THE MECHANICS OF WEBS ENCOUNTERING CONCAVE ROLLS

By

MICHAEL LOUIS LEPORT

Bachelor of Science in Mechanical Engineering

Oklahoma State University

Stillwater, Oklahoma

1985

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
MASTER OF SCIENCE
May, 1987

THE MECHANICS OF WEBS ENCOUNTERING CONCAVE ROLLS

Thesis Approved:

James K. Good
Thesis Adviser

R. L. Lowery

Gary E. Young

Norman N. Durham
Dean of the Graduate College

ACKNOWLEDGMENTS

I sincerely appreciate the aid of those who helped during this study. I would like to thank and express my appreciation to Dr. J. K. Good, my graduate adviser, for his instruction, advice, and support during my graduate study. I would also like to thank Dr. R. L. Lowery and Dr. G. E. Young for their participation as committee members.

My appreciation is expressed to Ms. Charlene Fries for her excellent work in preparing this manuscript.

I am grateful to Sankaran Mohan for his assistance with the computer systems and codes. I would also like to thank Jon Baker for his work in developing a computer code to plot the model and stresses for this study.

I am deeply grateful to my family, especially my parents, whose support and encouragement made my education possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Web handling involves the transport of thin flexible materials called "webs" through manufacturing systems. Webs, which have little lateral compressive stiffness, often wrinkle during transport. The use of concave rolls in web process lines has been a common practice for some time. They are often used to correct seam bows and other misalignments but they most commonly see duty as spreading devices to remove or prevent wrinkles. In spite of this, they are perhaps the least understood and most controversial of the spreading devices.

The concave roll, by definition, has a larger diameter at the outer edges of the roll than in the center. The exact profile used is dependent on the particular problem and application, and many times the profile is the result of guesswork. The roll, by virtue of its profile, creates a speed effect and tends to move material faster at its outer edges than at the center. This profile also induces a spreading effect as the material climbs toward the larger diameter at the edges.

Before one can begin to understand and predict the behavior of concave rolls on-line, an understanding of the mechanics of the web on the roll is essential. At present, no real attempt has been made to solve this problem. In industry certain rules of thumb exist, but these are not universally applicable; and trial-and-error methods are often resorted to. Therefore, the development of a method to analyze the interaction between a web and concave roll is in order.

1.1 Objectives

The objectives of this study are:

1. To develop a general finite element model which suitably approximates the doubly curved geometry of a web on a concave roll.

2. To develop a finite element computer code to analyze the model. The code must be able to iteratively spread the web to the point at which the friction forces between the web and roll are equal to the spreading forces.

3. To investigate the effects of spreading on the entry and exit spans and upon the concave roll for circular arc and linear taper profiles.

1.2 Literature Survey

As previously stated, no conclusive work has been done on this problem. The work most closely related has been done on belt pulleys.

In the early part of this century, one of the most common ways of transmitting power was through a belt and pulley system. The interaction between a belt and pulley is not unlike the interaction of a web and roll. Smith [1] investigates the principles underlying the action of various pulley profiles and geometries and from these arrives at a theory of pulley camber. The author discusses the effects of camber with no regard to drive geometry and then presents a detailed discussion of the combined effects of drive geometry and pulley camber. Particular attention is given to the use of pulley camber to correct for inconsistencies in drive geometry. Smith puts forth the theory that at the heart of camber action is the bending couple induced in the free entry span which corresponds directly to the pulley camber. He emphasizes that this assumption does not hold for the exit of the pulley. Smith also shows mathematically how and why belts always tend to climb toward the pulley crown which directly relates to the climbing (spreading) action of the concave roll.

In a more recent work, Butler [2] discusses the use of concave rolls to correct symmetrical bowing of a dryer fabric seam. The paper discusses locations for the concave roll as well as locations to avoid. A few estimates of the amount of concavity needed for various fabric widths are also presented.

An information brochure by United States Steel Corporation [3] discusses cylindrical, crowned, and concave rolls and the theory of "planar action" as it applies to each roll. Planar action is defined as the action between the rotating roll surface and the material being transported. This theory also accounts for the lateral movement of a web strip on a roll. If a flat strip is perfectly aligned on the roll, there will be no lateral movement. If, however, the strip enters the roll at an angle, the strip will follow a helical path over the roll and lateral movement will result (Figure 1). The lateral velocity of the web strip across the roll is given by

$$V_L = V\tan\theta \tag{1.1}$$

where V is the roller surface speed, and θ is the helix angle. Therefore, the web will be laterally static only when it is normal to the roll axis ($\theta = 0$). It is known that the web seeks this position.

A web strip entering a concave roll will, due to the geometry of the roll, take a helical path over the roll. It is shown in Figure 2 that the helix angle increases as the strip moves to the outside edge as does the

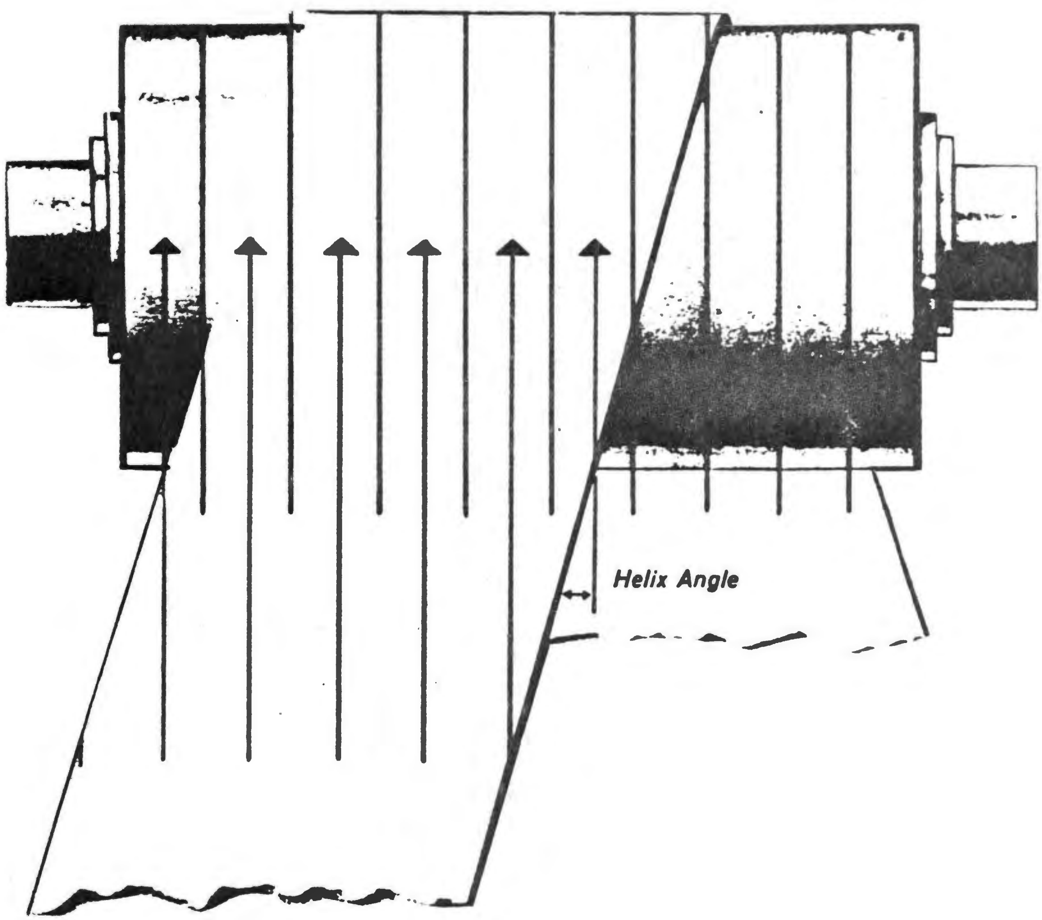


Figure 1. Web Strip on Roll

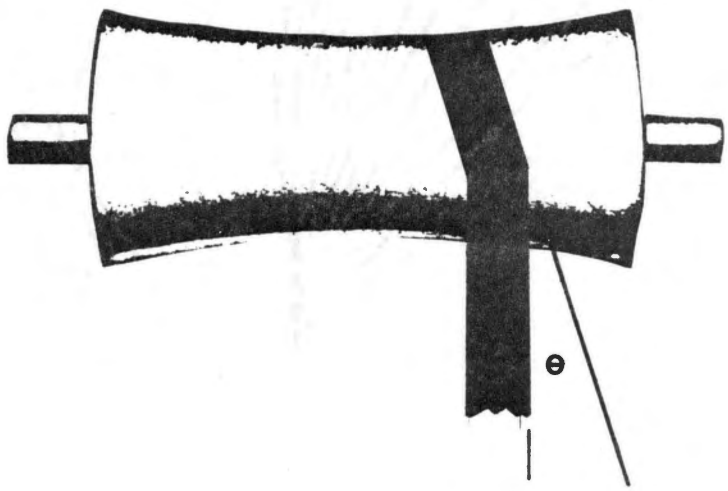


Figure 2. Helix Angle for a Web Strip on a Concave Roll

surface speed of the roll. These effects account for the spreading action of the concave roll.

Daly [4] discusses the effects of various process parameters on the traction between the web and the roll. He presents the concept that web movement on a roll only occurs when the condition of traction changes, and the ability of a roll to guide or spread a web is dependent on the traction between the two.

Soong and Li [5] present a general method for analyzing the dynamic behavior of a web strip on a friction cylinder.

1.3 Organization

Chapter II consists of the finite element theory behind this study. A description of the finite element model geometry and necessary coordinate transformations is presented. Chapter III provides a description of the computer code developed based on the theory of Chapter II. The assumptions made in modeling the web and roll are also presented in Chapter III. Chapter IV presents the results of this study. The data are presented using plots and tables for better understanding. Chapter V presents the conclusions based on the results of this study and includes recommendations for future study.

CHAPTER II

DEVELOPMENT OF THE FINITE ELEMENT EQUATIONS

The fundamental finite element relationships used in this study are presented in this chapter. Familiarity of the reader with the finite element method is assumed.

One objective of this study involves calculating the friction forces between the web and roll. In order to calculate these friction forces, normal forces must be determined. Therefore, it is necessary to know the direction normal to the roll at each node on the roll. This normal coordinate is also necessary for enforcing the roll surface in the model giving it rigidity in space. The normals to the roll are determined through coordinate transformations. The development of the transformation matrices needed for this study is discussed in section 2.2. These matrices are based on a method for assembling a shell from flat elements as presented by Zienkiewicz [6]. Section 2.3 discusses the use of these matrices to transform the stiffness matrices. A discussion of the average coordinates used is also presented in this section.

2.1 Fundamental Relationships

In a two-dimensional elasticity problem, there are two unknown displacements: $u(x,y)$ and $v(x,y)$. The object of a finite element analysis is to determine the relationships between these displacements and the two coordinate directions.

In the finite element method, these relationships are approximated over individual elements by continuous, piecewise smooth shape functions. In general, these relationships can be written as

$$\begin{Bmatrix} u(x,y) \\ v(x,y) \end{Bmatrix} = [N]\ \{U^{(e)}\} \tag{2.1a}$$

where [N] is the matrix of shape functions, and U is the column vector of the element nodal displacements given by

$$\{U^{(e)}\} = [U_{2i-1}\ U_{2i}\ U_{2j-1}\ U_{2j}\ U_{2k-1}\ U_{2k}]^T \tag{2.1b}$$

A strain vector and stress vector may be defined as

$$\{\varepsilon\} = [\varepsilon_x\ \varepsilon_y\ \gamma_{xy}]^T \tag{2.2a}$$

$$\{\sigma\} = [\sigma_x\ \sigma_y\ \sigma_{xy}]^T \tag{2.2b}$$

respectively. The strain components in $\{\varepsilon\}$ are related to the displacements by the well-known strain-displacement equations. These equations are given by

$$\varepsilon_x = \frac{\partial u}{\partial x} \qquad \varepsilon_y = \frac{\partial v}{\partial y} \qquad \gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \tag{2.3}$$

and are valid for small strains. A general equation relating the nodal displacements to the strain components can be written as

$$\{\varepsilon\} = [B]\ \{U^{(e)}\} \tag{2.4}$$

where [B] is known as the gradient matrix and is formed by differentiating the displacement equations for u(x,y) and v(x,y). The stress and strain vectors are also related and this relationship can be written as

$$\{\sigma\} = [D]\ \{\varepsilon\} \tag{2.5}$$

where [D] is the matrix of constitutive relations given by

$$[D] = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1 - \nu)}{2} \end{bmatrix} \tag{2.6}$$

The contribution of each element to the system of equations is through the element's stiffness matrix and force vector. The element stiffness matrix for a triangular plane stress element is given by

$$[K^{(e)}] = [B]^T \, [D] \, [B] \, tA \tag{2.7}$$

where t is the thickness of the element, and A is its area. The element nodal displacements are found using the relationship

$$\{U^{(e)}\} = [K^{(e)}]^{-1} \, \{F\} \tag{2.8}$$

where {F} is the applied force vector. With the nodal displacements known, the strain and stress vectors may be determined using Equations (2.4) and (2.5).

2.2 The Transformation Matrices

The element stiffness matrix $[K^{(e)}]$ defined by Equation (2.7) is computed using nodal coordinates in the element's local coordinate system. Assembly of the element stiffness matrices into one common global stiffness matrix requires that the element matrices be transformed to the global set of coordinates prior to assembly.

In general, the relationship between the local and global systems can be written as

$$\begin{Bmatrix} x' \\ y' \\ z' \end{Bmatrix} = [\lambda] \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} \tag{2.9}$$

where x'y'z' represents the local system, and xyz represents the global system. The matrix [λ] is a 3x3 matrix of direction cosines of the angles formed between the two sets of axes (Figure 3). This matrix is given by

$$[\lambda] = \begin{bmatrix} \lambda_{x'x} & \lambda_{x'y} & \lambda_{x'z} \\ \lambda_{y'x} & \lambda_{y'y} & \lambda_{y'z} \\ \lambda_{z'x} & \lambda_{z'y} & \lambda_{z'z} \end{bmatrix} \tag{2.10}$$

in which $\lambda_{x'x}$ is the cosine of the angle between the x and x' axes and so forth for the other entries. Following is a general method for determining the direction cosines in the [λ] matrix.

For a triangular element in space, one choice of local axis direction is arbitrary. For convenience the x' axis will always be directed along the side ij of the triangle as shown in Figure 3. The vector V_{ij}, which defines this side, may be written in terms of the global coordinates as

$$V_{ij} = \begin{Bmatrix} x_j - x_i \\ y_j - y_i \\ z_j - z_i \end{Bmatrix} \tag{2.11}$$

The direction cosines are obtained by dividing each component of this vector by its length, giving

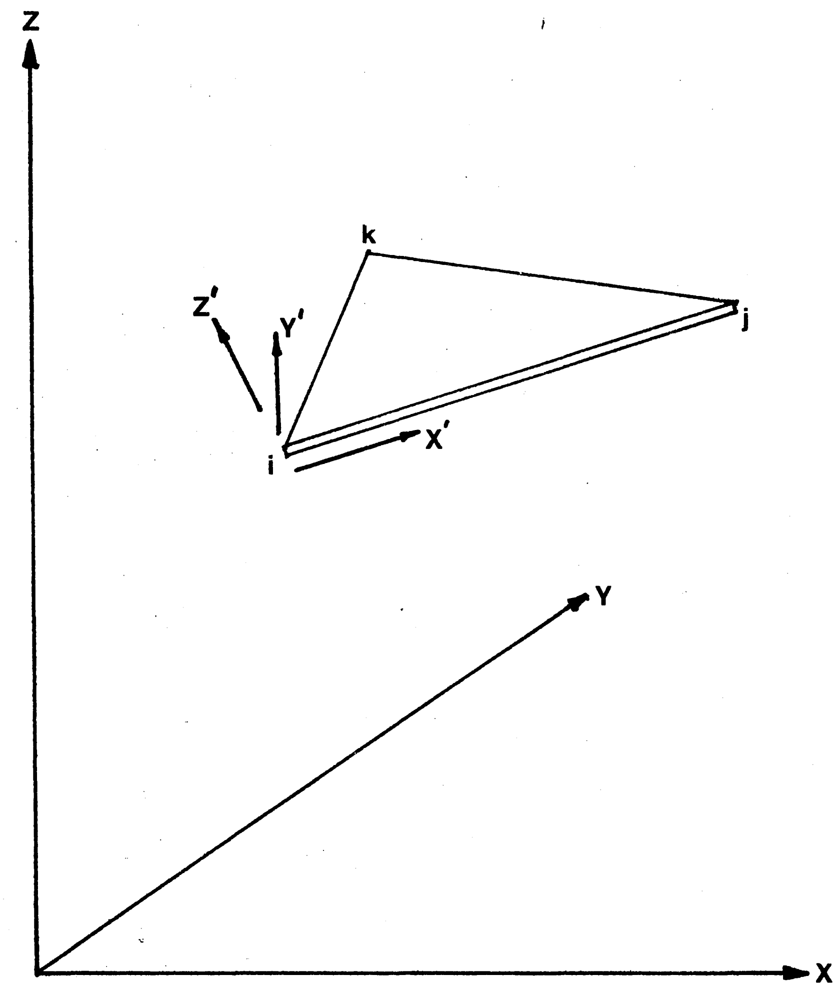


Figure 3. Local and Global Coordinates
for a Triangular Element

$$v_{x'} = \begin{Bmatrix} \lambda_{x'x} \\ \lambda_{x'y} \\ \lambda_{x'z} \end{Bmatrix} = \frac{1}{L_{ij}} \begin{Bmatrix} x_{ji} \\ y_{ji} \\ z_{ji} \end{Bmatrix} \tag{2.12}$$

where

$$L_{ij} = \sqrt{x_{ji}^2 + y_{ji}^2 + z_{ji}^2}$$

and $x_{ji} = x_j - x_i$, etc. The vector V_{ik} of side ik is derived in the same manner and is given by

$$V_{ik} = \begin{Bmatrix} x_k - x_i \\ y_k - y_i \\ z_k - z_i \end{Bmatrix} \tag{2.13}$$

The direction cosines for the z' direction, which is normal to the plane of the triangle, are obtained by the vector cross product of the two sides V_{ij} and V_{ik} of the triangle. The direction cosines may be written as

$$v_{z'} = \begin{Bmatrix} \lambda_{z'x} \\ \lambda_{z'y} \\ \lambda_{z'z} \end{Bmatrix} = \frac{1}{2A} \begin{Bmatrix} y_{ji}z_{ki} - y_{ki}z_{ji} \\ -x_{ji}z_{ki} - x_{ki}z_{ji} \\ x_{ji}y_{ki} - x_{ki}y_{ji} \end{Bmatrix} \tag{2.14}$$

where 2A is twice the area of the triangle and is, by definition, given by the length of vector $v_{z'}$.

The direction cosines of the y' direction are found in a similar manner as the direction cosines of a vector normal to the x' and z' directions. This vector may be written as

$$v_{y'} = \begin{Bmatrix} \lambda_{y'x} \\ \lambda_{y'y} \\ \lambda_{y'z} \end{Bmatrix} = \begin{Bmatrix} \lambda_{z'y}\lambda_{x'z} - \lambda_{z'z}\lambda_{x'y} \\ -\lambda_{z'x}\lambda_{x'z} + \lambda_{z'z}\lambda_{x'x} \\ \lambda_{z'x}\lambda_{x'y} - \lambda_{z'y}\lambda_{x'x} \end{Bmatrix} \tag{2.15}$$

With all of the elements of the matrix $[\lambda]$ known, the required coordinate transformations may be carried out by the methods presented in the next section.

2.3 Transformation of the Stiffness Matrices

By the rules of orthogonal transformation, the stiffness matrix in global coordinates may be determined from the following relation:

$$[K^{(e)}_{Global}] = [T]^T \; [K^{(e)}_{Local}] \; [T] \tag{2.16a}$$

where

$$[T] = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix} \tag{2.16b}$$

in which λ represents the $[\lambda]$ matrix of Equation (2.10), and 0 represents a 3x3 matrix of zeroes. In order to make the multiplication of Equation (2.16a) possible, the local element stiffness matrix must be expanded to a 9x9 matrix. This is accomplished by inserting zero entries in the third, sixth, and ninth rows and columns. These rows and columns correspond to the out-of-plane stiffness entries which, for a membrane element, are negligible in the local system. The components of the vector $\{U^{(e)}\}$ of nodal displacements given in Equation (2.1b) are now as shown in Figure 4.

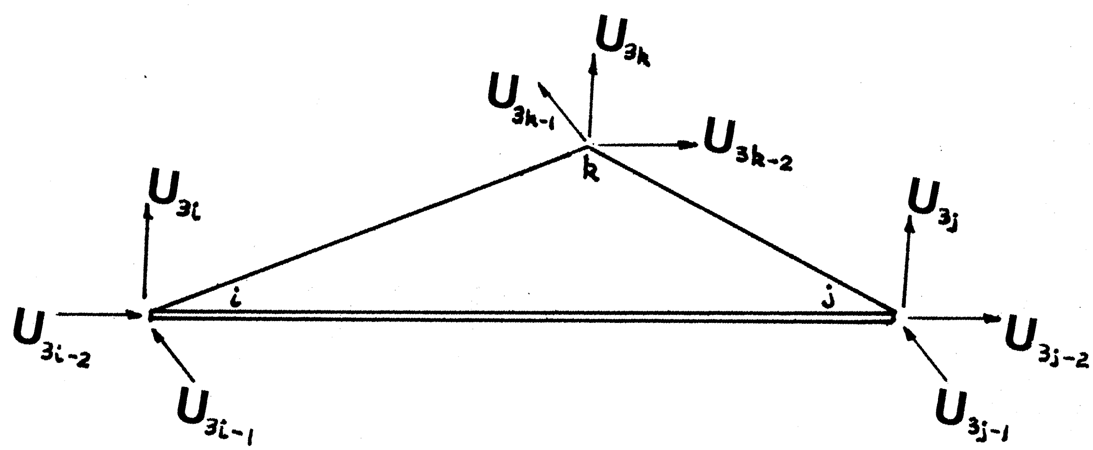


Figure 4. Nodal Displacements for a Triangular Element

After the global stiffness matrix has been assembled, the matrix is transformed to a system of average local coordinates by an average transformation matrix $[\bar{\lambda}_n]$ so that displacements and constraints may be properly enforced (see section 3.1). It is important to note that whereas the matrix $[\lambda]$ was associated with one particular element, the matrix $[\bar{\lambda}_n]$ is associated with a node but has the same form as $[\lambda]$. The $[\bar{\lambda}_n]$ matrix is determined by first computing the direction cosines of the surrounding four planes as shown in Figure 5. These values are then averaged to determine the $[\lambda]$ matrix for that node. If the node is on one of the boundaries of the model, then the average consists of only the two planes adjacent to the node. The transformation to the average coordinates consists of premultiplying the rows of the stiffness matrix corresponding to the degrees of freedom for a particular node by $[\bar{\lambda}_n]$ and then postmultiplying the appropriate columns of the stiffness matrix by $[\bar{\lambda}_n]^T$. Now each node on the model has its own coordinate system associated with it and proper constraints can be enforced.

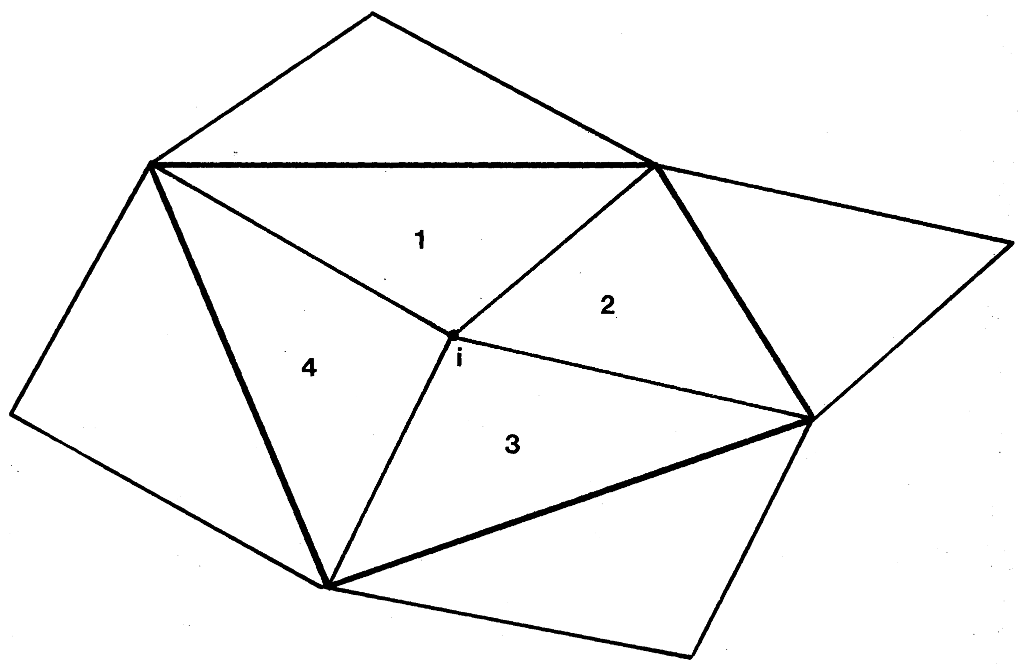


Figure 5. Four Averaging Planes for Node i

CHAPTER III

ANALYTICAL STUDY

The purpose of this study is to investigate the spreading effect of a concave roll. Using the theory and relationships presented in Chapters I and II as a basis, an analytical model of a web on a concave roll was developed.

Descriptions of how the web and roll are modeled and the assumptions governing this study are presented in this chapter. Descriptions of the FORTRAN programs MSHGNR.FOR and CONCAVE.FOR used in this study are also presented. Copies of these codes are included in the Appendix.

3.1 The Finite Element Model

The web on the concave roll was modeled in three dimenions to simulate its location between two cylindrical rollers in a manufacturing process (Figure 6). A view of the actual finite element model showing the arrangement of the triangular elements is presented in Figure 7. In the interest of stability and practicality, only symmetrical roll profiles are used in industry. Therefore, in taking advantage of this symmetry, only one-half of the web and roll are modeled. This also allows higher element densities and improved accuracy. Butler [2] indicated that locations permitting high wrap angles are best for concave rolls; note that the geometry of this model results in a 90 degree wrap angle.

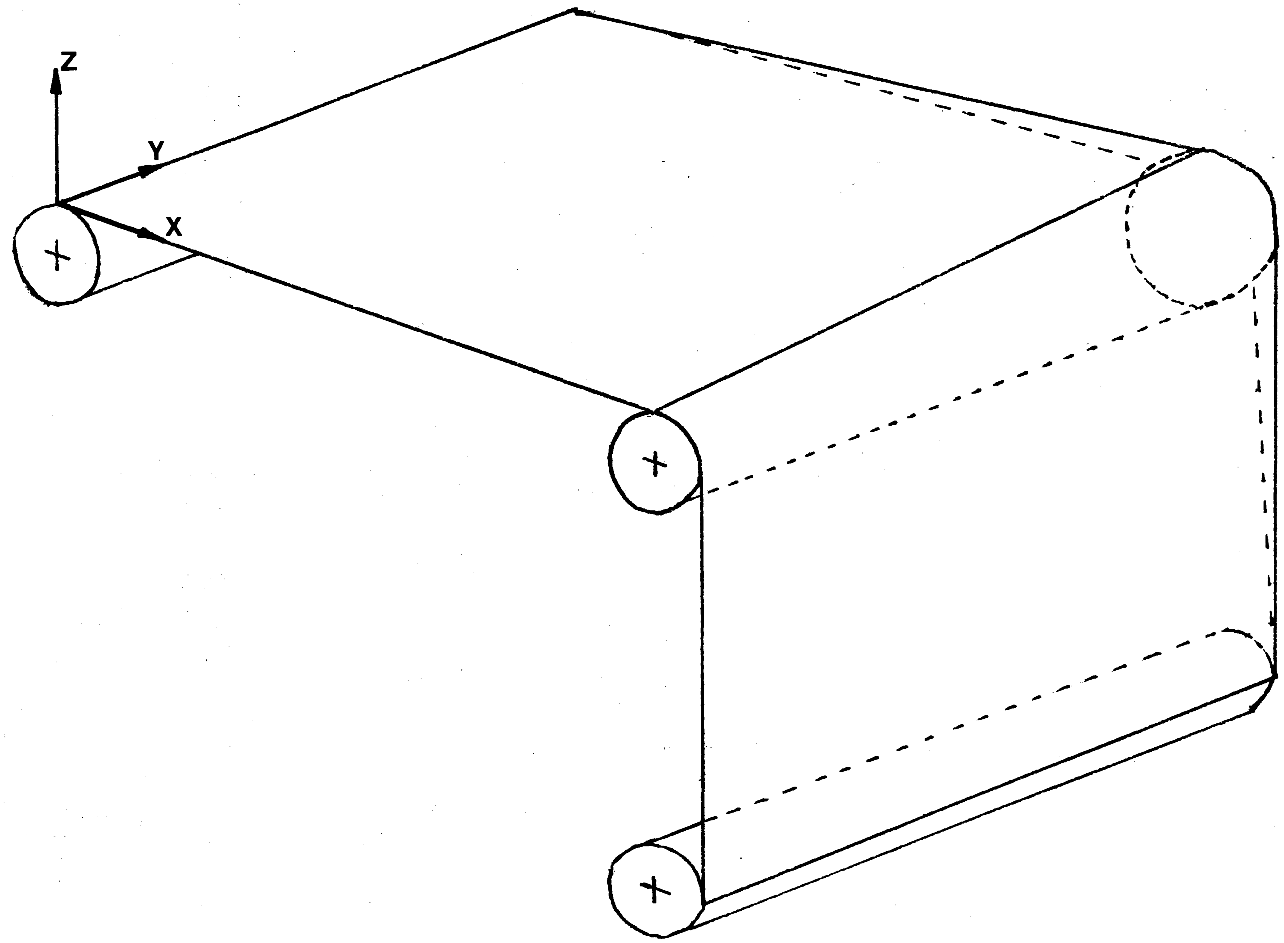


Figure 6. Physical Arrangement of Web and Rolls as Modeled

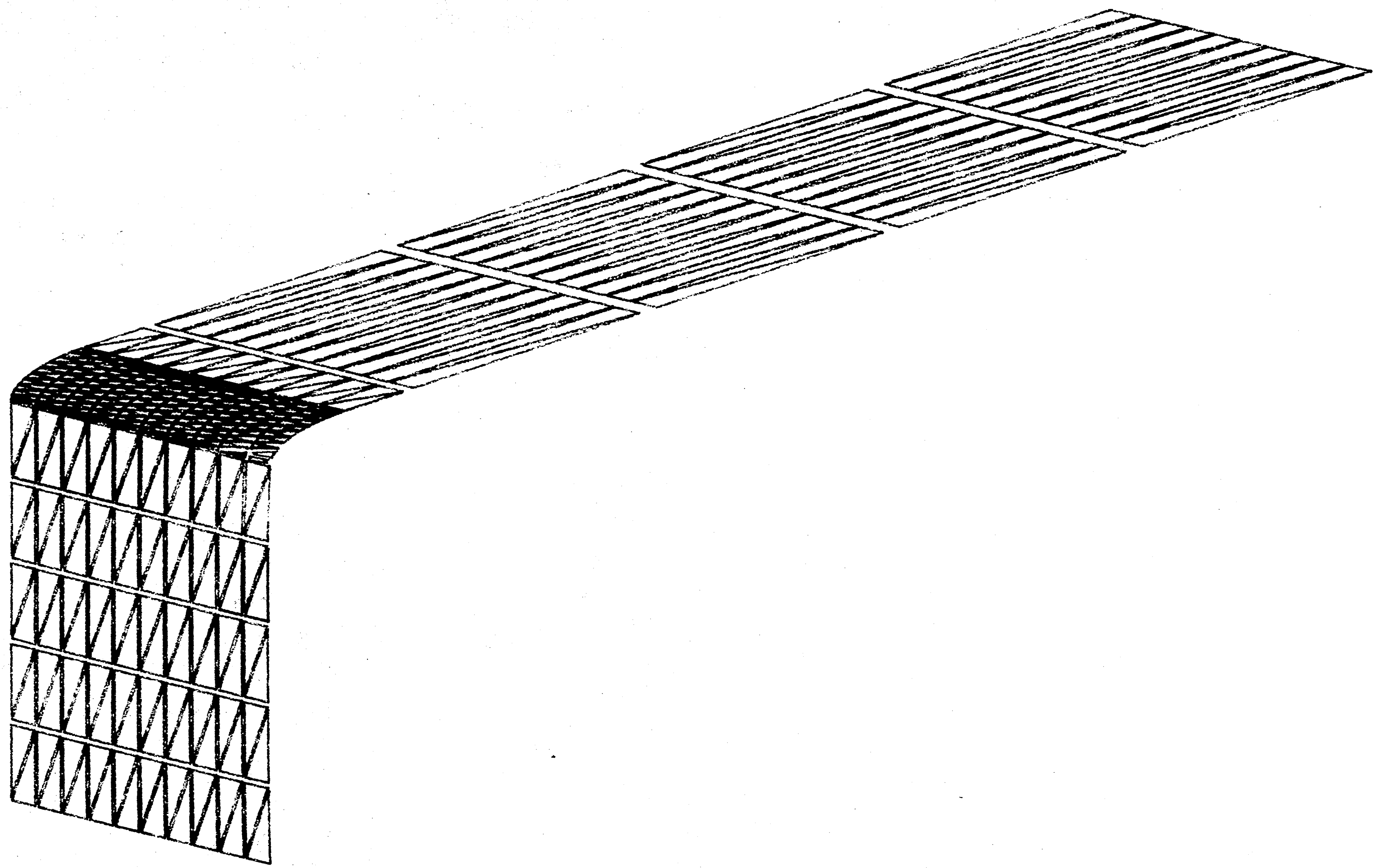


Figure 7. Web Model Using Triangular Elements

It is assumed that the cylindrical rolls are of the same radius as the base (centerline) radius of the concave roll. The model is further assumed to be initially taut; therefore, the outside edge of the web is necessarily higher (global z direction) than the centerline. The height at the outside edge is equal to the difference in radius between the centerline and the outside edge. This has been taken into account in the finite element model by assuming the height of each node increases linearly from the front edge of the model up the roll and decreases linearly from the outside edge to the centerline. A similar assumption holds for the exit span.

If it is assumed that there is no loss or change in traction between the incoming web and the cylindrical roll preceding the concave roll, no relative motion in the lateral or machine direction can exist between the web and roll. Therefore, this end of the model is considered to be completely restrained. It is assumed that no relative motion occurs in the lateral direction on the cylindrical roll following the concave roll, although it is possible to have relative motion in the machine direction due to the machine direction deformations induced in the web by the concave roll. Therefore, the trailing edge of the model is constrained laterally and only machine direction displacements are allowed. Since only half of the web is modeled, it is necessary to laterally constrain the centerline as well. The web is modeled using membrane elements which cannot react out-of-plane loads; thus every node is constrained in the (average) local normal direction.

The most important assumptions governing this study concern the modeling of the portion of the web actually on the roll. The system of average local coordinates discussed in section 2.3 was derived so that the roll surface could be properly modeled. Without the roll surface enforced, the model has no stiffness in space. The average coordinates determine the direction normal to the roll at each node on the roll (Figure 8). Knowledge of this normal direction allows the calculation of the normal forces, and when this direction is constrained the roller surface is enforced. The friction forces are calculated from the normal forces according to the familiar relation

$$F_f = \mu N \tag{3.1}$$

where μ is the coefficient of friction, and N is the normal force. The nodes on the roll are otherwise free to move laterally and in the machine direction on the roll surface under the combined action of line tension and the spreading forces.

The ability of a concave roll to spread the web ultimately depends on the condition of traction between the two. Spreading will occur until the available traction forces are exceeded by the spreading forces and the web slips. The web at this point must be in a constant state of lateral slip. There is no relative web motion in the machine direction. The web may be thought to be "locked" on the roll in the machine direction and any machine direction displacements existing in the incoming web are carried over the roll unchanged. In the finite element model, this phenomenon is simulated by enforcing machine direction displacements at the entrance and exit of the roll.

These enforced displacements can be derived by first considering the elasticity of the web. Shelton [7], in considering the transport of strain over a roll, derives the continuity equation for a web span. The continuity equation states that the change in the unstretched length of

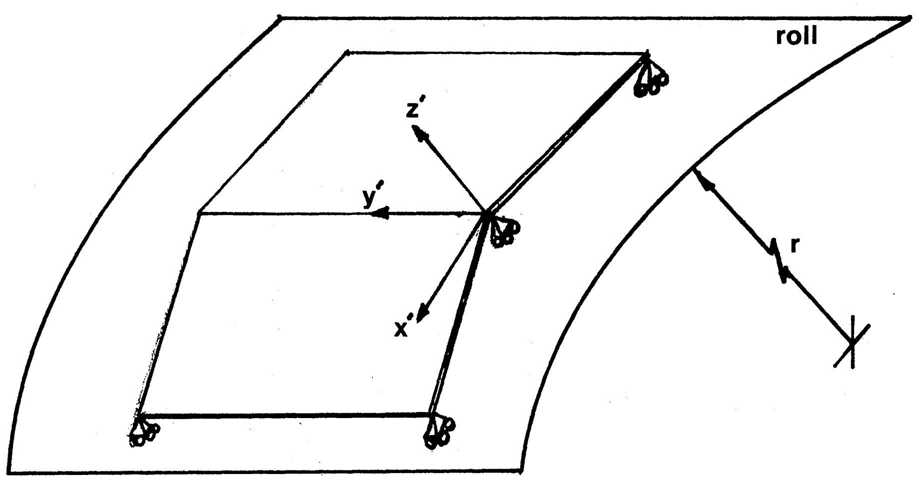


Figure 8. Elements on Roll Surface

the span is equal to the difference between the length of unstretched web entering the span in a given time and the length of unstretched web leaving the span in the same amount of time. Mathematically, the continuity equation can be written as

$$\frac{d}{dt}\left[L_B(1 - \varepsilon_B)\right] - (1 - \varepsilon_A)V_1 + (1 - \varepsilon_B)V_2 = 0 \tag{3.2}$$

where ε_A is the strain in span A, ε_B is the strain in span B, and the other variables are as defined in Figure 9. Assuming steady state conditions, the first term of this equation drops out, leaving

$$-(1 - \varepsilon_A)V_1 + (1 - \varepsilon_B)V_2 = 0 \tag{3.2a}$$

Assuming the cylindrical roll preceding the concave roll both rotate with the same angular velocity ω, the peripheral velocity of each roll may be written as

$$V_1 = r_1\omega_1 = r_1\omega \tag{3.3a}$$

and

$$V_2 = r_2\omega_2 = r_2\omega \tag{3.3b}$$

Equation (3.3b) applies to the concave roll and the radius r_2 is not constant but is a function of the lateral location y. Assuming, as discussed previously, that the base radius r_1 of the concave roll is the same as the radius of the cylindrical roll, the radius r_2 may be written as

$$r_2 = r_1 + (\Delta r)_i \tag{3.4}$$

where $(\Delta r)_i$ is the difference in radius between the cylindrical roll and the concave roll at lateral location i. Equation (3.2a) may now be written as

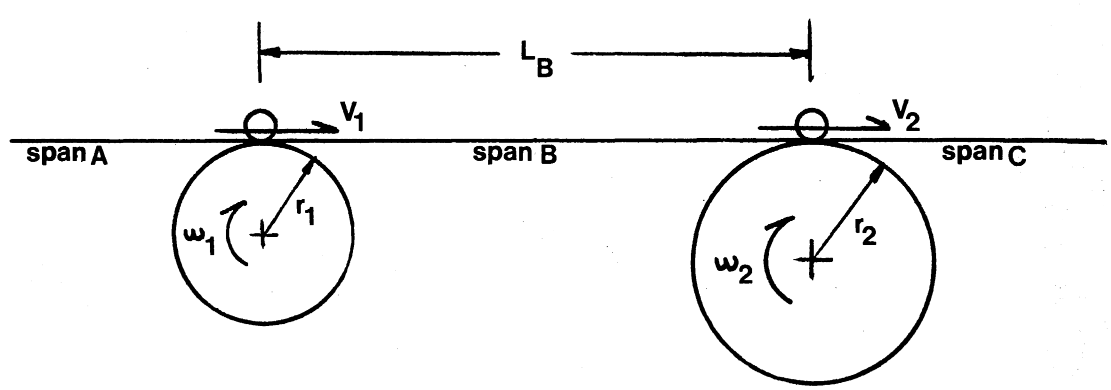


Figure 9. Parameters Used in the Continuity Equation

$$\frac{1 - \varepsilon_A}{1 - \varepsilon_B} = \frac{V_2}{V_1} \tag{3.5a}$$

Substituting for V_1 and V_2 from Equations (3.3a) and (3.3b) gives

$$\frac{1 - \varepsilon_A}{1 - \varepsilon_B} = \frac{(r_1 + (\Delta r)_i)\omega}{r_1\omega} = 1 + \frac{(\Delta r)_i}{r_1} \tag{3.5b}$$

Solving for ε_B yields

$$\varepsilon_B = \frac{\Delta L_B}{L_B} = 1 - \left[\frac{1 - \varepsilon_A}{1 + \frac{(\Delta r)_i}{r_1}}\right] \tag{3.6}$$

The change in length of the entry span due to the concave roll is given by the following relation:

$$(\Delta L_B)_i = L_B\left[1 - \left(\frac{1 - \varepsilon_A}{1 + \frac{(\Delta r)_i}{r_1}}\right)\right] \tag{3.7}$$

where ε_A is calculated from the following relationship:

$$\varepsilon_A = \frac{P}{AE} \tag{3.8}$$

in which P is the product of the average line tension and the width of the web, A is the cross-sectional area of the web, and E is the modulus of elasticity of the web.

At the entrance to the concave roll, the enforced displacements take the form

$$(\mu_1)_i = (\Delta L_B)_i \tag{3.9}$$

where $(\Delta L_B)_i$ is given by Equation (3.7). These displacements are applied to the model at a row of nodes one-quarter inch back from the first row of nodes on the roll to prevent the web from being lifted off the roll, resulting in negative normal forces.

At the exit of the roll, the enforced displacements take the form

$$(\mu_2)_i = (\mu_1)_i + (\Delta L_{BC})_i \tag{3.10}$$

where $(\mu_1)_i$ is the same value calculated using Equation (3.9) at the entrance to the roll, and $(\Delta L_{BC})_i$ represents the change in length of the span BC on the roll and is given by

$$(\Delta L_{BC})_i = L_{BC}\left[1 - \left(\frac{1 - \varepsilon_A}{1 + \frac{(\Delta r)_i}{r_i}}\right)\right] \tag{3.11}$$

in which L_{BC} is taken as the length of the span on the concave roll and is calculated using the base radius of the roll. The nodes at which these displacements are enforced have no lateral constraints.

In terms of the normal forces, it can be shown that for a web on a roll, the normal force is maximum near the midpoint of the contact arc and decreases to zero at the ends of the arc. With no normal force, there is no friction force capability and consequently no spreading force. Therefore, the first and last rows of nodes on the roller are neglected when applying the spreading forces.

It is also assumed that spreading will only occur if, under the action of tensile forces and the enforced displacements, the elements at the entry to the roll are in a condition of positive slope (Figure 10). This condition has as its basis the web strip theory presented in Chapter I. In the positive slope condition, strip theory dictates that the web will move to the outer edge of the roll; this relates to the spreading action in a full web. A web strip in a negative slope condi-

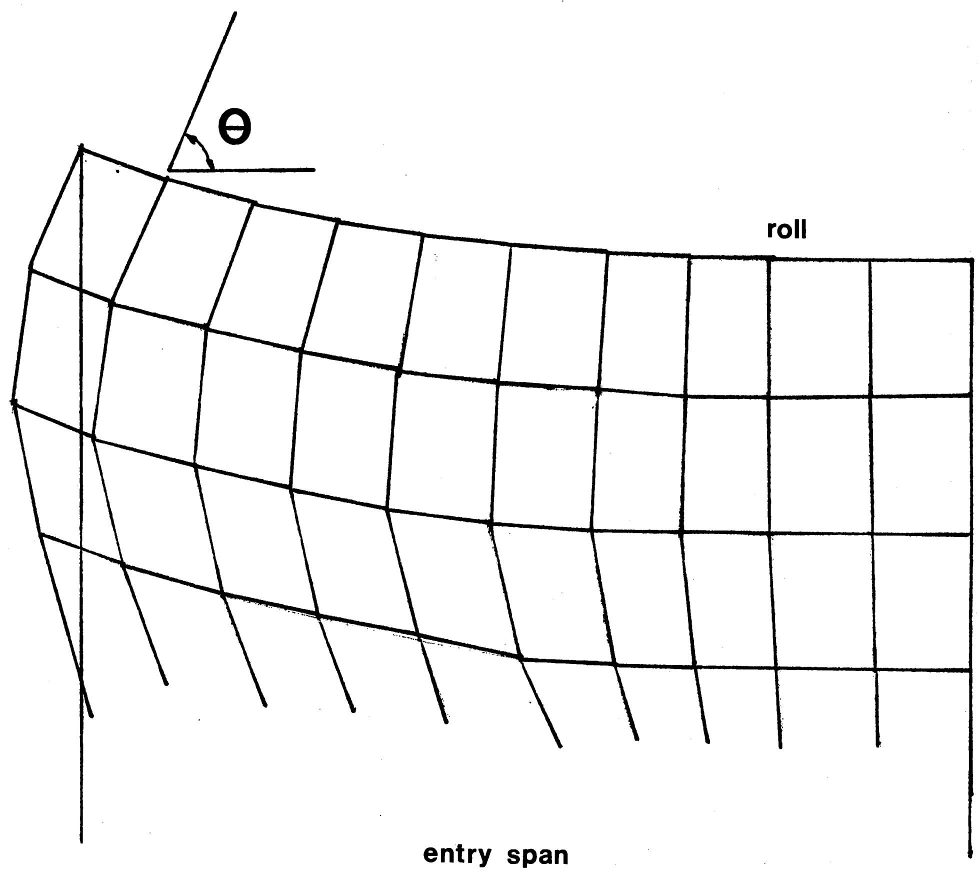


Figure 10. Positive Slope Condition at Entry to Roll

tion will tend to move toward the inside of the roll and spreading will not occur.

3.2 Description of the Computer Code MSHGNR.FOR

The computer code MSHGNR.FOR is a mesh generation program which generates the data needed for the finite element program CONCAVE.FOR (see section 3.3). The data are contained in three data files created by the program. The program allows the user to vary the physical dimensions of the model, including width, length of the entry span, length of the exit span, and base radius of the roll. Within the program, the profile of the roll is represented as a function of lateral location on the roll and may be changed by the user for different profiles. The code also allows the user to independently vary the element density of the entry span, exit span, and the web on the roll.

The file MESH.DAT contains the finite element mesh data. This includes the total number of nodes and elements, the location of each node in global coordinates, and the element connectivities.

The file CONSTR.DAT contains the enforced nodal displacements, both zero and nonzero values. The nonzero displacements are calculated from Equations (3.9) and (3.10). All displacements are in the average local coordinate systems.

The file FORCE.DAT contains the tensile forces placed on the nodes at the trailing edge of the model. These are also referred to the average local coordinate systems.

3.3 Description of the Computer Code CONCAVE.FOR

The computer code CONCAVE.FOR is based on the code STRESS.FOR written by Segerlind [8]. The program is used to analyze a finite element model of a web on a concave roll. The code allows the use of a three-dimensional model composed of two-dimensional triangular elements. To simulate the spreading effect of the concave roll, the program calculates and applies lateral spreading forces to the web on the roll and then checks these forces against the available friction forces which it determines from normal force calculations. Based on the ratio of these two forces, the program either continues to spread the web or it quits and prints out the element stresses and strains, nodal displacements, normal forces, and force ratios. To aid in interpreting the results, the code generates a plot data set for viewing the undeformed and deformed model.

The code is limited to the use of linear triangular plane stress elements. The code does not allow the elements to change their elastic behavior; that is, they are assumed to remain taut. Other possibilities such as temperature effects, body forces, and variable coefficients of friction are not included in this code.

The ability of the program to correctly perform the aforementioned tasks is highly dependent on the code "knowing" exactly where the roll is located on the model. The code must know how many nodes are in the entry span, on the roll, and in the exit span. To properly model the geometry of the roll surface, the correct constraints must be specified. To insure all the necessary date are correctly generated, the use of the computer code MSHGNR.FOR to generate the data is advisable.

Once the code has read the mesh data from the data files, it assembles the global stiffness matrix for the whole model. This matrix is then transformed to the system of average coordinates. As mentioned previously, it is assumed that subsequent deformations in the model will not significantly alter its stiffness and hence the transformed stiffness matrix is not recalculated after each iteration.

At this point in the execution, the only forces which have been applied to the web are the tensile loads at the end of the exit span and the machine direction enforced displacements on the concave roll. The code solves for the nodal displacements and calculates the resulting stresses and strains for this loading condition and then writes these data to the output file.

One of the two criteria for stopping execution of the code is now checked, namely, the condition of the slope of the elements at the entry to the roll (see section 3.1). If the slope of the elements is such that spreading will occur, then execution continues; otherwise, it stops.

The iterative spreading process now begins with the calculation of the normal forces at each node on the roll. These normal forces determine the friction forces as given in Equation (3.1). The amount of spreading force applied to each node is dependent on the friction forces according to the following relation:

$$(F_s)_{i,n} = (F_s)_{i,n-1} + \frac{1}{n}\left[(F_f)_{i,n} - (F_s)_{i,n-1}\right] \tag{3.11}$$

where

F_s = spreading force;

F_f = friction force;

i = node number; and

n = iteration number.

It can be seen that with the spreading force initialized to zero, the full friction force is used to spread the web in the first iteration.

In subsequent iterations, the amount that the spreading force is either increased or decreased from the previous iteration decreases as the inverse of the number of iterations. This is done to provide damping to the iteration process should numerical oscillations occur.

The new displacements and normal forces resulting from spreading the web are now calculated. The friction forces calculated from the new normal forces represent the friction force capability of the roll at this condition of spreading. The code calculates the ratio of the spreading force to the friction force at each node on the roll and checks to see if it is within the specified limits. For this study, limits of 0.98 to 1.02 were specified. This amounts to an allowable error of plus or minus two percent. If the ratio is greater than 1.02, the spreading force has exceeded the friction force and the node has slipped. Ratios less than 0.98 indicate that a particular node can withstand a higher spreading force. If any node on the roll is outside the range, then the output file is rewritten with the new data, and the iteration continues until the ratios for all the nodes are within the limits. At this point the web is in the constant state of lateral slip mentioned previously and no more spreading is possible. This is the second criterion for stopping execution. The logic of the code just described is shown in flowchart form in Figure 11.

A brief description of each subroutine called by CONCAVE.FOR is now presented.

Subroutine INPUTS reads in the mesh data from the data files. Diagnostic checks are included to catch certain, more common errors in the mesh data.

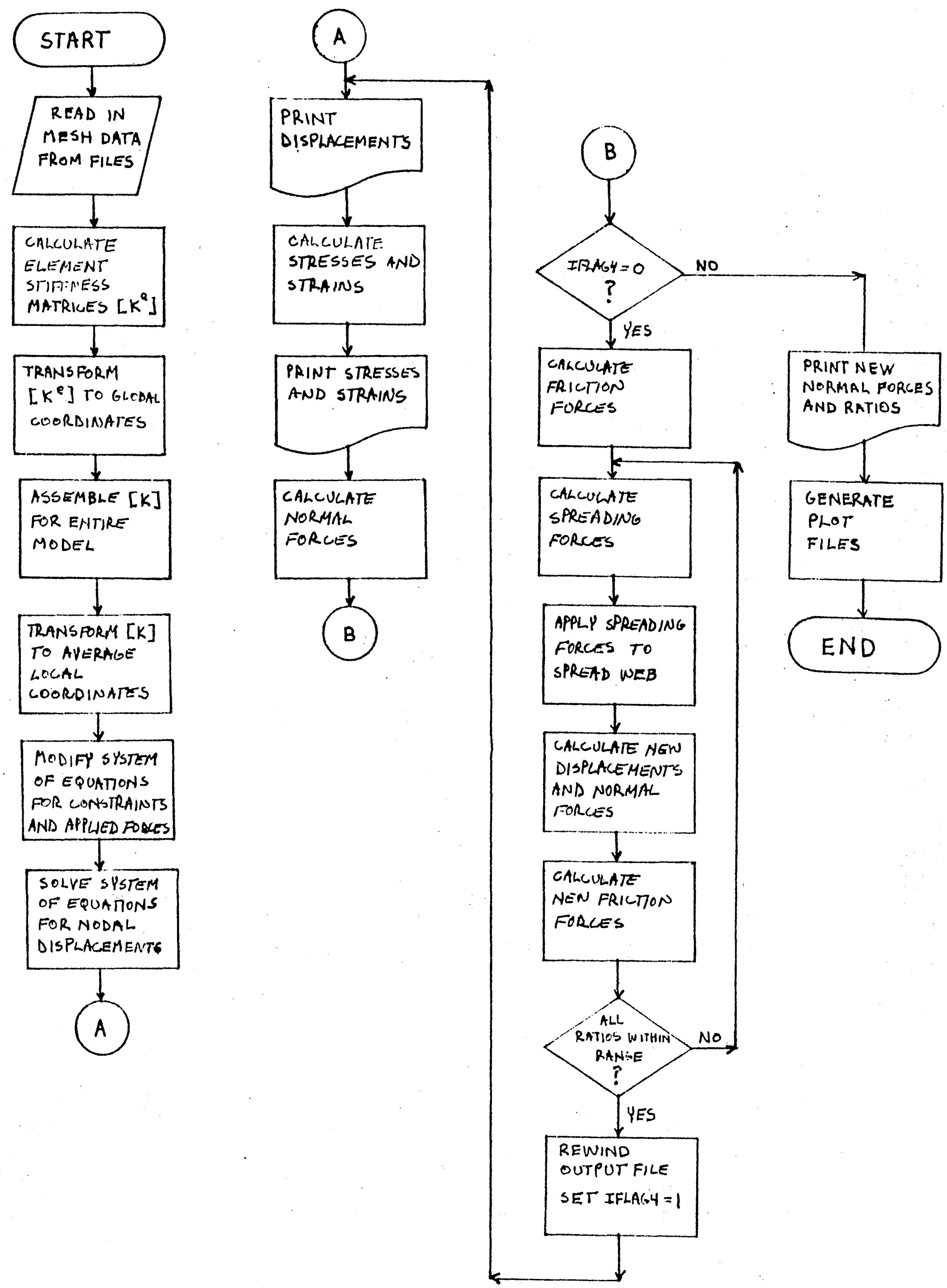


Figure 11. Flowchart for Computer Code CONCAVE.FOR

Subroutine ELSTMX calculates the element stiffness matrix $[K^{(e)}]$ using Equation (2.7). This subroutine also evaluates the gradient matrix [B] in the loop in which the element strains are calculated.

Subroutine LAMDA evaluates the transformation matrix $[\lambda]$ of Equation (2.10) using the method outlined in section 2.2. This subroutine also calculates the inverse cosine of the direction cosines in $[\lambda]$ for use in the coordinate averaging scheme in subroutine SKEWED.

Subroutine TRNSMX evaluates the transformation matrix [T] of Equation (2.16b) and its transpose. Subroutine TRNSFM performs the transformation of the local element stiffness matrix to the global coordinates using Equation (2.16a).

Subroutine ASSEMB uses the direct stiffness method (see Reference [8]) to assemble the global stiffness matrix for the entire model.

Subroutine SKEWED performs the transformation of the global stiffness matrix to the average local coordinates. The subroutine determines the location of the node and based on its location calculates the average nodal transformation matrix $[\bar{\lambda}_n]$.

Subroutine MODIFY reads in the enforced nodal displacements and the nodal forces applied at the end of the exit span. The subroutine then modifies the system of equations to account for these known values using the method of deletion of rows and columns described in Appendix III of Reference [8].

Subroutine SANDS calculates the strain vector and stress vector using Equations (2.4) and (2.5). The subroutine also computes the principle stresses and the principle stress angle.

Subroutines FACTOR and SUBST transform the system of equations to upper triangular form (FACTOR) and back substitute (SUBST) to find the solution the the system of equations. These subroutines are taken from Conte and deBoor [9].

Subroutine CHKSLP determines the condition of the slopes of the elements at the entry to the roll as detailed in section 3.1.

Subroutine FRICTN calculates the friction forces for a particular node using Equation (3.1). Subroutine APFORS then uses Equation (3.11) to calculate the spreading force for each node. The subroutine then "applies" the force in the lateral direction by placing it in the force vector {F}.

Subroutine CHKFOR calculates the ratio of the spreading force to the friction force at each node on the roll. It then checks to see if the ratio is within the specified range.

Subroutine PLOt generates the plot data used by the computer code PLOT.FOR (see Appendix). For ease of viewing the model is divided into three sections: before, on, and after the roll. The subroutine writes the plot data to three data files: BEFORE.DAT, ON.DAT, and AFTER.DAT. The undeformed and deformed mesh for each of these three sections of the model may be viewed using PLOT.FOR.

3.4 Summary

In this chapter a description of the finite element model used in this study was presented. The assumptions made in arriving at the final model and those assumptions governing its use are also presented. In section 3.2 a description of the mesh generation computer code MSHGNR.FOR was presented. A description of the logic and use of the finite element computer code CONCAVE.FOR used to analyze the model was presented in section 3.3. Chapter IV in this study presents the results of this study and provides some discussion as to the implications of these results.

CHAPTER IV

ANALYTICAL RESULTS

As stated previously, one objective of this study is to investigate the effects of spreading on the entry and exit spans, and upon the roll for circular arc and linear taper profiles. This chapter presents the results of this investigation. Descriptions of the normal force distributions, lateral displacements, and induced stresses with accompanying data tables and plots are presented.

4.1 Model Parameter Values

To permit future experimental verification of the results of this study on an existing eight-inch web machine, a model (of one-half of the web) four inches wide was chosen. Recall that the finite element computer code written for this study analyzes one-half of the web. To produce lead-in and lead-out distances suitable for investigation of the upstream and downstream effects of the roll, entry and exit span lengths of 16 inches were chosen.

Element density is important in achieving numerical accuracy. High element density also results in increased computer execution times and increased storage requirements. In considering these facts, a model consisting of 231 nodes and 400 elements was chosen. This allows the placement of 11 nodes across the width of the web, giving element widths of 0.4 inches throughout the model. The accuracy of the results depends upon the accuracy with which the roll is modeled; therefore, one-half of the total number of nodes and elements is placed on the roll itself.

Web properties typical of those encountered in industry were chosen for this study. The values chosen are a 220,000 psi modulus of elasticity, 0.002 inch thickness, 0.30 Poisson's ratio, and a 0.30 coefficient of friction between the web and roll. A tensile load of 6.25 lb/in. is applied at the trailing edge of the model. A base radius of 1.125 in. is used for each profile studied. These model parameters and web properties are used throughout this study.

The two roll profiles under study are the circular arc and linear taper. The profiles of these two rolls can be written algebraically as

$$R(y) = -\sqrt{R^2 - y^2} + R + R_b \tag{4.1a}$$

for the circular arc profile, and

$$R(y) = \left(\frac{h_o}{w}\right) y + R_b \tag{4.1b}$$

for the linear taper profile, where the variables are as defined in Figures 12 and 13. A numerical comparison of the profile heights of the rolls used in this study are presented in Table I.

4.2 Normal Force Distribution

At the condition of maximum spreading (when spreading ceases), the spreading forces must be equal to the available friction forces. Examination of the friction force relationship (Equation (3.1)) reveals that friction force is directly proportional to the normal force. Assuming the coefficient of friction is constant everywhere on the roll (as in this study), then the friction force and normal force distributions will

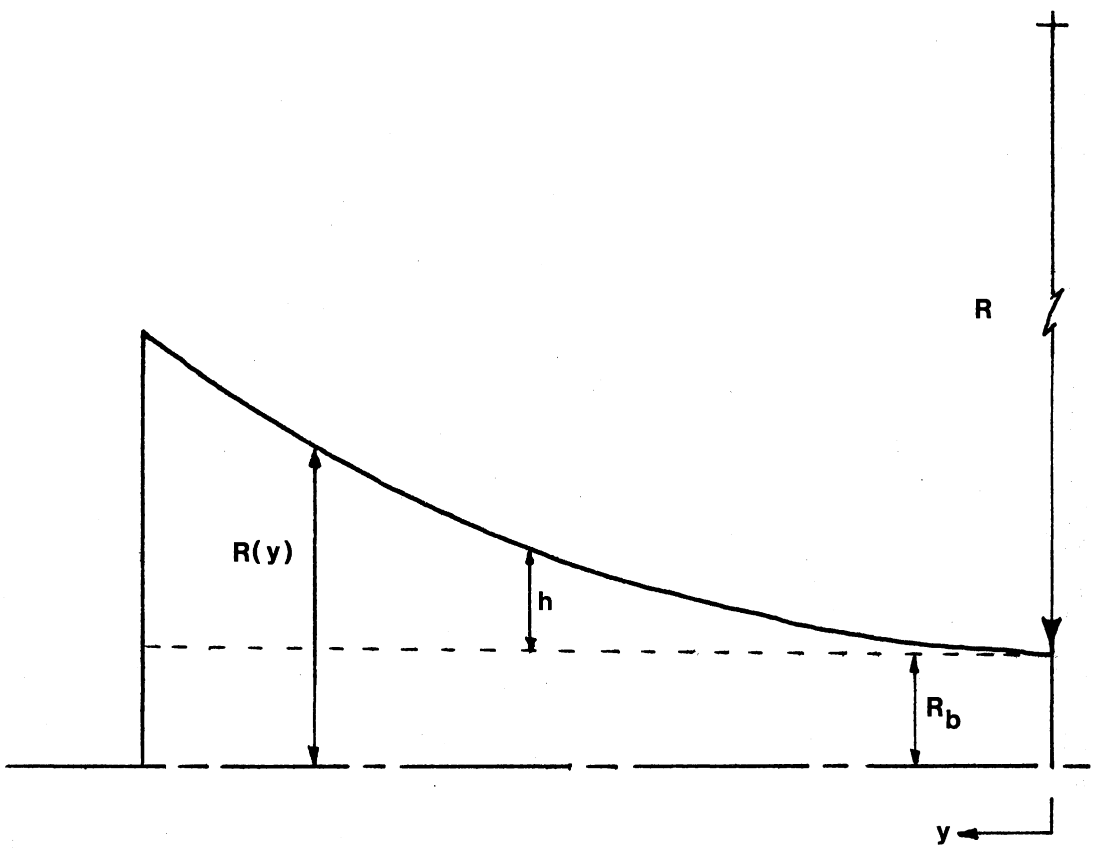


Figure 12. Definition of Circular Arc Profile Parameters

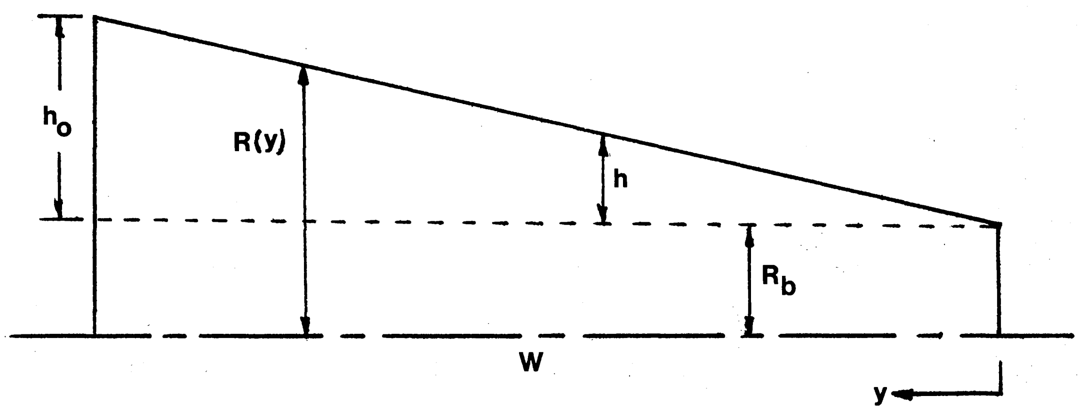


Figure 13. Definition of Linear Taper Profile Parameters

TABLE I

PROFILE HEIGHTS OF ROLLS UNDER STUDY

Ψ (In.)	Profile Height h_o — Circular Arc R = 750 in.	3500 in.	25000 in.	Linear Taper $h_o = 0.00229$ in.
0.0	0.00000	0.00000	0.00000	0.00000
0.4	0.00011	0.00002	0.00000	0.00023
0.8	0.00043	0.00009	0.00001	0.00046
1.2	0.00096	0.00021	0.00003	0.00069
1.6	0.00171	0.00027	0.00005	0.00092
2.0	0.00267	0.00057	0.00008	0.00115
2.4	0.00384	0.00082	0.00012	0.00137
2.8	0.00523	0.00112	0.00016	0.00160
3.2	0.00683	0.00146	0.00020	0.00183
3.6	0.00864	0.00185	0.00026	0.00206
4.0	0.00864	0.00229	0.00032	0.00229

be identical and the magnitudes of the forces will differ by a constant factor equal to μ. Therefore, determining the normal force distribution determines the distribution of spreading forces as well.

The normal force distributions for the circular arc 3500 (radius of curvature of 3500 in.) and the linear taper profiles are presented in Figures 14 and 15, respectively. In both figures, the axis labeled "Y Location" refers to the y-axis location where 0.00 is the centerline and 4.00 is the outside edge. The axis labeled "Theta" refers to the angular location on the roll where 9 represents the entry and 81 represents the exit of the roll.

While the two distributions are almost identical in form, the linear taper profile, on average, generates the larger normal forces due to its larger diameter (see Table I). The most unusual feature of the two distributions is the large normal force present on the outer edge at the exit of the roll. At first glance this value seems out of place, yet examination of the values for the outside edge shows a sharp increase in normal force toward the exit of the roll and this large "spike" fits the trend. This phenomenon may be explained using the web strip theory of Chapter I. Recall that the helical path that a web (strip) takes around the roll tends toward the outside edge of the roll exit. This implies that the largest spreading should occur at this location and, therefore, the largest normal force should be present there as well.

The sharp decrease at the centerline of the web is due to the model boundary conditions at this location. Since only one-half of the web is modeled, the nodes at the centerline see only the normal force contribution of half of the web. If these values were doubled, they would be consistent with the distribution.

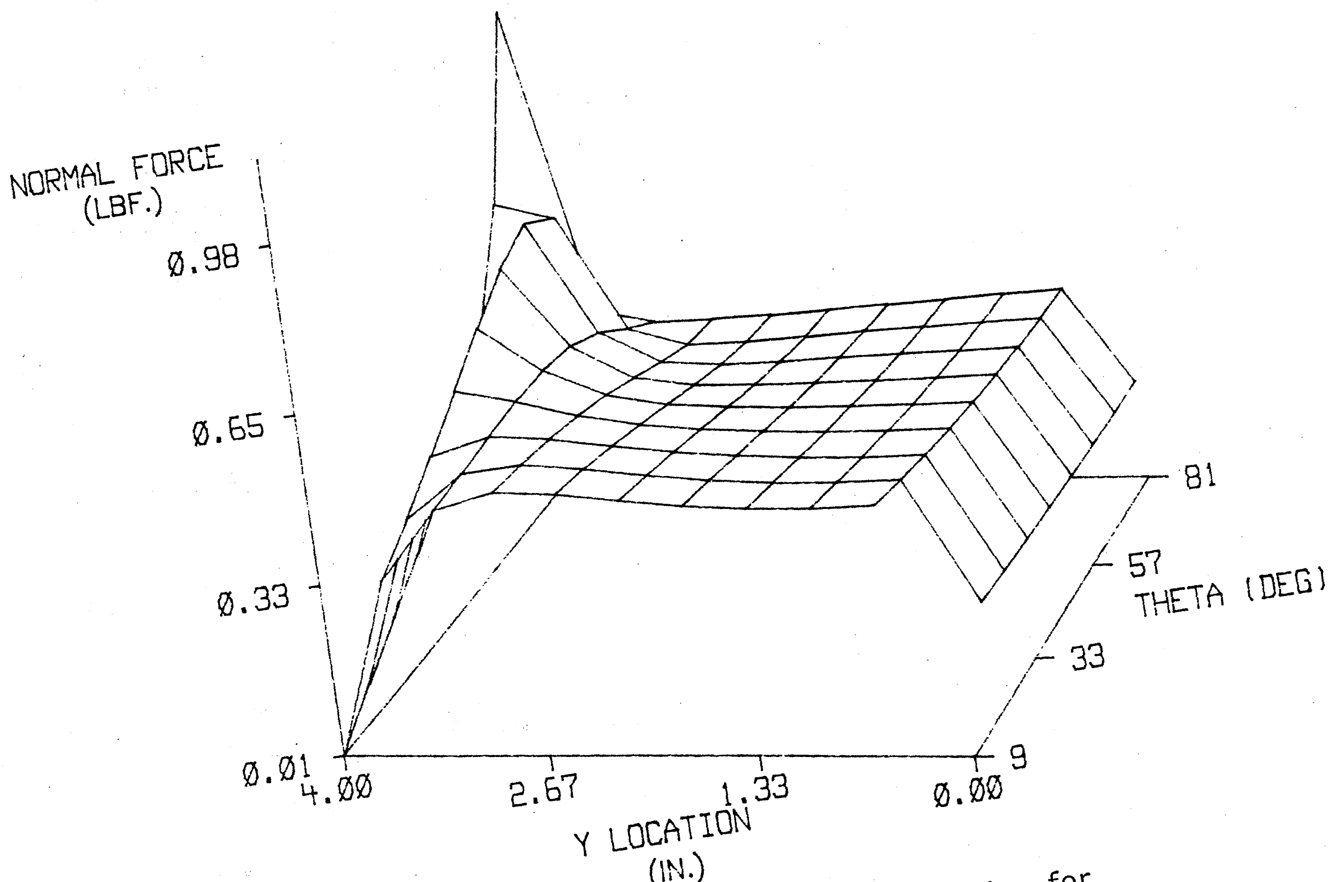


Figure 14. Normal Force Distribution for Circular Arc 3500 Profile

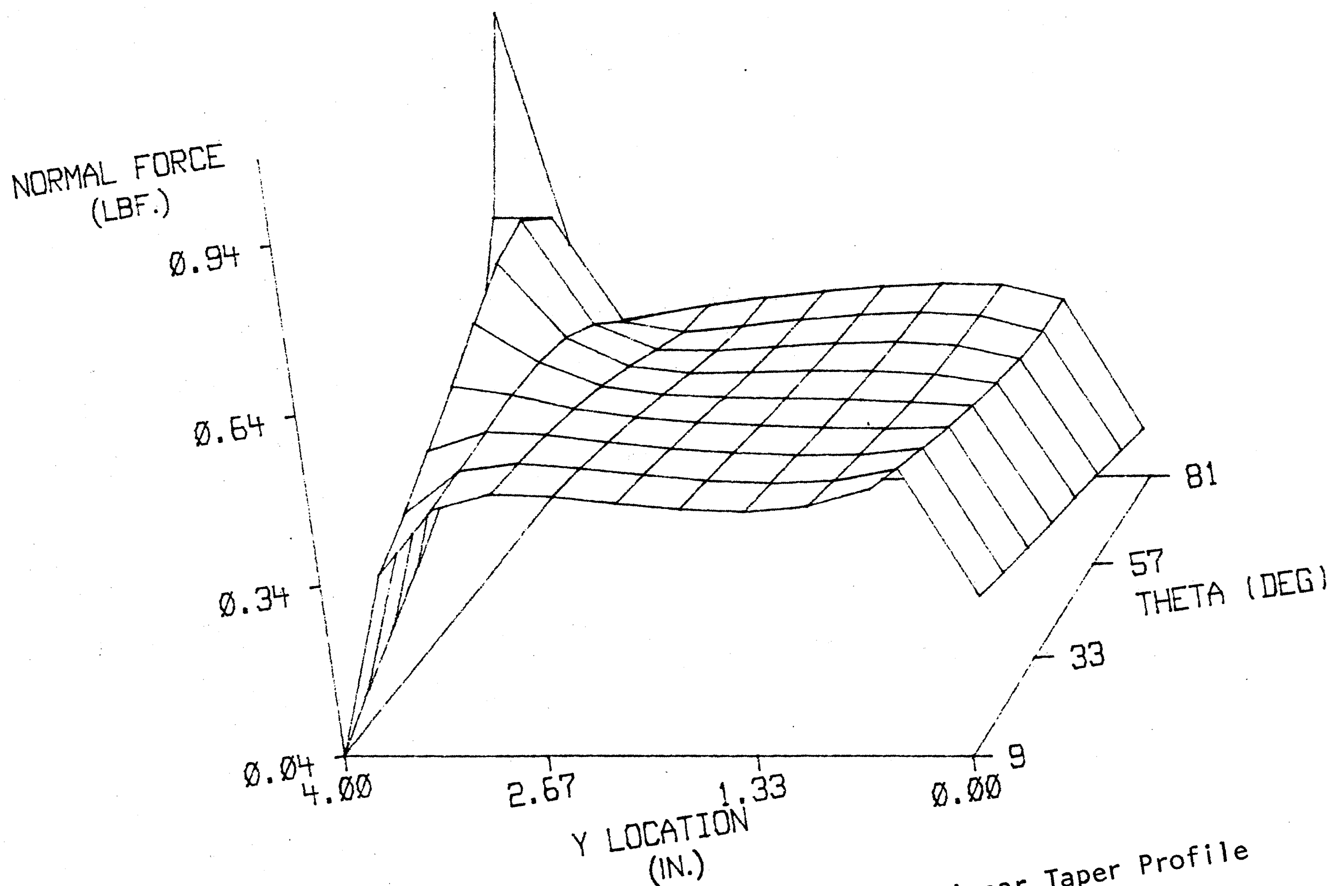


Figure 15. Normal Force Distribution for Linear Taper Profile

Tracing a path on the distribution at a constant angular value from the centerline outward reveals a slight decrease in force followed by an increase to a maximum value occurring one row of nodes (0.4 in.) inboard from the outside edge (except for the first and last rows). This increase is consistent with the increasing radius toward the outside edge and the preceding decrease can be attributed to the spreading forces trying to lift the web off the roll. The decrease in normal force from the maximum to the value at the outside edge is similar to the effect at the centerline. If smaller elements were used, the normal forces would continue to increase toward the outer edge before finally dropping off at the edge itself.

The data presented in these two distributions represent the normal forces at the condition of maximum spreading. This condition is a function of the friction force capability of the roll which depends on the normal force distribution which is in turn dependent on the roll profile and web properties. Therefore, curve fitting the above data would serve no useful purpose and consequently was not done in this study.

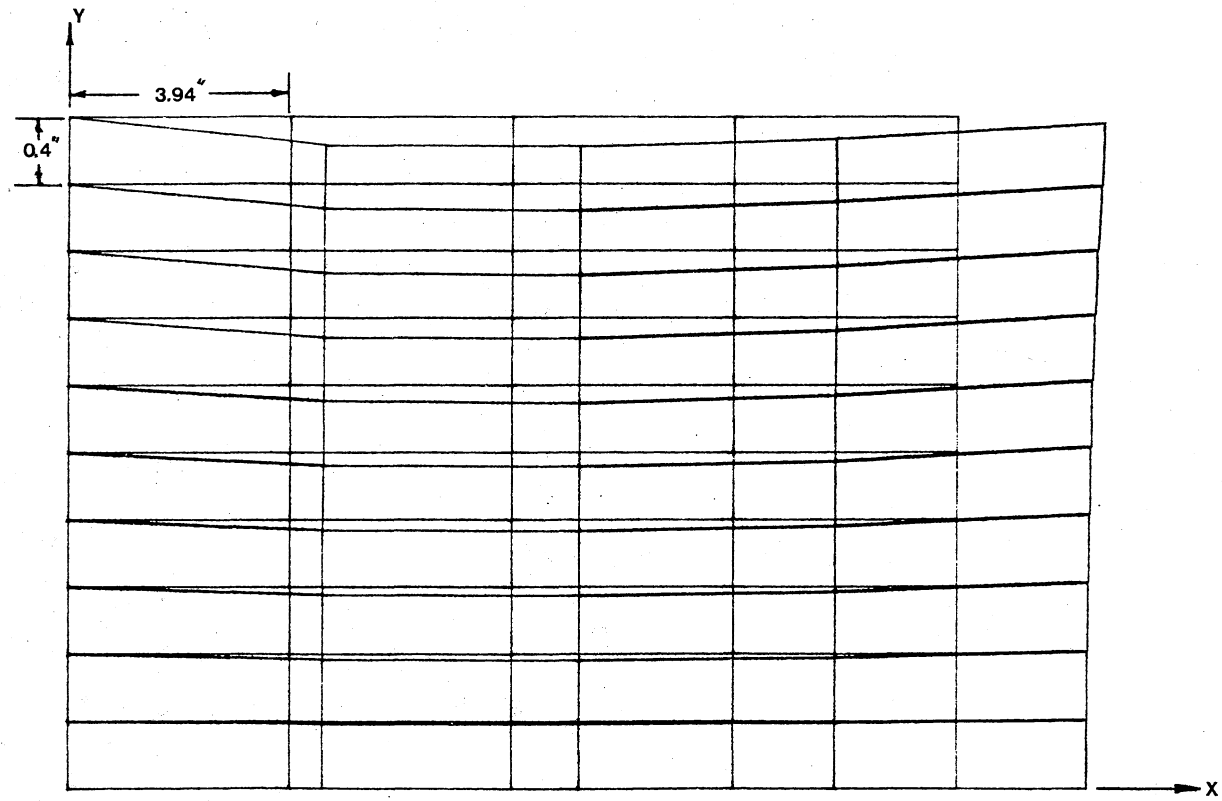
4.3 Lateral Displacements

One measure of a roll's spreading capability and characteristics are the lateral displacements of the web produced by the roll. Examination of the lateral displacement of the outside edge of the web shows the magnitude and distribution of the spreading effect for a given roll profile. All results presented in this section are for the web at its maximum spreading condition.

To provide a basis for comparison, a cylindrical roll of the same radius as the base radius of the concave roll is studied. All of the model parameters are as discussed in section 4.1. In a thin body, such as a web, a two-dimensional state of strain exists. Therefore, the elongation due to a tensile load applied along one of the in-plane coordinate directions is accompanied by a contraction along the coordinate direction normal to the applied load and is known as the Poisson effect. The cylindrical roll has no spreading effect on the web and the resulting lateral displacements (contraction) of the outside edge of the web are due to the Poisson effect.

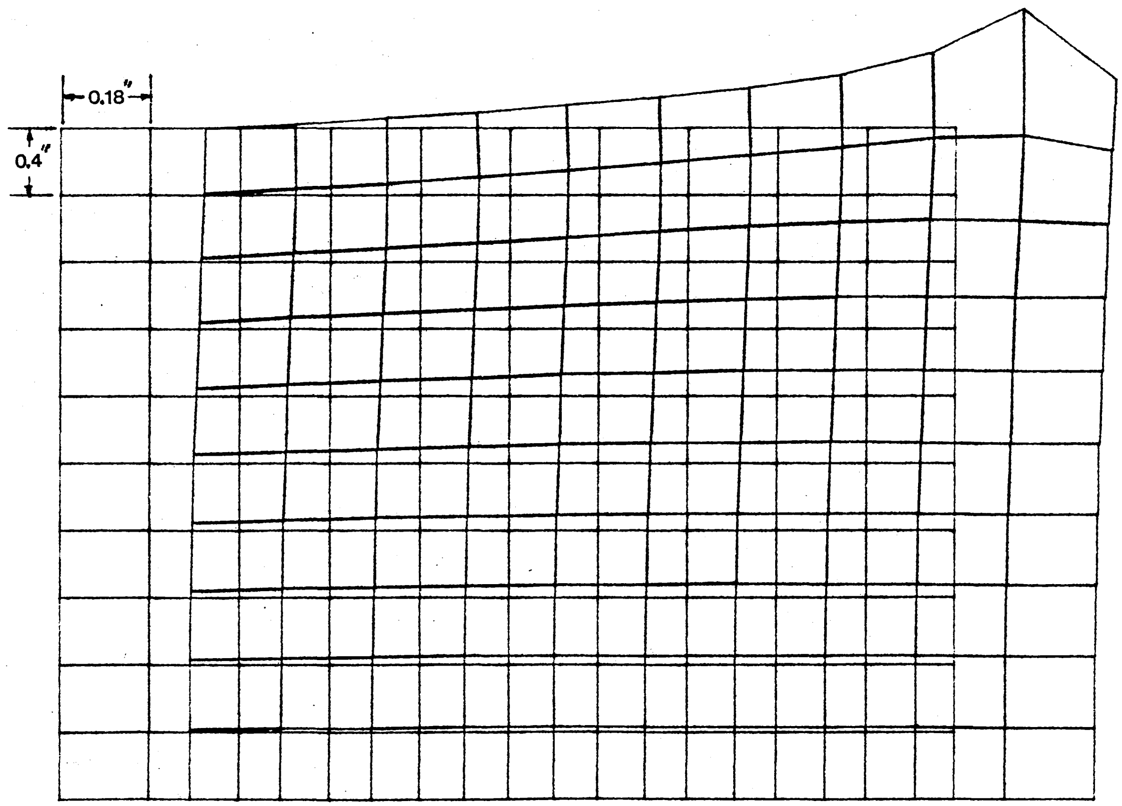
The first profile studied is the circular arc 3500. The behavior of this roll is typical of the circular arc profiles and the spreading effect is evident in Figure 16a, 16b, and 16c. This figure shows the deformed model superimposed on the undeformed model. To facilitate viewing the entire model, the model is divided into three sections: entry span, roll, and exit span. To make viewing easier, the nodes are connected to form rectangular elements instead of the triangles actually used in the study. To view the displacements of the nodes on the roll (Figure 16b), the mesh on the roll is opened up to form a flat plane. These plots are presented to aid in the qualitative understanding of the spreading effect and all deformations are shown ten times their actual value.

The lateral displacements of the outside edge of the web for the circular arc 3500 profile take the form shown in Figure 17. Examination of the plot reveals that the necking (lateral contraction) of the web in the entry span is more than that produced by the Poisson effect alone, due to the bending moment induced in the span by the roll profile. This effect is overcome by the spreading forces and the web is nearly spread back to its original taut condition at the entry to the roll, as evidenced by the near zero lateral displacement at the roll. The most signifi-



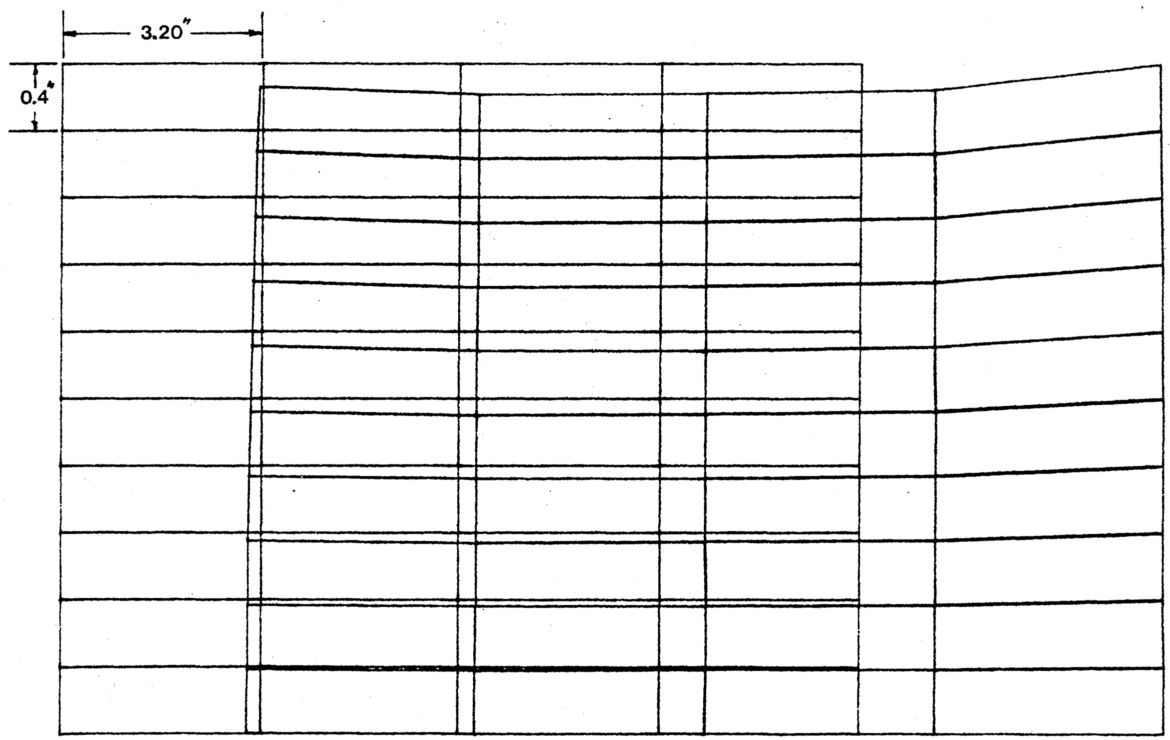
(a) Deformation of the Entry Span

Figure 16. Deformations for Circular Arc 3500 Profile



(b) Deformations on the Roll

Figure 16. (Continued)



(c) Deformation of the Exit Span

Figure 16. (Continued)

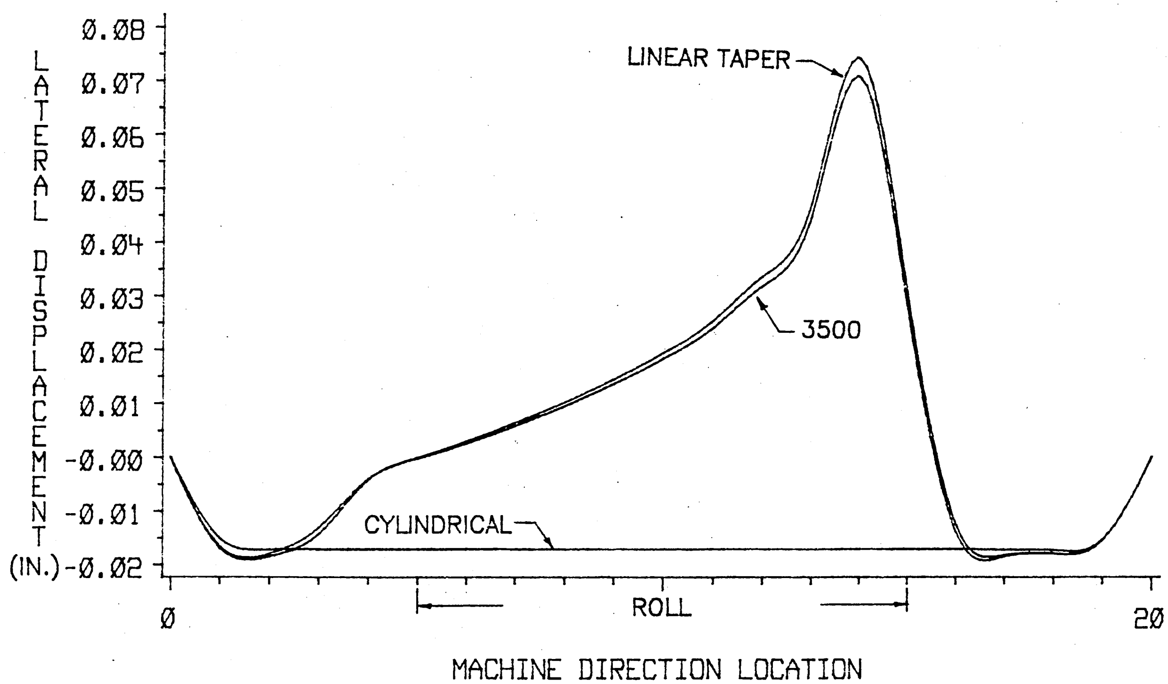


Figure 17. Lateral Displacement of Outside Edge of Web for Circular Arc 3500 and Linear Taper Profiles

cant spreading occurs while the web is on the roll with the maximum lateral displacement just prior to the exit of the roll. Recall that the largest normal force is present at this location (see Figure 14). This spreading is lost in the exit span as the web loses the effects of the spreading forces. As in the entry span, the lateral contraction is greater than the Poisson effect and all the spreading is lost by the time the web reaches one-quarter of the length of the exit span. The web is brought back to zero lateral displacement at the cylindrical roll due to the lateral constraints imposed at this roll. Figure 17 also presents the lateral displacements for the linear tapered roll (with the same base and outside radius as the circular arc 3500). The two curves are almost identical in form with the linear taper profile producing approximately 5 percent greater lateral displacement (Table II).

Changing the radius of curvature of the circular arc profile to 750 inches and then to 25000 inches produces the results shown in Figure 18 and Table III. The circular arc 750 and 25000 curves of Figure 18 show that decreasing the radius of curvature (increasing the profile height) results in a decrease in spreading; and increasing the radius of curvature (decreasing profile height) increases the spreading effect. At increasing profile heights, a larger component of the spreading force is directed at lifting the web off the roll than is directed at spreading the web. The reverse is true for smaller profile heights where almost all of the spreading force is directed at spreading the web. Examination of the 25000 curve of Figure 18 also shows that besides increasing the spreading, the extent to which the spreading is retained in the exit span is also increased. At some point, increasing the radius of curvature must result in a decrease in spreading action as the profile height

TABLE II

LATERAL DISPLACEMENTS OF OUTSIDE EDGE FOR CIRCULAR ARC 3500 AND LINEAR TAPER PROFILES

Machine Direction Location	Lateral Displacement (In.)	
	3500	Linear Taper
0	0.00000	0.00000
1	-0.01655	-0.01690
2	-0.01789	-0.01840
3	-0.01302	-0.01423
4	-0.00398	-0.00415
5	-0.00013	0.00000
6	0.00263	0.00303
7	0.00587	0.00644
8	0.00952	0.01023
9	0.01359	0.01445
10	0.01828	0.01933
11	0.02400	0.02533
12	0.03181	0.03356
13	0.04452	0.04691
14	0.07068	0.07412
15	0.02822	0.03094
16	-0.01382	-0.01245
17	-0.01843	-0.01818
18	-0.01794	-0.01794
19	-0.01547	-0.01548
20	0.00000	0.00000

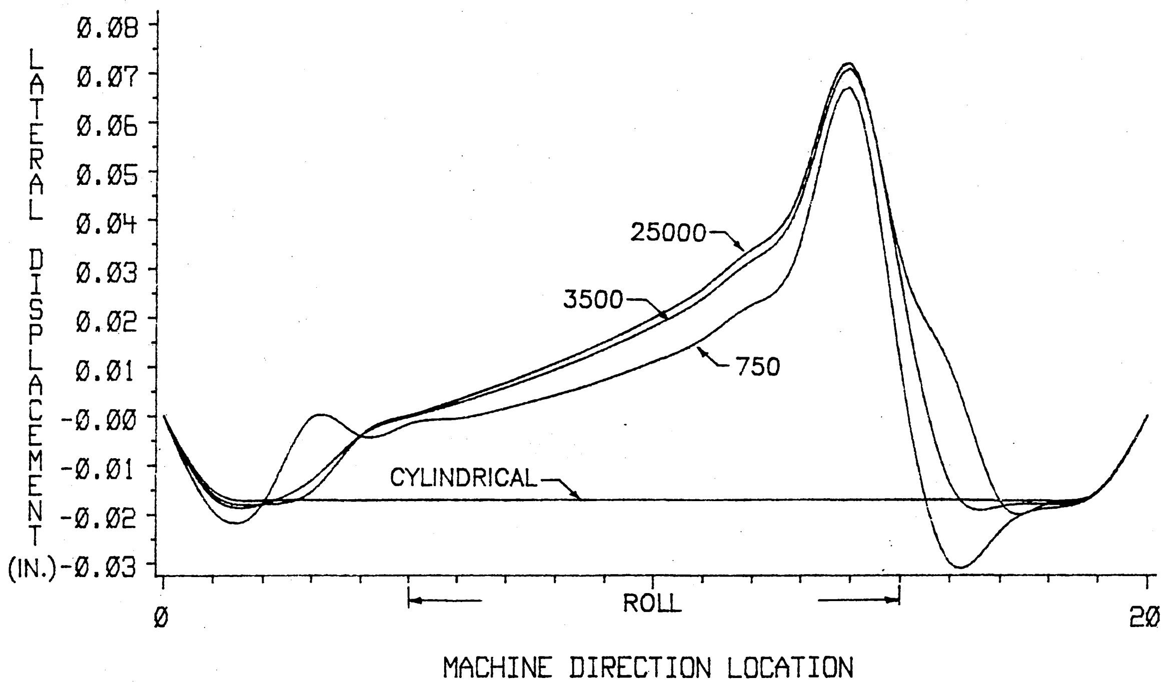


Figure 18. Lateral Displacement of Outside Edge of Web for Circular Arc Profiles

TABLE III

LATERAL DISPLACEMENTS OF OUTSIDE EDGE
FOR CIRCULAR ARC PROFILES

Machine Direction Location	Lateral Displacement (In.) Radius of Curvature (In.) 750	3500	25000
0	0.00000	0.00000	0.00000
1	-0.01922	-0.01655	-0.01592
2	-0.01785	-0.01789	-0.01791
3	-0.00309	-0.01303	-0.01539
4	-0.00404	-0.00398	-0.00399
5	-0.00157	-0.00013	0.00019
6	-0.00041	0.00263	0.00335
7	0.00163	0.00587	0.00688
8	0.00425	0.00952	0.01078
9	0.00736	0.01359	0.01509
10	0.01103	0.01828	0.02003
11	0.01572	0.02400	0.02601
12	0.02271	0.03181	0.03402
13	0.03577	0.04452	0.04667
14	0.06665	0.07068	0.07177
15	0.01065	0.02822	0.03248
16	-0.02988	-0.01382	0.00999
17	-0.02359	-0.01843	-0.01729
18	-0.01873	-0.01795	-0.01776
19	-0.01545	-0.01547	-0.01547
20	0.00000	0.00000	0.00000

decreases to such a value that smaller spreading forces are produced. This continues until the roll becomes cylindrical and no more spreading forces are produced. From the data of Figure 18 (considering only peak lateral displacements) and assuming the cylindrical roll to have a circular arc profile with an infinite radius of curvature, the curve presented in Figure 19 may be inferred. For plotting purposes the radius of curvature which resulted in a profile height of 0.0001 inch at the outer edge of the roll was taken to be infinity, since a profile of this magnitude, in terms of machining, is essentially flat.

4.4 Lateral and Machine Direction Stresses

The final portion of the study investigates the lateral and machine direction stresses induced in the web by the various roll profiles. The lateral stresses are another measure of a roll's spreading effect. Of most concern is the area of highest stresses which, for all the profiles studied, occurs at the exit of the roll where the greatest amount of spreading is present. This portion of the study will concentrate on this area of the model. In this section machine direction stress refers to the first principal stress σ_1 and lateral stress refers to the second principle stress σ_2. For the simplex triangle elements used to model the web, the stress is constant throughout an element; therefore, all stresses are plotted at the centroid of the element.

The first profile studied is the circular arc 3500. The variation in lateral stress across the width of the web is shown in Figure 20, where lateral location "0" is the centerline of the web. The machine direction stresses are presented in Figure 21. The numerical data are presented in Tables IV and V. Subsequent plots will show that the form

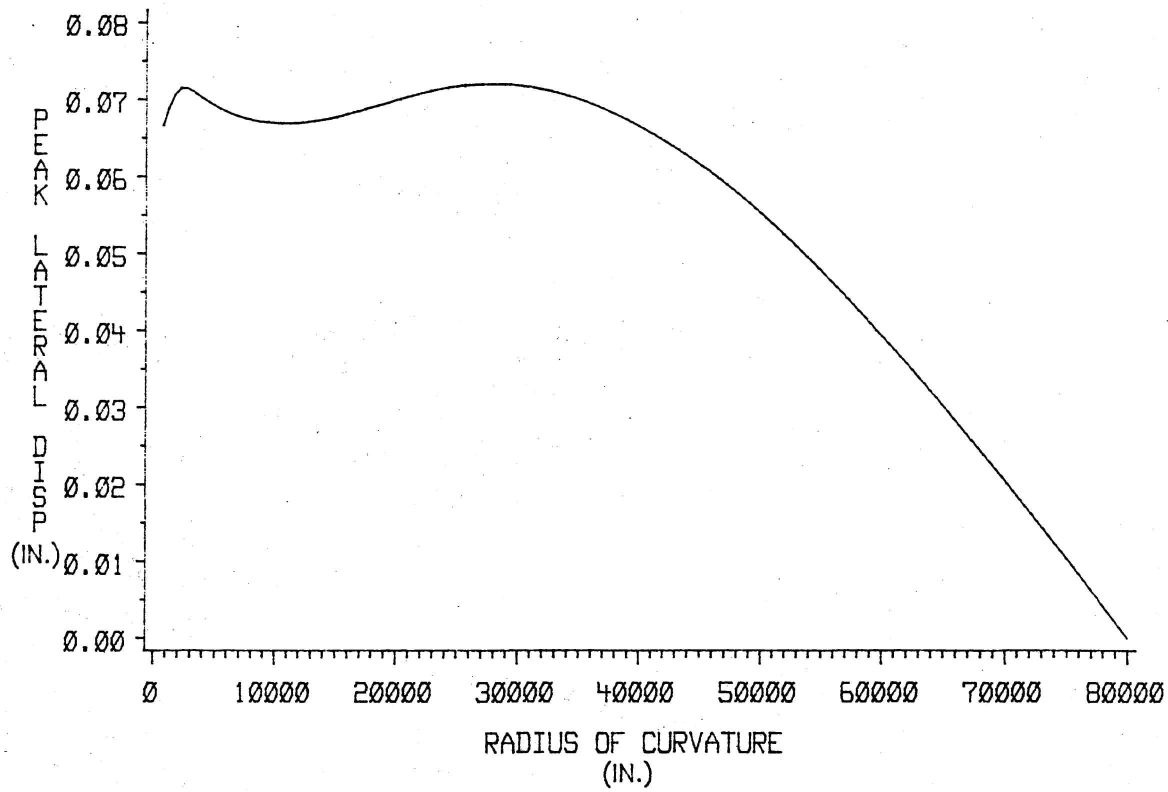


Figure 19. Peak Lateral Displacement of Outside Edge of Web for Circular Arc Profiles

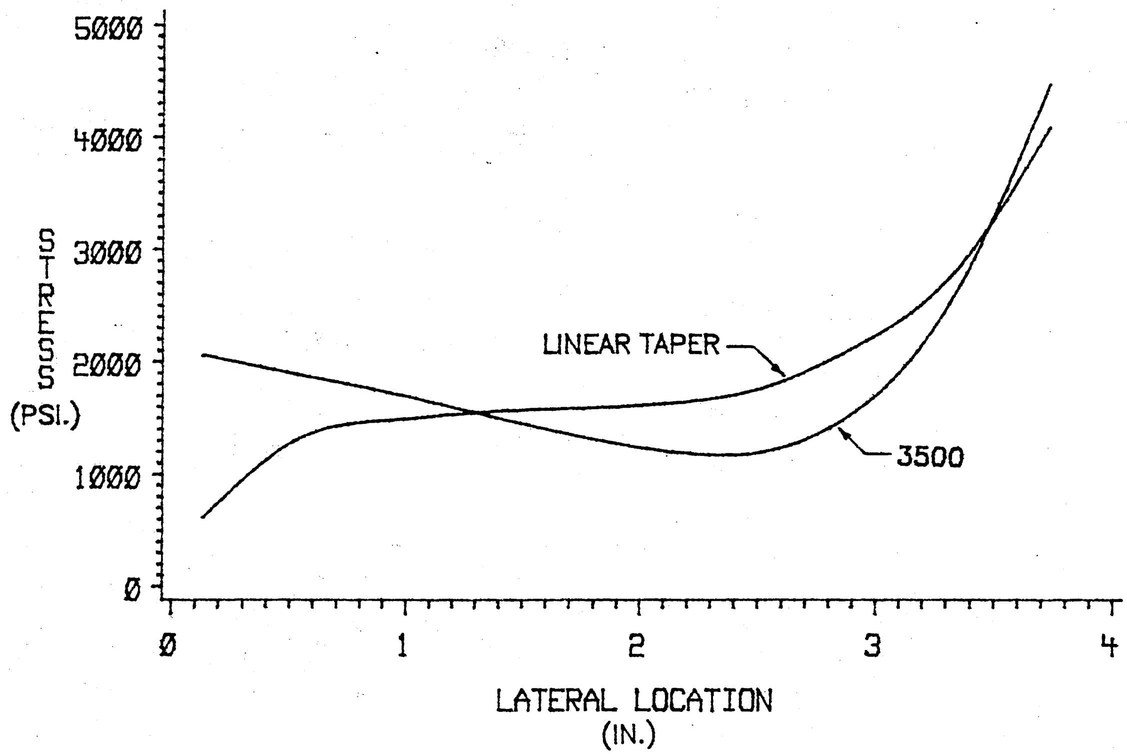


Figure 20. Lateral Stresses at Exit of Roll for Circular Arc 3500 and Linear Taper Profiles

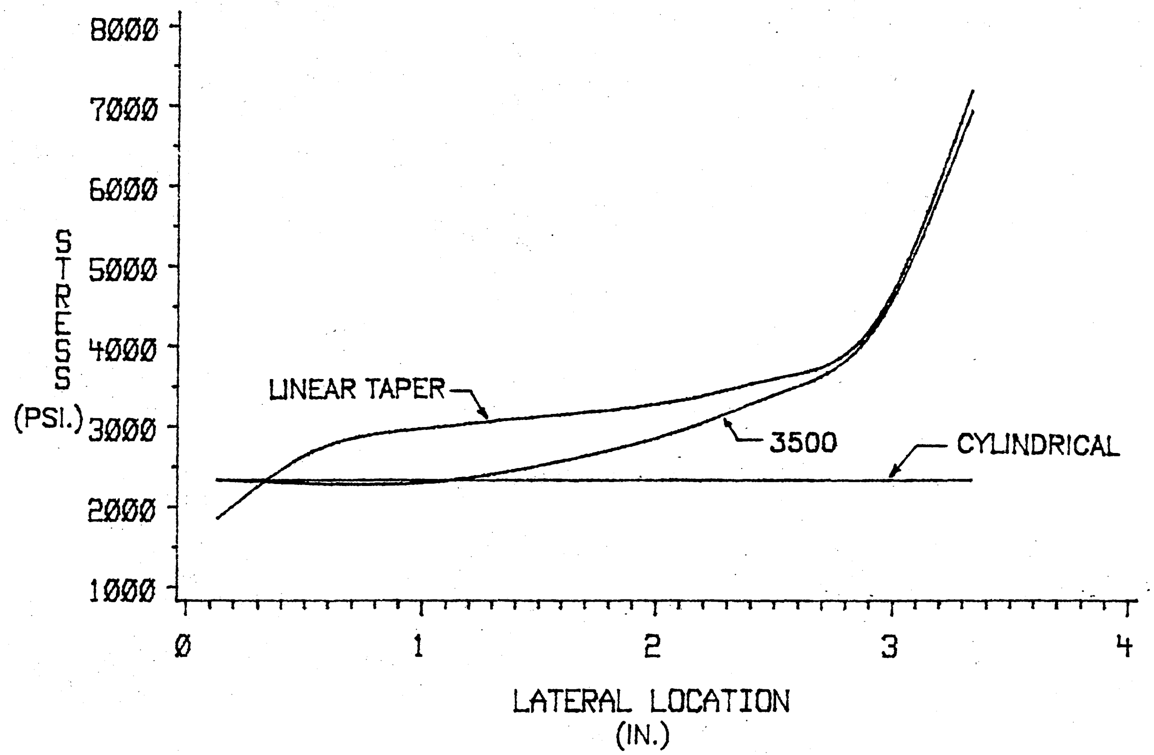


Figure 21. Machine Direction Stresses at Exit of Roll for Circular Arc 3500 and Linear Taper Profiles

TABLE IV

MACHINE DIRECTION STRESSES AT EXIT OF CONCAVE ROLL

Ψ (In.)	Stress (PSI)				
	Cylindrical	750	3500	25000	Linear Taper
0.1333	2337.4	1551.4	2332.2	2660.2	1855.9
0.5333	2338.0	1742.9	2285.9	2585.1	2686.8
0.9333	2338.3	2317.2	2293.5	2497.1	2957.7
1.3333	2337.9	3138.7	2428.8	2411.7	3081.4
1.7333	2338.0	4146.0	2658.9	2424.7	3190.8
2.1333	2338.7	5326.0	2981.2	2564.4	3345.9
2.5333	2338.3	6715.0	3435.0	2876.1	3619.9
2.9333	2338.9	8513.4	4258.4	3697.6	4332.8
3.3333	2339.1	11769.0	6933.4	6510.0	7188.1

TABLE V

LATERAL STRESSES AT EXIT OF CONCAVE ROLL

Ψ (In.)	Stress (PSI) 3500	25000	Linear Taper
0.1333	2059.2	2502.8	611.7
0.5333	1885.3	2387.8	1306.9
0.9333	1720.4	2288.6	1482.9
1.3333	1529.2	2252.9	1553.7
1.7333	1338.3	2194.8	1587.7
2.1333	1195.4	2103.3	1632.9
2.5333	1202.5	2054.1	1774.1
2.9333	1584.4	2109.0	2155.6
3.3333	2583.8	2338.9	2791.5
3.7333	4466.8	3235.5	4081.3

of the 3500 curves in Figures 20 and 21 are typical of the concave profiles; that is, they are all concave-up with no inflection points. This suggests a second order variation in the stresses since a second-order curve can have no inflection points. Recall that the equation of the circular arc profile is itself second-order; a second-order profile leads to a second-order stress variation. Since the lateral stress is related to the spreading forces, the nonlinearity of the curve--with its decrease then rapid increase toward the outside edge--illustrates the nonlinearity of the spreading action. The variation of lateral stresses for the linear taper profile is also shown in Figure 20. The linear taper profile is a first-order curve and the presence of an inflection point in the lateral stress curve suggests a third-order variation in lateral stresses for this profile.

The variations of lateral and machine direction stresses for the circular arc 750 and 25000 profiles are presented in Figures 22 and 23, respectively. These curves are of the same form as the circular arc 3500 profile discussed previously. The circular arc 25000 curve of Figure 22 illustrates the greater spreading ability of this profile over the 3500 profile by higher lateral stress everywhere except at the outer edge. The 25000 curve is flatter, indicative of the smoother profile and more even spreading. The circular arc 750 curve is omitted from this plot because of the negative (compressive) lateral stresses it induces at the exit of the roll which cannot be supported by the membrane elements used in modeling the web. The 750 profile does induce large machine direction stresses (Figure 23), so large as to induce a stress inversion near the centerline. The machine direction stress variations for the 3500 and 25000 profiles are similar to the lateral stress varia-

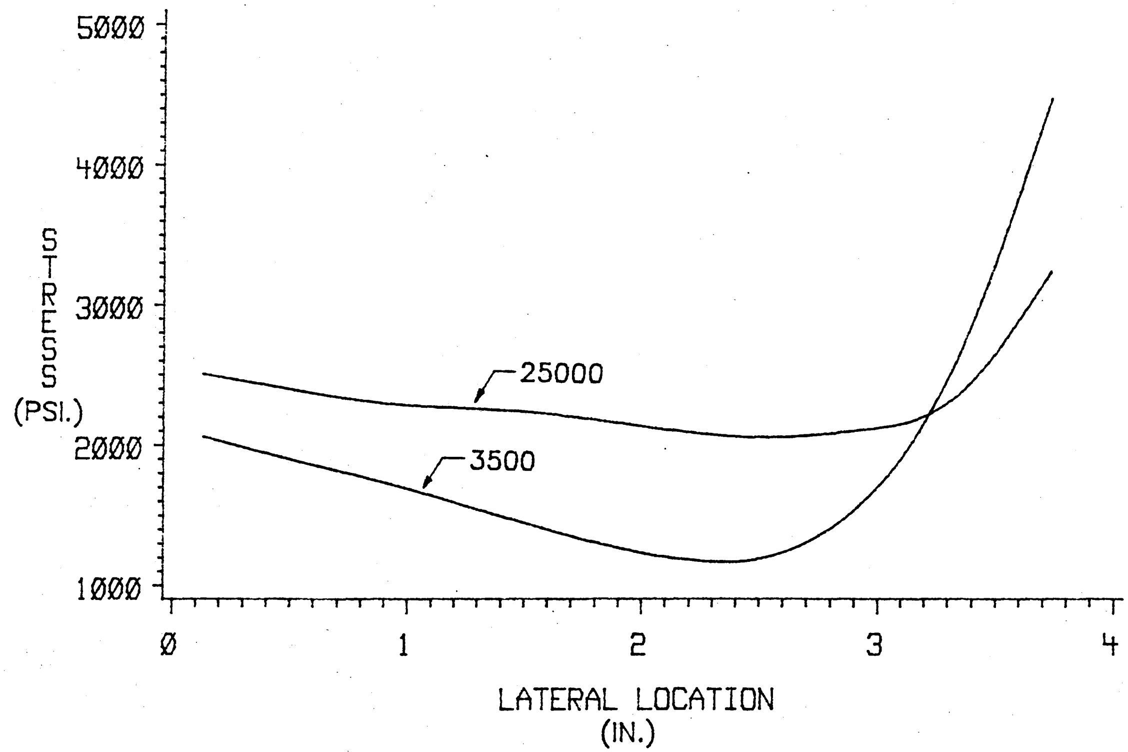


Figure 22. Lateral Stresses at Exit of Roll
for Circular Arc Profiles

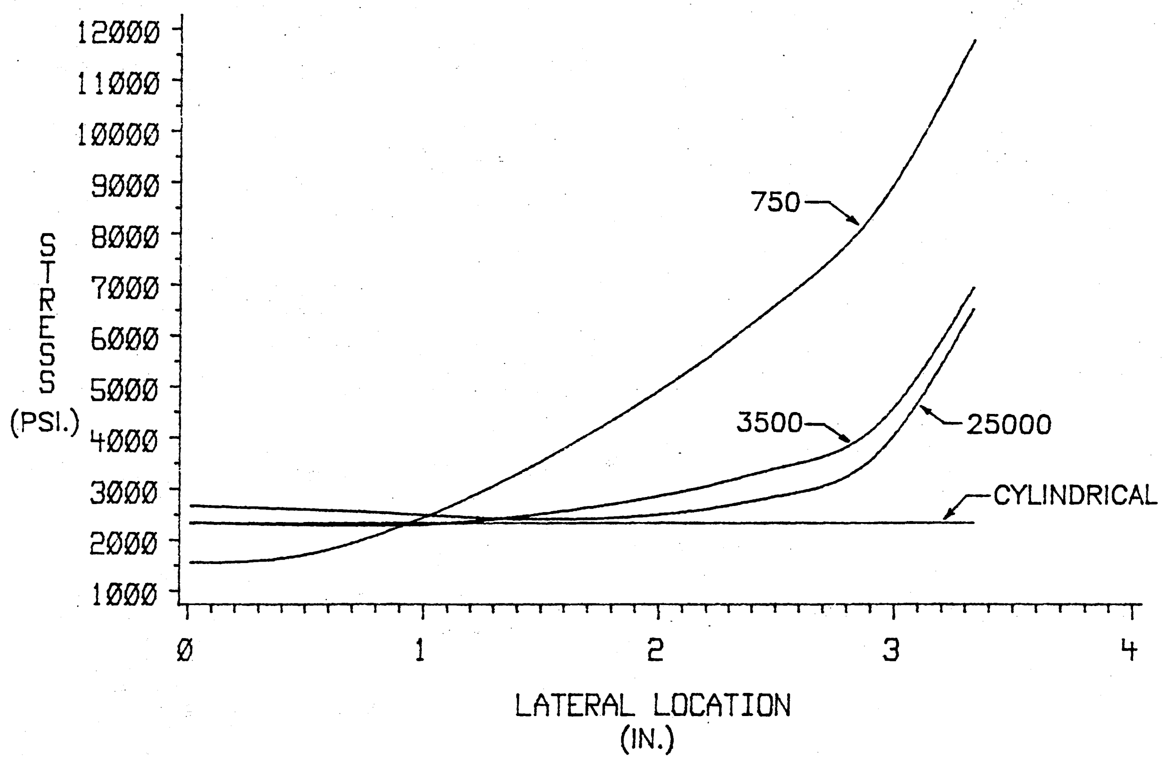


Figure 23. Machine Direction Stresses at Exit of Roll for Circular Arc Profiles

tions in that at the centerline the 25000 profile produces larger stresses but near the outside edge the larger 3500 profile dominates the two.

CHAPTER V

CONCLUSIONS

5.1 Overview

In this study finite element methods were used to study the mechanics of a web on a concave roll. A general finite element model was developed. Through a series of coordinate transformations, the global structure stiffness matrix is transformed to systems of average local coordinates which allows the doubly curved geometry of a web on a concave roll to be properly modeled. The finite element computer code CONCAVE. FOR was developed to analyze the model. The code simulates the spreading action of the concave roll by iteratively calculating and applying spreading forces to the web on the roll until the spreading forces and friction forces are equal. The computer code was used to investigate the effects of spreading on the entry and exit spans, and on the concave roll for circular arc and linear taper profiles.

Examination of the normal force distributions revealed little difference between the circular arc and linear taper profiles. For both profiles the normal forces increased nonlinearly toward the outside edge with the maximum normal force occurring on the outside edge at the roll exit. Consequently, the greatest amount of spreading occurs at this location.

Studies of the relative magnitudes and distributions of the spreading effect produced by various roll profiles were also conducted. Plotting the lateral displacements of the outside edge of the web provided a means of analysis and comparison. For all profiles studied, this investigation showed that the lateral contraction of the entry and exit spans was greater than that produced by the Poisson effect alone, and that the largest lateral displacements occurred at the roll exit. This spreading effect is lost in the exit span; thus, if the spreading is to be retained, placement of a cylindrical roll near the exit of the concave roll is required. Comparing a circular arc profile and a linear taper profile with the same base and outside radii revealed an advantage in the use of the linear taper profile in terms of spreading and ease of manufacture. Investigations of various circular arc profiles indicated that as the radius of curvature is increased, the spreading is also increased up to a point at which any further increases result in rapid decrease in spreading.

Investigations of the variations of the maximum lateral and machine direction stresses which occur at the exit of the roll showed a second-order variation in lateral and machine direction stresses for the (second-order) circular arc profiles and a third-order variation in stress for the (first-order) linear taper profile. For all profiles studied, the stresses increased rapidly to a maximum at the outside edge.

5.2 Recommendations for Future Study

Future research should include experimental verification of the finite element model and results presented in this study. The results of the experimental investigation should then be used to modify the existing model if necessary. After verification of the model, sensitivity analyses of the various model parameters may be conducted.

The model used in this study assumes a constant coefficient of friction between the web and roll. To more realistically model the actual situation, the computer code CONCAVE.FOR may be modified to account for variations in traction between the web roll. The code may also be modified so as to make it more efficient, allowing the use of larger models with shorter execution times.

The mesh generation computer program MSHGNR.FOR may be modified to generate models for other roll profiles and roll types such as the curved axis roll.

REFERENCES

[1] Swift, H. W. "Cambers for Belt Pulleys." Proceedings, Institution of Mechanical Engineers (London), Vol. 122 (June, 1932), pp. 627-683.

[2] Butler, T. "How Concave Rolls Can Correct Dryer Fabric Bowing." Paper Trade Journal, Vol. 169, No. 3 (March, 1982), pp. 70-72.

[3] "Theory and Application of LORIG-ALIGNER Self-Centering Rolls." U.S. Steel Corporation product brochure. Pittsburg, Pennsylvania, no date.

[4] Daly, D. A. "Factors Controlling Traction Between Webs and Their Carrying Rolls." TAPPI, Vol. 48, No. 9 (September, 1965), pp. 88A-90A.

[5] Soong, Tsai-Chen, and Chun Li. "An Elastic Analysis of Multiroll Endless Web Systems." Journal of Dynamic Systems, Measurements, and Control, Vol. 101 (December, 1979), pp. 308-313.

[6] Zienkiewicz, O. C. The Finite Element Method. 3rd ed. New York: McGraw-Hill, 1977.

[7] Shelton, J. J. "Dynamics of Web Tension Control With Velocity or Torque Control." Proceedings, 1986 American Control Conference, Seattle, Washington, Vol. 3, pp. 1423-1427.

[8] Segerlind, L. J. Applied Finite Element Analysis. 2nd ed. New York: John Wiley and Sons, Inc., 1984.

[9] Conte, S. D., and C. deBoor. Elementary Numerical Analysis. 3rd ed. New York: McGraw-Hill, 1980.

[10] Pfieffer, J. D. "Web Guidance Concepts and Applications." TAPPI, Vol. 60, No. 12 (December, 1977), pp. 53-58.

[11] Papandreadis, A. "The Development of Finite Element Modeling Techniques of Webs and the Analysis of Web Wrinkle Formation." (Unpublished M.S. thesis, Oklahoma State University, 1986.)

APPENDIX

```
C ***************************************************************
C
C                         PROGRAM MSHGNR.FOR
C
C THIS PROGRAM GENERATES THE FINITE ELEMENT DATA NEEDED FOR
C THE PROGRAM CONCAVE.FOR
C IT GENERATES THE MESH DATA, ENFORCED DISPLACEMENTS,
C AND THE APPLIED FORCES
C THE PROGRAM CREATES THREE FILES: MESH.DAT CONSTR.DAT FORCE.DAT
C
C ***************************************************************
C
       DOUBLE PRECISION R,Y(400),CAPR
       DIMENSION I(500),J(500),K(500)
       DIMENSION X(400),XC(400),Z(400)
C
C **********************************
C DEFINITION OF THE INPUT VARIABLES
C **********************************
C
C MODEL PARAMETERS
C      WW - WIDTH OF THE WEB
C      ALRB - LENGTH OF WEB BEFORE THE ROLLER
C      ALAR - LENGTH OF WEB AFTER THE ROLLER
C      RZERO - ROLLER BASE RADIUS
C      R - THE RADIUS OF ROLLER AS A FUNCTION OF Y
C      CAPR - CIRCULAR ARC PROFILE: RADIUS OF CURVATURE
C             LINEAR TAPER PROFILE: RADIUS AT OUTSIDE EDGE
C      NXB - NUMBER OF ELEMENT INTERVALS IN THE X-DIRECTION
C      NY - NUMBER OF ELEMENT INTERVALS IN THE Y-DIRECTION
C      NZA - NUMBER OF ELEMENT INTERVALS IN THE Z-DIRECTION
C      EM - MODULUS OF ELASTICITY OF WEB
C      TH - THICKNESS OF WEB
C      FORCE - NODAL FORCE APPLIED AT END OF WEB
C
C NODAL COORDINATES
C      X(I) - X COORDINATES OF NODES IN NUMERICAL SEQUENCE
C      Y(I) - Y COORDINATES OF NODES IN NUMERICAL SEQUENCE
C      Z(I) - Z COORDINATES OF NODES IN NUMERICAL SEQUENCE
C
C ELEMENT CONNECTIVITIES
C      N - ELEMENT NUMBER
C      I(N) - NUMERICAL VALUE OF NODE I
C      J(N) - NUMERICAL VALUE OF NODE J
C      K(I) - NUMERICAL VALUE OF NODE K
C
C
C ****************************************
C      INPUT SECTION OF THE PROGRAM
C ****************************************
C
       READ(5,*) WW
       READ(5,*) ALBR
       READ(5,*) ALAR
       READ(5,*) RZERO
       READ(5,*) CAPR
       READ(5,*) NXB
       READ(5,*) NZA
       READ(5,*) NY
       READ(5,*) EM
C
       FORCE=2.5
C
C CALCULATION OF NN AND NE
C
 10    NXP1=NXB+1
       NYP1=NY+1
       NZP1=NZA+1
```

```
C
C NUMBER OF NODES NN
C
C                    BEFORE ROLLER
C
       NNB=NYP1*NXP1-NYP1
C
C                    ON ROLLER
C
       NNO=11*NYP1
C
C                    AFTER ROLLER
C
       NNA=NZP1*NYP1-NYP1
C
C                    TOTAL
C
       NN=NNB+NNO+NNA
C
C NUMBER OF ELEMENTS NE
C
C                    BEFORE ROLLER
C
       NEB=(NY*2)*NXB
C
C                    ON ROLLER
C
       NEO=(NY*2)*10
C                    AFTER ROLLER
C
       NEA=(NY*2)*NZA
C
C                    TOTAL
C
       NE=NEB+NEO+NEA
C
C WRITE MESH DATA TO FILE MESH.DAT
C
       WRITE(10,*) NN
       WRITE(10,*) NE
C
C ***********************************************
C        CALCULATION OF THE NODAL COORDINATES
C ***********************************************
C
C MESH BEFORE ROLLER
C
C X COORDINATES
C
       DXB=(ALBR-0.25)/(NXB-1)
       X(1)=0.0
       DO 100 II=2,NXP1
       IF(II.EQ.NXP1) THEN
         X(II)=X(II-1)+0.25
         GO TO 100
       END IF
       X(II)=X(II-1)+DXB
 100   CONTINUE
C
C Y COORDINATES
C
       DY=WW/NY
       Y(1)=0.0
       DO 200 II=2,NYP1
  200  Y(II)=Y(II-1)+DY
C
C Z COORDINATES
C WRITE NODAL COORDINATES TO FILE MESH.DAT
C
       DO 400 IX=1,NXP1
       DO 300 IY=1,NYP1
       R=-DSQRT(CAPR**2-Y(IY)**2)+CAPR+RZERO
       SLOPEX=(R-RZERO)/ALBR
       Z(IY)=SLOPEX*X(IX)
```

```
      WRITE(10,201) X(IX),Y(IY),Z(IY)
 201  FORMAT(F9.5,2X,F9.5,2X,F15.10)
 300  CONTINUE
 400  CONTINUE
C
C MESH ON ROLLER
C
C X,Z COORDINATES
C WRITE NODAL COORDINATES TO FILE MESH.DAT
C
      THETA=9.
      DO 500 ISTEP=1,10
      THRAD=(THETA*3.14159265)/180.
      DO 475 IY=1,NYP1
      R=-DSQRT(CAPR**2-Y(IY)**2)+CAPR+RZERO
      XC(IY)=ALBR+R*SIN(THRAD)
      Z(IY)=(R*COS(THRAD))-RZERO
      WRITE(10,201) XC(IY),Y(IY),Z(IY)
 475  CONTINUE
      THETA=THETA+9.
 500  CONTINUE
C
C MESH AFTER ROLLER
C
C X,Y,Z COORDINATES
C WRITE NODAL COORDINATES TO FILE MESH.DAT
C
      DZ=ALAR/NZA
      Z(1)=-RZERO
      DO 750 L=2,NZP1
 750  Z(L)=Z(L-1)-DZ
C
      DO 700 IZ=2,NZP1
      DO 600 IY=1,NYP1
      R=-DSQRT(CAPR**2-Y(IY)**2)+CAPR+RZERO
      SLOPEZ=(R-RZERO)/(ALAR+RZERO)
      XC(IY)=ALBR+SLOPEZ*Z(IZ)+R
      WRITE(10,201) XC(IY),Y(IY),Z(IZ)
 600  CONTINUE
 700  CONTINUE
C
C ******************************************************
C        CALCULATION OF THE ELEMENT CONNECTIVITIES
C ******************************************************
C ODD NUMBERED ELEMENTS
C
      ICOUNT=0
      I(1)=2
      J(1)=1
      K(1)=NYP1+1
      NJ=1
      WRITE(10,*) NJ,I(1),J(1),K(1)
      DO 1200 N=3,NE,2
      ICOUNT=ICOUNT+1
      IREM=MOD(ICOUNT,NY)
      IF(IREM.EQ.0) GO TO 1100
      I(N)=I(N-2)+1
      J(N)=J(N-2)+1
      K(N)=K(N-2)+1
      GO TO 1200
 1100 I(N)=I(N-2)+2
      J(N)=J(N-2)+2
      K(N)=K(N-2)+2
 1200 CONTINUE
C
C EVEN NUMBERED ELEMENTS
C
      ICOUNT=0
      I(2)=2
      J(2)=NYP1+1
      K(2)=NYP1+2
      NK=2
```

```
      WRITE(10,*) NK,I(2),J(2),K(2)
      DO 1400 N=4,NE,2
      ICOUNT=ICOUNT+1
      IREM=MOD(ICOUNT,NY)
      IF(IREM.EQ.0) GO TO 1300
      I(N)=I(N-2)+1
      J(N)=J(N-2)+1
      K(N)=K(N-2)+1
      GO TO 1400
 1300 I(N)=I(N-2)+2
      J(N)=J(N-2)+2
      K(N)=K(N-2)+2
 1400 CONTINUE
C
C WRITE ELEMENT DATA TO FILE MESH.DAT
C
      DO 1500 N=3,NE
 1500 WRITE(10,*) N,I(N),J(N),K(N)
C
C WRITE MESH PARAMETERS TO FILE MESH.DAT
C
      WRITE(10,*) NY,NYP1,WW,NNB,NNO
C
C
C **********************
C     CONSTRAINTS
C     ENFORCED DISPLACEMENTS
C **********************
C
C BEFORE ROLLER
C
      VALUE=0.0
      DO 2001 IA=1,NYP1
      DO 2000 JA=1,3
      IF(JA.EQ.1) IDOF=3*IA-2
      IF(JA.EQ.2) IDOF=3*IA-1
      IF(JA.EQ.3) IDOF=3*IA
 2000 WRITE(14,55) IA,IDOF,VALUE
  55  FORMAT(I4,2X,I4,2X,F3.1)
 2001 CONTINUE
C
      DO 2100 IB=NYP1+1,NNB-NY
      IDOF=3*IB
      WRITE(14,55) IB,IDOF,VALUE
      IREM=MOD((IB-1),NYP1)
      IF(IREM.EQ.0) THEN
        IDOF=3*IB-1
        WRITE(14,55) IB,IDOF,VALUE
      END IF
 2100 CONTINUE
C
C ENFORCED DISPLACEMENTS ON ROW IN FRONT OF ROLLER
C
      LA=1
      DO 2110 IW=NNB-NY,NNB
      IDOF=3*IW-2
      EPSILN=(FORCE/DY)/(TH*EM)
      R=-DSQRT(CAPR**2-Y(LA)**2)+CAPR
      DELTA1=ALBR*(1.-((1.-EPSILN)/(1.+(R/RZERO))))
      UTOT=DELTA1
      WRITE(14,56) IW,IDOF,UTOT
      IDOF=3*IW
      WRITE(14,55) IW,IDOF,VALUE
      LA=LA+1
 2110 CONTINUE
C
C ON ROLLER
C
      DO 2300 ID=NNB+1,NNB+NNO
      IDOF=3*ID
      WRITE(14,55) ID,IDOF,VALUE
      IREM=MOD((ID-1),NYP1)
```

```
         IF(IREM.EQ.0) THEN
            IDOF=3*ID-1
            WRITE(14,55) ID,IDOF,VALUE
         END IF
 2300    CONTINUE
C
C ENFORCED DISPLACEMENTS ON LAST ROW ON ROLLER
C
         LL=1
         DO 2310 IV=NNB+NNO-NY,NNB+NNO
         IDOF=3*IV-2
         EPSILN=(FORCE/DY)/(TH*EM)
         ALEN=(3.14159*RZERO)/2.0
         R=-DSQRT(CAPR**2-Y(LL)**2)+CAPR
         DELTA1=ALBR*(1.-((1.-EPSILN)/(1.+(R/RZERO))))
         DELTA2=ALEN*(1.-((1.-EPSILN)/(1.+(R/RZERO))))
         UTOT2=DELTA2+DELTA1
         WRITE(14,56) IV,IDOF,UTOT2
  56     FORMAT(I4,2X,I4,2X,F14.12)
         LL=LL+1
 2310    CONTINUE
C
C AFTER ROLLER
C
         DO 2200 IC=NNB+NNO+1,NN
         IDOF=3*IC
         WRITE(14,55) IC,IDOF,VALUE
         IREM=MOD((IC-1),NYP1)
         IF((IREM.EQ.0).OR.((IC.GE.(NN-NY)).AND.(IC.LE.NN))) THEN
            IDOF=3*IC-1
            WRITE(14,55) IC,IDOF,VALUE
         END IF
 2200    CONTINUE
C
         IQ=0
         IDOF=0
         WRITE(14,55) IQ,IDOF,VALUE
C
C ******************
C        FORCES
C ******************
C
         DO 2500 IE=NN-NY,NN
         IDOF=3*IE-2
         VF=FORCE
         IF((IE.EQ.(NN-NY)).OR.(IE.EQ.NN)) VF=FORCE/2.
         WRITE(12,*) IE,IDOF,VF
 2500    CONTINUE
         IDOF=0
         WRITE(12,*) IQ,IDOF,VALUE
C
         STOP
         END
```

```
C ************************************************************
C
C                    PROGRAM CONCAVE.FOR
C
C THIS PROGRAM ANALYZES A WEB ON A CONCAVE ROLLER
C IT ITERATIVELY SPREADS THE WEB ON THE ROLLER TO THE
C POINT AT WHICH THE SPREADING FORCES EQUAL THE AVAILABLE
C FRICTION FORCES
C THE PROGRAM HANDLES ONLY LINEAR TRIANGULAR PLANE STRESS ELEMENTS
C
C ************************************************************
C
C
        PARAMETER(IMAXNN=231)
        COMMON/GRAD/B(3,6),AR2
        COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $  GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
        COMMON/MTL/EM,PR,TH,AMU
        COMMON/TLE/TITLE(20)
        COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
        COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
        COMMON/DUMM/DUM1(9,9),DUM2(9,9),DUM3(3,3*IMAXNN),
     $  DUM4(3,3*IMAXNN),DUM5(3*IMAXNN,3),DUM6(3*IMAXNN,3)
        COMMON/TMATX/AL(3,3),T(9,9),ALT(3,3),TT(9,9)
        COMMON/DOF/F(3*IMAXNN),U(3*IMAXNN),STRA(6),STRE(6),THM,
     $  TM,S1,S2
        COMMON/LOCAL/XYZ(9),STOR(9),XL(3),YL(3),ZL(3),
     $  FL(3*IMAXNN)
        COMMON/LAM/TEMP(3,3),ALN(3,3),ALNT(3,3)
        COMMON/PARAM/NY,NYP1,W
        COMMON/PARAM2/NNB,NNO
        COMMON/ANGL/SLOPE,FRICTF(IMAXNN),FEXT(IMAXNN),ADIV(IMAXNN),
     $  CONST(IMAXNN)
        DIMENSION W(3*IMAXNN,3*IMAXNN)
        INTEGER IPIVOT(3*IMAXNN)
C
C ****************************************************************
C
C                            MAIN PROGRAM
C
C ****************************************************************
C
C
C *******************
C       INPUT SECTION OF THE PROGRAM
C ******************
C
        CALL INPUTS
C
C ******************
C       GENERATION OF THE SYSTEM OF EQUATIONS
C ******************
C
C INITIALIZATION OF THE STIFFNESS MATRIX
C
        DO 555 I=1,NP
        DO 555 J=1,NP
 555    GSM(I,J)=0.0
C
C START OF THE LOOP FOR THE ELEMENT MATRICES
C
        ITER=0
        IELR=0
        KK=1
C
C GENERATION OF THE NODAL DEGREES OF FREEDOM
C RETRIEVAL OF THE NODAL COORDINATES
C
 5      DO 10 I=1,3
        J=NEL(KK,I)
        NS(3*I-2)=J*3-2
        NS(3*I-1)=J*3-1
```

```
         NS(3*I)=J*3
         X(I)=XC(J)
         Y(I)=YC(J)
 10      Z(I)=ZC(J)
C
         CALL LAMDA(KK)
         CALL TRNSMX
C
C GENERATION OF THE LOCAL COORDINATES
C
         JJ=1
         DO 30 I=1,9,3
         XYZ(I)=X(JJ)
         XYZ(I+1)=Y(JJ)
         XYZ(I+2)=Z(JJ)
         JJ=JJ+1
  30     CONTINUE
C
         DO 40 I=1,9
         STOR(I)=0.0
         DO 40 J=1,9
  40     STOR(I)=STOR(I)+T(I,J)*XYZ(J)
C
         JK=1
         DO 50 I=1,3
         XL(I)=STOR(JK)
         YL(I)=STOR(JK+1)
         ZL(I)=STOR(JK+2)
         JK=JK+3
  50     CONTINUE
C
C *******************
C        CALCULATION OF THE ELEMENT MATRICES
C        TRANSFORMATION TO GLOBAL COORDINATES
C        ASSEMBLY VIA DIRECT STIFFNESS PROCEDURE
C *******************
C
         CALL ELSTMX(KK)
         CALL TRNSFM
         CALL ASSEMB
C
         KK=KK+1
         IF(KK.LE.NE) GO TO 5
C
C *******************
C        SKEWED COORDINATE TRANSFORMATION
C *******************
C
         KN=1
C
 200     CALL SKEWED(KN)
C
         KN=KN+1
         IF(KN.LE.NN) GO TO 200
C
C *******************
C        MODIFICATION OF THE SYSTEM OF EQUATIONS
C        DATA IS CALLED BY MODIFY
C ******************
C
         CALL MODIFY
C
C ******************
C        SOLUTION OF THE MODIFIED SYSTEM OF EQUATIONS
C ******************
C
         CALL FACTOR(GTSM,IPIVOT,IFLAG)
         IF(IFLAG.EQ.0) THEN
           WRITE(15,*) 'SYSTEM CANNOT BE SOLVED--0 ON DIAGONAL'
           GO TO 110
         END IF
         CALL SUBST(GTSM,IPIVOT,F,U)
```

```
C
C ******************
C        OUTPUT OF THE CALCULATED NODAL DISPLACEMENTS
C        DATA IS WRITTEN TO FILE OUT.DAT
C ******************
C
         IFLAG4=0
  250    WRITE(15,31) TITLE,NN,NE
  31     FORMAT(1H1,////,10X,20A3,//,13X,'NN =',I4/
     $   ,13X,5HNE = ,I4)
         WRITE(15,32) EM,PR,TH,AMU
  32     FORMAT(//10X,'PARAMETER VALUES'
     $   /13X,4HEM =,E15.5/13X,4HPR =,
     $   E15.5/13X,4HTH =,E15.5,/13X,
     $   4HMU =,E15.5)
         WRITE(15,2)
  2      FORMAT(///10X,25HNODAL DISPLACEMENT VALUES/
     $   //10X,4HNODE,6X,12HX DEFLECTION,6X,12HY DEFLECTION,
     $   6X,12HZ DEFLECTION)
C
         DO 6 I=1,NN
         WRITE(15,4) I,U(3*I-2),U(3*I-1),U(3*I)
  4      FORMAT(11X,I3,3X,E15.6,3X,E15.6,3X,E15.6)
  6      CONTINUE
C
C ******************
C        CALCULATION OF THE ELEMENT STRESS AND STRAIN COMPONENTS
C        AND THE PRINCIPAL STRESSES
C ******************
C
         IELR=1
C
C START OF THE ELEMENT LOOP
C
         KK=1
C
C GENERATION OF THE NODAL DEGREES OF FREEDOM
C
  12     DO 15 I=1,3
         J=NEL(KK,I)
         NS(3*I-2)=3*J-2
         NS(3*I-1)=3*J-1
         NS(3*I)=3*J
         X(I)=XC(J)
         Y(I)=YC(J)
  15     Z(I)=ZC(J)
C
         CALL LAMDA(KK)
         CALL TRNSMX
C
C GENERATION OF THE LOCAL COORDINATES
C
         JJ=1
         DO 60 I=1,9,3
         XYZ(I)=X(JJ)
         XYZ(I+1)=Y(JJ)
         XYZ(I+2)=Z(JJ)
         JJ=JJ+1
  60     CONTINUE
C
         DO 70 I=1,9
         STOR(I)=0.0
         DO 70 J=1,9
  70     STOR(I)=STOR(I)+T(I,J)*XYZ(J)
C
         JK=1
         DO 80 I=1,3
         XL(I)=STOR(JK)
         YL(I)=STOR(JK+1)
         ZL(I)=STOR(JK+2)
         JK=JK+3
  80     CONTINUE
```

```
C
C RETRIEVAL OF THE NODAL DISPLACEMENTS
C
        DO 16 I=1,9,3
        NS1=NS(I)
        NS2=NS(I+1)
        UU(I)=U(NS1)
 16     UU(I+1)=U(NS2)
C
        CALL SANDS(KK)
C
C *******************
C       OUTPUT OF THE STRESS RESULTS TO FILE OUT.DAT
C *******************
C
        WRITE(15,8) KK
  8     FORMAT(/10X,'ELEMENT ',I4)
        WRITE(15,20) STRA(1),STRE(1),S1,STRA(2),STRE(2),S2,
     $  STRA(3),STRE(3),TM,THM
 20     FORMAT(15X,5HEXX =,E12.5,5X,5HSXX =,E12.5,5X,5HS1  =,
     $  E12.5/15X,5HEYY =,E12.5,5X,5HSYY =,E12.5,5X,5HS2  =,
     $  E12.5/15X,5HGXY =,E12.5,5X,5HTXY =,E12.5,4X,
     $  6HTMAX =,E12.5/59X,5HANGLE,F8.2,4H DEG)
C
        KK=KK+1
        IF(KK.LE.NE) GO TO 12
C
C *******************
C       CALCULATION OF NORMAL FORCES
C *******************
C
 101    IROWS=3*(NNB+1)-2
        IROWF=3*(NNB+NNO)
        DO 90 I=IROWS,IROWF
        FL(I)=0.0
        DO 90 J=1,NP
        FL(I)=FL(I)+GSM(I,J)*U(J)
  90    CONTINUE
        IF(IFLAG4.EQ.1) THEN
          WRITE(15,361)
          DO 500 KN=NNB+NYP1+2,NNB+NNO-NYP1
          WRITE(15,363) KN,FEXT(KN),FRICTF(KN),ADIV(KN)
 500      CONTINUE
          WRITE(15,362) ITER
          GO TO 100
        END IF
C
C ******************
C CHECK TO SEE IF SPREADING WILL OCCUR IN FIRST ITERATION
C  ****************
C
 95      ITER=ITER+1
C
        IF(ITER.EQ.1) THEN
        DO 275 KN=NNB+3,NNB+NYP1
        CALL CHKSLP(KN,IFLAG1)
        IREM=MOD((KN-1),NYP1)
        IF(IREM.EQ.0)  IFLAG1=1
        IF(IFLAG1.EQ.0) THEN
          WRITE(15,*) ' SPREADING WILL NOT OCCUR-SLOPE = 0'
          WRITE(15,*) ' IN ITERATION NUMBER 1'
          GO TO 100
        END IF
275     CONTINUE
        END IF
C
C *******************
C CALCULATE SPREADING FORCES
C APPLY SPREADING FORCES TO NODES ON ROLLER
C *******************
C
```

```
  290     DO 300 KN=NNB+NYP1+2,NNB+NNO-NYP1
          IREM=MOD((KN-1),NYP1)
          IF(IREM.EQ.0) GO TO 300
          CALL FRICTN(KN)
          CALL APFORS(KN,ITER)
  300     CONTINUE
C
C ******************
C SOLVE FOR NEW DISPLACEMENTS AND FORCES
C ******************
C
          CALL SUBST(GTSM,IPIVOT,F,U)
C
C CALCULATE NEW NORMAL FORCES
C
          IROWS=3*(NNB+1)-2
          IROWF=3*(NNB+NNO)
          DO 325 I=IROWS,IROWF
          FL(I)=0.0
          DO 325 J=1,NP
          FL(I)=FL(I)+GSM(I,J)*U(J)
  325     CONTINUE
C
C ******************
C CHECK TO SEE IF SLIPPAGE OCCURRED
C ******************
C
          IFLAG3=0
 361      FORMAT(/' NODE',5X,'SPREADING FORCE',5X,'FRICTION FORCE',
     $    5X,'RATIO'/)
          DO 350 KN=NNB+NYP1+2,NNB+NNO-NYP1
          CALL FRICTN(KN)
          CALL CHKFOR(KN,IFLAG2)
          IREM=MOD((KN-1),NYP1)
          IF(IREM.EQ.0) IFLAG2=1
  363     FORMAT(I4,7X,E12.5,8X,E12.5,6X,F6.4)
          IF(IFLAG2.EQ.0) IFLAG3=1
  350     CONTINUE
C
          IF(ITER.GT.100) GO TO 520
          IF(IFLAG3.EQ.1) GO TO 95
  362       FORMAT(/' IN ITERATION NUMBER ',I4)
C
C
 520      REWIND(15)
          IFLAG4=1
          GO TO 250
C
C
  100     CALL PLOT
C
  110     STOP
          END
C
C ****************************************************************
C
C
C ****************************
          SUBROUTINE INPUTS
C ****************************
C
          PARAMETER(IMAXNN=231)
          COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $    GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN)
          COMMON/MTL/EM,PR,TH,AMU
          COMMON/TLE/TITLE(20)
          COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
          COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
          COMMON/PARAM/NY,NYP1,W
          COMMON/PARAM2/NNB,NNO
```

```
C
C ******************
C ******************
C
C DEFINITON OF THE INPUT VARIABLES FOR THIS PROGRAM
C
C ******************
C ******************
C TITLE AND PARAMETERS
C
C     TITLE - A DESCRIPTIVE STATEMENT OF THE PROBLEM
C     NN - NUMBER OF NODES
C     NE - NUMBER OF ELEMENTS
C
C MATERIAL PROPERTIES AND THICKNESS
C
C     EM - MODULUS OF ELASTICITY
C     PR - POISSON'S RATIO
C     TH - THICKNESS OF THE REGION
C     AMU - COEFFICIENT OF FRICTION BETWEEN THE WEB AND ROLL
C
C NODAL COORDINATES
C
C     XC(I) - X COORDINATES OF THE NODES IN NUMERICAL SEQUENCE
C     YC(I) - Y COORDINATES OF THE NODES IN NUMERICAL SEQUENCE
C     ZC(I) - Z COORDINATES OF THE NODES IN NUMERICAL SEQUENCE
C
C ELEMENT DATA
C
C     N - ELEMENT NUMBER
C     NEL(N,I) - NUMERICAL VALUE OF NODE I
C     NEL(N,J) - NUMERICAL VALUE OF NODE J
C     NEL(N,K) - NUMERICAL VALUE OF NODE K
C
C ******************
C        INPUT SECTION
C ******************
C
C INPUT OF THE TITLE CARD AND PARAMETERS
C
        READ(5,3) TITLE
 3      FORMAT(20A3)
        READ(10,*) NN
        READ(10,*) NE
        NP=3*NN
C
C COMPARISON CHECK OF NN AND NE WITH THE VALUES USED
C IN THE DIMENSION STATEMENTS
C
        ISTOP=0
C
C CHECK NUMBER OF NODES
C
        IF(NN.LE.400) GO TO 15
        WRITE(15,10)
 10     FORMAT(10X,'NUMBER OF NODES EXCEEDS 400'/
     $  10X,26HCHECK DIMENSION STATEMENTS/
     $  10X,20HEXECUTION TERMINATED)
        ISTOP=1
C
C CHECK NUMBER OF ELEMENTS
C
 15     IF(NE.LE.500) GO TO 25
        WRITE(15,20)
 20     FORMAT(10X,'NUMBER OF ELEMENTS EXCEEDS 500'/
     $  10X,26HCHECK DIMENSION STATEMENTS/
     $  10X,20HEXECUTION TERMINATED)
        ISTOP=1
 25     IF(ISTOP.EQ.1) STOP
C
C INPUT OF THE MATERIAL PROPERTIES
C
```

```
      READ(5,*) EM
      READ(5,*) PR
      READ(5,*) TH
      READ(5,*) AMU
C
C INPUT OF THE NODAL COORDINATES FROM FILE MESH.DAT
C
      READ(10,*)(XC(I),YC(I),ZC(I),I=1,NN)
C
C INPUT AND ECHO PRINT OF ELEMENT DATA
C CHECK TO SEE IF THE ELEMENTS ARE IN SEQUENCE
C
      NID=0
      DO 45 KK=1,NE
      READ(10,*) N,(NEL(N,I),I=1,3)
      IF((N-1).NE.NID) WRITE(15,42) N
 42   FORMAT(10X,7HELEMENT,I4,16H NOT IN SEQUENCE)
      NID=N
 45   CONTINUE
C
C READ MESH PARAMETERS FROM FILE MESH.DAT
C
      READ(10,*) NY,NYP1,W,NNB,NNO
C
C
C *******************
C     ANALYSIS OF THE NODE NUMBERS
C *******************
C
C INITIALIZATION OF A CHECK VECTOR
C
      DO 50 I=1,NN
 50   ICK(I)=0
C
C CHECK TO SEE IF ANY NODE NUMBER EXCEEDS NP
C
      DO 54 I=1,NE
      DO 52 J=1,3
      K=NEL(I,J)
      ICK(K)=1
 52   IF(K.GT.NN) WRITE(15,53)
 53   FORMAT(/,10X,4HNODE,I4,11H OF ELEMENT,I4,
     $ 13H EXCEEDS NN = ,I4)
 54   CONTINUE
C
C CHECK TO SEE IF ALL NODE NUMBERS THROUGH NN ARE INCLUDED
C
      DO 55 I=1,NN
 55   IF(ICK(I).EQ.0) WRITE(15,56) I
 56  FORMAT(/10X,4HNODE,I4,15H DOES NOT EXIST)
C
      RETURN
      END
C
C
C ********************************
      SUBROUTINE ELSTMX(KK)
C ********************************
C
      PARAMETER(IMAXNN=231)
      COMMON/MTL/EM,PR,TH,AMU
      COMMON/GRAD/B(3,6),AR2
      COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $ GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
      COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
      COMMON/DUMM/DUM1(9,9),DUM2(9,9),DUM3(3,3*IMAXNN),
     $ DUM4(3,3*IMAXNN),DUM5(3*IMAXNN,3),DUM6(3*IMAXNN,3)
      COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
      COMMON/LOCAL/XYZ(9),STOR(9),XL(3),YL(3),ZL(3),
     $ FL(3*IMAXNN)
      DIMENSION C(6,3)
```

```
C
C GENERATION OF THE B MATRIX
C
        DO 20 I=1,3
        DO 20 J=1,6
 20     B(I,J)=0.0
        B(1,1)=YL(2)-YL(3)
        B(1,3)=YL(3)-YL(1)
        B(1,5)=YL(1)-YL(2)
        B(2,2)=XL(3)-XL(2)
        B(2,4)=XL(1)-XL(3)
        B(2,6)=XL(2)-XL(1)
 110    B(3,1)=B(2,2)
        B(3,2)=B(1,1)
        B(3,3)=B(2,4)
        B(3,4)=B(1,3)
        B(3,5)=B(2,6)
        B(3,6)=B(1,5)
        C1=(Y(1)*Z(2)+Y(2)*Z(3)+Y(3)*Z(1)-Y(1)*Z(3)-
     $  Y(2)*Z(1)-Y(3)*Z(2))**2
        C2=(Z(1)*X(2)+Z(2)*X(3)+Z(3)*X(1)-Z(1)*X(3)-
     $  Z(2)*X(1)-Z(3)*X(2))**2
        C3=(X(2)*Y(3)+X(3)*Y(1)+X(1)*Y(2)-X(2)*Y(1)-
     $ X(3)*Y(2)-X(1)*Y(3))**2
        AR2=SQRT(C1+C2+C3)
C
C GENERATION OF THE MATERIALS PROPERTY MATRIX MATRIX D
C
 100    R=EM/(1.-PR**2)
        D(1,1)=R
        D(2,2)=D(1,1)
        D(3,3)=R*(1.0-PR)/2.
        D(1,2)=PR*R
        D(2,1)=D(1,2)
        D(1,3)=0.0
        D(3,1)=0.0
        D(2,3)=0.0
        D(3,2)=0.0
C
        IF(IELR.EQ.1) RETURN
C
C MATRIX MULTIPLICATION TO OBTAIN C = BT * D
C
        DO 22 I=1,6
        DO 22 J=1,3
        C(I,J)=0.0
        DO 22 K=1,3
 22     C(I,J)=C(I,J)+B(K,I)*D(K,J)
C
C MATRIX MULTIPLICATION TO OBTAIN ESM
C
        DO 27 I=1,6
        DO 27 J=1,6
        SUM=0.0
        DO 28 K=1,3
 28     SUM=SUM+C(I,K)*B(K,J)
        ESM(I,J)=SUM*TH/(2.*AR2)
 27     CONTINUE
C
        DO 55 I=1,9
        DO 50 J=1,9
 50     DUM1(I,J)=0.0
 55     CONTINUE
C
C EXPAND ELEMENT STIFFNESS MATRIX TO INCLUDE
C THE OUT-OF-PLANE DEGREES OF FREEDOM
C
        DO 65 I=1,2
        DO 60 J=1,6
        JJ=J
        IF((J.EQ.3).OR.(J.EQ.4)) JJ=J+1
        IF((J.EQ.5).OR.(J.EQ.6)) JJ=J+2
 60     DUM1(I,JJ)=ESM(I,J)
 65     CONTINUE
```

```
C
      DO 75 I=4,5
      DO 70 J=1,6
      JJ=J
      IM1=I-1
      IF((J.EQ.3).OR.(J.EQ.4)) JJ=J+1
      IF((J.EQ.5).OR.(J.EQ.6)) JJ=J+2
   70 DUM1(I,JJ)=ESM(IM1,J)
   75 CONTINUE
C
      DO 85 I=7,8
      DO 80 J=1,6
      JJ=J
      IM2=I-2
      IF((J.EQ.3).OR.(J.EQ.4)) JJ=J+1
      IF((J.EQ.5).OR.(J.EQ.6)) JJ=J+2
   80 DUM1(I,JJ)=ESM(IM2,J)
   85 CONTINUE
C
      DO 95 I=1,9
      DO 90 J=1,9
   90 ESM(I,J)=DUM1(I,J)
   95 CONTINUE
C
      RETURN
      END
C
C
C ********************************
      SUBROUTINE LAMDA(KK)
C ********************************
C
      PARAMETER(IMAXNN=231)
      COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $ GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
      COMMON/TMATX/AL(3,3),T(9,9),ALT(3,3),TT(9,9)
      COMMON/LAM/TEMP(3,3),ALN(3,3),ALNT(3,3)
C
      A=((Y(2)-Y(1))*(Z(3)-Z(1)))-((Y(3)-Y(1))*(Z(2)-Z(1)))
      B=-((X(2)-X(1))*(Z(3)-Z(1)))+((X(3)-X(1))*(Z(2)-Z(1)))
      C=((X(2)-X(1))*(Y(3)-Y(1)))-((X(3)-X(1))*(Y(2)-Y(1)))
C
      IF((A.EQ.0.).AND.(B.EQ.0.)) THEN
         CXX=1.0
         CXY=0.0
         CXZ=0.0
         CYX=0.0
         CYY=1.0
         CYZ=0.0
         CZX=0.0
         CZY=0.0
         CZZ=1.0
         IF(KK.EQ.0) GO TO 100
         GO TO 5
      END IF
C
C DIRECTION COSINES OF THE NORMAL Z
C
      CONST=SQRT(A**2+B**2+C**2)
      CZX=A/CONST
      CZY=B/CONST
      CZZ=C/CONST
C
C DIRECTION COSINES OF X
C
C DETERMINE SIDE OF ELEMENT PARALLEL WITH GLOBAL X AXIS
C
      IF((Y(2)-Y(1)).EQ.0.) THEN
        XL=SQRT((X(2)-X(1))**2+(Y(2)-Y(1))**2+(Z(2)-Z(1))**2)
        CXX=(X(2)-X(1))/XL
        CXY=(Y(2)-Y(1))/XL
        CXZ=(Z(2)-Z(1))/XL
        GO TO 50
      END IF
```

```
C
        IF((Y(3)-Y(1)).EQ.0.) THEN
          XL=SQRT((X(3)-X(1))**2+(Y(3)-Y(1))**2+(Z(3)-Z(1))**2)
          CXX=(X(3)-X(1))/XL
          CXY=(Y(3)-Y(1))/XL
          CXZ=(Z(3)-Z(1))/XL
          GO TO 50
        END IF
C
        IF((Y(3)-Y(2)).EQ.0.) THEN
          XL=SQRT((X(3)-X(2))**2+(Y(3)-Y(2))**2+(Z(3)-Z(2))**2)
          CXX=(X(3)-X(2))/XL
          CXY=(Y(3)-Y(2))/XL
          CXZ=(Z(3)-Z(2))/XL
          GO TO 50
        END IF
C
C DIRECTION COSINES OF Y
C
  50    CYX=(CZY*CXZ)-(CZZ*CXY)
        CYY=-(CZX*CXZ)+(CZZ*CXX)
        IF(ABS(CYY-1.0).LE.0.00005) CYY=1.00
        CYZ=(CZX*CXY)-(CZY*CXX)
C
C GENERATION OF THE LOCAL NODAL LAMDA MATRIX
C
  100   IF(KK.EQ.0) THEN
          TEMP(1,1)=ACOS(CXX)
          TEMP(1,2)=ACOS(CXY)
          TEMP(1,3)=ACOS(CXZ)
          TEMP(2,1)=ACOS(CYX)
          TEMP(2,2)=ACOS(CYY)
          TEMP(2,3)=ACOS(CYZ)
          TEMP(3,1)=ACOS(CZX)
          TEMP(3,2)=ACOS(CZY)
          TEMP(3,3)=ACOS(CZZ)
          RETURN
        END IF
C
C GENERATION OF THE LAMDA MATRIX
C
  5     AL(1,1)=CXX
        AL(1,2)=CXY
        AL(1,3)=CXZ
        AL(2,1)=CYX
        AL(2,2)=CYY
        AL(2,3)=CYZ
        AL(3,1)=CZX
        AL(3,2)=CZY
        AL(3,3)=CZZ
C
C GENERATION OF THE TRANSPOSE OF LAMDA
C
        DO 10 I=1,3
        DO 10 J=1,3
  10    ALT(J,I)=AL(I,J)
C
        RETURN
        END
C
C
C ***************************
        SUBROUTINE TRNSMX
C ***************************
C
        PARAMETER(IMAXNN=231)
        COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $  GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
        COMMON/TMATX/AL(3,3),T(9,9),ALT(3,3),TT(9,9)
C
C GENERATION OF THE TRANSFORMATION MATRIX T
```

```
C
        DO 10 I=1,9
        DO 10 J=1,9
  10    T(I,J)=0.0
        DO 15 I=1,3
        DO 15 J=1,3
 15     T(I,J)=AL(I,J)
C
        DO 20 I=4,6
        DO 20 J=4,6
        IA=I-3
        JA=J-3
 20     T(I,J)=AL(IA,JA)
C
        DO 30 I=7,9
        DO 30 J=7,9
        IB=I-6
        JB=J-6
 30     T(I,J)=AL(IB,JB)
C
C GENERATION OF THE TRANSPOSE OF T
C
        DO 40 I=1,9
        DO 40 J=1,9
  40    TT(J,I)=T(I,J)
C
        RETURN
        END
C
C
C ****************************
        SUBROUTINE TRNSFM
C ****************************
C
        PARAMETER(IMAXNN=231)
        COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $  GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
        COMMON/TMATX/AL(3,3),T(9,9),ALT(3,3),TT(9,9)
        COMMON/DUMM/DUM1(9,9),DUM2(9,9),DUM3(3,3*IMAXNN),
     $  DUM4(3,3*IMAXNN),DUM5(3*IMAXNN,3),DUM6(3*IMAXNN,3)
C
C MATRIX MULTIPLICATION TO OBTAIN DUM2 = TT * ESM
C
        DO 10 I=1,9
        DO 10 J=1,9
        DUM2(I,J)=0.0
        DO 10 K=1,9
  10    DUM2(I,J)=DUM2(I,J)+TT(I,K)*ESM(K,J)
C
C MATRIX MULTIPLICATION TO OBTAIN GLOBAL ELEMENT STIFFNESS MATRIX
C
        DO 20 I=1,9
        DO 20 J=1,9
        GESM(I,J)=0.0
        DO 20 K=1,9
  20    GESM(I,J)=GESM(I,J)+DUM2(I,K)*T(K,J)
C
        RETURN
        END
C
C
C *****************************
        SUBROUTINE ASSEMB
C *****************************
C
        PARAMETER(IMAXNN=231)
        COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $  GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
        COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
C
C DIRECT STIFFNESS PROCEDURE
C
```

```
      DO 20 I=1,9
      IC=NS(I)
      DO 10 J=1,9
      JC=NS(J)
 10   GSM(IC,JC)=GSM(IC,JC)+GESM(I,J)
 20   CONTINUE
C
      RETURN
      END
C
C
C *********************************
      SUBROUTINE SKEWED(KN)
C *********************************
C
      PARAMETER(IMAXNN=231)
      COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $ GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
      COMMON/TMATX/AL(3,3),T(9,9),ALT(3,3),TT(9,9)
      COMMON/DUMM/DUM1(9,9),DUM2(9,9),DUM3(3,3*IMAXNN),
     $ DUM4(3,3*IMAXNN),DUM5(3*IMAXNN,3),DUM6(3*IMAXNN,3)
      COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
      COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
      COMMON/LAM/TEMP(3,3),ALN(3,3),ALNT(3,3)
      COMMON/PARAM/NY,NYP1,W
      DIMENSION N(12),NB(6),TEMP1(3,3),TEMP2(3,3)
C
C CHECK TO SEE IF NODE KN IS ON A BOUNDARY
C AND WHICH BOUNDARY IT IS ON
C
C ONE OF THE CORNERS
C
      IF(KN.EQ.1) THEN
        N1=KN+1
        N2=KN
        N3=KN+NYP1
        GO TO 200
      END IF
C
      IF(KN.EQ.NYP1) THEN
        N1=KN
        N2=KN-1
        N3=KN+NYP1
        GO TO 200
      END IF
C
      IF(KN.EQ.(NN-NY)) THEN
        N1=KN-NYP1
        N2=KN
        N3=KN+1
        GO TO 200
      END IF
C
      IF(KN.EQ.NN) THEN
        N1=KN-NYP1
        N2=KN-1
        N3=KN
        GO TO 200
      END IF
C
C FRONT EDGE
C
      IF((KN.GT.1).AND.(KN.LT.NYP1)) THEN
        NB(1)=KN
        NB(2)=KN+NYP1
        NB(3)=KN+1
        NB(4)=NB(1)
        NB(5)=KN-1
        NB(6)=NB(2)
        GO TO 400
      END IF
C
C RIGHT-HAND EDGE
```

```
C
          IF(YC(KN).EQ.0.) THEN
            NB(1)=KN
           NB(2)=KN+NYP1
            NB(3)=KN+1
            NB(4)=KN-NYP1
            NB(5)=KN
            NB(6)=KN+1
            GO TO 400
          END IF
C
C LEFT-HAND EDGE
C
          IF(YC(KN).EQ.W) THEN
            NB(1)=KN
            NB(2)=KN-1
            NB(3)=KN+NYP1
            NB(4)=KN-NYP1
            NB(5)=KN-1
            NB(6)=KN
            GO TO 400
          END IF
C
C REAR EDGE
C
          IF((KN.GT.(NN-NY)).AND.(KN.LT.NN)) THEN
            NB(1)=KN-NYP1
            NB(2)=KN-1
            NB(3)=KN
            NB(4)=KN-NYP1
            NB(5)=KN
            NB(6)=KN+1
            GO TO 400
          END IF
C
          GO TO 245
C
C RECALL NODAL COORDINATES
C
 200      X(1)=XC(N1)
          Y(1)=YC(N1)
          Z(1)=ZC(N1)
          X(2)=XC(N2)
          Y(2)=YC(N2)
          Z(2)=ZC(N2)
          X(3)=XC(N3)
          Y(3)=YC(N3)
          Z(3)=ZC(N3)
C
          KK=0
          CALL LAMDA(KK)
C
C GENERATION OF LOCAL NODAL LAMDA MATRIX
C FOR NODES ON ONE OF THE CORNERS
C
          DO 20 I=1,3
          DO 20 J=1,3
          AA=TEMP(I,J)
          ALN(I,J)=COS(AA)
          IF(ABS(ALN(I,J)).LE.0.00001) ALN(I,J)=0.0
          ALNT(J,I)=ALN(I,J)
  20      CONTINUE
C
          GO TO 300
C
C NODES NOT ON A BOUNDARY
C
 245      N(1)=KN
          N(2)=KN-1
          N(3)=KN+NYP1
          N(4)=KN
          N(5)=N(3)
```

```
      N(6)=KN+1
      N(7)=KN-NYP1
      N(8)=N(1)
      N(9)=KN+1
      N(10)=N(7)
      N(11)=N(2)
      N(12)=N(1)
C
      KK=0
      IFLAG=0
      DO 210 I=1,3
      DO 210 J=1,3
 210  TEMP2(I,J)=0.0
C
C CALCULATION OF AVERAGE NODAL LAMDA MATRIX
C FOR NODES ONT ON A BOUNDARY
C
      JK=1
      DO 45 PLNE=1,4
      DO 40 I=1,3
      X(I)=XC(N(JK))
      Y(I)=YC(N(JK))
      Z(I)=ZC(N(JK))
      JK=JK+1
  40  CONTINUE
      CALL LAMDA(KK)
      DO 50 I=1,3
      DO 50 J=1,3
  50  TEMP2(I,J)=TEMP2(I,J)+TEMP(I,J)*0.25
  45  CONTINUE
      GO TO 250
C
C CALCULATION OF AVERAGE NODAL LAMDA MATRIX
C FOR NODES ON ONE OF THE EDGES
C
C
 400  KK=0
      DO 401 I=1,3
      DO 401 J=1,3
 401  TEMP1(I,J)=0.0
C
      JJ=1
      DO 420 PLNE=1,2
      DO 410 I=1,3
      X(I)=XC(NB(JJ))
      Y(I)=YC(NB(JJ))
      Z(I)=ZC(NB(JJ))
      JJ=JJ+1
 410  CONTINUE
      CALL LAMDA(KK)
      DO 405 I=1,3
      DO 405 J=1,3
 405  TEMP1(I,J)=TEMP1(I,J)+TEMP(I,J)*0.5
 420  CONTINUE
      IFLAG=1
C
C GENERATION OF THE AVERAGE NODAL LAMDA MATRIX AND ITS TRANSPOSE
C
 250  DO 60 I=1,3
      DO 60 J=1,3
      IF(IFLAG.EQ.0) AB=TEMP2(I,J)
      IF(IFLAG.EQ.1) AB=TEMP1(I,J)
      ALN(I,J)=COS(AB)
      IF(ABS(ALN(I,J)).LE.0.00001) ALN(I,J)=0.0
      ALNT(J,I)=ALN(I,J)
  60  CONTINUE
C
C ******************
C       PRE-MULTIPLY ROWS BY LAMDA
C ******************
C
 300  DO 80 I=1,3
      IF(I.EQ.1) IROW=3*KN-2
```

```
        IF(I.EQ.2) IROW=3*KN-1
        IF(I.EQ.3) IROW=3*KN
        DO 80 J=1,NP
  80    DUM3(I,J)=GSM(IROW,J)
C
C MATRIX MULTIPLICATION TO OBTAIN DUM4 = ALN * DUM3
C
        DO 90 I=1,3
        DO 90 J=1,NP
        DUM4(I,J)=0.0
        DO 90 K=1,3
  90    DUM4(I,J)=DUM4(I,J)+ALN(I,K)*DUM3(K,J)
C
C CHANGE BACK TO GSM = DUM4
C
        DO 100 I=1,3
        IF(I.EQ.1) I2ROW=3*KN-2
        IF(I.EQ.2) I2ROW=3*KN-1
        IF(I.EQ.3) I2ROW=3*KN
        DO 100 J=1,NP
  100   GSM(I2ROW,J)=DUM4(I,J)
C
C ******************
C       POST-MULTIPLY COLUMNS BY TRANSPOSE OF LAMDA
C ******************
C
        DO 110 J=1,3
        IF(J.EQ.1) ICOL=3*KN-2
        IF(J.EQ.2) ICOL=3*KN-1
        IF(J.EQ.3) ICOL=3*KN
        DO 110 I=1,NP
  110   DUM5(I,J)=GSM(I,ICOL)
C
C MATRIX MULTIPLICATION TO OBTAIN DUM6 = DUM5 * ALNT
C
        DO 120 I=1,NP
        DO 120 J=1,3
        DUM6(I,J)=0.0
        DO 120 K=1,3
  120   DUM6(I,J)=DUM6(I,J)+DUM5(I,K)*ALNT(K,J)
C
C CHANGE BACK TO GTSM = DUM6
C
        DO 130 J=1,3
        IF(J.EQ.1) I2COL=3*KN-2
        IF(J.EQ.2) I2COL=3*KN-1
        IF(J.EQ.3) I2COL=3*KN
        DO 130 I=1,NP
  130   GSM(I,I2COL)=DUM6(I,J)
C
        DO 150 I=1,NP
        DO 150 J=1,NP
 150    GTSM(I,J)=GSM(I,J)
C
        RETURN
        END
C
C
C ***********************************
        SUBROUTINE MODIFY
C ***********************************
C
C
        PARAMETER(IMAXNN=231)
        COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $  GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
        COMMON/DOF/F(3*IMAXNN),U(3*IMAXNN),STRA(6),STRE(6),
     $  THM,TM,S1,S2
        COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
C
C ******************
C       INPUT OF THE NODAL FORCE VALUES FROM FILE FORCE.DAT
C ******************
```

```
C
C        NV - NODE NUMBER
C        IDOF - DEGREE OF FREEDOM OF THE FORCE
C        VF - VALUE OF THE FORCE      '
C
C        INPUT IS TERMINATED BY INPUTTING A ZERO VALUE FOR NV
C
C READ IN VALUES AND PLACE IN FORCE VECTOR {F}
C
C INTITIALIZE {F}
C
         DO 2 I=1,NP
  2      F(I)=0.0
C
  5      READ(12,*) NV,IDOF,VF
         IF(NV.EQ.0) GO TO 40
         F(IDOF)=VF
         GO TO 5
C
C
C ******************
C        INPUT OF THE PRESCRIBED NODAL VALUES AND CONSTRAINTS
C        FROM FILE CONSTR.DAT
C ******************
C
C        NV - NODE NUMBER
C        IDOF - DEGREE OF FREEDOM OF THE KNOWN DISPLACEMENT
C        VD - VALUE OF THE DISPLACEMENT
C
C        INPUT IS TERMINATED BY INPUTTING A ZERO VALUE FOR NV
C
C READ IN VALUES FROM CONSTR.DAT
C
  40     READ(14,*) NV,IDOF,VD
         IF(NV.EQ.0) GO TO 100
         U(IDOF)=VD
C
C ******************
C        MODIFICATION OF GLOBAL STIFFNESS MATRIX ¢GTSM
C        AND GLOBAL FORCE VECTOR
C ******************
C
C SET COEFFICIENTS OF ROW IDOF EQUAL TO ZERO
C
         DO 50 J=1,NP
         IF(J.EQ.IDOF) GO TO 50
         GTSM(IDOF,J)=0.0
  50     CONTINUE
C
C REPLACE IDOF COMPONENT IN FORCE VECTOR
C
         F(IDOF)=GTSM(IDOF,IDOF)*VD
         IF(GTSM(IDOF,IDOF).EQ.0.0) GTSM(IDOF,IDOF)=1.0
C
         DO 60 I=1,NP
         IF(I.EQ.IDOF) GO TO 60
         F(I)=F(I)-GTSM(I,IDOF)*VD
         GTSM(I,IDOF)=0.0
   60    CONTINUE
C
         GO TO 40
C
  100    RETURN
         END
C
C
C ******************************
         SUBROUTINE SANDS(KK)
C ******************************
C
         PARAMETER(IMAXNN=231)
         COMMON/GRAD/B(3,6),AR2
```

```
      COMMON/ELMATX/ESM(9,9),X(3),Y(3),Z(3),D(3,3),GESM(9,9),
     $ GSM(3*IMAXNN,3*IMAXNN),GTSM(3*IMAXNN,3*IMAXNN),IELR
      COMMON/DOF/F(3*IMAXNN),U(3*IMAXNN),STRA(6),STRE(6),
     $ THM,TM,S1,S2
      COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
      COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
      COMMON/LOCAL/XYZ(9),STOR(9),XL(3),YL(3),ZL(3),
     $ FL(3*IMAXNN)
C
C CALCULATION OF THE STRAIN VECTOR  STRAIN = B * U
C
      CALL ELSTMX(KK)
      DO 30 I=1,3
      STRA(I)=0.0
      DO 30 K=1,6
      KA=K
      IF((K.EQ.3).OR.(K.EQ.4)) KA=K+1
      IF((K.EQ.5).OR.(K.EQ.6)) KA=K+2
 30   STRA(I)=STRA(I)+B(I,K)*UU(KA)/AR2
C
C CALCULATION OF THE STRESS VECTOR  STRESS = D * STRAIN
C
      DO 40 I=1,3
      STRE(I)=0.0
      DO 40 K=1,3
 40   STRE(I)=STRE(I)+D(I,K)*STRA(K)
C
C CALCULATION OF THE PRINCIPAL STRESSES
C
      AA=(STRE(1)+STRE(2))/2.
      AB=SQRT(((STRE(1)-STRE(2))/2.)**2+STRE(3)**2)
      S1=AA+AB
      S2=AA-AB
      TM=AB
      IF(ABS(STRE(1)-STRE(2)).LT.0.001) GO TO 50
      AC=ATAN2(2.*STRE(3),STRE(1)-STRE(2))
      THM=((180./3.14159265)*AC)/2.
      GO TO 100
 50   THM=90.0
C
 100  RETURN
      END
C
C
C *************************************************
      SUBROUTINE FACTOR(W,IPIVOT,IFLAG)
C *************************************************
C
      PARAMETER(IMAXNN=231)
      COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
      INTEGER IPIVOT(3*IMAXNN)
      REAL D(3*IMAXNN),W(3*IMAXNN,3*IMAXNN)
C
C INPUTS
C   W - ARRAY CONTAINING THE MATRIX TO BE FACTORED
C   N - ORDER OF THE MATRIX
C
      N=NP
      IFLAG=1
C
C INITIALIZE IPIVOT,D
C
      DO 9 I=1,N
      IPIVOT(I)=I
      ROWMAX=0.
      DO 5 J=1,N
 5    ROWMAX=AMAX1(ROWMAX,ABS(W(I,J)))
      IF(ROWMAX.EQ.0.) THEN
      IFLAG=0
      ROWMAX=1.
      END IF
 9    D(I)=ROWMAX
      IF(N.LE.1) GO TO 30
```

```
C
C FACTORIZATION
C
        DO 20 K=1,N-1
C
C DETERMINE PIVOT ROW, THE ROW ISTAR
C
        COLMAX=ABS(W(K,K))/D(K)
        ISTAR=K
        DO 13 I=K+1,N
        AWIKOD=ABS(W(I,K))/D(I)
        IF(AWIKOD.GT.COLMAX) THEN
        COLMAX=AWIKOD
        ISTAR=I
        END IF
  13    CONTINUE
        IF(COLMAX.EQ.0.) THEN
        IFLAG=0
        ELSE
        IF(ISTAR.GT.K) THEN
C
C MAKE K THE PIVOT ROW BY INTERCHANGING IT WITH
C THE CHOSEN ROW ISTAR
C
        IFLAG=-IFLAG
        I=IPIVOT(ISTAR)
        IPIVOT(ISTAR)=IPIVOT(K)
        IPIVOT(K)=I
        TEMP=D(ISTAR)
        D(ISTAR)=D(K)
        D(K)=TEMP
        DO 15 J=1,N
        TEMP=W(ISTAR,J)
        W(ISTAR,J)=W(K,J)
  15    W(K,J)=TEMP
        END IF
C
C ELIMINATE X(K) FROM ROWS K=1,....,N
C
  16    DO 19 I=K+1,N
        W(I,K)=W(I,K)/W(K,K)
        RATIO=W(I,K)
        DO 19 J=K+1,N
        W(I,J)=W(I,J)-RATIO*W(K,J)
  19    CONTINUE
        END IF
  20    CONTINUE
        IF(W(N,N).EQ.0.) IFLAG=0
C
  30    RETURN
        END
C
C
C *******************************************
        SUBROUTINE SUBST(W,IPIVOT,B,X)
C *******************************************
C
        PARAMETER(IMAXNN=231)
        COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
        INTEGER IPIVOT(3*IMAXNN)
        REAL B(3*IMAXNN),X(3*IMAXNN),W(3*IMAXNN,3*IMAXNN)
C
        N=NP
        IF(N.LE.1) THEN
        X(1)=B(1)/W(1,1)
        RETURN
        END IF
C
        IP=IPIVOT(1)
        X(1)=B(IP)
        DO 15 I=2,N
        SUM=0.
```

```
      DO 14 J=1,I-1
 14   SUM=W(I,J)*X(J)+SUM
      IP=IPIVOT(I)
 15   X(I)=B(IP)-SUM
C
      X(N)=X(N)/W(N,N)
      DO 20 I=N-1,1,-1
      SUM=0.
      DO 19 J=I+1,N
 19   SUM=W(I,J)*X(J)+SUM
 20   X(I)=(X(I)-SUM)/W(I,I)
C
      RETURN
      END
C
C
C ******************************************
      SUBROUTINE CHKSLP(KN,IFLAG1)
C ******************************************
C
      PARAMETER(IMAXNN=231)
      COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
      COMMON/DOF/F(3*IMAXNN),U(3*IMAXNN),STRA(6),STRE(6),THM,
     $ TM,S1,S2
      COMMON/PARAM/NY,NYP1,W
      COMMON/PARAM2/NNB,NNO
      COMMON/ANGL/SLOPE,FRICTF(IMAXNN),FEXT(IMAXNN),ADIV(IMAXNN),
     $ CONST(IMAXNN)
      DIMENSION XCORD(IMAXNN)
C
      IFLAG1=1
C
      XCORD(KN)=XC(KN)
      XCORD(KN-1)=XC(KN-1)
      A=XCORD(KN)+U(3*KN-2)
      B=XCORD(KN-1)+U(3*(KN-1)-2)
C
      IF(A.LT.B) IFLAG1=0
C
C     ANUMER=YC(KN-NYP1)+U(3*(KN-NYP1)-1)-YC(KN)-U(3*KN-1)
C     ADENOM=XC(KN)+U(3*KN-2)-XC(KN-NYP1)-U(3*(KN-NYP1)-2)
C     SLOPE=ANUMER/ADENOM
C
      XCORD(KN)=XC(KN)+U(3*KN-2)
      XCORD(KN-1)=XC(KN-1)+U(3*(KN-1)-2)
C
      RETURN
      END
C
C
C ************************************
      SUBROUTINE FRICTN(KN)
C ************************************
C
      PARAMETER(IMAXNN=231)
      COMMON/MTL/EM,PR,TH,AMU
      COMMON/LOCAL/XYZ(9),STOR(9),XL(3),YL(3),ZL(3),
     $ FL(3*IMAXNN)
      COMMON/ANGL/SLOPE,FRICTF(IMAXNN),FEXT(IMAXNN),ADIV(IMAXNN),
     $ CONST(IMAXNN)
C
C CALCULATE FRICTION FORCE
C
      FRICTF(KN)=FL(3*KN)*AMU
C
      RETURN
      END
C
C
```

```
C ***********************************
        SUBROUTINE APFORS(KN,ITER)
C ***********************************
C
        PARAMETER(IMAXNN=231)
        COMMON/MTL/EM,PR,TH,AMU
        COMMON/DOF/F(3*IMAXNN),U(3*IMAXNN),STRA(6),STRE(6),THM,
     $  TM,S1,S2
        COMMON/ANGL/SLOPE,FRICTF(IMAXNN),FEXT(IMAXNN),ADIV(IMAXNN),
     $  CONST(IMAXNN)
C
C APPLY SPREADING FORCE TO NODE KN IN Y-DIRECTION
C
        AITER=FLOAT(ITER)
        IDOF=3*KN-1
        IF(ITER.EQ.1) FEXT(KN)=0.0
        FEXT(KN)=FEXT(KN)+(1./AITER)*(FRICTF(KN)-FEXT(KN))
        F(IDOF)=FEXT(KN)
C
        RETURN
        END
C
C
C ******************************************
        SUBROUTINE CHKFOR(KN,IFLAG2)
C ******************************************
C
        PARAMETER(IMAXNN=231)
        COMMON/LOCAL/XYZ(9),STOR(9),XL(3),YL(3),ZL(3),
     $  FL(3*IMAXNN)
        COMMON/ANGL/SLOPE,FRICTF(IMAXNN),FEXT(IMAXNN),ADIV(IMAXNN),
     $  CONST(IMAXNN)
C
        IFLAG2=1
C
C CHECK TO SEE IF THE RATIO OF SPREADING FORCE TO FRICTION FORCE
C IS WITHIN THE SPECIFIED LIMITS
C
        ADIV(KN)=FEXT(KN)/FRICTF(KN)
        IF((ADIV(KN).LT.0.9800).OR.(ADIV(KN).GT.1.02)) IFLAG2=0
C
        RETURN
        END
C
C
C ***************************
        SUBROUTINE PLOT
C ***************************
C
        PARAMETER(IMAXNN=231)
        COMMON/N/NP,NN,NE,NS(9),ICK(500),NS1,NS2,UU(9)
        COMMON/ELM/NEL(500,3),XC(IMAXNN),YC(IMAXNN),ZC(IMAXNN)
        COMMON/DOF/F(3*IMAXNN),U(3*IMAXNN),STRA(6),STRE(6),THM,
     $  TM,S1,S2
        COMMON/PARAM/NY,NYP1,W
        COMMON/PARAM2/NNB,NNO
        DIMENSION XDEF(IMAXNN),YDEF(IMAXNN),X(50)
C
        VALUE=1000.0
C
C WRITE 'BEFORE ROLLER' MESH DATA TO FILE BEFORE.DAT
C
C UNDEFORMED MESH
C
        WRITE(9,*) NYP1
        DO 10 I=1,NNB
        WRITE(9,15) XC(I),YC(I)
 15     FORMAT(F12.5,2X,F12.5)
        IF(I.EQ.NNB) WRITE(9,15) VALUE,VALUE
 10     CONTINUE
```

```
C
C DEFORMED MESH
C
        DO 20 I=1,NNB
        XDEF(I)=XC(I)+10.*U(3*I-2)
        YDEF(I)=YC(I)+10.*U(3*I-1)
        WRITE(9,15) XDEF(I),YDEF(I)
        IF(I.EQ.NNB) WRITE(9,15) VALUE,VALUE
 20     CONTINUE
C
C WRITE 'ON ROLLER' MESH DATA TO FILE ON.DAT
C
C UNDEFORMED MESH
C
        WRITE(11,*) NYP1
        ALEN=4.*W
        DX=ALEN/10.
        X(1)=0.0
        DO 30 I=2,11
 30     X(I)=X(I-1)+DX
C
        II=0
        DO 35 I=NNB+1,NNB+NNO
        IREM=MOD((I-1),NYP1)
        IF(IREM.EQ.0) II=II+1
        WRITE(11,15) X(II),YC(I)
        IF(I.EQ.NNB+NNO) WRITE(11,15) VALUE,VALUE
 35     CONTINUE
C
C DEFORMED MESH
C
        II=0
        DO 40 I=NNB+1,NNB+NNO
        IREM=MOD((I-1),NYP1)
        IF(IREM.EQ.0) II=II+1
        XDEF(I)=X(II)+10.*U(3*I-2)
        YDEF(I)=YC(I)+10.*U(3*I-1)
        WRITE(11,15) XDEF(I),YDEF(I)
        IF(I.EQ.NNB+NNO) WRITE(11,15) VALUE,VALUE
 40     CONTINUE
C
C WRITE 'AFTER MESH' DATA TO FILE AFTER.DAT
C
C UNDEFORMED MESH
C
        WRITE(16,*) NYP1
        DO 50 I=NNB+NNO+1,NN
        WRITE(16,15) ABS(ZC(I)),YC(I)
        IF(I.EQ.NN) WRITE(16,15) VALUE,VALUE
 50     CONTINUE
C
C DEFORMED MESH
C
        DO 60 I=NNB+NNO+1,NN
        XDEF(I)=ABS(ZC(I))+10.*U(3*I-2)
        YDEF(I)=YC(I)+10.*U(3*I-1)
        WRITE(16,15) XDEF(I),YDEF(I)
        IF(I.EQ.NN) WRITE(16,15) VALUE,VALUE
 60     CONTINUE
C
        RETURN
        END
```

```
C ************************************************************
C
C                        PROGRAM PLOT.FOR
C
C ************************************************************
C
C IN PARAMETER STATEMENT :
C     IN=9  PLOTS MESH BEFORE ROLLER
C     IN=11 PLOTS MESH ON ROLLER
C     IN=16 PLOTS MESH AFTER ROLLER
C
C
      PARAMETER(IN=11)
C
      DIMENSION X(500),Y(500),XD(500),YD(500)
C
C READ MESH DATA FROM FILE
C
      ICOUNT=1
      READ(IN,*) NYP1
 5    READ(IN,*) X(ICOUNT),Y(ICOUNT)
      IF(X(ICOUNT).EQ.1000.0) GO TO 10
      ICOUNT=ICOUNT+1
      GO TO 5
C
C FIND MINIMUM AND MAXIMUM COORDINATE VALUES FOR SCALING
C
 10   XMIN=X(1)
      XMAX=X(1)
      YMIN=Y(1)
      YMAX=Y(1)
      DO 15 I=2,ICOUNT-1
      XMIN=AMIN1(XMIN,X(I))
      YMIN=AMIN1(YMIN,Y(I))
      XMAX=AMAX1(XMAX,X(I))
      YMAX=AMAX1(YMAX,Y(I))
 15   CONTINUE
C
C DEFINE WINDOW BOUNDARIES
C
      CALL INITT(120)
      CALL TWINDO(0,1023,0,780)
      CALL DWINDO(XMIN,XMAX*1.5,YMIN,YMAX*1.50)
C
C DRAW UNDEFORMED MESH
C
      DO 20 I=1,ICOUNT-NYP1-1
      IREM=MOD(I,NYP1)
      IF(IREM.EQ.0) GO TO 20
      IA=I
      IB=I+1
      IC=IB+NYP1
      ID=IC-1
      CALL RECT(X(IA),Y(IA),X(IB),Y(IB),X(IC),Y(IC),X(ID),Y(ID))
 20   CONTINUE
C
C DRAW DEFORMED MESH
C
      JCOUNT=1
 25   READ(IN,*) XD(JCOUNT),YD(JCOUNT)
      IF(XD(JCOUNT).EQ.1000.0) GO TO 30
      JCOUNT=JCOUNT+1
      GO TO 25
C
 30   DO 35 I=1,JCOUNT-NYP1-1
      IREM=MOD(I,NYP1)
      IF(IREM.EQ.0) GO TO 35
      IA=I
      IB=I+1
      IC=IB+NYP1
      ID=IC-1
      CALL RECT(XD(IA),YD(IA),XD(IB),YD(IB),XD(IC),YD(IC),
     $ XD(ID),YD(ID))
 35   CONTINUE
```

```
C
      CALL FINITT(0,760)
C
      STOP
      END
C
C
C **************************************************
      SUBROUTINE RECT(XA,YA,XB,YB,XC,YC,XD,YD)
C **************************************************
C
      CALL MOVEA(XA,YA)
      CALL DRAWA(XB,YB)
      CALL DRAWA(XC,YC)
      CALL DRAWA(XD,YD)
      CALL DRAWA(XA,YA)
C
      RETURN
      END
```

VITA

Michael Louis Leport

Candidate for the Degree of

Master of Science

Thesis: THE MECHANICS OF WEBS ENCOUNTERING CONCAVE ROLLS

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born on February 27, 1963, in Tulsa, Oklahoma, the son of Louis and Mary Jo Leport.

Education: Graduated from Bishop Kelley High School, Tulsa, Oklahoma, in May, 1981; received the Bachelor of Science in Mechanical Engineering degree from Oklahoma State University in July, 1985; completed the requirements for the Master of Science degree at Oklahoma State University, May, 1987.

Professional Experience: Teaching Assistant, Oklahoma State University, Stillwater, Oklahoma, 1985-1986.

Professional Societies: American Society of Mechanical Engineers.