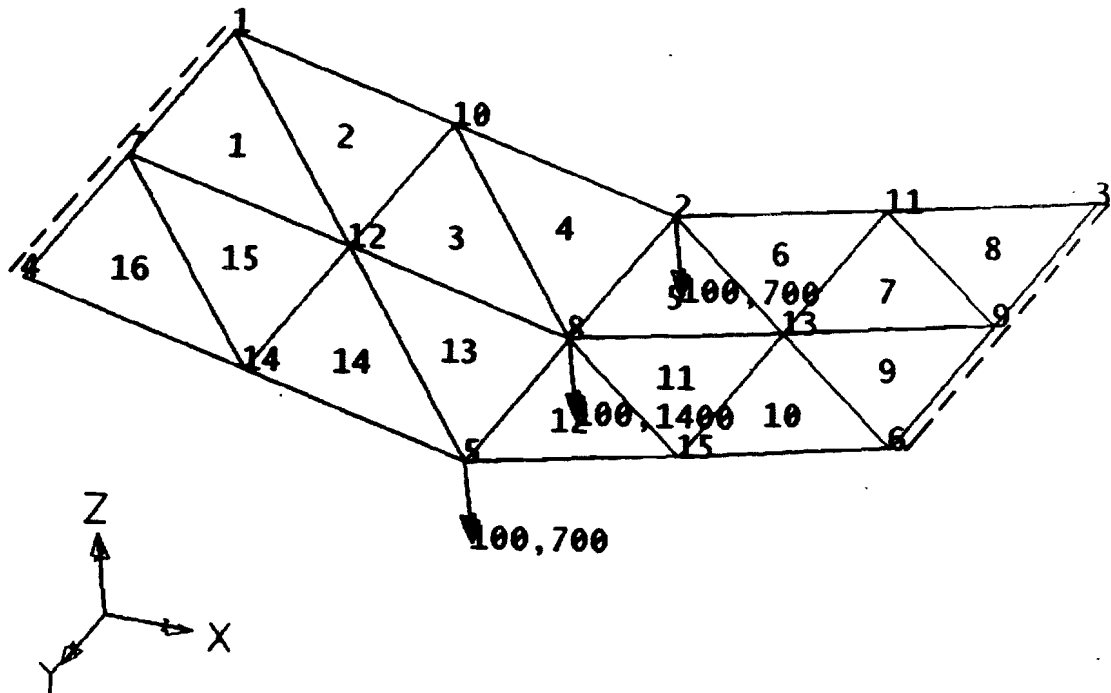


Figure 11.7.3 TRIA3 Model of Rectangular Plate with Line Load.

Table 11.7.3 Input Data Listing for TRIA3 Model with Line Load

```

ID TK6ROT2, V66 $ CSK 3/16/90
TIME 5 $ CPU MINUTES
SOL 66 $ NONLINEAR STATIC ANALYSIS
DIAG 50
CEND
TITLE = S.S. RECTANGULAR PLATE WITH 16 TRIA3 ELEMENTS AND LINE LOAD
  SEALL = ALL
  DISP = ALL
  OLOAD = ALL
SUBCASE 1
  NLPARM = 10
  LOAD = 100
SUBCASE 2
  NLPARM = 20
  LOAD = 200
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 = ALL
  MAXI DEFO = 1.
  VIEW 30., 20., 0.
  FIND SCALE ORIGIN 1 SET 1
  PTITLE = ISOMETRIC VIEW
  PLOT SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
  PLOT STATIC 0 MAXI DEFO 1. SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$$ PARAMETERS
PARAM K6ROT 1.E+2
PARAM LGDISP 1
$
NLPARM 10 4 AUTO 1 -50 YES
NLPARM 20 5 AUTO 1 -50 YES
$$ GEOMETRY
GRID 1 0. 0. 0. 123
GRID 2 200. 0. 0.
GRID 3 400. 0. 0. 123
GRID 4 0. 200. 0. 123
GRID 5 200. 200. 0.
GRID 6 400. 200. 0. 123
GRID 7 0. 100. 0. 123
GRID 8 200. 100. 0.
GRID 9 400. 100. 0. 123
GRID 10 100. 0. 0.
GRID 11 300. 0. 0.
GRID 12 100. 100. 0.
GRID 13 300. 100. 0.
GRID 14 100. 200. 0.
GRID 15 300. 200. 0.

```


\$\$ CONNECTIVITY

CTRIA3	1	1	1	12	7
CTRIA3	2	1	1	10	12
CTRIA3	3	1	10	8	12
CTRIA3	4	1	10	2	8
CTRIA3	5	1	2	13	8
CTRIA3	6	1	2	11	13
CTRIA3	7	1	11	9	13
CTRIA3	8	1	11	3	9
CTRIA3	9	1	13	9	6
CTRIA3	10	1	13	6	15
CTRIA3	11	1	8	13	15
CTRIA3	12	1	8	15	5
CTRIA3	13	1	12	8	5
CTRIA3	14	1	12	5	14
CTRIA3	15	1	7	12	14
CTRIA3	16	1	7	14	4

\$\$ PROPERTIES

MAT1	10	2.07E+5	.3
PSHELL	1	10	.5
			10

\$\$ LOADING

FORCE	100	2	700.	0.	0.	-1.
FORCE	100	5	700.	0.	0.	-1.
FORCE	100	8	1400.	0.	0.	-1.
LOAD	200	1.	10.	100		

\$

ENDDATA

The results of the K6ROT parametric study are summarized in Table 11.7.4. The solution converges for K6ROT values of 10 , 10^2 , 10^3 , and 10^5 . The analysis stops due to a diverging solution at the first load increment for $K6ROT \geq 10^6$ and the remaining K6ROT values of 1 and 10^4 . In this problem, the optimum value of K6ROT is 10^2 because it is the most efficient in terms of the number of stiffness updates, iterations, and line searches. The load-deflection curve of the plate at the center (GRID 8) is given in Fig. 11.7.4.

Table 11.7.4 Results of Parametric Study for K6ROT:
 TRIA3 Model with Line Load
 (K Updates/Iterations/Line Searches)

INC	LOAD	K6ROT						U_8
		1	10	10^{2**}	10^3	10^4	10^5	
1-1	3.5	diverging	8/49/50	6/42/28	5/40/26	diverging	9/48/38	6.364
1-2	7		2/20/9	1/18/9	1/18/9		2/21/12	8.030
1-3	10.5		1/18/9	1/18/10	1/19/7		1/21/5	9.197
1-4	14		1/20/8	1/20/8	1/20/9		1/21/8	10.127
2-1	39.2		1/26/14	1/26/15	2/32/15		1/31/11	14.292
2-2	64.4		1/24/16	1/25/16	1/27/13		1/36/8	16.877
2-3	89.6		2/29/18	1/26/18	3/30/20		1/39/8	18.851
2-4	114.8		1/27/18	1/28/16	2/33/20		1/41/14	20.484
2-5	140.0		2/31/20*	2/26/13	2/34/21		2/42/17*	21.895

- Notes :
1. Discrepancies in U_8 with different values of K6ROT is less than 0.011% at *.
 2. Solution diverges at the first increment beyond $K6ROT=10^6$.
 3. ** indicates the most effective case.

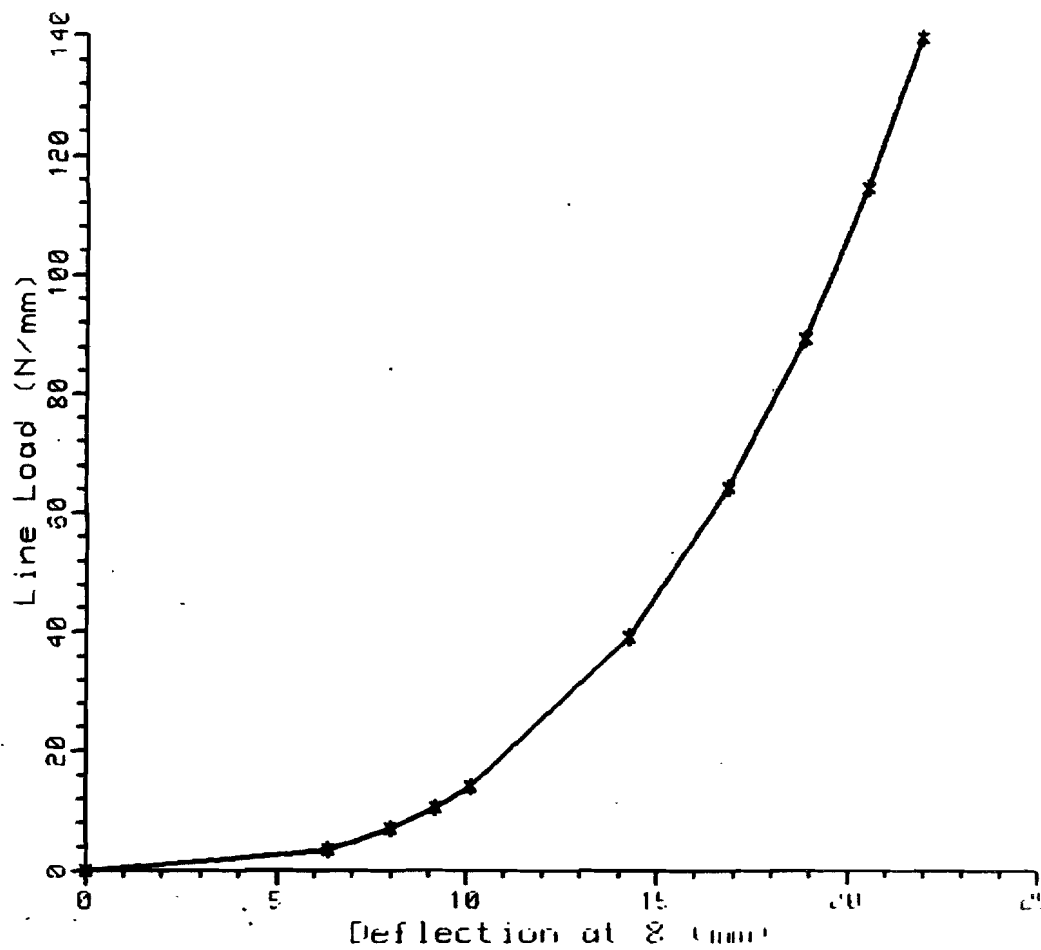


Figure 11.7.4 Load-Deflection at Center of TRIA3 Model with Line Load.

11.7.6 QUAD4 Model of Square Plate with Uniform Pressure

Fig. 11.7.5 shows a square plate which has a length of 400 mm and a thickness of 0.5 mm. The plate is simply supported at three of its sides and is subjected to a uniform pressure loading of 1 N/mm^2 applied normal to its plane.

The plate is modeled by 16 QUAD4 elements with linear elastic material. The simply supported boundary condition is imposed by constraining components 1, 2, and 3 of the GRID points at three sides of the plate. The uniform pressure loading is specified using PLOAD2 and LOAD Bulk Data entries. The input data are given in Table 11.7.5 for $K6ROT = 10^3$.

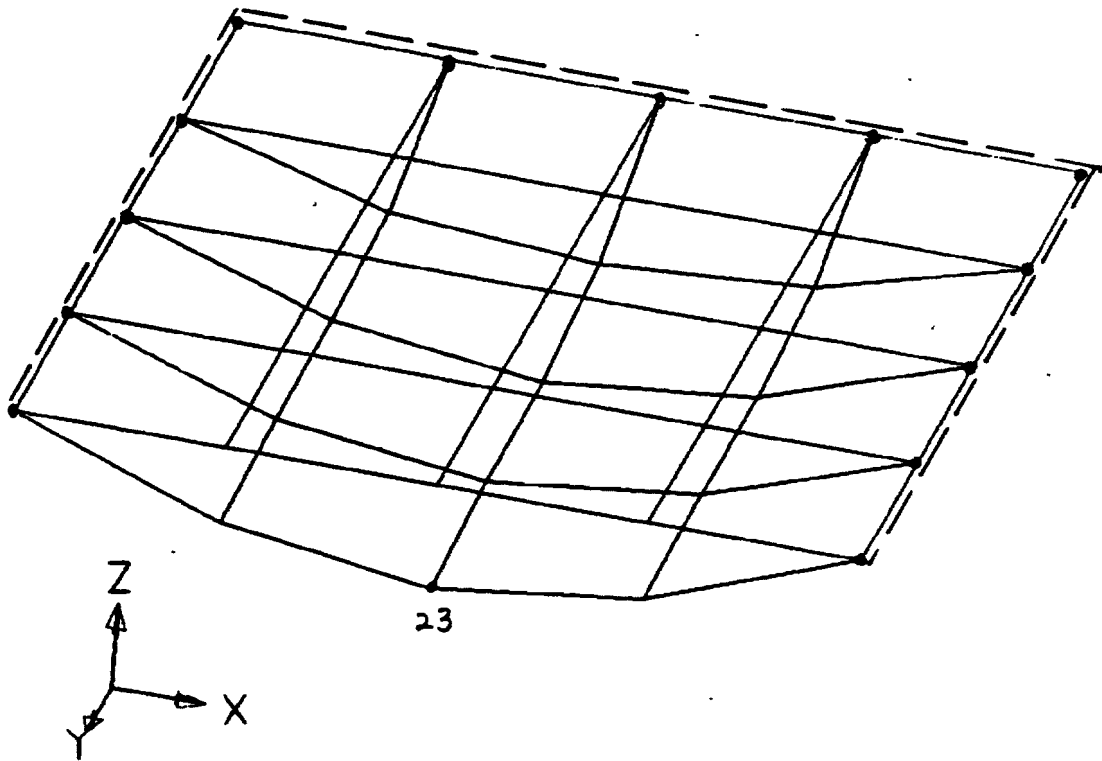


Figure 11.7.5 QUAD4 Model of Square Plate with Uniform Pressure.

Table 11.7.5 Input Data Listing for QUAD4 Model with Uniform Pressure

```

ID TK6ROT3, V66 $ RA 3/7/90
TIME      5 $ CPU MINUTES
SOL       66      $ NONLINEAR STATIC ANALYSIS
DIAG      50
CEND
TITLE = S.S. SQUARE PLATE WITH 16 QUAD4 ELEMENTS AND UNIFORM PRESSURE
  SEALL = ALL
  DISP  = ALL
  OLOAD = ALL
SUBCASE 1
  NLPARM = 10
  LOAD   = 100
SUBCASE 2
  NLPARM = 20
  LOAD   = 200
OUTPUT(PLOT)
  CSCALE 1.3
  PLOTTER NAST
  SET 1 = ALL
  MAXI DEFO = 1.
  VIEW 30., 20., 0.
  FIND SCALE ORIGIN 1 SET 1
  PTITLE = ISOMETRIC VIEW
  PLOT SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
  PLOT STATIC 0 MAXI DEFO 1. SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$$ PARAMETERS
PARAM K6ROT 1.E+3
PARAM LGDISP 1
$
NLPARM 10      4      AUTO 1      -50      YES
NLPARM 20      5      AUTO 1      -50      YES
$$ GEOMETRY
GRID 1      0.      0.      0.      123
GRID 2      100.    0.      0.      123
GRID 3      200.    0.      0.      123
GRID 4      300.    0.      0.      123
GRID 5      400.    0.      0.      123
GRID 6      0.      100.    0.      123
GRID 7      100.    100.    0.
GRID 8      200.    100.    0.
GRID 9      300.    100.    0.
GRID 10     400.    100.    0.      123
GRID 11     0.      200.    0.      123
GRID 12     100.    200.    0.
GRID 13     200.    200.    0.
GRID 14     300.    200.    0.
GRID 15     400.    200.    0.      123

```

GRID	16	0.	300.	0.	123
GRID	17	100.	300.	0.	
GRID	18	200.	300.	0.	
GRID	19	300.	300.	0.	
GRID	20	400.	300.	0.	123
GRID	21	0.	400.	0.	123
GRID	22	100.	400.	0.	
GRID	23	200.	400.	0.	
GRID	24	300.	400.	0.	
GRID	25	400.	400.	0.	123

\$\$ CONNECTIVITY

CQUAD4	1	1	1	2	7	6
CQUAD4	2	1	2	3	8	7
CQUAD4	3	1	3	4	9	8
CQUAD4	4	1	4	5	10	9
CQUAD4	5	1	6	7	12	11
CQUAD4	6	1	7	8	13	12
CQUAD4	7	1	8	9	14	13
CQUAD4	8	1	9	10	15	14
CQUAD4	9	1	11	12	17	16
CQUAD4	10	1	12	13	18	17
CQUAD4	11	1	13	14	19	18
CQUAD4	12	1	14	15	20	19
CQUAD4	13	1	16	17	22	21
CQUAD4	14	1	17	18	23	22
CQUAD4	15	1	18	19	24	23
CQUAD4	16	1	19	20	25	24

\$\$ PROPERTIES

MAT1	10	2.07E+5	.3
PSHELL	1	10	.5

\$\$ LOADING

PLOAD2	100	-.1	1	THRU	16
LOAD	200	1.	10.	100	

\$

ENDDATA

Table 11.7.6 summarizes the results of the K6ROT parametric study. The solution diverges at the third load increment of the second subcase for $K6ROT = 10^5$ and at the first load increment of the first subcase for $K6ROT \geq 10^6$. The analysis stops at the first load increment for $K6ROT = 1$ when the maximum number of iterations is reached. For the remaining K6ROT values of 10 through 10^4 , the solution converges. The optimum value of K6ROT is 10^3 in this problem. The load-deflection curve of the plate at GRID 23 is given in Fig. 11.7.6.

Table 11.7.6 Results of Parametric Study for K6ROT:
QUAD4 Model with Uniform Pressure
(K Updates/Iterations/Line Searches)

INC	LOAD	K6ROT						U_{23}
		1	10	10^2	10^{3**}	10^4	10^5	
1-1	0.025	MAXITER	8/39/25	3/35/15	4/31/15	5/34/17	6/40/30	7.320
1-2	0.05		1/12/0	1/12/0	1/12/0	1/12/1	2/25/7	9.231
1-3	0.075		1/13/0	1/13/0	1/12/4	1/12/4	1/24/3	10.572
1-4	0.1		1/10/1	1/10/1	1/13/1	1/12/3	1/24/7	11.640
2-1	0.28		1/17/1	1/17/1	1/17/1	1/18/2	1/6/3	16.428
2-2	0.46		1/12/4	1/12/4	1/12/4	1/17/2	1/22/10	19.396
2-3	0.64		2/16/2	3/16/6	2/20/9	2/19/10	diverging	21.664
2-4	0.82		1/11/0	1/13/6	1/10/3	2/21/6		23.544
2-5	1.0		2/15/9*	1/12/0	1/15/3	1/19/5*		25.162

- Notes :
1. Discrepancies in U_{23} with different values of K6ROT is less than 0.0087% at *.
 2. Solution diverges at the first increment beyond $K6ROT=10^6$.
 3. ** indicates the most effective case.

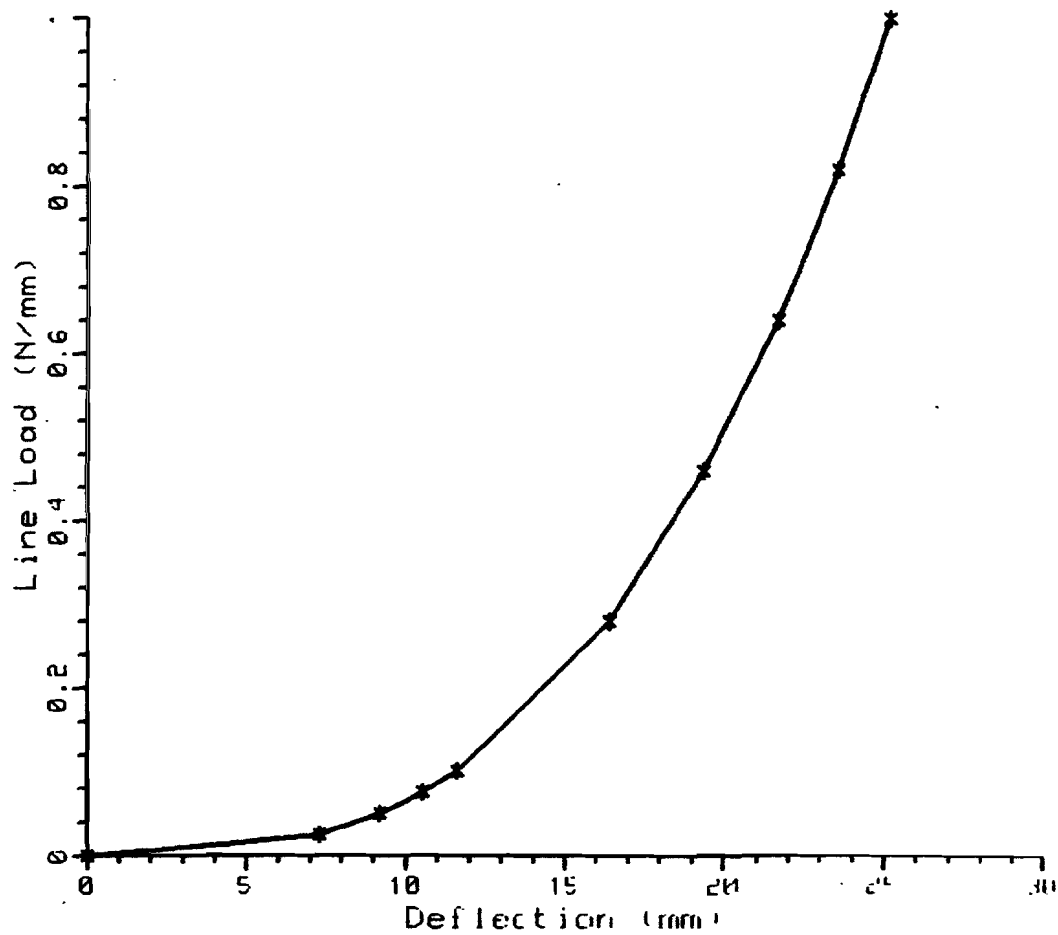


Figure 11.7.6 Load-Deflection at Grid 23 of QUAD4 Model with Uniform Pressure.

11.7.7 QUAD4 Model of Square Plate with Concentrated Load

This problem is similar to the previous one, but has a different loading condition. A square plate has a length of 400 mm and a thickness of 0.4 mm. The plate is simply supported on three sides and is subjected to a concentrated load of 13,200 N acting in the $-z$ direction as shown in Fig. 11.7.7.

The plate is modeled by 16 QUAD4 elements with linear elastic material. The simply supported boundary condition is imposed by constraining components 1, 2, and 3 of the GRID points at three sides of the plate. The concentrated load at GRID 19 is specified using FORCE and LOAD Bulk Data entries. Table 11.7.7 shows the input data for $K6ROT = 10^3$.

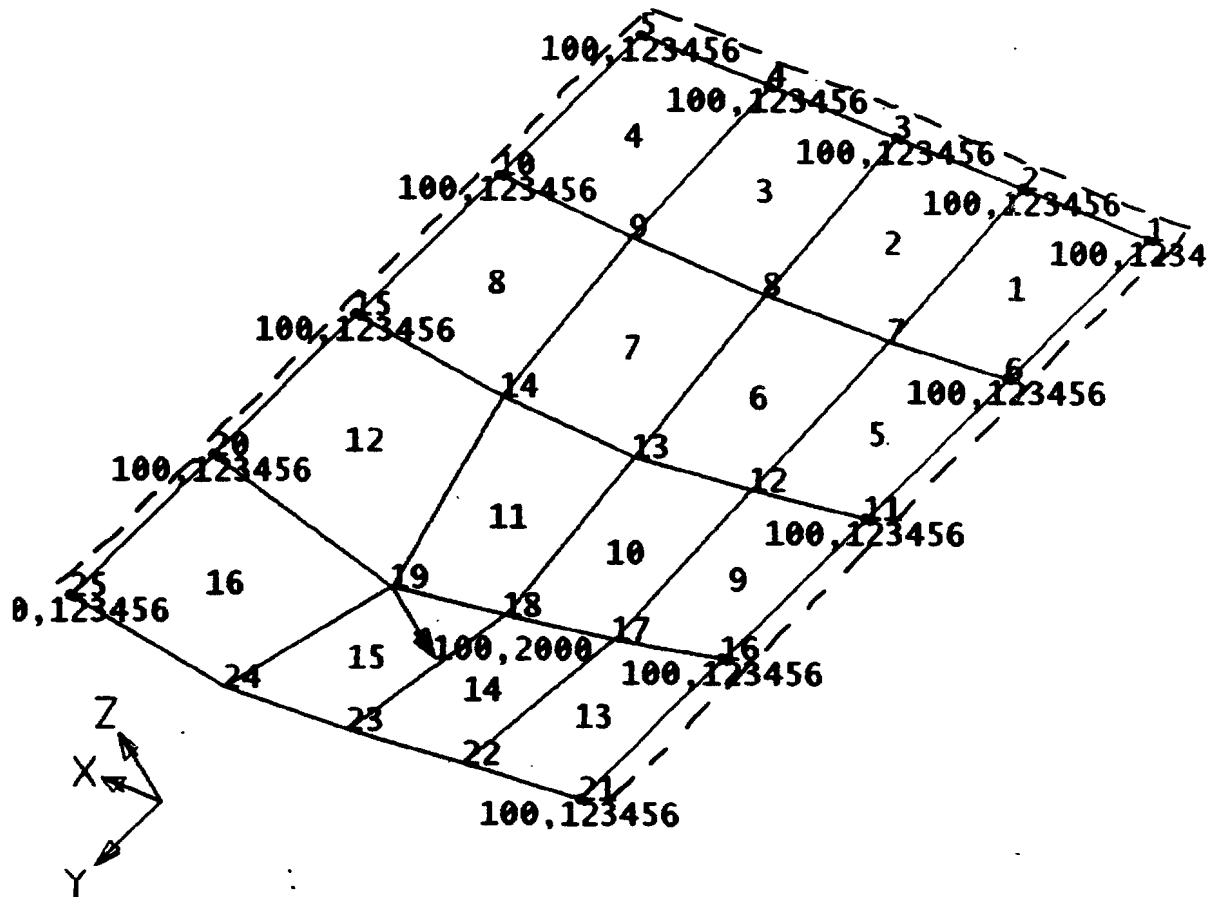


Figure 11.7.7 QUAD4 Model of Square Plate with Concentrated Load

Table 11.7.7 Input Data Listing for QUAD4 Model with Concentrated Load

```

ID TK6ROT4, V66 $ RA 3/7/90
TIME 5 $ CPU MINUTES
SOL 66 $ NONLINEAR STATIC ANALYSIS
DIAG 50
CEND
TITLE = S.S. SQUARE PLATE WITH 16 QUAD4 ELEMENTS AND CONCENTRATED LOAD
SEALL = ALL
DISP = ALL
OLOAD = ALL
SUBCASE 1
NLPARM = 10
LOAD = 100
SUBCASE 2
NLPARM = 20
LOAD = 200
OUTPUT(PLOT)
CSCALE 1.3
PLOTTER NAST
SET 1 = ALL
MAXI DEFO = 1.
VIEW 30., 20., 0.
FIND SCALE ORIGIN 1 SET 1
PTITLE = ISOMETRIC VIEW
PLOT SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
PLOT STATIC 0 MAXI DEFO 1. SET 1 ORIGIN 1 SYMBOL 1 LABEL GRID
BEGIN BULK
$$ PARAMETERS
PARAM K6ROT 1.E+3
PARAM LGDISP 1
$
NLPARM 10 4 AUTO 1 -40 YES
NLPARM 20 5 AUTO 1 -40 YES
$$ GEOMETRY
GRID 1 0. 0. 0. 123
GRID 2 100. 0. 0. 123
GRID 3 200. 0. 0. 123
GRID 4 300. 0. 0. 123
GRID 5 400. 0. 0. 123
GRID 6 0. 100. 0. 123
GRID 7 100. 100. 0.
GRID 8 200. 100. 0.
GRID 9 300. 100. 0.
GRID 10 400. 100. 0. 123
GRID 11 0. 200. 0. 123
GRID 12 100. 200. 0.
GRID 13 200. 200. 0.
GRID 14 300. 200. 0.
GRID 15 400. 200. 0. 123

```

GRID	16		0.	300.	0.		123
GRID	17		100.	300.	0.		
GRID	18		200.	300.	0.		
GRID	19		300.	300.	0.		
GRID	20		400.	300.	0.		123
GRID	21		0.	400.	0.		123
GRID	22		100.	400.	0.		
GRID	23		200.	400.	0.		
GRID	24		300.	400.	0.		
GRID	25		400.	400.	0.		123
\$\$ CONNECTIVITY							
CQUAD4	1	1	1	2	7	6	
CQUAD4	2	1	2	3	8	7	
CQUAD4	3	1	3	4	9	8	
CQUAD4	4	1	4	5	10	9	
CQUAD4	5	1	6	7	12	11	
CQUAD4	6	1	7	8	13	12	
CQUAD4	7	1	8	9	14	13	
CQUAD4	8	1	9	10	15	14	
CQUAD4	9	1	11	12	17	16	
CQUAD4	10	1	12	13	18	17	
CQUAD4	11	1	13	14	19	18	
CQUAD4	12	1	14	15	20	19	
CQUAD4	13	1	16	17	22	21	
CQUAD4	14	1	17	18	23	22	
CQUAD4	15	1	18	19	24	23	
CQUAD4	16	1	19	20	25	24	
\$\$ PROPERTIES							
MAT1	10	2.07E+5		.3			
PSHELL	1	10	.4	10			
\$\$ LOADING							
FORCE	100	19		2200.	0.	0.	-1.
LOAD	200	1.	6.	100			
\$							
ENDDATA							

The results of the K6ROT parametric study are summarized in Table 11.7.8. The results are similar to those of the previous problem of a square plate with a uniform pressure. The solution converges for K6ROT values of 10, 10^2 , 10^3 , and 10^4 . The analysis stops at the first load increment for K6ROT = 1 and K6ROT = 10^5 because the solution does not converge in MAXITER iterations. For K6ROT $\geq 10^6$, the analysis stops at the first load increment due to a diverging solution. The optimum value of K6ROT is also 10^3 in this problem. The load-deflection curve of the plate at GRID 19 is shown in Fig. 11.7.8.

Table 11.7.8 Results of Parametric Study for K6ROT:
QUAD4 Model with Concentrated Load
(K Updates/Iterations/Line Searches)

INC	LOAD	K6ROT						U_{19}
		1	10	10^2	10^{3**}	10^4	10^5	
1-1	550	MAXITER	3/38/18	4/38/18	3/39/16	4/38/20	MAXITER	6.4118
1-2	1100		1/15/1	1/15/1	1/15/1	1/16/1		8.095
1-3	1650		2/19/1	2/19/1	1/16/1	1/18/0		9.274
1-4	2200		1/17/0	1/17/0	1/17/0	1/19/1		10.213
2-1	4400		1/20/3	1/20/3	1/20/2	1/22/3		12.878
2-2	6600		2/14/12	1/11/10	1/12/9	1/25/2		14.746
2-3	8800		2/13/11	2/14/12	2/14/12	2/13/10		16.234
2-4	11000		2/12/11	2/12/11	2/12/11	2/14/13		17.489
2-5	13200		2/12/11*	2/12/11	2/12/11	1/16/9*		18.587

- Notes :
1. Discrepancies in U_{19} with different values of K6ROT is less than 0.12% at *.
 2. Solution diverges at the first increment beyond K6ROT= 10^6 .
 3. ** indicates the most effective case.

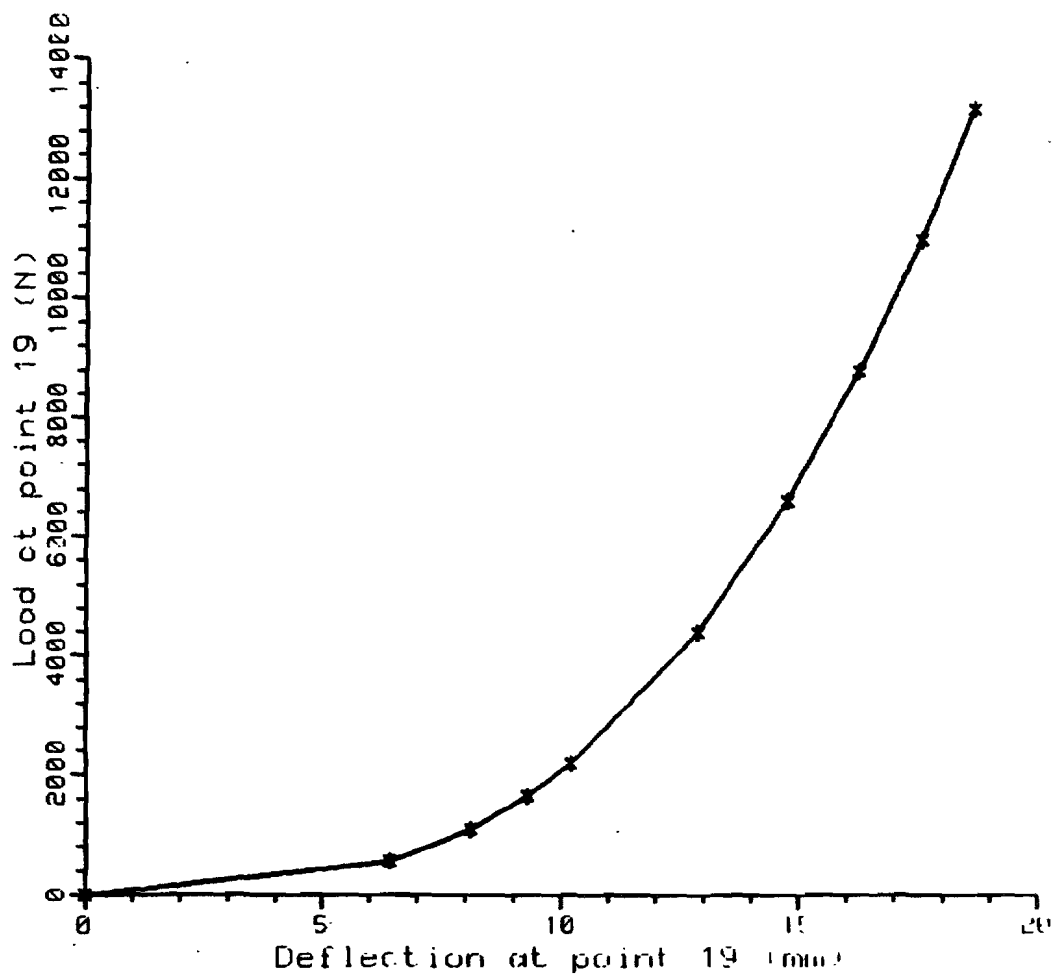


Figure 11.7.8 Load-Deflection at Grid 19 of QUAD4 Model with Concentrated Load.

11.7.8 Observations

At the beginning of the analysis, the plate is flexible. As the deformation increases, however, the plate becomes progressively stiffer due to the nonlinear geometric effect. Bending of the initially soft plate causes excessive rotation which propagates to the sixth DOF. This excessive rotation in the sixth DOF is alleviated by a larger value of K6ROT. Therefore, small values of K6ROT ($< 10^2$) make the solution convergence very difficult. In fact, the solution did not converge for K6ROT = 1 in any of these test problems. Values of K6ROT that are too large ($\geq 10^6$) also cause the solution to diverge in all of the problems. Since an excessively large value of K6ROT will lock the varying strain within an element by enforcing the rotation measured in the element (Ω_z) to conform to the GRID point rotation (θ_z), the solution will be inaccurate even if it converges with a large K6ROT value. The parametric study thus indicates that the iterative process in geometric stiffening is highly sensitive to K6ROT, whose proper value can contain the divergent process. The optimal range of values is found to be $10^2 \leq \text{K6ROT} \leq 10^4$.

11.8 ADAPTATION OF CREEP ANALYSIS CAPABILITY TO GENERAL VISCOELASTIC MATERIALS

11.8.1 Introduction

The software's creep capability was developed based on the Maxwell-Kelvin theological model, which can be applied to any viscoelastic materials [11.17,11.18]. To use this capability, users need to define the creep characteristics either by empirical creep laws in the form of

$$\varepsilon^c(\sigma, t) = A(\sigma)[1 - e^{-R(\sigma)t}] + K(\sigma)t$$

with

$$\begin{aligned} A(\sigma) &= a\sigma^b & \text{or} & \quad a\text{EXP}(b\sigma) \\ R(\sigma) &= c\text{EXP}(d\sigma) & \text{or} & \quad c\sigma^d \\ K(\sigma) &= e[\sinh(f\sigma)]^g & \text{or} & \quad e\text{EXP}(f\sigma) \end{aligned}$$

or by defining the rheological creep parameters (C_s , C_p and K_p) as functions of stresses, so that the creep strain is represented by

$$\varepsilon^c(\sigma, t) = \frac{\sigma}{K_p}[1 - e^{-(K_p/C_p)t}] + \frac{\sigma}{C_s}$$

If the user should model a material that is characterized by experimental data or by a creep law different from what is provided in the software, some kind of curve fitting is required to find equivalent coefficients or parameter values corresponding to the predefined empirical law (types 111 through 222) or the Maxwell-Kelvin theological model.

The least square fit is described to determine rheological parameters (C_s , C_p and K_p as functions of stresses) corresponding to the Findley's law representing the creep behavior in plastics as an example. An interactive program is developed to perform this numerical analysis procedure. This procedure should be followed in any similar application of the general viscoelastic or creep analysis.

11.8.2 Formulation for Least Square Fit of Rheological Parameters

Let us determine the parameter values (C_s , C_p and K_p) at a constant stress (σ) for the Maxwell-Kelvin rheological model represented by

$$\varepsilon^r = \frac{\sigma}{C_s}t + \frac{\sigma}{K_p}[1 - e^{-(K_p/C_p)t}] \quad (11.8.1)$$

which should simulate the behavior of plastics expressed by the Findley's law,

$$\varepsilon^f = \varepsilon_0 \sinh \frac{\sigma}{\sigma_0} + \varepsilon_t t^n \sinh \frac{\sigma}{\sigma_t} - \frac{\sigma}{E} \quad (11.8.2)$$

where

$$\begin{aligned}
 n &= 0.19 \\
 \varepsilon_0 &= 0.0067 \\
 \varepsilon_t &= 0.0011 \\
 \sigma_0 &= \sigma_t = 8500 \text{ psi} \\
 E &= 1.2 \times 10^6 \text{ psi}
 \end{aligned}$$

Using the data points (t_j, ε_j^f) on the reference curve in Eq. (11.8.2), the least square fit can be employed to minimize the error, i.e.,

$$\text{min. error} = \sum_{j=1}^j \delta_j^2 \quad (11.8.3)$$

where

$$\delta_j = \varepsilon_j^r - \varepsilon_j^f \quad (11.8.4)$$

Eq. (11.8.4) should be linearized with respect to C_s, C_p and K_p to apply an iterative and incremental process, i.e.,

$$\begin{aligned}
 C_s &= C'_s + \Delta C_s \\
 C_p &= C'_p + \Delta C_p \\
 K_p &= K'_p + \Delta K_p
 \end{aligned} \quad (11.8.5)$$

The first order approximation of Eq. (11.8.4) is obtained by a Taylor series with a substitution of Eq. (11.8.5), i.e.,

$$\delta \simeq \frac{\sigma}{C'_s} t + \frac{\sigma}{K'_p} [1 - e^{-(K'_p/C'_p)t}] - \varepsilon^f + \frac{\partial \varepsilon^r}{\partial C_s} \Delta C_s + \frac{\partial \varepsilon^r}{\partial C_p} \Delta C_p + \frac{\partial \varepsilon^r}{\partial K_p} \Delta K_p \quad (11.8.6)$$

where

$$\begin{aligned}
 \frac{\partial \varepsilon^r}{\partial C_s} &= -\frac{\sigma t}{C_s^2} \\
 \frac{\partial \varepsilon^r}{\partial C_p} &= -\frac{\sigma t}{C_p^2} e^{-(K_p/C_p)t} \\
 \frac{\partial \varepsilon^r}{\partial K_p} &= -\frac{\sigma t}{K_p C_p} e^{-(K_p/C_p)t} - \frac{\sigma}{K_p^2} [1 - e^{-(K_p/C_p)t}] \\
 &= \frac{\sigma}{K_p} e^{-(K_p/C_p)t} \left(\frac{1}{K_p} + \frac{t}{C_p} \right) - \frac{\sigma}{K_p^2}
 \end{aligned} \quad (11.8.7)$$

In order to satisfy the minimum error in Eq. (11.8.3), the following condition has to be satisfied, i.e.,

$$\sum_{j=1}^j \delta_j \frac{\partial \delta_j}{\partial \Delta C_s} = \sum_{j=1}^j \delta_j \frac{\partial \varepsilon^r}{\partial C_s} = 0$$

$$\sum_{j=1}^j \delta_j \frac{\partial \delta_j}{\partial \Delta C_p} = \sum_{j=1}^j \delta_j \frac{\partial \varepsilon^r}{\partial C_p} = 0 \quad (11.8.8)$$

$$\sum_{j=1}^j \delta_j \frac{\partial \delta_j}{\partial \Delta K_p} = \sum_{j=1}^j \delta_j \frac{\partial \varepsilon^r}{\partial K_p} = 0$$

When Eq. (11.8.6) is substituted into Eq. (11.8.8), there are three linear equations with ΔC_s , ΔC_p and ΔK_p as unknown variables, i.e.,

$$\begin{aligned} \sum_j [(\frac{\partial \varepsilon_j^r}{\partial C_s})^2 \Delta C_s + \frac{\partial \varepsilon_j^r}{\partial C_s} \frac{\partial \varepsilon_j^r}{\partial C_p} \Delta C_p + \frac{\partial \varepsilon_j^r}{\partial C_s} \frac{\partial \varepsilon_j^r}{\partial K_p} \Delta K_p] &= \sum_j [\frac{\partial \varepsilon_j^r}{\partial C_s} (\varepsilon_j^f - \varepsilon_j^r)] \\ \sum_j [\frac{\partial \varepsilon_j^r}{\partial C_p} \frac{\partial \varepsilon_j^r}{\partial C_s} \Delta C_s + (\frac{\partial \varepsilon_j^r}{\partial C_p})^2 \Delta C_p + \frac{\partial \varepsilon_j^r}{\partial C_p} \frac{\partial \varepsilon_j^r}{\partial K_p} \Delta K_p] &= \sum_j [\frac{\partial \varepsilon_j^r}{\partial C_p} (\varepsilon_j^f - \varepsilon_j^r)] \\ \sum_j [\frac{\partial \varepsilon_j^r}{\partial K_p} \frac{\partial \varepsilon_j^r}{\partial C_s} \Delta C_s + \frac{\partial \varepsilon_j^r}{\partial K_p} \frac{\partial \varepsilon_j^r}{\partial C_p} \Delta C_p + (\frac{\partial \varepsilon_j^r}{\partial K_p})^2 \Delta K_p] &= \sum_j [\frac{\partial \varepsilon_j^r}{\partial K_p} (\varepsilon_j^f - \varepsilon_j^r)] \end{aligned} \quad (11.8.9)$$

where the derivatives are defined in Eq. (11.8.7), and ε^r and ε^f in Eqs. (11.8.1) and (11.8.2), respectively. By solving the simultaneous equations in Eq. (11.8.9) for ΔC_s , ΔC_p and ΔK_p , these unknown parameters can be updated by Eq. (11.8.5) for the next iteration.

The rheological parameters may be evaluated by the least square fit of the strain-rate instead of the strain itself. For this approach, Eqs. (11.8.1), (11.8.2), (11.8.4), (11.8.6) and (11.8.7) should be replaced by the following equations:

$$\dot{\varepsilon}^r = \frac{\sigma}{C_s} + \frac{\sigma}{C_p} e^{-(K_p/C_p)t} \quad (11.8.1b)$$

$$\dot{\varepsilon}^f = n \varepsilon_t^{n-1} \sinh \frac{\sigma}{\sigma_t} \quad (11.8.2b)$$

$$\delta = \varepsilon^r - \varepsilon^f \quad (11.8.4b)$$

$$\delta \simeq \varepsilon^r - \varepsilon^f + \frac{\partial \dot{\varepsilon}^r}{\partial C_s} \Delta C_s + \frac{\partial \dot{\varepsilon}^r}{\partial C_p} \Delta C_p + \frac{\partial \dot{\varepsilon}^r}{\partial K_p} \Delta K_p \quad (11.8.6b)$$

$$\frac{\partial \dot{\varepsilon}^r}{\partial C_s} = -\frac{\sigma}{C_s^2} \quad (11.8.7b)$$

$$\frac{\partial \dot{\varepsilon}^r}{\partial C_p} = \frac{\sigma}{C_p^2} [\frac{K_p}{C_p} t - 1] e^{-(K_p/C_p)t}$$

$$\frac{\partial \dot{\varepsilon}^r}{\partial K_p} = -\frac{\sigma t}{C_p^2} e^{-(K_p/C_p)t}$$

11.8.3 Computational Process

The strain function in Eq. (11.8.1), which requires curve fitting, has two variables, namely, stress and time. The experimental data are usually measured at constant stress levels, which

become the basis for the empirical formula or the curve fitting. Therefore, the least square fit can be performed as formulated in the preceding section with data points in the operating time domain while the stress level is fixed. Newton's iteration starts to minimize the error in Eq. (11.8.3). The simultaneous equations in Eq. (11.8.9) are solved at every iteration and the iteration continues until the convergence is achieved. Upon convergence, a set of rheological parameters (C_s , C_p and K_p) has been determined for a constant stress. The least square fit is performed incrementally by sweeping along the entire operating stress range.

The iteration starts with initial values for C_s , C_p and K_p . Initial values should not be too far from the solution for a good convergence. The initial values can be appropriately estimated by considering a physical interpretation of the creep strain curve in relation to the rheological model in Eq. (11.8.1). The slope at a long time span (t_f) approximates the secondary creep rate, i.e.,

$$\dot{\epsilon}(t_f) \simeq \frac{\sigma}{C_s}$$

from which

$$C_s \simeq \frac{\sigma}{\dot{\epsilon}^f(t_f)} \quad (11.8.10)$$

The total primary creep strain at a long time span can be approximated by

$$\epsilon_{primary}(t_f) \simeq \frac{\sigma}{K_p} \quad (11.8.11)$$

from which

$$K_p \simeq \frac{\sigma}{\epsilon^f(t_f) - \frac{\sigma}{C_s} t_f} \quad (11.8.12)$$

The slope at $t=0$ can be related to C_p by

$$\dot{\epsilon}(t_0) \simeq \frac{\sigma}{C_s} + \frac{\sigma}{C_p}$$

from which

$$\frac{1}{C_p} \simeq \frac{\dot{\epsilon}^f(0)}{\sigma} - \frac{1}{C_s} \quad (11.8.13)$$

For $\sigma = 1000$ psi and $t_f = 500000$ hours

$$C_s \simeq \frac{1000}{5.964 \times 10^{-10}} = 1.6766 \times 10^{12} \text{ lb/in}^2/\text{hr}$$

$$K_p \simeq \frac{1000}{1.526 \times 10^{-3} - 2.982 \times 10^{-4}} = 8.1427 \times 10^5 \text{ psi}$$

However, C_p cannot be evaluated by Eq. (11.8.12) because $\dot{\epsilon}^f$ approaches infinity at $t = 0$. Instead, C_p may be estimated by using K_p , C_s and ϵ^f at some finite time span, e.g., for $t = 10000$ hours and $\sigma = 1000$ psi

$$\epsilon^f(10000) = 7.031 \times 10^{-4}$$

$$C_p \simeq 9.71 \times 10^9 \text{ lb/in}^2/\text{hr}$$

11.8.4 Analysis Procedure and Results

The computation process, as presented in the preceding sections, has been coded. Table 11.8.5 shows the source listing of this program. The analysis starts by running the program "RHEOLOGY", which is an interactive program. The program prompts six questions to be answered as shown in Table 11.8.1. These questions pertain to:

1. Least square fit on the strain curve or strain-rate curve (default = strain curve).
2. Title to appear in the output (no default).
3. Error tolerance for the convergence criterion (default = 10^{-6}).
4. Maximum time for creep (default = 5×10^5) and the number of data points to be used for curve fitting (default = 200).
5. Maximum operating stress (default = 1500) and stress increments (default = 100).
6. Output file names for time vs. strain data (default = RHEOLOGY.OUT) and stress vs. rheological parameters (default = RHEOLOGY.PAM).

Upon completion of the dialogue, the iterative and incremental computation process proceeds with the iteration information printed on the terminal screen as shown in Table 11.8.1. Upon termination of the job, there will be two output files as shown in Tables 11.8.2 and 11.8.3, respectively. Table 11.8.3 shows rheological parameter values (C_s , C_n and K_n) as functions of stress, which can be converted into data specified on the CREEP and TABLES1 Bulk Data entries.

An interactive graphics program (DISPLY) is available to plot curves for the data generated by the program RHEOLOGY. Table 11.8.6 shows the source listing of the DISPLY program (to be linked with Tektronix PLOT10 package).

The DISPLY program starts by prompting questions as shown in Table 11.8.4. The results of processing the file "RHEOLOGY.OUT" in Table 11.8.2 are shown in Fig. 11.8.1 and 11.8.2. Fig. 11.8.3, 11.8.4, and 11.8.5 show plots of rheological parameters saved in the file "RHEOLOGY.PAM" in Table 11.8.3.

The least square fit on the strain-rate curve was attempted to determine the rheological parameters. This approach was not fruitful because the Hessian matrix became singular within the operating time domain.

11.8.5 Recommendations

The rheological parameters corresponding to any empirical creep law may be obtained by using the program "RHEOLOGY". The listing in Table 11.8.5 shows FUNCTION EPSF defining the Findley's creep law in Eq. (11.8.2) and its time derivative. This FUNCTION has to be

redefined for a different type of empirical formula. It is also possible to use experimental data for the curve fitting in place of the empirical law with slight modifications to the current version of "RHEOLOGY".

The least square fit can also be applied to the creep laws defined by types 111 through 222 to be specified in the CREEP Bulk Data entry. For those creep laws, coefficients "a" through "g" should be determined, rather than the rheological parameters, by the curve fitting. In this case, the time and the stress must be treated as a unified variable. This means that all the data points obtained by sweeping through the time domain, nested under the stress increments across the operating stress range should be included in the iterative process for the least square fit.

Table 11.8.1 Run Progression of the Program "RHEOLOGY"

\$RUN RHEOLOGY

* PERFORMS LEAST SQUARE FIT FOR RHEOLOGICAL PARAMETERS

** WRITTEN BY SANG H. LEE.

1. LEAST SQUARE FIT ON THE STRAIN RATE? Y OR N (DEFAULT)

2. TYPE A TITLE(MAX. 40 CHARACTERS).

LEAST SQUARE FIT ON FINDLEY'S CREEP LAW FOR RHEOLOGICAL MODEL.

3. TYPE IN AN ERROR TOLERANCE FOR ITERATION. (DEFAULT= 1.E-6)
1.E-8

4. DEFINE AN X-AXIS IN TIME.

TYPE MAXIMUM TIME DOMAIN:TMAX (DEFAULT=5.E5)

HOW MANY POINTS WOULD YOU USE FOR CURVE FIT? (DEFAULT=200)

5. DEFINE THE RANGE OF SECOND VARIABLE, STRESS.

TYPE IN THE MAXIMUM OPERATING STRESS: SMAX (DEFAULT=1500)

TYPE IN THE STRESS INCREMENTS SWEEPING STRESS RANGE (DEFAULT=100)

6. OUTPUT FILE NAMES:

TYPE FILE NAME FOR TIME VS. STRAIN; DEFAULT=RHEOLOGY.OUT

TYPE FILE NAME FOR STRESS VS. CS/CP/KP; DEFAULT=RHEOLOGY.PAM

1 LEAST SQUARE FIT AT STRESS 1.000000E+02

ITERATION	CS	CP	KP	ERROR
0	1.6804607E+12	1.1392271E+10	8.1741463E+05	2.4587280E-08
1	1.4885375E+11	1.2051061E+10	9.6216294E+05	5.4180528E-06
2	2.7227526E+11	1.1337793E+10	9.9562881E+05	1.0784001E-06
3	4.5875800E+11	1.0767391E+10	1.0000585E+06	1.8038159E-07
4	6.7277285E+11	1.0465592E+10	1.0020148E+06	2.2354318E-08
5	8.1780644E+11	1.0304219E+10	1.0030722E+06	5.3207905E-09
6	8.5469646E+11	1.0216611E+10	1.0036497E+06	4.7783750E-09
7	8.5576620E+11	1.0168655E+10	1.0039670E+06	4.7770992E-09
8	8.5539796E+11	1.0142284E+10	1.0041417E+06	4.7770010E-09

2 LEAST SQUARE FIT AT STRESS 2.000000E+02

ITERATION	CS	CP	KP	ERROR
0	1.6803444E+12	1.1390349E+10	8.1731938E+05	9.8367003E-08
1	1.4883239E+11	1.2048701E+10	9.6204513E+05	2.1678970E-05

Table 11.8.2 Strain/Strain-Rate vs. Time (RHEOLOGY.OUT File Content)

1 AT STRESS 1.0000E+02 CS= 8.5494E+11 CP= 1.0110E+10 KP= 1.0044E+06

TIME	STRAIN/EMPIRICAL	RHEOLOGICAL	STR-RATE/EMPIRICAL	RHEOLOGICAL
2.500000E+03	5.271788E-05	2.218916E-05	4.349165E-09	7.832982E-09
5.000000E+03	6.077321E-05	3.956276E-05	2.480685E-09	6.136065E-09
7.500000E+03	6.600114E-05	5.317984E-05	1.786231E-09	4.812336E-09
1.000000E+04	6.996244E-05	6.386654E-05	1.414938E-09	3.779724E-09
1.250000E+04	7.318766E-05	7.226731E-05	1.180974E-09	2.974205E-09
1.500000E+04	7.592629E-05	7.888488E-05	1.018834E-09	2.345837E-09
.				
.				
4.850000E+05	1.511870E-04	1.562951E-04	6.099392E-11	1.169668E-10
4.875000E+05	1.513392E-04	1.565875E-04	6.074044E-11	1.169668E-10
4.900000E+05	1.514907E-04	1.568799E-04	6.048929E-11	1.169668E-10
4.925000E+05	1.516417E-04	1.571723E-04	6.024047E-11	1.169668E-10
4.950000E+05	1.517919E-04	1.574648E-04	5.999391E-11	1.169668E-10
4.975000E+05	1.519416E-04	1.577572E-04	5.974959E-11	1.169668E-10
5.000000E+05	1.520907E-04	1.580496E-04	5.950749E-11	1.169668E-10

2 AT STRESS 2.0000E+02 CS= 8.5488E+11 CP= 1.0108E+10 KP= 1.0042E+06

TIME	STRAIN/EMPIRICAL	RHEOLOGICAL	STR-RATE/EMPIRICAL	RHEOLOGICAL
2.500000E+03	1.054546E-04	4.438558E-05	8.698933E-09	1.566844E-08
5.000000E+03	1.215664E-04	7.913807E-05	4.961714E-09	1.227393E-08
7.500000E+03	1.320230E-04	1.063761E-04	3.572710E-09	9.625984E-09
1.000000E+04	1.399461E-04	1.277523E-04	2.830072E-09	7.560396E-09
1.250000E+04	1.463970E-04	1.445559E-04	2.362111E-09	5.949092E-09
.				
.				
4.900000E+05	3.030140E-04	3.137931E-04	1.209870E-10	2.339499E-10
4.925000E+05	3.033158E-04	3.143780E-04	1.204893E-10	2.339499E-10
4.950000E+05	3.036164E-04	3.149629E-04	1.199961E-10	2.339499E-10
4.975000E+05	3.039158E-04	3.155478E-04	1.195075E-10	2.339499E-10
5.000000E+05	3.042140E-04	3.161326E-04	1.190232E-10	2.339499E-10

3 AT STRESS 3.0000E+02 CS= 8.5476E+11 CP= 1.0103E+10 KP= 1.0040E+06

TIME	STRAIN/EMPIRICAL	RHEOLOGICAL	STR-RATE/EMPIRICAL	RHEOLOGICAL
2.500000E+03	1.582290E-04	6.660714E-05	1.304991E-08	2.351199E-08
5.000000E+03	1.823994E-04	1.187549E-04	7.443429E-09	1.841699E-08
7.500000E+03	1.980861E-04	1.596242E-04	5.359683E-09	1.444280E-08
.				
.				
.				

Table 11.8.3 $C_s/C_p/K_p$ vs. Stress (RHEOLOGY.PAM File Content)

LEAST SQUARE FIT ON FINDLEY'S CREEP LAW FOR RHEOLOGICAL MODEL.

STRESS	CS	CP	KP
1.000000E+02	8.549437E+11	1.010986E+10	1.004357E+06
2.000000E+02	8.548840E+11	1.010815E+10	1.004229E+06
3.000000E+02	8.547586E+11	1.010342E+10	1.004030E+06
4.000000E+02	8.545849E+11	1.009695E+10	1.003749E+06
5.000000E+02	8.543681E+11	1.008911E+10	1.003385E+06
6.000000E+02	8.541048E+11	1.007964E+10	1.002940E+06
7.000000E+02	8.537940E+11	1.006847E+10	1.002414E+06
8.000000E+02	8.534354E+11	1.005559E+10	1.001808E+06
9.000000E+02	8.530334E+11	1.004132E+10	1.001120E+06
1.000000E+03	8.525761E+11	1.002481E+10	1.000356E+06
1.100000E+03	8.520762E+11	1.000697E+10	9.995104E+05
1.200000E+03	8.515345E+11	9.987841E+09	9.985831E+05
1.300000E+03	8.509324E+11	9.966111E+09	9.975832E+05
1.400000E+03	8.502909E+11	9.943285E+09	9.965015E+05
1.500000E+03	8.496271E+11	9.920581E+09	9.953303E+05

Table 11.8.4 Run Progression of the Program "DISPLY"

```
$RUN DISPLY
THIS IS A GENERAL PURPOSE PROGRAM FOR XYPLOT
      WRITTEN BY SANG H. LEE.

TYPE IN BAUD RATE

1.  TYPE A TITLE(MAX. 40 CHARACTERS).
    DEFAULT IS TO READ THE FIRST LINE FROM INPUT FILE

2.  NUMBER OF DATA POINTS?
    DEFAULT WILL INCLUDE ALL THE DATA POINTS IN THE FILE.

3.  X-AXIS DEFINITION.
    LINEAR OR LOGARITHMIC? TYPE  LN OR LG

    WHICH DOMAIN DO YOU WANT TO DISPLAY? TYPE XMIN,XMAX

    TYPE A LABEL(MAX. 10 CHARACTERS).

    TYPE A LABEL(MAX. 10 CHARACTERS).
TIME IN HOUR
4.  Y-AXIS DEFINITION.
    LINEAR OR LOGARITHMIC? TYPE  LN OR LG

RANGE FOR Y-AXIS WILL BE SET AUTOMATICALLY.
    TYPE A LABEL(MAX. 10 CHARACTERS).
CREEP STRAIN
5.  GRID LINES? Y OR N.

6.  OVERLAY TWO CURVES? Y OR N.
Y
    TYPE 1, 2, 3 OR 4 TO SPECIFY INPUT DATA FORMAT:
        1 IF INPUTS ARE  X,  Y1, Y2
        2 IF INPUTS ARE  X1, X2, Y
        3 IF INPUTS ARE  X1, Y1, X2, Y2
        4 IF INPUTS ARE  X, DUMMY, DUMMY, Y1, Y2
1
    1ST CURVE WILL BE A SOLID LINE; 2ND WILL BE DASHED
    TYPE A LEGEND FOR SOLID LINE.
FINDLEY
    TYPE A LEGEND FOR DASHED LINE.
RHEOLOGY
7.  TYPE THE FILE NAME.

RHEOLOGY.OUT
```

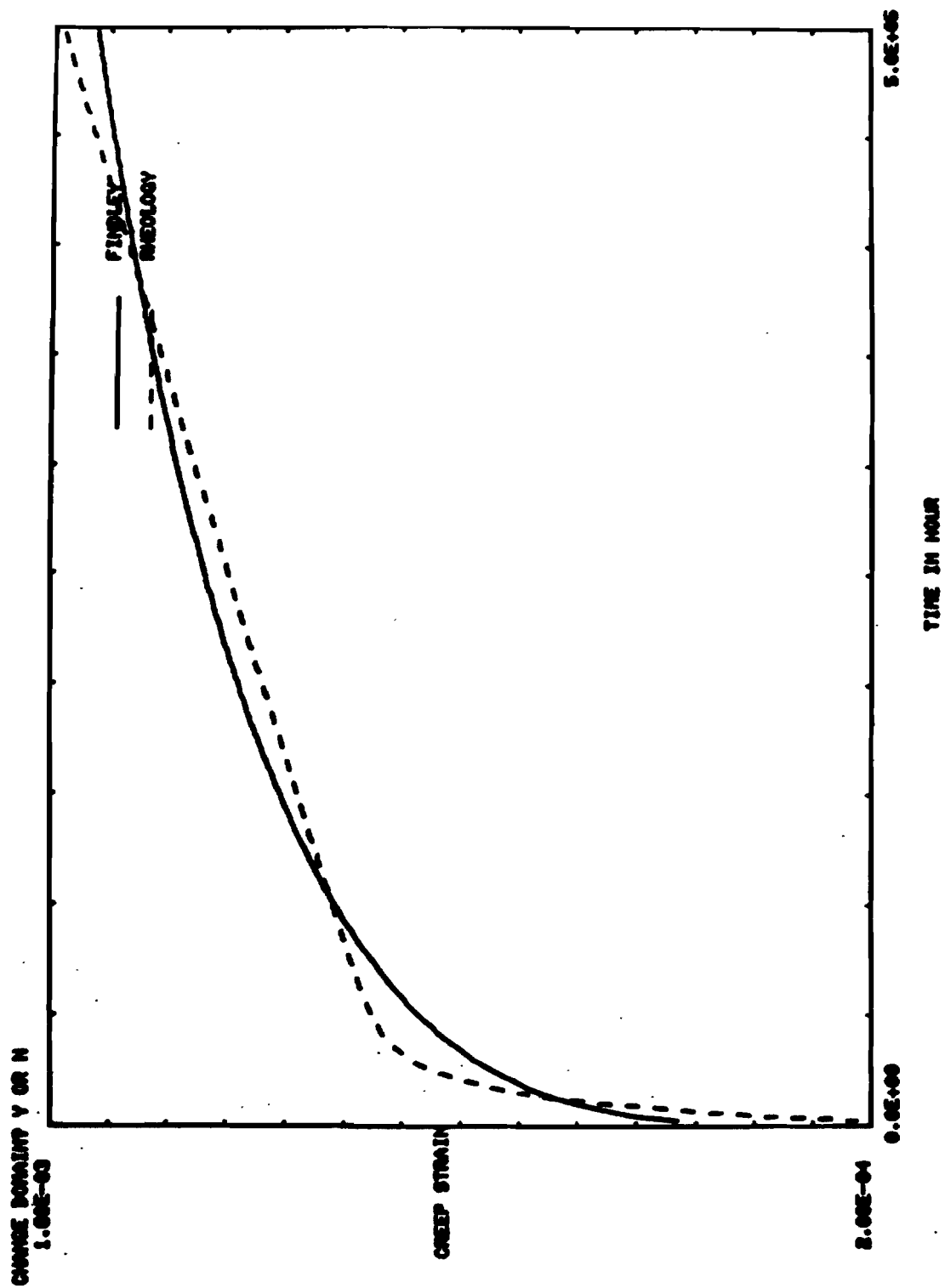

NUMBER OF DATA POINTS= 200

XMIN= 0.000E+00 XMAX= 0.500E+06

YMIN= 0.200E-04 YMAX= 0.160E-03

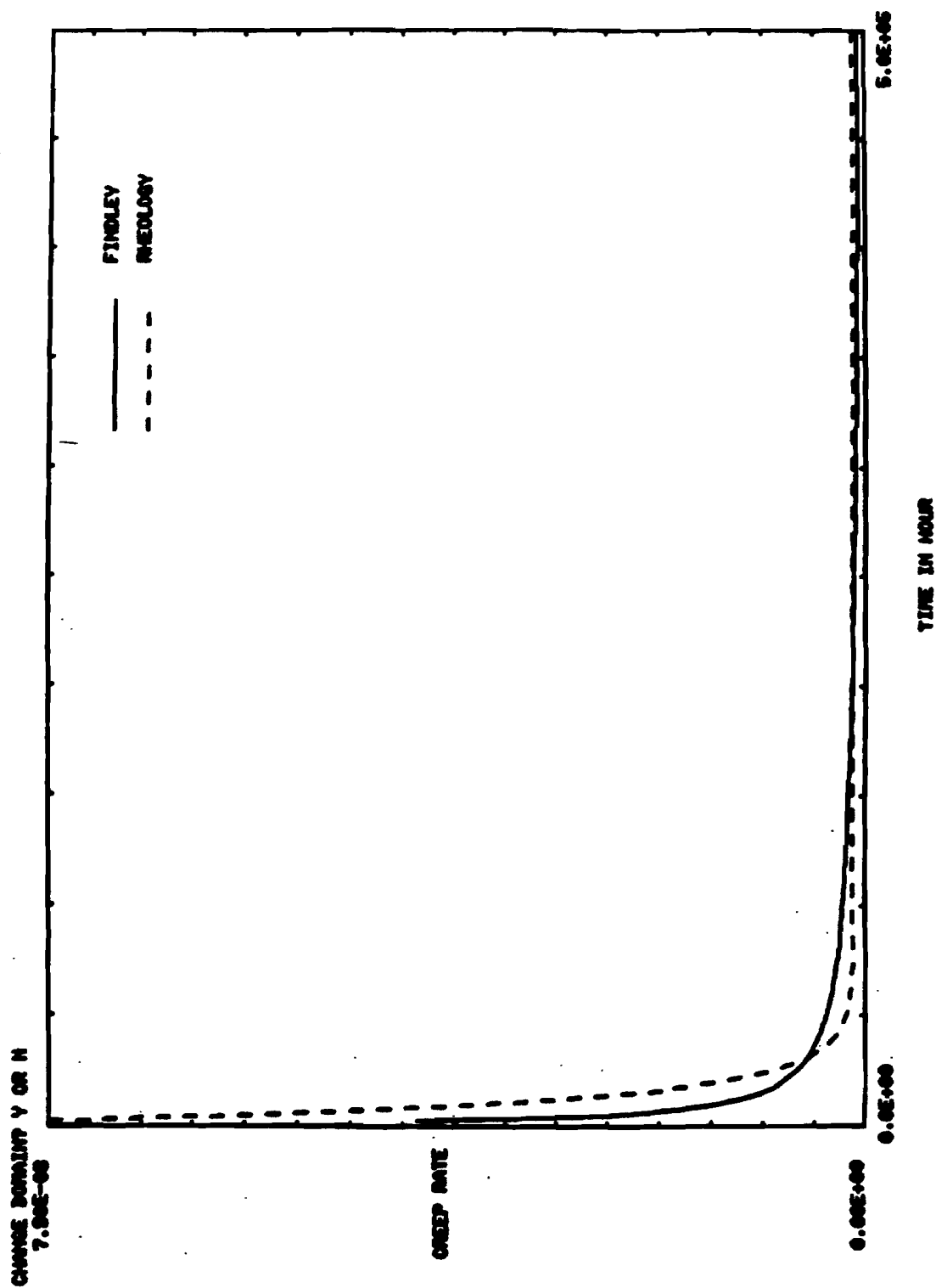
HIT RETURN TO CONTINUE.

.
.
.



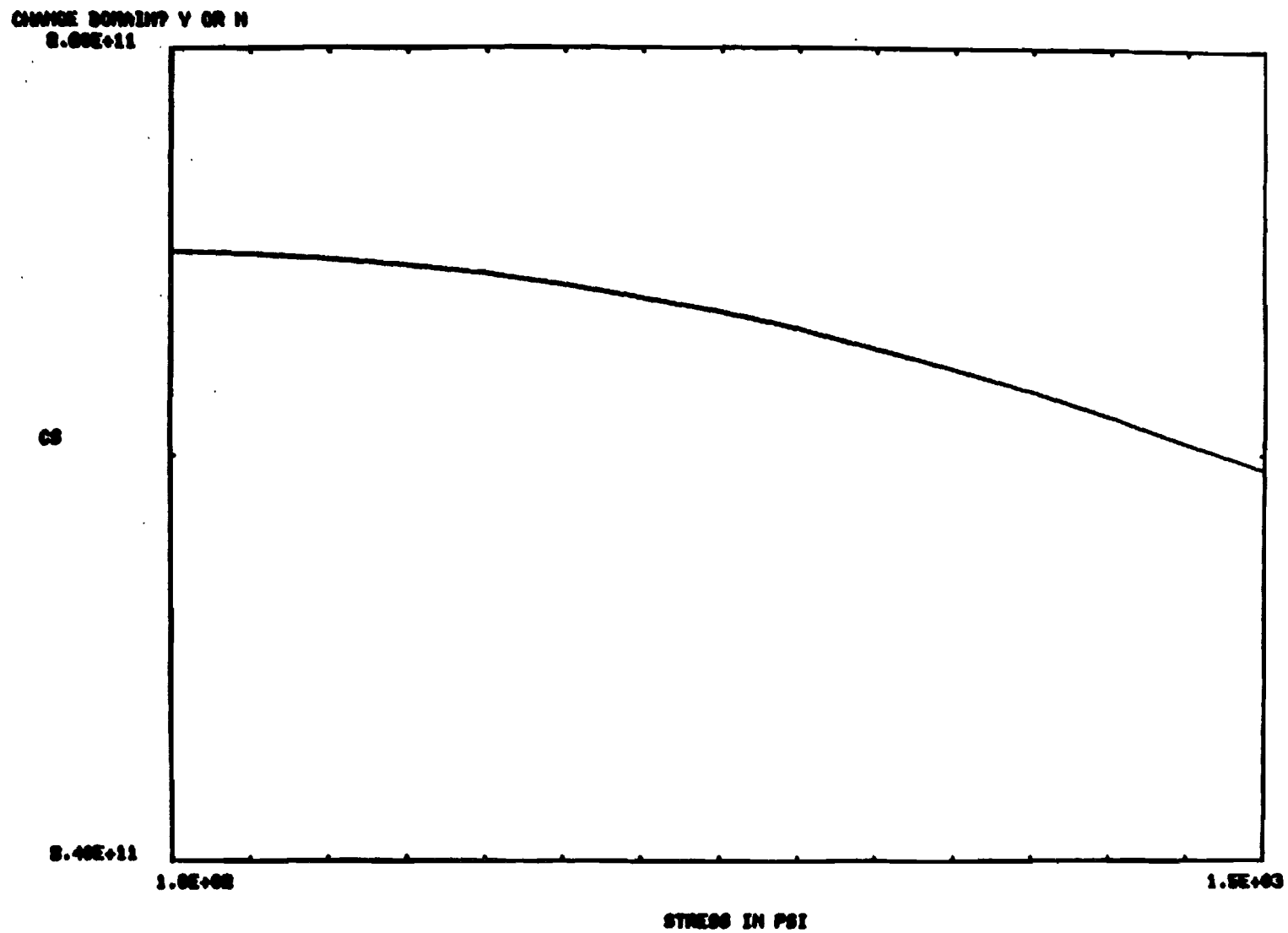
10 AT STRESS 1.0000E+03 CS- 8.5258E+11 CP- 1.0025E+10 KP- 1.0004E+06

Figure 11.8.1 *Creep Strain vs. Time at 1000 psi*



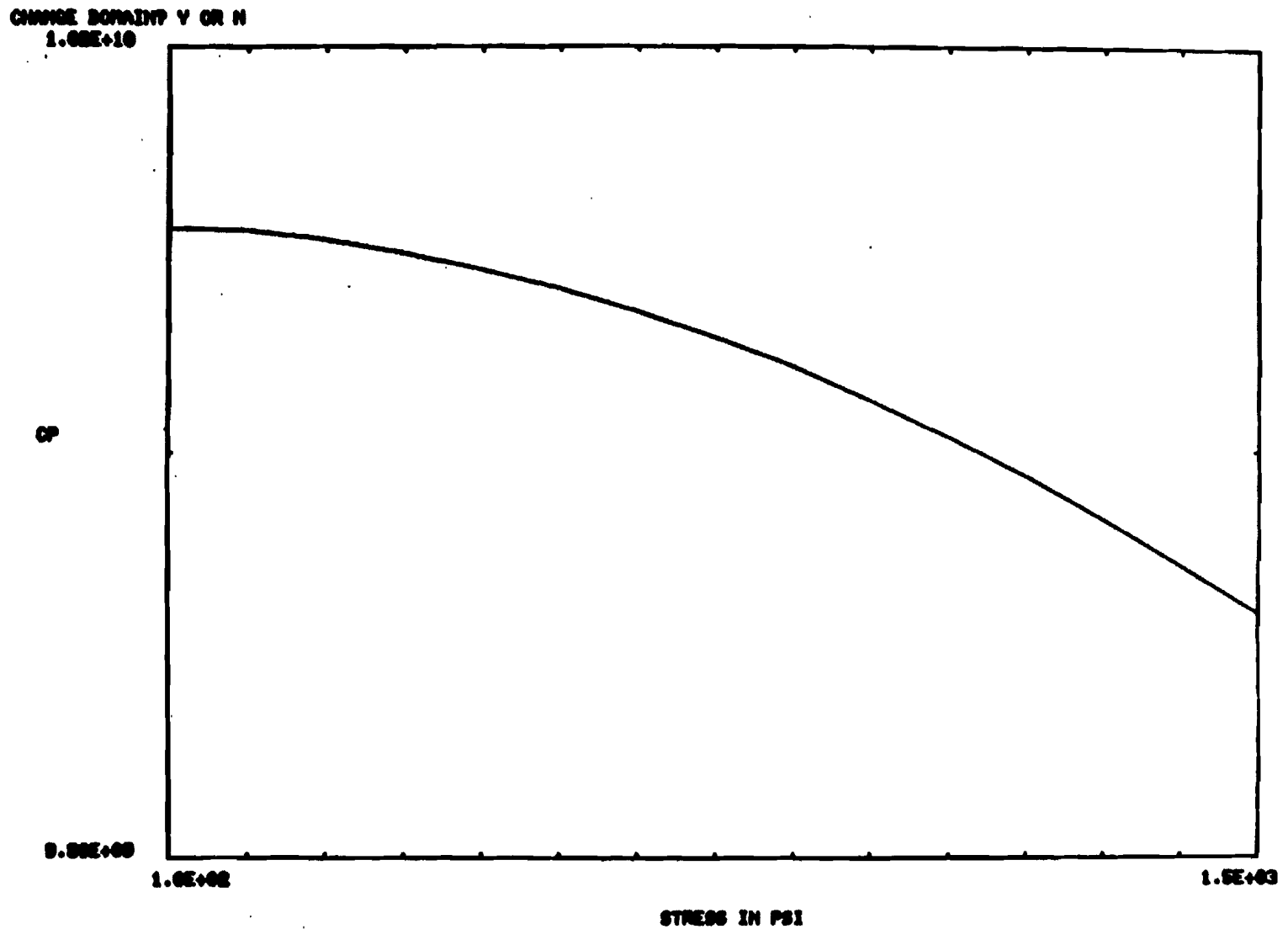
10 AT STRESS 1.0000E+03 CS- 8.5258E+11 CP- 1.0025E+10 KP- 1.0004E+06

Figure 11.8.2 *Creep Strain-Rate vs. Time*



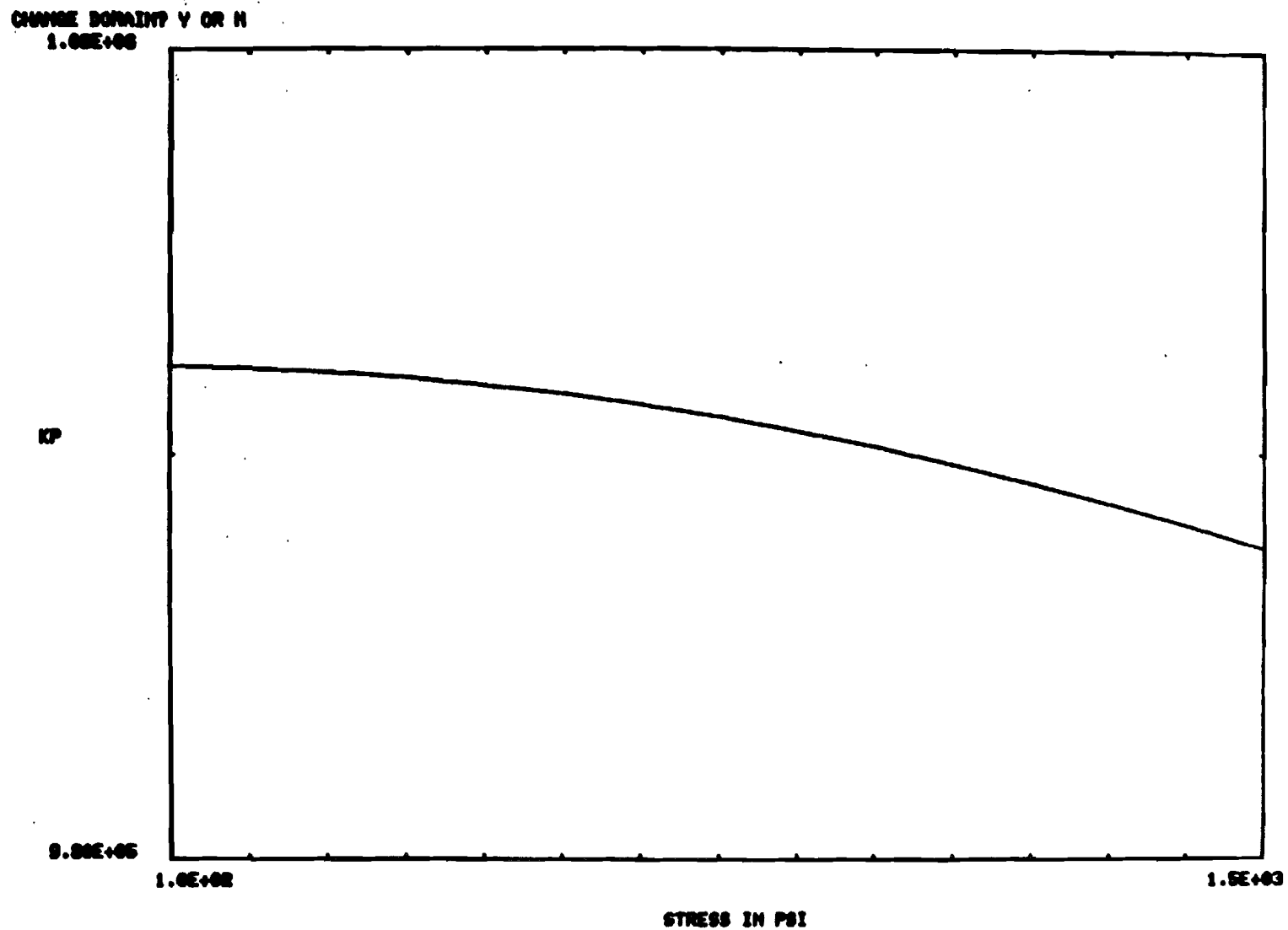
LEAST SQUARE FIT ON FINDLEY'S CREEP LAW FOR RHEOLOGICAL MODEL.

Figure 11.8.3 ***C_s vs. Stress***



LEAST SQUARE FIT ON FINDLEY'S CREEP LAW FOR RHEOLOGICAL MODEL.

Figure 11.8.4 ***Cp vs. Stress***



LEAST SQUARE FIT ON FINDLEY'S CREEP LAW FOR RHEOLOGICAL MODEL.

Figure 11.8.5 ***Kp vs. Stress***

Table 11.8.5. Source Listing for the Program "RHEOLOGY"

```

C
C
C   PROGRAM RHEOLOGY(INPUT,OUTPUT,TAPE6=OUTPUT)
C
C                                   BY SANG H. LEE,  11/6/87
C
C   THIS PROGRAM PROVIDES RHEOLOGICAL PARAMETERS KP, CP AND CS,
C   CORRESPONDING TO THE GIVEN EMPIRICAL CREEP LAW (FINDLEY'S)
C   USING THE LEAST SQUARES FIT.
C
C   COMMON / /TITLE(20)
C       CHARACTER NAME1*20,NAME2*20,IBK*1
C       DATA  IBK/' ' /
C INPUT DATA INTERACTIVELY
C
C       CALL INTERACT(NF,ERROR,TMAX,ANDP,SMAX,DSIGMA,NAME1,NAME2)
C
C *****
C
C       DEFAULT VALUES ARE DEFINED HERE.
C
C       IF(ERROR.EQ.0.)ERROR=1.E-6
C       TMIN=0.
C       IF(TMAX.EQ.0.)TMAX=500000.
C       IF(ANDP.EQ.0.) ANDP=200.
C       DTIME=(TMAX-TMIN)/ANDP
C       TINIT=DTIME*4.
C       IF(SMAX.EQ.0.)SMAX=1500.
C       IF(DSIGMA.EQ.0.) DSIGMA=100.
C CREATE AN EXTERNAL FILE.
C       IF(NAME1.EQ.IBK)NAME1='RHEOLOGY.OUT'
C       IF(NAME2.EQ.IBK)NAME2='RHEOLOGY.PAM'
C       OPEN(UNIT=7,FILE=NAME1,STATUS='NEW')
C       OPEN(UNIT=8,FILE=NAME2,STATUS='NEW')
C
C INPUT TITLE
C       WRITE(8,1001)(TITLE(I),I=1,20)
C       WRITE(8,1002)
1001 FORMAT(X,20A4)
1002 FORMAT(/5X,'STRESS',10X,'CS'14X'CP'14X'KP')
C NUMBER OF LINES
C       NL=0
C       SIGMA=0.
100 SIGMA=SIGMA+DSIGMA
C       IF(SIGMA.GT.SMAX)GO TO 200
C       NL=NL+1
C       CALL PROCESS(NL,NF,SIGMA,DTIME,TMAX,TINIT,ERROR)
C       WRITE(7,1003)

```

```

1003 FORMAT(1H )
      GO TO 100
200 CONTINUE
      CLOSE(UNIT=7)
      CLOSE(UNIT=8)
      STOP
      END

C
C *****
C
      FUNCTION EPSF(N,STR,TIM)
C
C FINDLEY'S CREEP LAW AND ITS TIME DERIVATIVE
C
      IF(N.EQ.1)THEN
          EPSF=0.0067*SINH(STR/8500.)
1          +0.0011*TIM**0.19*SINH(STR/8500.)-STR/1.2E6
          ELSE
          EPSF=0.19*0.0011*TIM**(0.19-1.)*SINH(STR/8500.)
      ENDIF
C
      RETURN
      ENTRY EPSFDT(STR,TIM)
      EPSFDT=0.19*0.0011*TIM**(0.19-1.)*SINH(STR/8500.)
      RETURN
      END

C
C *****
C
      FUNCTION EPSR(N,TIM)
      COMMON/STHD/STRESS,CS,CP,KP
      REAL KP
C
C RHEOLOGICAL MODEL
C
      IF(N.EQ.1)THEN
          EPSR=STRESS*TIM/CS+STRESS*(1.-EXP(-KP*TIM/CP))/KP
      ELSE
          EPSR=STRESS/CS+STRESS*EXP(-KP*TIM/CP)/CP
      ENDIF
      RETURN
C DERIVATIVES
      ENTRY FU1F(N,TIM)
      IF(N.EQ.1)THEN
          FU1F=-STRESS*TIM/CS**2
      ELSE
          FU1F=-STRESS/CS**2
      ENDIF
      RETURN

```



```

ENTRY FU2F(N,TIM)
  IF(N.EQ.1)THEN
    FU2F=-STRESS*TIM*EXP(-KP*TIM/CP)/CP**2
  ELSE
    FU2F=STRESS*(KP*TIM/CP-1.)*EXP(-KP*TIM/CP)/CP**2
  ENDIF
RETURN
ENTRY FU3F(N,TIM)
  IF(N.EQ.1)THEN
    FU3F=STRESS*EXP(-KP*TIM/CP)*(1./KP+TIM/CP)/KP-STRESS/KP**2
  ELSE
    FU3F=-STRESS*TIM*EXP(-KP*TIM/CP)/CP**2
  ENDIF
RETURN
END

C
C
SUBROUTINE INTERACT(NF,ERROR,TMAX,ANDP,SMAX,DSIG,NAME1,NAME2)
  CHARACTER NAME1*20,NAME2*20
  COMMON / /TITLE(20)
  DATA IY,IN,IBK/'Y','N',' '/
  WRITE(6,999)
999 FORMAT(/X'* PERFORMS LEAST SQUARE FIT FOR RHEOLOGICAL PARAMETERS'/
1 10X,'** WRITTEN BY SANG H. LEE.'/)

C
C  LEAST SQUARE FIT ON THE STRAIN OR STRAIN RATE?
C
  NF=0
100 WRITE(6,998)
998 FORMAT(X,'1.  LEAST SQUARE FIT ON THE STRAIN RATE?',
1 2X,' Y OR N (DEFAULT)')
  READ(5,1013)IANS
  IF(IANS.EQ.IBK)IANS=IN
  IF(IANS.EQ.IY)NF=2
  IF(IANS.EQ.IN)NF=1
  IF(NF.EQ.0)GO TO 100
C  INPUT TITLE
300 TYPE 1000
  ACCEPT 1001,(TITLE(I),I=1,20)
1000 FORMAT(X,'2.  TYPE A TITLE(MAX. 40 CHARACTERS).')
1001 FORMAT(20A4)
C  ERROR TOLERANCE
  TYPE 1002
1002 FORMAT(X,'3.  TYPE IN AN ERROR TOLERANCE FOR ITERATION.'
1 ,X,'(DEFAULT= 1.E-6)')
  ACCEPT 1003,ERROR
1003 FORMAT(F20.0)
C  X-AXIS IS TIME.
  TYPE 1004

```

```

1004 FORMAT(X,'4.  DEFINE AN X-AXIS IN TIME.')
```

20 TYPE 1005

```

1005 FORMAT(5X,'TYPE MAXIMUM TIME DOMAIN:TMAX (DEFAULT=5.E5)')
      ACCEPT 1003,TMAX
1006 FORMAT(2F15.0)
C   DATA POINTS IN TIME
      TYPE 1007
1007 FORMAT(5X,'HOW MANY POINTS WOULD YOU USE FOR CURVE FIT?'
1    ,2X'(DEFAULT=200)')
      ACCEPT 1003,ANDP
C   SECOND VARIABLE IS STRESS
      TYPE 1008
1008 FORMAT(X,'5.  DEFINE THE RANGE OF SECOND VARIABLE, STRESS.')
```

TYPE 1009

```

1009 FORMAT(5X,'TYPE IN THE MAXIMUM OPERATING STRESS: SMAX'
1    ,2X'(DEFAULT=1500)')
      ACCEPT 1003, SMAX
      TYPE 1010
1010 FORMAT(5X,'TYPE IN THE STRESS INCREMENTS SWEEPING STRESS RANGE'
1    ,2X'(DEFAULT=100)')
      ACCEPT 1003,DSIG
C   OUTPUT FILE NAMES FOR STRAIN VS. TIME AND CS/CP/KP VS. STRESS
      TYPE 1014
1014 FORMAT(X,'6.  OUTPUT FILE NAMES:')
```

TYPE 1011

```

      READ(5,1013)NAME1
1011 FORMAT(5X,'TYPE FILE NAME FOR TIME VS. STRAIN;'
1    X,'DEFAULT=RHEOLOGY.OUT')
      TYPE 1012
      READ(5,1013)NAME2
1012 FORMAT(5X,'TYPE FILE NAME FOR STRESS VS. CS/CP/KP;'
1    X'DEFAULT=RHEOLOGY.PAM')
1013 FORMAT(A)
      RETURN
      END
C
C
      SUBROUTINE PROCESS(NL,NF,SIGMA,DELTIME,TMAX,TINIT,ERLMT)
      COMMON/STHD/STRESS,CS,CP,KP
      DIMENSION X(901),Y(901),YRATE(901)
      REAL KP
C
      STRESS=SIGMA
      I=0
      X(I)=0.
C
10  I=I+1
      X(I)=X(I-1)+DELTIME
      IF(X(I).GT.TMAX)GO TO 200
```

```

      NP=I
      Y(I)=EPSF(1,STRESS,X(I))
      YRATE(I)=EPSF(2,STRESS,X(I))
      GO TO 10
C      READ 1001,X(I),Y(I),J
C      IF(J.EQ.0)GO TO 10
C      NP=I
C 1001 FORMAT(2F10.0,50X, I10)
      200 CONTINUE
      WRITE(6,1000)NL, STRESS
      CALL GUESS(TMAX,TINIT)
      IF(NF.EQ.1)THEN
        CALL CURVFIT(NF,ERLMT,X,Y,NP)
      ELSE
        CALL CURVFIT(NF,ERLMT,X,YRATE,NP)
      ENDIF
C
      WRITE(7,999)NL,STRESS,CS,CP,KP
      WRITE(8,1001)STRESS,CS,CP,KP
      WRITE(7,1002)
999 FORMAT(I4,2X'AT STRESS',1PE11.4,
1      X,3HCS=,E11.4,X,3HCP=,E11.4,X,3HKP=,E11.4/)
1000 FORMAT(1H0,I5,3X,'LEAST SQUARE FIT AT STRESS' ,1PE15.6/)
1001 FORMAT(5(1PE16.6))
1002 FORMAT(8X,'TIME      STRAIN/EMPIRICAL RHEOLOGICAL',
1      2X,'STR-RATE/EMPIRICAL RHEOLOGICAL')
C
      DO 99 I=1,NP
      STRAIN=EPSR(1,X(I))
      STRATE=EPSR(2,X(I))
      WRITE(7,1001) X(I),Y(I),STRAIN,YRATE(I),STRATE
99 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE GUESS(TIMEF,TIMEI)
C  COMPUTE INITIAL VALUES OF CS, CP AND KP FOR ITERATION.
C
      COMMON/STHD/STRESS,CS,CP,KP
      REAL KP
C
      CS=STRESS/EPSFDT(STRESS,TIMEF)
      TEMP=EPSF(1,STRESS,TIMEF)
      TEMP=TEMP-STRESS*TIMEF/CS
      KP=STRESS/TEMP
      TEMP=EPSF(1,STRESS,TIMEI)-STRESS*TIMEI/CS
      TEMP=TEMP*KP/STRESS
      TEMP=-ALOG(1.-TEMP)

```

```

      CP=KP*TIMEI/TEMP
C
      RETURN
      END
C
C
      SUBROUTINE CURVFIT(NF,ERLMT,X,YY,NN)
      DIMENSION X( 1),Y(901),C(20),YY( 1)
      COMMON/STHD/STRESS,CS,CP,KP
      REAL KP
C
      ERROR=1.
      M=3
      I=0
      WRITE(6,999)
50  ERR=0.
      DO 20 N=1,NN
      Y(N)=YY(N)-EPSR(NF,X(N))
      ERR=ERR+Y(N)**2
20  CONTINUE
C
      WRITE(6,1002) I,CS,CP,KP,ERR
C  CONVERGENCE TEST
      IF(ERR.LT.ERLMT**2)GO TO 100
      IF(ABS(ERROR/ERR-1.).LT.ERLMT)GO TO 100
      IF(ERR.GT.1000.*ERROR)GO TO 200
      IF(I.GT.30)GO TO 200
C
      I=I+1
      ERROR=ERR
      CALL LSTSQ(NF,ERLMT,X,Y,NN,C,M)
      CS=CS+C(1)
      CP=CP+C(2)
      KP=KP+C(3)
      GO TO 50
200 WRITE(6,1003)
100 CONTINUE
C
999  FORMAT(5X,'ITERATION'9X,2HCS,10X,2HCP,10X,2HKP,9X,5HERROR)
1001 FORMAT(2F10.0)
1002 FORMAT(X,I9,4(1PE15.7))
1003 FORMAT(/,5X,'ITERATION DOES NOT CONVERGE. ')
      RETURN
      END
C
C
      SUBROUTINE LSTSQ(NF,ERLMT,X,Y,N,C,M)
C      COMMON/STHD/STRESS,CS,CP,KP
      DIMENSION X(1),Y(1),C(20),F(20,901),A(20,20),B(20)

```

```

C      REAL KP
C      DERIVATIVES
C      FU1F(T)=-STRESS*T/CS**2
C      FU2F(T)=-STRESS*T*EXP(-KP*T/CP)/CP**2
C      FU3F(T)=STRESS*EXP(-KP*T/CP)*(1./KP+T/CP)/KP-STRESS/KP**2
C
      DO 1 J=1,N
      F(1,J)=FU1F(NF,X(J))
      F(2,J)=FU2F(NF,X(J))
1 F(M,J)=FU3F(NF,X(J))
C
      DO 3 I=1,M
      DO 3 K=1,I
      A(K,I)=0.
      DO 2 J=1,N
2 A(K,I)=A(K,I)+F(I,J)*F(K,J)
3 A(I,K)=A(K,I)
C
      DO 4 K=1,M
      B(K)=0.
      DO 4 J=1,N
4 B(K)=B(K)+Y(J)*F(K,J)
C      CALL GAUSID(A,M,B,C,ERLMT)
      CALL ELIM(A,M,B,C)
      RETURN
      END
C
C
      SUBROUTINE ELIM(AA,N,BB,X )
      DIMENSION AA(20,20),BB(20),A(20,21),X(20),ID(20)
      NN=N+1
      DO 100 I=1,N
      A(I,NN)=BB(I)
      ID(I)=I
      DO 100 J=1,N
100 A(I,J)=AA(I,J)
      K=1
      1 CALL EXCH1(A,N,NN,K,ID)
      2 IF(A(K,K))3,999,3
      3 KK=K+1
      DO 4 J=KK,NN
      A(K,J)=A(K,J)/A(K,K)
      DO 4 I=1,N
      IF(K-I)41,4,41
41 A(I,J)=A(I,J)-A(I,K)*A(K,J)
      4 CONTINUE
      K=KK
      IF(K-N)1,2,5
      5 DO 10 I=1,N

```

```

      DO 10 J=1,N
      IF(ID(J)-I)10,6,10
      6 X(I)=A(J,NN)
      10 CONTINUE
      RETURN
      999 WRITE(6,1000)
      1000 FORMAT(5X,'NO UNIQUE SOLUTION.')
```

C

C

```

      SUBROUTINE EXCH1(A,N,NN,K,ID)
      DIMENSION A(20,21),ID(20)
      NROW=K
      NCOL=K
      B=ABS(A(K,K))
      DO 2 I=1,N
      DO 2 J=1,N
      IF(ABS(A(I,J))-B)2,2,21
      21 NROW=I
      NCOL=J
      B=ABS(A(I,J))
      2 CONTINUE
      IF(NROW-K)3,3,31
      31 DO 32 J=K,NN
      C=A(NROW,J)
      A(NROW,J)=A(K,J)
      32 A(K,J)=C
      3 CONTINUE
      IF(NCOL-K)4,4,41
      41 DO 42 I=1,N
      C=A(I,NCOL)
      A(I,NCOL)=A(I,K)
      42 A(I,K)=C
      I=ID(NCOL)
      ID(NCOL)=ID(K)
      ID(K)=I
      4 CONTINUE
      RETURN
      END
```

C

C

```

      SUBROUTINE GAUSID(A,N,B,X,ERR)
      DIMENSION A(20,20),B(20),C(20,20),X(20)
      K=0
      NN=N+1
      DO 11 I=1,N
      IF(A(I,I))12,6,12
      12 X(I)=1.
```

```

      C(I,NN)=B(I)/A(I,I)
      DO 11 J=1,N
11  C(I,J)=A(I,J)/A(I,I)
      1 CONTINUE
      E=0.
C
      DO 3 I=1,N
      P=C(I,NN)
      DO 2 J=1,N
      P=P-C(I,J)*X(J)
2  CONTINUE
      X(I)=X(I)+P
      E=E+ABS(P)
3  CONTINUE
      IF(E-ERR)4,4,5
4  RETURN
5  K=K+1
      IF(100-K)6,1,1
6  WRITE(6,1000)
      RETURN
1000 FORMAT(5X,26H GAUSID DOES NOT CONVERGE.)
      END

```

Table 11.8.6. Source Listing for the Program "DISPLY"

```

C
C
C      PROGRAM DISPLY                      SHL 11/16/87
C
C      GENERALIZED PLOT PROGRAM
C
      COMMON/DT/NDP,XX1(901),YY1(901),XX2(901),YY2(901)
      COMMON/TTL/TITLE(20),XLABEL(4),YLABEL(4),SOLID(4),DASHED(4)
      CHARACTER NAME*20
      DATA IY,IN,IP,ID,LN,LG,IBK/'Y','N','P','D','LN','LG',' ' /
      WRITE(6,999)
999 FORMAT(X,'THIS IS A GENERAL PURPOSE PROGRAM FOR XYPLOT' /
1 10X,'WRITTEN BY SANG H. LEE.')
      WRITE(6,998)
998 FORMAT(/,X,'TYPE IN BAUD RATE')
      READ(5,1009)BRATE
      IBRATE=BRATE/10
      IF(IBRATE.EQ.0)IBRATE=960
      CALL INITT(IBRATE)
      CALL TERM(2,1024)
C
400 CALL ERASE
      CALL CHRSTZ(3)
      CALL ANMODE
C      INPUT TITLE
      TYPE 1000
      ACCEPT 1001,(TITLE(I),I=1,20)
1000 FORMAT(X,'1. TYPE A TITLE(MAX. 40 CHARACTERS).'/
1 5X,'DEFAULT IS TO READ THE FIRST LINE FROM INPUT FILE')
1001 FORMAT(20A4)
C      DATA POINTS(NDP)
      TYPE 1002
      ACCEPT 1009,ANDP
      NDP=ANDP
      IF(NDP.EQ.0)NDP=901
1002 FORMAT(X,'2. NUMBER OF DATA POINTS?'/
1 10X,'DEFAULT WILL INCLUDE ALL THE DATA POINTS IN THE FILE.')
C      X-AXIS
      TYPE 1004
      IXA=2
10 TYPE 1005
      ACCEPT 1006,IANs
      IF(IANS.EQ.IBK)IXA=0
      IF(IANS.EQ.LN)IXA=0
      IF(IANS.EQ.LG)IXA=1
      IF(IXA.NE.2)GO TO 20
      TYPE 1007
      GO TO 10

```



```

20 TYPE 1008
    ACCEPT 1009,XMIN,XMAX
21 TYPE 1010
    ACCEPT 1001,(XLABEL(I),I=1,4)
    IF(XLABEL(1).EQ.'    ')GO TO 21
1003 FORMAT(I1)
1004 FORMAT(X,'3.  X-AXIS DEFINITION.')
1005 FORMAT(5X,'LINEAR OR LOGARITHMIC? TYPE  LN OR LG ')
1006 FORMAT(A2)
1007 FORMAT(X,'ANSWER NOT ACCEPTABLE, TRY AGAIN.')
1008 FORMAT(5X,'WHICH DOMAIN DO YOU WANT TO DISPLAY? TYPE XMIN,XMAX')
1009 FORMAT(5F10.0)
1010 FORMAT(5X,'TYPE A LABEL(MAX. 10 CHARACTERS).')
C   Y-AXIS
    TYPE 1011
    IYA=2
30 TYPE 1005
    ACCEPT 1006,IANs
    IF(IANs.EQ.IBK)IYA=0
    IF(IANs.EQ.LN)IYA=0
    IF(IANs.EQ.LG)IYA=1
    IF(IYA.NE.2)GO TO 40
    TYPE 1007
    GO TO 30
40 TYPE 1012
41 TYPE 1010
    ACCEPT 1001,(YLABEL(I),I=1,3)
    IF(YLABEL(1).EQ.'    ')GO TO 41
1011 FORMAT(X,'4.  Y-AXIS DEFINITION.')
1012 FORMAT(X,'RANGE FOR Y-AXIS WILL BE SET AUTOMATICALLY.')
C   GRID OPTION
50 TYPE 1013
    ACCEPT 1006,IANs
    NG=2
    IF(IANs.EQ.IBK)NG=0
    IF(IANs.EQ.IY)NG=1
    IF(IANs.EQ.IN)NG=0
    IF(NG.NE.2)GO TO 60
    TYPE 1007
    GO TO 50
1013 FORMAT(X,'5.  GRID LINES? Y OR N.')
C   OVERLAY
60 TYPE 1014
    IDASH=0
    ACCEPT 1006,IANs
    IF(IANs.EQ.IBK)IANs=IN
    IF(IANs.EQ.IY)GO TO 100
    IF(IANs.EQ.IN)GO TO 200
    TYPE 1007

```

```

        GO TO 60
1014 FORMAT(X,'6.  OVERLAY TWO CURVES? Y OR N.')
```

C INPUT DATA FORMAT

```

100 IDASH=0
    TYPE 1016
    ACCEPT 1003,IDASH
    IF(IDASH.EQ.0 .OR. IDASH.GT.4)GO TO 100
```

C LEGEND

```

    TYPE 1015
    TYPE 1017
    ACCEPT 1001,(SOLID(I),I=1,4)
    TYPE 1018
    ACCEPT 1001,(DASHED(I),I=1,4)
1015 FORMAT(3X,'1ST CURVE WILL BE A SOLID LINE; 2ND WILL BE DASHED')
1016 FORMAT(3X,'TYPE 1, 2, 3 OR 4 TO SPECIFY INPUT DATA FORMAT: '/
1   10X,'1  IF INPUTS ARE  X,  Y1, Y2'/'
2   10X,'2  IF INPUTS ARE  X1, X2, Y  '/'
3   10X,'3  IF INPUTS ARE  X1, Y1, X2, Y2'/'
4   10X,'4  IF INPUTS ARE  X, DUMMY, DUMMY, Y1, Y2')
1017 FORMAT(5X,'TYPE A LEGEND FOR SOLID LINE.')
```

1018 FORMAT(5X,'TYPE A LEGEND FOR DASHED LINE.')

C DATA INPUT

```

200 INPUT=1
    TYPE 1021
    READ(5,1022)NAME
1021 FORMAT(X,'7.  TYPE THE FILE NAME.',/)
1022 FORMAT(A)
```

C CALL ASSIGN(1,NAME,-1)

```

    OPEN(UNIT=INPUT,FILE=NAME,READONLY,STATUS='OLD',ERR=200)
```

C CALL BELL

```

    IF(TITLE(1).EQ.'    ')READ(INPUT,1024)(TITLE(I),I=1,20)
    GO TO 500
300 READ(INPUT,1024)(TITLE(I),I=1,20)
500 CALL PREP(IXA,IYA,XMIN,XMAX,NG,INPUT,IDASH)
```

C

```

450 TYPE 1020,NAME
    ACCEPT 1006,IAN$
    IF(IAN$.EQ.IY)GO TO 300
    IF(IAN$.EQ.IN)GO TO 700
    TYPE 1007
    GO TO 450
1020 FORMAT(X'MORE PLOTS TO PROCESS IN FILE ',A12,'  Y OR N ?'//)
700 CLOSE(UNIT=INPUT)
```

C

C FOR MORE FILES TO PROCESS.

```

350 TYPE 1023
    ACCEPT 1006,IAN$
    IF(IAN$.EQ.IY)GO TO 400
    IF(IAN$.EQ.IN)STOP
```

```

        TYPE 1007
        GO TO 350
1023 FORMAT(X'MORE PLOTS TO PROCESS? Y OR N'//)
1024 FORMAT(20A4,/)
        END
C
C
        SUBROUTINE PREP(IXA,IYA,XMIN,XMAX,NG,INPUT,IDASH)
        COMMON/TTL/TITLE(20),XLABEL(4),YLABEL(4),SOLID(4),DASHED(4)
        COMMON/DT/NDP,XX1(901),YY1(901),XX2(901),YY2(901)
        DIMENSION GINC(20),TINC(20)
        DATA TINC/1.,1.,2.,2.,4.,5.,5.,5.,5.,5.,5.,5.,5.,5.,5.,
1          10.,10.,10.,10.,10.,10./
        DATA GINC/2.,2.,5.,5.,5.,10.,10.,10.,10.,10.,10.,10.,10.,10.,
1          20.,20.,20.,20.,20.,20./
        DATA IY,IN/'Y','N'/
C      INPUT DATA
        I=0
100 I=I+1
        XX2(I)=0.
        YY2(I)=0.
        IF(IDASH.EQ.0)READ(INPUT,1003,END=10)XX1(I),YY1(I)
        IF(IDASH.EQ.1)READ(INPUT,1003,END=10)XX1(I),YY1(I),YY2(I)
        IF(IDASH.EQ.1)XX2(I)=XX1(I)
        IF(IDASH.EQ.2)READ(INPUT,1003,END=10)XX1(I),XX2(I),YY1(I)
        IF(IDASH.EQ.2)YY2(I)=YY1(I)
        IF(IDASH.EQ.3)READ(INPUT,1003,END=10)XX1(I),YY1(I),XX2(I),YY2(I)
        IF(IDASH.EQ.4)THEN
            READ(INPUT,1003,END=10)XX1(I),DUMMY,DUMMY,YY1(I),YY2(I)
            XX2(I)=XX1(I)
        ENDIF
        IF(XX1(I).EQ.0.)THEN
            IF(XX2(I).EQ.0.)THEN
                IF(YY1(I).EQ.0.)THEN
                    IF(YY2(I).EQ.0.)THEN
                        GO TO 10
                    ENDIF
                ENDIF
            ENDIF
        ENDIF
        GO TO 100
10 CONTINUE
        NDP=I-1
        TYPE 1008,NDP
C      BOUNDARY
        CALL BOUND(XX1,NDP,IMINX,IMAXX,NSCALX)
        IF(XMIN.EQ.XMAX)THEN
            XMIN=IMINX*10.**NSCALX
            XMAX=IMAXX*10.**NSCALX

```

```

ENDIF
  CALL BOUND(YY1,NDP,IMIN1,IMAX1,NSCAL1)
  IF(IDASH.EQ.0)GO TO 50
  CALL BOUND(YY2,NDP,IMIN2,IMAX2,NSCAL2)
  IF(NSCAL2.GT.NSCAL1)GO TO 30
  IF(NSCAL2.LT.NSCAL1)GO TO 40
  IF(IMAX2.GT.IMAX1)IMAX1=IMAX2
  IF(IMIN2.LT.IMIN1)IMIN1=IMIN2
  GO TO 50
30 NSCAL1=NSCAL2
  IMAX1=IMAX2
40 IMIN1=IMIN2
50 YMIN=IMIN1*10.**NSCAL1
  YMAX=IMAX1*10.**NSCAL1
C   TICK MARK
200 ITX=(IMAXX-IMINX)/10+1
  ITY=(IMAX1-IMIN1)/10+1
  TINX=TINC(ITX)*10.**NSCALX
  TINY=TINC(ITY)*10.**NSCAL1
D   TYPE 1011,ITX,ITY,IMAX1,IMIN1,NSCAL1
C   GRID INCREMENTS
  GINX=GINC(ITX)*10.**NSCALX
  GINY=GINC(ITY)*10.**NSCAL1
D   TYPE 1010,TINX,TINY,GINX,GINY
D   TYPE 1011,IXA,IYA,NG,IDASH
  TYPE 1009,XMIN,XMAX
  TYPE 1002,YMIN,YMAX
C   CALL BELL
  TYPE 1004
  ACCEPT 1006,DUMMY
300 CALL GENPLT
  1 (IXA,IYA,XMIN,XMAX,YMIN,YMAX,NG,IDASH,TINX,TINY,GINX,GINY)
350 TYPE 1005
  ACCEPT 1006,IAN
  IF(IANS.EQ.IY)GO TO 400
  IF(IANS.EQ.IN)RETURN
  GO TO 350
400 TYPE 1007
  ACCEPT 1003,XXMIN,XXMAX
  IF(XXMIN.EQ.XXMAX)GO TO 500
  XMIN=XXMIN
  XMAX=XXMAX
500 TYPE 1012
  ACCEPT 1003,YYMIN,YYMAX
  IF(YYMIN.EQ.YYMAX)GO TO 300
  YMIN=YYMIN
  YMAX=YYMAX
  GO TO 300
1001 FORMAT(X,10A4)

```

```

1002 FORMAT(//,X,'YMIN= ',E10.3,5X,' YMAX= ',E10.3)
1003 FORMAT(BN,5F16.0)
1004 FORMAT(//,X,' HIT RETURN TO CONTINUE.')
```

1005 FORMAT(X,'CHANGE DOMAIN? Y OR N'//)

1006 FORMAT(A2)

1007 FORMAT(X,'TYPE XMIN,XMAX')

1008 FORMAT(//,X,'NUMBER OF DATA POINTS=',I5)

1009 FORMAT(//,X,'XMIN= ',E10.3,5X,' XMAX= ',E10.3)

1010 FORMAT(/,X,6E12.3)

1011 FORMAT(//,X,5I10)

1012 FORMAT(X,'TYPE YMIN,YMAX')

END

C

C

SUBROUTINE BOUND(YY,NDP,IMIN,IMAX,NSCALE)

DIMENSION YY(NDP)

NSCALE=0

C YMIN AND YMAX

YMIN=YY(1)

YMAX=YY(1)

DO 100 I=2,NDP

IF(YY(I).LE.YMAX)GO TO 90

YMAX=YY(I)

GO TO 100

90 IF(YY(I).GE.YMIN)GO TO 100

YMIN=YY(I)

100 CONTINUE

C TYPE 999,YMIN,YMAX

C TYPE 999,YY

C 999 FORMAT(/5E15.3)

C WHICH IS DOMINANT? YMAX OR YMIN

NOSWAP=-1

IF(ABS(YMAX).GE.ABS(YMIN))NOSWAP=1

IF(NOSWAP.EQ.1)GO TO 200

C TYPE 1001,NOSWAP

C1001 FORMAT(/I10)

YMAXX=YMAX

YMAX=YMIN

YMIN=YMAXX

C IMAX AND NSCALE

200 IF (ABS(YMAX).GE.100.)GO TO 210

IF(ABS(YMAX).LT.10.)GO TO 220

GO TO 300

210 NSCALE=NSCALE+1

YMAX=YMAX/10.

GO TO 200

220 NSCALE=NSCALE-1

YMAX=YMAX*10.

GO TO 200

```

300 IMAX=YMAX
    IF(IMAX.NE.YMAX)IMAX=IMAX+NOSWAP
C    IMIN
    YMIN=YMIN/10.**NSCALE
    IMIN=YMIN
    IF(IMIN.GT.YMIN)IMIN=IMIN-NOSWAP
    IF(NOSWAP.EQ.1)RETURN
    IMINX=IMIN
    IMIN=IMAX
    IMAX=IMINX
    RETURN
    END
C
C
    SUBROUTINE GENPLT(IXA,IYA,XMIN,XMAX,
1      YMIN,YMAX,NG,IDASH,TINX,TINY,GINX,GINY)
    COMMON/DT/NDP,XX1(901),YY1(901),XX2(901),YY2(901)
    COMMON/TTL/TITLE(20),XLABEL(4),YLABEL(4),SOLID(4),DASHED(4)
    INTEGER IBUF(20)
    CHARACTER ISTR*80
    CALL ERASE
C    FRAME
    CALL SWINDO(120,840,150,600)
    XL=(XMAX-XMIN)*1.001
    YL=(YMAX-YMIN)*1.001
    CALL VWINDO(XMIN,XL,YMIN,YL)
    CALL MOVEA(XMIN,YMIN)
    CALL DRAWA(XMAX,YMIN)
    CALL DRAWA(XMAX,YMAX)
    CALL DRAWA(XMIN,YMAX)
    CALL DRAWA(XMIN,YMIN)
C    TICK MARKS
    IF(XMIN.GT.0.)GO TO 80
    XTK=0.
70 CALL MOVEA(XTK,YMIN)
    CALL DRWREL(0,3)
    CALL MOVEA(XTK,YMAX)
    CALL DRWREL(0,-3)
    XTK=XTK-TINX
    IF(XTK.GT.XMIN)GO TO 70
80 XTK=TINX
90 IF(XTK.GT.XMIN)GO TO 100
    XTK=XTK+TINX
    GO TO 90
100 IF(XTK.GE.XMAX)GO TO 200
    CALL MOVEA(XTK,YMIN)
    CALL DRWREL(0,3)
    CALL MOVEA(XTK,YMAX)
    CALL DRWREL(0,-3)

```

```

      XTK=XTK+TINX
      GO TO 100
200 IF(YMIN.GT.0.)GO TO 209
      YTK=0.
201 CALL MOVEA(XMIN,YTK)
      CALL DRWREL(3,0)
      CALL MOVEA(XMAX,YTK)
      CALL DRWREL(-3,0)
      YTK=YTK-TINY
      IF(YTK.GT.YMIN)GO TO 201
209 YTK=TINY
210 IF(YTK.GT.YMIN)GO TO 250
      YTK=YTK+TINY
      GO TO 210
250 IF(YTK.GE.YMAX)GO TO 300
      CALL MOVEA(XMIN,YTK)
      CALL DRWREL(3,0)
      CALL MOVEA(XMAX,YTK)
      CALL DRWREL(-3,0)
      YTK=YTK+TINY
      GO TO 250
C   GRID LINES
300 IF(NG.EQ.0)GO TO 400
      IF(XMIN.GT.0.)GO TO 309
      XGD=0.
301 CALL MOVEA(XGD,YMIN)
      CALL DRAWA(XGD,YMAX)
      XGD=XGD-GINX
      IF(XGD.GT.XMIN)GO TO 301
309 XGD=GINX
310 IF(XGD.GT.XMIN)GO TO 320
      XGD=XGD+GINX
      GO TO 310
320 IF(XGD.GE.XMAX)GO TO 350
      CALL MOVEA(XGD,YMIN)
      CALL DRAWA(XGD,YMAX)
      XGD=XGD+GINX
      GO TO 320
350 IF(YMIN.GT.0.)GO TO 355
      YGD=0.
351 CALL MOVEA(XMIN,YGD)
      CALL DRAWA(XMAX,YGD)
      YGD=YGD-GINY
      IF(YGD.GT.YMIN)GO TO 351
355 YGD=GINY
360 IF(YGD.GT.YMIN)GO TO 370
      YGD=YGD+GINY
      GO TO 360
370 IF(YGD.GE.YMAX)GO TO 400

```

```

        CALL MOVEA(XMIN,YGD)
        CALL DRAWA(XMAX,YGD)
        YGD=YGD+GINY
        GO TO 370
C      CURVES
400 CALL MOVEA(XX1(1),YY1(1))
      DO 410 I=2,NDP
410 CALL DRAWA(XX1(I),YY1(I))
      IF(IDASH.EQ.0)GO TO 500
      CALL MOVEA(XX2(1),YY2(1))
      DO 420 I=2,NDP
420 CALL DASHA(XX2(I),YY2(I),34)
C      SCALE
500 CALL MOVEA(XMIN,YMIN)
      CALL MOVREL(-20,-20)
      WRITE(ISTR,1001)XMIN
      CALL CHTOIN (ISTR,IBUF,12)
      CALL AOUTST(12,IBUF)
C      TYPE 1001,XMIN
      CALL MOVEA(XMAX,YMIN)
      CALL MOVREL(-50,-20)
      WRITE(ISTR,1001)XMAX
      CALL CHTOIN (ISTR,IBUF,12)
      CALL AOUTST(12,IBUF)
C      TYPE 1001,XMAX
      CALL MOVEA(XMIN,YMIN)
      CALL MOVREL(-110,2)
      WRITE(ISTR,1002)YMIN
      CALL CHTOIN (ISTR,IBUF,12)
      CALL AOUTST(12,IBUF)
C      TYPE 1002,YMIN
      CALL MOVEA(XMIN,YMAX)
      CALL MOVREL(-110,2)
      WRITE(ISTR,1002)YMAX
      CALL CHTOIN (ISTR,IBUF,12)
      CALL AOUTST(12,IBUF)
C      TYPE 1002,YMAX
C      LABELING
      CALL MOVABS(500,100)
      CALL AOUTST(16,XLABEL)
C      TYPE 1003,(XLABEL(I),I=1,4)
      CALL MOVABS(20,460)
      CALL AOUTST(12,YLABEL)
C      TYPE 1003,(YLABEL(I),I=1,3)
C      TITLE
      CALL MOVABS(10,30)
      CALL CHRSTZ(2)
      CALL AOUTST(80,TITLE)
      CALL CHRSTZ(3)

```



```

C      TYPE 1003,(TITLE(I),I=1,10)
C      LEGEND
          IF(IDASH.EQ.0)GO TO 600
          CALL MOVABS(650,700)
          CALL DRWREL(100,0)
          CALL MOVREL(30,0)
          CALL ADUTST(16,SOLID)
C      TYPE 1003,(SOLID(I),I=1,4)
          CALL MOVABS(650,675)
          CALL DSHREL(100,0,34)
          CALL MOVREL(30,0)
          CALL ADUTST(16,DASHED)
600 CONTINUE
C      CALL HDCOPY
          CALL HOME
C      CALL BELL
          CALL ANMODE
          RETURN
1001 FORMAT(1PE8.1,4X)
1002 FORMAT(1PE10.2,4X)
1003 FORMAT(X,10A4)
      END

C
C
      SUBROUTINE CHTOIN (ICH,IBUF,NC)
      INTEGER IBUF(1)
      CHARACTER*80 ICH
C
      DO 100 I=1,NC
          I1 = ICHAR(ICH(I:I))
          CALL CFS (IBUF,I1,I,0)
100 CONTINUE
      RETURN
      END

```

BIBLIOGRAPHY

- [1.1] R. H. MacNeal, Ed., *HANDBOOK FOR LINEAR ANALYSIS*, MSC/NASTRAN Version 64, The MacNeal-Schwendler Corporation, August 1985.
- [1.2] D. N. Herting, R. L. Harder, K. K. Karlsten and R. N. Swami, "Architecture of Nonlinear Analysis," Conference on Finite Element Methods and Technology, Pasadena CA, March 19-20, 1981.
- [1.3] D. N. Herting and R. N. Swami, "Efficient Methods for Static Nonlinear Analysis of Large Order Problems," ASME Symposium on Nonlinear Finite Element Analysis of Shells, Washington D.C., November 15-20, 1981.
- [1.4] D. N. Herting, "Advances in Nonlinear Analysis in MSC/NASTRAN," MSC/NASTRAN User's Conference, Pasadena CA, March 18-19, 1982.
- [1.5] D. N. Herting, "The Reality of Cost-Effective Nonlinear Finite Element Analysis," AUTOFAC 5 Conference Proceedings, CASA Association of SME, Detroit MI, November 14-17, 1983.
- [2.1] R. H. Gallagher, *Finite Element Analysis Fundamentals*, Prentice-Hall, 1975.
- [2.2] O. C. Zienkiewicz, *The Finite Element Method in Engineering Science*, Third Edition, McGraw-Hill, 1977.
- [2.3] B. Irons and S. Ahmad, *Techniques of Finite Elements*, Wiley & Sons, 1980.
- [2.4] K. J. Bathe, *Finite Element Procedures in Engineering Analysis*, Prentice-Hall, 1982.
- [2.5] I. S. Sokolnikoff, *Mathematical Theory of Elasticity*, Second Edition, McGraw-Hill, 1956.
- [2.6] M. F. Rubinstein, *Structural Systems - Statics, Dynamics and Stability*, Prentice-Hall, 1970.
- [3.1] R. H. Pennington, *Introductory Computer Methods and Numerical Analysis*, Second Edition, Macmillan, 1970.
- [3.2] E. K. Blum, *Numerical Analysis and Computation: Theory and Practice*, Addison-Wesley, 1972.
- [3.3] S. H. Lee, "Basic Considerations for Expedient Solution Strategies in Nonlinear Finite Element Analysis," Proceedings, ASME Region IX Computer Applications Symposium, Los Angeles, pp. 4/1-18, October 1988.
- [3.4] D. G. Luenberger, *Linear and Nonlinear Programming*, Second Edition, Addison-Wesley, 1984.

- [3.5] C. S. Davis, "Line Search Methods for Extended Penalty Function Environments," *Int. J. Numerical Methods in Engineering*, Vol.15, 1980.
- [3.6] M. A. Crisfield, "A Faster Modified Newton-Raphson Iteration," *Computer Meth. Applied Mechanics and Engineering*, Vol.20, 1979.
- [3.7] M. A. Crisfield, "Accelerating and Damping the Modified Newton-Raphson Method," *Computers & Structures*, Vol.18, 1984.
- [3.8] H. Matthies and G. Strang, "The Solution of Nonlinear Finite Element Equations," *Int. J. Numerical Methods in Engineering*, Vol.14, 1979.
- [3.9] L. Zhang and D. R. J. Owen, "A Modified Secant Newton Method for Nonlinear Problems," *Computers & Structures*, Vol.15, 1982.
- [3.10] S. H. Lee, "Rudimentary Considerations for Effective Line Search Method in Nonlinear Finite Element Analysis," *Computers & Structures*, Vol.32, No.6, pp. 1287–1301, 1989.
- [3.11] S. H. Lee, "Rudimentary Considerations for Effective Quasi-Newton Updates in Nonlinear Finite Element Analysis," *Computers & Structures*, Vol.33, No.2, pp. 463–476, 1989.
- [3.12] K. J. Bathe and A. P. Cimento, "Some Practical Procedures for the Solution of Nonlinear Finite Element Equations," *Computer Meth. Applied Mechanics and Engineering*, Vol.22, 1980.
- [3.13] P. G. Bergan and R. W. Clough, "Convergence Criteria for Iterative Processes," *AIAA Journal*, Vol.10, No.8, 1972.
- [3.14] S. H. Lee, "Rudimentary Considerations for Effective Convergence Criteria in Nonlinear Finite Analysis," *Proceedings, 1988 ASME Int. Computers in Engineering Conference, San Francisco*, Vol.3, pp. 593–598, August 1988.
- [3.15] E. Ramm, "The Riks/Wempner Approach – An Extension of the Displacement Control Method in Nonlinear Analysis," *Recent Advances in Nonlinear Computational Mechanics*, Pineridge Press, 1982.
- [3.16] M. A. Crisfield, "A Fast Incremental/Iterative Solution Procedure that Handles 'Snap-Through'," *Computers & Structures*, Vol.13, 1981.
- [3.17] G. A. Wempner, "Discrete Approximations Related to Nonlinear Theories of Solids," *Int. J. Solids and Structures*, Vol.7, 1971.
- [3.18] E. Riks, "The Application of Newton's Method to the Problem of Elastic Stability," *J. of Applied Mechanics*, Vol.39, 1972.
- [3.19] R. Kao, "A Comparative Study on the Elastic-Plastic Collapse Strength of Initially Imperfect Deep Spherical Shells," *Int. J. Num. Mech. Eng.*, Vol.17, pp. 427–444, 1981.

- [3.20] S. H. Lee and D. N. Herting, "Comments on 'A Simple Approach to Bifurcation and Limit Point Calculations'," *Int. J. Numerical Methods Eng.*, Vol.21, pp. 1935–1937, 1985.
- [4.1] H. G. Schaeffer, "MSC/NASTRAN Primer, Static and Normal Modes Analysis," Wallace Press, 1979.
- [4.2] M. Okabe, Y. Yamada and I. Nishiguchi, "Basis Transformation of Trial Function Space in Lagrange Interpolation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 20, pp. 85–99, 1980.
- [4.3] T. J. R. Hughes, R. L. Taylor, J. L. Sackman, A. Curnier and W. Kanoknukulchai, "A Finite Element Method for a Class of Contact Problems," *Comp. Meth. in Appl. Mech. Eng.*, Vol. 8, pp. 249–279, 1976.
- [4.4] K. J. Bathe and A. Chaudhary, "A Solution Method for Planar and Axisymmetric Contact Problems," *Int. J. Num. Meth. Eng.*, 21, pp. 65–88, 1985.
- [4.5] A. B. Chaudhary and K. J. Bathe, "A Solution Method for Static and Dynamic Analysis of Three-dimensional Contact Problems with Friction," *Computers & Structures*, Vol. 24, No. 6, pp. 855–873, 1986.
- [4.6] J. C. Simo, P. Wriggers and R. L. Taylor, "A Perturbed Lagrangian Formulation for the Finite Element Solution of Contact Problems," *Comp. Meth. Appl. Mech. Eng.*, 50, pp. 163–180, 1985.
- [4.7] J. W. Ju and R. L. Taylor, "A Perturbed Lagrangian Formulation for the Finite Element Solution of Nonlinear Frictional Contact Problems," *J. Theo. and Appl. Mech.*, 7, 1988.
- [4.8] J. A. Landers and R. L. Taylor, "An Augmented Lagrangian Formulation for the Finite Element Solution of Contact Problems," UCB/SESM-85/09, Dept. of Civil Eng., University of California, Berkeley, 1985.
- [4.9] L. T. Campos, J. T. Oden, and N. Kikuchi, "A Numerical Analysis of a Class of Contact Problems with Friction in Elastostatics," *Comp. Meth. Appl. Mech. Eng.*, 34, pp. 821–845, 1982.
- [4.10] J. H. Cheng and N. Kikuchi, "An analysis of Metal Forming Processes using Large Deformation Elastic-Plastic Formulations," *Comp. Meth. Appl. Mech. Eng.*, Vol. 49, pp. 71–108, 1985.
- [4.11] S. H. Lee, "Some Salient Considerations for Computational Procedure in Elasto-plastic Material Model," Proc. of MSC/NASTRAN Users Conference, Seoul, Korea, October 22–23, 1990.
- [4.12] S. H. Lee and S. S. Hsieh, "Expedient Implicit Integration with Adaptive Time Stepping Algorithm for Nonlinear Transient Analysis," *Comp. Meth. Appl. Mech. Eng.*, Vol. 81, No. 22, pp. 151–172, 1990.

- [4.13] R. F. Steidel, Jr., *An Introduction to Mechanical Vibrations*, John Wiley & Sons, New York, 1971.
- [4.14] S. Timoshenko, D. H. Young and W. Weaver, Jr., *Vibration Problems in Engineering*, Fourth Edition, pp. 198–199, John Wiley & Sons, New York, 1974.
- [4.15] S. Timoshenko and J. N. Goodier, *Theory of Elasticity*, Third Edition, pp. 409–421, McGraw-Hill, New York, 1970.
- [4.16] S. H. Lee, S. S. Hsieh and T. L. Bock, "Adaptive Arc-Length Methods in MSC/NASTRAN," proceedings of 1990 MSC World Users Conference, 5/1-22, Los Angeles, March 28, 1990.
- [5.1] R. Szilard, *Theory and Analysis of Plates*, Prentice-Hall, 1974.
- [5.2] S. H. Lee and A. Raiten, "Characterization of Geometric Nonlinearity with Some Salient Examples," Proceedings of the MSC/NASTRAN User's Conference, 15, Tokyo, Japan, October 17-18, 1991.
- [5.3] O. C. Zienkiewicz, *The Finite Element Method in Engineering Science*, Third Edition, McGraw-Hill, 1977.
- [5.4] H. D. Hibbitt, "Some Follower Forces and Load Stiffness," *Int. J. Num. Met. in Engineering*, Vol.14, pp. 937–941, 1979.
- [5.5] D. N. Herting and G. W. Haggemacher, "Pressure Follower Matrix for Geometric Non-linear Finite Element Analysis," Proceedings, MSC/NASTRAN User's Conference, Los Angeles, 1987.
- [5.6] T. R. Kane, P. W. Likins, and A. Levinson, *Spacecraft Dynamics*, McGraw Hill, 1983.
- [5.7] S. P. Timoshenko and J. M. Gere, *Theory of Elastic Stability*, McGraw-Hill, 1961.
- [5.8] H. D. Hibbit, P. V. Marcal and J. R. Rice, "A Finite Element Formulation for Problems of Large Strain and Large Displacement," *Int. J. Solids Structures*, Vol.6, 1970.
- [5.9] R. H. McMeeking and J. R. Rice, "Finite Element Formulations for Problems of Large Elastic-Plastic Deformation," *Int. J. Solids Structures*, Vol.11, 1975.
- [5.10] J. C. Nagtegaal, "Some Computational Aspects of Elastic-Plastic Large Strain Analysis," *Int. J. Numerical Methods in Engineering*, Vol.17, 1981.
- [5.11] J. H. Argyris and M. Kleiber, "Incremental Formulation in Nonlinear Mechanics and Large Strain Elasto-Plasticity: Natural Approach (Part 1 and 2)," *Computer Methods in Applied Mechanics and Engineering*, Vol.11, 1977 and Vol.14, 1978.
- [5.12] N. T. Tseng and G. C. Lee, "Inelastic Finite Strain Analysis of Structures Subjected to Nonproportional Loading," *Int. J. Numerical Methods in Engineering*, Vol.21, 1985.
- [5.13] L. E. Malvern, *Introduction to the Mechanics of a Continuous Medium*, Prentice-Hall, 1969.

- [5.14] Y. C. Fung, *Foundation of Solid Mechanics*, Prentice-Hall, 1965.
- [5.15] G. E. Mase, *Continuum Mechanics*, McGraw-Hill, 1976.
- [5.16] W. Prager, *Introduction to Mechanics of Continua*, Ginn and Company, 1961.
- [6.1] R. Hill, *The Mathematical Theory of Plasticity*, Oxford at the Clarendon Press, First Edition 1950; Fifth Edition, 1967.
- [6.2] E. G. Thomsen, C. T. Yang and S. Kobayashi, *Mechanics of Plastic Deformation in Metal Processing*, The MacMillan Company, New York.
- [6.3] S. H. Lee, "Some Salient Considerations for Computational Procedure in Elastoplastic Material Model," Proceedings, MSC/NASTRAN User's Conference, Seoul, Korea, October 22-23, 1990.
- [6.4] S. H. Lee, "Self-Adaptive Algorithm for Efficient Computation of Elasto-plastic Material Process," Proceedings, 1991 ASME Int. Computers in Engineering Conference, Santa Clara, Calif., August 18-22, 1991.
- [6.5] O. C. Zienkiewicz, *The Finite Element Method*, Third Edition, McGraw-Hill, 1977.
- [6.6] G. C. Nayak and O. C. Zienkiewicz, "Convenient Form of Stress Invariants for Plasticity," *Journal of the Structural Division*, ASCE, Vol.98, N. ST4, April 1972.
- [6.7] D. R. J. Owen and E. Hinton, *Finite Elements in Plasticity*, Pineridge Press, 1980.
- [6.8] Y. N. Yamada and T. Sakurai, "Plastic Stress Strain Matrix and Its Application for the Solution of Elastic-Plastic Problems by the Finite Element Method," *Int. J. Mech. Sci.*, Vol.10, pp. 343-354, 1968.
- [6.9] H. Armen, "Assumptions, Models and Computational Methods for Plasticity," *Trends in Computerized Structural Analysis and Synthesis*, ed. A. K. Noor and H. G. McComb, Jr., pp. 161-174, 1978.
- [6.10] R. S. Rivlin, "Some Topics in Finite Elasticity," *Structural Mechanics: Proceedings of the First Symposium on Naval Structural Mechanics*, Pergamon Press, New York, pp. 169-198, 1960.
- [6.11] Y. Yamada, H. Takabatake and T. Sato, "Effect of Time-dependent Material Properties on Dynamic Response," *Int. J. Numerical Methods in Engineering*, Vol. 8, pp. 403-414, 1974.
- [6.12] M. P. Badani, "A New Method to Solve Nonlinear Creep Problems in Structural Mechanics Using Incompatible Isoparametric Finite Elements," Ph.D. Dissertation, U. C. Berkeley, 1980.
- [6.13] G. A. Greenbaum and M. F. Rubinstein, "Creep Analysis of Axisymmetric Bodies Using Finite Elements," *Nuclear Engineering and Design*, Vol.7, 1968.

- [6.14] J. A. Clinard et al., "Elevated-Temperature Deformation and Failure Testing and Analysis of Nozzle-to-Spherical Shells: Specimens NS-2 and NS-1," Oak Ridge National Laboratory, ORNL-5939, January, 1983.
- [6.15] A. Levy, "High-Temperature Inelastic Analysis," *Computers and Structures*, Vol.13, 1981.
- [6.16] A. Levy and A. B. Pifko, "On Computational Strategies for Problems Involving Plasticity and Creep," *International Journal for Numerical Methods in Engineering*, Vol.17, 1981.
- [6.17] S. H. Lee, "Application of the Rheological Model to the Creep Analysis Coupled with Plastic Deformation," Transactions, The 8th SMIRT Conference, Brussels, Vol. L, pp. 85-92, August, 1985.
- [6.18] S. H. Lee, "Generalized Viscoelastic Model for Creep Analysis Coupled with Plastic Deformation," *Int. J. Numerical Methods in Engineering*, John Wiley & Sons, Vol.26, pp. 153-165, 1988.
- [6.19] H. Kraus, *Creep Analysis*, John Wiley and Sons, 1980.
- [6.20] N. H. Polakowski, and E. J. Ripling, *Strength and Structure of Engineering Materials*, Prentice-Hall, 1966.
- [6.21] R. C. Juvinall, *Stress, Strain and Strength*, McGraw-Hill, 1967.
- [6.22] A. Levy, "High-Temperature Inelastic Analysis of a Pressure Vessel Nozzle Attachment," Grumman Aerospace Corporation, ORNL/Sub-4485/3, June 1980.
- [6.23] S. H. Lee, "Creep Analysis Capability in MSC/NASTRAN," Proceedings, MSC User's Conference, Pasadena CA, March 1984.
- [7.1] J. M. Chern, "Intermediate Heat Exchanger for Fast Flux Test Facility: Evaluation of Inelastic Computer Programs," FWR-27, Foster Wheeler Corporation, March 1972.
- [7.2] R. Kao, "A Comparative Study on the Elastic-Plastic Collapse Strength of Initially Imperfect Deep Spherical Shells", *International Journal of Numerical Methods in Engineering*, Vol.17, No.3, 1981.
- [7.3] Dinno and Gill, "An Experimental Investigation into the Plastic behavior of Flush Nozzles in Spherical Pressure Vessels," *Int. J. Mesh. Sci.*, Vol.7, p. 817, 1965.
- [8.1] D. N. Herting, "A General Purpose Nonlinear Transient Integration System", *Innovative Methods for Nonlinear Problems*, edited by W. K. Liu, T. Belytschko and K. C. Park, Pineridge Press, 1984.
- [8.2] S. H. Lee and S. S. Hsieh, "Expedient Implicit Integration with Adaptive Time Stepping Algorithm for Nonlinear Transient Analysis", *Int. J. Computer Methods in Applied Mechanics and Engineering*, Vol.81, No. 22, pp. 151-172, 1990.

- [8.3] N. M. Newmark, "A Method of Computation for Structural Dynamics," *J. of Engineering Mechanics Division of ASCE*, July, 1959.
- [8.4] P. G. Bergan and E. Mollestad, "An Automatic Time Stepping Algorithm for Dynamic Problems," *Computer Methods in Applied Mechanics and Engineering*, Vol.49, pp. 299–318, 1985.
- [8.5] S. H. Lee, "Rudimentary Considerations for Effective Quasi-Newton Updates in Nonlinear Finite Element Analysis," *Computers & Structures*, Vol.33, No.2, pp. 463–476, Pergamon Press, 1989.
- [8.6] S. H. Lee, "Rudimentary Considerations for Effective Line Search in Nonlinear Finite Element Analysis," *Computers & Structures*, Vol.32, No.6, pp. 1287–1301, 1989.
- [8.7] S. H. Lee, T. L. Bock and S. S. Hsieh, "Adaptive Time Stepping Algorithm for Nonlinear Transient Analysis," *Proceedings, 1988 MSC World Users Conference, Los Angeles*, Vol.2 (54), March 1988.
- [8.8] J. M. Biggs, *Structural Dynamics*, pp. 69–76, McGraw-Hill, 1964.
- [8.9] H. A. Balmer and E. A. Witmer, "Theoretical-Experimental Correlation of Large Dynamic and Permanent Deformation of Impulsively Loaded Simple Structures," Air Force Flight Dynamic Laboratory, Report FDP-TDR-64-108, Wright-Patterson AFB, Ohio, July, 1964.
- [8.10] T. B. Belytschko, J. I. Lin and C. S. Tsay, "Explicit Algorithms for the Nonlinear Dynamics of Shells," *Computer Methods in Applied Mechanics and Engineering*, Vol.42, pp. 225–251, 1984.
- [8.11] D. P. Mondkar and G. H. Powell, "Finite Element Analysis of Nonlinear Static and Dynamic Response," *Int. J. Numerical Methods in Engineering*, Vol.11, pp. 499–520, 1977.
- [10.1] S. H. Lee and D. N. Herting, "Comment on 'A Simple Approach to Bifurcation and Limit Point Calculations,'" *Int. J. Numerical Methods Eng.*, Vol.21, pp. 1935–1937, 1985.
- [10.2] M. Fujikake, "A Simple Approach to Bifurcation and Limit Point Calculations," *Int. J. Numerical Methods Eng.*, Vol.21, pp. 185–191, 1985.
- [10.3] R. Kao, "A Comparative Study on the Elastic-Plastic Collapse Strength of Initially Imperfect Deep Spherical Shells," *Int. J. Numerical Methods Eng.*, Vol.17, pp. 427–444, 1981.
- [10.4] W. P. Schulze Schwering, "The Cologne Challenge," *Finite Element News*, No.4, 1984.
- [11.1] S. H. Lee, "Rudimentary Considerations for Effective Quasi-Newton Updates in Nonlinear Finite Element Analysis," *Computer & Structures*, Vol.33, No.2, pp. 463–476.

- [11.2] S. H. Lee, "Rudimentary Considerations for Effective Line Search Method in Nonlinear Finite Element Analysis," *Computers & Structures*, Vol.32, No.6, pp. 1287-1301, 1989.
- [11.3] S. H. Lee, "Basic Considerations for Expedient Solution Strategies in Nonlinear Finite Element Analysis," Proceedings, the ASME Region IX Computer Applications Symposium, Los Angeles, pp. 4/1-18, 1988.
- [11.4] W. P. Schulze Schwering, "The Cologne Challenge," *Finite Element News*, No.4, 1984.
- [11.5] S. H. Lee, "The Cologne Challenge—The Replies (Part IV)," *Finite Element News*, No.5, pp. 24-25, 1986.
- [11.6] R. Kao, "A Comparative Study on the Elastic-Plastic Collapse Strength of Initially Imperfect Deep Spherical Shells," *Int. J. Numerical Methods Eng.*, Vol.17, pp. 427-444, 1981.
- [11.7] S. H. Lee and D. N. Herting, "Comments on 'A Simple Approach to Bifurcation and Limit Point Calculations'," *Int. J. Numerical Methods Eng.*, Vol.21, pp. 1935-1937, 1985.
- [11.8] S. H. Lee, "Rudimentary Considerations for Effective Convergence Criteria in Nonlinear Finite Element Analysis," Proceedings, 1988 ASME Int. Computers in Engineering Conference, San Francisco, Vol.3, pp. 593-598, 1988.
- [11.9] G. A. Greenbaum and M. F. Rubinstein, "Creep Analysis of Axisymmetric Bodies Using Finite Elements," *Nuclear Engineering and Design*, Vol.7, 1968.
- [11.10] L. Morino, J. W. Leech, and E. A. Witmer, "An Improved Numerical Calculation Technique for Large Elastic-Plastic Transient Deformations of Thin Shells: Part 2 – Evaluation and Applications," *ASME Journal of Applied Mechanics*, Vol.38, pp. 429-435, 1971.
- [11.11] C. Tsay and R. Keene, "MSC/DYNA Verification and Demonstration Problem Manual," 1989.
- [11.12] J. M. Kennedy, T. Belytschko, and J. I. Lin, "Recent Developments in Explicit Finite Element Techniques and Their Application to Reactor Structures," *Nuclear Engineering and Design*, Vol.97, pp. 1-24, 1986.
- [11.13] T. Belytschko, J. I. Lin, and C. S. Tsay, "Explicit Algorithms for the Nonlinear Dynamics of Shells," *Computer Methods in Applied Mechanics and Engineering*, Vol.42, pp. 225-251, 1984.
- [11.14] Roark and Young, *Formulas for Stress and Strain*, McGraw-Hill, Fifth Edition, p. 220.
- [11.15] S. Timoshenko and J. N. Goodier, *Theory of Elasticity*, pp. 497-504, 2nd ed., McGraw-Hill, New York, 1951.

- [11.16] R. H. MacNeal and R. L. Harder, "A Refined Four-Noded Membrane Element with Rotational Degrees of Freedom," *Computers & Structures*, Vol. 28, No. 1, pp. 75–84, 1988.
- [11.17] S. H. Lee, "Application of Rheological Model to Creep Analysis Coupled with Plastic Deformation," Transactions, The 8th SMiRT Conference, Brussels, Vol. L, pp. 85–92, August, 1985.
- [11.18] S. H. Lee, "Generalized Viscoelastic Model for Creep Analysis Coupled with Plastic Deformation," *Int. J. Numerical Methods in Engineering*, Vol. 26, pp. 153–165, 1988.

Appendix A

Nonlinear Bulk Data Entries

For information on the following Nonlinear Bulk Data Entries, see the Quick Reference Guide.:

- CGAP
- CREEP
- MATS1
- NLPARM
- NLPCI
- PBCOMP
- PGAP
- TABLEST
- TABLES1
- TSTEPNL

Index

- absolute convergence, 3.6-5, 9.1-8
- absolute divergence, 3.3-3
- adaptive
 - arc-length method, 3.7-4
 - bisection, 3.7-6, 8.4-4
 - GAP, 4.3-6
 - Newmark's method, 8.3-2
 - penalty value adjustment, 4.3-11,12
 - solution algorithm, 3.1-1
 - time stepping, 8.4-1
- Almansi finite strain, 5.6-2
- ALRATIO, 3.7-5, 7.1-6
- angular velocity tensor, 5.6-5
- anisotropic friction, 4.3-13
- anisotropy, 6.6-1
- apparent Poisson's ratio, 6.2-27
- arc-length method, 3.7-1, 7.1-6
- associated flow rule, 6.2-12,13,14
- automatic time stepping, 8.1-1, 8.4-1
- AUTOSPCR, 1.4-8
- auxiliary angle, 5.4-2

- basic coordinate system, 2.3-1, 5.2-2
- Bauschinger Effect, 1.5-2, 6.2-2
- BEAM, 4.1-3,6
- BEND, 4.1-4
- BETA, 1.4-8
- BFGS update, 3.5-1,2, 11.2-1
 - criteria, 3.5-3
 - for arc-length method, 3.5-11
- biaxial loading, 7.4-2
- bilateral stress-strain relation, 6.3-4
- bisection algorithm, 8.4-4, 8.5-5
- boundary conditions, 2.1-2, 7.5-1
- BUCKLE, 1.4-8, 10.1-3

- buckling, 10.1-1
- bulk data, 1.4-6

- case control, 1.4-4, 7.1-1, 9.1-1
- Cauchy's deformation tensor, 5.6-2
- Cauchy stress tensor, 5.6-3
- CDAMPi, 9.3-6
- combined hardening, 6.2-10
- constitutive relations, 6.1-1, 5.6-10
- constraint, 7.5-1
- contact analysis, 4.3-1, 11.5-1
- continuum mechanics, 5.6-1
- contraction factor, 3.6-4
- convective coordinates, 5.2-1
- convective rate of change, 5.6-5
- convergence, 3.6-1
 - absolute convergence, 3.6-5, 9.1-8
 - criteria, 3.6-1
 - information output, 7.1-7, 9.1-7
 - rate, 3.3-2
 - tolerance, 3.6-5
- coordinate system, 2.3-1
 - convective, 5.2-1
 - element, 5.2-2
 - isoparametric, 4.2-1
 - material, 4.1-4
- coordinate transformations, 2.3-1, 5.2-2
- corotational stress rate tensor, 5.6-5
- Coulomb friction law, 4.3-7
- COUPMASS, 9.3-3
- CREEP, 4.1-2, 6.1-5
- creep
 - analysis, 6.4-1,4, 11.3-1
 - strain, 11.8-1
 - strain rate, 6.4-2,4, 11.8-3

Crisfield's method, 3.7-3, 7.1-6
 criteria
 BFGS update, 3.5-3
 convergence, 3.6-1
 divergence, 3.3-2, 3.6-6, 8.4-4
 line search, 3.4-3
 time expiration, 3.3-3
 yield, 6.2-3
 critical buckling load, 10.1-2
 CSHEAR, 4.1-4
 CVISC, 9.3-6
 cyclic loading, 1.5-2, 6.2-29

damping
 matrix, 9.3-7
 modeling, 9.3-4
 numerical, 1.4-9, 8.3-6, 9.1-2
 DBDRNL, 1.4-8
 deformation gradient, 5.6-1,3
 DESITER, 7.1-6
 DFP method, 3.5-2
 diagnostic outputs (DIAG 50,51),
 SOL 66 and 106, 7.2-4
 SOL 99 and 129, 9.5-1
 differential stiffness, 3.2-1, 5.1-5
 direct time integration, 8.1-1
 discrete damping elements, 9.3-4
 discretization, 2.1-2
 displaced element coordinate system, 2.3-1,
 5.2-3
 displacement sets, 2.4-1
 divergence criteria, 3.3-2, 3.6-6, 8.4-4
 DLMAG, 7.1-7, 9.1-8
 DMAP, 1.3-1
 doubling scheme, 3.4-4,13
 Druker-Prager yield criterion, 6.2-5
 DT, 7.1-3, 9.1-4
 dynamic loads, 9.4-1

effective plastic strain increment, 6.2-14
 effective strain for nonlinear elastic, 6.3-1
 E-FIRST, 7.1-7, 9.1-8

E-FINAL, 7.1-7, 9.1-8
 EIGB, 2.5-1
 elasto-plastic-creep relations, 6.4-6
 elasto-plastic material, 6.2-19
 element
 coordinate system, 2.3-1, 4.2-8, 5.2-2
 matrix, 4.2-5,6, 5.1-2
 nonlinear analysis, 4.1-2
 stiffness, 2.2-3, 4.2-6
 enforced motion, 7.5-1, 9.4-5
 ENIC, 7.1-7
 EPI, 7.1-7
 equilibrium, 2.2-3, 3.2-1, 8.2-1
 error functions, 3.6-3
 error vector, 3.2-1, 8.2-1
 EUI, 7.1-7
 Eulerian strain, 5.6-2,3,11
 Eulerian formulation, 5.6-1
 EWI, 7.1-7
 Executive System, 1.3-1
 EXP, 6.1-6

FACTOR, 7.1-7, 9.1-8
 FBS, 3.5-5
 FIAT (File Allocation Table), 1.3-1
 finite element method, 2.2-1, 4.1-1
 finite strain, 5.6-2
 follower force, 5.1-5
 follower matrix, 3.7-1, 5.3-2
 frame-indifference, 5.6-5
 friction element, 4.3-3
 frictional yield function, 4.3-7
 FSTRESS, 7.1-5, 9.1-5
 functional module, 1.3-1

G, 9.3-7
 GAP element, 4.3-1, 11.5-1
 Gaussian elimination method, 3.2-1
 generalized effective plastic strain, 6.2-14
 generalized logarithmic strain, 5.6-10
 geometric nonlinearity, 1.5-1, 2.1-2, 5.1-1
 geometric stiffness, 5.1-5

- gimbal angles, 5.4-1
- gimbal lock, 5.4-2
- GINO, 1.3-1
- global coordinate system, 2.3-1
- Green's deformation tensor, 5.6-2
- Green's finite strain, 5.6-2

- Herzian problem, 4.3-25
- Hessian matrix, 3.5-1
- hierarchy of nonlinear looping, 1.3-2
- hyperelastic material, 5.6-8,11
- hysteresis, 6.2-29

- IFP (Input File Processor), 1.3-1
- Illinois algorithm, 3.4-4
- impact analysis, 11.6-1
- implicit integration, 8.1-1, 9.1-2
- increment, 2.5-1,2
- initial anisotropy, 6.6-1
- input data structure, 1.4-1
- integration
 - direct time, 8.1-1
 - Gauss, 4.2-6
 - implicit, 8.1-1, 9.1-2
 - Newmark, 8.3-1
 - three-point method, 8.2-1
 - two-point method, 8.3-1
- INTOUT, 3.7-6, 7.1-4
- isoparametric elements, 4.2-1
- isotropic hardening, 6.2-9
- iteration, 2.5-2, 8.2-2,3, 8.5-1

- Jaumann stress, 5.6-5,6,10

- K6ROT, 1.4-8, 11.7-1
- Kelvin-Maxwell model, 6.4-3, 11.8-1
- kinematic hardening, 6.2-9
- Kirchhoff & Love, plate theory, 5.1-1
- Kirchhoff stress, 5.6-3,10
- KMETHOD, 3.3-1,2, 11.1-12
- KSTEP, 3.3-1, 7.1-3, 9.1-4

- Lagrange multiplier method, 4.3-1
- Lagrangian description, 5.6-3
- Lagrangian formulation, 5.6-1
- Lagrangian strain, 5.6-2,3
- Lagrangian strain rate, 5.6-10
- LAMBDA, 7.1-7, 9.1-8
- LANGLE, 1.4-9, 5.4-1,3
- large rotation, 5.1-2, 5.4-1, 5.5-1
- left stretch tensor, 5.6-3
- LGDISP, 1.4-9, 4.1-2, 5.1-2
- LIMIT1, LIMIT2, 6.1-2
- line elements, 4.1-2
- line search, 3.4-1, 11.2-1
 - algorithm, 3.4-10
 - criteria, 3.4-3
 - iteration, 3.4-11
- linear interpolation, 3.4-4,13
- load,
 - dynamic, 9.4-1
 - static, 7.3-1
 - superelement, 7.3-3
- LOADINC, 1.4-9
- loading and unloading, 6.2-1
- local coordinate system, 2.3-1
- logarithmic strain, 5.6-4,10
- LOOPID, 1.4-9
- LSEQ, 10.3-1
- LSTOL, 3.4-4, 7.1-5

- mass
 - matrix, 9.3-2
 - modeling, 9.3-1
 - properties, 9.3-1
- MAT2, 6.6-1
- MAT9, 6.6-1
- material coordinate system, 2.3-1, 4.1-4
- material derivative, 5.6-4,10
- material nonlinearity, 1.5-1, 2.1-2, 6.1-1
- material properties entries, 6.6-1
- material stiffness, 2.2-2
- MATS1, 6.1-2
- MAXALR, 3.7-5, 7.1-6

MAXBIS, 7.1-5, 9.1-5
 MAXDIV, 3.3-2, 7.1-5, 9.1-5, 11.1-12
 MAXITER, 7.1-4, 9.1-4
 MAXLP, 1.4-9
 MAXLS, 3.4-4, 7.1-5, 9.1-5
 MAXQN, 3.5-5, 7.1-5, 9.1-5
 MAXR, 7.1-5,
 Maxwell-Kelvin model, 6.4-3, 11.8-1
 method of false position, 3.4-2,5
 MINALR, 3.7-5, 7.1-6
 modal coordinate system, 2.3-1
 modeling, 1.5-1
 Mohr-Coulomb yield criterion, 6.2-5
 MPC, 2.4-1,3
 MPL (Module Properties List), 1.3-1
 MXINC, 7.1-7

 natural strain, 5.6-4
 NDAMP, 1.4-9, 8.3-6, 9.1-2
 NDIV, 3.3-3, 8.4-4
 neutral loading, 7.4-2
 NEWK, 3.3-1
 Newmark integration, 8.3-1
 Newton-Raphson method, 2.5-2,4, 3.2-1
 NINC, 3.7-4, 7.1-3
 NLAYERS, 1.4-9
 NLITER, 7.2-1
 input data block, 7.2-1,2
 output data block, 7.2-2,3
 parameters, 7.2-3,4
 NLPARM, 2.5-1, 7.1-2
 NLPCI, 7.1-6
 NLTRD module, 8.2-1,4, 9.5-1
 NLTRD2 module, 8.3-1,4, 8.5-6, 9.5-2,3
 NMATD, 6.3-2
 NMEP, 6.2-24
 NMEVD, 6.3-2
 NMLOOP, 1.4-10, 10.2-1
 nodal forces, 2.2-2, 4.2-7
 NOGO flag, 3.3-3
 NOLIN, 9.4-1
 non-adaptive GAP element, 4.3-13
 nonlinear capabilities, 1.2-1

nonlinear characteristics, 1.5-1
 nonlinear elasticity, 6.3-1
 nonlinear element, 1.2-1, 4.1-2
 properties, 1.2-3
 nonlinear modal analysis, 10.2-1
 nonlinear static analysis, 7.1-1, 11.1-1
 nonlinear transient response analysis, 9.1-1
 nonproportional loading, 7.4-1
 normality principle, 6.2-13
 N-QNV, 7.1-7, 9.1-8
 numerical damping, 1.4-9, 8.3-6, 9.1-2

 one-step integration, 8.3-1
 open/close boundary, 4.3-9
 OSCAR, 1.3-1
 output
 diagnostics, 7.2-4, 9.5-1
 interpretation for large rotation, 5.4-6
 iteration process, 7.1-7, 9.1-7
 line search, 3.4-8

 parameters, 1.4-8
 parametric studies
 automatic time stepping, 8.6-1
 BFGS updates, 11.2-1
 K6ROT, 11.7-2
 line search, 11.2-1
 penalty value in GAP, 4.3-17
 static solution method, 11.1-14
 penalty GAP element, 4.3-2,3
 penalty values
 adaptive adjustment, 4.3-11
 as input parameters, 4.3-6
 selection of Gap stiffness, 11.5-3
 Piola-Kirchhoff stress, 5.6-3
 PGAP, 4.3-5
 plane strain, 6.2-17,18
 plane strain element, 4.1-5
 plane stress, 6.2-18
 plasticity, 6.2-1
 plate & shell elements, 4.1-4
 plotting output codes, 1.4-17

- polar decomposition, 5.6-2
- post-buckling analysis, 3.7-1,12
- Prandtl-Reuss stress-strain relation, 6.2-12
- Preface Operation, 1.3-1
- primary creep, 6.4-1
- probable divergence, 3.3-3
- program architecture, 1.3-1
- proportional loading, 6.2-1

- QUAD4, 7.8-1, 11.7-2
- quadratic convergence, 3.2-2
- quadratic interpolation, 3.4-13
- quasi-Newton (QN) update, 3.5-1, 11.2-1

- rate of deformation tensor, 5.6-4
- reduction of system equations, 2.4-1
- restarts
 - in SOL 66 and 106, 7.6-1
 - in SOL 99 and 129, 9.2-1
- RF63D89, 10.2-7
- rheological model, 6.1-8, 6.4-1, 11.8-1
- right stretch tensor, 5.6-3
- Rik's method, 3.7-2, 7.1-6
- Ritz method, 2.2-1
- ROD element, 4.1-2,3
- rotation vector, 5.4-3
- RT, 6.1-6
- RTOLB, 5.4-6, 7.1-6, 9.1-7

- SCALE, 7.1-7
- SDATA, 1.4-10, 9.2-2
- secondary creep, 6.4-1
- shape function, 4.2-1,2,3
- shell element, 4.1-4
- slip yield surface, 4.3-8
- SLOOPID, 1.4-10, 9.2-1
- SMALLDB, 1.4-10
- snap-through, 3.7-1,9, 11.1-1
- SOL 64, 1.1-1, 5.5-2
- SOL 66 and 106, 1.4-1, 1.5-1, 7.2-6
 - examples, 11.1-1, 11.3-1, 11.5-1
 - user interface, 7.1-1
- SOL 99 and 129, 1.4-1, 1.5-1, 8.1-3
 - capabilities, 1.2-2
 - examples, 11.4-1, 11.6-1
 - static analysis, 10.3-1
 - user interface, 9.1-1
- solid elements, 4.1-4
- solution method, 2.5-1, 3.1-1
- spatial derivative, 5.6-4
- SPC, 2.4-2,4
- spin tensor, 5.6-4
- SSG3, 1.4-11
- stability analysis, 8.2-3,4
- static loads, 7.3-1
- stress stiffness, 5.1-4
- stretching tensor, 5.6-4
- stick-slip behavior, 4.3-8
- stick-slip boundary, 4.3-9
- stiffness
 - differential, 3.2-1, 5.1-5
 - element, 2.2-3, 4.2-6
 - geometric, 5.1-5
 - material, 2.2-2, 6.5-1
 - stress, 5.1-4
 - updates, 3.2-1, 3.3-1
- STIME, 1.4-11, 9.2-1
- strain hardening, 6.2-1,9
- strain-rate increment,
- stress data recovery, 4.2-9
- structural damping, 9.3-4,5
- SUBID, 1.4-11
- subincremental algorithm, 4.3-9
- superelement, 7.3-3
- surface elements, 4.1-4
- SYSTEM(141), 3.4-13
- SYSTEM(145), 3.5-12

- TABLEST, 6.5-2
- TABS, 1.4-11
- tangent material stiffness, 6.5-1
- tangential stiffness, 3.2-2
- TEMP, 6.5-1,2,3
- tertiary creep, 6.4-1

- TESTNEG, 1.4-12, 3.3-1
- thermal strain, 6.5-1
- thermal stress analyses, 6.5-3
 - required input, 6.5-3, 7.3-4
- thermo-elasticity, 6.5-1
- three-point method, 8.2-1
- THRESH, 6.1-6
- time expiration criteria, 3.3-3
- total Lagrangian, 5.6-6,7
- transverse shear stiffness, 4.3-6
- Tresca yield criterion, 6.2-5
- TRIA3, 11.7-2,8
- TSTATIC, 1.4-12, 10.3-1
- TSTEPNL, 8.2-1, 8.3-1, 9.1-1,2, 10.3-1
- two-point method, 8.3-1

- uniaxial stress-strain, 6.3-3
- uniaxial stress, 6.2-18,19
- UPARTN, 2.4-2
- updated Lagrangian, 2.2-2, 5.6-1,6,7
- user interface, 1.4-1
 - arc-length, 7.1-6
 - buckling analysis, 10.1-2
 - creep analysis, 6.1-5
 - GAP element, 4.3-4
 - geometric nonlinearity, 5.1-1
 - loads, 7.3-1, 9.3-1
 - material nonlinearity, 6.1-1
 - nonlinear modal analysis, 10.2-1
 - static analysis, 7.1-1
 - thermo-elasticity, 6.5-1
 - transient analysis, 9.1-1

- velocity gradient, 5.6-4
- verification and validation problems
 - adaptive time stepping, 8.6-1
 - arc-length method, 3.7-9
 - buckling analysis, 10.1-3
 - follower forces, 5.3-7
 - creep, 6.4-9, 6.4-13
 - enforced displacement, 7.5-2
 - friction, 4.3-19

- GAP, 4.3-14
- initial anisotropy, 6.6-1
- large rotation, 5.5-1
- nonlinear buckling analysis, 10.1-7
- nonlinear modal analysis, 10.2-1
- nonproportional loading, 7.4-1
- plasticity, 6.2-27
- restart in SOL 66, 7.6-2
- restart in SOL 99, 9.2-3
- static analysis in SOL 66, 7.7-1
- static analysis in SOL 99, 10.3-2
- surface contact, 4.3-25, 4.3-29
- thermo-elasticity, 6.5-3
- transient response analysis, 8.6-1
- viscoelastic, 6.4-1, 11.8-1
- viscous damping, 9.3-4,5
- volume integration, 4.2-6,7
- von Mises yield criterion, 6.2-3
- vorticity tensor, 5.6-4

- W3, 1.4-12
- W4, 1.4-12
- weighted normalization, 3.6-3,4

- yield criteria, 6.2-1,3
- yield function, 6.2-6
- yield function derivatives, 6.2-15
- yield stress, 6.2-1
- yield surface, 6.2-1