APPLICATIONS OF FINITE ELEMENT TECHNIQUES IN THE SIMULATIONS OF STRUCTURE INTERACTIONS DURING WELDING

by

Haoshi Song

B.S. IN APPLIED MECHANICS
ZHEJIANG UNIVERSITY, CHINA
July, 1982

M.S. IN MECHANICAL ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
September, 1988

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY
IN MECHANICAL ENGINEERING

at

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

December, 1990

Signature of Author: _____
Department of Mechanical Engineering
December, 1990

Certified by: _____
Stuart B. Brown, thesis supervisor
Professor, Department of Materials science & Engineering

Accepted by: _____
A. Sonin, Chairman
Department of Mechanical Engineering Graduate Committee

Applications of Finite Element Techniques in the Simulations of Structure Interactions During Welding

by

Haoshi Song

Submitted to the Department of Mechanical Engineering on December 19, 1990
in partial fulfillment of the requirements for the Degree of
Doctor of Philosophy in Mechanical Engineering

ABSTRACT

Distortion and residual stresses resulting from welding represent significant problems in the accurate fabrication of large structures. Although the capability to predict these phenomena would provide substantial assistance to the design and fabrication of welded structures, both welding distortion and induced stresses are difficult to simulate accurately given the intensive computational demands of such a severely nonlinear process.

This thesis first examines structure/weld interaction issues during welding processes through two- and three-dimensional simulations of a ring-stiffened structure. The analyses indicate that the effect of overall structure in a welded structure plays a more important role than has been appreciated. The analyses also indicate that variations in the original weld root opening can have a significant effect on residual stresses and distortion. Variations in fixturing and tack weld placement can have similarly significant influence on both the overall structure and the weld zone. Experimental results on a ring-stiffened cylinder support the above conclusions. Welding simulation therefore requires a full consideration of the overall structure and the fixture setups. Three-dimensional analysis is required to perform such simulations. The high demands on computational resources of three-dimensional finite element analysis prevent successful simulations and require reductions of the size of a model.

The second half of this thesis develops dynamic remeshing and substructuring techniques which can substantially reduce the size of a three-dimensional finite element model for a welding process. The welding simulation is broken into many incremental analyses. A local nonlinear zone around the heat source, modeled by a denser mesh, moves from one incremental analysis to another. Models for each incremental analysis is remeshed at the beginning of that increment, and variables are mapped between two consecutive incremental analyses. The dynamic remeshing and substructuring techniques are then implemented on a plate welding simulation and are proven to be effective and successful.

These techniques not only can reduce the size of a welding model and thus reduce the computer time and storage space, but also can provide a step-wise coupling between the heat transfer and deformation analyses; this increases the accuracy of the simulation, particularly in the case of large deformations. After further modification, the dynamic remeshing and substructuring techniques can be used in other welding simulations.

Thesis Supervisor: Dr. Stuart B. Brown
Title: Richard P. Simmons Assistant Professor
of Materials Manufacturing

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank many people who have given me advice, help and support during my stay at MIT.

First, I would like to thank my thesis supervisor, Prof. Stuart B. Brown, for his constant guidance, encouragement and support. His influence extends far beyond my academic work. His advice will guide me as I pursue excellence in every aspect of my life. I am also grateful to the other members of my thesis committee, Prof. Thomas Eagar and Prof. David Hardt, for their advice, suggestions and guidance during my thesis work.

I would like to thank Mr. John Freeman for his help on the welding experiments. I would also like to thank Mr. Jeffrey Bellsey, and Miss Katherine Konner for their careful proof-reading of this manuscript. The members in my research group, Mr. Georges Abou-Chedid, Mr. Vivek R. Dave, Mr. Pratyush Kumar and Mr. John J. Wlassich deserve special thanks, as do Miss Sandra DeJong and Miss Hilary Sheldon. My memory of MIT will always be pleasant because of their friendship.

I want to thank my wife, Yan, for her love, support and patience. There were many evenings and weekends I had to work away from home, and I thank her for understanding.

Finally, I appreciate the financial support from the Office of Naval Research. This thesis work was sponsored under contract number N00014-89-J-1187. I would also like to acknowledge the Material Processing Center at MIT that provided the APOLLO DN10000 computer for part of the finite element analyses in this thesis.

Nomenclature

α	Coefficient of thermal expansion.
C	Correction for unbalanced force around the local zone.
c	Specific heat.
χ	Constant, $1 + \frac{\sqrt{3}}{3}$.
ϵ	Strain, heat emissivity.
$\dot{\epsilon}_{kk}$	Dilatational strain rate.
$\{F\}$	Nodal force vector.
γ	Overlapping factor of moving local zone.
h	Coefficient of boundary heat convection
$[K]$	Stiffness matrix.
k	Thermal conductivity.
l_e	Length of one side of a typical element.
M	Modulus of strain hardening.
N_i	Shape function.
ϕ	Constant, $1 - \frac{\sqrt{3}}{3}$.
ψ	Constant, $\frac{\sqrt{3}}{2}$.
q	Heat input or generation per unit volume.
$\{R\}$	Nodal reaction force vector.
R	Residual forces along the boundary of local zone.
ρ	Density.
σ	Stress, Stefan-Boltzman constant.
T	Temperature.
t	Time.
$\{U\}$	Nodal displacement vector.
v	Velocity of traveling heat source.
v_e	Propagation velocity of dilatational waves.

Contents

1 Introduction 1

1.1 Numerical Simulations of Welding Processes 3

1.2 Literature Review 4

1.3 Elastic Coupling in Welding of Large Structures 6

2 Fundamentals of Welding Analysis 10

2.1 Basic Equations 10

2.2 Boundary and Initial Conditions 13

2.3 Material Properties 15

2.4 FEM Formulations 17

3 2-D Simulations of Welding of a Circular Ring-Stiffened Cylinder 20

3.1 The 2-D FEM Model 20

3.2 Heat Transfer Analysis 24

3.3 Stress/strain Analysis 25

3.4 Fitup Effect 29

3.5 Fixturing Effects 34

3.6 Correlation with Welded Structure 39

3.7 Elastic Coupling 41

3.8 The Applicability of 2-D Models 45

4 3-D Simulation of a Short Weld on the Ring Stiffened Cylinder 48

4.1 The 3-D Model 48

4.2 Heat Transfer Analysis 50

4.3 Deformation Analyses 52

4.4 Fixturing Effects 56

4.5 Experimental Correlations 64

4.5.1 Measurement of Distortions 67
4.5.2 Correction of Gravity Effect 68
4.5.3 Comparison of Predicted and Measured Distortions 70
4.6 Deformation Mechanism During Welding Processes 72
4.7 Discussion on 3-D Simulations 75

5 Substructuring and Dynamic Remeshing **78**
5.1 Substructuring Technique 78
5.2 Reduction of Computational Effort by Substructuring 81
5.2.1 Substructuring Based on the Same Mesh 83
5.2.2 Substructuring Based on Different Meshes 84
5.3 Dynamic Remeshing 85
5.3.1 Characteristics of Dynamic Remeshing 87
5.3.2 Variable Extrapolation 89
5.3.3 Locating a Spatial Point in a Mesh 91
5.3.4 Normalized Coordinates 95
5.4 The Shell Program 101

6 Dynamic Remeshing Applied to Plate Welding **104**
6.1 The FEM Model 104
6.2 Remeshing Frequency and Size of the Local Zone 107
6.3 Dynamic Remeshing without Substructuring 111
6.4 Dynamic Remeshing with Substructuring 121
6.4.1 Initial Stresses in a Substructure 123
6.4.2 Effect of Cooling Far behind the Local Zone 128
6.4.3 Results and Comparison 129
6.5 Discussion 134

7 Conclusions and Suggestions for Future Work **135**
7.1 Conclusions 135
7.2 Future Work 137

A Lists of Papers in Welding Simulations and Related Areas **146**

B List of Computer Programs and Subroutines **150**
B.1 User Subroutines for Heat Flux and Boundary Heat Losses 150
B.1.1 3-D Model 150

B.1.2 2-D Plate Model 154
B.2 Programs Used for Dynamic Remeshing 157
B.2.1 The Shell programs 158
B.2.2 The Command Files to Compose an ABAQUS Input Deck 183
B.2.3 The Time Period for an Incremental Analysis 187
B.2.4 The Program to Read Stresses and Extrapolate to Nodal Points . . 188
B.2.5 The User Subrountines to Interpolate Stresses 201
B.2.6 Interaction Forces between the Local Zone and Substructures 206

Chapter 1

Introduction

The welding process is one of the most important manufacturing processes, and is widely used for joining metal in structures such as marine vessels, buildings, and bridges. The welding process is also a remarkably complicated, nonlinear operation involving extremely high temperatures, severe distortions, and high levels of residual stresses which reduce the resistances of a structure to fracture, buckling, corrosion and other type of failures.

Since its invention more than a century ago, welding has been more an art than a science. The behavior of metal during the process—the displacement pattern, stress state, and micro-structural evolution—has not been fully understood. More and more welded structures require higher geometric accuracy, and lower residual stress levels to increase service capacity; recent developments in welding automation also requires the understanding and prediction of metal behavior during the process, particularly distortion for in-process sensing and feedback control.

Although welding is one of the most common technologies used in metal fabrication, there are significant complications when designers attempt to simulate a welding operation

using numerical methods. The energetics of localized melting and mass transfer encompass plasma processes, phase transformations. complicated mass transfer, radiative and convective heating and cooling among different gaseous, liquid, and solid phases. Additionally, this energy transfer is occurring in a multiphase field with each phase possessing poorly characterized, highly temperature-dependent material properties.

The mechanical response of the welded region is similarly complex. Each material point in space may change phase, structure, and composition, with the constitutive behavior of each combination of these parameters temperature-dependent, elastic-viscoplastic. Each parameter of constitutive behavior is poorly understood and characterized, yet full representation of the possible parameters might still be inadequate, for the spatial gradient of compositions may be difficult to map as a function of weld history. All of the above thermal and mechanical processes, which may be strongly coupled, occur within the timespan of a few seconds, involving very large gradients of temperature, stress, deformation[1] and complex micro-structures.

This catalog of complexities also includes the coupling of a weld region with a greater fabricated structure, with associated boundary conditions and deformation responses. The progression of a weld across a structure acts to distort the structure through three dimensions, which couples back to the weld process. One weld pass may not be sufficient, so multiple passes may occur, necessitating multiple simulations of the same region.

1.1 Numerical Simulations of Welding Processes

To fully understand the mechanism of material behavior during and after the welding process, and thus to be able to predict distortions and residual stresses resulting from welding, a set of equations needs to be solved which involves transient heat transfer and thermo-elastic-plastic differential equations. While analytical solutions to such equations are impossible to obtain, numerical solutions become necessary even though many difficulties still remain.

With appropriate boundary and initial conditions, such high order differential equations become the core obstacles to welding simulations. The advent of numerical methods such as finite element, boundary element, finite difference schemes has provided the capability of addressing the initial features of complicated boundary-valued field problems like welding, though the solutions are difficult to obtain due to the high nonlinearity of material properties at high temperatures and high gradients of variables around the electrode. A reasonably accurate solution requires high resolution of mesh around the electrode, which usually pushes most main-frame computers to the limit. This may explain why numerical simulations of most welding processes are still not readily available.

The simulation procedure usually consists of two parts: the heat transfer, and mechanical stress/deformation. With no other loading than heat expansion and contraction, the deformation of a structure under welding depends solely on the temperature field and history. A coupled solution of temperature and deformation is much more difficult to obtain; uncoupled numerical simulations are usually used, greatly simplifying the simulation procedure. In such uncoupled analyses, the temperature history is first obtained in a heat

transfer simulation based on the undeformed structure, and the temperature information is then supplied as loading to the structure in the deformation analyses. The temperature field does depend on the deformation process if large displacements appear, in which case coupled or at least a step-wise coupled analysis is required to accommodate the geometry changes during the heat transfer analysis.

1.2 Literature Review

The research activity in the welding simulation started decades ago. Rosenthall first solved the heat conduction equation for the case of a moving heat source on a semi-infinite body[2]. This solution, though oversimplified, has been used exclusively in the simulation of welding process.

Over the last two decades, many investigators have worked on the numerical simulations of welding process. Most of the early papers report work on two-dimensional finite element analyses of plate or pipe butt-weldings. The most popular were two-dimensional axisymmetric models of cylinders [3, 4, 5, 6] and plane strain models of plates [7, 8, 9]; some reports on two-dimensional plane stress [10] and thin shell models [11] are also available. Three-dimensional models only appear in recent papers [12, 13, 14, 15] on finite element simulations of plate and pipe butt-weldings. All of these papers concentrate on the local phenomena surrounding the electrode or heat source.

There are many papers in related areas. Some investigators have attempted to develop simple models for the welding process so that analytical formulations may be obtained [16, 17, 18]. Some others have worked on developing methods such as the shrinkage force

method [19] and the tendon force method [20, 17]. Many other papers were also published in closely related fields such as mass transfer in weld, phase transformation and microstructure evolution during welding process, rate-dependent constitutive models for metals at high temperatures, welding under loaded conditions, and pre-deformation and pre-heating before welding.

Although recent developments in nonlinear finite element methods have provided the opportunity for complex welding simulations, very little attention has been directed to the issue of welding/structure interactions. The vast majority of finite element simulations have concentrated on resolution of processes occurring within the weld zone, including investigation of such field variables as state of stress (including both transient and residual), local distortion, temperature, microstructure, and composition. Much of the recent successes in welding simulation have resulted from the coupling of nonlinear aspects of the process. Advances by Goldak [21] and Easterling [22, 23] in coupling temperature fields with microstructure have provided some initial steps in addressing the fully coupled welding process as well as extending simulations into three dimensions. The finite element simulations associated with these investigations involve very dense meshes with particular emphasis on resolution within the weld zone, with the consequence that predicted gradients may be resolved with very high accuracy.

Appendix A lists the abundant literature in welding simulations and related research areas. A more detailed review of the welding simulation literature is presented in a separate paper [24].

1.3 Elastic Coupling in Welding of Large Structures

In contrast to resolution of local effects, the issue of the interaction between the welding process and a larger elastic structure has not been considered in detail. Practical welding technology must deal with problems of distortion of large structures due to thermal expansion, lack of appropriate tolerances, and variation in weld variables such as welding gap clearance (or fitup) and fixture location. The distortion of large structures is a particular problem, for the coupled thermo-mechanical character of welding becomes pronounced when thermal expansion and distortion lead to changes in both the orientation of parts and gap clearances.

Practical observations indicate that such large structure effects are very significant. The welding of a ring stiffened cylinder described later in this thesis yielded large relative deformation and twisting of the cylinder under certain fixture setup. Welding on the same structure conducted by Machuca shows similar large deformation[25]. Such phenomena related to large welded structures might not develop in the butt welding of small plates with low ratios of overall structure size to weld zone dimension. Welding simulations are thus required on large structures instead of on only small simple welding situations.

The analysis presented herein addresses issues associated with the welding of large structures. These issues, listed below, are not usually considered in typical analyses of welding which, as mentioned previously, concentrate on weld zone phenomena:

1. **Elastic Coupling - Type 1**

As the weld cools behind the travelling electrode, the newly deposited weld material

couples the two previously independent structures, thereby changing the response of the structure to the ensuing deposition of material. The full effect of this behavior can be simulated only through three dimensional models.

2. **Elastic Coupling - Type 2**

The effect of heating and cooling of the overall structure during welding may introduce large elastic distortions which can not be anticipated when examining only the weld zone. These distortions may be strongly dependent on the geometry of the structure and may have a strong influence on the final states of residual stresses and distortions, within both the weld and the overall structure.

3. **Large Structure Complexity**

The geometric complexity of large structures may require commensurately large finite element models, where the weld zone represents only a small portion of the total structure. Methods to simplify the complexity of the model are therefore necessary to permit reasonable analysis times. The application of such techniques as substructuring and dynamic remeshing reduces the complexity of the model substantially, as is demonstrated later.

4. **Large Deformations**

Given the potentially large coupling between the weld and other parts of a structure, portions of the structure may move dramatically relative to each other, particularly during the initial heating of the first weld pass. These deformations can be large enough to necessitate coupling the thermal and mechanical analyses, since the thermal analysis based on the original, undeformed configuration is no longer applicable to the deformed shape. Coupled, or at least step-wise coupled, heat transfer and stress/strain analyses are then re-

quired, and appropriate interface elements are needed to model the thermal and mechanical interactions along the weld gap when both sides displace to different relative positions.

Welding of large structures therefore involves problems which can not be investigated simply via examination of the weld zone. Additionally, the extent of the interaction between a large structure and the welding of that structure may significantly dominate the final distortion and residual stresses—to the extent that these effects govern the final distorted geometry and state of stress within the weld zone. Full three-dimensional models are thus required to obtain accurate predictions on distortion and residual stresses.

There are several constraints within current finite element analyses which have hindered the analysis of welding of large structures. The primary constraint is the three dimensional complexity of structures, which inhibits reduction of a simulation to a two dimensional model. A full three dimensional analysis increases the complexity of an analysis significantly due to the dramatic increase in total number of degrees of freedom associated with a model; a simple three dimensional model can overwhelm the computational capabilities of a mainframe which could handle a two dimensional geometry with relative ease.

Finite element analysis codes also frequently do not easily accommodate special elements which are more easily implemented for two dimensional models. The absence of special elements to accommodate deforming, conducting, and sliding interfaces inhibits the modeling alternatives available for three dimensional models. There is similar difficulties in modeling the motion of the welding electrode and deposition of weld material in three dimensional models. Additionally, although welding finite element analyses typically decouple the thermal and mechanical simulations, the deposition of a weld bead requires

correlation between the movement of the heat source during the thermal analysis and the deposition of weld material during the mechanical analysis.

A three-dimensional model of a welded structure is computationally too demanding on currently available computer facilities, particularly for those structures with long weld paths. Even though three-dimensional FEM simulations could be carried out for simple problems, the computer CPU time and storage space needed are far from practical for engineering design purposes. This thesis describe a strategy to reduce the total number of degrees of freedom of a three-dimensional FEM model for welding by exploiting the fact that only a small part of the model around the electrode is nonlinear during the process; three-dimensional welding simulation is thus simplified to the extent that a widely available workstation can handle a reasonably complicated welding simulation.

This thesis mainly consists of three parts: the first part discusses the two-dimensional simulation of welding process on a circular ring-stiffened cylinder, the second part involves three-dimensional simulation of a short weld on the same cylinder, and the last part describes the dynamic remeshing procedure to reduce the size of FEM models for welding.

Chapter 2

Fundamentals of Welding Analysis

Welding simulations could be summarized as solving a set of differential equations governing the heat transfer and the mechanical deformation. Proper treatment of boundary and initial conditions and parameters representing material properties are necessary before attempting to solve the equations.

2.1 Basic Equations

The heat transfer process in a solid is governed by the following diffusive equation which takes mechanical coupling into account[26]:

$$\rho c_v \frac{\partial T}{\partial t}[1 + \delta(\frac{\lambda + 2\mu}{3\lambda + 2\mu})(\frac{\dot{\epsilon}_{kk}}{\alpha \dot{T}})] = \bigtriangledown(k \bigtriangledown T) + \frac{\partial q}{\partial t}. \tag{2.1}$$

Here T is temperature, t is time, and q is heat input or generation per unit volume. ρ, c_v, k and α are the material properties, namely, the density, specific heat (under constant volume), thermal conductivity and coefficient of thermal expansion, respectively; λ and μ are Lame's elastic constants. The dilatational strain rate is defined by:

$$\dot{\epsilon_{kk}} = \sum_{i=1}^{3} \frac{\partial \epsilon_{ii}}{\partial t} \tag{2.2}$$

The dimensionless parameter δ is defined by:

$$\delta = \frac{(3\lambda + 2\mu)^2 \alpha^2 T_0}{\rho^2 c_v v_e^2} \tag{2.3}$$

with v_e the velocity of propagation of dilatational waves expressed as:

$$v_e = \sqrt{(\lambda + 2\mu)/\rho} \tag{2.4}$$

The coupling term $\delta(\frac{\lambda+2\mu}{3\lambda+2\mu})(\frac{\dot{\epsilon_{kk}}}{\alpha \dot{T}})$ in Eq. 2.1 is much less than unity for most of the engineering materials such as steel and aluminum. This term is usually dropped in heat transfer simulations of welding processes. The decoupled heat transfer equation is then simplified to:

$$\rho c_v \frac{\partial T}{\partial t} = \nabla(k \nabla T) + \frac{\partial q}{\partial t}. \tag{2.5}$$

With proper boundary and initial conditions, this equation can be solved to obtain a temperature field as a function of time.

Equilibrium requires that the stress components satisfy the following equations:

$$\sum_{j=1}^{3} \frac{\partial \sigma_{ij}}{\partial x_j} + b_i = \rho \frac{\partial^2 u_i}{\partial t^2} \qquad (i = 1, 2, 3) \tag{2.6}$$

Where b_i is the body force such as gravity. The material accelerations (or inertial forces) $\rho \frac{\partial^2 u_i}{\partial t^2}$ are caused by the transient expansion or contraction due to temperature change. The magnitudes of such inertial forces are usually very small unless at extreme temperature rate[26]. These dynamic forces are generally neglected, so that a static mechanical

deformation process can be used in welding simulations. The governing equation is then simplified to:

$$\sum_{j=1}^{3} \frac{\partial \sigma_{ij}}{\partial x_j} + b_i = 0 \qquad (i = 1, 2, 3) \tag{2.7}$$

The strain increment during deformation can be divided into three parts, the elastic, plastic and thermal expansion, with the plastic strain increments only occurring when material yield stress is reached:

$$d\epsilon_{ij} = d\epsilon^e_{ij} + d\epsilon^p_{ij} + d\epsilon^t_{ij} \tag{2.8}$$

Where $d\epsilon_{ij}$ is the ij component of the incremental strain tensor, and δ_{ij} that of a unit tensor called the Kronecker delta and defined as:

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

The material constitutive relations are defined separately according to elastic and plastic deformation as well as thermal expansion:

$$d\epsilon^t_{ij} = \alpha \delta_{ij} dT \tag{2.10}$$

$$d\epsilon^e_{ij} = \frac{1}{E}[(1+\nu)d\sigma_{ij} - \nu\delta_{ij}(\sum_{l=1}^{3} d\sigma_{ll})], \tag{2.11}$$

$$d\epsilon^p_{ij} = \frac{3}{2}\frac{d\bar{\epsilon}^p}{\bar{\sigma}}[\sigma_{ij} - \frac{1}{3}\delta_{ij}(\sum_{l=1}^{3} \sigma_{ll})], \tag{2.12}$$

where E and ν are Young's modulus and Poisson's ratio, $\bar{\sigma}$ is the Mises equivalent tensile stress, and $d\bar{\epsilon}^p$ is the equivalent plastic strain increment defined by:

$$d\bar{\epsilon}^p = \frac{3}{2M} \sum_{k,l=1}^{3} \sigma'_{ij} d\sigma'_{ij}. \tag{2.13}$$

here M is the plastic modulus representing strain hardening. The deviatoric stress and stress increment tensors are defined by:

$$\sigma'_{ij} = \sigma_{ij} - \frac{1}{3}\delta_{ij}\sum_{l=1}^{3}\sigma_{ll}, \tag{2.14}$$

$$d\sigma'_{ij} = d\sigma_{ij} - \frac{1}{3}\delta_{ij}\sum_{l=1}^{3}d\sigma_{ll}. \tag{2.15}$$

In all the equations above, material properties are nonlinear functions of temperature as described in the following sections. Such material parameters need to be updated frequently during an analysis. The strain increments must also satisfy other equations which ensure the material continuity and compatibility.

Welding simulation is to solve these equations with appropriate tools so that the structure interactions during welding can be analyzed. Transient, final distortions and stress states can also be predicted.

2.2 Boundary and Initial Conditions

The boundary conditions for mechanical analyses are based on the physical constraints supplied in a welding process, such as the constraints supplied by a clamp or a tack weld. Pressure from the high speed plasma on the base metal is currently ignored, though experiments indicate that such pressure has an effect on the shape of the molten weld pool[27, 28]. The current analyses do not take into account the weld pool formation and evolution.

Thermal boundary conditions are more complicated, including heat input and nonlinear boundary convection and radiation. The following summarize their special treatments in our finite element analyses.

Heat input: Heat transfer to the base metal from the electrode arc is modeled by a surface heat flux on elements near the weld. The net power input is chosen from the gross power and the arc efficiency. The power distribution for an electrode above a flat surface is generally assumed to be a Gaussian curve following Tsai[29]. For the case of the cylinder and stiffener structure, the local weld base metal is a T-shaped geometry; the Gaussian function needs modification as we discuss later in the three-dimensional modeling of the ring-stiffened cylinder.

Boundary heat loss: Boundary heat transfer is modeled by heat convection and radiation. Convection follows Newton's law, where the coefficient of convective heat transfer is assumed to vary with both temperature and orientation of the boundary[30]:

$$h_c = \frac{kN_u}{L} \tag{2.16}$$

Here k is the thermal conductivity, L the characteristic length of the plate (or surface), and N_u the Nusselt number defined by:

$$N_u = 0.137 P_r^{1/3} G_r^{1/3} \qquad \text{(Horizontal plate, facing upward)} \tag{2.17}$$

$$N_u = 0.54 P_r^{1/4} G_r^{1/4} \qquad \text{(Horizontal plate, facing downward)} \tag{2.18}$$

$$N_u = 0.517 P_r^{1/4} G_r^{1/4} \qquad \text{(Vertical plate)} \tag{2.19}$$

P_r is the Prandtl number and G_r the Grashof number, both of them functions of gas properties and temperature differences between the wall and the environment[31]. They are defined by:

$$G_r = \frac{g\beta(T - T_a)L^3\rho^2}{\mu^2} \tag{2.20}$$

$$P_r = \frac{c\mu}{k} \tag{2.21}$$

Where g is the gravitational acceleration, β the material inertial pressure drop constant, T_a the temperature of the ambient air, L the characteristic length of the plate (e.g. the width), ρ the material density, and μ the viscosity of the ambient air.

Radiation is modeled by the standard Boltzman relation:

$$q_r = \epsilon\sigma(T^4 - T_a^4) \tag{2.22}$$

Where ϵ is the heat emissivity, and σ the Stefan-Boltzman constant. Radiation is assumed to be from surface to ambient air; no radiative interaction between surfaces is considered. Comparison of the two factors shows that the effect of radiation is generally smaller than that of convection except near the melting temperature.

The structure is assumed to be initially at room temperature and stress-free before the welding starts. Welding materials are assumed to be at melting point just before their deposition. Note that complicated initial temperatures and stress distributions could be applied with the currently available finite element codes.

2.3 Material Properties

Many different kinds of materials have been used in structures where welding is involved, with low carbon steel the most common. Mild steel is chosen for the analyses throughout this thesis. The deposited weld material is also assumed to be mild steel. Welding process

on other materials could be similarly simulated by replacing the material definitions without much difficulty.

1. *Thermal properties*: Material thermal properties are assumed to be temperature-dependent. The variations of thermal conductivity, specific heat, thermal emissivity, and convective heat transfer coefficient are given in Figure 2.1(a). Different convective coefficients are used for the vertical and the upper and lower horizontal surfaces. An averaged value of thermal emissivity is used since the FEM code (ABAQUS) we are using does not permit the use of temperature-dependent thermal emissivity. Rate-dependent effects are currently ignored.

2. *The latent heat effect*: The release or absorption of energy upon solidification or melting is accommodated through a latent heat model available in ABAQUS, which modifies the specific heat to accommodate the extra heat absorption or release[32]. The latent heat effect associated with the solid-solid phase change, which is much smaller than that associated with the solid-liquid phase change, is already included in the specific heat, as shown in Figure 2.1(a).

3. *Mechanical Properties*: Mechanical properties are also temperature-dependent, including Young's modulus, yield stress, and coefficient of thermal expansion[33]. Material hardening corresponding to mild steel is assumed to be isotropic and also temperature-dependent. Figure 2.1(b) shows the dependencies of material properties on temperature. Young's modulus and yield stress are given small, finite values at high temperatures to avoid difficulties with numerical convergence.

Rate dependence and microstructure evolution are not included currently in the analyses. Accuracy of welding simulations would improve due to the inclusion of vicoplasticity and microstructure evolution. Understanding of the material behavior and deformation mechanism could also be extended by examining the microstructure changes. Future work may include such effects since they are relatively significant when the material is at high homologous temperatures and experiencing rapid temperature changes [34, 35, 36].

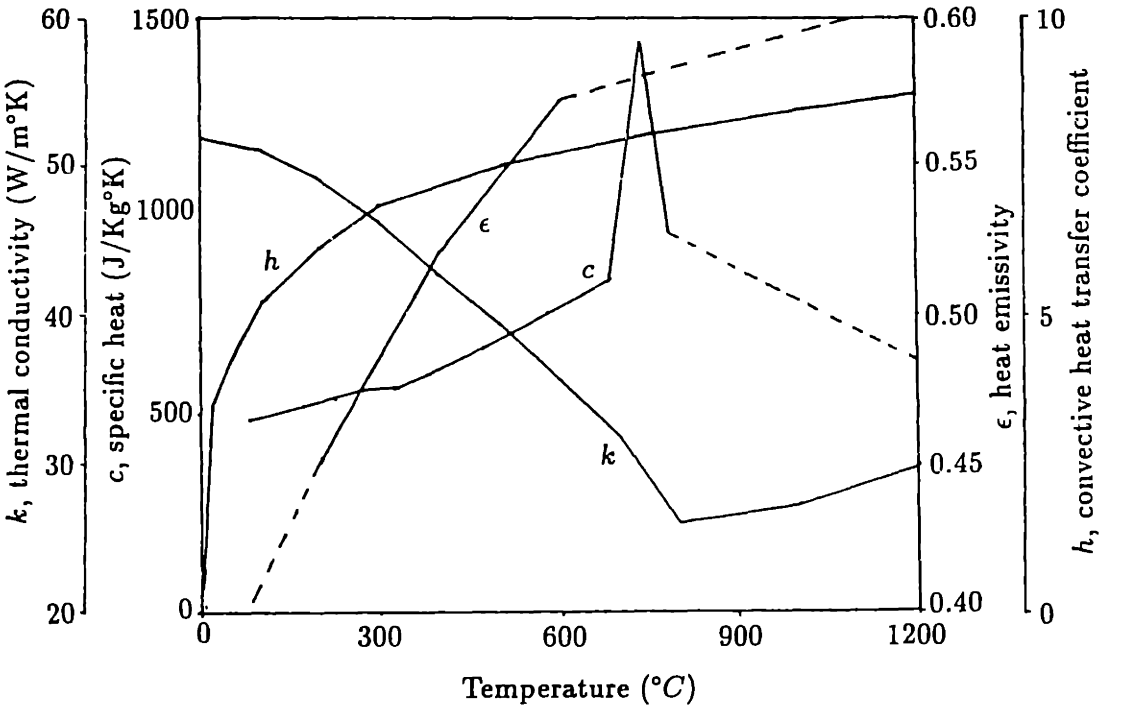
2.4 FEM Formulations

The finite element method is a generalized Rayleigh-Ritz method which uses interpolation to express variables in terms of their values at a finite number of "nodes." These values are found by solving a set of simultaneous algebraic equations. The accuracy of the solutions depends mainly on the number of nodes selected.

Within one element, a variable could be interpolated in the following manner by interpolation or shape functions in terms of the nodal values:

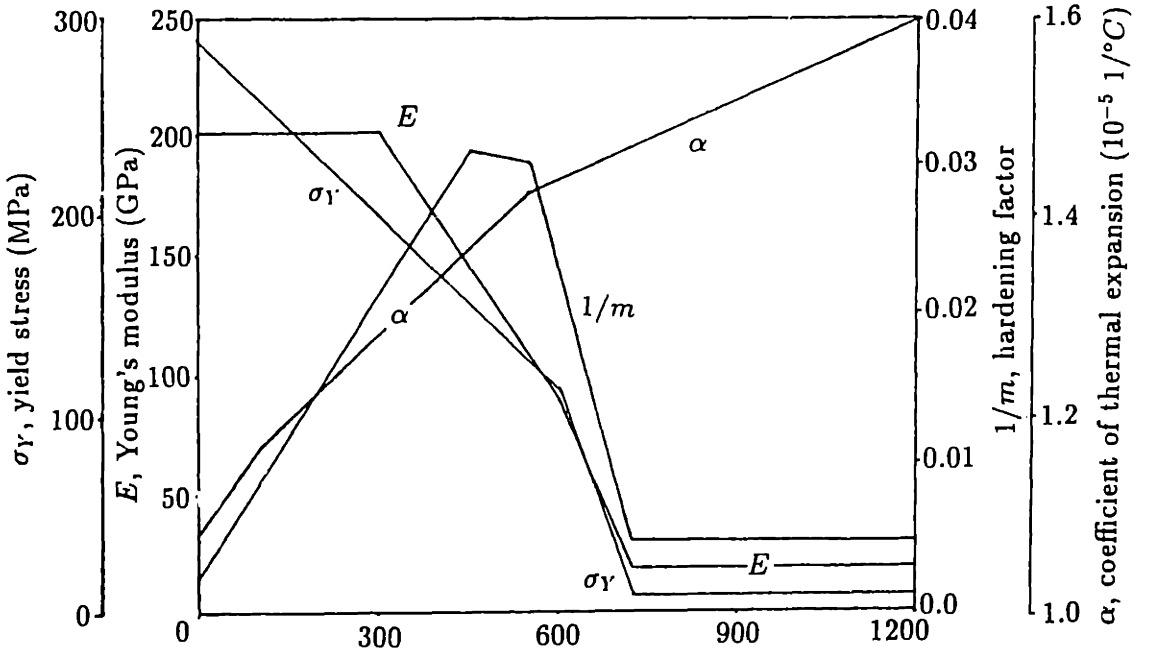
$$u = \sum_{i=1}^{n} N_i(x, y, z) U_i \tag{2.23}$$

here n is the number of nodes composing the element, and the shape functions $N_i(x, y, z)$ are usually polynomial functions and satisfy certain conditions such as compatibility along element boundaries. Through such interpolation functions, strain components can be derived based on the displacements, and stress components can be further obtained through the constitutive relations mentioned before. A static problem and a steady state heat transfer are then fomulated by the following algebraic matrix equation:[37, 38]



[h from Lewis, 85. k, c, ϵ from ASM *Metal Reference Book*, 83]

(a) Thermal properties



[from Iwamura & Rybicki, 73]

(b) Mechanical properties

Figure 2.1: Thermal and mechanical material properties of mild steel

$$[K]\{U\} = \{F\} \tag{2.24}$$

$[K]$ is the stiffness matrix, $\{U\}$ is the nodal displacement vector, and $\{F\}$ the nodal force vector. In the heat transfer analysis, they correspond to the conductivity matrix, the nodal temperature vector, and the nodal heat input vector.

Finite element formulations for transient processes are similar to the above except that displacements or temperatures are expressed in incremental form. The formulation of transient heat transfer is different from Eq. 2.24, though the incremental formulation can be simplified to a similar form.

Chapter 3

2-D Simulations of Welding of a Circular Ring-Stiffened Cylinder

A circular cylinder welded with a web stiffener, as shown in Figure 3.1, is one of the most common sections in the variety of large structures. Because of the local asymmetry around the weld pass and elastic coupling from the other side opposite to the currently welding electrode, the deformation pattern is also complicated. Such a structure typically shows large structure effects. The next two sections will be exclusively devoted to the FEM simulations of welding process on such a structure.

3.1 The 2-D FEM Model

As a first step in the simulation of the welding process on this flange-web structure, a two-dimensional FEM model simplifies the simulations by reduction of requirement on computer resource yet still gives reasonable insight to the problem, as demonstrated later in this chapter.

The axisymmetric finite element model of a ring stiffened cylinder is shown in Figure 3.2.

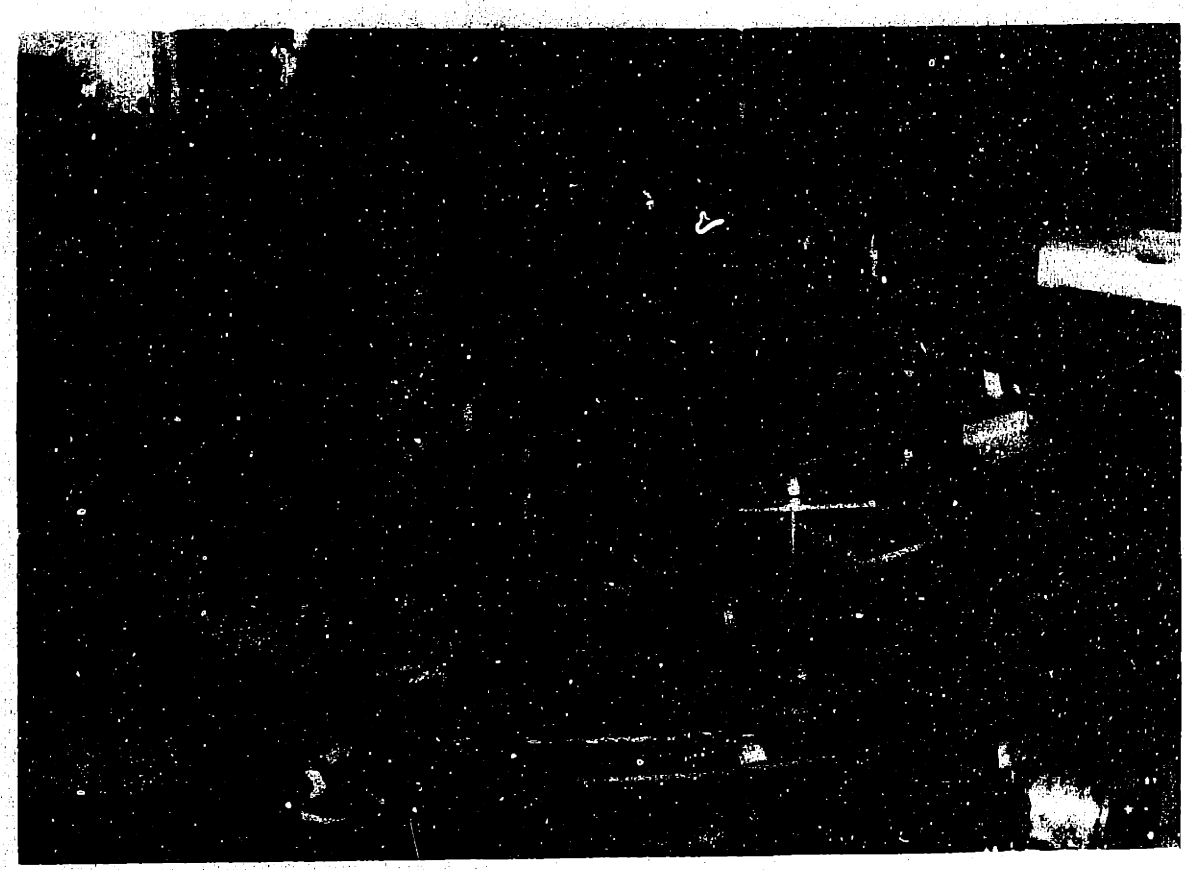


Figure 3.1: Picture of the ring stiffened cylinder

Such a model assumes symmetry of the welding process where the weld is performed simultaneously around the circumference of the structure. This assumption is rigorously valid only when the traveling velocity of the electrode is high enough to make the three-dimensional effect negligible. Such electrode velocity rarely, if ever, occurs in actual welding. In spite of this rather severe approximation, features of structure/weld interactions can be investigated in two-dimensional models.

The mesh was chosen for its simplicity. Denser meshes were also investigated; minor differences were found between the mesh provided herein and the finer meshes. The same meshes were used for both the thermal and mechanical analyses. All the two-dimensional analyses use the nonlinear finite element code ABAQUS and were performed on a VAX-3500 computer.

Each welding pass is simulated by three steps. The base metal is first heated, and then at the end of the heating, weld material is deposited; finally, the whole structure cools through conduction and boundary heat losses. The deposition of material is modeled by an element activation technique available in ABAQUS. Elements of weld material are deactivated at the beginning of the heating, and then reactivated at the melting temperature of the metal, following by cooling after the weld electrode passes[13].

The welding simulations were carried out by decoupled analyses, that is to say, a heat transfer analysis provides a temperature field history to be used in a following stress/strain analysis. Such analyses greatly simplify the simulations by recognizing the dependency of stress/strain field on temperature, while ignoring the coupling of the deformed geometry and the temperature field. When the relative displacement of the parts to be welded is

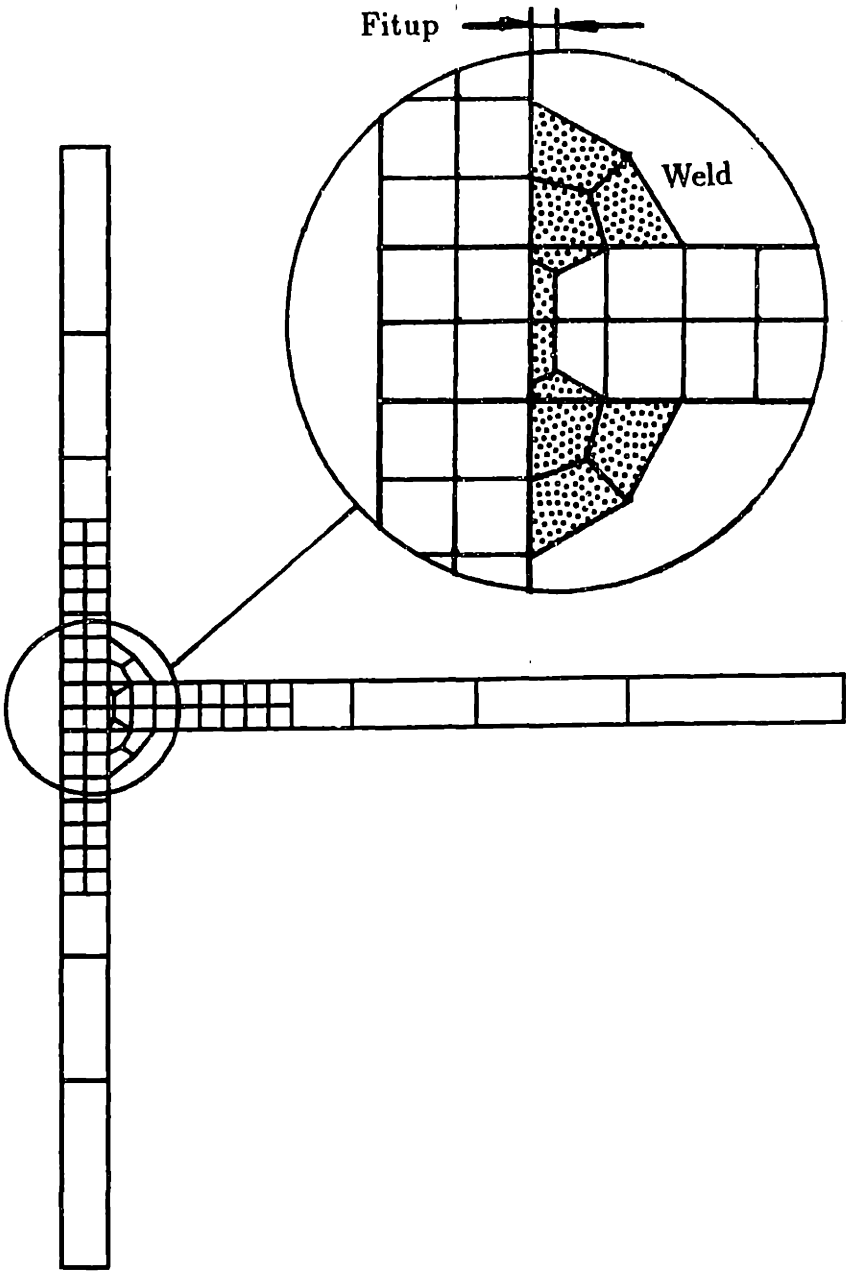


Figure 3.2: 2-D mesh of the ring stiffened cylinder

large, a coupled heat transfer and stress/strain analysis is required to include the effect of deformation in the temperature field.

3.2 Heat Transfer Analysis

The heat transfer analysis follows the assumptions discussed in Chapter 2. Each of the weld passes involves two steps, heating and cooling. The structure is heated near the notch with the weld material deactivated; at the end of heating, elements inside the weld region are reactivated at a temperature close to the melting point (1600°C in these cases) to simulate the deposition of molten material. During the cooling step the deposited material cools with the rest of the structure through conduction, boundary convection, and radiation. Conduction, however, dominates the process. It takes several minutes for conduction to equilibrate the flange and web to a uniform temperature, and a couple of hours for boundary heat transfer to bring the structure to room temperature.

In the analyses, the average net power input or heat flux to the base metal is assumed to be 800 Watts/cm^2, spreading on a surface of about 0.4 cm^2 in area near the weld groove. The duration of heating is 15 seconds, which is equivalent to a travelling velocity of 5 cm/min. A net power of about 400 Watts/cm^2 is also required to melt the electrode continuously. Such welding parameters correspond to a typical welding process with a gross power input of 2,000 Watts/cm^2 and an arc efficiency of 60%. Material properties are those of mild steel and are shown in Figure 2.1.

Figure 3.3 shows the transient temperature distribution after weld material is deposited. Discontinuities in the temperature contour are artifacts of the contour plotting

software. These temperature histories are then imposed on the model for the stress/strain analyses.

3.3 Stress/strain Analysis

Deposited weld material is modeled, as described earlier, by reactivating elements which are deactivated at the beginning of the analysis. The activation procedure is to deform the deactivated element from its original shape so that it will be compatible with other parts of the structure in a state of force equilibrium. Although this procedure creates a stress/strain field in the newly born element, the high temperatures of the element reduce the effect of this virtual stress on the structure, since the yield stress is very small and material flows at melting temperature. Similar treatment of deposited welding material can be found in Tekriwal [39, 13].

Plastic deformation develops due to the temperature spatial gradients and material softening at elevated temperatures. After cooling, the structure retains a distorted configuration and a state of high residual stress, particularly near the weld path.

Figure 3.4 shows the distorted geometry of the structure predicted by the axisymmetric model after one weld has been deposited. A 3.2 mm ($\frac{1}{8}$ inch) weld gap clearance (fitup) is used in this model. One point on both the flange and the web is pinned in the axial direction to remove free body displacements. The pinned point on the web is located about 11.1 mm ($\frac{7}{16}$ inch) from the weld center line. Notice the very large distortions, primarily due to the elastic deflection associated with thermal expansion, which is then retained by the weld.

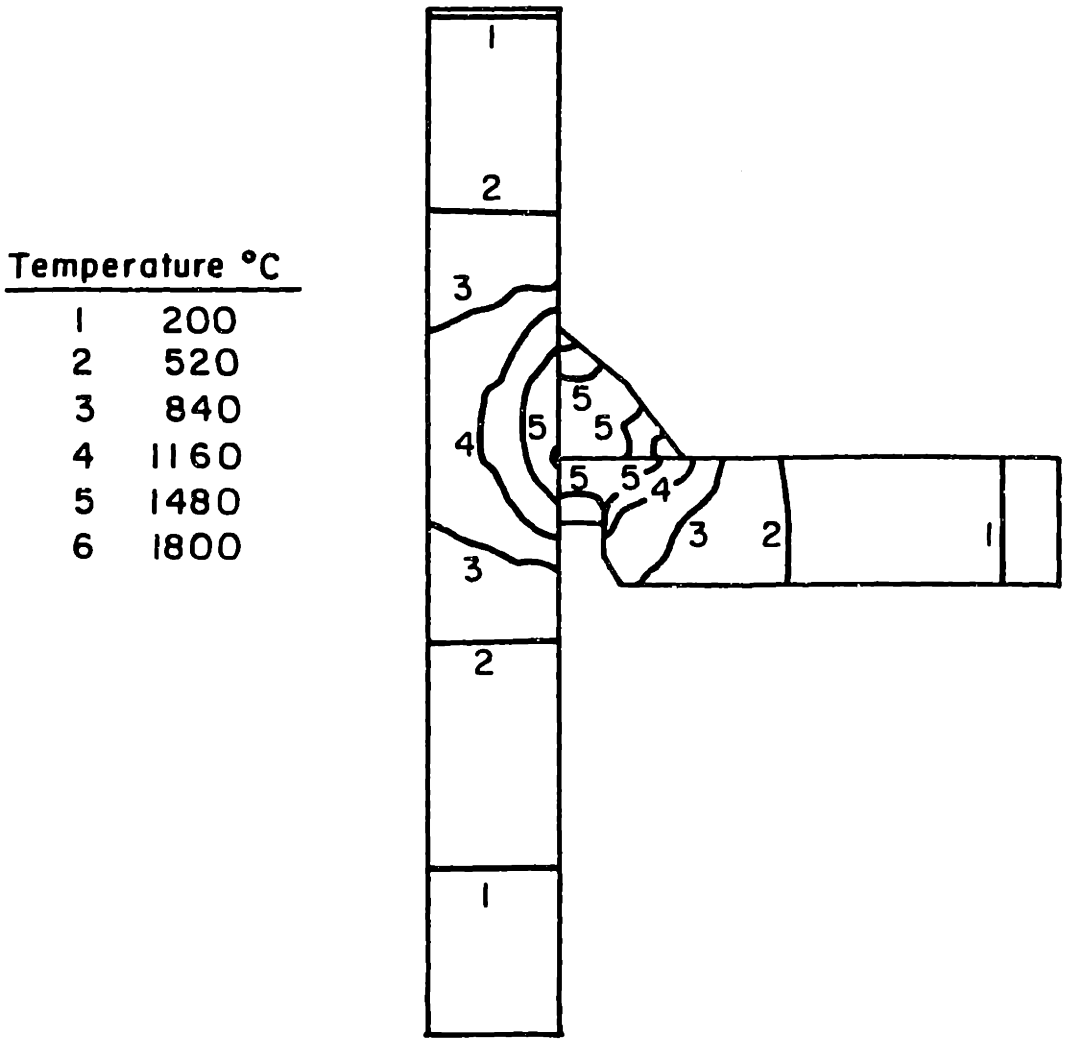


Figure 3.3: Transient temperature distribution immediately after the deposition of the first weld on the 2-D model of the ring-stiffened cylinder. The plot only displays the denser part of the mesh

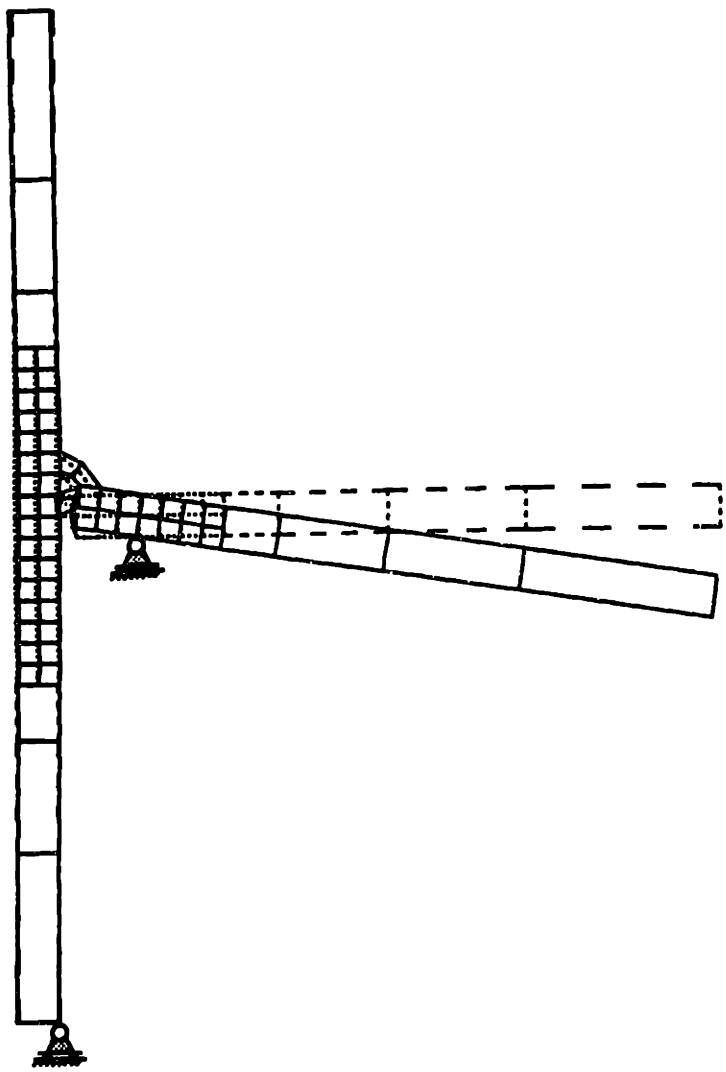
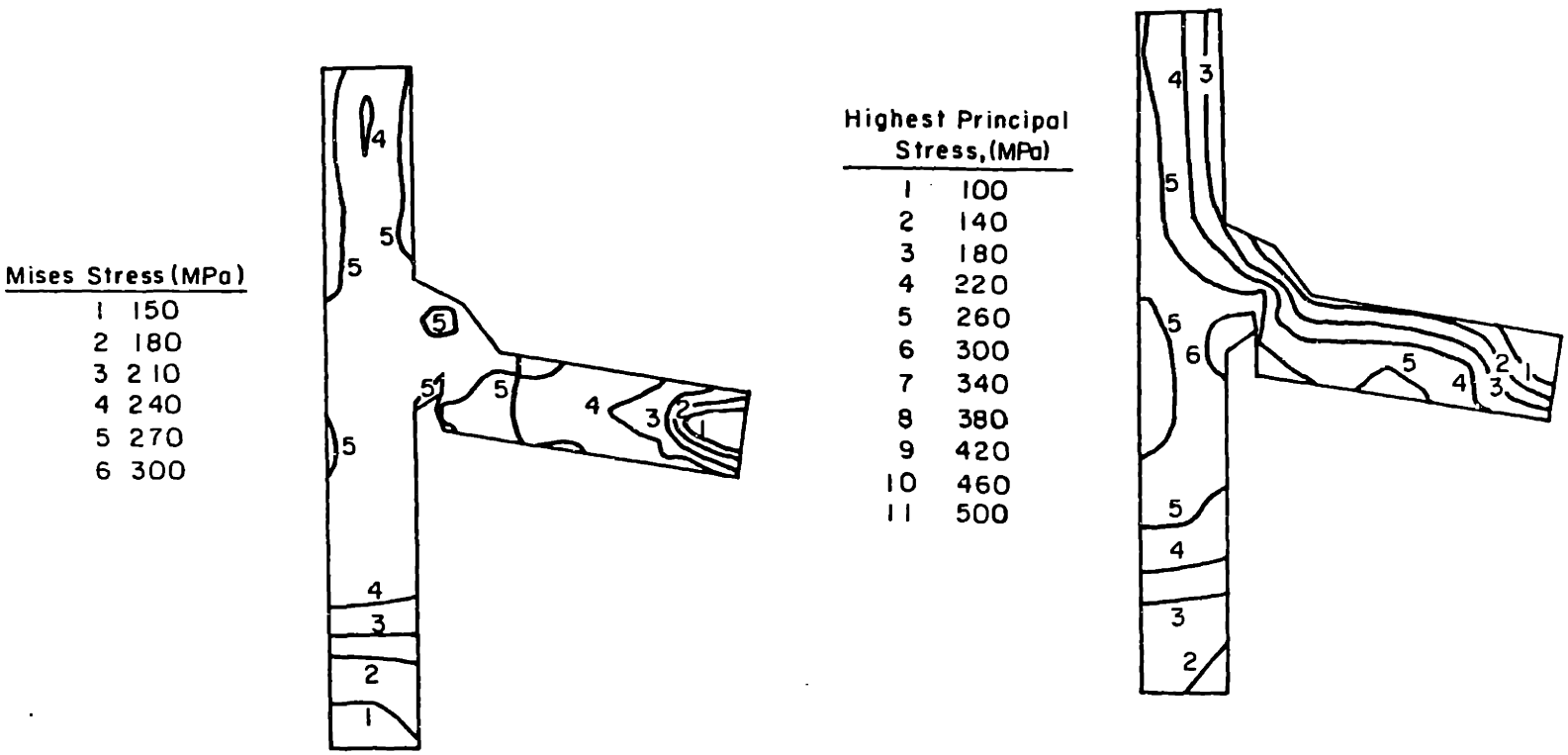


Figure 3.4: Deformed 2-D mesh of the ring-stiffened cylinder after the first weld. Web pinned at 11.1 mm ($\frac{7}{16}$ inch) from the weld center line.



(a) Mises stress

(b) Highest principal stress

Figure 3.5: Residual stress distributions after first welding pass with fixture removed. Two-dimensional model and 3.2 mm ($\frac{1}{8}$ inch) weld fitup.

Figure 3.5 shows the corresponding residual Mises stress and highest principal stress contours. The highest principal stress corresponds closely to the circumferential stress. High tensile residual stress components develop in the local area around and inside the weld, particularly in the gap between the flange and web.

In the welding process, fitup between parts, along with the appropriate fixture style and positions, are usually selected based on individual design experience. Our analysis suggests that the importance of such selections on the final states of residual stress and distortion has been underappreciated. Distortion and residual stresses are highly dependent on the setup of the welding process, i.e. the fitup condition, the position and style of fixtures. The 2-D FEM analyses described below illustrate the sensitivity of distortion and residual stresses to various weld parameters.

3.4 Fitup Effect

The relative configuration of components to be welded has a significant effect on the distribution of stresses near the weld. Welded components are frequently asymmetric, and distort either due to asymmetric elastic distortion or due to asymmetry of the welding process. If interaction between the components (contact, for example) occurs before deposition of weld material, the deformation pattern due to such interaction will obviously be different from that when no contact occurs.

The ring-stiffened cylinder considered here illustrates this sensitivity to fitup. The cylinder is more flexible radially than the web stiffener, and when heated, it expands more in the radial direction. If contact occurs during heating, further expansion of the cylinder

is confined by the stiffener. Such incompatibilities results in deformation and stress/strain fields different from the case when no contact occurs. These differences are accentuated after cooling. We present two cases here, with fitup between the web and flange being 3.2 mm ($\frac{1}{8}$ inch) and 0.4 mm ($\frac{1}{64}$ inch). There is no contact during heating for the former case, while contact occurs in the latter before the end of heating.

Figures 3.6 and 3.7 shows the stresses at integration points near the free surface of the weld for different weld clearance gaps. Although the Mises equivalent tensile stresses are similar, there is a large difference in individual components of stress.

Highest principal stresses

The highest principal stress corresponds to the circumferential stress components in this problem. The two cases differ most in highest principal stresses along the free surface of the weld. The highest principal stress values for the case of 0.4 mm ($\frac{1}{64}$ inch) fitup is about 50 to 100 percent higher than those for the case of 3.2 mm ($\frac{1}{8}$ inch) of fitup gap. We include highest principal stress as an indicator of susceptibility to various failure modes, including hot cracking, brittle fracture, and fatigue [40].

Mean normal stress

Mean normal or hydrostatic stresses in the 3.2 mm ($\frac{1}{8}$ inch) fitup case are lower than those in the 0.4 mm ($\frac{1}{64}$ inch) case. More significantly, the mean normal stresses are either negative or close to zero in the former case but positive in the latter case. The change of signs in the hydrostatic stress will change the state of quality of the weld since a positive hydrostatic stress (negative pressure) favors void nucleation.

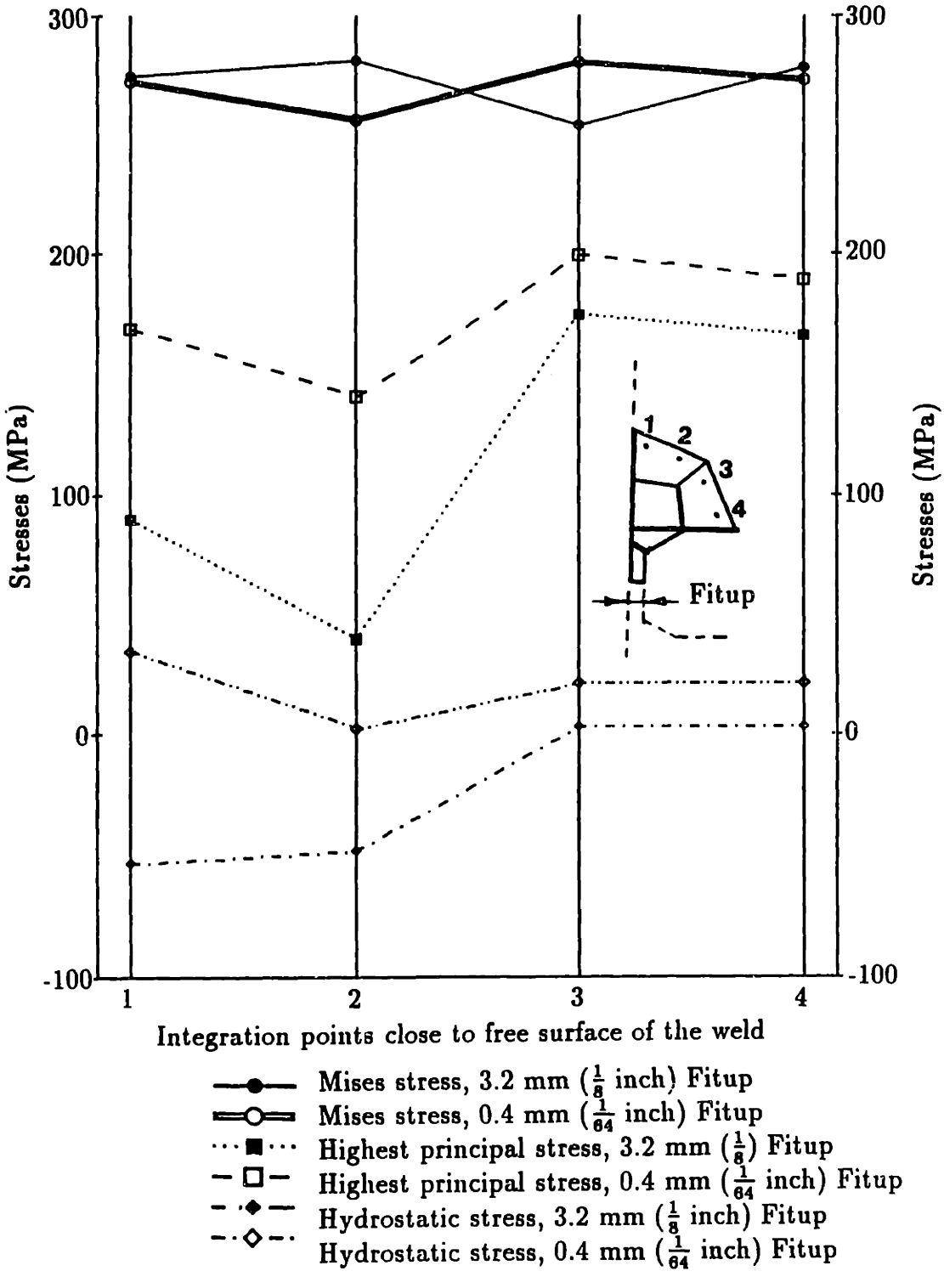


Figure 3.6: Values of residual stresses at integration points near the free surface of the first weld. Two-dimensional model.

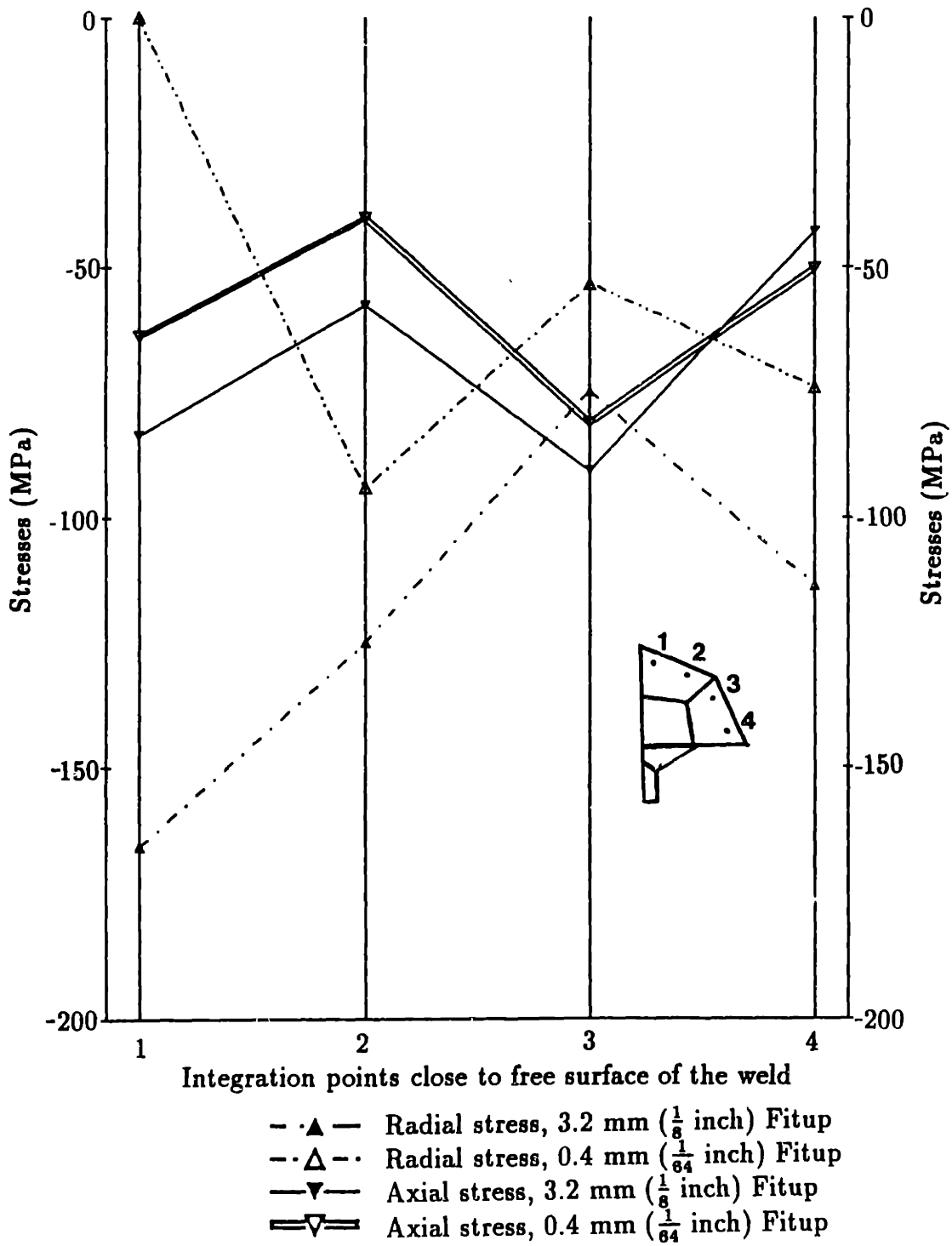


Figure 3.7: Values of residual stresses at integration points near the free surface of the first weld. Two-dimensional model.

Radial and axial stress

Both radial and axial stresses in the case of 3.2 mm ($\frac{1}{8}$ inch) gap are lower (more negative) than those of 0.4 mm ($\frac{1}{64}$ inch), though the differences in axial stresses are small.

Quality criterion based on crack or void initiation and growth indicate that, a fitup of 3.2 mm ($\frac{1}{8}$ inch) appears significantly less susceptible than a fitup of 0.4 mm ($\frac{1}{64}$ inch). The major difference between the two fitups is the interaction in the gap between the flange and web during heating. The fitup case with 0.4 mm ($\frac{1}{64}$ inch) gap generates contact between the web and flange during heating. Other FEM calculations show that the stress fields are similar in cases with no gap contact. Cases where contact occurs possess commensurately similar stress fields. The critical fitup gap that determines the occurrence of contact or no contact is in the range of 0.8 mm ($\frac{1}{32}$ inch) to 1.6 mm ($\frac{1}{16}$ inch) for this particular structure. The value of critical fitup gap depends on the heat input, material properties, and geometry of the structure.

The large difference in residual stress components in the case of 0.4 mm ($\frac{1}{64}$ inch) of fitup may be explained by contact of the welded parts which occurs during heating. A portion of the resulting elastic incompatibility remains after the structure cools, which in turn results in a state of higher residual stress. Given the significant effect of fitup variation on residual stresses, it seems reasonable to include fitup variation as a integral part of any weld simulation.

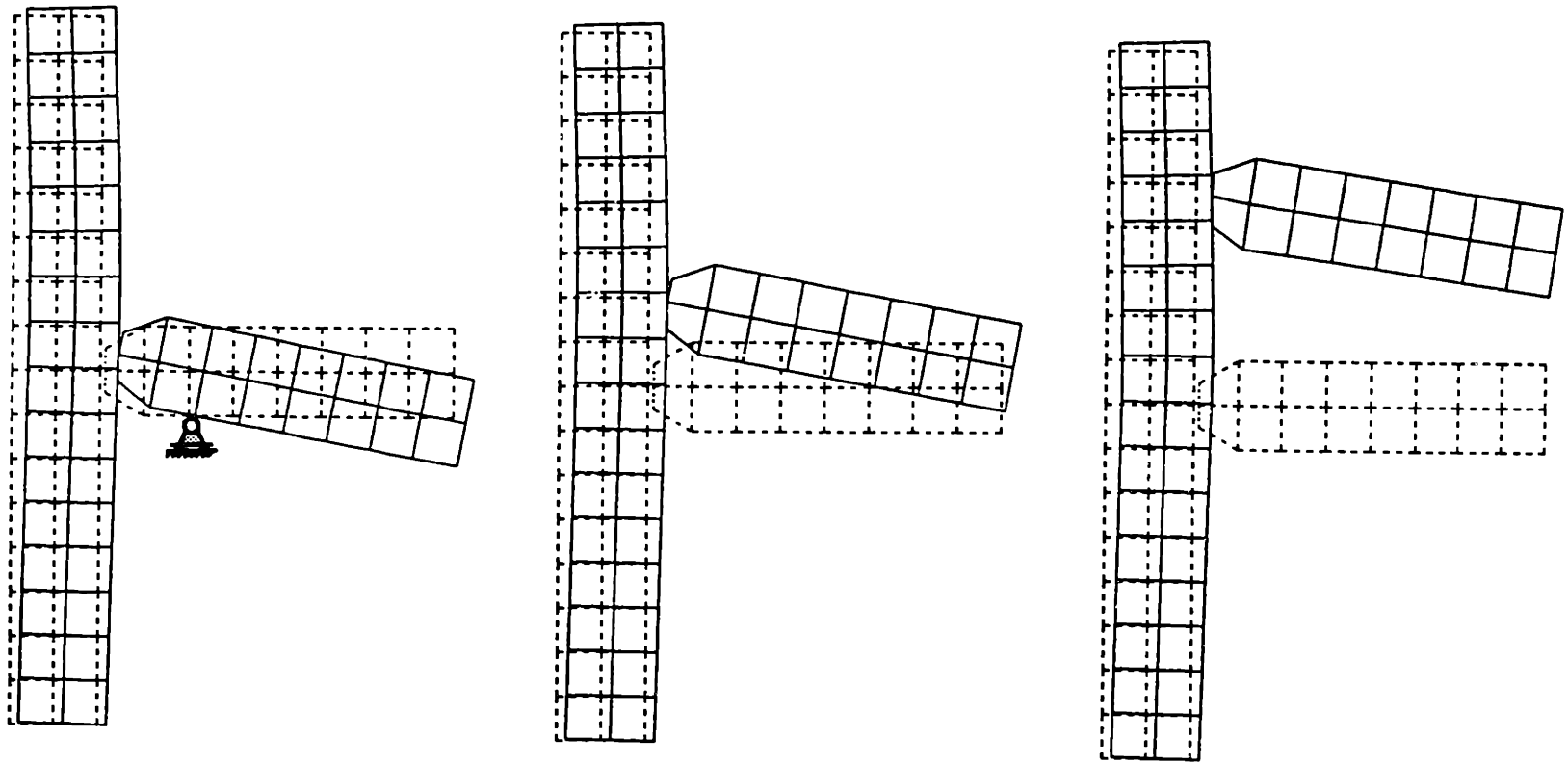
3.5 Fixturing Effects

Although the analysis of fixturing may be considered to be a subset of the larger topic of elastic coupling, we consider here an example of fixturing to illustrate how small variations in the linear elastic portion of a structure can lead to potentially very large variations in deformation and stress response. Fixturing limits the deformation of a structure during welding by the imposition of what can be considered to be very stiff elastic constraints on the structure. The appropriate location of fixtures plays an important role in the final quality of welds, particularly in terms of distortion and residual stresses. The following paragraphs consider different aspects of the influence of fixtures.

1.Pinning of axial rigid body motion

Since an axisymmetric finite element model requires that at least one node on each independent structure be constrained in the axial direction to prevent rigid body motion, we simulate the positioning of a fixture by removing one node's axial degree of freedom. We fix both the flange and the web in the axial direction (direction 2 in the figures), but permit radial displacement. After the first welding pass the flange and web are joined, requiring only one constraint.

The constrained node on the flange is placed at its bottom edge, and the fixture on the web is placed at different positions along its lower surface. Figure 3.8 illustrates the relative motion of the web and flange at the end of the first heating sequence for three positions of the web fixture: near the web, at the middle of the web, and at the outside edge of the web. Heat is applied in this simulation first to the upper weld.



(a) Pinned near the weld as indicated

(b) Pinned at middle of web

(c) Pinned at outside edge of web.

Figure 3.8: The deformed shapes of two dimensional models at end of heating for different fixture locations.

Maximum relative displacement between the web and flange results when the outside of the web is pinned. The magnitude of this motion carries implications for both the actual welding process and the modeling of that process. Large displacements like that of Figure 3.8(c) obstruct the use of automatic welding equipment. Such large displacements invalidate the use of a decoupled thermal and mechanical analyses, and alter the state of residual stress in the cooled weld zone due to the return of the elastic structure from its distorted geometry. Relatively minor variations in fixture location therefore carry significant consequences for the state of deformation and stress in the final welded structure.

2. Clamping both sides of the web

The model was modified to simulate more accurately a clamping fixture which restricts motion of both sides of the web. The effect of this greater constraint, imposed again through the elimination of the axial degree of freedom on opposing nodes on the upper and lower web surfaces, is to reduce rotation of the web. Figure 3.9 illustrates the deformed shape of the flange/web structure after simulating the deposition of one weld pass. Much less distortion results relative to that illustrated in Figure 10, but greater residual stresses result.

Figure 3.10 contains principal stress contours for the clamped fixture case. Higher principal residual stresses develop near the free surface of the weld metal. The less constrained fixture geometries, on the other hand, have reduced highest principal stresses near the free surface, as shown in Figure 3.5(b). The effect of increased constraints is to reduce the distortion while increasing the residual stress by limiting the structure's accommodation of shrinkage of the weld pool.

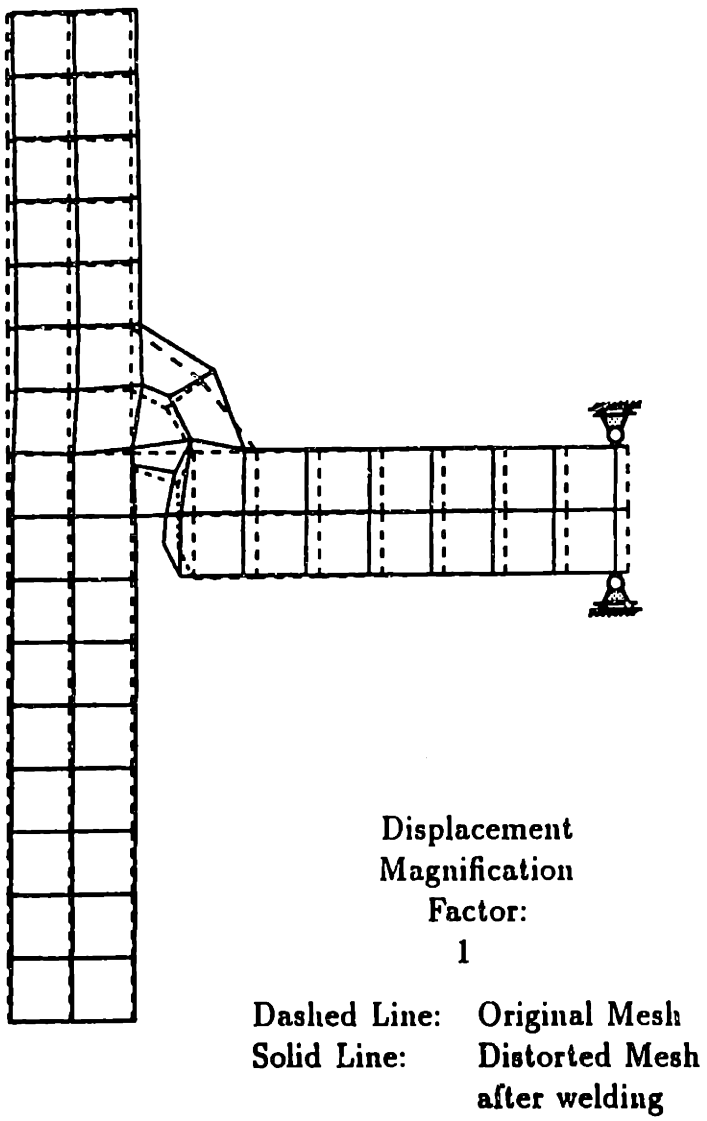


Figure 3.9: Relative distortion of two-dimensional model for clamped web fixturing. Distortion represents condition after first weld pass and removal of fixtures. Only the local region near the weld is pictured here.

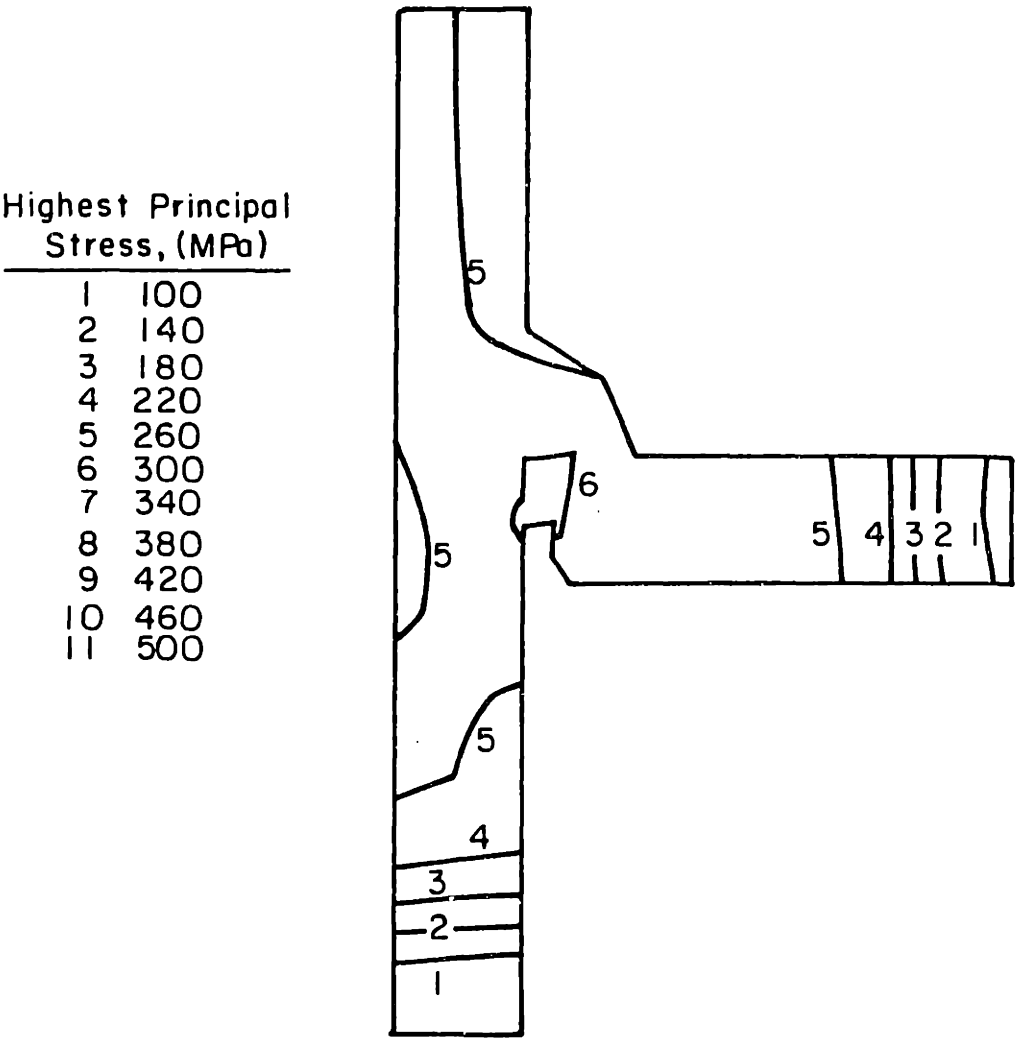


Figure 3.10: Residual highest principal stress contours after first weld pass. Two-dimensional model with web clamped. Only the local region near the weld is plotted.

3.6 Correlation with Welded Structure

Instrumented welding of a full circular web and flange structure pictured in Figure 3.1 has been performed, permitting comparison with our two-dimensional analyses. The web and flange in the test specimen were machined to reduce the fitup gap to less than 1.6 mm ($\frac{1}{16}$ inch). Both the web and flange rested on supports but were otherwise not clamped. Welding was performed using an indexing table which rotated the structure relative to a fixed welding electrode. The weld path traversed a 90 degree arc, or approximately 718.8 mm, welding one side of the web/flange seam.

The primary distortion consisted of an axial motion of the web from the center line of the flange due to the thermal expansion. This axial (vertical) distortion of the specimen is illustrated in Figure 3.11, where the motion of the web relative to the flange center line is plotted as a function of circumferential position. Other distortions included tilting of the web towards the flange in the welded region and some loss of axisymmetry of the flange.

A special two dimensional model of the actual welding process was necessary to duplicate the boundary conditions used in the experiment. A rigid surface was defined below the web to simulate the support without clamping; soft spring elements were attached to the web to remove rigid body motions. The remainder of the simulation employed the same boundary conditions and material constants in the previously described analyses with the exception of the accommodation of large deformations. In this case uncoupled thermal and mechanical analyses were performed twice. The first analysis obtained an estimate of the motion of the web relative to the flange. The second analysis then duplicated this motion by positioning the heat input to the flange at the predicted, distorted location, thereby

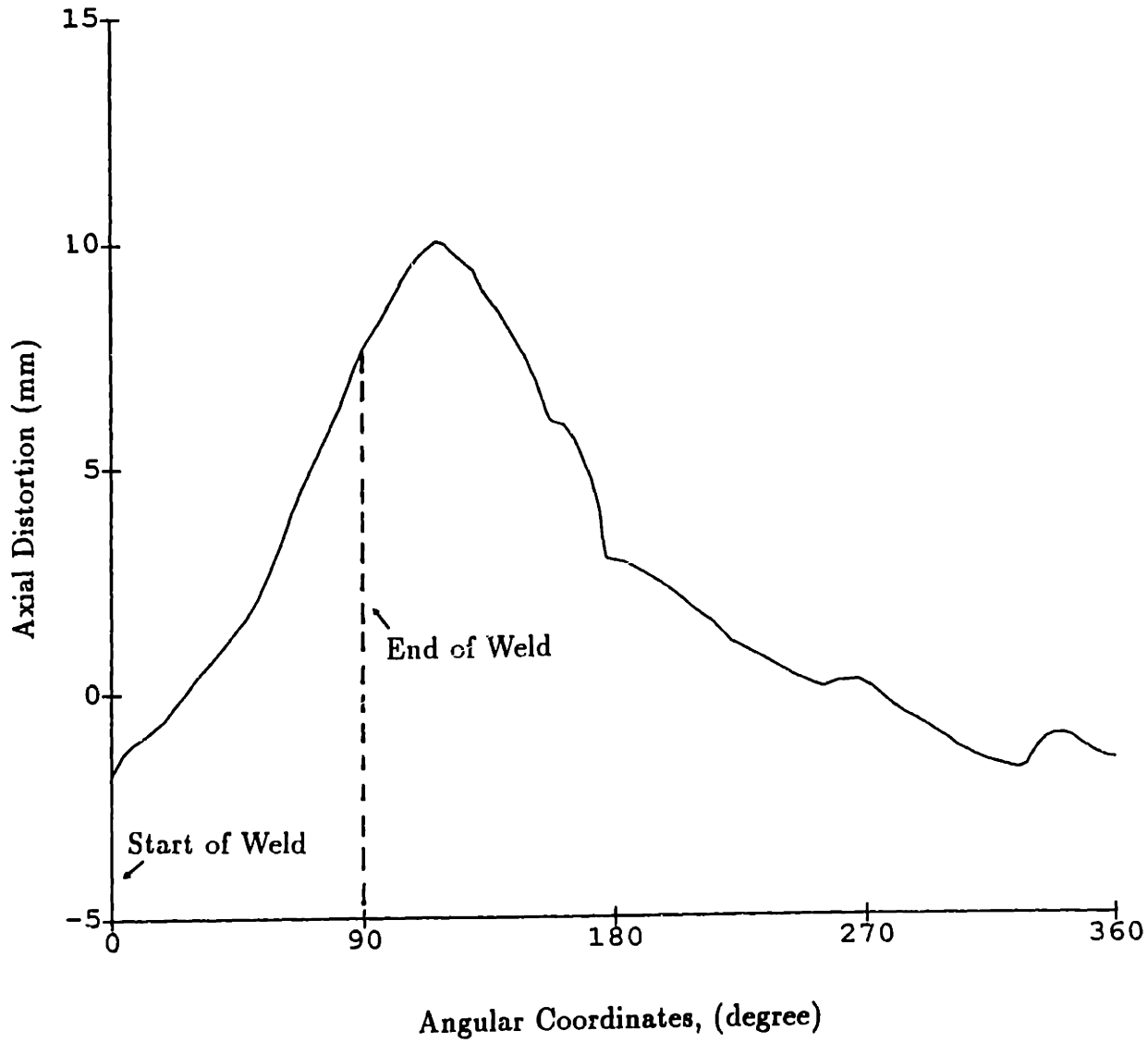


Figure 3.11: The axial distortion of the web measured after 90 degree weld was completed

accomplishing a more accurate simulation of the thermal analysis while still employing an uncoupled analysis.

Figure 3.12 shows the predicted final shape of the structure superimposed with the range of distortion encountered in the experiment. The model duplicates the primary, axial deformation of the web that results from the thermoelastic expansion of the elastic portion of the web. The counterclockwise rotation of the web towards the flange due to cooling of the weld material is also predicted, although the initial heat-induced clockwise rotation of the web is much larger in the finite element simulation.

The overestimation of the rotation of the web during the heating segment of the simulation results from two limitations of the finite element analysis. First, friction between the actual web and flange constrains their relative motion. Second, and perhaps more importantly, the real three-dimensional structure provides an out-of-plane elastic constraint absent in the two-dimensional analysis. The implicit, two-dimensional assumption of simultaneous, circumferential heating ignores the character of electrode motion. Material behind and ahead of a moving electrode constrains distortion at the electrode. Accurate simulations of the complete features of welding distortions therefore require a full three-dimensional analysis.

3.7 Elastic Coupling

The effect of the coupling of the larger elastic structure to the weld zone appears to be underappreciated in the welding literature. The examples of fitup or weld gap clearance and fixture variation which are discussed above illustrate the significant effect of thermal

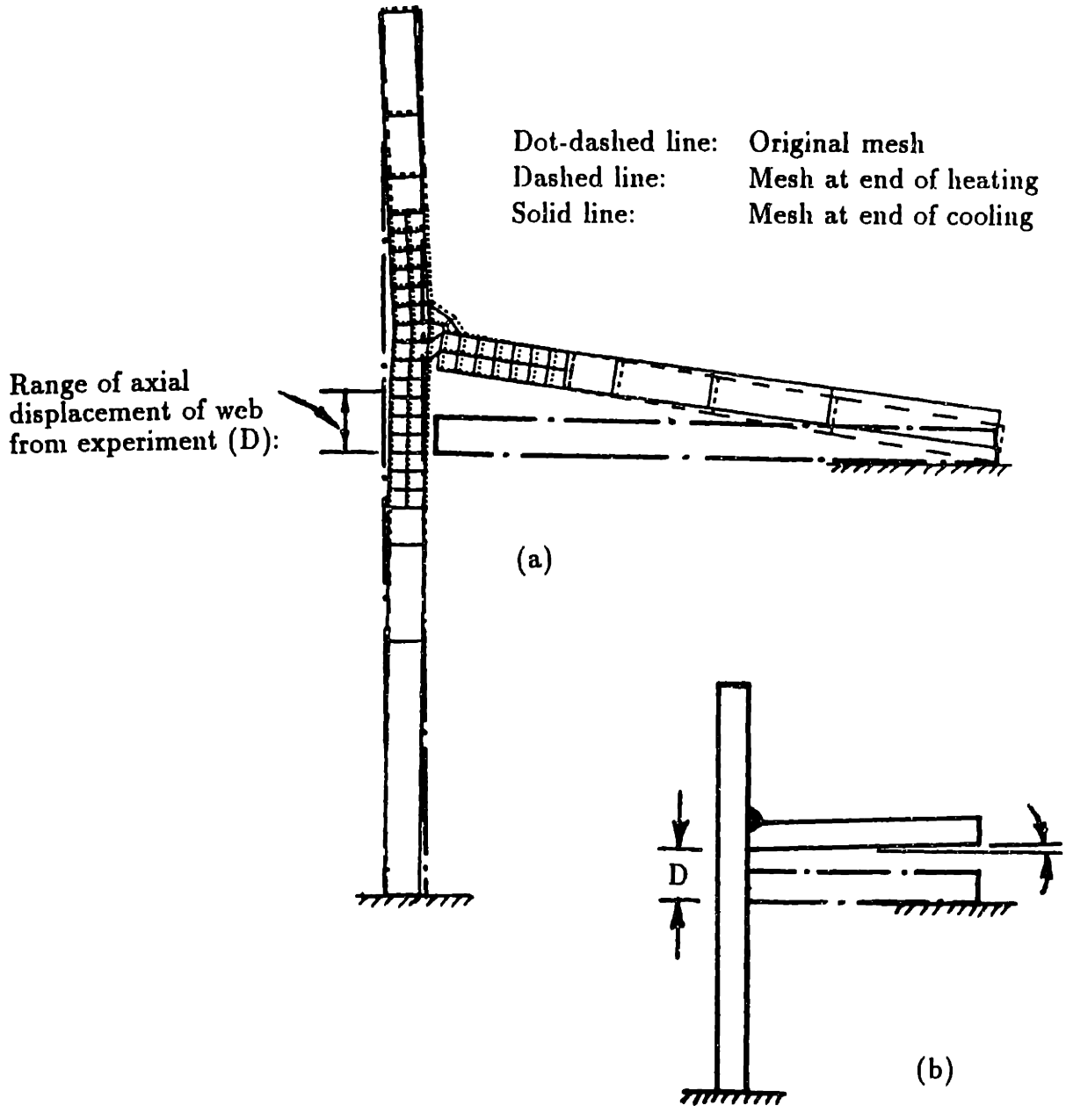


Figure 3.12: (a). Predicted distortion of web and flange using axisymmetric model, (b). Schematic drawing of measured distortion after completion of 90 degree weld.

distortion outside the weld zone. The contribution of the elastic structure may indeed dominate the state of distortion and stress in the final welded structure, producing a final state which is much different from one predicted by simulation of the weld zone alone. The implication of such results is that a reasonable simulation of welding must include a representation of the structure away from the weld.

A simple-minded, yet revealing example of structure/weld interaction is provided in the evaluation of the same two dimensional mesh illustrated in Figure 3.2 as a plane strain problem. The temperature field used in the axisymmetric simulation is used as the loading of the plane strain problem. The model therefore consists of the welding of a T-joint using two plates. One side of the weld region is heated, hot weld material is deposited at the end of the heating process, then the "T" structure is cooled. Boundary conditions are applied to prevent rigid body motion. This model therefore is identical to the axisymmetric one with the exception of the character of the far field structure.

Figure 3.13 illustrates the maximum principal stress distribution in the plane strain analysis. The highest principal stresses is much lower than that in the axisymmetric case, especially in the weld region. Distortion after the first weld is also much less noticeable in this plane strain model.

The representation of the complete welded structure is necessary to reproduce several phenomena. First, the elastic boundary conditions on the weld zone represented by the larger elastic structure may be much different from a simple clamped or pinned boundary. Deformation within the weld zone depends heavily upon these boundary conditions and therefore an accurate accounting of these boundary conditions is required. Second,

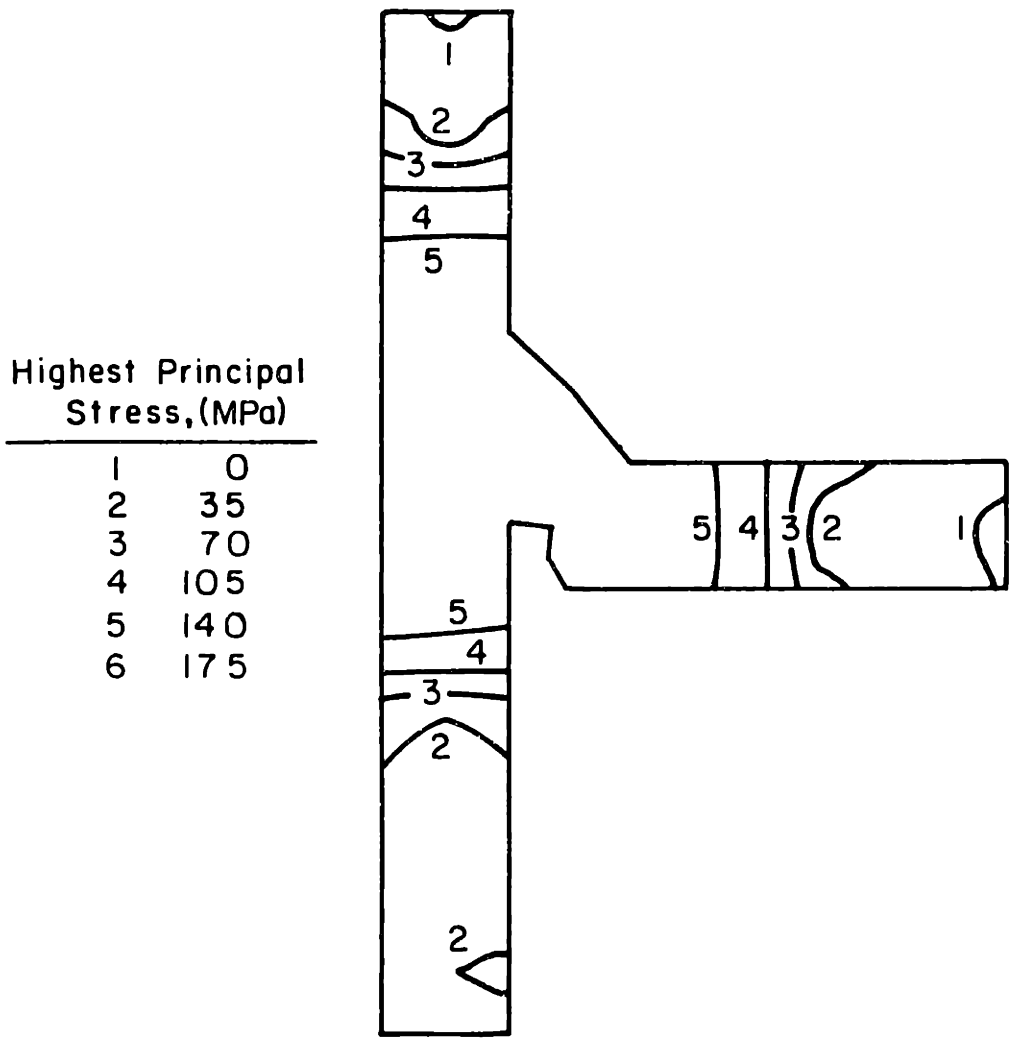


Figure 3.13: Residual highest principal stress contour after first weld. Plane strain model and local weld region.

the thermal *elastic* distortion of the structure near the weld can dominate the final state of residual stress and distortion in the welded structure. Inclusion of the elastic structure in the thermal and mechanical analyses is therefore needed to include the potentially controlling effect of these thermal distortions. Third, simple boundary conditions such as location and type of fixtures and weld gap variation may dramatically affect the final state of the weld. Evaluation of the true effect of these boundary conditions therefore requires representation of the entire structure.

Our evaluation of the two dimensional approximation to the full three dimensional problem suggests that complete representation of a structure to be welded requires a full three dimensional model. Motion of a welding electrode in a three dimensional structure continually alters the elastic coupling between the components being welded, thereby affecting subsequent distortion. The two dimensional model used here to illustrate the effect of structure/weld coupling accentuates distortions in the plane of the model since welding is assumed to occur around the entire circumference. This distortion should be reduced in the actual three dimensional structure since the electrode more closely approximates a point source and the structure ahead of and behind the electrode will constrain motion. The three-dimensional model predicts the variation in weld gap clearance as the electrode moves, not possible given a two dimensional model.

3.8 The Applicability of 2-D Models

Two dimensional idealizations have been accurate in predicting temperature fields and overall distortion in spite of their inherent inability to duplicate a full three dimensional

response. At the limit, if the electrode velocity is sufficiently high relative to the rate of material cooling, the welding process may approximate a simultaneous weld along a large portion of the weld path. In this case a two dimensional analysis is adequate. A two dimensional analysis is also justified where true symmetry exists, such as axisymmetric spot welding.

Criteria can be established to indicate whether a two-dimensional analysis (plane strain normal to the weld path) may be adequate. One possible criterion is to compare the time needed for the welding process with the time elapsed during the cooling of the heated metal. For example, if we define:

$$\Delta t_{weld} = L/v \tag{3.1}$$

$$\Delta t_{cool} \sim t_{8/5} \tag{3.2}$$

where L is the length of the weld path, v the velocity of the moving electrode, and $\Delta t_{8/5}$ the time elapsed when temperature drops from $800°C$ to $500°C$ [1]. If $\Delta t_{weld} \ll \Delta t_{cool}$, the welding may be considered to occur simultaneously along its entire length. Otherwise, the coupling (constraints to deformation) supplied by other parts of the structure may change with the electrode motion. Such structural change can not be accommodated merely by a two-dimensional model, necessitating a three-dimensional simulation.

A simple example of this criterion is given here to indicate how well a practical welding process may be analyzed by a two-dimensional model. In plate welding, this criterion suggests:

[1] This parameter is a characteristic indication of cooling rates, and is used in heat treatment designs[22, 41]

$$\Delta t_{weld} = \frac{L}{v} \ll \Delta t_{cool} \tag{3.3}$$

or

$$v \gg \frac{L}{\Delta t_{cool}} \sim \frac{L}{\Delta t_{8/5}} \tag{3.4}$$

Assuming a value of $\Delta t_{8/5}$ to be 20 seconds and L to be 1 meter, then:

$$v \gg 0.05 \text{ m/sec. or } 2 \text{ in/sec.} \tag{3.5}$$

This is substantially greater than the electrode velocity in a practical welding situation. The implication is that full representation of welding distortion and stress/strain requires a three-dimensional analysis.

Chapter 4

3-D Simulation of a Short Weld on the Ring Stiffened Cylinder

Three-dimensional analyses have also been performed of the ring stiffened cylinder, as shown in Figure 4.1. Only a short weld pass of 25.5 mm long (an arc of 3.2 degrees) is deposited in this three-dimensional model to keep the problem tractable with our computational resources.

4.1 The 3-D Model

A denser mesh is used in the weld region to capture the high local gradients, while a coarser mesh consisting of shell elements represents the rest of the structure away from the weld. We assume that the far field is isothermal and elastic during the process and use a substructure to represent its effect on the local zone. Such a model, as depicted in Figure 4.1, consists of approximately 1,100 nodes for the local weld zone and 2,500 nodes for the substructure; the corresponding total numbers of degrees of freedom for the deformation analyses are about 4,000 and 12,000, respectively.

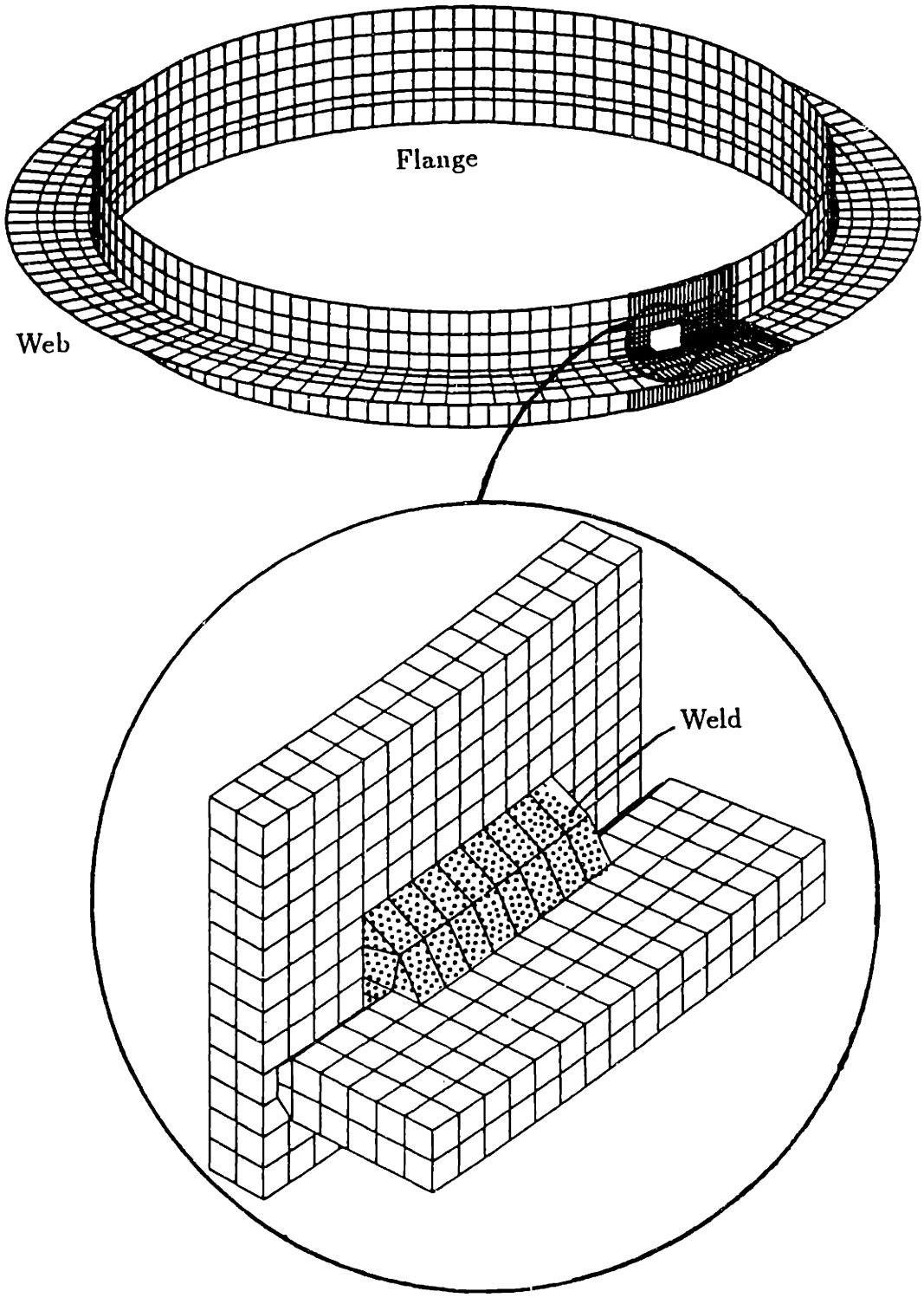


Figure 4.1: 3-D finite element mesh of a ring stiffened cylinder. Magnified portion of model is used in full analysis. Remainder of model is reduced to elastic substructures.

As in the two-dimensional analyses, decoupled thermal and mechanical analyses are also employed. The structure is assumed to be initially at room temperature of 20 °C and free of stresses and strains. Deposition of weld material is simulated by activating elements at melting temperature when the electrode passes the area where the elements are located. Boundary conditions are described separately in Sections 4.2 and 4.3.

4.2 Heat Transfer Analysis

Heat transfer during welding is a process of heat flux into the structure around the electrode, addition of extra weld material at melting temperature, fast heat conduction to the rest of the structure, and eventual heat release through boundary convection and radiation. The heat flux pattern strongly affects temperature distribution in the local weld zone.

Heat flux from an electrode to a flat plate is commonly assumed to be distributed as a two dimensional Gaussian function[29]. A three-dimensional double ellipsoidal heat flux model, proposed by Goldak et al.[42], can capture the flux penetration effectively, which is important for high energy intensity welding processes such as laser beam and electron beam welding. The Gaussian function is used here since arc welding involves less flux penetration. The T-shaped joint in the ring stiffened cylinder requires modification of this distribution to produce reasonable temperature fields. The total power is divided into two components, one to the flange and the other to the web. Only part of the Gaussian function is used for both of these two components. If the coordinates of the electrode is x_0, y_0, and z_0 at time t, the heat flux to the web depends on the location on its upper surface:

$$q = pe^{-\sigma[(x-x_0)^2+(y-y_0)^2]} \tag{4.1}$$

where p and σ represents the net power input and the width of the Gaussian curve. They are adjusted so that melting occurs and a reasonable temperature field is obtained before the deposition of any weld material.

Heat flux on the outer surface of the cylinder can be expressed as follows:

$$q = pe^{-\sigma r^2} \tag{4.2}$$

in which the radius r from the center of electrode to a point is defined as:

$$r^2 = (x^2 + y^2)\left[\tan^{-1}\frac{\sqrt{(x-x_0)^2+(y-y_0)^2}}{\sqrt{x^2+y^2}}\right]^2 + (z-z_0)^2 \tag{4.3}$$

The first term represents the square of the distance on the circumferential direction of the outer surface of the cylinder; it is very close to the chord between the spatial point and the center of the electrode.

The deposition of weld material is simulated by the activation of elements at melting temperature. Temperatures around the weld groove are either higher than or close to the melting temperature at the moment before the weld element is activated. This prevents extreme temperature gradients at the boundary of the weld elements and speeds up the convergence of the analyses.

Boundary conditions are modeled by heat emissivity coefficients and nonlinear heat convection factors for different surfaces of the structure (vertical, upward, and downward), as discussed in detail in Chapter 2.

Figure 4.2 shows the temperature distribution within the local weld region at the time when the last set of weld elements have been activated; the plotted areas cover the outside

surface of the cylinder and the top surface of the web. Heat conduction from the local zone to rest of the structure can be clearly observed. Temperatures outside the local region remain below 300 °C.

The temperature contours indicate that conductive heat transfer dominates the thermal process when weld material is deposited, while boundary heat loss through convection and radiation only becomes dominant in the longer period during cooling. The temperature gradients (or density of the contour levels in the plots) and changing rate ($\frac{\partial T}{\partial t}$) will have significant importance in the following deformation analyses since temperature gradients and changes are the only loading on the structure.

4.3 Deformation Analyses

The deformation process is assumed to be rate-independent. A thermo-elastic-plastic constitutive model is used in this three-dimensional analysis where all the material properties are assumed temperature-dependent, including Young's modulus, yield stress, isotropic strain hardening, and linear expansion coefficient.

Boundary conditions on the three-dimensional model are applied to simulate an actual weld experiment conducted earlier (see Section 3.6); both the cylinder and the web sit on four metal blocks on a rotating table while a number of clamps are applied to constrain rigid body motions of the structure. As is shown in Figure 4.3, the four supports are equally spaced on the circumference. The model is constrained at two locations on both the cylinder and the web by removing the degrees of freedom of the corresponding nodes to prevent rigid body motions. The supports are modeled by nonlinear springs which have very

Temperature	($°C$)
1	300
2	900
3	1500
4	2100
5	2400

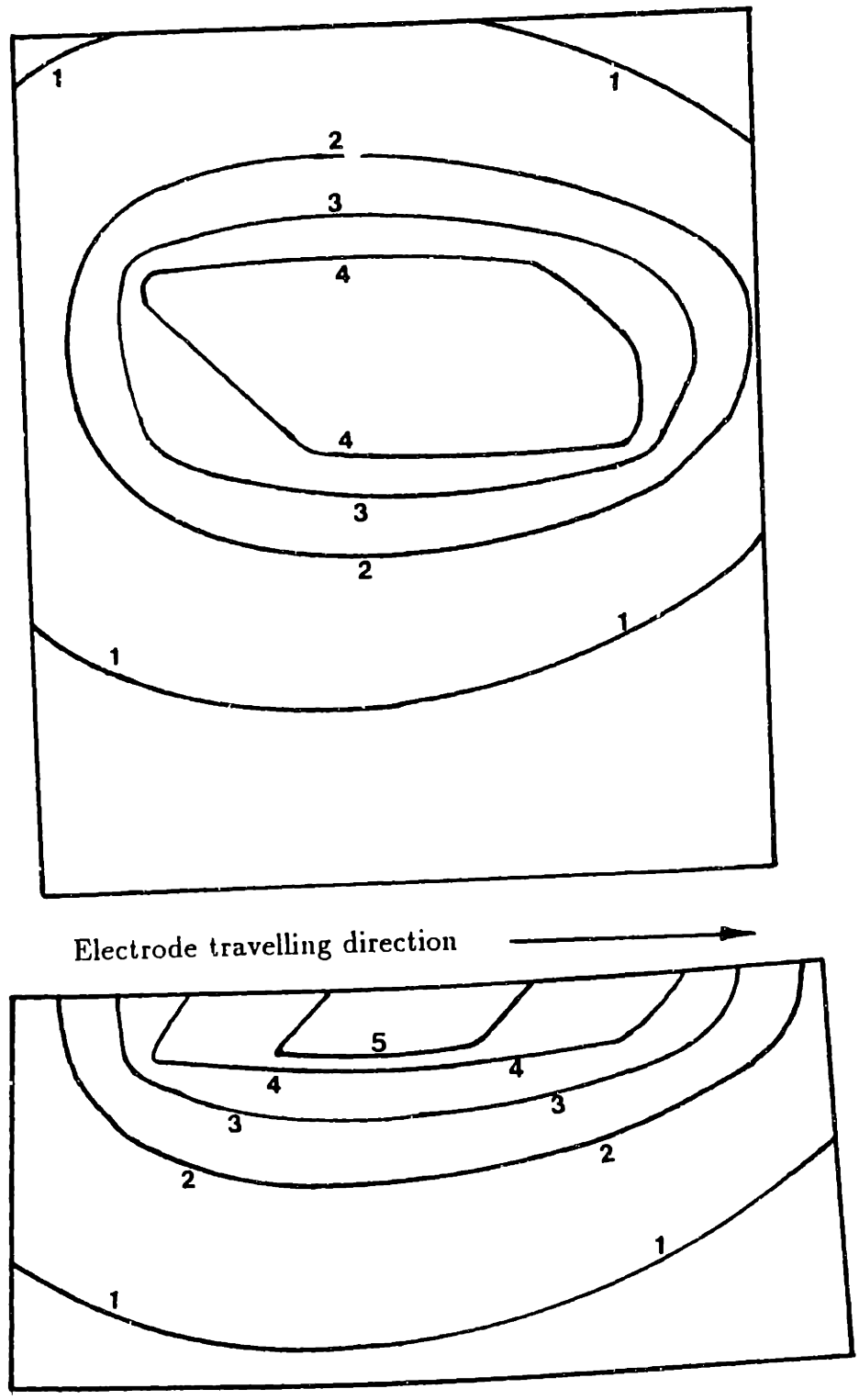


Figure 4.2: The Temperature contours on top of the web and outside of the flange in the weld region, at the time when the last weld element has been deposited.

low stiffnesses for upward displacements and high stiffnesses for downward displacements; typical values for upward and downward spring stiffnesses are respectively 5,000 N/m (25 lb/in) and 1.0×10^{12} N/m (5.0×10^{9} lb/in). The downward stiffness is much larger than that of a typical element in the system stiffness matrix. To simulate the presence of the supports throughout the welding process, the springs are kept active until after the structure cools down, and are then removed at the end of the simulation. Some of the initial constraints on the web only remove rigid body motions and are removed as soon as the web is welded to the cylinder; these constraints are therefore modeled by linear stiff springs (stiffness taken as 1.0×10^{12} N/m). Other combinations of constraints are also simulated, and the resulting distortion patterns are compared in Section 4.4.

The final distorted shape of the ring stiffened cylinder is shown in Figure 4.4. The constraints and supports control the deformation of the overall structure during welding. As soon as they are removed, the structure distorts due to the release of elastic energy accumulated during the process and stored inside the structure, resulting in an elastic relaxation. We will discuss the deformation mechanism in the following section. The weld elements activated during the simulation are defined according to the initial geometry; they are forced to deform at activation since the weld groove alters its shape due to heat expansion. The final shape of the weld elements, as shown in Figure 4.4, therefore may not represent the actual shape of a real weld. One solution to this problem is to redefine the weld elements before the activation. This requires an interruption of the analysis each time one element or one set of elements is activated. Many elements are normally activated at different times in a simulation; such definitions of weld elements require large amount of manual work and become impossible when the number of weld elements increase substan-

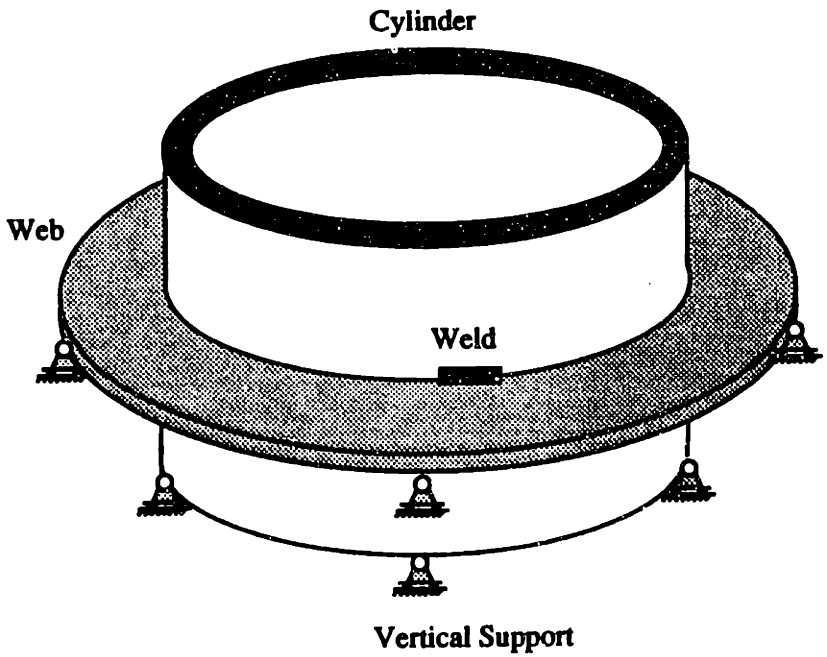


Figure 4.3: The boundary conditions on the 3-D model simulating the sitting on four supports.

tially. Fortunately the weld material is soft at melting temperature and only accounts for a fraction of the total heat affected area at lower temperatures. The effect of different sizes of weld elements is negligible under such circumstances. Residual stresses in and around the weld are beyond the initial yield stress of the material, indicating that plastic deformation occurs during the process.

The distortion of the whole structure is asymmetric and consists of both the tilting and twisting of the web and cylinder. The asymmetry of such a distortion illustrates the limitations of two-dimensional efforts to model welding distortion and/or residual stresses. Three-dimensional analyses for large structures are therefore required to simulate welding processes.

The transient and final distortions depend on many factors during the welding process. The geometry of a structure is a dominant factor affecting the final distortion pattern; among the other important factors are boundary conditions or fixture setups and weld parameters such as power input, traveling velocity of the electrode, rate of material deposition, and local geometries of weld grooves. The following section concentrates on investigations of the effects on distortion by different boundary conditions.

4.4 Fixturing Effects

Fixtures are very important in welding practice. A minimal number of fixtures are required in all welding situations to remove the free movements of workpieces; more fixtures are needed in controlling and reducing welding distortions. Intuition indicates that the more fixtures applied the less distortion will result. True or not, such a conclusion is investigated

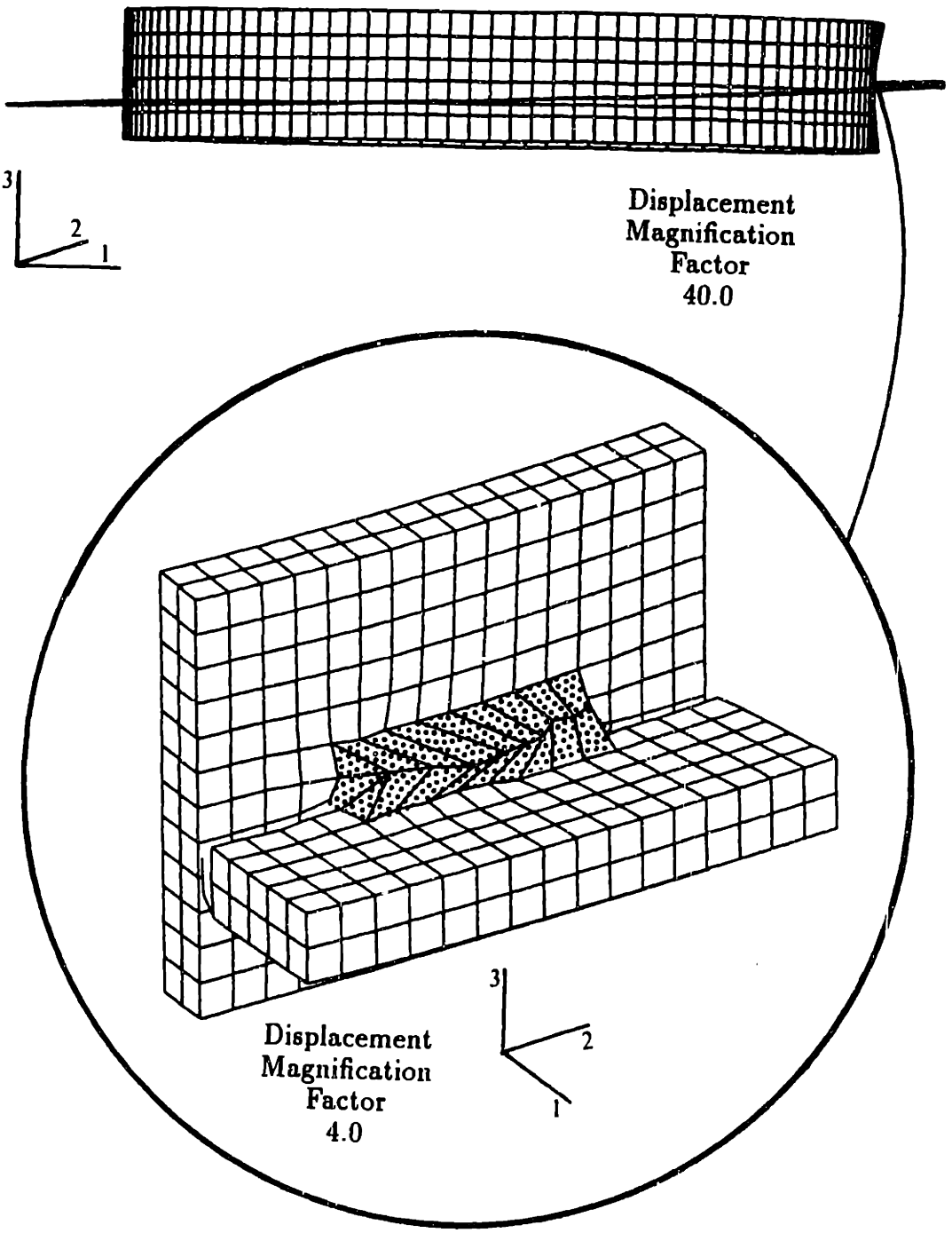


Figure 4.4: The distorted shape of the ring stiffened cylinder after a weld of 25.5 mm has been laid.

in the following simulations.

Three different fixturing schemes are applied to the finite element model and the resulting distortions are then examined. These three cases have different levels of constraint, nevertheless, all represent possible fixturing setups in an actual welding situation. Case I simulates the minimal constraints on the structure sitting on a flat surface at four locations, only the translational degrees of freedom are removed at the minimal number of nodes, as shown in Figure 4.5(a). The supports are represented by nonlinear springs described earlier. In case II, both the cylinder and the web are clamped at four locations as show in Figure 4.5(b). Clamping here refers to the removal of translational degrees of freedom while retaining rotational degrees of freedom. This fixture setup represents a stiffer fixture setup than the previous one and could be approximately accomplished by using tack welds or physical clamps at these locations. Case III simulates another possible fixture setup, where the web and cylinder are clamped at two locations near the weld (behind and ahead of the weld at an angle of 19.2 degree) as is shown in Figure 4.5(c). The structure is also supported at the other locations as in case I. Such a fixturing strategy will, according to our intuition, limit the local distortion around the weld.

The boundary conditions on the web for case I are only to remove rigid body motions and are therefore removed as soon as two layers of weld elements are activated. At this time the cylinder and the web are connected to each other and the boundary conditions on the cylinder will serve the purpose of removing rigid body motions. The springs representing the supports are active until the end of the cooling. At the end of cooling, extra boundary conditions and springs are removed except those necessary to keep the structure free of any rigid body motion, as explained in the captions of Figure 4.5. These remaining constraints

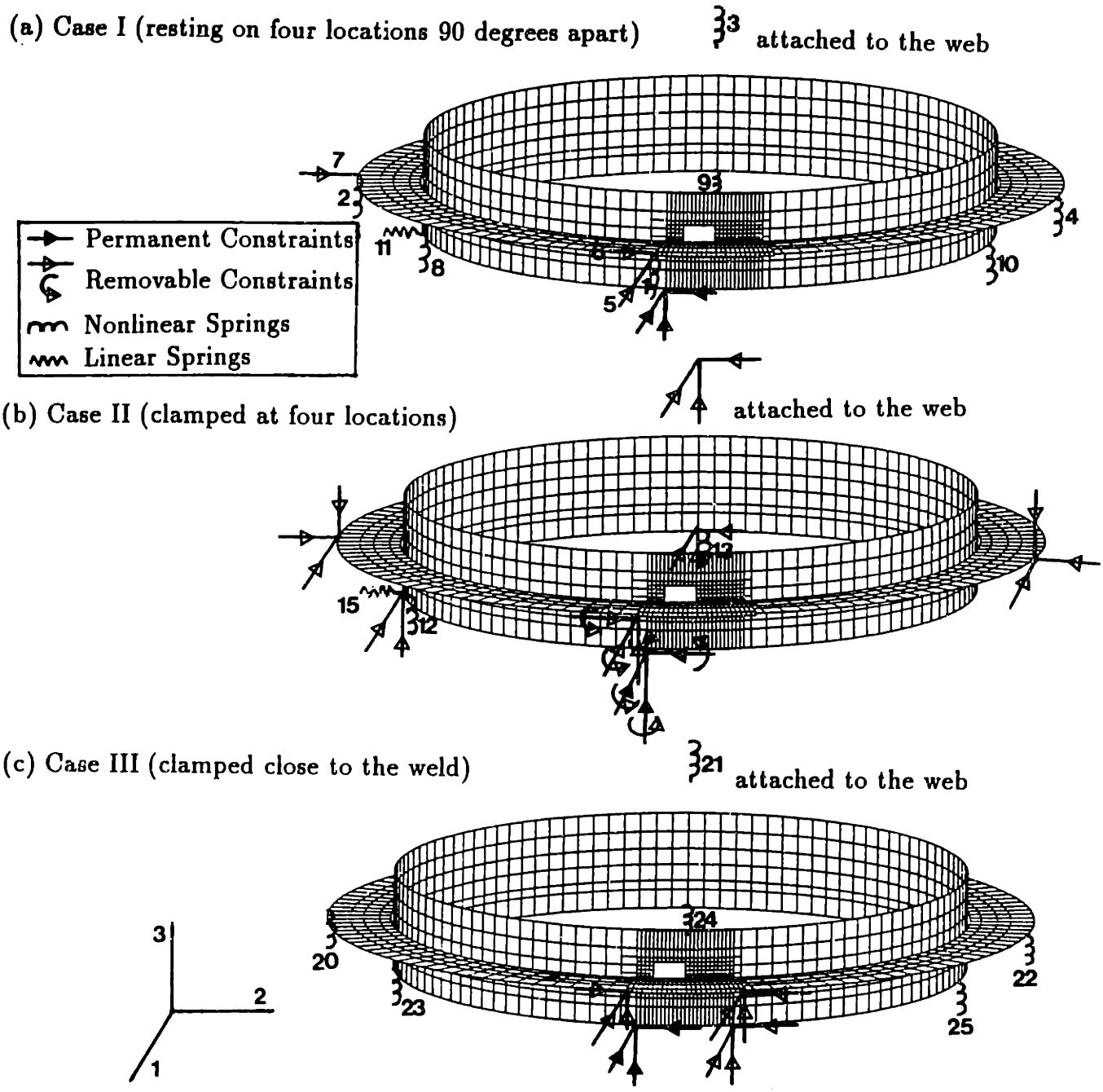


Figure 4.5: Fixture setups of three different cases. (a). Case I, boundary conditions 5, 6, and 7 initially prevent rigid body motions of the web. These boundary conditions and springs 1, 2, 3, 4, and 10 are removed at the end of the analysis. (b). Case II, boundary conditions are active until their removal at the end of the analysis except 15 which is required to prevent rigid body motions. Springs 12 and 13 will become active after the boundary conditions are removed at their corresponding nodes. (c). Case III, boundary conditions and springs are active until their removal at the end of the analysis. Springs 23 and 24 are never removed.

constitute a statically determinant constraint system. Since there are no external forces, the reaction forces by these constraints are zero, representing the situation when the structure is finally removed from the supporting table.

At the end of cooling, when all the supports are still present, the distortions are as shown in Figure 4.6. To our surprise, the structure distorts more in case II and III than in case I. The cylinder twists in case II and bows upward in case III. Such distortion can be explained from the different setups of fixtures as follows. In case II, the translational degrees of freedom are removed at four locations for both the cylinder and the web, while rotational distortions are allowed. The distortion shows zero displacements at the four locations where constraints are present. Twisting of the cylinder accommodates local expansion or contraction around the weld yet still meets the 'pinned' boundary conditions at the four locations. In case III, the constraints behind and ahead of the weld limit the local distortion. Note that the constraints on the cylinder are located at its bottom; while translational displacements are removed, the rotational displacements will occur when the local weld expands or contracts. The upward bowing of the cylinder is a result of the rotational displacements around the fixtures at the bottom of the cylinder. The upward displacement of the middle section of the cylinder is allowed by the nonlinear springs. Case I shows a different distortion pattern from our expectation, since more fixtures are supposed to limit distortions.

Figure 4.7 shows the distortion patterns after all redundant boundary conditions and springs are removed, leaving the structure in a statically determinant state. The distortions clearly differ among the three cases because of the different fixture setups.

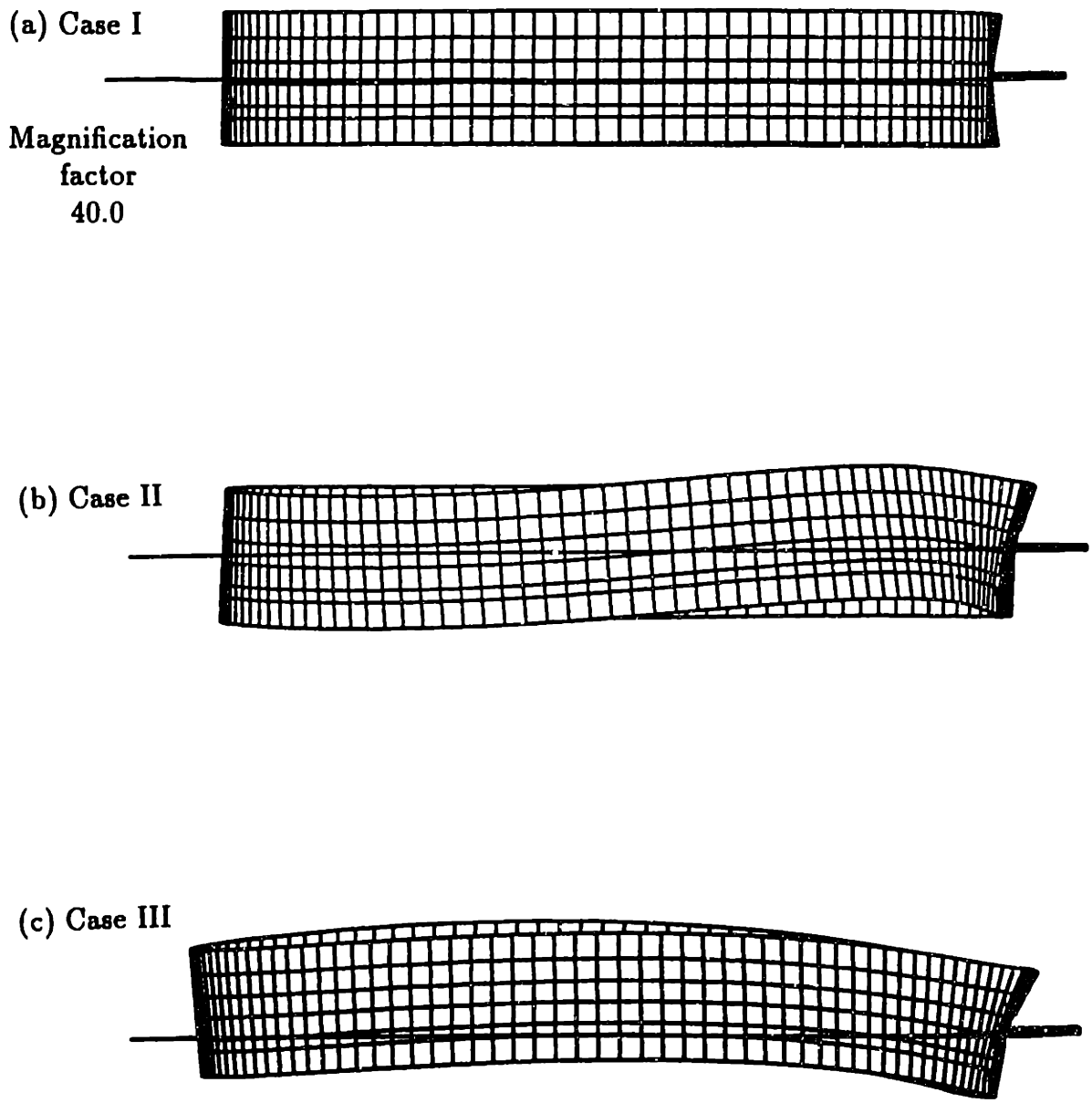


Figure 4.6: Distortions at the end of cooling, before removal of constraints. (a). Case I. (b). Case II. (c). Case III

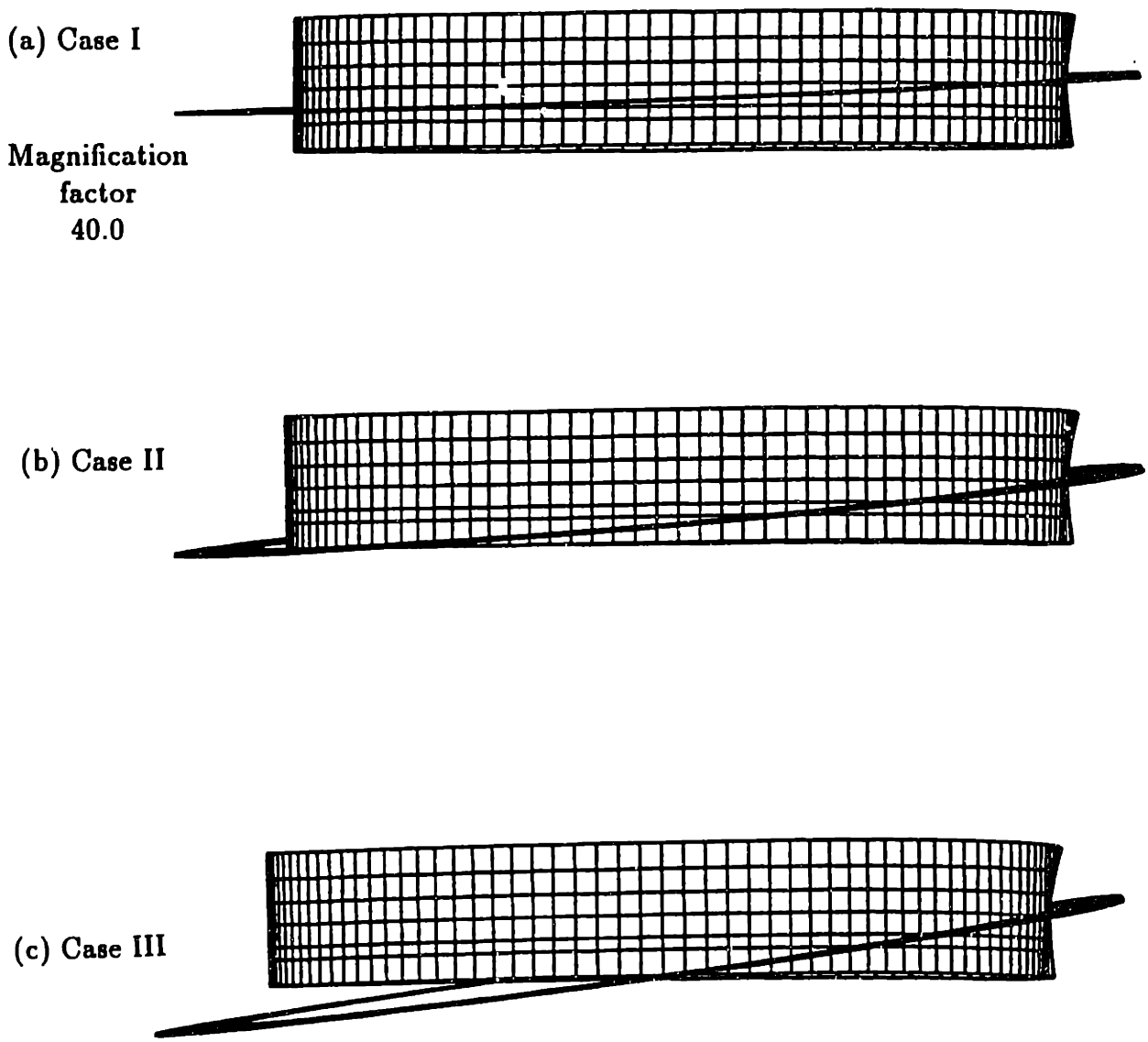


Figure 4.7: Distortions at the end of the analyses when removable constraints are deactivated. (a). Case I. (b). Case II. (c). Case III

In case I, the cylinder twists after the removal of supports, while the web tilts. There is no clear twisting of the web which can be observed. In case II, however, the original twist of the cylinder at the end of cooling recovered leaving no visible twist. The web tilts more than twice as much as in case I. Much of the original twisting and bowing of the cylinder in case III also disappears after the removal of constraints, while the web tilts much more than it does in the previous two cases. The cylinder's final amounts of twisting in cases I and III are small and basically the same. The major differences in the overall distortion patterns among the three cases of fixture setups are the magnitudes of the web's tilts. The distorted shapes of the local weld zones, though not depicted in the figures, are also different from each other.

These results were not obvious before the simulations were performed. Conventional reasoning of welding distortion, based on the assumption that weld contraction causes distortion, cannot explain these distortions. There must be some other factors affecting the deformation both during and after the welding process. As we can see from these different fixture setups, fixturing obviously plays an important role. When the local zone is heated, extremely high temperatures and temperature gradients cause the local zone to deform severely. The constraints supplied by fixtures will both affect the plastic deformation of the local zone and result in elastic deformation in the far field, as is shown in Figure 4.6. The three cases result in different local deformation around the welds. The more constraints supplied by fixtures, the more complicated and more severe deformation occur both near the welds and in the overall structure. The structure is forced to deform in a way to accommodate both the local expansion/contraction and the fixture constraints. When the weld material is deposited, however, such deformation is "locked in" by the new

weld, and much of such deformation is then retained after the removal of fixtures. Such "locking" of deformation by the new weld plays a more important part in the large structure deformation than has been appreciated. We will discuss this mechanism in Section 4.6.

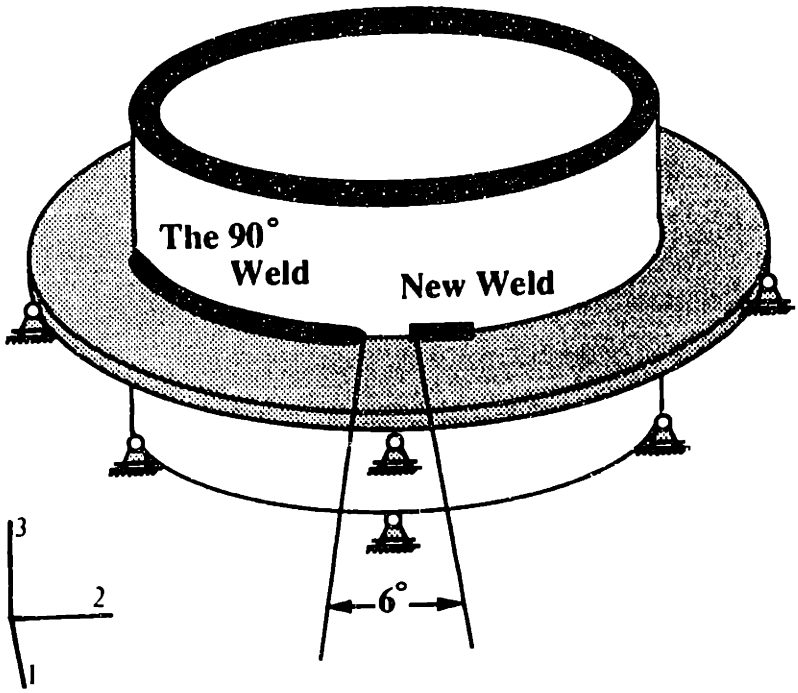
Another configuration of the structure is also considered in the simulations, where the cylinder and the web are welded 90 degrees behind the new, short weld. Fixture setup is similar to those of Case I; both the cylinder and the web sit on top of four supports. Figure 4.8 shows the relative locations of the 90 degree weld and the new weld, and the final distortion of the structure after the new weld. Figure 4.9 is a plot of the distortion of the top surface of the web around the cylinder-web interface.

The magnitude of distortion in this case is more than that of Case I; the 90 degree weld does not reduce distortion. The distortion pattern is also complicated. This result further supports the conclusion that welding distortion of a large structure is complicated and non-intuitive. Numerical simulations are necessary in predicting the final distorted shape of a large structure after welding.

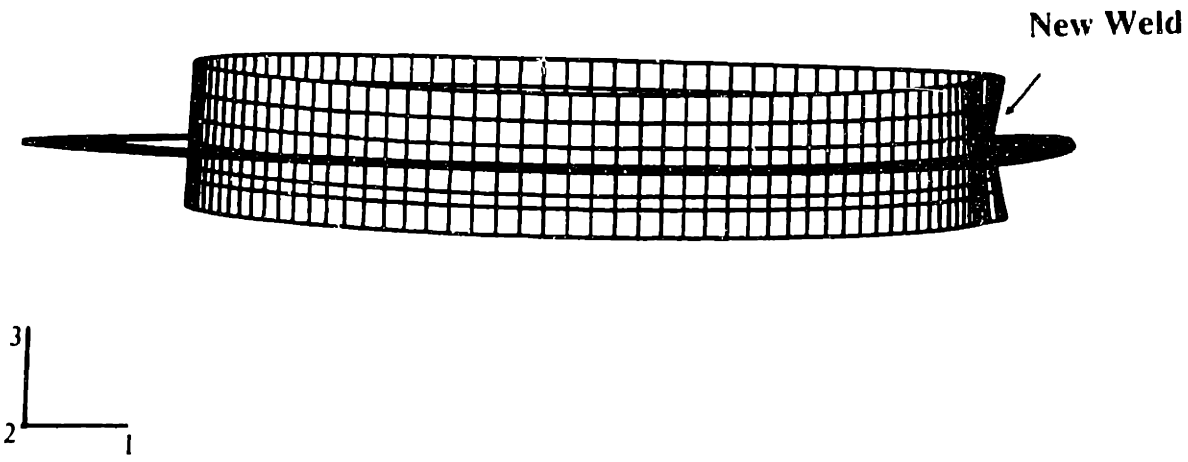
The following conclusions can be drawn from these results: Three-dimensional simulations can predict asymmetric deformation; the fixture setup is very important in welding practice, and the final distortions can not be simply predicted from intuition.

4.5 Experimental Correlations

A welding experiment was performed to compare with the conclusions of the numerical simulations. Similar to the numerical model, a cylinder and web stiffener was used with a one-inch weld deposited along the circumference of the cylinder-web interface. Two sets of



(a) The 90 degree weld and the new weld.



(b) The final distortion after the new weld.

Figure 4.8: The welding of one-inch weld after 90-degree are welded

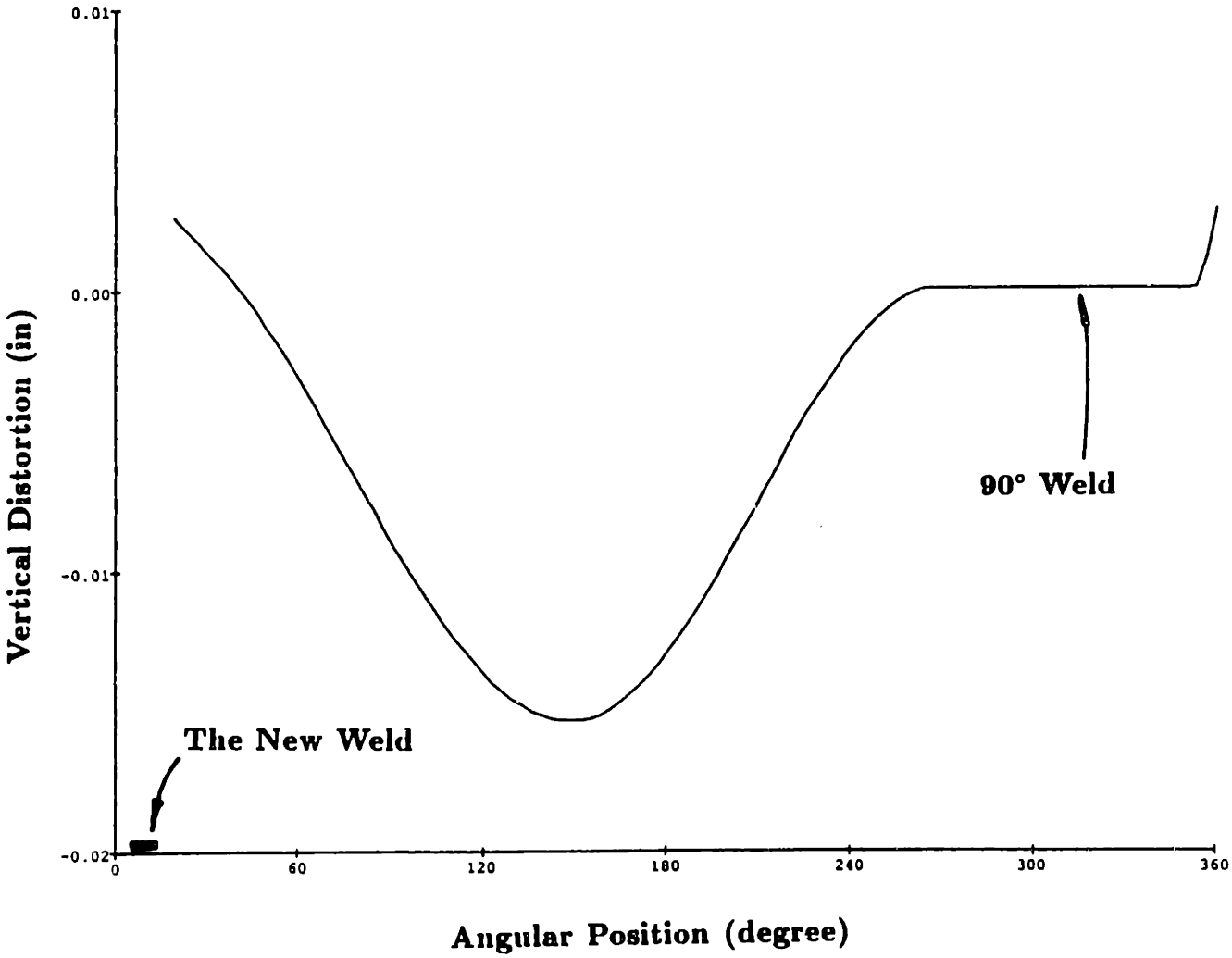


Figure 4.9: Vertical distortion of the top surface of the web around the cylinder-web interface

fixture setups were used; in the first case (Case I), both the cylinder and the web sat on four supports on top of a table, and in the second (Case III in the numerical model), the cylinder and the web were tack welded at four points around the local weld zone, similar to the numerical models.

The fitup gap between the cylinder and the web is about 3/16 of an inch, so that there is no contact between the cylinder and the web before the weld deposition, and the side of the web opposite to the weld can move freely up and down. The power supply used in the welding was 26 Volts at 140 Amperes, which resulted in a gross power of 3.64 KW. The electrode speed was controlled by an experienced welder so that the total time elapsed for the weld was about 10 seconds, approximately the same used in the numerical analyses. The size of the weld was also controlled to match that of the model.

4.5.1 Measurement of Distortions

A depth gauge was used to measure the distance between the top of the cylinder and the top of the web at 36 locations around the circumference of the cylinder-web interface. Since the web is not perfectly flat before the welding, we measured these distances before and after the welding process and calculated the differences to obtain the distortion.

Gravity was not modeled in the numerical simulations. In the actual welding, there are two stages where gravity may influence the distortion pattern. The first stage occurs during the welding and the second stage occurs after the welding but during the measurement of the distortions. In the first stage, the gravity effect is not significant for the following reasons: 1), when the local zone is hot, the weld metal has a low flow strength; the local

thermal expansion or contraction will therefore not cause much displacement far from the weld. Gravity only assists the structure to sit on the supports. 2), when the weld cools and contracts, there will be a tendency of downward displacement at the other section of the ring opposite the weld; but such a tendency was constrained by the supports. Gravity again assists the structure to sit on the supports and does not affect the deformation pattern.

After the welding process and the removal of fixtures, gravity does play a role in the distortion of the structure. If the cylinder is oriented horizontally while the web hangs freely, the weight of the web will generate a bending moment on the weld and the web will thus rotate, since the other side of web and cylinder opposite the weld is not connected. To minimize the gravity effect, the welded cylinder-web structure was oriented vertically, and effect of gravity was then calibrated and corrected.

4.5.2 Correction of Gravity Effect

The gravity effect on the distortion is first calibrated and then corrected in the distortion measurements. As is schematically shown in Figure 4.10(b), gravity generates a bending moment on the weld and exaggerates the distortion. Two assumptions for the calibrations were considered: 1), the deformation caused by gravity on the weld is elastic, and 2), the web is macroscopically flat. The first assumption can be verified by imposing different moments about the weld which result in proportional distortion changes (these calibration moments are greater than the moment caused by gravity). The second assumption is obviously true since out-of-plane distortion of the web is very small compared to the radius of the ring.

The structure is placed vertically as shown in Figure 4.10 with one side of the cylinder vertical. The definitions of variables used in the following equations are shown in the figure. Note that only Y, Z, and W are actually measured.

Suppose the rotational stiffness of the weld is K, gravity will then be balanced by the reaction moment supplied by the weld:

$$G\frac{(Y-W)}{2} = K\Delta\theta_1 \tag{4.4}$$

Where G is the gravity of the web and $\Delta\theta_1$ the rotation caused by gravity, and

$$\Delta\theta_1 = (\theta_1 - \theta) = \frac{(Y-X)}{D} \tag{4.5}$$

If a known calibration force F is added to the web as shown in Figure 4.10(c), the following holds by equilibrium:

$$G\frac{(Z-W)}{2} - F \cdot D_1 = K(\theta_2 - \theta) = K\frac{(Z-X)}{D} \tag{4.6}$$

Substituting Eq. 4.5 into Eq. 4.4 and then subtracting by Eq. 4.6 yields:

$$G\frac{(Y-Z)}{2} + F \cdot D_1 = K\frac{(Y-Z)}{D} \tag{4.7}$$

The rotational stiffness of the weld can then be derived as:

$$K = \frac{G}{2D} + \frac{D_1}{D}\frac{F}{Y-Z} \tag{4.8}$$

and the distortion change on top of the web due to gravity can be further derived from Eqs. 4.4 and 4.5:

$$Y - X = \frac{D}{2K}(Y - W)G \tag{4.9}$$

This value can then be used to correct the gravity effect in the measured data. The actual tests indicate that the rotational stiffness, K, of the weld is approximately 5,300 lb-in, and the distortion caused by gravity is only about 7% of the total distortion.

4.5.3 Comparison of Predicted and Measured Distortions

Figure 4.11 shows the distortions in the case of locally constrained welding (Case III). The general trends of distortion match each other closely. The discrepancy in the exact magnitude of distortion can be attributed to many factors: the differences between the model and the actual welding in power input, electrode speed, and size of the weld as well as the idealization of fixtures in the model.

Figure 4.12 shows the distortion patterns in the case when the structure sits on four supports on top of a table. Note that the magnitude of distortion is much less (about 50%) than that in Case III, as predicted in the numerical modeling. The only difference between these two cases is the fixture setup. The upward distortion between 0 and 90 degrees is due to a transverse rotation which the numerical model did not simulate; the predicted distortion resembles the experimental results after this rotation is removed. This transverse rotation is most likely due to the fact that the weld and the closest fixtures are not aligned radially. When the weld cools, the contraction of the weld results in a bending moment around the clamp on the web and causes a transverse rotation pivoted around the clamp. More effort is still needed to fine-tune the discrepancy between the predicted and the measured results. This can be done by using an instrumented experiment to

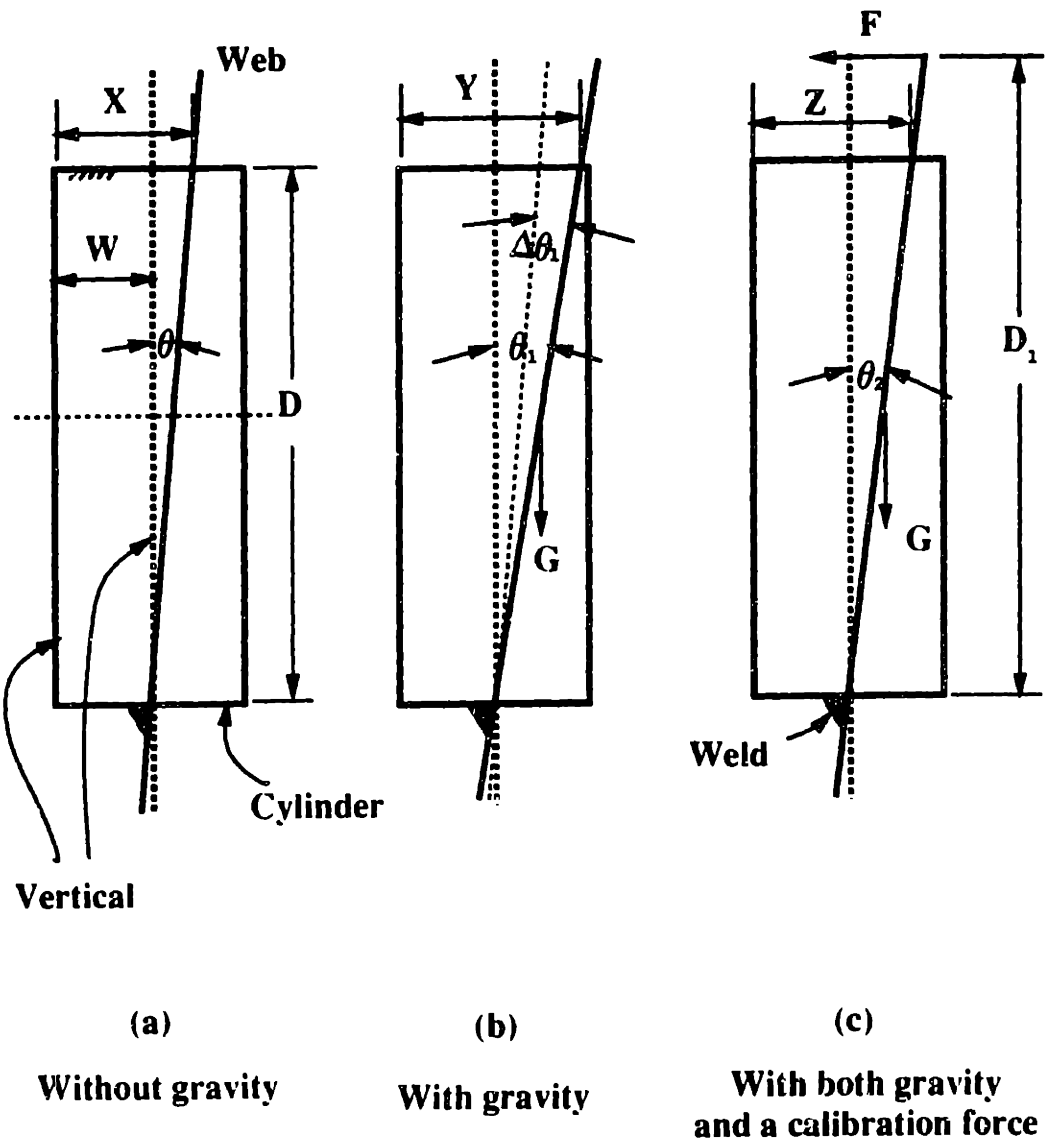


Figure 4.10: The calibration and correction of gravity effect when measuring the distortion of a welded ring-stiffened cylinder

measure the many variables during and after the welding process, such as deformation and reaction forces on the supports, so that the numerical predictions can be correlated with the experiments.

Two major conclusions can be drawn from the experimental results. First, the fixture setup is very important in welding distortion of large structures; it may increase distortion contrary to general expectations. Second, welding distortion of a large structure is complicated and non-intuitive; numerical simulations are necessary to predict such a distortion.

4.6 Deformation Mechanism During Welding Processes

In order to understand why fixtures may increase distortions, it is necessary to explore the mechanism of deformation during welding processes.

A structure will deform when one part is heated, just as the welding starts at one place. This deformation is primarily a result of the expansion of the local zone, but the exact pattern of deformation depends on the geometry of the structure and the fixture setups as well as the clearance gap before the weld is deposited. The local thermal expansion or contraction around the weld demands a deformation of the whole structure to accommodate the constraints supplied by fixtures. The deformation pattern, therefore, may be very complicated depending on the complexities of the fixture setup. When the weld is deposited and cools, however, it will lock the deformation pattern around the weld and retain much of the deformation after the removal of fixtures.

Such locking of deformation provided by the weld plays an important part in the

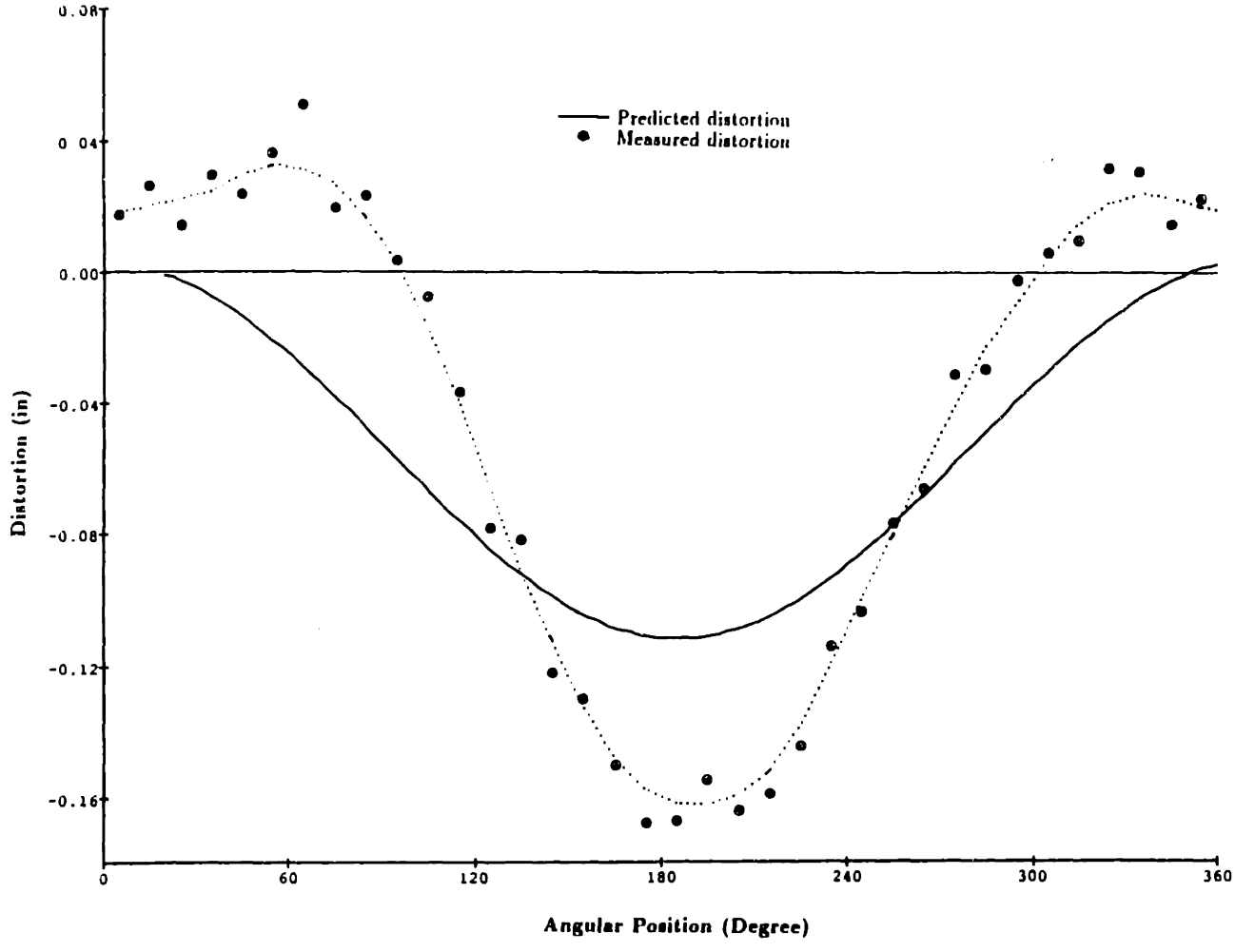


Figure 4.11: Welding distortion of the web around the circumference of the cylinder-web interface of the ring-stiffened cylinder. Four locations near the weld are tack welded for a locally constrained fixture setup (Case III)

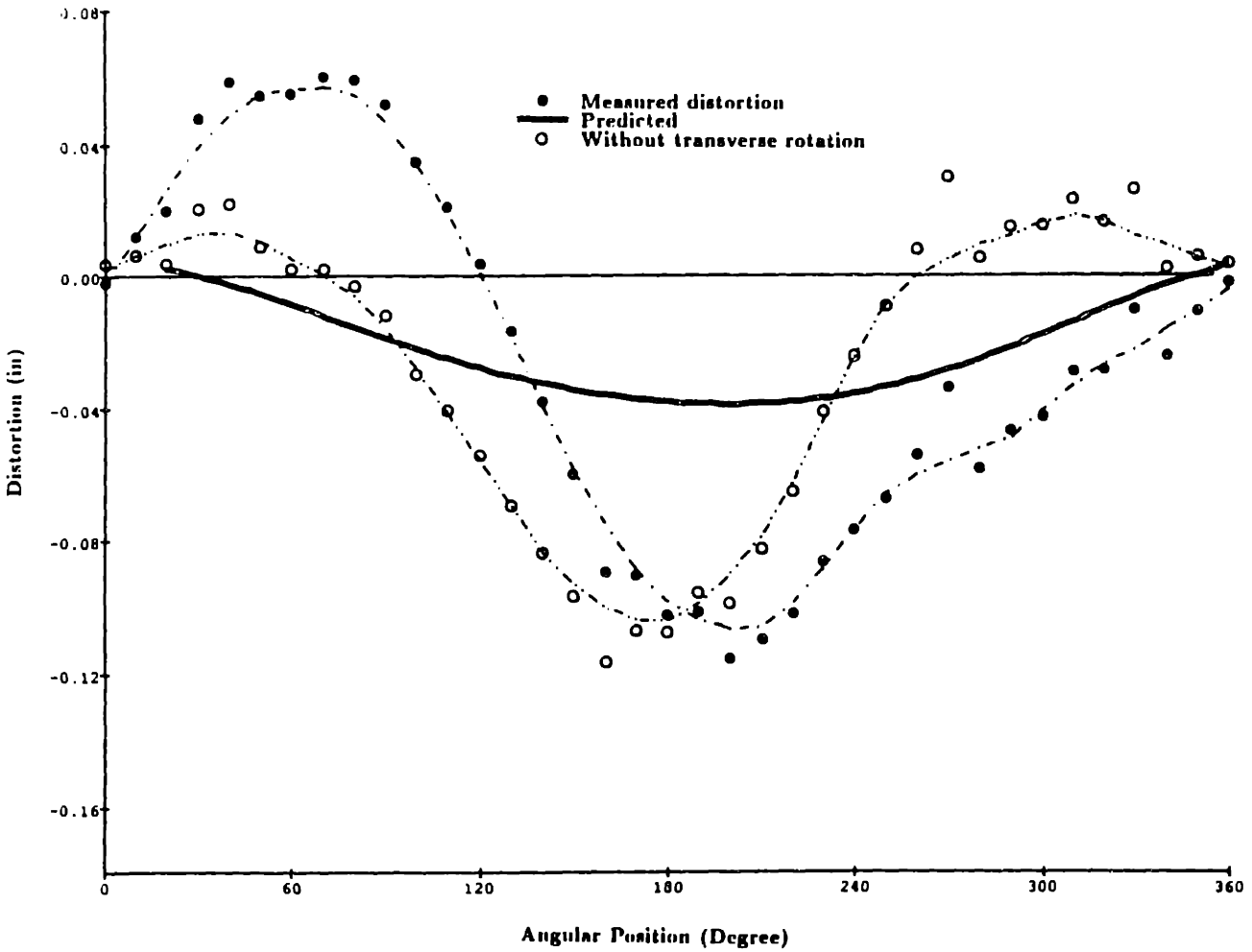


Figure 4.12: Welding distortion of the web around the circumference of the cylinder-web interface of the ring-stiffened cylinder. The cylinder and web sit on four supports on top of a table (Case I)

welding of the ring-stiffened cylinder, as demonstrated in both the numerical model and the experimental results. In the case of the locally constrained weld, deformation in the local zone is much more severe than in the case when the structure sits on a table. The deformation must accommodate the local constraints by the tack welds, as is shown in Figure 4.6(c). The later deposited weld locks such deformation in place and retains much of the deformation when the fixtures are finally removed.

In contrast, conventional reasoning of welding deformation is primarily based on the weld shrinkage, attributing distortion to the local weld deformation. Both the numerical simulations and the experiment in this thesis clearly indicate the importance of the effect of the interactions within the overall structure. Such interactions include the fixturing of the overall structure and weld fitup gaps. Successful numerical simulations or designs of welding processes should therefore consider the entire structure, not just a small part near the weld.

4.7 Discussion on 3-D Simulations

All the results from the three-dimensional simulations and the experiments show asymmetric distortions of both the cylinder and web. These asymmetric distortions can not be simulated with two-dimensional analyses. The need of three dimensional analyses for welding processes are obvious, particularly for large, complex structures. These results also imply that distortions due to welding in large structures can not be simulated and predicted on simple, small models. A large structure also involves complicated interactions among its components and deserves special consideration in welding simulations.

Fixture effects are not only important in welding practice, but also are difficult to predict without numerical simulation. Complex geometries of large structures complicate the interactions among the local weld zone, the rest of the structure, and fixtures; final distortions are usually not apparent. Numerical simulations are therefore necessary to predict the distortions so that costly, time-consuming welding experiments can be avoided. A designer can also gain insights to the mechanism behind the distortions through these numerical simulations, and therefore can design better structures. This is important particularly for distortion-sensitive structures such as underwater vessels.

One of these simulations takes about 60 hours of CPU time on an Apollo DN10000, including thermal and deformation analyses. A typical analysis consumes more than 100 megabytes of disk space. If the model is going to be expanded to include the whole weld pass around the ring stiffened cylinder, not only the disk storage is inadequate, but also the analysis will take a much longer period of time. The computer time consumption of a finite element analysis is roughly a cubic function of the total number of the degrees of freedom involved in the model, as explained in detail in the next chapter. The CPU time required to solve the finite element problem will increase sharply as the total number of degrees of freedom grows. For the three-dimensional model used above, the expansion to include the whole weld pass implies an increase of weld length from 1 inches to 112 inches, an increase of 112 times. The total number of nodes would increase roughly by 100 times and the total number of degrees of freedom by 300 times. A cubic function of this number is more than 27 million times! Such a simulation can not be performed in a reasonable period of time even with the aid of the fastest supercomputer existing today.

On one hand, predictions of welding distortions of a large structure need a three-

dimensional model, but on the other hand, as mentioned earlier, existing computational resources limit the successful simulations of such a model. The increase of computer speed by a factor of more than 1 million is not foreseeable in the near future. Moreover, there exist much larger and more complicated structures than the ring stiffened cylinder. Waiting for more powerful computers in terms of storage and computational speed will certainly take many years before we can effectively solve such problems.

Fortunately, the welding process has unique characteristics. Usually the part of the structure with high temperature, high gradients of variables and nonlinear material properties is localized in a small region around the weld. We can exploit this fact to reduce the three-dimensional model substantially. The next chapter describes the dynamic remeshing and substructuring techniques which makes the welding simulations more tractable.

Chapter 5

Substructuring and Dynamic Remeshing

The next half of this thesis will describe the dynamic remeshing technique and substructuring as well as their applications in welding simulations.

5.1 Substructuring Technique

The major tasks of the finite element method are to construct the stiffness matrix $[K]$ and to solve the matrix equation as expressed in Eq. 2.24. Usually the latter consumes much more time than the former. The substructuring technique was originally developed to condense part of a structure statically, therefore solving the problem of a large structure by parts. It has many other applications such as compiling many substructures in a library and choosing the appropriate substructure in an analysis. As is explained in this section, the substructuring technique can also substantially reduce the size of the stiffness matrix in an elastic or at least partly elastic structure, and thus reduce the time involved in solving the equations.

As shown in Figure 5.1, suppose we divide a structure to two parts, α and β, and suppose further that α is linear elastic during the process. Two sets of equations can then be constructed for α and β: [43]

$$[K^{\alpha}]\{U^{\alpha}\} = \{F^{\alpha}\} \tag{5.1}$$

$$[K^{\beta}]\{U^{\beta}\} = \{F^{\beta}\} \tag{5.2}$$

The dimensions of these two stiffness matrices are n_{α} and n_{β}, respectively. Both of them are smaller than the original dimension of the whole structure's stiffness matrix. If the degree of freedom associated with a node in the common boundary of the two substructures is represented by a subscript '$_{cb}$', and the others by '$_{n}$', then the above equation can be expanded as:

$$\begin{bmatrix} K^{\alpha}_{n} & K^{\alpha}_{n,cb} \\ K^{\alpha}_{cb,n} & K^{\alpha}_{cb} \end{bmatrix} \begin{Bmatrix} U^{\alpha}_{n} \\ U^{\alpha}_{cb} \end{Bmatrix} = \begin{Bmatrix} F^{\alpha}_{n} \\ F^{\alpha}_{cb} + R^{\alpha}_{cb} \end{Bmatrix} \tag{5.3}$$

$$\begin{bmatrix} K^{\beta}_{n} & K^{\beta}_{n,cb} \\ K^{\beta}_{cb,n} & K^{\beta}_{cb} \end{bmatrix} \begin{Bmatrix} U^{\beta}_{n} \\ U^{\beta}_{cb} \end{Bmatrix} = \begin{Bmatrix} F^{\beta}_{n} \\ R^{\beta}_{cb} \end{Bmatrix} \tag{5.4}$$

Here $\{R\}$ is the internal reaction force vector between the two substructures. Obviously,

$$\begin{aligned} \{R^{\beta}_{cb}\} &= -\{R^{\alpha}_{cb}\}, \\ \{U^{\beta}_{cb}\} &= \{U^{\alpha}_{cb}\} \end{aligned} \tag{5.5}$$

Reorganizing the above equations to represent the effect of substructure α on β will give us a set of equations as follows:

$$[\bar{K}^{\beta}]\{U^{\beta}\} = \{\bar{F}^{\beta}\} \tag{5.6}$$

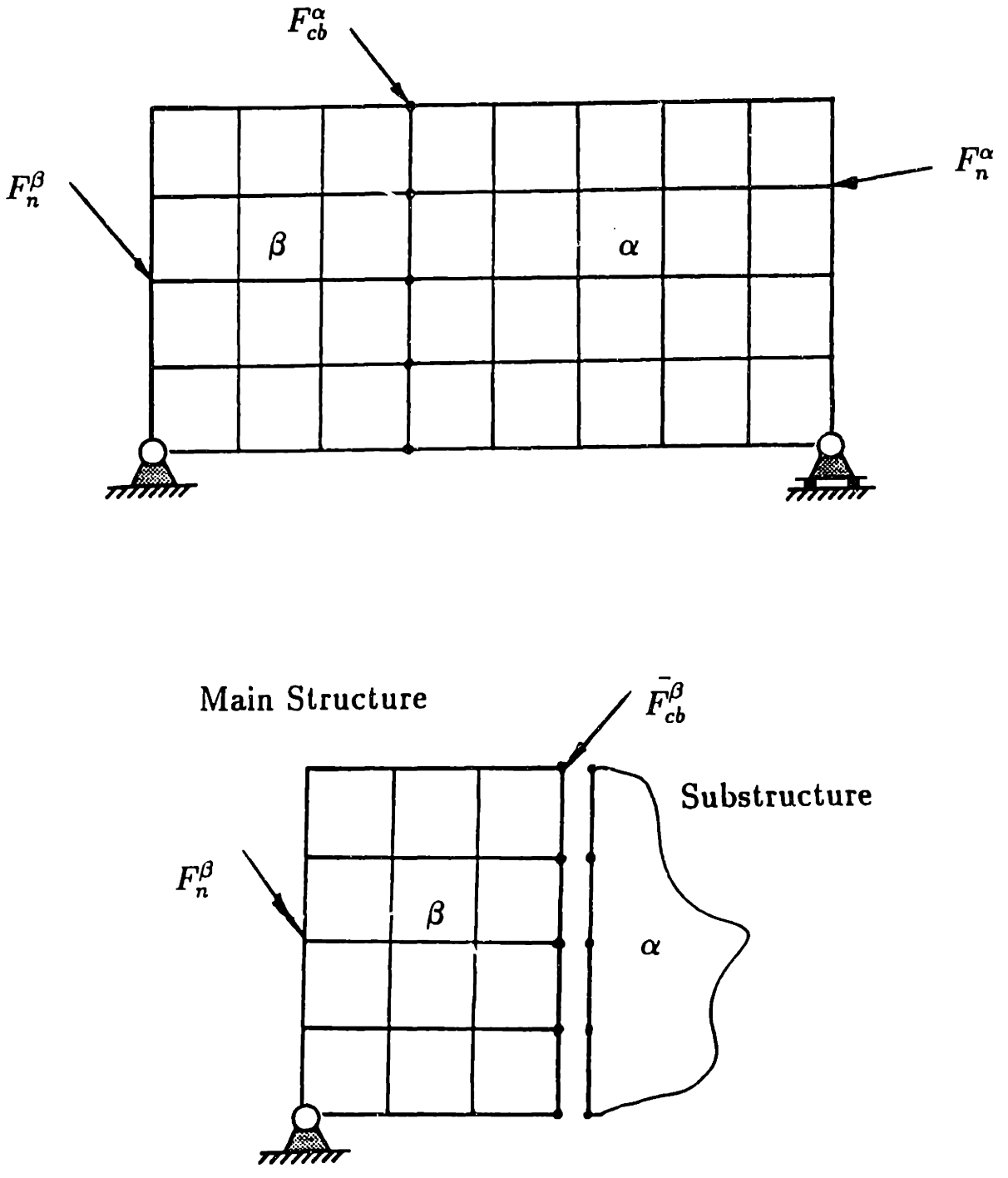


Figure 5.1: A structure can be divided into substructures and each of them is in equilibrium under the reaction forces along the boundaries.

where:

$$[\bar{K}^{\beta}] = \begin{bmatrix} [K_n^{\beta}] & [K_{n,cb}^{\beta}] \\ [K_{cb,n}^{\beta}] & [K_{cb}^{\beta}] + [K_{cb}^{\alpha}] - [K_{cb,n}^{\alpha}][K_n^{\alpha}]^{-1}[K_{n,cb}^{\alpha}] \end{bmatrix} \tag{5.7}$$

$$\{\bar{F}^{\beta}\} = \begin{Bmatrix} \{F_n^{\beta}\} \\ \{F_{cb}^{\beta}\} - [K_{cb,n}^{\alpha}][K_n^{\alpha}]^{-1}\{F_n^{\alpha}\} \end{Bmatrix} \tag{5.8}$$

The dimension of the new stiffness matrix $[\bar{K}^{\beta}]$ remains n_{β}, and could be much smaller than that of the original stiffness matrix representing the whole structure. Solving Eq. 5.6 therefore requires much less effort than the original Eq. 2.24. Better performance of the substructuring technique could be achieved by reducing the size of the main structure β.

If a solution on the elastic substructure α is required, it can be easily obtained from Eq. 5.5 and the following formulation:

$$\{U_n^{\alpha}\} = [K_n^{\alpha}]^{-1}(\{F_n^{\alpha}\} - [K_{n,cb}^{\alpha}]\{U_{cb}^{\beta}\}). \tag{5.9}$$

The corresponding strain or stress components can all be derived from the displacement vector $\{U^{\alpha}\}$.

5.2 Reduction of Computational Effort by Substructuring

We use operational count as an indicator of computational effort. Operational count, sometimes called *flop*, is defined as a multiplication followed by an addition. When dimensions of the stiffness matrices in Eq. 5.6 are relatively high (e.g. 1,000), computational efforts will be concentrated on solving the linear equations, and on inverting a matrix; other efforts associated with constructing the matrices are relatively small.

The substructuring technique obtains the inversion of matrix $[K_n^{\alpha}]$ first. If its dimension is large, the inversion could consume a substantial number of operational counts. According to Hager [44], the asymptotic numbers of operational counts required for an $n \times n$ matrix (when n is relatively large) are listed in Table 5.1.

Nonlinear finite element analysis solves the matrix equations in small increments. Usually one step takes many increments and each increment involves more than one iteration before a convergent solution is obtained. According to Table 5.1, the number of operational counts associated with the conventional FEM for i increments and an average of j iterations is:

$$NOC_1 = \frac{1}{2} ijw^2 n_1 \tag{5.10}$$

n_1 is the total number of degrees of freedom of the meshed structure, and w the associated band width of the stiffness matrix.

If the substructuring technique is used, a similar formula could be derived. Let n_2 denote the number of degrees of freedom of the substructure, n_3 of the main structure (β), n_4 of the common boundary, and w_2 the band width of matrix $[K_n^{\alpha}]$ associated with the substructure. Suppose that the band width of the new stiffness matrix $[\bar{K}^{\beta}]$ is w_3. Similar

Table 5.1: Operational counts needed for $n \times n$ matrix

Matrix Type	Factor Matrix & Solve Equations	Invert the Matrix
General solid matrix ($n \times n$)	$\frac{1}{3}n^3$	n^3
Symmetric, solid	$\frac{1}{6}n^3$	$\frac{1}{2}n^3$
Symmetric, band width=w	$\frac{1}{2}w^2 n$	$\frac{1}{2}(w+2n)wn$

to Eq. 5.10, if i increments and j iterations are involved,

$$NOC_2 = \underbrace{\frac{1}{2}[w_2 + 2(n_2 - n_4)]w_2(n_2 - n_4)}_{\text{Matrix Inversion}} + \underbrace{\frac{1}{2}ijw_3^2 n_3}_{\text{Solving New Equations}} \tag{5.11}$$

If more than one substructure are used, the equation above could be rewritten as:

$$NOC_2 = \sum_{l=1}^{k} \frac{1}{2}[w_{2l} + 2(n_{2l} - n_{4l})]w_{2l}(n_{2l} - n_{4l}) + \frac{1}{2}ijw_3^2 n_3 \tag{5.12}$$

where k is the number of substructures used. Note that the part related to matrix inversions is independent of the numbers of both the increments in the step and the iterations associated with each increment. This implies that matrix inversions can be done once for all the iterations. The more increments and iterations are needed the more powerful the substructuring technique will be in terms of reducing the computational effort. This is particularly valuable for nonlinear analyses since many increments and iterations are involved to obtain a convergent solution.

5.2.1 Substructuring Based on the Same Mesh

The power of substructuring could be demonstrated by comparing the operational counts needed for two models with the same mesh density. One model uses substructures while the other does not.

Suppose a model has a total number of degrees of freedom $n_1 = 25,000$ and a stiffness matrix band width $w = 2,500$. Suppose further that the analyses consist of ten increments and each increment involves an average of five iterations. If substructuring is not used, the total number of operational counts needed to solve the equations is, according to Eq. 5.10,

4.0×10^{12}: for a generally available workstation (assuming a speed of 0.5 *mflops*), this takes more than three months of CPU time.

If a substructure is used, assuming that the total number of degrees of freedom of the substructure is $n_2 = 22,000$ and of the main structure and the common boundary are $n_3 = 3,000$ and $n_4 = 200$, respectively, the time needed to solve the new problem can be calculated from Eq. 5.11. Let's use a conservative number for the band width of the new system stiffness matrix, $w_3 = w = 2,500$ and that of the substructure $w_2 = 1,500$. The total operational counts needed for ten increments and five iterations is 1.2×10^{12}, or less than one third of the original CPU time on a half megflop workstation. The improvement is obvious. Note that the number of increments is only ten, and normally in a nonlinear analysis many more increments are involved. If this number increases, the amount of saving in computer time will accordingly increase. The band width of a stiffness matrix is also critical in the effort needed to solve a finite element problem; a reduction in stiffness matrix band width yields greater improvements in the reduction of computer time needed.

Further reduction of computer time consumption can be achieved by using many substructures instead of one. The more substructures used the lower the dimensions of the substructure stiffness matrices, and the less effort needed to invert them.

5.2.2 Substructuring Based on Different Meshes

If the substructure is different from the main structure in material properties and deformation magnitude, we might want to use a different mesh density in the substructure (usually a coarser mesh). Such a difference in mesh density will result in a much smaller substruc-

ture stiffness matrix as well as a reduction in the band width of the main structure stiffness matrix.

Using the same example, suppose that the dimension of the substructure stiffness matrix is now $n_2 = 5,000$, its band width $w_2 = 500$, and at the same time the band width of the main structure stiffness matrix is reduced to $w_3 = 750$, the total operational count needed for the analysis will be 5.4×10^{10}, or about 30 hours of CPU time on a workstation.

It can be seen that the last term in Eq. 5.12 dominates when the dimensions of substructure stiffness matrices are relatively small. This implies that computational effort will be determined solely by the size of the main structure under consideration. This conclusion is very important for choosing an appropriate mesh for the main structure based on the available computational resource.

The actual factor of reduction could be even higher due to the fact that convergence is easier with fewer degrees of freedom and thus fewer iterations are needed for each increment, but such a reduction could not be characterized easily.

The substructure in the new model acts as coupled boundary conditions, and only applies to those parts of a structure that are linearly elastic. It can be applied to linear structures, including those with small, local nonlinear zones.

5.3 Dynamic Remeshing

Some manufacturing processes cause a structure to be only nonlinear locally and briefly during the processes; one typical example is the fusion welding process. During welding, a heat source (an electrode, a torch, or a electron beam) heats part of a structure along a pre-

designated path, raises the temperature and causes melting of the material. The material behaves nonlinearly at high temperatures; the rest of the structure acts as constraints on the deformation of the local heated zone. In such a case, the substructure can be used for the parts of the structure away from the heat source.

The conventional finite element method does not permit frequent changes of meshes during an analysis. Any place where the heat source has passed or will pass should be modeled with a denser mesh to capture nonlinear deformation, even though such nonlinear deformation only occurs briefly during the process. The simulation of welding process will be much more efficient if the mesh can change or evolve during an analysis. A dynamically evolving mesh implies that denser meshes are only needed at a particular period of time, and therefore the size of the model remains minimal. A huge waste of computational effort can then be avoided.

Research in the area of dynamic remeshing became active in the last two or three years. J. Cheng used an automatic adaptive remeshing technique in the simulation of forming processes[45]; Nishioka and Takemoto used a 'moving finite element method' in dynamic fracture simulations [46]. These dynamic remeshing are all for large deformations and are required to prevent singularities due to excessive deformation. Murti and Valliappan successfully developed a strategy to obtain the normalized coordinates and of a spatial point inside an element and a effective method to determine if a point is inside or outside an element[47]. These methods are important in the dynamic remeshing procedure as described in the following sections.

A complete reformulation of the finite element method to incorporate such needs of

mesh evolution during an analysis may be extremely difficult or impossible. Another strategy is to break the analysis into many increments and remesh between these incremental analyses, that is to say, to remesh the model dynamically during an analysis.

5.3.1 Characteristics of Dynamic Remeshing

Dynamic remeshing can clearly keep the size of a model minimal by focusing attention only on the nonlinear part of the structure. There are also other advantages.

Uncoupled analyses of thermal and deformation processes cannot track the deformation of the structure during the process and feed it back to the thermal analysis. This may cause inaccuracies in the temperature field in the case of relatively large displacements, and such inaccuracies will be carried over to the deformation analysis and result in even more erroneous displacement and stress/strain results. If the remeshing step between incremental analyses is applied to the thermal analyses, relative displacements can be updated and fed back to the temperature field; the analyses will be piece-wise coupled. Such feedback in the analyses would adequately serve the purpose of adjusting the temperature field as long as the incremental displacements within one incremental analysis are relatively small.

Major tasks of dynamic remeshing are to obtain deformed geometry of the model after one incremental analysis, create a new model, and then map variables from the previous results onto the new mesh. The next incremental analysis continues using initial conditions specified by the variable values which have been interpolated from the previous results.

The mapping of variables from one mesh to another of the same structure involves the process of obtaining the value of a variable at an integration point on the new mesh

from the corresponding data on the old mesh. One such process includes three parts: first extrapolate variables of the old mesh from the integration points to the nodes, and then average the values at a node from different elements to which the node is connected. This part may be done directly by the FEM code; the code we are using, ABAQUS, does not give proper results for data extrapolation when substructures are present. This step in the dynamic remeshing is therefore carried out by our own codes. Second, locate the integration points of the new mesh in the old one; that is to say, determine which element of the old mesh a new integration point belongs to. Finally, find the normalized coordinates of the integration point in that element and interpolate a variable value in terms of its values at the old nodes.

This procedure only provides a guidance to dynamic remeshing. There exists many different strategies to carry out these tasks. The variable interpolations, for example, may be directly performed based on the values at the integration points, though a completely new set of formulae are required.

During a common welding simulation, such remeshing occurs many times. Manual manipulations of the input decks and results to perform the remeshing would be laborious and tedious. We use a shell program to accomplish this task automatically by exploiting the fact that most of these remesh procedures are similar, and are only different from each other in the spatial position of the local zone and the transient distributions of variables such as temperature, displacement and stresses. This shell program will generate new meshes according to the immediately precedent results, map the value of variables to the new mesh, and drive the FEM code to carry out the next incremental analysis. A major part of this thesis work is on the design and actual implementation of this shell program.

5.3.2 Variable Extrapolation

The process of variable extrapolation from integration points to the nodes of an element is the reversal of variable interpolation. Suppose the values of a variable u are known at the integration points of an element. According to Eq. 2.23, the following equation holds:

$$u_j = \sum_{i=1}^{n} N_i(x_j, y_j, z_j) U_i \qquad (j = 1, 2, ...m) \tag{5.13}$$

where m is the total number of integration points. A common isoparametric element has the same numbers of integration points and nodes; that is to say, $m = n$. The above equation could be rewritten as:

$$\{u\} = [\bar{W}]\{U\} \tag{5.14}$$

The two vectors denote respectively the variable values at integration points and nodes. An element of the matrix $[\bar{W}]$ can be expressed in normalized coordinates as:

$$\bar{W}_{ij} = N_i(x_j, y_j, z_j) = N_i(\xi_j, \eta_j, \zeta_j) \tag{5.15}$$

Solving Eq. 5.14 will give the extrapolated value of a variable at nodal points:

$$\{U\} = [\bar{W}]^{-1}\{u\} \tag{5.16}$$

For a common non-singular element, the matrix $[\bar{W}]$ is always positive definite, and thus its inverse exists. By using normalized coordinates, the inverse of $[\bar{W}]$ will be the same for all elements of the same category, greatly simplifying the extrapolation process.

As two examples, the inverses of the shape function matrix $[\bar{W}]$ are given here for a quadrilateral four node element and an eight node brick element. Using the numbering convention shown in Figure 5.3, the shape function is expressed as:

$$N_i = \frac{1}{4}(1 + \xi_i\xi)(1 + \eta_i\eta) \tag{5.17}$$

Here ξ_i and η_i are the normalized nodal coordinates.

The Gauss integration points in this case are located at $\xi_j,\ \eta_j = \pm\frac{\sqrt{3}}{3}$, so that the matrix $[\bar{W}]$ is, letting $\chi = 1 + \frac{\sqrt{3}}{3}$ and $\phi = 1 - \frac{\sqrt{3}}{3}$:

$$[\bar{W}] = \frac{1}{4}\begin{bmatrix} \chi^2 & \chi\phi & \phi^2 & \chi\phi \\ \phi\chi & \chi^2 & \chi\phi & \phi^2 \\ \phi^2 & \chi\phi & \chi^2 & \phi\chi \\ \chi\phi & \phi^2 & \phi\chi & \chi^2 \end{bmatrix} \tag{5.18}$$

Its inverse can then be derived as:

$$[\bar{W}]^{-1} = \frac{1}{2}\begin{bmatrix} 2+\psi & -1 & 2-\psi & -1 \\ -1 & 2+\psi & -1 & 2-\psi \\ 2-\psi & -1 & 2+\psi & -1 \\ -1 & 2-\psi & -1 & 2+\psi \end{bmatrix} \tag{5.19}$$

with $\psi = \sqrt{3}/2$.

Similarly, for an eight node brick element show in Figure 5.4, shape functions can be expressed as:

$$N_i = \frac{1}{8}(1 + \xi_i\xi)(1 + \eta_i\eta)(1 + \zeta_i\zeta) \tag{5.20}$$

The corresponding shape function matrix and its inverse are, respectively:

$$[\bar{W}] = \frac{1}{8}\begin{bmatrix} \chi^3 & \chi^2\phi & \chi\phi^2 & \chi^2\phi & \chi^2\phi & \chi\phi^2 & \phi^3 & \chi\phi^2 \\ & \chi^3 & \chi^2\phi & \chi\phi^2 & \chi\phi^2 & \chi^2\phi & \chi^2\phi & \phi^3 \\ & & \chi^3 & \chi^2\phi & \phi^3 & \chi\phi^2 & \chi^2\phi & \chi\phi^2 \\ & & & \chi^3 & \chi\phi^2 & \phi^3 & \chi\phi^2 & \chi^2\phi \\ & & & & \chi^3 & \chi^2\phi & \chi\phi^2 & \chi^2\phi \\ & & \text{Sym.} & & & \chi^3 & \chi^2\phi & \chi\phi^2 \\ & & & & & & \chi^3 & \chi^2\phi \\ & & & & & & & \chi^3 \end{bmatrix} \tag{5.21}$$

Note that if the numbering convention is different, the matrices and their inverses will also change.

After one incremental analysis, values of variables are available at either nodes or integration points. The above extrapolation process allows all the variables to be expressed at nodal points, and then to be used later when interpolation is performed.

5.3.3 Locating a Spatial Point in a Mesh

Before any interpolation of a variable to a specific point is performed, the exact location where this point belongs to needs to be determined. We adopt a variation of a strategy proposed by Murti and Valliappan[47] to determine if a point is inside or outside an element. This variation allows a readily implementation and the in sensitiveness of the results to numerical shortcomings such as round-off errors. The following describes this procedure to determine if a point is inside or outside an element.

For a two-dimensional, quadrilateral four node element, as shown in Figure 5.5, a point is inside an element if:

$$A_i = \frac{1}{2}\vec{a}_i \times \vec{b}_i \geq 0 \qquad (i = 1, 2, 3, 4) \tag{5.23}$$

in which $\vec{b}_i$ is the ith side of the element. The constant $\frac{1}{2}$ does not affect this inequality

$$
[\bar{W}]^{-1} = \begin{bmatrix}
98\sqrt{3}+78 & -92\sqrt{3}+18 & 21\sqrt{3}-27 & -12\sqrt{3}+21 & 12\sqrt{3}+21 & 145\sqrt{3}-200 & 32\sqrt{3}-35 & 47\sqrt{3}-60 \\
41\sqrt{3}+26 & 64\sqrt{3}+91 & -10\sqrt{3}+82 & -22\sqrt{3}+24 & 22\sqrt{3}+24 & 198\sqrt{3}+10 & 77\sqrt{3}+36 & 91\sqrt{3}-2 \\
 & 95\sqrt{3}+4 & -27\sqrt{3}+45 & 145\sqrt{3}-200 & 21\sqrt{3}-27 & -27\sqrt{3}+45 & -71\sqrt{3}+50 & 32\sqrt{3}-35 \\
 & 84\sqrt{3}-81 & -46\sqrt{3}+90 & 198\sqrt{3}+10 & -10\sqrt{3}+82 & -46\sqrt{3}+90 & 48\sqrt{3}+79 & 77\sqrt{3}+36 \\
 & & 98\sqrt{3}+56 & -91\sqrt{3}+67 & -111\sqrt{3}+187 & 70\sqrt{3}-96 & -27\sqrt{3}+45 & 145\sqrt{3}-200 \\
 & & 78\sqrt{3}-44 & 43\sqrt{3}+66 & 66\sqrt{3}+61 & 95\sqrt{3}+61 & -46\sqrt{3}+90 & 198\sqrt{3}+10 \\
 & & & 98\sqrt{3}-55 & 70\sqrt{3}-91 & -111\sqrt{3}+187 & 21\sqrt{3}-27 & 12\sqrt{3}+21 \\
 & & & 12\sqrt{3}+25 & 56\sqrt{3}+73 & 66\sqrt{3}+61 & -10\sqrt{3}+82 & 22\sqrt{3}+24 \\
 & & & & 99\sqrt{3}-55 & -91\sqrt{3}+67 & 145\sqrt{3}-200 & -12\sqrt{3}+21 \\
 & & & & 12\sqrt{3}+25 & 43\sqrt{3}+66 & 198\sqrt{3}+10 & 22\sqrt{3}+24 \\
 & & & & & 98\sqrt{3}+56 & -27\sqrt{3}+45 & 21\sqrt{3}-27 \\
 & & & & & 78\sqrt{3}-44 & 46\sqrt{3}+90 & -10\sqrt{3}+82 \\
 & & & & & & 95\sqrt{3}+4 & -92\sqrt{3}+18 \\
 & & & & & & 84\sqrt{3}-81 & 64\sqrt{3}+91 \\
 & & & & & & & 98\sqrt{3}+78 \\
 & & & & & & & 41\sqrt{3}+26
\end{bmatrix} \quad (5.22)
$$

Sym.

Figure 5.2: The inverse of the shape function matrix $[\bar{W}]$ for a three-dimensional brick element

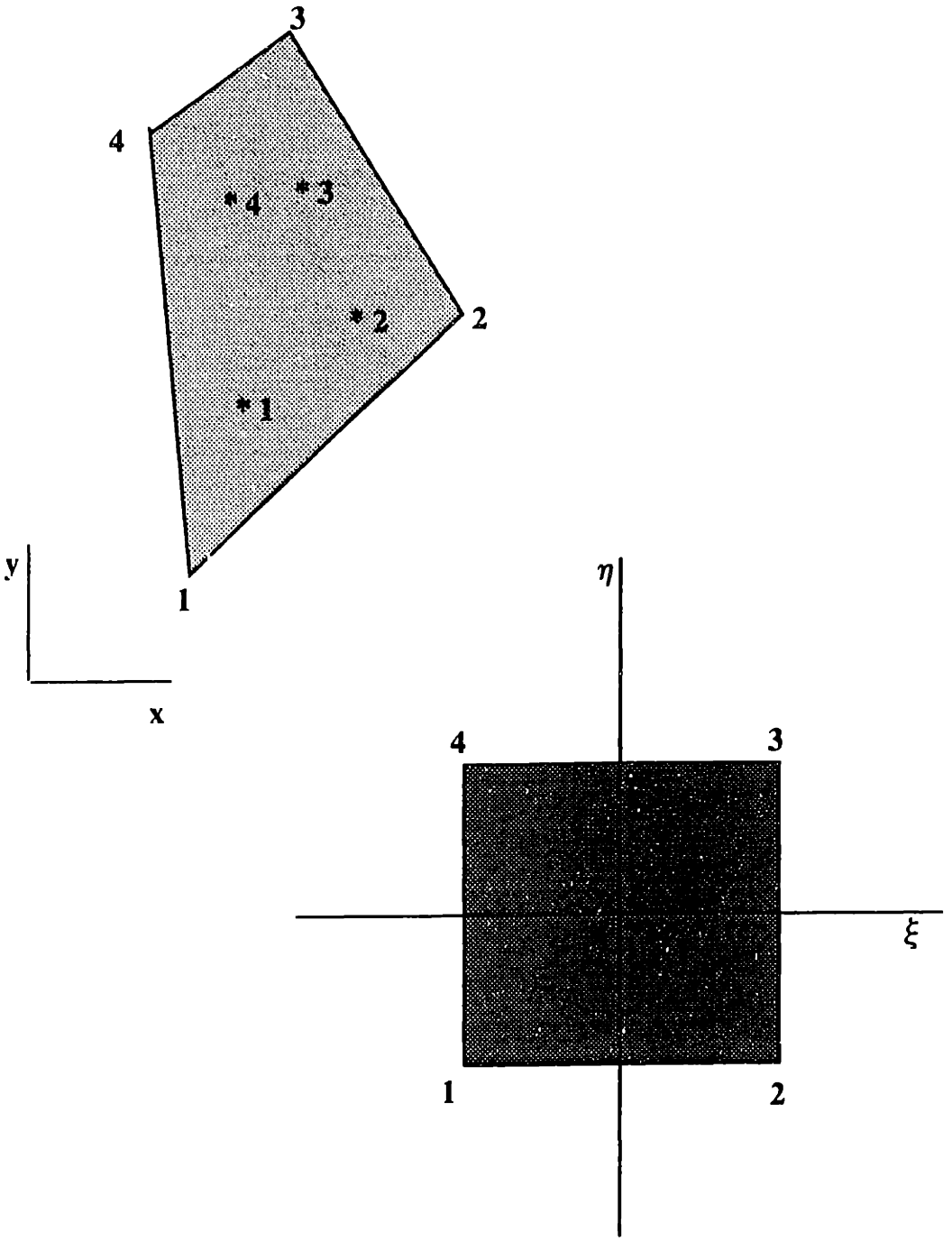


Figure 5.3: A typical four node quadrilateral element

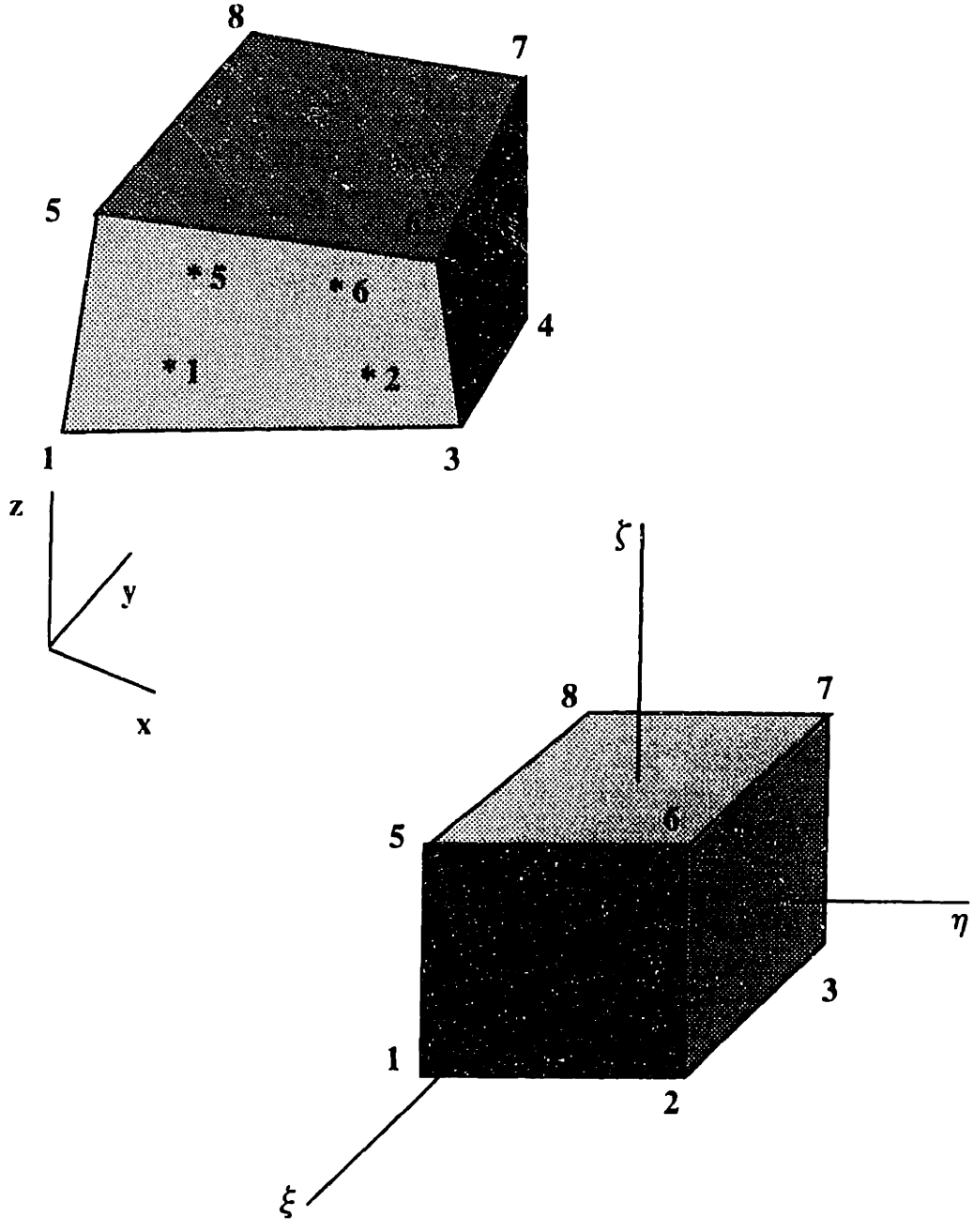


Figure 5.4: A typical eight node brick element

but gives a clear definition of A_i as the area of the triangle formed by $\vec{a}$ and $\vec{b}$. Its direction is determined by the right-hand rule so that the element is always on the left side of the vector; $\vec{a}_i$ is the vector connecting the point to the starting node of the ith side. If the point is not inside the element, there exists at least one negative product as shown in Figure 5.5. The equal sign holds when the point lies on the edges of the element.

Similarly, for an eight node, three-dimensional brick element, a point lies inside an element if the following holds:

$$V_i = \frac{1}{3}\vec{a}_i \cdot (\vec{b}_i \times \vec{c}_i) \geq 0 \qquad (i = 1, 2, 3...6) \tag{5.24}$$

where i is the ith surface of the element and the vectors are defined as shown in Figure 5.6.

Such inequalities can be checked, and the exact location of the point, where interpolation of a variable will be performed, can be effectively determined.

5.3.4 Normalized Coordinates

After the location of a point is determined, variable interpolation can be performed based on its normalized coordinates inside the element to which it belongs. The normalized coordinates, however, need to be determined first.

Many different strategies can be used to obtain the normalized coordinates of a point inside an element[37, 47]. The following procedure is straightforward and easy to implement. Referring to Eq. 2.23, all variables can be interpolated according to their values at nodal points. The coordinate transformation follows the same rule for isoparametric

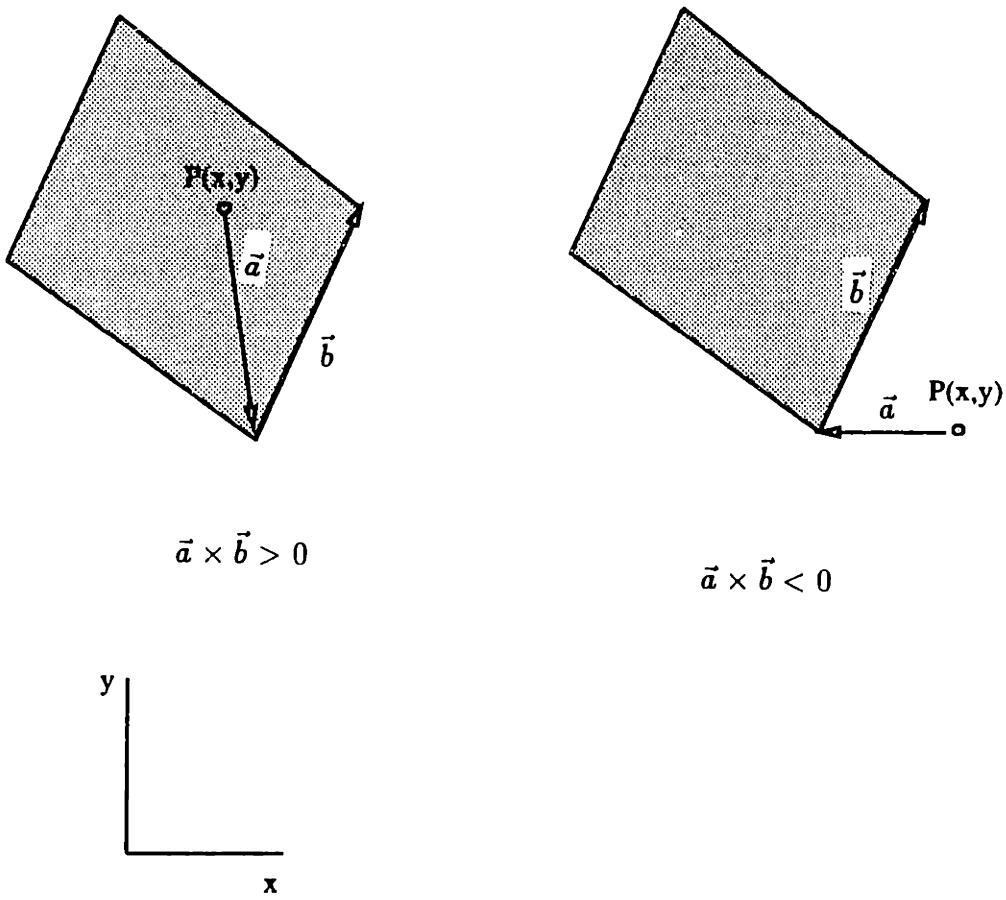


Figure 5.5: A spatial point is either inside or outside an element

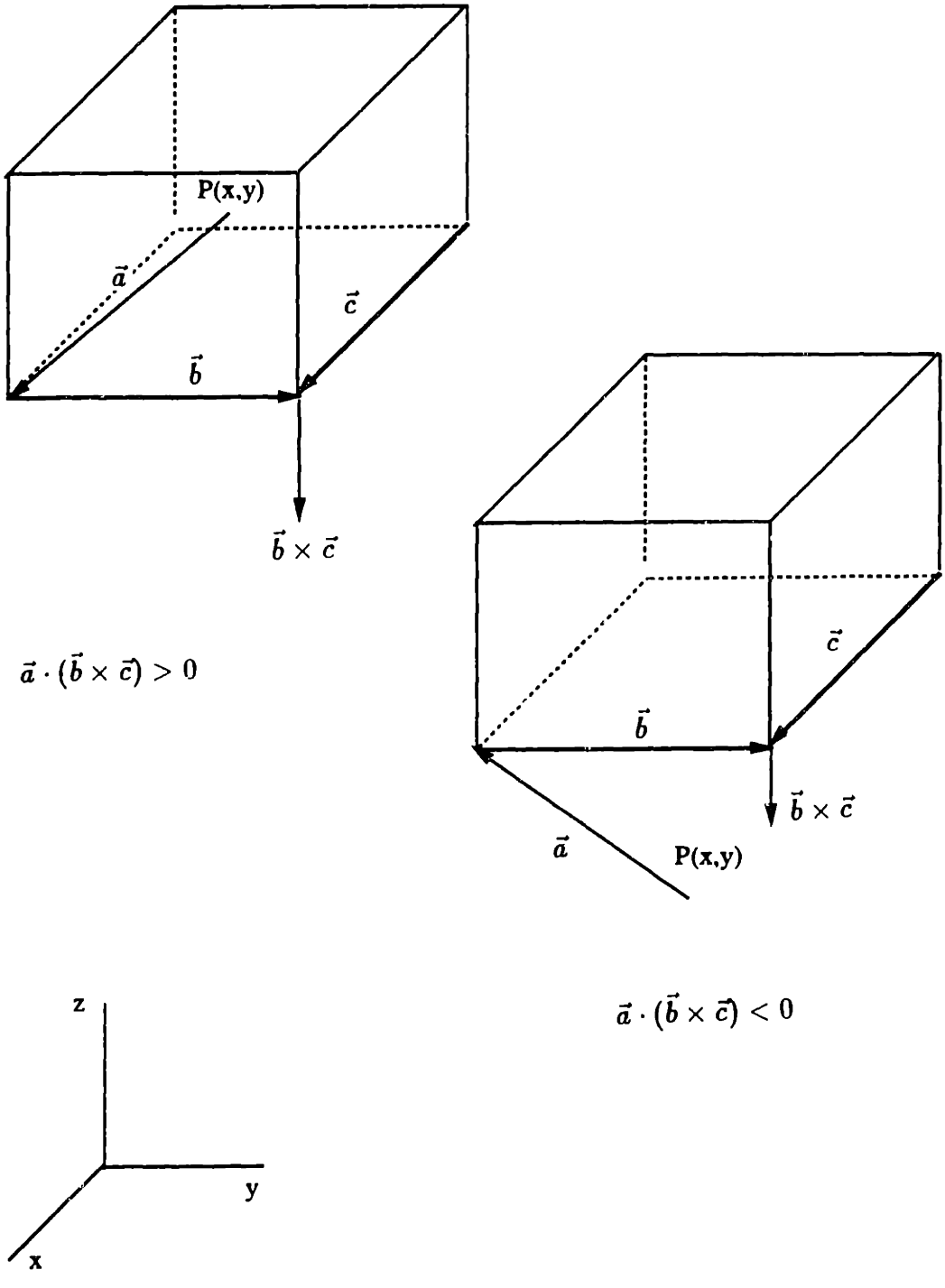


Figure 5.6: The location of spatial point in relation to a 3-D element

elements:

$$\begin{aligned} x &= \sum_{i=1}^{n} N_i(\xi,\eta,\zeta)x_i \\ y &= \sum_{i=1}^{n} N_i(\xi,\eta,\zeta)y_i \\ z &= \sum_{i=1}^{n} N_i(\xi,\eta,\zeta)z_i \end{aligned} \tag{5.25}$$

Figure 5.7 shows a schematic drawing of the coordinate transformations. If the nodal coordinates x_i, y_i, and z_i are known, these three equations uniquely determine a set of normalized coordinates ξ, η, and ζ associated with a spatial point with global coordinates x, y, and z.

For a four-node two-dimensional element, only two coordinates are needed for any point, and the above equations become:

$$\begin{aligned} a_0 + a_1\eta + a_2\xi + a_3\xi\eta &= 0 \\ b_0 + b_1\eta + b_2\xi + b_3\xi\eta &= 0 \end{aligned} \tag{5.26}$$

The coefficients are:

$$\begin{aligned} a_0 &= 4x + x_1 + x_2 + x_3 + x_4 \\ a_1 &= -x_1 - x_2 + x_3 + x_4 \\ a_2 &= -x_1 + x_2 + x_3 - x_4 \\ a_3 &= x_1 - x_2 + x_3 - x_4 \\ b_0 &= 4y + y_1 + y_2 + y_3 + y_4 \\ b_1 &= -y_1 - y_2 + y_3 + y_4 \\ b_2 &= -y_1 + y_2 + y_3 - y_4 \\ b_3 &= y_1 - y_2 + y_3 - y_4 \end{aligned} \tag{5.27}$$

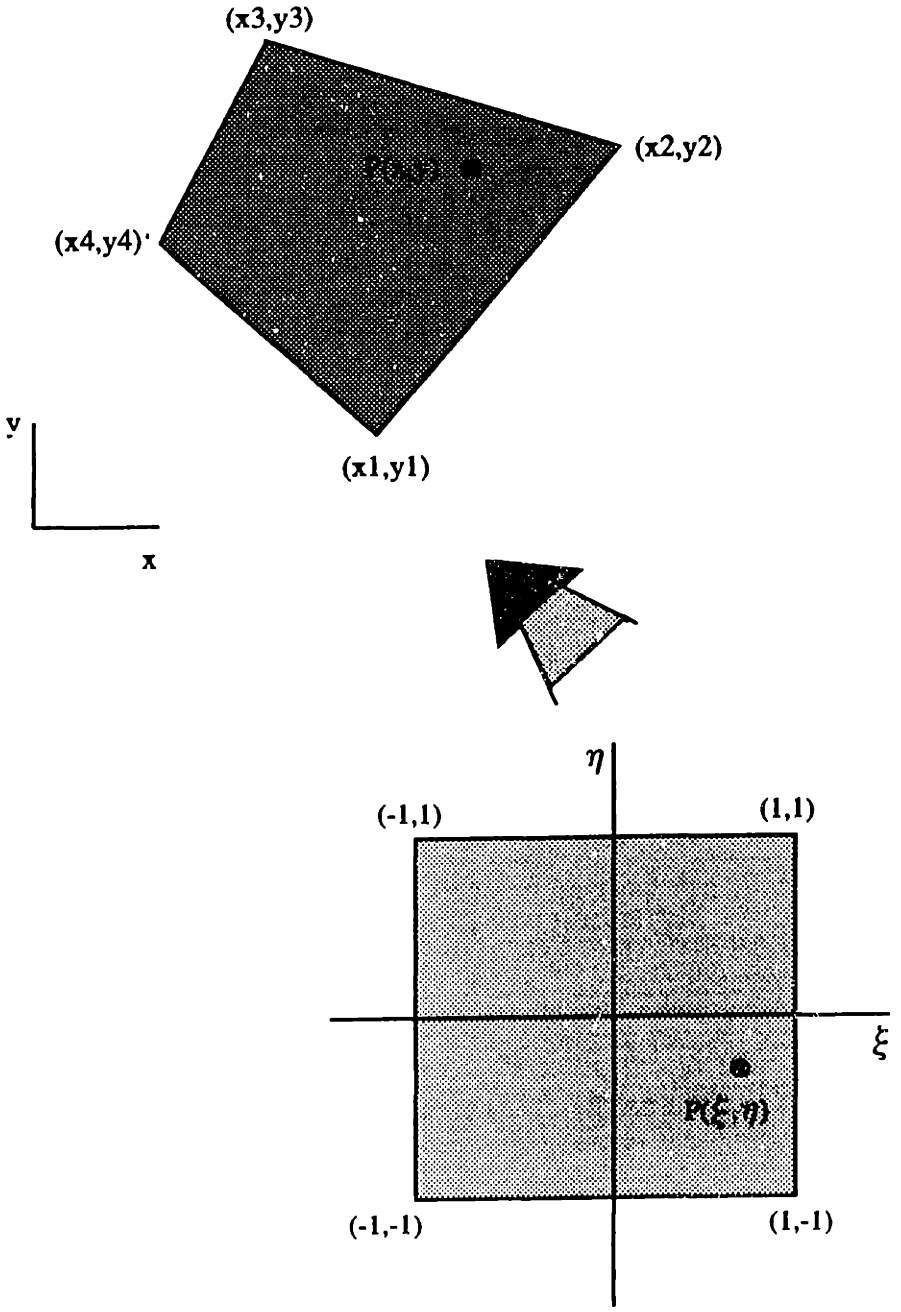


Figure 5.7: The coordinates of a point in an element can be uniquely transformed to a set of normalized coordinates

In the case of a rectangular element, we have $a_3 = 0$, and $b_3 = 0$. The equations in Eq. 5.26 become linear and an analytical solution is not difficult to obtain. For other elements with arbitrary geometries, efficient solution strategies remain the most difficult parts of the dynamic remeshing procedures.

Similarly, the corresponding equations for a three-dimensional linear element can also be derived as follows:

$$\begin{aligned} a_0 + a_1\xi + a_2\eta + a_3\zeta + a_4\xi\eta + a_5\eta\zeta + a_6\xi\zeta + a_7\xi\eta\zeta &= 0 \\ b_0 + b_1\xi + b_2\eta + b_3\zeta + b_4\xi\eta + b_5\eta\zeta + b_6\xi\zeta + b_7\xi\eta\zeta &= 0 \\ c_0 + a_1\xi + c_2\eta + c_3\zeta + c_4\xi\eta + c_5\eta\zeta + c_6\xi\zeta + c_7\xi\eta\zeta &= 0 \end{aligned} \tag{5.28}$$

with the coefficients defined by:

$$\begin{aligned} a_0 &= 8x + x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 \\ a_1 &= -x_1 + x_2 + x_3 - x_4 - x_5 + x_6 + x_7 - x_8 \\ a_2 &= -x_1 - x_2 + x_3 + x_4 - x_5 - x_6 + x_7 + x_8 \\ a_3 &= -x_1 - x_2 - x_3 - x_4 + x_5 + x_6 + x_7 + x_8 \\ a_4 &= x_1 - x_2 + x_3 - x_4 + x_5 - x_6 + x_7 - x_8 \\ a_5 &= x_1 + x_2 - x_3 - x_4 - x_5 - x_6 + x_7 + x_8 \\ a_6 &= x_1 - x_2 - x_3 + x_4 - x_5 + x_6 + x_7 - x_8 \\ a_7 &= -x_1 + x_2 - x_3 + x_4 + x_5 - x_6 + x_7 - x_8 \end{aligned} \tag{5.29}$$

The other coefficients b_i and c_i can be obtained by replacing x_i in the above equations by y_i and z_i, respectively.

When dynamic remeshing is performed, the structure is in the middle of a deformation process; geometries of elements are arbitrary even if they are initially defined as regular rectangles or cubes. The irregular shapes of elements imply that the higher order terms in Eqs. 5.26 and 5.28 are not eliminated.

We use iteration to solve the high order simultaneous equations. Since remeshing involves mapping variables all over the model, a fairly large number of such solutions are usually needed, particularly for large models with many degrees of freedom. The computational effort may not be negligible, and better solution strategies are apparently needed[47]. Note that if the elements are initially defined as regular rectangles or cubes and deformation in the process is relatively small, the coefficient of higher order terms could be relatively small, and iteration will be much faster in such cases.

The interpolation is straightforward after the normalized coordinates are obtained, as is expressed in Eq. 2.23. The formulations described above for two- and three-dimensional elements can also be expanded to other types of elements. Special elements such as springs and interfaces, however, may need special attention.

5.4 The Shell Program

The source codes of most commercial packages of finite element analyses are not available to the general public; the implementation of dynamic remeshing by manipulating the source codes of these packages is not possible. Since the dynamic remeshing procedure is independent of the incremental analyses, a separate set of subroutines can be developed to carry out the above-mentioned remeshing in between incremental analyses. A shell pro-

gram is thus required to drive both the finite element codes and the remeshing subroutines. This shell program is written with the operating system command language DCL (Digital Command Language) in VMS.

Figure 5.8 is the flow chart of this shell program. The shell program basically calls the FEM and the remeshing codes alternately, while keeping the current position of the local zone and other information stored in a readily accessible data file. The remeshing will be carried out based on this data file and on the results from the previous incremental analysis. The shell program used in this thesis is written in DCL and listed in Appendix B.

The substructuring technique will permit small models for welding simulations, and dynamic remeshing will enable us to focus our attention only on the local, moving, nonlinear weld zone. Such techniques will substantially reduce computational effort and at the same time allow the model to capture the important parts of nonlinear deformation (or temperature field). The next chapter describes in detail the implementation of these techniques to the simulation of a two-dimensional plate welding situation.

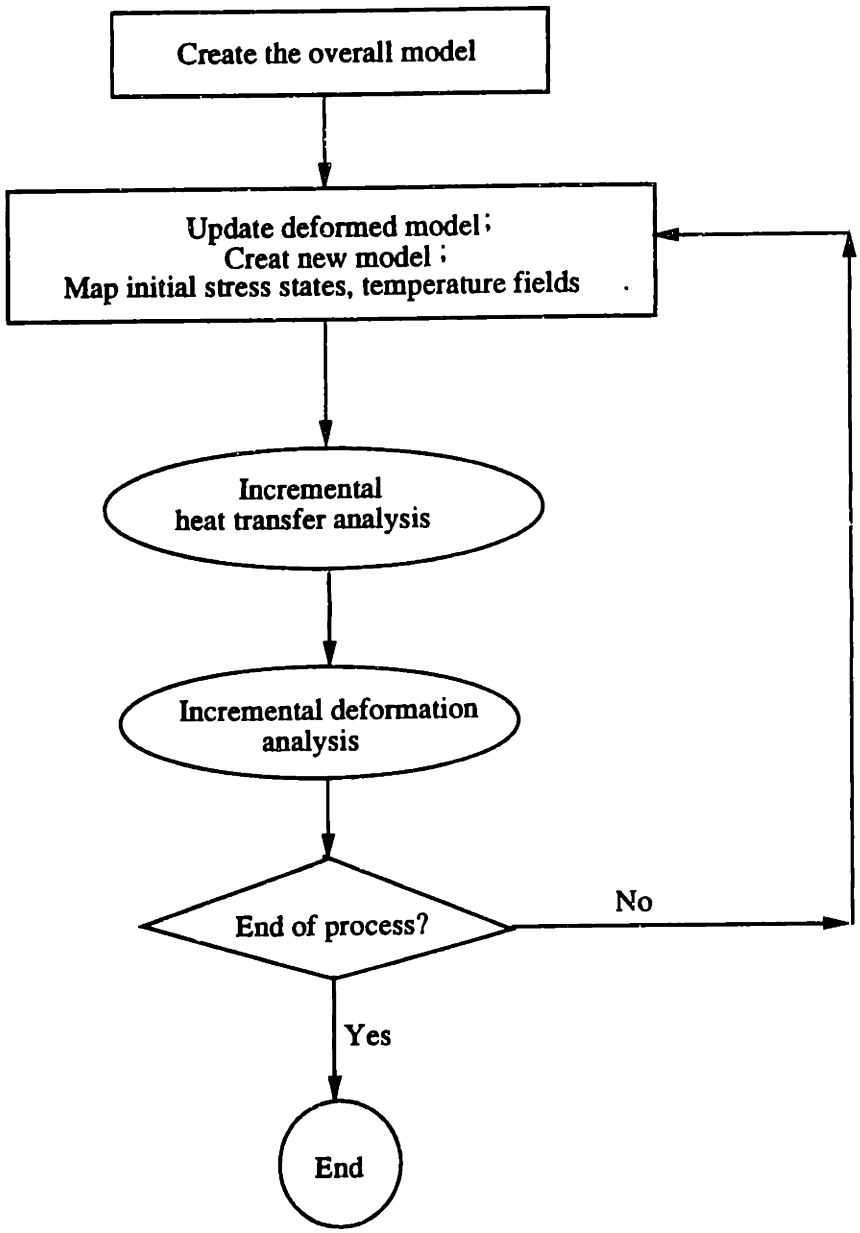


Figure 5.8: The flow chart of the shell program

Chapter 6

Dynamic Remeshing Applied to Plate Welding

The dynamic remeshing technique is implemented in the simulation of a plate welding situation. The plate geometry is chosen based on many factors such as simplicity for easier implementation and availability of results in the literature for comparisons.

6.1 The FEM Model

Figure 6.1 shows the finite element mesh of a plate. A heat source translates from one end to the other of the edge shown as the denser mesh. Since this model is only a test case for the implementation of the dynamic remeshing technique, no weld material deposition is considered in order to simplify the procedure. The model could be interpreted as an edge heating rather than a welding process; nonetheless, after modifying the boundary conditions it could also be interpreted as an electron or laser beam welding.

To apply the dynamic remeshing technique to the model, the in-plane two-dimensional model is chosen. We assume the thickness of the plate to be thin enough so that differences

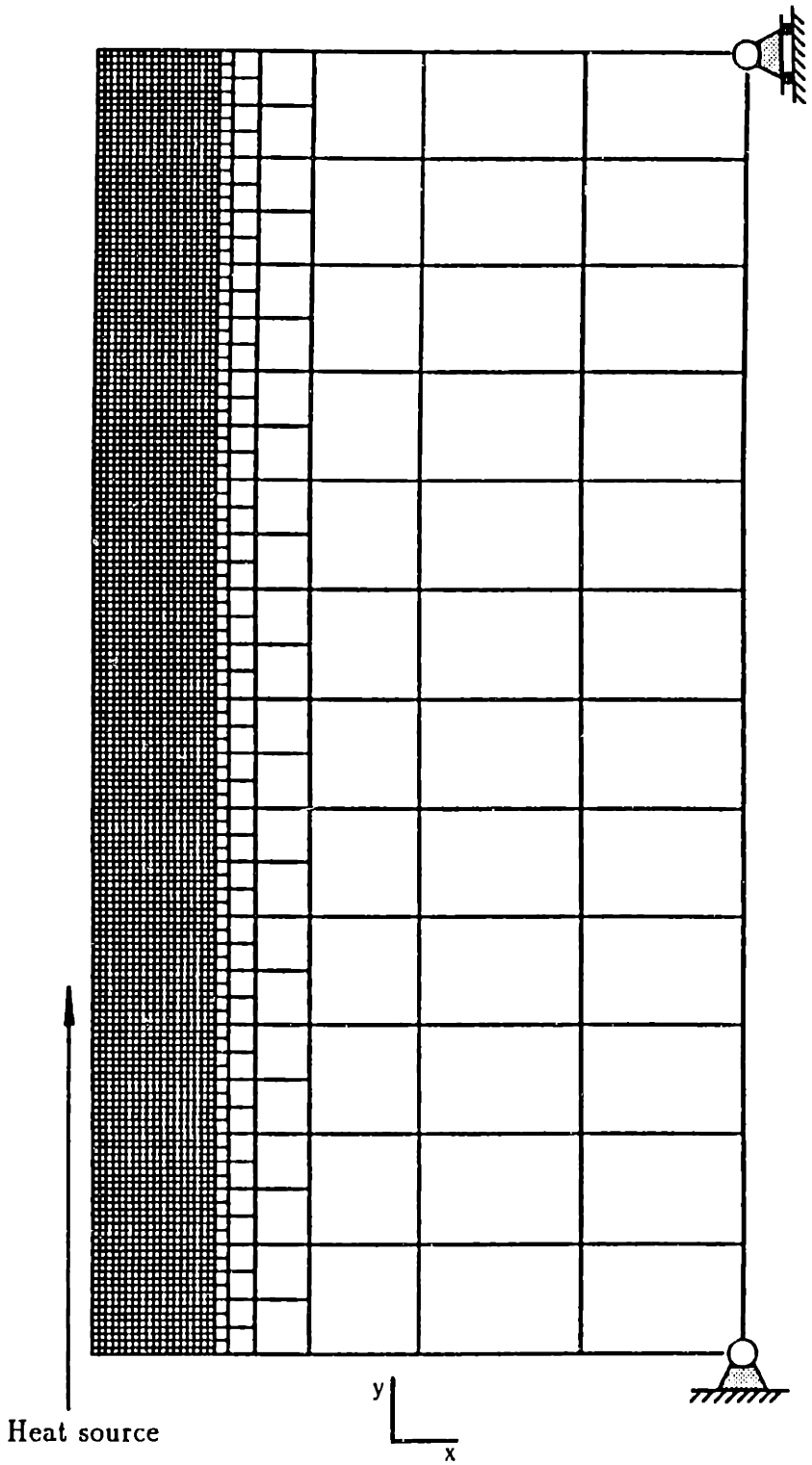


Figure 6.1: The full mesh of a plate heated on one edge

and gradients of variables between the two surfaces can be neglected. Normal stress components on the surfaces are zero. and thus a plane stress formulation would appropriately represent the situation. Minimal constraints are used as boundary conditions for the deformation analyses, as shown in Figure 6.1, so that the transient deformation is relatively large and the effectiveness of remeshing on the coupling of temperature and deformation can be tested.

In the thermal analyses, the strength of heat flux into the plate is defined as a Gaussian function of coordinates:

$$q = pe^{-\sigma[(x-x_0)^2+(y-y_0)^2]} \tag{6.1}$$

where x_0 and y_0 are the current coordinates of the heat source, and the factors p and σ are adjusted so that melting occurs along the welding path.

Boundary heat losses due to convection and radiation are mainly from the two surfaces of the plate; such boundary heat transfer could not be applied to the two-dimensional model. An equivalent volumetric heat sink is used all over the plate to add the boundary effect to the model. The following equation formulates the volumetric heat sink representing the boundary heat losses, assuming a unit thickness:

$$q = -2[h_c(T - T_a) + \epsilon\sigma(T^4 - T_a^4)] \tag{6.2}$$

where the factor 2 denotes the two surfaces of the plate, T_a is the ambient air temperature, and the coefficient h_c and ϵ take their corresponding values at temperature T.

A full model of such a plate is also used to obtain both temperature and deformation results which can be used in comparison with dynamic remeshing results. Figure 6.2 shows

the distorted mesh of the full model at the end of the process, and Figure 6.3 shows the corresponding Mises stress distribution. The Mises stress is the equivalent tensile stress for a three-dimensional stress tensor, and is particularly important when plastic deformation occurs. The Mises stress is defined by:

$$\bar{\sigma} = \sqrt{\frac{3}{2} s_{ij} s_{ij}} \tag{6.3}$$

Where the deviatoric stress component s_{ij} is defined as:

$$s_{ij} = \sigma_{ij} - \frac{1}{3}\delta_{ij}\sigma_{kk} \tag{6.4}$$

The stress distributions in Figure 6.3 are very close to the reported stress distribution for a similar plate welding situation in Ueda et al.'s paper [48].

6.2 Remeshing Frequency and Size of the Local Zone

The first step in dynamic remeshing is to decide the remeshing frequency and the size of the local zone. These two factors are closely interwoven. Obviously, the heat source should be always inside the local zone; a smaller local zone therefore implies a higher frequency of remeshing. To increase the efficiency of dynamic remeshing, the size of the local zone should be kept small so that the total number of degrees of freedom associated with the model is also small. However, higher frequency of remeshing costs more since remeshing needs both computer time and storage space. An optimal selection of these two factors should be well balanced based on consideration of the efficiency and overhead cost.

Fortunately, the analyses described later prove that the overhead cost on computer time is very small compared to the time needed for an incremental analysis in one typical

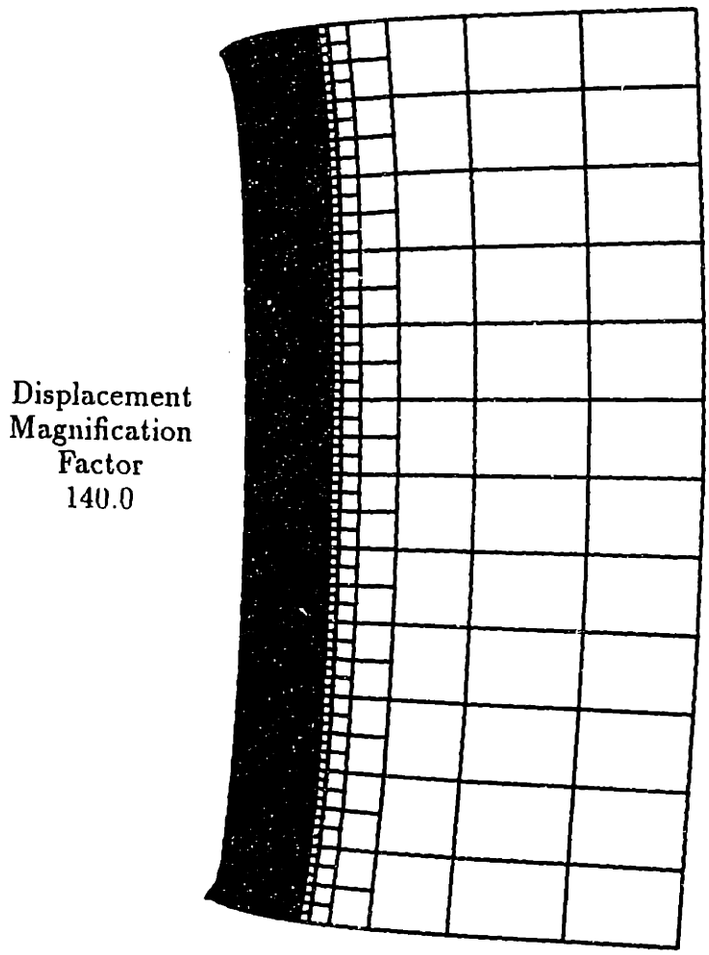


Figure 6.2: Final distortion of the plate

Mises Stress (Mpa)	
1	0
2	30
3	60
4	90
5	120
6	150
7	180
8	210
9	240
10	270
11	300

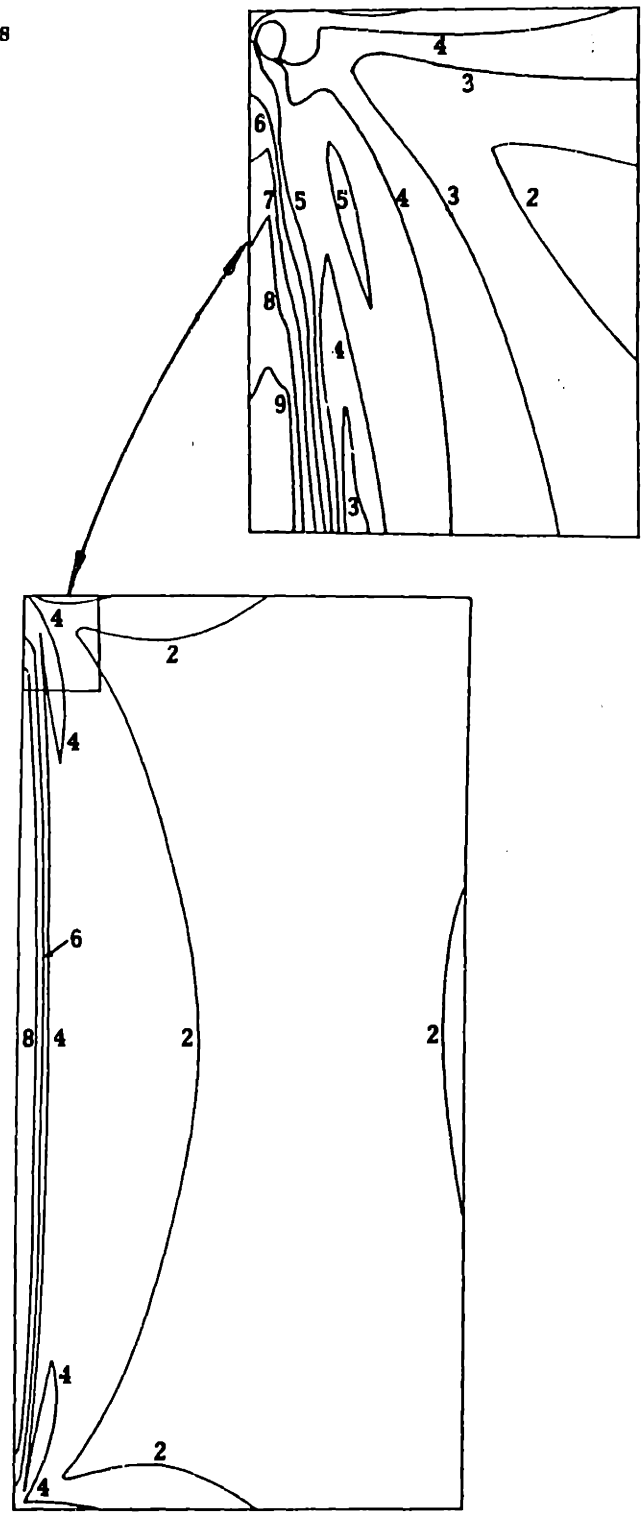


Figure 6.3: Final Mises stress distribution on the plate

simulation with 10 to 20 remeshes. A very small local zone can be selected. Other factors also affect the selection of the size of the local zone; such as thermal expansion and material properties. Thermal expansions in substructures cannot be tracked since substructures only act as coupled boundary conditions on the local zone. The size of the local zone is therefore carefully chosen to exclude both high temperature and thermal expansion. In the plate model, a temperature of 150 °C is used to delimit the boundary of the local zone.

Since we have the results of temperature distribution from the analysis with the full model, it is not difficult to determine the boundary of the local zone. In the case of a large complicated structure, such a boundary could be determined based on related information such as previous thermal analyses (if any), the geometry of the base metal around the weld, the power input, and any previous temperature measurements. The mesh density inside the local zone could also vary to catch the high gradients of temperatures and strains close to the weld, while keeping the size of the model small, as we demonstrate in the following sections.

Once the size of the local zone is chosen, the frequency of remeshing can be determined based on the traveling velocity of the heat source and how much the local zone overlaps between two consecutive incremental analyses. Such overlapping is necessary to ensure that the heat source is well inside the local zone. The remeshing frequency and the overlapping factor are both saved in a data file for quick retrieval whenever needed.

6.3 Dynamic Remeshing without Substructuring

The programs written for dynamic remeshing are specific to the plate problem, though they can be modified to apply to other geometries and for different welding parameters. These programs can also be modified in the future either to include all possible welding situations or to include several categories of generic geometries such as plates, pipes, rings, T-shaped structures and so on.

To simplify the remeshing procedure the same node numbering convention is used in both the full model and the models with dynamic remeshing. The coordinates of the nodes are initially supplied for the first incremental analysis by a separate subroutine (or manually if necessary), and are subsequently read from the output data file of the previous incremental analysis. The element definition varies from one incremental analysis to another. Figure 6.4 shows the meshes used in nine incremental analyses for the same plate as is illustrated in Figure 6.1. Since no substructure is used, the total mesh is redefined each time remeshing is performed.

We selected the following procedure to determine the exact location of the boundaries of the local zone. In the case of this plate, the width of the local zone is kept constant as is shown in Figure 6.4. Suppose the total length of the plate is L, the length of the local zone l, and the overlapping factor γ, which is defined as the ratio between the length of the part of the local zone that overlaps the previous one and the length of the total local zone, as is shown in Figure 6.5. The local zone therefore translate a distance of $(1-\gamma)l$ from one incremental analysis to the next. The y coordinates of the boundaries in front of

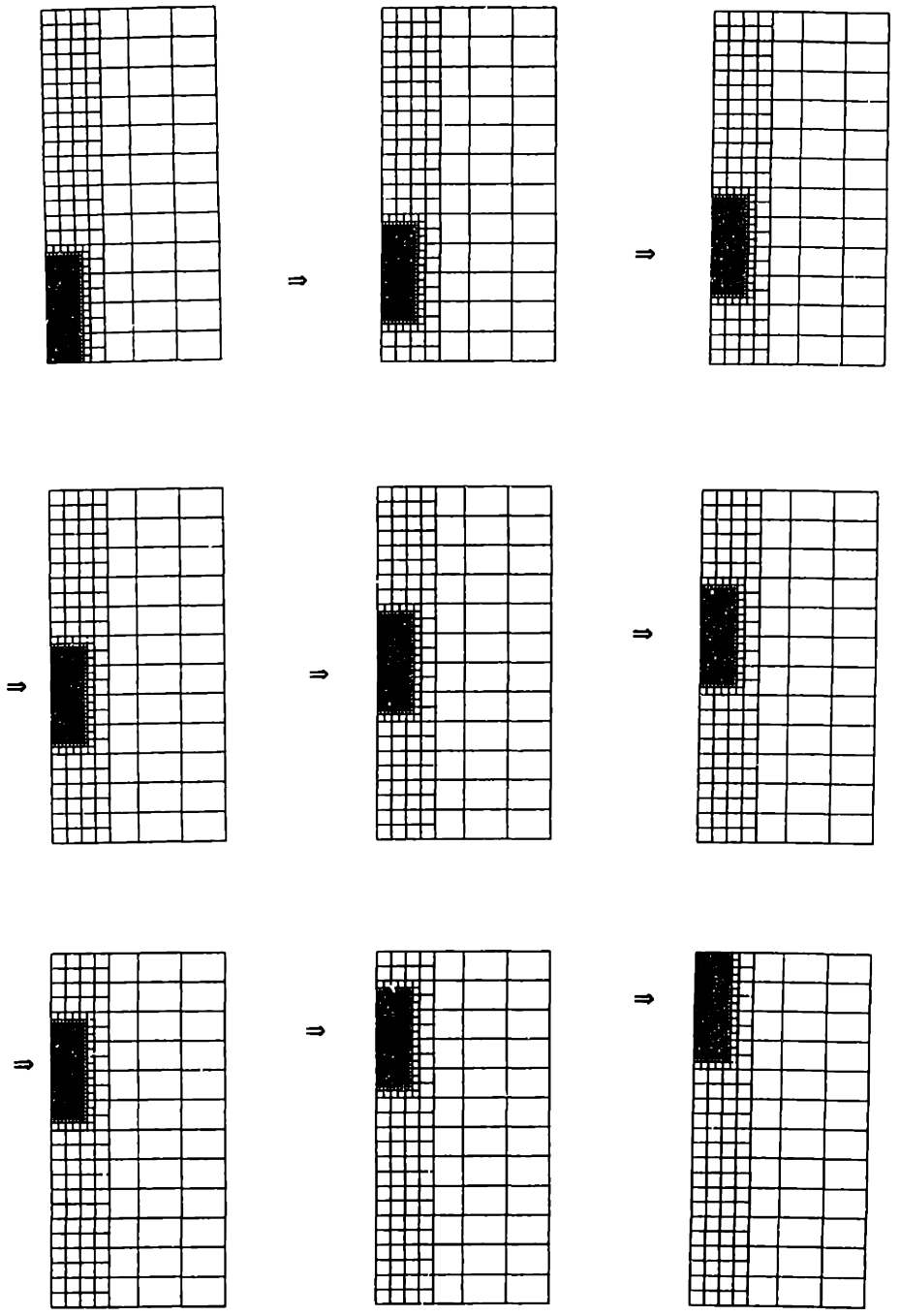


Figure 6.4: Different meshes for the incremental analyses

and behind the electrode are therefore:

$$\begin{aligned} y_1 &= [(i-1)(1-\gamma)]l \\ y_2 &= y_1 + l = [1 + (i-1)(1-\gamma)]l \end{aligned} \tag{6.5}$$

Here i indicates the ith incremental analysis.

Because the coordinates of all the nodes change throughout the process, we actually define the local zone boundary by layers of elements counted from the starting end of the dense mesh. Three factors are actually used; the total number of layers of elements n along the dense mesh, the number of incremental analyses m, and the overlapping factor γ. These three numbers uniquely determine the size of the local zone and the remeshing frequency required. Suppose the total number of element layers associated with the local zone is l_n. The number of layers that the local zone translate from one incremental analysis to another is therefore $(1-\gamma)l_n$; the following equation then holds:

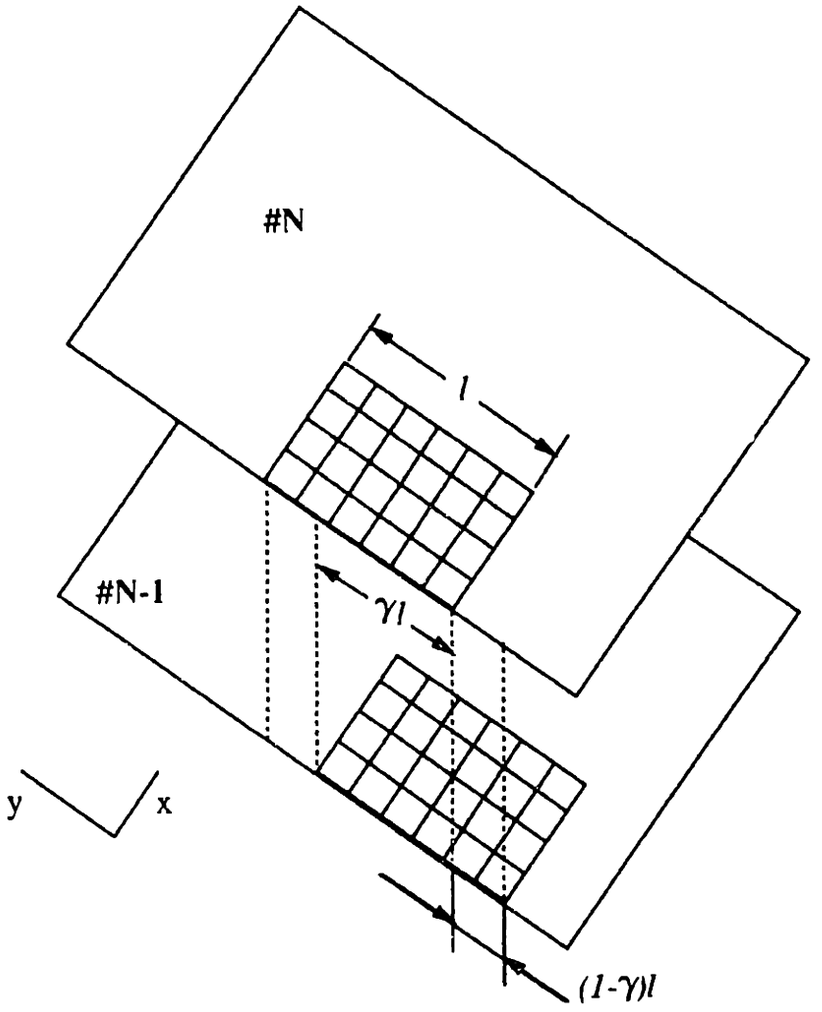
$$n = m(1-\gamma)l_n + \gamma l_n \tag{6.6}$$

Solving this equation for l_n, we have:

$$l_n = \frac{n}{m(1-\gamma)+\gamma}. \tag{6.7}$$

This equation indicates that the size of the local zone is determined by the three factors chosen. The layer numbers for the back and front boundaries can be derived as:

$$\begin{aligned} a_1 &= (i-1)(1-\gamma)l_n = (i-1)\frac{n}{m + \frac{\gamma}{1-\gamma}} \\ a_2 &= l_n + (i-1)(1-\gamma)l_n = (i + \frac{\gamma}{1-\gamma})\frac{n}{m + \frac{\gamma}{1-\gamma}} \end{aligned} \tag{6.8}$$



Traveling time between these two increments: $(1-\gamma)l/v$

Figure 6.5: Models of two consecutive incremental analyses and the definitions of overlapping factor γ

Because of the nonuniform mesh densities, multi-point constraints are required when the mesh density changes. The above two numbers are rounded to allow a smooth transition from a finer mesh to a coarser mesh and thus provide convenient definitions of multi-point constraints. In the case of substructuring, the gradual transition of mesh densities from the local zone to the substructure should be considered when rounding these two numbers.

The length of time elapsed in an incremental analysis depends on the distance d between the heat source and the front boundary of local zone at the time when the incremental analysis ends. This distance is determined according to the temperature profile, and usually is much less than l since the temperature gradients ahead of the heat source are very high. If the speed of the traveling heat source is v, as is shown in Figure 6.5:

$$\begin{aligned} t &= (l-d)/v && (i=1), \\ t &= (1-\gamma)l/v && (i=2,3,4...m-1), \\ t &= [(1-\gamma)l+d]/v && (i=m). \end{aligned} \tag{6.9}$$

The cooling part of the process after the heat source stops can be simulated in a separate incremental analysis based on either a new mesh or the one from the previous increment. The time span for the cooling part is relatively long in order to let the structure cool completely; it is usually about two hours or more depending on the geometry of the structure and the power input.

Other factors varying from one incremental analysis to another are the definitions of multi-point constraints, assignments of material properties to different parts of the structure, and the allocation of an element set for heat flux. All of them can be determined based on the parameters described above, such as a_1 and a_2.

The step-wise coupling of thermal and deformation analyses is accomplished by updating the coordinates of nodes at the beginning of an incremental thermal analysis. The new coordinates come from the previous deformation analysis. The coupling is well simulated if the displacements or the changes of nodal coordinates are relatively small within one incremental analysis, which is usually the case. As we discussed earlier, energy coupling between the thermal and mechanical activities is very small, and the major coupling effect comes from the deformation on the temperature field. The step-wise coupling is therefore adequate to simulate the actual coupling phenomena.

When plotting the deformed shape of a structure, we often magnify the nodal deformation in order to see the distortion clearly. In the case of dynamic remeshing, however, an incremental analysis only records the incremental displacements during the incremental period. The magnified distortion plot at the end of the last incremental analysis is therefore not possible since the displacement data do not take into account the previous values. It is therefore necessary to accumulate nodal displacements; once they are known, a distorted mesh can be plotted by magnifying the displacements, adding the original coordinates and then plotting the mesh of the 'new' model.

There are basically two solutions to this problem. The first involves keeping a separate record where the displacements of all nodes can be saved and accumulated from one incremental analysis to another. The second is reading the nodal coordinates at the end of the last incremental analysis and obtaining the nodal displacements by subtracting the original coordinates. The first method is obviously tedious, but it is effective when the structure is complicated and the original coordinates are not systematically assigned to all the nodes. The second is efficient particularly when the structure is simple and the original

coordinates of a node can be easily determined simply from its number.

Figure 6.6 shows the distorted shape of the plate after the displacements are magnified by the same factor as is used in Figure 6.2. No clear difference can be observed from these two plots. Small differences can be found if the displacement data are examined, but they are minor and can be ignored for engineering purpose. The close match between the distorted shapes of the full model and of the model using dynamic remeshing indicates that dynamic remeshing is effective and that the idea of singling out the local nonlinear weld zone is correct in simulations of welding processes.

There is not much difficulty in plotting the residual stress contours on the plate since the stress components are updated by interpolation when remeshing is performed. The final stress state at the end of the last incremental analysis represents the true residual stress state of the plate; the stress contours, therefore, can be plotted based on the stress data of the last incremental analysis. The normal stress component in the transverse direction of the plate is very small compared with the longitudinal component, as is the shear stress component. This is because the plate is virtually free from stress in the transverse direction, and thermal expansion or contraction only involves changes of normal components of strains. The longitudinal normal stress is the dominant element in this case. The Mises stress, which characterizes the magnitude of the deviatoric state of stress, is plotted in Figure 6.7.

Unlike the distortion plots, some differences can be clearly observed between Figures 6.3 and 6.7: The top parts of the two figures show very good resemblances, but the contours on the rest of the plate differ, particularly along the weld path. Such differences can

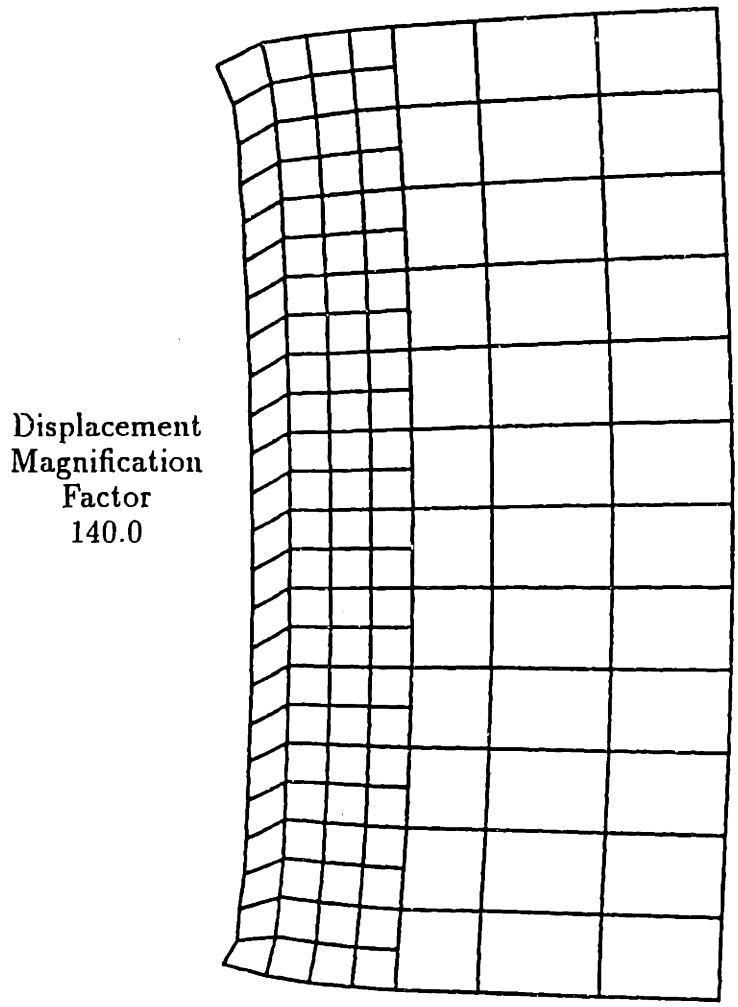


Figure 6.6: The final distorted shape of the plate predicted by FEM using dynamic remeshing, no substructure is used

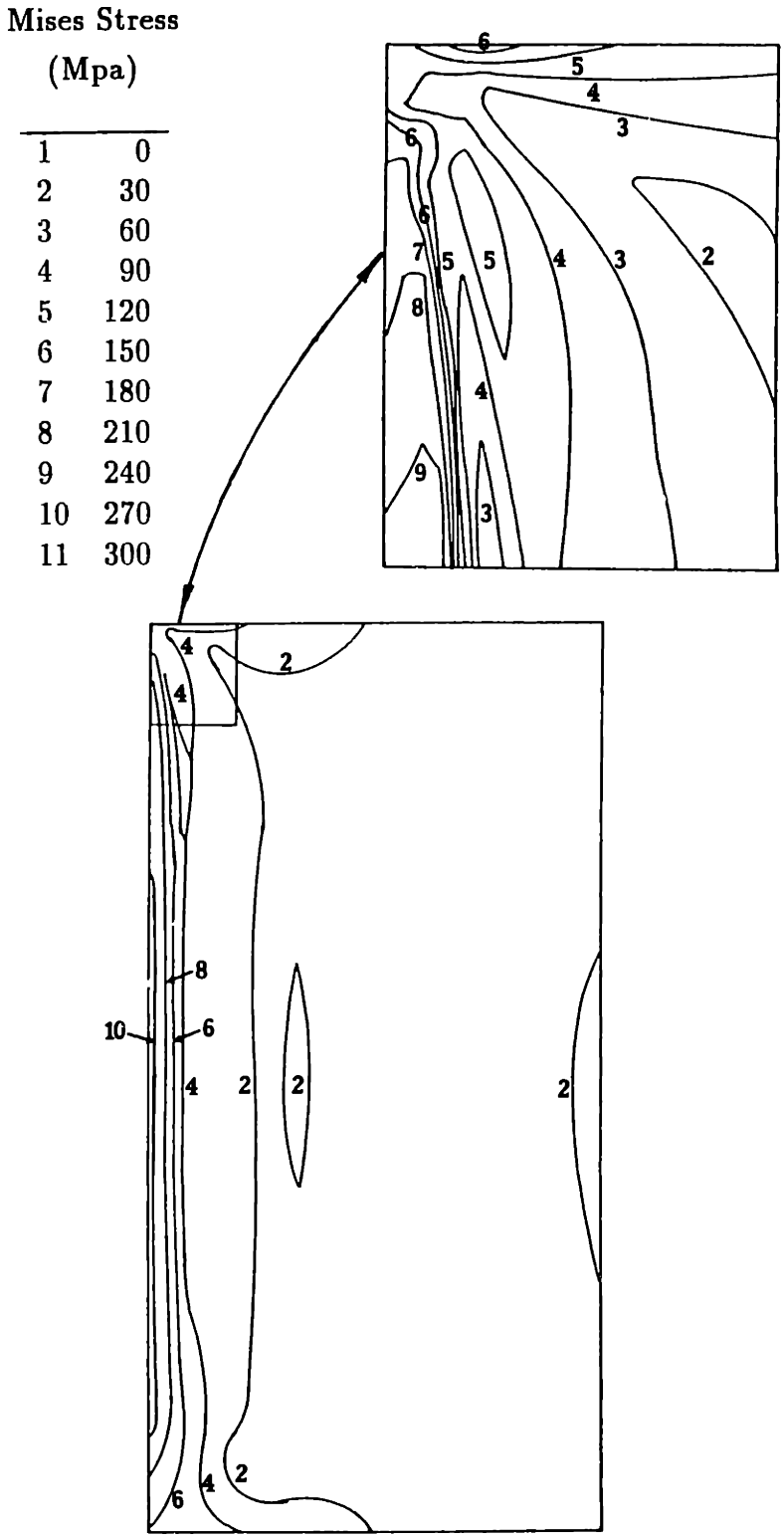


Figure 6.7: Final Mises stresses predicted by using dynamic remeshing, without substructure.

be explained by the different mesh densities used in the two models. In the full model, fine mesh is used all along the weld path; high gradients of both temperature and stress components can be captured by the fine grids of mesh. The model with dynamic remeshing, however, uses a much coarser mesh in the wake of the heat source. The high gradients of stress components will be lost in the coarse mesh as is shown in Figure 6.7.

On the one hand we need the reduction of the size of an FEM model and this implies a coarser mesh. A coarser mesh, however, could not capture the high gradients of variables such as stresses and strains. This conflict can only be solved by making compromises either on the size of the model or on the accuracy of the results. Fortunately, we usually need accurate stress or strain information only on a limited number of "spots" on a structure, and such conflicts could be solved by using a finer mesh only at the specific "spots" where we have to resolve the state of stress with accuracy. Careful comparisons of the two plots indicate the magnitudes of the stresses are very close, and that the only major differences are the stress gradients along the weld wake.

It should be noted that the dynamic remeshing procedure keeps the previous plastic deformation during remeshing but resets the plastic strain, or erases the strain hardening. This will not cause much inaccuracy if the strain hardening of the material is small, as is the case for mild steel. If the material hardening is significant, a state variable is required to map the plastic strain to the new model after remeshing.

The total time needed for the simulation with dynamic remeshing is about 170 hours on the VAXstation 3500 computer, which is equivalent to 34 hours on the Apollo DN10000. While the full model consumes a total of 200 hours on the Apollo for both the thermal

and mechanical simulations. The computer time needed for the case with dynamic remeshing therefore is reduced by a factor of six. Interestingly, the full model requires a more tolerant convergence criterion, otherwise the simulation could not finish simply because of convergence problems. If the same convergence tolerances were used for both cases, more reduction in computer time could be expected. Another interesting phenomenon is that the total numbers of degrees of freedom and the band widths of the stiffness matrices only show a reduction of about three times in computer time consumption, while the actual reduction is close to six. The running records of the analyses show difficulties of convergence in the full model; more iterations imply more processing time. Another possible explanation for the above-mentioned difference in computational time reductions is that when the total number of degrees of freedom is large, the size of data files are correspondingly large and the computer memory needed is also substantially larger; all disk I/O and memory swapping also need computer time.

The total time needed to carry out the remeshing only accounts for about 1 to 2% of the reduced time needed for the simulation, or less than 0.5% of the original time needed for the full model. This overhead cost of computer time can be generally neglected. A higher remeshing frequency can be used without increasing much of the overhead cost, but computer storage space will increase accordingly when the frequency goes up.

6.4 Dynamic Remeshing with Substructuring

The far field outside the local nonlinear zone of a structure can be very large, as it is in the case of the three-dimensional model of the ring-stiffened cylinder described in Chapter

4 where the total degrees of freedom of the far field is more than three times those of the local zone. The use of substructuring can condense the large number of degrees of freedom associated with the far field to only a very small number at the retained degrees of freedom. Therefore the size of the model is reduced virtually to the same as that of the local zone. The dynamic remeshing technique will be greatly enhanced by the use of substructuring for the rest of the structure.

There are some particular problems with the use of substructuring. The transient heat transfer process is inherently nonlinear, and substructuring cannot be applied to its simulations. Other strategies are needed to simplify the heat transfer analyses. Fortunately, the thermal model has only one-third of the total degrees of freedom of the same mechanical model and requires much less computer time.

The information about a substructure is saved only through the definition of its stiffness matrix; no stress or strain information is stored. The stress and strain distributions are uniquely determined by the displacements at the retained degrees of freedom since the substructure is linearly elastic. In the case of dynamic remeshing, initial stresses exist inside the substructure at the beginning of an incremental analysis. Special treatment is needed to add the initial stresses to the substructure.

No thermal expansion or contraction can be simulated inside a substructure, either. This implies that the boundary of substructures should be far away from the local weld zone to include thermal expansion and contraction wherever temperature changes are not negligible.

6.4.1 Initial Stresses in a Substructure

Substructures cannot accommodate initial stresses. At the end of the first increment, stresses build up in the substructures; we need an effective strategy to deal with such existing stresses in the next increment.

We assume that temperature changes and gradients inside the substructure are small so that thermal expansions or contractions are negligible. We also assume that there are no other nonlinearities in the substructure such as contact points, inelastic deformation, and nonlinear boundary conditions (if there are, however, the part with the nonlinearity can always be included in the local zone). The dynamic remeshing is frequent enough so that both the displacement and strain increments during one incremental analysis are very small.

The dynamic substructuring technique involves the redefinition of a substructure frequently. At each redefinition, the accumulated stresses inside a substructure need special attention since substructures can not accommodate initial stresses at the beginning of each incremental analysis. We use a special procedure to separate and record the accumulated stresses in a substructure before an incremental analysis. The stresses are updated after the incremental analysis by superimposing the incremental stresses to the accumulated. The following describes the details of the procedure.

As is shown in Figure 6.8, suppose that the stress state of the structure at the end of incremental analysis #N-1 is as shown (State 1). The stress state at the beginning of the next incremental analysis can be obtained by interpolation as described in the previous sections. At this stage (State 2), there are accumulated stresses in the new substructure,

which are denoted by σ_{sb}^{N}. Since the substructure is elastic, the state of stresses can be decomposed to two parts (State 3): the stressed local zone balanced by a stress-free substructure and the boundary forces along the substructure interface, and a stressed but otherwise identical substructure. The next incremental analysis can be performed on the first part while leaving the second part intact.

Figure 6.9 shows the superposition of stresses after incremental analysis #N. After the incremental analysis, stresses change in both the local zone and the substructure as show at State 4. The incremental stresses in the substructure ($\Delta\sigma_{s}^{N}$) are the results of incremental deformation due to thermal expansion and contraction in the local zone. This incremental stresses can also be decomposed and then superposed; the results are shown as State 5. The stress state all over the structure (State 6) can then be combined. This procedure enables the substructure in an incremental analysis to be initially stress-free, and avoids the complications of initial stresses in a substructure.

The boundary interaction forces can be taken from the traction forces based on the stress components along the boundary. Theoretically, this set of boundary forces on the local zone should be self-equilibrated since there are no other external forces. The actual data, however, show small residuals of both forces and moments due to roundoff errors and other numerical treatments such as extrapolations and interpolations. The magnitudes of such residuals are indeed small, but if left uncorrected, the accumulated residuals will act as a set of external forces on the rest of the structure and create a false stress field. Corrections on the components of forces of each node on the boundary are performed to counter-balance the effect of the residuals. Assume that the force components at each node is denoted as F_x and F_y, the corrections at this node as C_x and C_y, and the residual forces

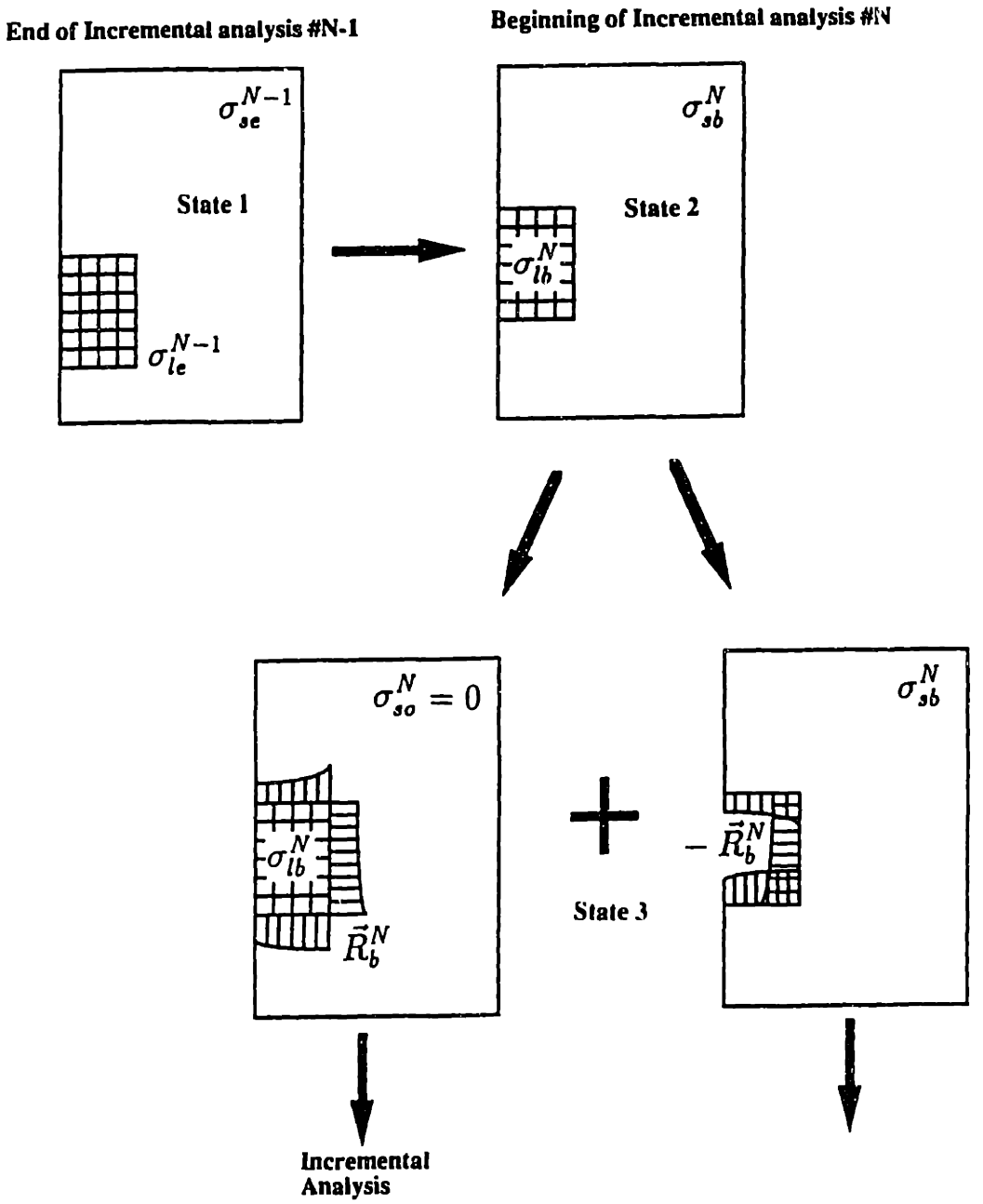


Figure 6.8: Initial stresses inside a substructure. The decomposition of the initial stress field and a initially stress-free substructure.

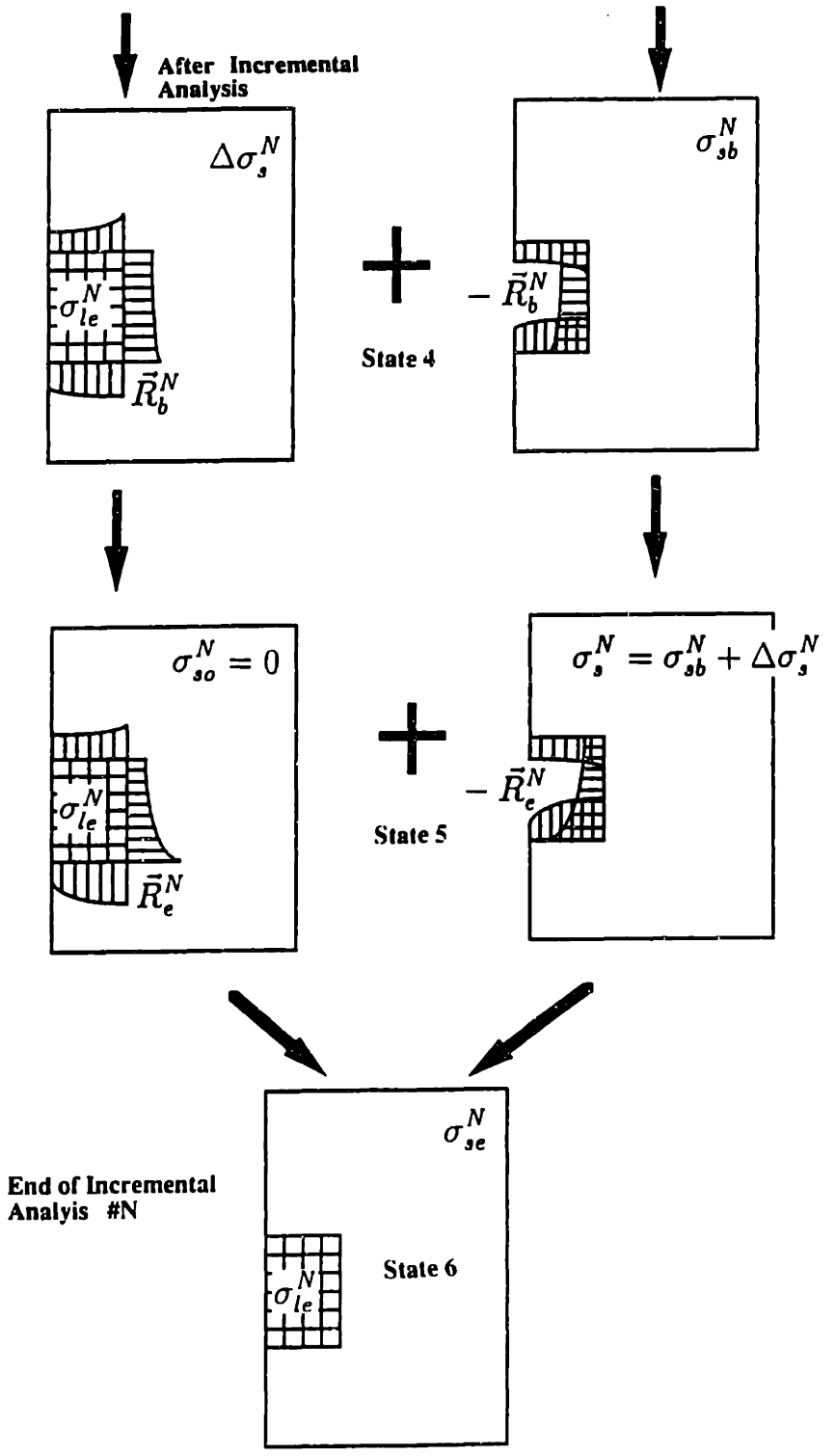


Figure 6.9: Initial stresses inside a substructure. The superimposition of the initial and incremental stress states in a substructure.

and moment (with respect to the origin) as R_x, R_y and R_m. The following correction functions can be used:

$$\begin{aligned} C_x^i &= (c_1 + c_2\frac{y}{l_e})(-\frac{R_x}{n_b}) \\ C_y^i &= (c_1 + c_3\frac{x}{l_e})(-\frac{R_y}{n_b}) \end{aligned} \tag{6.10}$$

Where i is the node number, l_e the characteristic length of one side of an element, and n_b the total number of nodes where degrees of freedom are retained; c_1, c_2 and c_3 are dimensionless constants to be determined. x and y denote the nodal coordinates. There are many other functions which can also be selected; we choose the above simply because they are effective and easier to implement. To balance the effect of the residuals, the following conditions must be met:

$$\begin{aligned} \sum F_x &= \sum_{i=1}^{n_b} C_x^i + R_x = 0 \\ \sum F_y &= \sum_{i=1}^{n_b} C_y^i + R_y = 0 \\ \sum M_z &= \sum_{i=1}^{n_b} (xC_y^i - yC_x^i) + R_m = 0 \end{aligned} \tag{6.11}$$

Substituting Eq. 6.10 into the above equations, we can solve them for the three constants:

$$c_1 = \frac{\frac{R_y \sum x^2}{\sum x} - \frac{R_x \sum y^2}{\sum y} - R_m}{\frac{R_x \sum y - R_y \sum x}{n_b} - \frac{R_x \sum y^2}{\sum y} + \frac{R_y \sum x^2}{\sum x}} \tag{6.12}$$

$$c_2 = \frac{n_b l_e}{\sum y}(1 - c_1) \tag{6.13}$$

$$c_3 = \frac{n_b l_e}{\sum x}(1 - c_1) \tag{6.14}$$

$\sum$ in the above equations represents the summation over all the nodes where degrees of freedom are retained. Once the constants are determined, Eq. 6.10 can be used to add

corrections to the forces on the local zone, which will ensure a self-equilibrated state of force. The actual corrections during a typical remeshing are within 2% of the original force magnitudes, and usually less than 1% which is negligible.

The results show that the above treatments on initial stress in a substructure and the corrections of force residuals are very effective.

6.4.2 Effect of Cooling Far behind the Local Zone

The first time we used substructures with dynamic remeshing, the substructure boundary was set the same as that of the local zone. Surprisingly, the final distorted shape of the plate was much different from that of the full model. Careful examination of the distorted mesh shows that there apparently is inadequate contraction inside the substructure and particularly in the wake of the weld path where the temperature is supposedly lower than 150 °C. The possible explanation for this phenomenon is that thermal contraction at low temperature (below 150 °C) is important and cannot be ignored as is the case in our model.

In order to avoid this problem, the boundary of the substructure is delimited where the temperature is lower than 75 °C. The temperatures in the wake of the weld turn out to be higher than 100 °C before the heat source reaches the end of its path. This implies that for this particular plate model, substructuring cannot be used behind the heat source; the local zone needs to extend to the starting point of the heat source in order to take the low temperature cooling into account. Such an extension of the local zone will increase the model dramatically, more than offsetting the gain of using substructuring. We solve this problem by using a coarser mesh behind the heat source, or a variable mesh density inside

the local zone, as is shown in Figure 6.10.

We wonder why the low temperature cooling makes that much difference. One plausible line of reasoning is that, at low temperatures, the material is stronger and stiffer, and any contraction will cause high stresses and more deformation at places where temperatures are higher and material is weaker. This rationale also suggests that low temperature cooling is very important in welding processes; more attention should be paid to the wake of a weld path where temperature may be low but still plays an important part in the deformation process. This is particularly important in the actual welding practices when welders use various strategies to control distortions.

6.4.3 Results and Comparison

The final distorted mesh is plotted in Figure 6.11. This final distortion pattern closely matches the one from the full model shown in Figure 6.2. The residual Mises stresses are plotted in Figure 6.12 which also shows a close match with its counterpart in Figure 6.3. The same phenomenon of stress gradient dispersion is also observed in the wake of the weld because of the coarser mesh. The remarkable resemblances in the distortion patterns and in the stress distributions between the results from the full model and the dynamic remeshing model with substructure support the idea of dynamic remeshing and its application in finite element simulations of welding processes.

The total time needed for the simulation with both dynamic remeshing and substructuring is close to 140 hours of CPU time on a VAXstation 3500, this gives a reduction of computer time by a factor of more than seven as compared with the full model simulation.

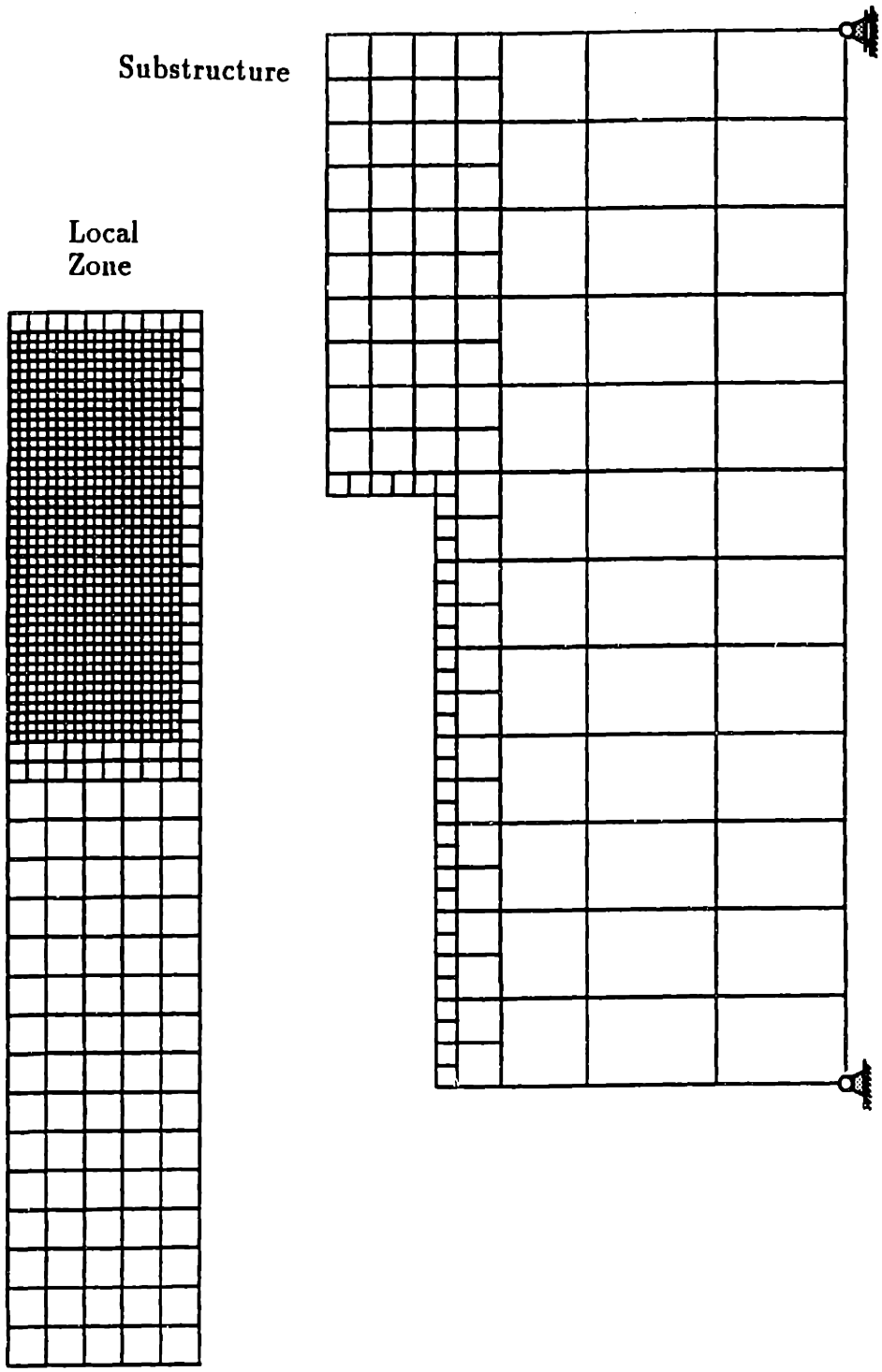


Figure 6.10: Mesh of the local zone and the substructure.

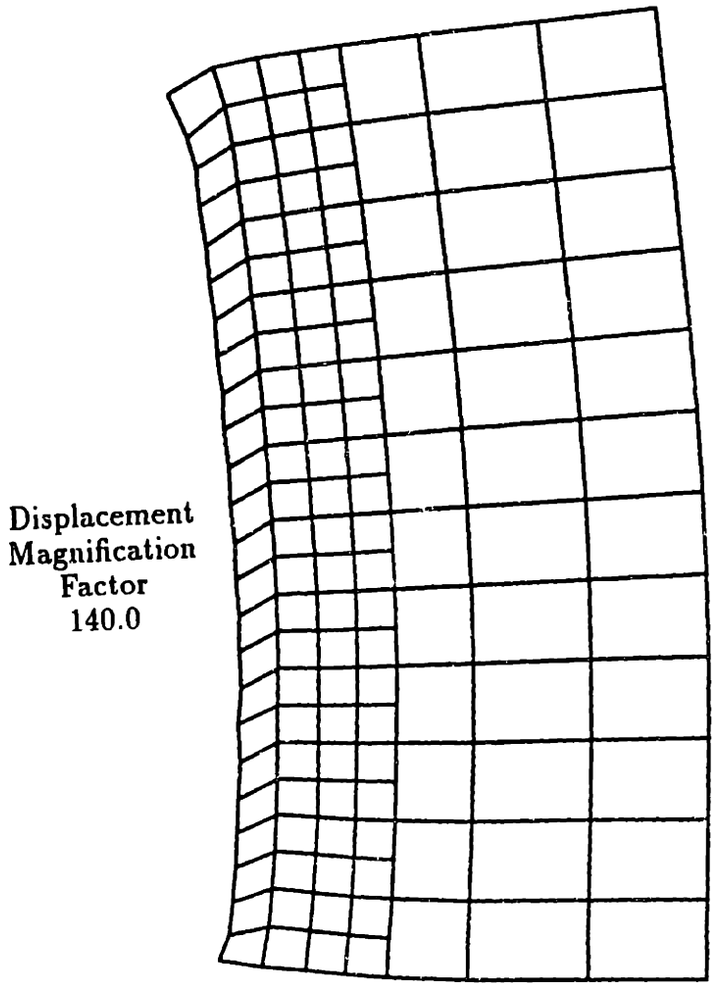


Figure 6.11: Final distortion predicted by using dynamic remeshing and substructuring.

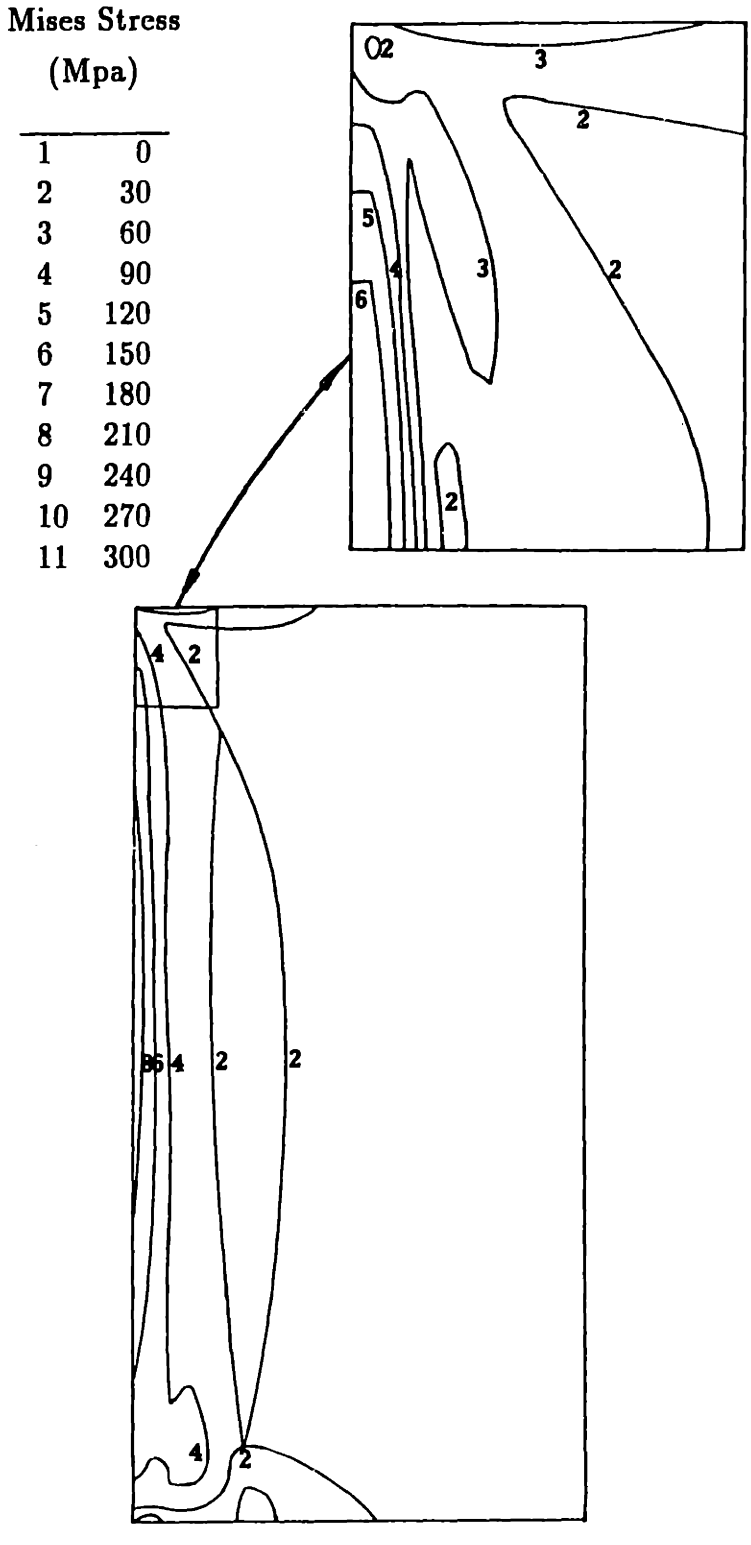


Figure 6.12: Residual Mises stresses predicted by using dynamic remeshing and substructuring.

A comparison of the total number of degrees of freedom and the band widths of the stiffness matrices for the full model with those for a typical model of an incremental analysis indicates a reduction of required computer time by a factor of five.

It should be noted that the plate model with a long slender weld path gives us the worst performance of dynamic remeshing, since the band width depends mainly on the number of nodes in the transverse direction and does not change much when the model includes a shorter, denser mesh. The best performance of dynamic remeshing can be obtained if the mesh is connected at the two ends, forming a circular structure. We performed one test in which the two ends of the plate are artificially connected. The band width of the full model shows an increase of about 80% which triples the computer time requirements. It can be seen that the computer time reduction of more than one order of magnitude can be easily achieved by using dynamic remeshing. Much higher reduction is also expected when the structure is larger and more complex.

Remarkable savings of computer storage spaces are also observed in dynamic remeshing both with and without substructures. The temperature file for the full model, which includes the temperature data for all the nodes and all the time increments, takes about 30 megbytes of disk space. The size of this single file is almost the same as the total size of all the files of the incremental analyses involved in dynamic remeshing. Moreover, many of the working files associated with an incremental analysis could be deleted during the analysis without creating any problem if storage space were limited.

6.5 Discussion

The dynamic remeshing technique is shown to be very effective. The substructuring technique will further enhance the power of dynamic remeshing, though it will need special treatment for the initial stresses. The idea of dividing the structure into a local nonlinear zone and a linear, isothermal substructure has proved to be effective and powerful. The codes we use in these simulations with dynamic remeshing are convenient for the simple plate geometry; they could be further expanded to include other welding situations. The dynamic remeshing will enable the simulations of welding processes to be performed on a widely available workstation.

A close examination of the temperature fields and the stress distributions shows remarkable resemblances between two consecutive incremental analyses. This implies that a quasi-static model could also be developed to further simplify the analyses, particularly when the heat source is in the middle of the plate and when the plate and weld path are very long. We will discuss suggestions for such a model in the next chapter.

Chapter 7

Conclusions and Suggestions for Future Work

The following major conclusions may be drawn from this thesis, and a number of guidlines concerning further work are suggested in finite element analyses of the structure interations in the welding processes.

This thesis first proved the importance of structure interactions during welding processes such as the fitup gaps and fixturing, and then point out both the necessity of three-dimensional models and the need of reduction in model size during welding simulations. This thesis used the dynamic remeshing and developed the dynamic substructuring technique in welding simulations. Both of these techniques reduce the size of a finite element model substantially for welding simulations.

7.1 Conclusions

First of all, we see the inadequacy of two-dimensional simulations of welding analysis. The actual welding situation is always three-dimensional; the deformation cannot be simulated

with a two-dimensional model unless it is a very special case such as a short plate, a very thin plate or a shell, or a tack welding where only a very short weld is deposited. Two-dimensional analyses are particularly inadequate for large structures with long welds and structures with asymmetric geometries or deformations.

Most welding situations involve asymmetric deformations. Even though the structure may be symmetric, the travelling heat source will cause asymmetric deformation as long as it travels from one place to another. Three-dimensional analyses are therefore important in welding simulations, particularly for the simulations of asymmetric displacements as is demonstrated in Chapter 4. The high demands on computational resources are a disadvantage of three-dimensional models and make most simulations intractable on existing computers. Successful three-dimensional analyses could not be achieved without a further reduction in the size of a finite element model.

The dynamic remeshing technique presented in this thesis can reduce substantially the size of the three-dimensional models for welding processes. The use of this technique together with substructuring will enable a reduction of computer time by more than an order of magnitude. These techniques could be much more effective if the structure under investigation is larger and more complex. The step-wise coupling of thermal and mechanical processes accomplished through the use of dynamic remeshing greatly increases its applicability to a welding situation which involves large deformations. Further modifications and expansion are expected before we can readily apply these techniques to the simulation of an arbitrary welding situation.

The dynamic remeshing and the substructuring techniques are not limited to their

applications in welding simulations. Any other problem involving a small nonlinear zone traveling in a volume of linear medium can also profit from these techniques in numerical simulations. Dynamic fracture mechanics, fluid flows, and flame bending of sheet metals all involve moving, highly nonlinear zones and are potential fields where dynamic remeshing and substructuring techniques find applications.

7.2 Future Work

The expansion of the dynamic remeshing technique requires many different tasks. For the technique to apply to arbitrary welding situations, formulations of variable interpolations need to be developed for many different types of elements. The most difficult task is to derive the formulations for special elements such as spring and interface elements. Efficient solution strategies for high order simultaneous equations are also needed in determining the normalized coordinates of a spatial point in an element.

There are basically two generic strategies for carrying out dynamic remeshing in welding simulations. One is to develop a universal software package which will work on any arbitrary geometry and welding parameter. The other is to develop several models based on different generic common geometries such as plate, pipe, and T-shaped structures.

The universal model should have the weld path pre-defined before the actual simulation starts. The rest of the structure outside this weld path can be treated as substructures, and the weld path will guide the dynamic remeshing or the movements of the local zone. The optimal goal of this universal model should be its applicability to any shape of weld path whether straight or curved. The applications of dynamic remeshing through generic

geometries are easier but more tedious to develop. Almost all the structure weldings can be grouped from different geometries; these are usually straight weld, as on a plate, circular welds as in a pipe or ring, T-shaped structures, and deep-groove multi-path welds. Different packages of codes can be developed to fit the different geometries. The actual developments of the computer codes may include different subroutines to generate nodes and elements for the different geometries, since the basic remeshing strategies are the same.

To reduce the size of the heat transfer model, equivalent lumps of masses at the boundaries of the local zone can be added to represent the effect of the outside part of the structure on the local temperature fields. Other strategies are also available such as equivalent thermal masses and sinks. A thermal analysis could then be performed on the local zone, and the computational effort would be greatly reduced. As we noticed in the plate welding simulations, both the transient temperature and stress fields for two consecutive incremental analyses show remarkable similarities. Many of the intermediate analyses could have been omitted with a quasi-static model; while the analyses at the ends of the weld path need close examination. A mechnical quasi-static model may be difficult to develop; a thermal model is clearly possible when deformation is relatively small as in the case of the plate welding. A similar situation could also be expected when a circular weld path such as a pipe butt-welding or the welding on ring structures is involved.

These are only preliminary ideas. Further problems may arise during the development of a particular strategy or model; changes or modifications are indeed expected.

Finally, work in the area of material behavior at high temperatures is strongly suggested. Much of the material behavior is still poorly understood; micro-structure evolutions

and their effects on material properties as well as the rate dependency of material constitutive relations remain to be explored. Accurate results from numerical simulations of manufacturing processes require more accurate material characterizations, particularly at high temperatures.

Bibliography

[1] K. Masubuchi. *Analysis of welded structures.* Pergamon Press, 1980.

[2] D. Rosentall. Theory of moving source of heat and its application to metal treatment. *Transactions of ASME*, Nov. 1946.

[3] M.B. Hsu. Analysis of welds by finite element method. *Numerical Modeling of Manufacturing Process*, 1977.

[4] D.W. Lobitz, J.D. McClure, and R.E. Nickell. Residual stresses and distortion in multipass welding. *Numerical Modeling of Manufacturing Process*, Feb. 1977.

[5] E.F. Rybicki, D.W. Schmueser, and R.B. Stonesifer. A finite element model for residual stresses in girth-butt welded pipes. *Numerical Modeling of Manufacturing Process*, 1977.

[6] F.W. Brust and E.F. Rybicki. A computational model of backlay welding for controlling residual stresses in welded pipes. *Journal of Pressure Vessel Technology*, Aug 1981.

[7] E. Friedman. Numerical simulation of the gas tungsten-arc welding process. *Numerical Modeling of Manufacturing Process*, 1977.

[8] M. Jonsson, L. Karlsson, and L-E. Lindgren. Simulation of tack welding procedures in butt joint welding of plates. *Welding Journal*, Oct. 1985.

[9] A.P. Chakravarti, J.A. Goldak, and A.S. Rao. Prediction of thermal history of repair welds. *Welding for challenging environments*, 1986.

[10] B.L. Josefson and R.I. Karlsson. Finite element analysis of in-plane deformations and stresses in restrained butt welded plates. *Supplied by Author*, 1988.

[11] L. Josefson, M. Jonsson, L. Karlsson, T. Karlsson, R. Karlsson, and L-E. Lindgren. Transient and residual stresses in a single-pass butt welded pipe. *Second International Conference on Residual Stresses, Nancy, France*, Nov. 1988.

[12] J. Goldak, M. McDill, A. Oddy, R. House, X. Chi, and M. Bibby. Computational heat transfer for weld mechanics. *International Conference on Trends in Welding Research, Gatlinburg, Tennessee, USA*, May 1986.

[13] P. Tekriwal and J. Mazumder. Finite element analysis of three-dimensional transient heat transfer in gma welding. *Welding Journal*, pages 150s–156s, July 1988.

[14] R.I. Karlsson and B.L. Josefson. Three-dimensional finite element analysis of temperature and stresses in a single-pass butt welded pipe. *Supplied by Author*, 1988.

[15] K.W. Mahin, Wm. Winters, J. Krafcik, T. Holden, R . Hosbons, and S. MacEwen. Residual strain distributions in gas tungsten arc welds: a comparison between experimental results and weld model predictions. In S.A. David and J.M. Vitek, editors, *Recent Trends in Welding Science and Technology*. ASM International, May 1989.

[16] V.I. Makhnenko, V.M. Shekera, and L.A. Izebenko. Special features of the distribution of stresses and strains caused by making circumferential welds in cylindrical shells. *Automatic Welding*, Dec 1970.

[17] R.H. Leggatt. Residual stresses at girth welds in pipes. *Welding in Energy-related Projects*, 1984.

[18] H. Song. Prediction of distortion in welding ring stiffened cylinders. Master's thesis, Massachusetts Institute of Technology, Sept. 1988.

[19] P.J. Cacciatore. Analytical modeling of heat flow and structural distortion in ship structure produced by welding. Technical Report U443-79-085, Office of Naval Research, 1977.

[20] J.D. White and J.B. Dwight. Weld shrinkage in large stiffened tubular. *Residual Stresses in Welded Construction and Their Effects*, 1978.

[21] J. Goldak, A. Oddy, M. McDill, and A. Chakravarti. Progress in computing residual stress and strain in welds. In *International Conference on Trends in Welding Research, Gatlinburg, Tennessee, USA*, May 1986.

[22] K.E. Easterling. Predicting heat-affected zone microstructures and properties in fusion welds. *Advances in Welding Science and Technology*, 1986.

[23] K. Easterling. *Introduction to the Physical Metallurgy of Welding*. Butterworths, 1983.

[24] Stuart B. Brown and Haoshi Song. Finite element analyses of welding process. *To be sent for publication.*

[25] B. Machuca. Distortion and residual stresses in a welded stiffened ring. Master's thesis, Massachusetts Institute of Technology, June 1988.

[26] B.A. Boley and J.H. Weiner. *Theory of Thermal Stresses.* Robert E. Krieger Publishing Company, 1985.

[27] M.L. Lin and T.W. Eagar. Influence of arc pressure on weld pool geometry. *Welding Journal*, pages 163s–169s, June 1985.

[28] M.L. Lin and T.W. Eagar. Pressures produced by gas tungsten arcs. *Metallurgical Transactions B*, 17B:601–607, Sept. 1986.

[29] N.S. Tsai and T. W. Eagar. Distribution of the heat and current fluxes in gas tungsten arcs. *Metallurgical Transactions B*, 16B:841–846, Dec. 1985.

[30] R.W. Lewis and K. Morgan. *Numerical Methods in Heat Transfer*, volume 3. John Willey & Sons, 1985.

[31] W.M. Rohsenow and J.P. Hartnett. *Handbook of Heat Transfer.* McGraw-Hill Book Company, 1973.

[32] J. Goldak, M. Bibby, J. Moore, R. House, and B. Patel. Computer modeling of heat flow in welds. *Metallurgical Transactions B*, Sept. 1986.

[33] Y. Iwamura and E.F. Rybicki. A transient elastic plastic thermal stress analysis of flame forming. *Journal of Engineering for Industry*, Feb. 1973.

[34] S. Brown, L. Anand, and K. H. Kim. An internal variable constitutive model for the hot working of metals. *International Journal of Plasticity*, March 1989.

[35] K.E. Easterling. Modeling the weldability of structural steels. *Welding Metallurgy of Structural Steels*, 1987.

[36] S. Denis, S. Sjostrom, and A. Simon. Coupled temperature, stress, phase transformation calculation model. application to the prediction of phase transformation during cooling of an eutectold carbon steel. *Residual Stresses in Science and Technology*, 1987.

[37] K. Bathe. *Finite Element Procedures in Engineering Analysis.* Prentice-Hall, Inc., 1982.

[38] Robert D. Cook. *Concepts and Applications of Finite Element Analysis.* John Wiley & Sons, 1981.

[39] P. Tekriwal and J. Mazumder. Finite element modeling of arc welding processes. *Advances in Welding Science and Technology*, May 1986.

[40] J. Goldak. Modeling thermal stresses and distortions in welds. *Second International Conference on Trends in Welding Research, Gatlinburg, Tennessee, USA*, May 1989.

[41] B.A.B. Andersson. Thermal stresses in a submerged-arc welded joint considering phase transformation. *Journal of Engineering Materials and Technology*, 100:356–362, October 1978.

[42] J. Goldak, A. Chakravarti, and M. Bibby. A new finite element model for welding heat sources. *Metallurgical Transactions B*, June 1984.

[43] John Robinson. *Integrated Theory of Finite Element Methods.* John Wiley & Sons, 1973.

[44] William W. Hager. *Applied Numerical Linear Algebra.* Prentice Hall, 1988.

[45] Jung-Ho Cheng. Automatic adaptive remeshing for finite element simulation of forming processes. *International Journal for Numerical Methods in Engineering*, 26, 1988.

[46] Toshihisa Nishioka and Yutaka Takemoto. Moving finite element method aided by computerized symolic manipulation and its application to dynamic fracture simulation. *JSME International Journal*, 32(3), 1989.

[47] V. Murti and S. Valliappan. Numerical inverse isoparametric mapping in remeshing and nodal quantity contouring. *Computers and Structures*, 22(6), 1986.

[48] Y. Ueda, Y. C. Kim, and M. G. Yuan. A predicting method of welding residual stress using source of residual stress (report 1). *Transactions of JWRI*, 18(1):135–141, 1989.

[49] K.W. Mahin, A.B. Shapiro, and J. Hallquist. Assessment of boundary condition limitations on the development of a general computer model for fusion welding. In *International Conference on Trends in Welding Research, Gatlinburg, Tennessee, USA*, May 1986.

[50] T. Zacharia, A.H. Eraslan, and D.K. Aidun. Modeling of non-autogenous welding. *Welding Journal*, pages 18s–27s, Jan. 1988.

[51] T. Zacharia, A.H. Eraslan, and D.K. Aidun. Modeling of autogenous welding. *Welding Journal*, pages 53s–62s, March 1988.

[52] E. W. Kim. *Analyses of Resistance Spot Welding Lobe Curve.* PhD thesis, Massachusetts Institute of Technology, June 1989.

[53] E. W. Kim and T. W. Egar. Measurement of transient temperature response during resistance spot welding. *The Welding Journal*, August 1989.

[54] Stuart B. Brown and Haoshi Song. Finite element simulation of welding of large structures. *Submitted to Journal of Engineering for Industry*, September 1989.

[55] Stuart B. Brown and Haoshi Song. Implications of 3-d numerical simulations of welding of large structures. In *1990 AWS International Exposition and 71st Annual Convention*, Anaheim, CA, April 1990.

[56] Stuart B. Brown and Haoshi Song. Dynamic remeshing and substructuring in welding simulations. *To be sent for publication*, 1990.

[57] H.D. Hibbitt and P.V. Marcal. A numerical thermomechanical model for the welding and subsequent loading of a fabricated structure. *Computer and structures*, 1973.

[58] E. Friedman. Thermomechanical analysis of the welding process using the finite element method. *Journal of Pressure Vessel Technology*, Aug. 1975.

[59] E.F. Rybicki and R.B. Stonesifer. Computation of residual stresses due to multipass welds in piping system. *Journal of Pressure Vessel Technology*, May 1979.

[60] B.L. Josefson. Residual stresses and their redistribution during annealing of a girth-butt welded thin walled pipes. *Journal of Pressure Vessel Technology*, Aug 1982.

[61] S. Nair, E. Pang, and R.C. Dix. Residual stress generation and relaxation in but-welded pipes. *Journal of Pressure Vessel*, Feb 1982.

[62] J.H. Argyris, J. Szimmat, and K.J. Willam. Computational aspects of welding stress analysis. *Computer Methods in Applied Mechanics and Engineering*, 33:635–666, 1982.

[63] K.W. Mahin, S. MacEwen, W. Winters, W. Mason, M. Kanouff, and E.A. Fuchs. Evaluation of residual stress distributions in a traveling gta weld using finite element and experimental techniques. In A. F. Giamei and G.J. Abbaschian, editors, *Modeling of Casting and Welding Processes IV*. The minerals, Metals & Materials Society, 1988.

[64] C.L. Tsai. Computer-aided welding design. *Welding Journal*, June 1988.

[65] C. D. Allemand, R. Schoeder, D.E. Ries, and T.W. Eagar. A method of filming metal transfer in welding arcs. *Welding Journal*, pages 45–47, Jan. 1985.

[66] G.J. Dunn and T.W. Eagar. Metal vapors in gas tungsten arcs: part ii. theoretical calculations of transport properties. *Metallurgical Transactions A*, 17A:1865–1871, oct. 1986.

[67] G.J. Dunn, C.D. Allemand, and T. W. Eagar. Metal vapors in gas tungsten arcs: part i. spectroscopy and monochromatic photography. *Metallurgical Transactions A*, 17A:1851–1863, Oct. 1986.

[68] S.A. Gedeon, C. D. Sorensen, K.T. Ulrich, and T.W. Egar. Measurement of dynamic electrical and mechanical properties of resistance spot welds. *Welding Journal*, pages 378s–385s, Dec. 1987.

[69] G.M. Oreper, J. Szekely, and T.W. Eagar. The role of transient convection in the melting and solidification in arc weld pools. *Metallurgical Transactions B*, 17B:735–744, Dec. 1986.

[70] V.J. Papazoglou and K. Masubuchi. Numerical analysis of thermal stresses during welding including phase transformation effects. *Journal of Pressure Vessel Technology*, 104:198–203, August 1982.

[71] Y. Ueda and K. Nakacho. Calculation and measurement of three-dimensional welding residual stresses. *The Effect of Fabrication Related Stresses on Product Manufacturing and Performance*, Sept. 1985.

[72] Y. Ueda, K. Fukuda, and M Tanigawa. New measuring method of three dimensional residual stresses based on theory of inherent strain. *Trasactions of JWRI*, (2), 1979.

[73] Y. Ueda and K. Fukuda. Simplified measuring methods of three dimensional welding residual stresses. *Transactions of JWRI*, (2), 1982.

[74] Y. Ueda, K. Nakacho, and S Moriyama. Simple prediction methods for welding deflection and residual stress of stiffened panels. *Transactions of JWRI*, (2), 1986.

Appendix A

Lists of Papers in Welding Simulations and Related Areas

There are many papers dealing with numerical simulations of welding processes. They can be grouped in different categories such as thermal analyses, two- and three-dimensional mechanical analyses, and other related areas. The author has done an extensive computerized literature search and lists the related papers in different categories here for reference.

Table A.1: Representative thermal analyses of welding process

Year	Author	Description	Reference
1977	Hsu	Pipe welding	[3]
1984	Goldak et. al	Modeling the heat input	[42]
1985	Jonsson et. al	Butt welding of plate (2-D in plane), tack welding	[8]
1986	Goldak et. al	2-D analysis on plate butt welding	[32]
1986	Mahin et. al	2-D and 3-D heat transfer analyses	[49]
1988	Song	2-D axisymmetric	[18]
1988	Tekriwal & Mazumder	3-D FEM analysis on butt welding of plate	[13]
1988	Zacharia et. al	Mass flow and thermal analysis of GTA	[50] [51]
1989	Kim & Eagar	2-D, resistance spot welding, experimental measurements	[52] [53]
1989	Brown & Song	2-D and 3-D thermal analyses	[54]
1990		3-D Ring-stiffened cylinder	[55]
1990		2-D Dynamic substructuring	[56]

Table A.2: Chronological list of selected 2-D numerical analyses on welding stresses

Year	Investigators	Notes	References
1973	Hibbitt & Marcal	Axisymmetric analysis	[57]
1975	Friedman	Plate butt welding, 2-D plain strain	[58]
1977	Rybicki et. al.	Axisymmetric two pass butt welding	[5]
1979	Rybicki & Stonesfer	Pipes, multi-pass welding	[59]
1981	Brust & Rybicki	Pipes	[6]
1982	Josefson	Pipes, including heat treatment	[60]
1982	Nair et. al.	Pipes, including heat treatment	[61]
1985	Jonsson et. al.	Tack welding, and in-plane analysis of plate butt welding (Plane stress)	[8]
1986	Chakravarti et. al.	Plane strain, plate welding	[9]
1988	Song	Cylinders, Axisymmetric analysis	[18]
1988	Josefson & Karlsson	In plane analysis of plate welding	[10]
1989	Brown & Song	Axisymmetric, ring stiffener	[54]
1989	Kim	Axisymmetric, Spot Welding	[52]
1989	Mahin et al.	2-D plate welding, coupled analysis	[15]
1989	Brown & Song	2-D Plate, dynamic substructuring	[56]

Table A.3: Chronological list of 3-D numerical investigations of stress during welding

Year	Investigators	Notes	References
1982	Argyris et al.	Plate welding	[62]
1986	Goldak et al.	Tack welding	[21]
1986	Tekriwal & Mazumder	Plate butt welding	[39]
1988	Karlssong & Josefson	Cylinder butt welding	[14]
1988	Mahin et al.	Plate	[63]
1990	Brown & Song	Ring stiffener	[55]

Table A.4: Papers in related research areas

Year	Investigators	Notes	References
1985	Tsai and Eagar	Mass and heat flow in welding	[64]
1986	Lin and Eagar	Mass and heat flow in welding	[27]
1985	Lin and Eagar	Mass and heat flow in welding	[27, 28]
1986		and arc pressure	
1985	Allemand et al.	Mass and heat flow in welding	[65]
1986	Dunn and Eagar	Mass and heat flow in welding	[66]
1986	Dunn Et al.	Mass and heat flow in welding	[67]
1987	Gedeon et al.	Measurement of dynamic	[68]
		properties in weld	
1986	Goldak et al.	Heat flux in arc	[32]
1986	Oreper et al.	Mass and heat transfer in welds	[69]
1982	Papazoglou and Masubuchi	Phase transformation	[70]
1985	Ueda and Nakacho	Multi-pass welding	[71]
1982	Ueda et al.	Multi-pass welding	[72, 73]
1986			[74]

Appendix B

List of Computer Programs and Subroutines

The computer codes used in this thesis are listed here as references. The codes are written in Fortran and Digital Commands Language (DCL) on a VAXstation 3500 running VMS; modifications are expected for other machines running different operating systems.

B.1 User Subroutines for Heat Flux and Boundary Heat Losses

B.1.1 3-D Model

```
C
**The user FILM and DFLUX subroutines
C

      SUBROUTINE FILM(H,SINK,TEMP,KSTEP,KINC,TIME,NOEL,NPT,COCRDS,JLTYP)
C
C     This subroutine calculate the temperature-dependent boundary
C     convection coefficient for 3-d FEM analyses.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COORDS(3),TEMPER(12),HC(6,12)
C
C     Set discrete temperatures.
```

```
C
      TEMPER(1)=0.0
      TEMPER(2)=50.0
      TEMPER(3)=100.0
      TEMPER(4)=200.0
      TEMPER(5)=300.0
      TEMPER(6)=500.0
      TEMPER(7)=750.0
      TEMPER(8)=1000.0
      TEMPER(9)=1600.0
      TEMPER(10)=5000.0
      TEMPER(11)=15000.0
C
C     Set discrete values of convection coefficient.
C
      HC(1,1)=2.5
      HC(1,2)=4.143
      HC(1,3)=5.0
      HC(1,4)=5.93
      HC(1,5)=6.57
      HC(1,6)=7.286
      HC(1,7)=7.93
      HC(1,8)=8.286
      HC(1,9)=9.0
      HC(1,10)=10.0
      HC(1,11)=10.0
      HC(2,1)=2.5
      HC(2,2)=4.643
      HC(2,3)=5.714
      HC(2,4)=6.786
      HC(2,5)=7.429
      HC(2,6)=8.071
      HC(2,7)=8.5
      HC(2,8)=8.786
      HC(2,9)=9.143
      HC(2,10)=10.0
      HC(2,11)=10.0
      DO 10 I=1,11
      HC(3,I)=HC(1,I)*0.517/0.54
      HC(4,I)=HC(3,I)
```

```
      HC(5,I)=HC(3,I)
      HC(6,I)=HC(3,I)
10    CONTINUE
C
C     The convention for element face numbering is:
C             1---Down                    Nu=0.54*SQRT(SQRT(Pr*Gr))
C             2---Up                      Nu=0.137*(Pr*Gr)**(1/3)
C             3---vertical                Nu=0.517*SQRT(SQRT(Pr*Gr))
C             4---Vertical
C             5---Vertical
C             6---Vertical
C
      JFACE=JLTYP-10
C
      SINK=20.0
C     Determine temperature range.
C
      DO 20 I=1,10
      TEMP1=TEMP-SINK
      IF (TEMP1.GE.TEMPER(I)) THEN
            II=I+1
      ENDIF
20    CONTINUE
C
      H=(HC(JFACE,II)-HC(JFACE,II-1))
      H=H/(TEMPER(II)-TEMPER(II-1))
      H=(TEMP-TEMPER(II-1))*H+HC(JFACE,II-1)
C
      RETURN
      END

      SUBROUTINE DFLUX(FLUX,TEMP,KSTEP,KINC,TIME,NOEL,NPT,COORDS,JLTYPE)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COORDS(3),COORDS1(3)
C
C     Power=4.5 KW
C     Speed=6.25 in/min, or 0.0026458333 m/sec
C
      POWER=9.675E+7
      SPEED=0.0026458333
```

```
      SIGMA=2.5
C*************************************************************************
C     Center of electrode (in polar system, radial, circumferential, and
C                                                  axial)
C*************************************************************************
      OFFSET=0.25
      X1CENTER=0.4572+0.25*0.0254*(0.25-OFFSET)
      Y1CENTER=SPEED*TIME+0.0319185
      Z1CENTER=0.003175-0.25*0.0254*(0.25-OFFSET)
C************************************
C     Transfer coordinates to polar system
C************************************
      COORDS1(1)=SQRT(COORDS(1)*COORDS(1)+COORDS(2)*COORDS(2))
      COORDS1(2)=COORDS1(1)*ASIN(COORDS(2)/COORDS1(1))
      COORDS1(3)=COORDS(3)
C************************************
      IF (JLTYPE.EQ.14) GO TO 400
      IF (JLTYPE.EQ.13) GO TO 300
      IF (JLTYPE.EQ.12) GO TO 200
C************************************
C     On web, Face 2
C************************************
200   CONTINUE
      X=ABS(COORDS1(1)-X1CENTER)/0.00635
      Y=ABS(COORDS1(2)-Y1CENTER)/0.00635
      IF (Y .GT. 1.25) GO TO 290
210   R=X*X+Y*Y
      FLUX=POWER*EXP(-SIGMA*R)
      GO TO 1000
290   FLUX=0.0
      GO TO 1000
C************************************
C     On web, Face 3 (6 node element)
C************************************
300   CONTINUE
      X=ABS(COORDS1(1)-X1CENTER)/0.00635
      Y=ABS(COORDS1(2)-Y1CENTER)/0.00635
      IF (Y .GT. 1.25) GO TO 390
      R=X*X+Y*Y
      FLUX=POWER*EXP(-SIGMA*R)
```

```
      GO TO 1000
390   FLUX=0.0
      GO TO 1000
C**************************************
C     On flange, Face 4
C**************************************
400   CONTINUE
      Z=(COORDS1(3)-Z1CENTER)/0.00635
      Y=ABS(COORDS1(2)-Y1CENTER)/0.00635
      IF (Y .GT. 1.25) GO TO 490
      R=Z*Z+Y*Y
      FLUX=POWER*EXP(-SIGMA*R)
      GO TO 1000
490   FLUX=0.0
1000  CONTINUE
      IF (TIME .LT. 2.4) THEN
          T=TIME/2.4
          FLUX=FLUX*((1-EXP(-10*T*T))
     1                +EXP(-10000*T*T)*SIN((T-0.5)*2*3.1415926))
      ELSE
      ENDIF
      IF (Y1CENTER.GT.0.0635) THEN
           Y1=(Y1CENTER-0.0635)/0.00635
           FLUX=FLUX*EXP(-2.5*Y1*Y1)
      ELSE
      ENDIF
      RETURN
      END
```

B.1.2 2-D Plate Model

```
C
C     User subroutine for heat flux and boundary heat loss.
C
C  Convection and radiation are modeled as negative volumetric heat flux.
C
      SUBROUTINE DFLUX(FLUX,TEMP,KSTEP,KINC,TIME,NOEL,NPT,COORDS,JLTYPE)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COORDS(3),TEMPER(12),HC(2,12)
```

```
C     Power=0.6 KW
C     Speed=2 mm/sec,
C     Plate thickness: 0.01 m.
C
      SPEED=0.002D0
        TIME0=0.0D0
      IF (JLTYPE.EQ.1) GO TO 1000
      YCENTER1=TIME0*SPEED
      SIGMA=3.0D0
      R=0.0025D0
      P=1.1392177D0*600.0D0/R/0.01D0
      YCENTER=-0.0025D0+TIME*0.002D0+YCENTER1
      POS=DABS(COORDS(2)-YCENTER)/0.0025D0
      IF (POS.GT.1.25D0) GO TO 100
      FLUX=P*DEXP(-SIGMA*POS*POS)
      GO TO 2000
100   CONTINUE
      FLUX=0.0D0
      GO TO 2000
C*************************************************
1000  CONTINUE
C
C     Defining the boundary conditions.
C
C     This subroutine calculate the temperature dependent free convection
C     coefficient for 2-d FEM analysis.
C
C     Set discrete temperatures.
C
      TEMPER(1)=0.0D0
      TEMPER(2)=50.0D0
      TEMPER(3)=100.0D0
      TEMPER(4)=200.0D0
      TEMPER(5)=300.0D0
      TEMPER(6)=500.0D0
      TEMPER(7)=750.0D0
      TEMPER(8)=1000.0D0
      TEMPER(9)=1600.0D0
      TEMPER(10)=5000.0D0
      TEMPER(11)=15000.0D0
```

```
C
C     Set discrete values of convection coefficient.
C
      HC(1,1)=2.5D0
      HC(1,2)=4.393D0
      HC(1,3)=5.357D0
      HC(1,4)=6.404D0
      HC(1,5)=7.0D0
      HC(1,6)=7.679D0
      HC(1,7)=8.215D0
      HC(1,8)=8.536D0
      HC(1,9)=9.072D0
      HC(1,10)=10.0D0
      HC(1,11)=10.0D0
      HC(2,1)=0.0D0
      HC(2,2)=0.2D0
      HC(2,3)=0.4D0
      HC(2,4)=0.45D0
      HC(2,5)=0.475D0
      HC(2,6)=0.54D0
      HC(2,7)=0.58D0
      HC(2,8)=0.59D0
      HC(2,9)=0.6D0
      HC(2,10)=0.65D0
      HC(2,11)=0.7D0
      SB=5.6696D-8
C
C  H(1,I)----Convection,
C            (0.54*SQRT(SQRT(Pr*Gr))+0.137*(Pr*Gr)**(1/3))/2
C            Q=H(1,I)*(T-SINK)
C  H(2,I)----Radiation,
C            Q=H(2,I)*SB*(T**4-SINK**4)
C            SB: the Stefan-Boltzman constant.
C**************************************
C The value is divided by the thickness to obtain the volumetric flux.
C**************************************
C
      SINK=20.0D0
C     Determine temperature range.
C
```

```
      TEMP1=TEMP-SINK
      IF (TEMP1.LT.0D0) THEN
          FLUX=0.0D0
          GO TO 2000
      ENDIF
      DO 20 I=1,10
      IF (TEMP1.GE.TEMPER(I)) THEN
             II=I+1
      ENDIF
20    CONTINUE
C
      H=(HC(1,II)-HC(1,II-1))
      H=H/(TEMPER(II)-TEMPER(II-1))
      H=(TEMP-TEMPER(II-1))*H+HC(1,II-1)
      H1=(HC(2,II)-HC(2,II-1))
      H1=H1/(TEMPER(II)-TEMPER(II-1))
      H1=(TEMP-TEMPER(II-1))*H1+HC(2,II-1)
      FLUX=H*(TEMP-SINK)+H1*SB*(TEMP**4D0-SINK**4D0)
      FLUX=-FLUX/0.01D0
C*************************************************
C
2000  CONTINUE
      RETURN
      END
```

B.2 Programs Used for Dynamic Remeshing

The following programs and subroutines are used during the dynamic remeshing procedures. Note that all the files are in the directory **disk$user:[song.fem.msub.shell.coupled]** when substructuring is not used. In the case of substructuring, files related to heat transfer analyses are in the directory **disk$user:[song.fem.msub.shell.thermal]** and those related to deformation analyses in **disk$user:[song.fem.msub.shell.stress]**. These sub-directories should be changed to apply all these programs and subroutines in other places.

B.2.1 The Shell programs

Dynamic Remeshing without Substructuring

```
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!    This shell program drives the heat transfer analyses and the
$!    deformation analyses alternately. Working files are stored in
$!    the subdirectory disk$user:[song.fem.msub.shell.coupled].
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!
$ SET NOVERIFY
$ SET NOON
$ ON CONTROL_Y THEN GOTO LAB11
$ ON ERROR THEN GOTO LAB11
$ ON SEVERE_ERROR THEN GOTO LAB11
!
! SET LOGICAL ASSIGNMENTS FOR YOUR SCRATCH DISK AND THE LOCATION OF THE
! ABAQUS PRE, SHARE AND MAIN MODULES:
!
$ SCRATCH := $1$DUA1:[SCRATCH]
$ PRE     := $1$DUAO:[ABAQUS]ABQ7PRE
$ SHR     := $1$DUAO:[ABAQUS]ABQ7SHR
$ MAIN    := $1$DUAO:[ABAQUS]ABQ7LIB
$ INPUT   := DISK$USER:[SONG.FEM.MSUB.SHELL.COUPLED]INPUTFILE.COM
!
! SET LOGICAL ASSIGNMENTS FOR YOUR STARTUP FILE
!
$ STARTUP := $1$DUAO:[ABAQUS.STP]STARTUP.STP
!
! SET LOGICAL ASSIGNMENTS FOR YOUR BATCH QUEUES
! DELETE QUEUES NOT USED
! SUBMIT DEFAULTS TO SYS$BATCH
!
$ SLOW    := SLOW$BATCH
$ FAST    := FAST$BATCH
$ BIG     := BIG$BATCH
$ SYS     := SYS$BATCH
$ MULTI   := MULTI$BATCH
$ PR      := PR$BATCH
$ JOIN    := JOINER$BATCH
```

```
!
!  PARAMETERS 'P1'  JOB IDENTIFIER,MUST BE UNIQUE FROM OTHER
!                   JOBS THAT MAY EXECUTE SIMULTANEUOSLY
!!!              'P2'  ABAQUS DATA INPUT FILE
!!!              'P3'  FORTRAN SOURCE CODE OF USER SUBROUTINES
!!!              'P4'  RESTART READ FILE
!!!              'P5'  OLD FILE IF THIS IS A RESTART READ RUN
!!!              'P6'  SET TO NON-BLANK TO FINISH AFTER DATA CHECK
!
$ LPB2:
$ INQUIRE P10 -
  " IDENTIFIER                                                    "
$ IF P10 .NES. "" THEN  GOTO LPB1
$ WRITE SYS$OUTPUT " JOB IDENTIFIER CANNOT BE BLANK "
$ GOTO LPB2
$ LPB1:
$ C1 = F$SEARCH("''P10'.COM")
$ IF C1 .EQS. "" THEN GOTO LC1
$! DEL/LOG/NOCONFIRM 'P10'.COM;*
$ LC1:
$ INQUIRE TOTALNUM -
  " INPUT NUMBER OF RUNS NEEDED FOR PROBLEM                       "
$ INQUIRE SEQUENCENUM -
  " INPUT NUMBER OF LAST RUN                                      "
$ INQUIRE ENDNUM -
  " INPUT NUMBER WHERE YOU WANT TO STOP                           "
$ INQUIRE STARTTHERMAL -
  " DO YOU WANT TO START FROM THERMAL ANALYSIS (Y/N)              "
$ OPEN/WRITE COM_F 'P10'.COM
$ ADIR=F$ENVIROMENT("DEFAULT")
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ SET VERIFY"
$ WRITE COM_F "$ P9 = """" "
$ WRITE COM_F "$ IF F$GETDVI(""_XQAO:"",""EXISTS"") THEN P9 = ""XQAO:"" "
$ WRITE COM_F "$ IF F$GETDVI(""_XEAO:"",""EXISTS"") THEN P9 = ""XEAO:"" "
$ WRITE COM_F "$ IF F$GETDVI(""_ESAO:"",""EXISTS"") THEN P9 = ""ESAO:"" "
$ WRITE COM_F "$ IF F$GETDVI(""_XNAO:"",""EXISTS"") THEN P9 = ""XNAO:"" "
$ WRITE COM_F "$ IF P9 .EQS. """" THEN GOTO LPBA"
$ WRITE COM_F "$ ASSIGN 'P9' ETAO:"
$ WRITE COM_F "$ LPBA:"
```

```
$ WRITE COM_F "$ SET DEF ''ADIR'"
$ WRITE COM_F "$ SET VERIFY"
!!*********************************************************************
$SHELLLOOP:
$ P0:='P10''SEQUENCENUM'
$ SEQUENCENUM=SEQUENCENUM+1
$ P2:='P10''SEQUENCENUM'
$ WRITE COM_F "$ @''INPUT' ''P10' ''SEQUENCENUM' ''TOTALNUM'"
$ P1:='P2'
$ P0T:='P0'T
$ P1T:='P1'T
$ P2T:='P2'T
$ P0S:='P0'S
$ P1S:='P1'S
$ P2S:='P2'S
$ IF STARTTHERMAL .EQS. "N" THEN GOTO STRESSSTART
!!*********************************************************************
!!    THERMAL ANALYSIS
!!*********************************************************************
$ FSCR:='SCRATCH''P1T'
$ WRITE COM_F "$ F1 = F$SEARCH(""''P2T'.INP"")"
$ LPB3A:=LPB3T
$ LPB3B:='LPB3A''SEQUENCENUM'
$ WRITE COM_F "$ IF F1 .NES. """" THEN GOTO ''LPB3B'
$ WRITE COM_F -
  "$ WRITE SYS$OUTPUT "" CANNOT FIND ''P2T'.INP. CHECK THAT IT EXISTS""""
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LPB3B': "
$ WRITE COM_F "$ ASS ''P2T'.INP   FOR005"
$ WRITE COM_F "$ ASS ''P1T'.023   FOR023"
$ WRITE COM_F "$ ASS ''P1T'.028   FOR028"
$ WRITE COM_F "$ ASS ''P1T'.006   FOR006"
$ WRITE COM_F "$ ASS ''P1T'.FIL   FOR008"
$ WRITE COM_F "$ ASS ''P1T'.PPL   FOR013"
$ WRITE COM_F "$ ASS ''P1T'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1T'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1T'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1T'.018   FOR018"
$ WRITE COM_F "$ ASS ''P1T'.SUP   FOR020"
!**** ASSIGN SCRATCH FILES ***
```

```
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''STARTUP'    FOR042"
!*** ASSIGN RESTART WRITE FILE ***
$ WRITE COM_F "$ ASS ''P1T'.RES   FOR012"
!*** ASSIGN USER INPUT FILES ***
$ WRITE COM_F "$ ASS ''P1T'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1T'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1T'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1T'.018   FOR018"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''PRE'"
$ LAB7AA:=LAB7AT
$ LAB7AB:='LAB7AA''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB7AB'"
$ WRITE COM_F -
  "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING ABAQUS PRE PROGRAM ***"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1T'.006  ''P1T'.DAT"
$ WRITE COM_F "$ DELETE ''P1T'.006;*"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB7AB':"
$ WRITE COM_F "$ SET ON "
$ LAB7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1T'.006  ''P1T'.DAT"
$ WRITE COM_F "$ DELETE ''P1T'.006;*"
$ WRITE COM_F "$ SHOW PROC/ACC"
$ WRITE COM_F "$ SET ON "
```

```
!$ IF P6 .EQS. "Y" THEN GOTO LEND
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ FOR ''P1T'.028"
$ WRITE COM_F "$ SET ON "
$ LS7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ LAB9:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
! *** ASSIGN MAIN PLOT FILE ***
$ WRITE COM_F "$ ASS ''P1T'.MPL FOR013"
! *** ASSIGN MAIN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.001   FOR001"
$ WRITE COM_F "$ ASS ''FSCR'.002   FOR002"
$ WRITE COM_F "$ ASS ''FSCR'.003   FOR003"
$ WRITE COM_F "$ ASS ''FSCR'.004   FOR004"
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.019   FOR019"
$ WRITE COM_F "$ ASS ''P1T'.SUP     FOR020"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''P1T'.STA     FOR041"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ LINK/NOMAP/EXE=''P1T' ''P1T','MAIN'/LIB,-"
$ WRITE COM_F "''SHR'/LIB"
$ LAB12A:=LAB12T
$ LAB12B:='LAB12A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB12B'"
$ WRITE COM_F "$ IF $SEVERITY .EQS. 0  THEN GOTO ''LAB12B'"
```

```
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN LINKING MAIN ABAQUS LIBRARY *** "" "
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ ''LAB12B':"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''P1T'"
$ LAB14A:=LAB14T
$ LAB14B:='LAB14A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB14B'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING MAIN ABAQUS PROGRAM ***"
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB14B':"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ APP ''P1T'.006 ''P1T'.DAT"
$ WRITE COM_F "$ DELETE ''P1T'.006;*"
$ WRITE COM_F "$ SET ON "
$ LEND:
!$ WRITE COM_F "$ LEXIT:"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ F1 = F$SEARCH(""''P1T'.EXE"") "
$ WRITE COM_F "$ IF F1 .NES. """" THEN DELETE ''P1T'.EXE;*"
$ WRITE COM_F "$ F2 = F$SEARCH(""''P1T'.OBJ"") "
$ WRITE COM_F "$ IF F2 .NES. """" THEN DELETE ''P1T'.OBJ;*"
$ WRITE COM_F "$ F4 = F$SEARCH(""''P1T'.LIS"")"
$ WRITE COM_F "$ IF F4 .NES. """" THEN DELETE ''P1T'.LIS;*"
$ WRITE COM_F "$ F5 = F$SEARCH(""'P1T'.023"")"
$ WRITE COM_F "$ IF F5 .NES. """" THEN DELETE ''P1T'.023;*"
$ WRITE COM_F "$ F6 = F$SEARCH(""'P1T'.028"")"
$ WRITE COM_F "$ IF  F6 .NES. """" THEN DELETE ''P1T'.028;*"
!$ WRITE COM_F "$ LAB11:"
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$STRESSSTART:
$ STARTTHERMAL:=Y
$ WRITE COM_F "$ RENAME ''P1T'.FIL ''P1S'.015"
$ WRITE COM_F "$ RENAME ''P1T'.018 ''P1S'.018"
$ WRITE COM_F "$ RENAME ''P1T'.017 ''P1S'.017"
```

```
$ IF SEQUENCENUM .NES. 1 THEN WRITE COM_F "$ RENAME ''P1T'.016 ''P1S'.016"
$ WRITE COM_F "$ @STEPS.COM"
$ WRITE COM_F "$ APPEND STEPS.DAT ''P1S'.INP"
$ WRITE COM_F "$ DEL/NOCONFIRM STEPS.DAT;*"
$ WRITE COM_F -
  "$ IF ''SEQUENCENUM' .NES. 1 THEN APPEND USERS.FOR ''P1S'.INP"
$ WRITE COM_F "$ DEL/NOCONFIRM INPUT.001;*"
!!*****************************************************************
!!    STRESS/STRAIN ANALYSIS
!!*****************************************************************
$ FSCR:='SCRATCH''P1S'
$ WRITE COM_F "$ F1 = F$SEARCH(""''P2S'.INP"")"
$ LPB3A:=LPB3S
$ LPB3B:='LPB3A''SEQUENCENUM'
$ WRITE COM_F "$ IF F1 .NES. """" THEN GOTO ''LPB3B'
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" CANNOT FIND ''P2S'.INP. CHECK THAT IT EXISTS""""
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LPB3B': "
$ WRITE COM_F "$ ASS ''P2S'.INP   FOR005"
$ WRITE COM_F "$ ASS ''P1S'.023   FOR023"
$ WRITE COM_F "$ ASS ''P1S'.028   FOR028"
$ WRITE COM_F "$ ASS ''P1S'.006   FOR006"
$ WRITE COM_F "$ ASS ''P1S'.FIL   FOR008"
$ WRITE COM_F "$ ASS ''P1S'.PPL   FOR013"
$ WRITE COM_F "$ ASS ''P1S'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1S'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1S'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1S'.018   FOR018"
$ WRITE COM_F "$ ASS ''P1S'.SUP   FOR020"
$ WRITE COM_F "$ ASS ''P1S'.090   FOR090"
$ WRITE COM_F "$ ASS ''P1S'.091   FOR091"
$ WRITE COM_F "$ ASS ''POS'.ELM   FOR092"
!**** ASSIGN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
```

```
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''STARTUP'    FOR042"
!*** ASSIGN RESTART WRITE FILE ***
$ WRITE COM_F "$ ASS ''P1S'.RES   FOR012"
!*** ASSIGN RESTART READ FILE ***
$ WRITE COM_F "$ ASS ''POS'.RES   FOR011"
!*** ASSIGN USER INPUT FILES ***
$ WRITE COM_F "$ ASS ''P1S'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1S'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1S'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1S'.018   FOR018"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''PRE'"
$ LAB7AA:=LAB7AS
$ LAB7AB:='LAB7AA''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB7AB'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING ABAQUS PRE PROGRAM ***"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1S'.006  ''P1S'.DAT"
$ WRITE COM_F "$ DELETE ''P1S'.006;*"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB7AB':"
$ WRITE COM_F "$ SET ON "
$ LAB7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1S'.006  ''P1S'.DAT"
$ WRITE COM_F "$ DELETE ''P1S'.006;*"
$ WRITE COM_F "$ SHOW PROC/ACC"
$ WRITE COM_F "$ SET ON "
!$ IF P6 .EQS. "Y" THEN GOTO LEND
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$! WRITE COM_F "$ @MOD28 ''P1S'"
```

```
$ WRITE COM_F "$ FOR ''P1S'.028"
$ WRITE COM_F "$ SET ON "
$ LS7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ LAB9:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
! *** ASSIGN MAIN PLOT FILE ***
$ WRITE COM_F "$ ASS ''P1S'.MPL FOR013"
! *** ASSIGN MAIN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.001   FOR001"
$ WRITE COM_F "$ ASS ''FSCR'.002   FOR002"
$ WRITE COM_F "$ ASS ''FSCR'.003   FOR003"
$ WRITE COM_F "$ ASS ''FSCR'.004   FOR004"
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.019   FOR019"
$ WRITE COM_F "$ ASS ''P1S'.SUF     FOR020"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''P1S'.STA     FOR041"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ LINK/NOMAP/EXE=''P1S' ''P1S',''MAIN'/LIB,-"
$ WRITE COM_F "''SHR'/LIB"
$ LAB12A:=LAB12S
$ LAB12B:='LAB12A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB12B'"
$ WRITE COM_F "$ IF $SEVERITY .EQS. 0  THEN GOTO ''LAB12B'"
$ WRITE COM_F -
  "$ WRITE SYS$OUTPUT "" *** ERROR IN LINKING MAIN ABAQUS LIBRARY *** "" "
$ WRITE COM_F "$ GOTO GRANDEXIT"
```

```
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ ''LAB12B':"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''P1S'"
$ LAB14A:=LAB14S
$ LAB14B:='LAB14A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB14B'"
$ WRITE COM_F -
  "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING MAIN ABAQUS PROGRAM ***"
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB14B':"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ APP ''P1S'.006 ''P1S'.DAT"
$ WRITE COM_F "$ DELETE ''P1S'.006;*"
$ WRITE COM_F "$ SET ON "
$ LEND:
!$ WRITE COM_F "$ LEXIT:"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ F1 = F$SEARCH(""''P1S'.EXE"") "
$ WRITE COM_F "$ IF F1 .NES. """" THEN DELETE ''P1S'.EXE;*"
$ WRITE COM_F "$ F2 = F$SEARCH(""''P1S'.OBJ"") "
$ WRITE COM_F "$ IF F2 .NES. """" THEN DELETE ''P1S'.OBJ;*"
$ WRITE COM_F "$ F4 = F$SEARCH(""''P1S'.LIS"")"
$ WRITE COM_F "$ IF F4 .NES. """" THEN DELETE ''P1S'.LIS;*"
$ WRITE COM_F "$ F5 = F$SEARCH(""''P1S'.023"")"
$ WRITE COM_F "$ IF F5 .NES. """" THEN DELETE ''P1S'.023;*"
$ WRITE COM_F "$ F6 = F$SEARCH(""''P1S'.028"")"
$ WRITE COM_F "$ IF  F6 .NES. """" THEN DELETE ''P1S'.028;*"
!$ WRITE COM_F "$ LAB11:"
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ IF SEQUENCENUM .LT. ENDNUM THEN GOTO SHELLLOOP
$ WRITE COM_F "$GRANDEXIT:"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ F1 = F$SEARCH(""''P10'*.EXE"") "
$ WRITE COM_F "$ IF F1 .NES. """" THEN DELETE ''P10'*.EXE;*"
$ WRITE COM_F "$ F2 = F$SEARCH(""''P10'*.OBJ"") "
$ WRITE COM_F "$ IF F2 .NES. """" THEN DELETE ''P10'*.OBJ;*"
$ WRITE COM_F "$ F4 = F$SEARCH(""''P10'*.LIS"")"
```

```
$ WRITE COM_F "$ IF F4 .NES. """" THEN DELETE ''P10'*.LIS;*"
$ WRITE COM_F "$ F5 = F$SEARCH(""''P10'*.023"")"
$ WRITE COM_F "$ IF F5 .NES. """" THEN DELETE ''P10'*.023;*"
$ WRITE COM_F "$ F6 = F$SEARCH(""''P10'*.028"")"
$ WRITE COM_F "$ IF  F6 .NES. """" THEN DELETE ''P10'*.028;*"
$ WRITE COM_F "$ EXIT:"
$ CLOSE COM_F
!!!!!!!!!!!!!!!!!!!!!!!
!! Added by H.S. to organize queues automatically
!!!!!!!!!!!!!!!!!!!!!!!
$ REPEAT:
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT -
 " PLEASE choose a proper queue by using the following numbers:"
$! WRITE SYS$OUTPUT " "
$! esc[0,8]=27
$! write sys$output "  "
$! write sys$output "''esc'#3''esc'[1mATTENTION!!"
$! write sys$output "''esc'#4ATTENTION!!''esc'[0m"
$ write sys$output "  "
$ write sys$output -
 "   ------------------------------------------------------------"
$ WRITE SYS$OUTPUT -
 "       NODE::QUEUE       MAXCPU  MAXMEMORY  JOBLIMIT  PRIORITY"
$ write sys$output -
 "   ------------------------------------------------------------"
$ WRITE SYS$OUTPUT -
 "0: ULYSSES::FAST$BATCH,   5 MIN    0.26 Mb       1         4"
$ WRITE SYS$OUTPUT -
 "1: ULYSSES::SYS$BATCH,     2 H     0.52 Mb       1         3"
$ WRITE SYS$OUTPUT -
 "2: ULYSSES::SLOW$BATCH,    6 H     0.79 Mb       1         2"
$ WRITE SYS$OUTPUT -
 "3: ULYSSES::BIG$BATCH,     NO      2.10 Mb       2         1"
$ WRITE SYS$OUTPUT -
 "M: ULYSSES::MULTI$BATCH,   6 H     0.52 Mb       1         1"
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT -
 "4: JOINER::JOINER$BATCH,   NO      2.10 Mb       2         3"
```

```
$ write sys$output -
 "   ------------------------------------------------------"
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT " "
$ inquire QUE
$ IF QUE .EQS. "" THEN GOTO REPEAT
$!
$ WRITE SYS$OUTPUT " GIVE THE AMOUNT OF TIME YOU WANT TO HOLD THE JOB FOR"
$ INQUIRE HOLD -
  " THE FORMAT IS HH:MM: ( <RET> TO IGNORE)                    "
$ IF QUE .EQS. "M"
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD10
$ SUBMIT/NOTIFY/QUEUE=MULTI$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD10:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=MULTI$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG  'P10'
$ ENDIF
$ IF HOLD .NES. "" THEN GOTO HOLD11
$ SUBMIT/NOTIFY/QUEUE=PR$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD11:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=PR$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG  'P10'
$ ENDIF
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ IF QUE .EQS. 4
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD4
$ SUBMIT/NOTIFY/QUEUE=JOINER$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD4:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=JOINER$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG  'P10'
$ ENDIF
$ IF QUE .EQS. 3
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD2
$ SUBMIT/NOTIFY/QUEUE=BIG$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
```

```
$ GOTO LEXIT
$ HOLD2:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=BIG$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG  'P10'
$ ENDIF
$ IF QUE .EQS. 2
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD3
$ SUBMIT/NOTIFY/QUEUE=SLOW$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD3:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=SLOW$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$ IF QUE .EQS. 1
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD4
$ SUBMIT/NOTIFY/QUEUE=SYS$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD4:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=SYS$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$ IF QUE .EQS. 0
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD5
$ SUBMIT/NOTIFY/QUEUE=FAST$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD5:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=FAST$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG  'P10'
$ ENDIF
$ LAB11:
$ LEXIT:
$ EXIT:
```

Dynamic Remeshing with Substructuring

```
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!    This shell program drives the heat transfer analyses and the
```

```
$!    deformation analyses alternately. Files associated with thermal
$!    analyses are stored in disk$user:[song.fem.msub.shell.thermal],
$!    and those associated with the deformation analyses are in the
$!    subdirectory disk$user:[song.fem.msub.shell.stress].
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!
$ SET NOVERIFY
$ SET NOON
$ ON CONTROL_Y THEN GOTO LAB11
$ ON ERROR THEN GOTO LAB11
$ ON SEVERE_ERROR THEN GOTO LAB11
!
! SET LOGICAL ASSIGNMENTS FOR YOUR SCRATCH DISK AND THE LOCATION OF THE
! ABAQUS PRE, SHARE AND MAIN MODULES:
!
$ SCRATCH := $1$DUA1:[SCRATCH]
$ PRE     := $1$DUA0:[ABAQUS]ABQ7PRE
$ SHR     := $1$DUA0:[ABAQUS]ABQ7SHR
$ MAIN    := $1$DUA0:[ABAQUS]ABQ7LIB
$ INPUTS  := DISK$USER:[SONG.FEM.MSUB.SHELL.STRESS]INPUTFILE.COM
$ INPUTS1 := DISK$USER:[SONG.FEM.MSUB.SHELL.STRESS]INPUTFILE1.COM
$ INPUTT  := DISK$USER:[SONG.FEM.MSUB.SHELL.THERMAL]INPUTFILE.COM
!
! SET LOGICAL ASSIGNMENTS FOR YOUR STARTUP FILE
!
$ STARTUP := $1$DUA0:[ABAQUS.STP]STARTUP.STP
!
! SET LOGICAL ASSIGNMENTS FOR YOUR BATCH QUEUES
! DELETE QUEUES NOT USED
! SUBMIT DEFAULTS TO SYS$BATCH
!
$ SLOW    := SLOW$BATCH
$ FAST    := FAST$BATCH
$ BIG     := BIG$BATCH
$ SYS     := SYS$BATCH
$ MULTI   := MULTI$BATCH
$ PR      := PR$BATCH
$ JOIN    := JOINER$BATCH
!
!  PARAMETERS 'P1'  JOB IDENTIFIER,MUST BE UNIQUE FROM OTHER
```

```
!                    JOBS THAT MAY EXECUTE SIMULTANEUOSLY
!!!              'P2'  ABAQUS DATA INPUT FILE
!!!              'P3'  FORTRAN SOURCE CODE OF USER SUBROUTINES
!!!              'P4'  RESTART READ FILE
!!!              'P5'  OLD FILE IF THIS IS A RESTART READ RUN
!!!              'P6'  SET TO NON-BLANK TO FINISH AFTER DATA CHECK
!
$ LPB2:
$ INQUIRE P10 -
  " IDENTIFIER                                                    "
$ IF P10 .NES. "" THEN  GOTO LPB1
$ WRITE SYS$OUTPUT " JOB IDENTIFIER CANNOT BE BLANK "
$ GOTO LPB2
$ LPB1:
$ C1 = F$SEARCH("''P10'.COM")
$ IF C1 .EQS. "" THEN GOTO LC1
$ DEL/LOG/NOCONFIRM 'P10'.COM;*
$ LC1:
$ INQUIRE TOTALNUM -
  " INPUT NUMBER OF RUNS NEEDED FOR PROBLEM                       "
$ INQUIRE SEQUENCENUM -
  " INPUT NUMBER OF LAST RUN                                      "
$ INQUIRE ENDNUM -
  " INPUT NUMBER WHERE YOU WANT TO STOP                           "
$ INQUIRE STARTTHERMAL -
  " DO YOU WANT TO START FROM THERMAL ANALYSIS (Y/N)              "
$ OPEN/WRITE COM_F 'P10'.COM
$ ADIR=F$ENVIROMENT("DEFAULT")
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ SET VERIFY"
$ WRITE COM_F "$ P9 = """" "
$ WRITE COM_F "$ IF F$GETDVI(""_XQAO:"",""EXISTS"") THEN P9 = ""XQAO:"" "
$ WRITE COM_F "$ IF F$GETDVI(""_XEAO:"",""EXISTS"") THEN P9 = ""XEAO:"" "
$ WRITE COM_F "$ IF F$GETDVI(""_ESAO:"",""EXISTS"") THEN P9 = ""ESAO:"" "
$ WRITE COM_F "$ IF F$GETDVI(""_XNAO:"",""EXISTS"") THEN P9 = ""XNAO:"" "
$ WRITE COM_F "$ IF P9 .EQS. """" THEN GOTO LPBA"
$ WRITE COM_F "$ ASSIGN 'P9' ETAO:"
$ WRITE COM_F "$ LPBA:"
$ WRITE COM_F "$ SET DEF ''ADIR'"
$ WRITE COM_F "$ SET VERIFY"
```

```
!!*********************************************************************
!!** Added by H.SONG for shell program.
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$SHELLLOOP:
$ PO:='P10''SEQUENCENUM'
$ SEQUENCENUM=SEQUENCENUM+1
$ P2:='P10''SEQUENCENUM'
$ WRITE COM_F "$ SET DEF [SONG.FEM.MSUB.SHELL.STRESS]"
$ WRITE COM_F "$ @''INPUTS' ''P10' ''SEQUENCENUM' ''TOTALNUM'"
$ P1:='P2'
$ POT:='PO'T
$ P1T:='P1'T
$ P2T:='P2'T
$ POS:='PO'S
$ P1S:='P1'S
$ P2S:='P2'S
$ IF STARTTHERMAL .EQS. "N" THEN GOTO STRESSSTART
!!*********************************************************************
!!    THERMAL ANALYSIS
!!*********************************************************************
$ WRITE COM_F "$ SET DEF [SONG.FEM.MSUB.SHELL.THERMAL]"
$ IF SEQUENCENUM.EQS.1 THEN GOTO FIRSTR
$ WRITE COM_F "$ COPY [-.STRESS]''P10'''SEQUENCENUM'.016 *.*"
$FIRSTR:
$ WRITE COM_F "$ COPY [-.STRESS]''P10'''SEQUENCENUM'.081 *.018"
$ WRITE COM_F -
 "$ APPEND [-.STRESS]''P10'''SEQUENCENUM'.018  ''P10'''SEQUENCENUM'.018"
$ WRITE COM_F "$ @''INPUTT' ''P10' ''SEQUENCENUM' ''TOTALNUM'"
$ FSCR:='SCRATCH''P1'
$ WRITE COM_F "$ F1 = F$SEARCH(""''P2'.INP"")"
$ LPB3A:=LPB3T
$ LPB3B:='LPB3A''SEQUENCENUM'
$ WRITE COM_F "$ IF F1 .NES. """" THEN GOTO ''LPE3B'
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" CANNOT FIND ''P2'.INP. CHECK THAT IT EXISTS""""
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LPB3B': "
$ WRITE COM_F "$ ASS ''P2'.INP   FOR005"
$ WRITE COM_F "$ ASS ''P1'.023   FOR023"
$ WRITE COM_F "$ ASS ''P1'.028   FOR028"
```

```
$ WRITE COM_F "$ ASS ''P1'.006   FOR006"
$ WRITE COM_F "$ ASS ''P1'.FIL   FOR008"
$ WRITE COM_F "$ ASS ''P1'.PPL   FOR013"
$ WRITE COM_F "$ ASS ''P1'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1'.018   FOR018"
$ WRITE COM_F "$ ASS ''P1'.SUP   FOR020"
!**** ASSIGN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''STARTUP'    FOR042"
!*** ASSIGN RESTART WRITE FILE ***
$ WRITE COM_F "$ ASS ''P1'.RES   FOR012"
!*** ASSIGN USER INPUT FILES ***
$ WRITE COM_F "$ ASS ''P1'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1'.018   FOR018"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''PRE'"
$ LAB7AA:=LAB7AT
$ LAB7AB:='LAB7AA''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB7AB'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING ABAQUS PRE PROGRAM ***"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1'.006  ''P1'.DAT"
$ WRITE COM_F "$ DELETE ''P1'.006;*"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ GOTO GRANDEXIT"
```

```
$ WRITE COM_F "$ ''LAB7AB':"
$ WRITE COM_F "$ SET ON "
$ LAB7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1'.006  ''P1'.DAT"
$ WRITE COM_F "$ DELETE ''P1'.006;*"
$ WRITE COM_F "$ SHOW PROC/ACC"
$ WRITE COM_F "$ SET ON "
!$ IF P6 .EQS. "Y" THEN GOTO LEND
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ FOR ''P1'.028"
$ WRITE COM_F "$ SET ON "
$ LS7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ LAB9:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
! *** ASSIGN MAIN PLOT FILE ***
$ WRITE COM_F "$ ASS ''P1'.MPL FOR013"
! *** ASSIGN MAIN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.001   FOR001"
$ WRITE COM_F "$ ASS ''FSCR'.002   FOR002"
$ WRITE COM_F "$ ASS ''FSCR'.003   FOR003"
$ WRITE COM_F "$ ASS ''FSCR'.004   FOR004"
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.019   FOR019"
$ WRITE COM_F "$ ASS ''P1'.SUP     FOR020"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
```

```
$ WRITE COM_F "$ ASS ''P1'.STA     FOR041"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ LINK/NOMAP/EXE=''P1' ''P1',''MAIN'/LIB,-"
$ WRITE COM_F "''SHR'/LIB"
$ LAB12A:=LAB12T
$ LAB12B:='LAB12A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB12B'"
$ WRITE COM_F "$ IF $SEVERITY .EQS. 0  THEN GOTO ''LAB12B'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN LINKING MAIN ABAQUS LIBRARY *** "" "
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ ''LAB12B':"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''P1'"
$ LAB14A:=LAB14T
$ LAB14B:='LAB14A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB14B'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING MAIN ABAQUS PROGRAM ***"
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB14B':"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ APP ''P1'.006 ''P1'.DAT"
$ WRITE COM_F "$ DELETE ''P1'.006;*"
$ WRITE COM_F "$ SET ON "
$ LEND:
!$ WRITE COM_F "$ LEXIT:"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ F1 = F$SEARCH(""''P1'.EXE"") "
$ WRITE COM_F "$ IF F1 .NES. """" THEN DELETE ''P1'.EXE;*"
$ WRITE COM_F "$ F2 = F$SEARCH(""''P1'.OBJ"") "
$ WRITE COM_F "$ IF F2 .NES. """" THEN DELETE ''P1'.OBJ;*"
$ WRITE COM_F "$ F4 = F$SEARCH(""''P1'.LIS"")"
$ WRITE COM_F "$ IF F4 .NES. """" THEN DELETE ''P1'.LIS;*"
$ WRITE COM_F "$ F5 = F$SEARCH(""''P1'.023"")"
$ WRITE COM_F "$ IF F5 .NES. """" THEN DELETE ''P1'.023;*"
```

```
$ WRITE COM_F "$ F6 = F$SEARCH(""''P1'.028"")"
$ WRITE COM_F "$ IF  F6 .NES. """" THEN DELETE ''P1'.028;*"
!$ WRITE COM_F "$ LAB11:"
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$STRESSSTART:
$ STARTTHERMAL:=Y
$ WRITE COM_F "$ SET DEF [SONG.FEM.MSUB.SHELL.STRESS]"
$ WRITE COM_F "$ COPY [-.THERMAL]''P1'.FIL ''P1'.015"
$ WRITE COM_F "$ @''INPUTS1' ''P10' ''SEQUENCENUM' ''TOTALNUM'"
!!*************************************************************
!!    STRESS/STRAIN ANALYSIS
!!*************************************************************
$ FSCR:='SCRATCH''P1'
$ WRITE COM_F "$ F1 = F$SEARCH(""''P2'.INP"")"
$ LPB3A:=LPB3S
$ LPB3B:='LPB3A''SEQUENCENUM'
$ WRITE COM_F "$ IF F1 .NES. """" THEN GOTO ''LPB3B'
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" CANNOT FIND ''P2'.INP. CHECK THAT IT EXISTS""""
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LPB3B': "
$ WRITE COM_F "$ ASS ''P2'.INP   FOR005"
$ WRITE COM_F "$ ASS ''P1'.023   FOR023"
$ WRITE COM_F "$ ASS ''P1'.028   FOR028"
$ WRITE COM_F "$ ASS ''P1'.006   FOR006"
$ WRITE COM_F "$ ASS ''P1'.FIL   FOR008"
$ WRITE COM_F "$ ASS ''P1'.PPL   FOR013"
$ WRITE COM_F "$ ASS ''P1'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1'.018   FOR018"
$ WRITE COM_F "$ ASS ''P1'.SUP   FOR020"
$ WRITE COM_F "$ ASS ''P1'.090   FOR090"
$ WRITE COM_F "$ ASS ''P1'.091   FOR091"
$ WRITE COM_F "$ ASS ''P0'.082   FOR092"
$ WRITE COM_F "$ ASS ''P0'.017   FOR093"
!**** ASSIGN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
```

```
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''STARTUP'    FOR042"
!*** ASSIGN RESTART WRITE FILE ***
$ WRITE COM_F "$ ASS ''P1'.RES   FOR012"
!*** ASSIGN RESTART READ FILE ***
$ WRITE COM_F "$ ASS ''P0'.RES   FOR011"
!*** ASSIGN USER INPUT FILES ***
$ WRITE COM_F "$ ASS ''P1'.015   FOR015"
$ WRITE COM_F "$ ASS ''P1'.016   FOR016"
$ WRITE COM_F "$ ASS ''P1'.017   FOR017"
$ WRITE COM_F "$ ASS ''P1'.018   FOR018"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''PRE'"
$ LAB7AA:=LAB7AS
$ LAB7AB:='LAB7AA''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB7AB'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING ABAQUS PRE PROGRAM ***"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1'.006  ''P1'.DAT"
$ WRITE COM_F "$ DELETE ''P1'.006;*"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB7AB':"
$ WRITE COM_F "$ SET ON "
$ LAB7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ COPY ''P1'.006  ''P1'.DAT"
$ WRITE COM_F "$ DELETE ''P1'.006;*"
$ WRITE COM_F "$ SHOW PROC/ACC"
$ WRITE COM_F "$ SET ON "
!$ IF P6 .EQS. "Y" THEN GOTO LEND
```

```
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$! WRITE COM_F "$ @MOD28 ''P1'"
$ WRITE COM_F "$ FOR ''P1'.028"
$ WRITE COM_F "$ SET ON "
$ LS7:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ LAB9:
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
! *** ASSIGN MAIN PLOT FILE ***
$ WRITE COM_F "$ ASS ''P1'.MPL FOR013"
! *** ASSIGN MAIN SCRATCH FILES ***
$ WRITE COM_F "$ ASS ''FSCR'.001   FOR001"
$ WRITE COM_F "$ ASS ''FSCR'.002   FOR002"
$ WRITE COM_F "$ ASS ''FSCR'.003   FOR003"
$ WRITE COM_F "$ ASS ''FSCR'.004   FOR004"
$ WRITE COM_F "$ ASS ''FSCR'.010   FOR010"
$ WRITE COM_F "$ ASS ''FSCR'.019   FOR019"
$ WRITE COM_F "$ ASS ''P1'.SUP     FOR020"
$ WRITE COM_F "$ ASS ''FSCR'.021   FOR021"
$ WRITE COM_F "$ ASS ''FSCR'.022   FOR022"
$ WRITE COM_F "$ ASS ''FSCR'.024   FOR024"
$ WRITE COM_F "$ ASS ''FSCR'.025   FOR025"
$ WRITE COM_F "$ ASS ''FSCR'.026   FOR026"
$ WRITE COM_F "$ ASS ''FSCR'.027   FOR027"
$ WRITE COM_F "$ ASS ''FSCR'.029   FOR029"
$ WRITE COM_F "$ ASS ''FSCR'.030   FOR030"
$ WRITE COM_F "$ ASS ''P1'.STA     FOR041"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ LINK/NOMAP/EXE=''P1' ''P1',''MAIN'/LIB,-"
$ WRITE COM_F "''SHR'/LIB"
$ LAB12A:=LAB12S
$ LAB12B:='LAB12A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB12B'"
$ WRITE COM_F "$ IF $SEVERITY .EQS. 0  THEN GOTO ''LAB12B'"
```

```
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN LINKING MAIN ABAQUS LIBRARY *** "" "
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ ''LAB12B':"
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ RUN ''P1'"
$ LAB14A:=LAB14S
$ LAB14B:='LAB14A''SEQUENCENUM'
$ WRITE COM_F "$ IF $STATUS THEN GOTO ''LAB14B'"
$ WRITE COM_F -
 "$ WRITE SYS$OUTPUT "" *** ERROR IN EXECUTING MAIN ABAQUS PROGRAM ***"
$ WRITE COM_F "$ GOTO GRANDEXIT"
$ WRITE COM_F "$ ''LAB14B':"
$ WRITE COM_F "$ SET ON "
$ WRITE COM_F "$ SET NOON "
$ WRITE COM_F "$ ON ERROR THEN GOTO GRANDEXIT "
$ WRITE COM_F "$ APP ''P1'.006 ''P1'.DAT"
$ WRITE COM_F "$ DELETE ''P1'.006;*"
$ WRITE COM_F "$ SET ON "
$ LEND:
!$ WRITE COM_F "$ LEXIT:"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ F1 = F$SEARCH(""''P1'.EXE"") "
$ WRITE COM_F "$ IF F1 .NES. """" THEN DELETE ''P1'.EXE;*"
$ WRITE COM_F "$ F2 = F$SEARCH(""''P1'.OBJ"") "
$ WRITE COM_F "$ IF F2 .NES. """" THEN DELETE ''P1'.OBJ;*"
$ WRITE COM_F "$ F4 = F$SEARCH(""''P1'.LIS"")"
$ WRITE COM_F "$ IF F4 .NES. """" THEN DELETE ''P1'.LIS;*"
$ WRITE COM_F "$ F5 = F$SEARCH(""''P1'.023"")"
$ WRITE COM_F "$ IF F5 .NES. """" THEN DELETE ''P1'.023;*"
$ WRITE COM_F "$ F6 = F$SEARCH(""''P1'.028"")"
$ WRITE COM_F "$ IF  F6 .NES. """" THEN DELETE ''P1'.028;*"
!$ WRITE COM_F "$ LAB11:"
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ IF SEQUENCENUM .LT. ENDNUM THEN GOTO SHELLLOOP
$ WRITE COM_F "$GRANDEXIT:"
$ WRITE COM_F "$ SET ON"
$ WRITE COM_F "$ F1 = F$SEARCH(""''P10'*.EXE"") "
$ WRITE COM_F "$ IF F1 .NES. """" THEN DELETE ''P10'*.EXE;*"
```

```
$ WRITE COM_F "$ F2 = F$SEARCH(""''P10'*.OBJ"") "
$ WRITE COM_F "$ IF F2 .NES. """" THEN DELETE ''P10'*.OBJ;*"
$ WRITE COM_F "$ F4 = F$SEARCH(""''P10'*.LIS"")"
$ WRITE COM_F "$ IF F4 .NES. """" THEN DELETE ''P10'*.LIS;*"
$ WRITE COM_F "$ F5 = F$SEARCH(""''P10'*.023"")"
$ WRITE COM_F "$ IF F5 .NES. """" THEN DELETE ''P10'*.023;*"
$ WRITE COM_F "$ F6 = F$SEARCH(""''P10'*.028"")"
$ WRITE COM_F "$ IF  F6 .NES. """" THEN DELETE ''P10'*.028;*"
$ WRITE COM_F "$ EXIT:"
$ CLOSE COM_F
!!!!!!!!!!!!!!!!!!!!!!!!
!! Added by H.S. to organize queues automatically
!!!!!!!!!!!!!!!!!!!!!!!!
$ REPEAT:
$ inquire QUE
$ IF QUE .EQS. "" THEN GOTO REPEAT
$!
$ WRITE SYS$OUTPUT " GIVE THE AMOUNT OF TIME YOU WANT TO HOLD THE JOB FOR"
$ INQUIRE HOLD -
  " THE FORMAT IS HH:MM: ( <RET> TO IGNORE)                    "
$ IF QUE .EQS. "M"
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD10
$ SUBMIT/NOTIFY/QUEUE=MULTI$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD10:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=MULTI$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$!!!!!!!! This queue can only be used by Haoshi
$ IF QUE .EQS. "P"
$ THEN
$ USNM=F$GETJPI("","USERNAME")
$ IF USNM .NES. "SONG        " THEN GOTO REPEAT
$ IF HOLD .NES. "" THEN GOTO HOLD11
$ SUBMIT/NOTIFY/QUEUE=PR$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD11:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=PR$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
```

```
$ ENDIF
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ IF QUE .EQS. 4
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD4
$ SUBMIT/NOTIFY/QUEUE=JOINER$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD4:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=JOINER$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$ IF QUE .EQS. 3
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD2
$ SUBMIT/NOTIFY/QUEUE=BIG$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD2:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=BIG$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$ IF QUE .EQS. 2
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD3
$ SUBMIT/NOTIFY/QUEUE=SLOW$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD3:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=SLOW$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$ IF QUE .EQS. 1
$ THEN
$ IF HOLD .NES. "" THEN GOTO HOLD4
$ SUBMIT/NOTIFY/QUEUE=SYS$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD4:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=SYS$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$ IF QUE .EQS. 0
$ THEN
```

```
$ IF HOLD .NES. "" THEN GOTO HOLD5
$ SUBMIT/NOTIFY/QUEUE=FAST$BATCH/NOPRINT/LOG ='ADIR''P10'.LOG 'P10'
$ GOTO LEXIT
$ HOLD5:
$ SUBMIT/NOTIFY/AFTER="''HOLD'"/QUEUE=FAST$BATCH/NOPRINT-
  /LOG='ADIR''P10'.LOG 'P10'
$ ENDIF
$!!!!!!!!!!!!
$ LAB11:
$!!!!!!!!!!
$ LEXIT:
$ EXIT:
```

B.2.2 The Command Files to Compose an ABAQUS Input Deck

The following two command files are used to create ABAQUS input decks for heat transfer and deformation analyses.

Heat Transfer Analyses

```
$!
$ FILENAME := 'P1'
$ SNUMBER  := 'P2'
$ TOTALNUM := 'P3'
$ ADIR=F$ENVIRONMENT("DEFAULT")
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!Create a file containing curent increment information
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ OPEN/WRITE TRANSDATA INPUT.001
$ WRITE TRANSDATA "''FILENAME'"
$ WRITE TRANSDATA SNUMBER
$ WRITE TRANSDATA TOTALNUM
$ CLOSE TRANSDATA
$ HEADING:='ADIR'HEADING.COM
$ @'HEADING'
$ @'ADIR'NODE.COM
$ ELEMENT:='ADIR'ELEMENT.COM
$ @'ELEMENT'
```

```
$ SOLID:='ADIR'SOLID.COM
$ @'SOLID'
$ @'ADIR'MPC.COM
$ @'ADIR'INITIAL.COM
$ @'ADIR'PLOT.COM
$ @'ADIR'STEP.COM
$ @'ADIR'USER.COM
$!
$!
$!  COMPILING INPUT DECKS.
$!
$!
$ APPEND NODE.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM NODE.DAT;*
$ APPEND ELEMENT.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM ELEMENT.DAT;*
$ APPEND MATERIAL.DAT 'FILENAME''SNUMBER'.INP
$ APPEND SOLID.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM SOLID.DAT;*
$ APPEND MPC.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM MPC.DAT;*
$ APPEND INITIAL.DAT 'FILENAME''SNUMBER'.INP
$ DEL INITIAL.DAT;*/NOCONFIRM
$ APPEND PLOT.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM PLOT.DAT;*
$ APPEND STEP.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM STEP.DAT;*
$ APPEND USER.DAT 'FILENAME''SNUMBER'.INP
$ DEL/NOCONFIRM USER.DAT;*
$ EXIT:
```

Deformation Analyses

```
$ FILENAME := 'P1'
$ SNUMBER  := 'P2'
$ TOTALNUM := 'P3'
$ ADIR=F$ENVIRONMENT("DEFAULT")
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!Create a file containing curent increment information
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
$ OPEN/WRITE TRANSDATA INPUT.001
$ WRITE TRANSDATA "''FILENAME'"
$ WRITE TRANSDATA SNUMBER
$ WRITE TRANSDATA TOTALNUM
$ CLOSE TRANSDATA
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$! Heading for the input deck
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ HEADING:='ADIR'HEADING.COM
$ @'HEADING'
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$!Creating two files for both the local zone and the substructures
$!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
$ OPEN/WRITE LOCAL LOCAL.INP
$ WRITE LOCAL "**The following contains local zone information"
$ CLOSE LOCAL
$ OPEN/WRITE SUB SUBSTRUCTURE.INP
$ WRITE SUB "**The following contains substructure information"
$ WRITE SUB "*SUPER, ID=Z001"
$ CLOSE SUB
$!!!!!!!!!!!!!!!!!!
$! Node coordinates
$!!!!!!!!!!!!!!!!!!
$ NODE:='ADIR'NODE.COM
$ @'NODE'
$!!!!!!!!!!!!!!!!!!!!!
$! Eelement definition
$!!!!!!!!!!!!!!!!!!!!!
$ ELEMENT:='ADIR'ELEMENT.COM
$ @'ELEMENT'
$!!!!!!!!!!!!!!!!!!!!!!!!!
$! Solid section definition
$!!!!!!!!!!!!!!!!!!!!!!!!!
$ SOLID:='ADIR'SOLID.COM
$ @'SOLID'
$!!!!!!!!!!!!!!!!!!!!!
$! MPC definition
$!!!!!!!!!!!!!!!!!!!!!
$ @'ADIR'MPC.COM
$!!!!!!!!!!!!!!!!!!!!
```

```
$! Initial conditions
$!!!!!!!!!!!!!!!!!!!!
$ @'ADIR'INITIAL.COM
$ @'ADIR'INTLOAD.COM
$! @'ADIR'RETDOF.COM
$ @'ADIR'BOUNDARY.COM
$ @'ADIR'PLOT.COM
$!
$!
$ APPEND 'ADIR'NODE.DAT 'ADIR'LOCAL.INP
$ DEL/NOCONFIRM 'ADIR'NODE.DAT;*
$ APPEND 'ADIR'NODE1.DAT 'ADIR'SUBSTRUCTURE.INP
$ APPEND 'ADIR''FILENAME''SNUMBER'.081 'ADIR'SUBSTRUCTURE.INP
$ DEL/NOCONFIRM 'ADIR'NODE1.DAT;*
$! DEL/noconfirm 'ADIR''FILENAME''SNUMBER'.081;*
$!
$ APPEND ELEMENT.DAT LOCAL.INP
$ APPEND ELEMENT1.DAT SUBSTRUCTURE.INP
$ APPEND 'ADIR''FILENAME''SNUMBER'.082 SUBSTRUCTURE.INP
$ DEL/NOCONFIRM 'ADIR'ELEMENT.DAT;*
$ DEL/NOCONFIRM 'ADIR'ELEMENT1.DAT;*
$!
$ APPEND MATERIAL.DAT LOCAL.INP
$ APPEND MATERIAL.DAT SUBSTRUCTURE.INP
$!
$ APPEND SOLID.DAT LOCAL.INP
$ DEL/NOCONFIRM SOLID.DAT;*
$ APPEND SOLID1.DAT SUBSTRUCTURE.INP
$ DEL/NOCONFIRM SOLID1.DAT;*
$!
$ APPEND MPC.DAT LOCAL.INP
$ DEL/NOCONFIRM MPC.DAT;*
$ APPEND MPC1.DAT SUBSTRUCTURE.INP
$ DEL/NOCONFIRM MPC1.DAT;*
$!
$ APPEND INITIAL.DAT LOCAL.INP
$ DEL INITIAL.DAT;*/NOCONFIRM
$!
$ APPEND BOUNDARY1.DAT SUBSTRUCTURE.INP
$ DEL BOUNDARY1.DAT;*/NOCONFIRM
```

```
$!
$! APPEND RETDOF.DAT LOCAL.INP
$! DEL/NOCONFIRM RETDOF.DAT;*
$ APPEND RETDOF1.DAT SUBSTRUCTURE.INP
$ DEL/NOCONFIRM RETDOF1.DAT;*
$!
$ APPEND PLOT.DAT LOCAL.INP
$ DEL/NOCONFIRM PLOT.DAT;*
$ APPEND PLOT1.DAT SUBSTRUCTURE.INP
$ DEL/NOCONFIRM PLOT1.DAT;*
$!
$ APPEND AMP.DAT LOCAL.INP
$ DEL/NOCONFIRM AMP.DAT;*
$!
$ EXIT:
```

B.2.3 The Time Period for an Incremental Analysis

```
      PROGRAM STEP
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER NULL*2
*
      SPEED=2.0
*
      OPEN (UNIT=20,STATUS='NEW')
1     FORMAT(1X,I5,4(',',1X,I5))
2     FORMAT(2A)
C
C Read information about the incremental analysis.
C
      OPEN (UNIT=19,FILE='INPUT.001',STATUS='OLD')
      READ(19,2) NULL
      READ(19,*) ISNUMBER
      READ(19,*) ITOTALNUM
      CLOSE(UNIT=19)
C
      LENGTH=4
      LMINUS1=3
      IF (ISNUMBER.EQ.1) THEN
        TIME=(480*REAL(LENGTH)/REAL(ITOTALNUM+LMINUS1)-16.0*2.5+2.5)
```

```
        TIME=TIME/SPEED
        TMAX=0.5
      ELSEIF (ISNUMBER.EQ.ITOTALNUM) THEN
        TIME=(480*REAL(LENGTH)/REAL(ITOTALNUM+LMINUS1)-16.0*2.5+2.5)
        TIME=TIME+480.0*REAL(ITOTALNUM-2)/REAL(ITOTALNUM+LMINUS1)
        TIME=TIME/SPEED
        TIME=485.0/SPEED-TIME
        TMAX=0.5
      ELSE
        IF (ISNUMBER.EQ.ITOTALNUM+1) THEN
          TIME=10000.0
          TMAX=1000.0
        ELSE
          TIME=480/REAL(ITOTALNUM+LMINUS1)/SPEED
          TMAX=0.5
        ENDIF
      ENDIF
      TINIT=0.1
      TMIN=0.001
      WRITE(20,100) TINIT,TIME,TMIN,TMAX
100   FORMAT(F7.3,',',F10.3,2(',',F8.3))
      CLOSE(UNIT=20)
      STOP
      END
```

B.2.4 The Program to Read Stresses and Extrapolate to Nodal Points

This program needs to be linked with the ABAQUS library file **A3Q7SHR.OLB** before it can be executed.

```
      PROGRAM MESH
        IMPLICIT REAL*8(A-H,O-Z)
        DIMENSION ARRAY(513),JRRAY(2,513),LRUNIT(2,1),TEMP(2,6000)
        DIMENSION COORDS(6000,3),CDS(6000,3),LEL(6000,4),
     1            ISTRESS(6000),STRESS(6000,3),ST(6000,3,4),
     2            DISPL(6000,3),KN(6000),STRS1(6000,3),CONV(4,4),
     3            KNS(6000),STS(6000,3,4)
        COMMON/COORD/COORDS
```

```
      COMMON/EL/LEL
      EQUIVALENCE (ARRAY(1),JRRAY(1,1))
C*******************************
C     INITIALIZE
C*******************************
     DO 7002 I=1,6000
      KN(I)=0
      KNS(I)=0
      ISTRESS(I)=0
      TEMP(1,I)=0.0D0
      TEMP(2,I)=0.0D0
      LEL(I,4)=0
      DO 7001 J=1,3
      STRS1(I,J)=0.0D0
      DISPL(I,J)=0.0D0
     COORDS(I,J)=0.0D0
      CDS(I,J)=0.0D0
     STRESS(I,J)=0.0D0
      LEL(I,J)=0
      DO 6999 K=1,4
      ST(I,J,K)=0.0D0
      STS(I,J,K)=0.0D0
6999  CONTINUE
7001  CONTINUE
7002  CONTINUE
C
C     Obtain increment information
C
      OPEN (UNIT=19,FILE='INPUT.001',STATUS='OLD')
      READ(19,2) NULL
      READ(19,*) ISNUMBER
      READ(19,*) ITOTALNUM
      CLOSE(UNIT=19)
C
C     Obtain previous element definitions
C
      OPEN(UNIT=82,TYPE='OLD')
      DO 2005 I=1,3000
      READ(82,2003,END=2004) IE,NODE1,NODE2,NODE3,NODE4
      LEL(IE,1)=NODE1
```

```
        LEL(IE,2)=NODE2
        LEL(IE,3)=NODE3
        LEL(IE,4)=NODE4
2005    CONTINUE
2004    CONTINUE
        OPEN(UNIT=17,TYPE='OLD')
        DO 2000 I=1,5000
        READ(17,2003,END=2002) IE,NODE1,NODE2,NODE3,NODE4
C        IE=IE+1000
        LEL(IE,1)=NODE1
        LEL(IE,2)=NODE2
        LEL(IE,3)=NODE3
        LEL(IE,4)=NODE4
2000    CONTINUE
2002    CONTINUE
2003    FORMAT(1X,I5,4(',',1X,I5))
C
C*      Read results from previous incremental analysis.
C
        NRU=2
     LRUNIT(1,1)=8
     LRUNIT(1,2)=7
     LRUNIT(2,1)=2
     LRUNIT(2,2)=2
     LOUTF=2
     CALL INITPF(NRU,LRUNIT,LOUTF)
        JUNIT=8
     CALL DBRNU(JUNIT)
        DISP=0.0
        JRCD=0
        ISR=0
7003    CONTINUE
        DO 100 K1=1,999999
        CALL DBFILE(0,ARRAY,JRCD)
        IF(JRCD.NE.0) GO TO 900
150     KEY=JRRAY(1,2)
        IF (KEY.EQ.107) THEN
              NODE=JRRAY(1,3)
              COORDS(NODE,1)=ARRAY(4)
              COORDS(NODE,2)=ARRAY(5)
```

```
            COORDS(NODE,3)=ARRAY(6)
         ENDIF
         IF (KEY.EQ.101) THEN
               NODE=JRRAY(1,3)
               DISPL(NODE,1)=ARRAY(4)
               DISPL(NODE,2)=ARRAY(5)
               DISPL(NODE,3)=ARRAY(6)
         ENDIF
         IF (KEY.EQ.1) THEN
               IE=JRRAY(1,3)
               INTP=JRRAY(1,4)
         CALL DBFILE(0,ARRAY,JRCD)
               ST(IE,1,INTP)=ARRAY(3)
               ST(IE,2,INTP)=ARRAY(4)
               ST(IE,3,INTP)=ARRAY(5)
         ENDIF
         IF (KEY.EQ.1910) THEN
              KEY1=JRRAY(1,3)
              IF (KEY1.EQ.0) THEN
                     GO TO 1000
              END IF
         ENDIF
100   CONTINUE
900      CONTINUE
         GO TO 1010
1000  CONTINUE
         DO 110 K2=1,999999
         CALL DBFILE(0,ARRAY,JRCD)
         IF(JRCD.NE.0) GO TO 1010
         KEY=JRRAY(1,2)
         IF (KEY.EQ.1910) THEN
              KEY1=JRRAY(1,3)
              IF (KEY1.EQ.1) THEN
                     GO TO 7003
              END IF
         ENDIF
         IF (KEY.EQ.107) THEN
               NODE=JRRAY(1,3)
                  COORDS(NODE,1)=ARRAY(4)
                  COORDS(NODE,2)=ARRAY(5)
```

```
                COORDS(NODE,3)=ARRAY(6)
        ENDIF
        IF (KEY.EQ.101) THEN
              NODE=JRRAY(1,3)
                 DISPL(NODE,1)=ARRAY(4)
                 DISPL(NODE,2)=ARRAY(5)
                 DISPL(NODE,3)=ARRAY(6)
        ENDIF
        IF (KEY.EQ.1) THEN
              IE=JRRAY(1,3)
              INTP=JRRAY(1,4)
        CALL DBFILE(0,ARRAY,JRCD)
              STS(IE,1,INTP)=ARRAY(3)
              STS(IE,2,INTP)=ARRAY(4)
              STS(IE,3,INTP)=ARRAY(5)
        ENDIF
110  CONTINUE
1010    CONTINUE
C
C       Reading temperature infor.
C
        JUNIT=7
     CALL DBRNU(JUNIT)
        DISP=0.0
        JRCD=0
        ISR=0
        DO 160 K1=1,999999
        CALL DBFILE(0,ARRAY,JRCD)
        IF(JRCD.NE.0) GO TO 1060
        KEY=JRRAY(1,2)
        IF (KEY.EQ.201) THEN
              NODE=JRRAY(1,3)
              TEMP(1,NODE)=ARRAY(4)
              TEMP(2,NODE)=ARRAY(4)
      ENDIF
160  CONTINUE
1060    CONTINUE
C*************************************************************
C
C       Convert integration point variables to nodal variables.
```

```
C
C*************************************************************
        CONST=SQRT(3.0)/2.0
        CONV(1,1)=1+CONST
        CONV(1,2)=-0.5
        CONV(1,3)=1-CONST
        CONV(1,4)=-0.5
        CONV(2,1)=-0.5
        CONV(2,2)=1+CONST
        CONV(2,3)=-0.5
        CONV(2,4)=1-CONST
        CONV(3,1)=1-CONST
        CONV(3,2)=-0.5
        CONV(3,3)=1+CONST
        CONV(3,4)=-0.5
        CONV(4,1)=-0.5
        CONV(4,2)=1-CONST
        CONV(4,3)=-0.5
        CONV(4,4)=1+CONST
C
        DO 5000 I=1,6000
        II=I
        DO 5010 J=1,4
        SWIT=ABS(ST(I,1,1))+ABS(ST(I,2,1))
        IF (SWIT.EQ.0.0) GO TO 5010
        DO 5020 K=1,3
        DO 5030 L=1,4
        STRESS(LEL(II,J),K)=STRESS(LEL(II,J),K)+ST(I,K,L)*CONV(J,L)
5030    CONTINUE
5020    CONTINUE
        KN(LEL(II,J))=KN(LEL(II,J))+1
5010    CONTINUE
5000    CONTINUE
        DO 5100 I=1,6000
        DO 5110 J=1,4
        SWIT=ABS(STS(I,1,1))+ABS(STS(I,2,1))
C        IF (SWIT.EQ.0.0) GO TO 5110
        DO 5120 K=1,3
        DO 5130 L=1,4
        STRESS(LEL(I,J),K)=STRESS(LEL(I,J),K)+STS(I,K,L)*CONV(J,L)
```

```
5130    CONTINUE
5120    CONTINUE
        KNS(LEL(I,J))=KNS(LEL(I,J))+1
5110    CONTINUE
5100    CONTINUE
C*****************************************************
C       Get and update stresses inside substructures.
C*****************************************************
C
        IF (ISNUMBER.LE.2) GO TO 5200
        OPEN(UNIT=92,TYPE='OLD')
        DO 3500 I=1,6000
        READ(92,145,END=3505) NODE,STRESS1,STRESS2,STRESS3
        STRS1(NODE,1)=STRESS1
        STRS1(NODE,2)=STRESS2
        STRS1(NODE,3)=STRESS3
3500    CONTINUE
3505    CONTINUE
        CLOSE(UNIT=92)
C*************************************************
        DO 5200 I=1,6000
        IF (KNS(I).EQ.0) GO TO 5200
        DO 5210 J=1,3
        STRESS(I,J)=STRESS(I,J)+STRS1(I,J)*REAL(KNS(I))
5210    CONTINUE
5200    CONTINUE
        DO 5050 I=1,6000
        KN(I)=KN(I)+KNS(I)
        IF (KN(I).EQ.0) GO TO 5050
        DO 5060 J=1,3
        STRESS(I,J)=STRESS(I,J)/REAL(KN(I))
5060    CONTINUE
5050    CONTINUE
2       FORMAT(2A)
      LMINUS1=3
      LAST=11
      MAXLL=(ISNUMBER+LMINUS1)*192/(ITOTALNUM+LMINUS1)
      IF (MAXLL.GT.192) THEN
        MAXLL=192
```

```
      ENDIF
      MINLL=(ISNUMBER-LAST)*192/(ITOTALNUM+LMINUS1)
      IF (MINLL.LT.0) THEN
        MINLL=0
      ENDIF
      A=REAL(MINLL/4)
      B=REAL(MINLL)/4.0
      IF (A.NE.B) THEN
         MINLL=INT(B+0.5)*4
      ENDIF
      A=REAL((192-MAXLL)/4)
      B=REAL(192-MAXLL)/4.0
      IF (A.NE.B) THEN
         MAXLL=192-INT(B+0.5)*4
      ENDIF
CC
        MINLL1=(ISNUMBER+LMINUS1-1)*192/(ITOTALNUM+LMINUS1)
        IF (MINLL1.LT.0) THEN MINLL1=0
        A=REAL((192-MINLL1)/4)
        B=REAL((192-MINLL1))/4.0
        IF (A.NE.B) THEN
          MINLL1=192-INT(B+0.5)*4
        ENDIF
          MAXLL1=MAXLL+1
       LOW=(MINLL1-9)*25
       LHIGH=MAXLL1*25
C*****************************
C
C
        OPEN(UNIT=16,TYPE='NEW')
        OPEN(UNIT=18,TYPE='NEW')
        OPEN(UNIT=81,TYPE='NEW')
        OPEN(UNIT=90,TYPE='NEW')
        OPEN(UNIT=91,TYPE='NEW')
C
        DO 1005 I=1,6000
        DO 1005 K=1,3
        COORDS(I,K)=COORDS(I,K)+DISPL(I,K)
        CDS(I,K)=COORDS(I,K)
1005    CONTINUE
```

```
C
C       Old mesh.
C
C
C Interpolating Variables.
C
       DO 500 K=LOW,LHIGH
       IF (TEMP(1,K).NE.0.0D0) GO TO 500
       LAYER1=(K-1)/25
       NUMBER=K-LAYER1*25
       IF (NUMBER.LT.5) GO TO 500
        CALL COORDINATES(K,TEMP,TEMP(2,K),X,Y,Z)
        CDS(K,1)=X
        CDS(K,2)=Y
        CDS(K,3)=Z
500    CONTINUE
C
C
        IF (MINLL.EQ.0) THEN
            MINLL2=0
        ELSE
            MINLL2=MINLL+4
        ENDIF
        IF (MAXLL.EQ.192) THEN
            MAXLL2=192
        ELSE
            MAXLL2=MAXLL-4
        ENDIF
C
        DO 200 I=1,5000
C        IF (TEMP(2,I).EQ.0.0) GO TO 200
       LAYER1=(I-1)/25
        NUMBER=I-LAYER1*25
     IF (NUMBER.LT.6) GO TO 4250
       ISW=(LAYER1-MINLL)*(LAYER1-MAXLL)
       IF (ISW.GT.0) GO TO 4200
       ISW=(LAYER1-MINLL2)*(LAYER1-MAXLL2)
       IF (ISW.GT.0) GO TO 4260
        WRITE(16,120) I,TEMP(2,I)
        WRITE(18,130) I,CDS(I,1),CDS(I,2),CDS(I,3)
```

```
      IF (NUMBER.EQ.6) GO TO 4260
      IF (ISW.LT.0) GO TO 4300
       GO TO 4260
4200   CONTINUE
      IF (NUMBER.LE.9) GO TO 4260
      NUMBER1=(NUMBER-9)/8
      NUMBER=NUMBER-NUMBER1*8-9
      IF (NUMBER.LT.0) GO TO 4300
      IF (NUMBER.GT.0) GO TO 4300
4250   CONTINUE
     IF (NUMBER.GE.5) GO TO 4260
        ISW=LAYER1/8
        ISW=LAYER1-ISW*8
        IF (ISW.NE.0) GO TO 4300
4260   CONTINUE
        WRITE(16,120) I,TEMP(2,I)
        WRITE(81,130) I,CDS(I,1),CDS(I,2),CDS(I,3)
4300   CONTINUE
200    CONTINUE
C*********
        DO 300 I=1, 5000
        IF (TEMP(1,I).EQ.0.0) GO TO 300
        WRITE(90,140) I,COORDS(I,1),COORDS(I,2),COORDS(I,3)
        WRITE(91,145) I,STRESS(I,1),STRESS(I,2),STRESS(I,3)
300    CONTINUE
        CLOSE(UNIT=16)
        CLOSE(UNIT=18)
        CLOSE(UNIT=81)
        CLOSE(UNIT=90)
        CLOSE(UNIT=91)
120  FORMAT(1X,I5,',',F12.3)
130  FORMAT(1X,I5,3(',',F16.13))
140  FORMAT(1X,I5,3(',',F15.12))
145  FORMAT(1X,I5,3(',',E15.5))
10000   CONTINUE
        STOP
     END


        SUBROUTINE COORDINATES(K,TEMP,TEMP1,X,Y,Z)
```

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COORDS(6000,3),LEL(6000,4),TEMP(2,6000),LAYER(4)
      DIMENSION NUMBER(4),CDS(6000,3)
      COMMON/COORD/COORDS
      COMMON/EL/LEL
C
      LAYERO=(K-1)/25
      NUMBERO=K-LAYERO*25
      DO 100 I=1,6000
      DO 200 J=1,4
      LAYER(J)=(LEL(I,J)-1)/25
      NUMBER(J)=LEL(I,J)-LAYER(J)*25
200   CONTINUE
      IF (LAYERO.EQ.LAYER(1)) THEN
         IF ((NUMBERO-NUMBER(1))*(NUMBERO-NUMBER(2)).LT.0) THEN
             JJ=NUMBER(1)-NUMBERO
             CALL LAM(NUMBERO,JJ,LAMBDA)
             JJ=NUMBER(2)-NUMBERO
             CALL LAM(NUMBERO,JJ,LAMBDA1)
          NODE1=LEL(I,1)
          NODE2=LEL(I,2)
          GO TO 10000
         ENDIF
       ENDIF
      IF (LAYERO.EQ.LAYER(3)) THEN
         IF ((NUMBERO-NUMBER(3))*(NUMBERO-NUMBER(4)).LT.0) THEN
             JJ=NUMBER(3)-NUMBERO
             CALL LAM(NUMBERO,JJ,LAMBDA)
             JJ=NUMBER(4)-NUMBERO
             CALL LAM(NUMBERO,JJ,LAMBDA1)
          NODE1=LEL(I,3)
          NODE2=LEL(I,4)
          GO TO 10000
         ENDIF
       ENDIF
      IF (NUMBERO.EQ.NUMBER(1)) THEN
         IF ((LAYERO-LAYER(1))*(LAYERO-LAYER(4)).LT.0) THEN
             LAMBDA=ABS(LAYERO-LAYER(1))
             LAMBDA1=ABS(LAYERO-LAYER(4))
          NODE1=LEL(I,1)
```

```
            NODE2=LEL(I,4)
            GO TO 10000
           ENDIF
         ENDIF
        IF (NUMBER0.EQ.NUMBER(2)) THEN
           IF ((LAYER0-LAYER(2))*(LAYER0-LAYER(3)).LT.0) THEN
               LAMBDA=ABS(LAYER0-LAYER(2))
               LAMBDA1=ABS(LAYER0-LAYER(3))
            NODE1=LEL(I,2)
            NODE2=LEL(I,3)
            GO TO 10000
           ENDIF
         ENDIF
        II=(NUMBER0-NUMBER(1))*(NUMBER0-NUMBER(2))
        JJ=(LAYER0-LAYER(1))*(LAYER0-LAYER(3))
        IF (II.GT.0) GO TO 100
        IF (JJ.GT.0) GO TO 100
C************
C       IN THIS ELEMENT.
C************
               JJ=NUMBER(1)-NUMBER0
               CALL LAM(NUMBER0,JJ,LAMBDA1)
               JJ=NUMBER(2)-NUMBER0
               CALL LAM(NUMBER0,JJ,LAMBDA2)
            LAMBDA5=ABS(LAYER0-LAYER(1))
            LAMBDA6=ABS(LAYER0-LAYER(3))
            NODE1=LEL(I,1)
            NODE2=LEL(I,2)
            NODE3=LEL(I,3)
            NODE4=LEL(I,4)
            TEMP1=TEMP(1,NODE1)*REAL(LAMBDA2*LAMBDA6)+
     1        TEMP(1,NODE2)*REAL(LAMBDA1*LAMBDA6)+
     2        TEMP(1,NODE3)*REAL(LAMBDA1*LAMBDA5)+
     3        TEMP(1,NODE4)*REAL(LAMBDA2*LAMBDA5)
            TEMP1=TEMP1/REAL(LAMBDA1+LAMBDA2)/REAL(LAMBDA5+LAMBDA6)
            X=COORDS(NODE1,1)*REAL(LAMBDA2*LAMBDA6)+
     1        COORDS(NODE2,1)*REAL(LAMBDA1*LAMBDA6)+
     2        COORDS(NODE3,1)*REAL(LAMBDA1*LAMBDA5)+
     3        COORDS(NODE4,1)*REAL(LAMBDA2*LAMBDA5)
            X=X/REAL(LAMBDA1+LAMBDA2)/REAL(LAMBDA5+LAMBDA6)
```

```
            Y=COORDS(NODE1,2)*REAL(LAMBDA2*LAMBDA6)+
     1        COORDS(NODE2,2)*REAL(LAMBDA1*LAMBDA6)+
     2        COORDS(NODE3,2)*REAL(LAMBDA1*LAMBDA5)+
     3        COORDS(NODE4,2)*REAL(LAMBDA2*LAMBDA5)
            Y=Y/REAL(LAMBDA1+LAMBDA2)/REAL(LAMBDA5+LAMBDA6)
            Z=COORDS(NODE1,3)*REAL(LAMBDA2*LAMBDA6)+
     1        COORDS(NODE2,3)*REAL(LAMBDA1*LAMBDA6)+
     2        COORDS(NODE3,3)*REAL(LAMBDA1*LAMBDA5)+
     3        COORDS(NODE4,3)*REAL(LAMBDA2*LAMBDA5)
            Z=Z/REAL(LAMBDA1+LAMBDA2)/REAL(LAMBDA5+LAMBDA6)
         GO TO 99999
100      CONTINUE
         GO TO 99999
10000    CONTINUE
         TEMP1=TEMP(1,NODE1)*REAL(LAMBDA1)+TEMP(1,NODE2)*REAL(LAMBDA)
         TEMP1=TEMP1/REAL(LAMBDA+LAMBDA1)
         X=COORDS(NODE1,1)*REAL(LAMBDA1)+COORDS(NODE2,1)*REAL(LAMBDA)
         X=X/REAL(LAMBDA+LAMBDA1)
         Y=COORDS(NODE1,2)*REAL(LAMBDA1)+COORDS(NODE2,2)*REAL(LAMBDA)
         Y=Y/REAL(LAMBDA+LAMBDA1)
         Z=COORDS(NODE1,3)*REAL(LAMBDA1)+COORDS(NODE2,3)*REAL(LAMBDA)
         Z=Z/REAL(LAMBDA+LAMBDA1)
99999    CONTINUE
         RETURN
         END

         SUBROUTINE LAM(NUMBER,J,LAMBDA)
         IMPLICIT REAL*8(A-H,O-Z)
C
         IF (NUMBER.GE.7) THEN
               L1=NUMBER
            ELSE
            L=NUMBER-6
            IF(L) 1, 2, 3
1           L1=1
               GO TO 100
2              L1=5
               GO TO 100
3              L1=7
100         CONTINUE
```

```
            ENDIF
            IF (NUMBER+J.GE.7) THEN
               L2=NUMBER+J
            ELSE
            L=NUMBER+J-6
            IF(L) 5, 6, 7
5           L2=1
               GO TO 10
6              L2=5
               GO TO 10
7              L2=7
10             CONTINUE
             ENDIF
         LAMBDA=ABS(L1-L2)
1000     CONTINUE
         RETURN
         END
```

B.2.5 The User Subrountines to Interpolate Stresses

```
      SUBROUTINE SIGINI(SIGMA,COORDS,NTENS,NCRDS,NOEL)
         IMPLICIT REAL*8(A-H,O-Z)
         DIMENSION SIGMA(NTENS),COORDS(NCRDS),STRESSIN(5000,3)
         DIMENSION LEL(5000,5),CDS(5000,3)
         COMMON/STIN/STRESSIN,CDS
         COMMON/IIN/ISWITCH,LEL
C*********************************
C   Obtain element and nodal data.
C*********************************
         IF (ISWITCH.EQ.0) THEN
             CALL GETFIELD
         ENDIF
C*******************************
C Interpolation
C*******************************
         INTER=0
         DO 1000 I=1,5000
C*******************************
C        Determine the position of the point.
C*******************************
```

```
      A=(CDS(LEL(I,2),1)-COORDS(1))*(CDS(LEL(I,3),2)-COORDS(2))/2
      A=A-(CDS(LEL(I,2),2)-COORDS(2))*(CDS(LEL(I,3),1)-COORDS(1))/2
      IF (A.LE.0) GO TO 1000
      A=(CDS(LEL(I,3),1)-COORDS(1))*(CDS(LEL(I,4),2)-COORDS(2))/2
      A=A-(CDS(LEL(I,3),2)-COORDS(2))*(CDS(LEL(I,4),1)-COORDS(1))/2
      IF (A.LE.0) GO TO 1000
      A=(CDS(LEL(I,4),1)-COORDS(1))*(CDS(LEL(I,5),2)-COORDS(2))/2
      A=A-(CDS(LEL(I,4),2)-COORDS(2))*(CDS(LEL(I,5),1)-COORDS(1))/2
      IF (A.LE.0) GO TO 1000
      A=(CDS(LEL(I,5),1)-COORDS(1))*(CDS(LEL(I,2),2)-COORDS(2))/2
      A=A-(CDS(LEL(I,5),2)-COORDS(2))*(CDS(LEL(I,2),1)-COORDS(1))/2
      IF (A.LE.0) GO TO 1000
C************************
C     Obtain normalized coordinates, Xi and Eta
C************************
      A1=CDS(LEL(I,2),1)+CDS(LEL(I,3),1)+CDS(LEL(I,4),1)+
     1    CDS(LEL(I,5),1)
      A2=CDS(LEL(I,2),2)+CDS(LEL(I,3),2)+CDS(LEL(I,4),2)+
     1    CDS(LEL(I,5),2)
      B1=-CDS(LEL(I,2),1)+CDS(LEL(I,3),1)+CDS(LEL(I,4),1)-
     1    CDS(LEL(I,5),1)
      B2=-CDS(LEL(I,2),2)+CDS(LEL(I,3),2)+CDS(LEL(I,4),2)-
     1    CDS(LEL(I,5),2)
      C1=-CDS(LEL(I,2),1)-CDS(LEL(I,3),1)+CDS(LEL(I,4),1)+
     1    CDS(LEL(I,5),1)
      C2=-CDS(LEL(I,2),2)-CDS(LEL(I,3),2)+CDS(LEL(I,4),2)+
     1    CDS(LEL(I,5),2)
      D1=CDS(LEL(I,2),1)-CDS(LEL(I,3),1)+CDS(LEL(I,4),1)-
     1    CDS(LEL(I,5),1)
      D2=CDS(LEL(I,2),2)-CDS(LEL(I,3),2)+CDS(LEL(I,4),2)-
     1    CDS(LEL(I,5),2)
      CALL XIETA(A1,A2,B1,B2,C1,C2,D1,D2,COORDS,XI,ETA,NCRDS)
      NEL=LEL(I,1)
1900 CONTINUE
      AN1=(1-XI)*(1-ETA)/4
      AN2=(1+XI)*(1-ETA)/4
      AN3=(1+XI)*(1+ETA)/4
      AN4=(1-XI)*(1+ETA)/4
      SIGMA(1)=STRESSIN(LEL(I,2),1)*AN1+STRESSIN(LEL(I,3),1)*AN2+
     1         STRESSIN(LEL(I,4),1)*AN3+STRESSIN(LEL(I,5),1)*AN4
```

```
      SIGMA(2)=STRESSIN(LEL(I,2),2)*AN1+STRESSIN(LEL(I,3),2)*AN2+
     1         STRESSIN(LEL(I,4),2)*AN3+STRESSIN(LEL(I,5),2)*AN4
      SIGMA(3)=STRESSIN(LEL(I,2),3)*AN1+STRESSIN(LEL(I,3),3)*AN2+
     1         STRESSIN(LEL(I,4),3)*AN3+STRESSIN(LEL(I,5),3)*AN4
      INTER=1
      GO TO 2000
1000  CONTINUE
2000 CONTINUE
C*************************************************
      IF (INTER.EQ.0) THEN
       WRITE(6,*) 'INTEGRATION POINT OUT OF OLD MESH!!!!!!!'
       WRITE(6,*) XI,ETA,NOEL,NCRDS,COORDS(1),COORDS(2)
       WRITE(6,*) STRESSIN(LEL(I,2),1),STRESSIN(LEL(I,3),1),
     1              STRESSIN(LEL(I,4),1),STRESSIN(LEL(I,5),1)
      ENDIF
C      WRITE(96,3000) COORDS(1),COORDS(2),SIGMA(1)
C      WRITE(97,3000) COORDS(1),COORDS(2),SIGMA(2)
C      WRITE(98,3000) COORDS(1),COORDS(2),SIGMA(3)
C3000 FORMAT(1X,2(F12.8,2X),E12.6)
C*************************************************
10000 CONTINUE
      RETURN
      END

      SUBROUTINE XIETA(A1,A2,B1,B2,C1,C2,D1,D2,COORDS,XI,ETA,NCRDS)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COORDS(NCRDS)
      AO=DMAX1(ABS(A1),ABS(B1),ABS(C1),ABS(D1),ABS(COORDS(1)))
      BO=DMAX1(ABS(A2),ABS(B2),ABS(C2),ABS(D2),ABS(COORDS(2)))
      A1=A1/AO
      B1=B1/AO
      C1=C1/AO
      D1=D1/AO
      A2=A2/BO
      B2=B2/BO
      C2=C2/BO
      D2=D2/BO
      DIV=B1*C2-B2*C1
      IF (ABS(DIV).LT.1.0E-10) GO TO 9999
      XI1=0.0
```

```
      ETA1=0.0
      NUMBER=0
100   CONTINUE
      XI=XI1
      ETA=ETA1
      XI1=(C2*(4*COORDS(1)/AO-A1-D1*XI*ETA)-
     1       C1*(4*COORDS(2)/BO-A2-D2*XI*ETA))/DIV
      ETA1=(-B2*(4*COORDS(1)/AO-A1-D1*XI*ETA)+
     1        B1*(4*COORDS(2)/BO-A2-D2*XI*ETA))/DIV
      NUMBER=NUMBER+1
      IF (NUMBER.GT.20) GO TO 9998
      IF (ABS(XI).LE.0.0001) THEN
           XI2=ABS(XI1-XI)
     ELSE
          XI2=ABS((XI1-XI)/XI)
     ENDIF
     IF (ABS(ETA).LE.0.0001) THEN
           ETA2=ABS(ETA1-ETA)
     ELSE
          ETA2=ABS((ETA1-ETA)/ETA)
     ENDIF
      IF (XI2.GT.0.00001) GO TO 100
      IF (ETA2.GT.0.00001) GO TO 100
      XI=XI1
      ETA=ETA1
      IF (ABS(XI).GT.1.0) GO TO 9997
      IF (ABS(ETA).GT.1.0) GO TO 9997
      GO TO 10000
9999  WRITE(6,*) 'INTERPOLATION PROBLEM!!!!!!!!!!!!!!'
      WRITE(6,1001) A1,B1,C1,D1
      WRITE(6,1001) A2,B2,C2,D2
1001  FORMAT(1X,4(',',F15.9))
      GO TO 10000
9998  WRITE(6,*) 'TOO MANY ITERATIONS!!!!!!!!!!!!'
      GO TO 10000
9997  WRITE(6,*) 'INCORRECT INTERPOLATION!!!!!!!!!!!!!!!!'
10000 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GETFIELD
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/STIN/STRESSIN(5000,3),CDS(5000,3)
      COMMON/IIN/ISWITCH,LEL(5000,5)
C
      ISWITCH=1
      OPEN(UNIT=90,TYPE='OLD')
      DO 1000 I=1,6000
      READ(90,1100,END=1500) II,CDS1,CDS2,CDS3
      CDS(II,1)=CDS1
      CDS(II,2)=CDS2
      CDS(II,3)=CDS3
1000  CONTINUE
1100   FORMAT(1X,I5,3(',',F15.12),3(',',F14.2))
1500  CONTINUE
      CLOSE(UNIT=90)
      OPEN(UNIT=92,TYPE='OLD')
      OPEN(UNIT=93,TYPE='OLD')
      DO 2000 I=1,5000
      READ(92,2100,END=2500) LEL(I,1),LEL(I,2),LEL(I,3),LEL(I,4),
     1                       LEL(I,5)
2000  CONTINUE
2100  FORMAT(1X,I5,4(',',1X,I5))
2500  CONTINUE
      II=I
      OPEN(UNIT=93,TYPE='OLD')
      DO 2010 I=II,II+5000
      READ(93,2100,END=2600) LEL(I,1),LEL(I,2),LEL(I,3),LEL(I,4),
     1                       LEL(I,5)
2010  CONTINUE
2600  CONTINUE
      CLOSE(UNIT=92)
      CLOSE(UNIT=93)
      OPEN(UNIT=91,TYPE='OLD')
      DO 3000 I=1,6000
      READ(91,3100,END=3500),NODNUM,STRESS1,STRESS2,STRESS3
      STRESSIN(NODNUM,1)=STRESS1
      STRESSIN(NODNUM,2)=STRESS2
      STRESSIN(NODNUM,3)=STRESS3
3000  CONTINUE
```

```
3100    FORMAT(1X,I5,3(',',E15.5))
3500    CONTINUE
C*****************************************
C        OPEN(UNIT=96,FILE='S11.DAT',TYPE='NEW')
C        OPEN(UNIT=97,FILE='S22.DAT',TYPE='NEW')
C        OPEN(UNIT=98,FILE='S12.DAT',TYPE='NEW')
C*****************************************
        RETURN
        END
```

B.2.6 Interaction Forces between the Local Zone and Substructures

```
      PROGRAM INTLOAD
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER NULL*2
      DIMENSION STRESS(6000,3),NODRET(6000),ANORM(6000,3)
        DIMENSION SIGMA(3),CD(3),FORCE(6000,2)
        DIMENSION LEL(6000,5),COORDS(6000,3),CDS(6000,3)
      COMMON/RTNODE/NODRET
      COMMON/ANORMAL/ANORM
        COMMON/STIN/STRESS
        COMMON/CDS1/CDS
        COMMON/IIN/LEL
      COMMON/CDS2/COORDS
      COMMON/FC/FORCE
*
*
      OPEN (UNIT=20,STATUS='NEW')
      OPEN (UNIT=21,STATUS='NEW')
      OPEN (UNIT=22,STATUS='NEW')
C
C
      CALL RDF(NUMBER,ISNUMBER)
C
C
C     Retained ΓOF definition for super-element
C
2001  FORMAT(16I5)
```

```
2002  FORMAT(15H*NSET,NSET=RDOF)
2005  FORMAT(14H*RETAINED DOFS)
2006  FORMAT(4HRDOF)
      WRITE(22,2002)
      NUM=NUMBER/16
      DO 2100 I=0,NUM-1
      N=I*16
      WRITE(22,2001) (NODRET(N+J),J=1,16)
2100  CONTINUE
      J=NUMBER-NUM*16
      N=NUM*16
2101  FORMAT(<J>I5)
      WRITE(22,2101) (NODRET(N+K),K=1,J)
      WRITE(22,2005)
      WRITE(22,2006)
      CLOSE(UNIT=22)
C
C     SET definition of retained dof in the main structure.
C
2501  FORMAT(28H*USE SUPER,ID=Z001,ELSET=SUP)
2502  FORMAT(16I5)
      WRITE(21,2501)
      II=9999
      IF (NUMBER.LE.15) THEN
         J=NUMBER+1
2503     FORMAT(<J>I5)
         WRITE(21,2503) II,(NODRET(K),K=1,J-1)
      ELSE
         NUM=(NUMBER+1)/16
         WRITE(21,2502) II,(NODRET(K),K=1,15)
         DO 2600 I=1,NUM-1
         N=I*16-1
         WRITE(21,2502) (NODRET(N+J),J=1,16)
2600     CONTINUE
         J=NUMBER+1-NUM*16
         IF (J.EQ.0) GO TO 2505
         N=NUM*16-1
2504     FORMAT(<J>I5)
         WRITE(21,2504) (NODRET(N+K),K=1,J)
      ENDIF
```

```
2505  CONTINUE
      CLOSE(UNIT=21)
C
C     Determine loading between the interface
C
      IF (ISNUMBER.EQ.1) GO TO 3200
C
C     Read stress information.
C
      OPEN (UNIT=91,STATUS='OLD')
1     FORMAT(1X,I5,4(',',1X,I5))
140   FORMAT(1X,I5,3(',',E15.5))
      DO 1000 I=1,10000
      READ(91,140,END=1010) NODE,STRESS1,STRESS2,STRESS3
      STRESS(NODE,1)=STRESS1
      STRESS(NODE,2)=STRESS2
      STRESS(NODE,3)=STRESS3
1000  CONTINUE
1010  CONTINUE
      CLOSE(UNIT=91)
C
C     Read coordinate information.
C
      OPEN (UNIT=90,STATUS='OLD')
      DO 4000 I=1,10000
      READ(90,150,END=4010) NODE,CDS1,CDS2,CDS3
      CDS(NODE,1)=CDS1
      CDS(NODE,2)=CDS2
      CDS(NODE,3)=CDS3
4000  CONTINUE
4010  CONTINUE
      CLOSE(UNIT=90)
150   FORMAT(1X,I5,3(',',F15.12))
160   FORMAT(1X,I5,3(',',F16.13))
      OPEN (UNIT=18,STATUS='OLD')
      DO 4700 I=1,10000
      READ(18,160,END=4710) NODE,CDS1,CDS2,CDS3
      COORDS(NODE,1)=CDS1
      COORDS(NODE,2)=CDS2
      COORDS(NODE,3)=CDS3
```

```
4700  CONTINUE
4710  CONTINUE
      CLOSE(UNIT=18)
        OPEN(UNIT=92,TYPE='OLD')
        OPEN(UNIT=93,TYPE='OLD')
        DO 7000 I=1,5000
        READ(92,7100,END=7500) LEL(I,1),LEL(I,2),LEL(I,3),LEL(I,4),
     1                         LEL(I,5)
7000    CONTINUE
7100    FORMAT(1X,I5,4(',',1X,I5))
7500    CONTINUE
        II=I
        OPEN(UNIT=93,TYPE='OLD')
        DO 7010 I=II,II+5000
        READ(93,7100,END=7600) LEL(I,1),LEL(I,2),LEL(I,3),LEL(I,4),
     1                         LEL(I,5)
7010    CONTINUE
7600    CONTINUE
        CLOSE(UNIT=92)
        CLOSE(UNIT=93)
C
C       Interpolate.
C
      DO 4100 I=1,NUMBER
      SW=ABS(STRESS(NODRET(I),1))+ABS(STRESS(NODRET(I),2))
      SW=SW+ABS(STRESS(NODRET(I),3))
      IF (SW.GT.0.0) THEN
      ELSE
      CD(1)=COORDS(NODRET(I),1)
      CD(2)=COORDS(NODRET(I),2)
      CD(3)=COORDS(NODRET(I),3)
      CALL SIGINI(SIGMA,CD,3,3,1)
      STRESS(NODRET(I),1)=SIGMA(1)
      STRESS(NODRET(I),2)=SIGMA(2)
      STRESS(NODRET(I),3)=SIGMA(3)
      ENDIF
4100  CONTINUE
C
3001  FORMAT(21H*CLOAD,AMPLITUDE=INTL)
3002  FORMAT(I5,',',I3,',',D16.9)
```

```
      WRITE(20,3001)
      ELLENG=0.005
      DO 3090 I=1,NUMBER
      FORCE(I,1)=0.0
      FORCE(I,2)=0.0
3090  CONTINUE
      DO 3100 I=1,NUMBER-1
      SW=ABS(ANORM(I,1))+ABS(ANORM(I,2))
      IF (SW.GT.1.0) THEN
         AN1=ANORM(I+1,1)
         AN2=ANORM(I+1,2)
      ELSE
         AN1=ANORM(I,1)
         AN2=ANORM(I,2)
      ENDIF
      NODE=NODRET(I)
      LAYER=NODE/25
      ISW=NODE-25*LAYER
      IF (ISW.EQ.0) THEN
         NODE=NODRET(1)
         II=1
      ELSE
         II=I
      ENDIF
      NODE1=NODRET(I+1)
      ELL=DIS(COORDS(NODRET(I),1),COORDS(NODRET(I),2),
     1        COORDS(NODRET(I+1),1),COORDS(NODRET(I+1),2))
      FORCE(II,1)=FORCE(I,1)+ELL*(ANORM(I,1)*(STRESS(NODE,1)/3.0+
     1           STRESS(NODE1,1)/6.0)+ANORM(I,2)*(STRESS(NODE,3)/3.0+
     2           STRESS(NODE1,3)/6.0))
      FORCE(II,2)=FORCE(I,2)+ELL*(ANORM(I,1)*(STRESS(NODE,3)/3.0+
     1           STRESS(NODE1,3)/6.0)+ANORM(I,2)*(STRESS(NODE,2)/3.0+
     2           STRESS(NODE1,2)/6.0))
      FORCE(I+1,1)=FORCE(I+1,1)+ELL*(ANORM(I,1)*(STRESS(NODE,1)/6.0+
     1           STRESS(NODE1,1)/3.0)+ANORM(I,2)*(STRESS(NODE,3)/6.0+
     2           STRESS(NODE1,3)/3.0))
      FORCE(I+1,2)=FORCE(I+1,2)+ELL*(ANORM(I,1)*(STRESS(NODE,3)/6.0+
     1           STRESS(NODE1,3)/3.0)+ANORM(I,2)*(STRESS(NODE,2)/6.0+
     2           STRESS(NODE1,2)/3.0))
3100  CONTINUE
```

```
      CALL CORRECTION(XO,YO,CO,C1,C2,RX,RY,NUMBER,ELLENG)
      ID1=1
      ID2=2
      DO 3150 I=1,NUMBER
      FACTOR=C1*(COORDS(NODRET(I),2)-YO)/ELLENG
      FACTOR=(CO+FACTOR)/REAL(NUMBER)
      F1=FACTOR*(-RX)
      FACTOR=C2*(COORDS(NODRET(I),1)-XO)/ELLENG
      FACTOR=(CO+FACTOR)/REAL(NUMBER)
      F2=FACTOR*(-RY)
      FORCE(I,1)=FORCE(I,1)+F1
      FORCE(I,2)=FORCE(I,2)+F2
      WRITE(20,3002) NODRET(I),ID1,FORCE(I,1)
      WRITE(20,3002) NODRET(I),ID2,FORCE(I,2)
3150  CONTINUE
3200  CONTINUE
      CLOSE(UNIT=20)
      STOP
      END

      SUBROUTINE RDF(NUMBER,ISNUMBER)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION NODRET(6000),ANORM(6000,3)
      COMMON/RTNODE/NODRET
      COMMON/ANORMAL/ANORM
2     FORMAT(2A)
      OPEN (UNIT=19,STATUS='OLD')
      READ(19,2) NULL
      READ(19,*) ISNUMBER
      READ(19,*) ITOTALNUM
      CLOSE(UNIT=19)
      LMINUS1=3
      LAST=10
      MAXLL=(ISNUMBER+LMINUS1)*192/(ITOTALNUM+LMINUS1)
      IF (MAXLL.GT.192) THEN
        MAXLL=192
      ENDIF
      MINLL=(ISNUMBER-LAST)*192/(ITOTALNUM+LMINUS1)
      IF (MINLL.LT.0) THEN
        MINLL=0
```

```
      ENDIF
      A=REAL(MINLL/4)
      B=REAL(MINLL)/4.0
      IF (A.NE.B) THEN
         MINLL=INT(B+0.5)*4
      ENDIF
      A=REAL((192-MAXLL)/4)
      B=REAL(192-MAXLL)/4.0
      IF (A.NE.B) THEN
         MAXLL=192-INT(B+0.5)*4
      ENDIF
      IF (MAXLL.EQ.192) THEN
          MAXLL1=192
      ELSE
          MAXLL1=MAXLL-4
      ENDIF
      IF (MINLL.EQ.0) THEN
          MINLL1=0
      ELSE
          MINLL1=MINLL+4
      ENDIF
C
C
      NUMBER=0
      IF (MINLL.EQ.0) THEN
         NUMBER=NUMBER+1
         NODRET(NUMBER)=6
         ANORM(NUMBER,1)=1.0
         ANORM(NUMBER,2)=0
      ELSE
         NUMBER=NUMBER+1
         NODE=MINLL1*25
         NODRET(NUMBER)=NODE+6
         ANORM(NUMBER,1)=1.0
         ANORM(NUMBER,2)=-1.0
      DO 200 J=9,21,4
         NUMBER=NUMBER+1
         NODRET(NUMBER)=NODE+J
         ANORM(NUMBER,1)=0.0
         ANORM(NUMBER,2)=-1.0
```

```
200   CONTINUE
         NUMBER=NUMBER+1
         NODRET(NUMBER)=NODE+25
         ANORM(NUMBER,2)=-1.0
         ANORM(NUMBER,1)=0.0
      ENDIF
      DO 100 I=MINLL1+4,MAXLL1-4,4
         NUMBER=NUMBER+1
         NODRET(NUMBER)=(I)*25+6
         ANORM(NUMBER,1)=1.0
         ANORM(NUMBER,2)=0.0
100   CONTINUE
      IF (MAXLL.EQ.192) THEN
         NUMBER=NUMBER+1
         NODRET(NUMBER)=192*25+6
         ANORM(NUMBER,1)=1.0
         ANORM(NUMBER,2)=0.0
      ELSE
         NODE=MAXLL1*25
         NUMBER=NUMBER+1
         NODRET(NUMBER)=NODE+6
         ANORM(NUMBER,1)=1.0
         ANORM(NUMBER,2)=1.0
      DO 210 J=9,21,4
         NUMBER=NUMBER+1
         NODRET(NUMBER)=NODE+J
         ANORM(NUMBER,1)=0.0
         ANORM(NUMBER,2)=1.0
210   CONTINUE
         NUMBER=NUMBER+1
         NODRET(NUMBER)=NODE+25
         ANORM(NUMBER,2)=1.0
         ANORM(NUMBER,1)=0.0
      ENDIF
C
C
      RETURN
      END

      FUNCTION DIS(X1,Y1,X2,Y2)
```

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIS=SQRT((X2-X1)*(X2-X1)+(Y2-Y1)*(Y2-Y1))
      RETURN
      END

      SUBROUTINE CORRECTION(XO,YO,CO,C1,C2,RX,RY,NUMBER,ELLENG)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION NODRET(6000),FORCE(6000,2),COORDS(6000,3)
      COMMON/RTNODE/NODRET
      COMMON/CDS2/COORDS
      COMMON/FC/FORCE
C
C     Find the lower-left conner.
C
      XMIN=1.0D+10
      YMIN=1.0D+10
      XMAX=-1.0E+10
      YMAX=-1.0E+10
C
      DO 100 I=1,NUMBER
      IF (COORDS(NODRET(I),1).LT.XMIN) THEN
         XMIN=COORDS(NODRET(I),1)
      ENDIF
      IF (COORDS(NODRET(I),2).LT.YMIN) THEN
         YMIN=COORDS(NODRET(I),2)
      ENDIF
      IF (COORDS(NODRET(I),1).GT.XMAX) THEN
         XMAX=COORDS(NODRET(I),1)
      ENDIF
      IF (COORDS(NODRET(I),2).GT.YMAX) THEN
         YMAX=COORDS(NODRET(I),2)
      ENDIF
100   CONTINUE
      XO=(XMIN+XMAX)/2.0
      YO=(YMIN+YMAX)/2.0
C
C     Find residual forces and moment.
C
      SUMX=0.0
      SUMY=0.0
```

```
      SUMXY=0.0
      SUMXX=0.0
      SUMYY=0.0
      RX=0.0
      RY=0.0
      RM=0.0
      DO 200 I=1,NUMBER
      RX=RX+FORCE(I,1)
      RY=RY+FORCE(I,2)
      RM=RM+FORCE(I,2)*(COORDS(NODRET(I),1)-XO)
      RM=RM-FORCE(I,1)*(COORDS(NODRET(I),2)-YO)
      SUMX=SUMX+COORDS(NODRET(I),1)-XO
      SUMY=SUMY+COORDS(NODRET(I),2)-YO
      SUMXY=SUMXY+(COORDS(NODRET(I),1)-XO)*(COORDS(NODRET(I),2)-YO)
      SUMXX=SUMXX+(COORDS(NODRET(I),1)-XO)*(COORDS(NODRET(I),1)-XO)
      SUMYY=SUMYY+(COORDS(NODRET(I),2)-YO)*(COORDS(NODRET(I),2)-YO)
200   CONTINUE
C
C     Determine correction constants.
C
      AN=REAL(NUMBER)
      ANL=ELLENG*REAL(NUMBER)
      DENOM=(RX*SUMY-RY*SUMX)/AN/RM-RX*SUMYY/SUMY/RM+
     1      RY*SUMXX/SUMX/RM
      CO=-(1+RX*SUMYY/RM/SUMY-RY*SUMXX/RM/SUMX)/DENOM
      C1=ANL*(1-CO)/SUMY
      C2=ANL*(1-CO)/SUMX
      WRITE(6,*) 'ORIGIN:',XO,YO
      WRITE(6,*) 'RESIDUAL:',RX,RY,RM
      WRITE(6,*) 'CONSTANT:',CO,C1,C2
      RETURN
      END
```