

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Technical

F-C2762

Report

FINITE ELEMENT SOLUTION
OF
LUBRICATION PROBLEMS

by

M. M. Reddi

Prepared for

NASA-Lewis Research Center
Cleveland, Ohio 44135



N72-14487

(NASA-CR-125069) FINITE ELEMENT SOLUTION
OF LUBRICATION PROBLEMS Technical Report,
18 Feb. 1970 - 18 Feb. 1971 M. M. Reddi
(Franklin Inst.) 18 Feb. 1971 65 p
CSCL 11H (CATEGORY)

Unclas
11973

G3/15



THE FRANKLIN INSTITUTE RESEARCH LABORATORIES

Technical

F-C2762

Report

FINITE ELEMENT SOLUTION
OF
LUBRICATION PROBLEMS

by

M. M. Reddi

Prepared for

NASA-Lewis Research Center
Cleveland, Ohio 44135



THE FRANKLIN INSTITUTE RESEARCH LABORATORIES
BENJAMIN FRANKLIN PARKWAY • PHILADELPHIA, PENNSYLVANIA 19106

F-C2762

PREFACE

This report summarizes work done under Contract No. NAS3-13480, for NASA-Lewis Research Center, during the period 2-18-70 to 2-18-71. Mr. B. J. Hamrock acted as the Project Leader for NASA-Lewis.

The work was performed at the Franklin Institute Research Laboratories as Project No. 32G-C2762-01 through 06 with Mr. W. J. Shapiro acting as the Business Manager and Dr. M. M. Reddi as the Principal Investigator.

F-C2762

ABSTRACT

A variational formulation of the transient lubrication problem is presented and the corresponding finite element equations derived for three and six point triangles, and, four and eight point quadrilaterals. Test solutions for a one dimensional slider bearing used in validating the computer program 'FELUB' are given. Utility of the method is demonstrated by a solution of the shrouded step bearing.

Input map, input preparation instructions and program description of 'FELUB' computer program are presented.



TABLE OF CONTENTS

<u>Section</u>	<u>Title</u>	<u>Page</u>
	PREFACE	i
	ABSTRACT	ii
	NOMENCLATURE	v
1	INTRODUCTION	1-1
2	FINITE ELEMENT FORMULATION	2-1
	2.1 The Compressible Lubrication Problem.	2-1
	2.2 Variational Formulation	2-3
	2.3 Temporal Discretization	2-5
	2.4 Spatial Discretization.	2-7
	2.5 Triangular Elements	2-10
	2.6 Quadrilateral Elements.	2-13
	2.7 Boundary Conditions	2-17
	2.8 Solution Scheme	2-18
3	PROGRAM DESCRIPTION.	3-1
	3.1 Introduction.	3-1
	3.2 Major Program Variables	3-2
	3.3 Subroutine Descriptions	3-4
	3.4 Input Preparation	3-12
	3.5 Validation.	3-25
	3.6 Shrouded Step Bearing	3-29
4	CONCLUSIONS.	4-1
5	REFERENCES	5-1

LIST OF FIGURES

<u>Fig. No.</u>	<u>Title</u>	<u>Page</u>
2.5.1	Triangular Elements.	2-11
2.6.1	Quadrilateral Elements	2-14
3.2.1	General Flow Chart for FELUB	3-6
3.4.1	Cylindrical and Conical Coordinate Systems	3-19
3.4.2	Use of ILINE and NMID.	3-21
3.5.1	Comparison of Transient Solutions, Computed and Exact.	3-26
3.5.2	Steady State Solution for One-Dimensional Slider Bearing, $\Lambda = 10$	3-27
3.5.3	Steady State Solution for One-Dimensional Slider Bearing, $\Lambda = 50$	3-28
3.6.1	Shrouded Step Bearing Configuration.	3-30
3.6.2	Polar, Shrouded Step Bearing Mesh.	3-31
3.6.3	Rectangular Shrouded Step Bearing Mesh	3-32
3.6.4	Pad Geometry	3-34
3.6.5	Polar Velocity Field	3-35
3.6.6	Steady State Pressure Distribution for Rectangular Shrouded Step Bearing.	3-36

NOMENCLATURE

- B = Typical bearing dimension (in.)
 C_1 = Pressure prescribed boundary segment
 C_2 = Flow prescribed boundary segment
 g = Coordinate (in.)
 H = Dimensionless film thickness (h/h_o)
 h = Film thickness (in.)
 h_o = Typical film thickness (in.)
 $\hat{n}$ = Unit outer normal
 P = Dimensionless pressure (p/p_a)
 p = Pressure (lb. in.⁻²)
 $\bar{p}$ = Boundary pressure (lb. in.⁻²)
 p_a = Ambient pressure (lb. in.⁻²)
 p_o = Initial pressure (lb. in.⁻²)
 Q = Dimensionless, lineal mass flow (see eq. 2.1.6)
 q = Lineal mass flow (lb. sec.⁻¹)
 R = Interior of lubricant region
 $\bar{R}$ = Closed and bounded lubricant region
 T = Dimensionless time (wt)
 $\hat{U}$ = Surface velocity vector (in. sec.⁻¹)
 u = Viscosity (lb. sec. in.⁻²)
 w = Typical frequency (Hz.)



F-C2762

- X = Dimensionless coordinate (x/B)
- x = Coordinate (in.)
- Y = Dimensionless coordinate (y/B)
- Λ = Compressibility parameter, dimensionless (see eq. 2.1.4)
- g = Squeeze number, dimensionless (see eq. 2.1.4)

1. INTRODUCTION

With numerical procedures becoming increasingly important in lubrication analysis, both in theoretical studies and practical bearing design, the need for more flexible and unified techniques becomes more and more apparent. While finite difference techniques have been almost universally employed in this area, a review of which may be found in [1], two recent studies [2, 3] utilizing the finite element method provide an alternative approach. One of the objectives of the present work is to extend the recent finite element studies to the solution of the general, transient lubrication problem.

A major advantage of the finite element method is that it allows the analyst to discretize the spatial domain in exactly the way he chooses, that is, with more nodes in areas where pressure gradient undergoes rapid changes and less, in other areas. In addition, it encompasses all lubrication problems governed by Reynolds' equation within the framework of one approach. Thus, no special treatment is accorded to various analyses just because of discontinuities, differing boundary configurations, etc.

Another objective of the present work is to develop a general purpose computer program that will allow analysis of general lubrication and seal problems governed by the Reynolds' equation, be they transient or steady-state, compressible or incompressible, hydrodynamic or hybrid. The usefulness of such a program in the routine analysis of non-routine designs cannot be over stated.

A less obvious but more significant utility of such a program lies in the eventual possibility of obtaining true automation in optimized bearing and seal design. Indeed, with the availability of the general purpose program described in this report and a library of

F-C2762

subroutines for generating the geometries of several bearing configurations, the techniques of linear and non-linear mathematical programming can be readily applied within the current state-of-the-art to bring about complete automation in optimizing lubrication and seal systems. It is hoped that such an application will become a reality in the none too distant future.

While the theoretical development for the finite element equations of the transient lubrication problem is new, the remaining aspects of the theory required for the general purpose computer program is not and the reader is directed to references [2, 3] for pertinent developments of those areas and additional information relating to the advantages of the finite element method as well as its application to lubrication problems.



2. FINITE ELEMENT FORMULATION

2.1 The Compressible Lubrication Problem

Let $\bar{R}$ denote a closed and bounded lubricant region with the interior R and boundary $\bar{B}$. Let (x,y) be a generic point of $\bar{R}$. The compressible lubrication problem consists of finding a solution $p(x,y,t)$ on $\bar{R}$ of the isothermal Reynolds' equation

$$\nabla \cdot (h^3 \nabla p - 6\mu \underline{U} ph) = 12\mu (ph)_{,t} \quad (2.1.1)$$

with the boundary conditions

$$p = \bar{p} \text{ on } C_1$$

$$p(x,y,0) = p_0 \text{ on } R \quad (2.1.2)$$

and

$$\underline{n} \cdot (Uh - h^3 \nabla p / 6\mu)p = 2q \text{ on } C_2 \quad (2.1.3)$$

where, $\underline{n}$ is the unit outer normal to C_2 , $\bar{B}$ is $C_1 + C_2$ and q is the linear mass flow across the boundary C_2 . In addition, μ is the viscosity of the lubricant, $\underline{U}$ is the vector velocity of the moving surface, $h(x,y,t)$ is the film thickness measured normal to the moving surface and a subscript comma denotes differentiation. p_0 is the initial value of p .

Recalling the derivation of Reynolds' equation, we note that equation 2.1.1 is valid only in those cases where the moving surface is normal everywhere to one of the coordinate directions of a coordinate system in three dimensional Euclidean space; in other words, equation 2.1.1 is not a general equation applicable to all surfaces.

Equation 2.1.1 and the associated boundary and initial conditions 2.1.2 and 2.1.3 may be put in a dimensionless form by the following transformations in a cartesian coordinate system.

$$\begin{aligned}
 X &= x/B \\
 Y &= y/B \\
 T &= \omega t \\
 P &= p/p_a \\
 H &= h/h_o \\
 \Lambda &= 6\mu \bar{U} B / (p_a h_o^2) \\
 \sigma &= 12\mu B^2 \omega / (p_a h_o^2)
 \end{aligned} \tag{2.1.4}$$

where B , h_o and ω are typical bearing dimension, film thickness and frequency, respectively; and, p_a is the ambient pressure. If no typical frequency exists in a particular problem, ω may be conveniently set equal to 1.

With the transformations 2.1.4, the compressible lubrication problem becomes: Find a solution $P(X,Y,T)$ on $\bar{R}$ such that

$$\nabla \cdot (H^3 \nabla P) - \Lambda (PH) = \sigma (PH)_{,T} \tag{2.1.5}$$

$$P = \bar{p}/p_a \text{ on } C_1$$

$$P(X,Y,0) = p_o/p_a \text{ on } R$$

and

$$\underline{n} \cdot (\Lambda H - H^3 \nabla P)P = Q \text{ on } C_2$$

where, the dimensionless lineal mass flow Q is given by

$$Q = 12\mu Bq/h_o^3 p_a^2 \tag{2.1.6}$$

In previous studies [1,4] on the numerical solution of the transient, compressible lubrication problem, use of PH^n ($n = 1$ to 3) as the dependent variable has been argued to be advantageous, because gradients of PH are, in general, smaller than those of P , and, in addition, knowledge of $H_{,T}$ is not required. While these considerations may or may not hold for the finite difference method, at least the former is of little significance in applying the finite element method because magnitude of the gradient by itself is of no consequence in determining the solution accuracy. Rather, it is the spatial rate of change of gradient that is significant, because, it determines the gradation required in the element sizes for achieving an optimal solution from the viewpoint of accuracy vs. number of elements utilized in the solution. Of more fundamental importance in applying the finite element method, however, is the requirement for the dependent variable to be of class D_1 , namely, that

1. $p(x,y)$ is continuous for all $(x,y) \in \bar{R}$
2. $p_{,x}$ and $p_{,y}$ are continuous for all $(x,y) \in \bar{R}$, except at a finite number of points at which one-sided derivatives, not necessarily equal, exist.

In general, this requirement cannot be met by PH as the dependent variable, particularly in step and grooved bearings.

Thus, the subsequent developments in the present study are based on the use of P alone as the dependent variable. While this places a requirement on knowledge of $H_{,T}$, it may not be a major inconvenience in most applications.

2.2 Variational Formulation

The time dependent compressible lubrication problem defined by equations 2.1.5 may be put in a variational form: Using the customary notation of the calculus of variations, and defining an admissible function as a continuous function that satisfies prescribed boundary

and initial conditions, let F be the set of all admissible functions P . Let $\delta\Phi\{\cdot\}$ on F be given by

$$\begin{aligned} \delta\Phi\{P\} = & \int_{T_1}^{T_2} \int_R (PH^3 \nabla P \cdot \nabla \delta P - PH \underline{\Lambda} \cdot \nabla \delta P + \sigma PH \delta P_{,T}) dR \\ & + \int_{C_2} Q \delta P ds \} dT - \sigma \int_R (PH \delta P)_{T=T_2} dR \end{aligned} \quad (2.2.1)$$

Then,

$$\delta\Phi\{P\} = 0 \text{ over } F \quad (2.2.2)$$

at a particular function P if and only if P is a solution of the compressible lubrication problem.

A proof for the foregoing is as follows: Let $P + \delta P \in F$. It then follows that

$$\delta P(x, y, t) = 0 \text{ on } C_1$$

$$\delta P(x, y, 0) = 0 \text{ on } R \quad (2.2.3)$$

and in addition, by virtue of Green's Theorem,

$$\begin{aligned} \delta\Phi\{P\} = & 1 \int_{T_1}^{T_2} \left\{ \int_R [\nabla \cdot (PH^3 \nabla P - PH \underline{\Lambda}) - \sigma (PH)_{,T}] \delta P dR \right. \\ & \left. + \int_C \underline{n} \cdot (PH^3 \nabla P - PH \underline{\Lambda}) \delta P ds + \int_{C_2} Q \delta P ds \right\} dt \end{aligned} \quad (2.2.4)$$

First, if P is a solution of the compressible lubrication problem, then P satisfies equations 2.1.5 and hence, through 2.2.4, equation 2.2.2 holds. Conversely, if P satisfies equation 2.2.2, then equation 2.2.4 implies

$$\int_{T_1}^{T_2} \int_R [\nabla \cdot (PH^3 \nabla P - \Lambda PH) - \sigma (PH)_{,T}] \delta P dR dt = 0 \quad (2.2.5)$$

and

$$\int_{T_1}^{T_2} \int_{C_2} \left\{ \underline{n} \cdot (PH^3 \nabla P - PH \underline{\Lambda}) + Q \right\} \delta P ds = 0 \quad (2.2.6)$$

for every P that satisfies 2.2.2. By virtue of the fundamental lemma of the calculus of variations, P then satisfies equations 2.1.5.

2.3 Temporal Discretization

Let the time dependence of P be assumed to be of the form

$$P = P_1 + \phi P_2 \quad (2.3.1)$$

where P_1 and P_2 are spatial functions and ϕ is a temporal function in the interval (T_1, T_2) and defined as

$$\phi = (T - T_1)/(T_2 - T_1) \quad (2.3.2)$$

While it is not necessary, for convenience of avoiding numerical integration in the time domain, one may suppose that $\underline{\Lambda}$, H and Q are also of the form

$$\begin{aligned} \underline{\Lambda} &= \underline{\Lambda}_1 + \phi \underline{\Lambda}_2 \\ H &= H_1 + \phi H_2 \\ Q &= Q_1 + \phi Q_2 \end{aligned} \quad (2.3.3)$$

This will be a reasonable supposition provided that (T_1, T_2) is a small enough interval. Moreover, if the interval is small enough, $P_2 \ll P_1$.

Noting that

$$\int_{T_1}^{T_2} \phi^j dT = (T_2 - T_1)/(j + 1) \quad (2.3.4)$$

and

$$\delta P = \phi \delta P_2 \quad (2.3.5)$$

implying that P_1 is a solution of the compressible lubrication problem at time $T = T_1$, equation 2.2.1, with the aid of equations 2.3.1 through 2.3.5, yields

$$\begin{aligned} \delta \phi\{P\} = & \int_R \left\{ P_1 f_1 \nabla P_1 \cdot \nabla \delta P_2 + f_2 (P_1 \nabla P_2 + P_2 \nabla P_1) \cdot \nabla \delta P_2 + f_3 P_2 \nabla P_2 \cdot \nabla \delta P_2 \right. \\ & \left. + P_1 d_1 \cdot \nabla \delta P_2 + P_2 d_2 \cdot \nabla \delta P_2 + \alpha P_1 \delta P_2 + \beta P_2 \delta P_2 \right\} dR \\ & + \int_{C_2} \left(\frac{1}{2} Q_1 + \frac{1}{3} Q_2 \right) \delta P_2 ds \end{aligned} \quad (2.3.6)$$

where,

$$f_1 \equiv (H_1^3/2) + H_1^2 + H_1^2 H_2 + (3H_1 H_2^2/4) + (H_2^3/5)$$

$$f_2 \equiv (H_1^3/3) + (3H_1^2 H_2/4) + (3H_1 H_2^2/5) + (H_2^3/6)$$

$$f_3 \equiv (H_1^3/4) + (3H_1^2 H_2/5) + (H_1 H_2^2/2) + (H_2^3/7)$$

$$\alpha \equiv \sigma P_1 H_2 / (2\Delta T)$$

$$\beta \equiv \sigma (H_1 + 4H_2/3) / (2\Delta T)$$

$$d_1 = -[(H_1 \Lambda_1/2) + (H_1 \Lambda_2/3) + (H_2 \Lambda_1/3) + (H_2 \Lambda_2/4)]$$

$$d_2 = -[(H_2 \Lambda_1/4) + (H_2 \Lambda_2/5) + (H_1 \Lambda_1/3) + (H_1 \Lambda_2/4)]$$

$$\Delta T = T_2 - T_1 \quad (2.3.7)$$

Inasmuch as $f_3 < f_2$ and $P_2 \ll P_1$, in comparison with $P_1 f_1 \nabla P_1$, $P_2 f_3 \nabla P_2$ may be neglected. Indeed if P_1 is $O(1)$ and P_2 (10^{-4}), error resulting from this approximation is of the same order as the round-off error with eight digit, finite arithmetic.

With the foregoing approximation, the Euler-Lagrange equation of 2.3.6 is

$$\begin{aligned} \nabla \cdot [P_1 f_2 \nabla P_2 + P_2 f_2 \nabla P_1 + P_1 f_1 \nabla P_1 - P_1 d_1 - P_2 d_2] \\ = \sigma [P_1 H_2 + P_2 H_1 + 4 P_2 H_2/3]/(2\Delta t) \end{aligned} \quad (2.3.8)$$

Alternatively, equation 2.3.8 can be derived from 2.1.5 with the assumed functional form 2.3.1 by applying Galerkin's Principle and neglecting $P_2 f_3 \nabla P_2$ in comparison with $P_1 f_1 \nabla P_1$.

2.4 Spatial Discretization

The procedure for discretizing the continuum problem into finite element equations is described in [2,3]. Let the spatial region of interest be subdivided into M elements of J nodes each. For a typical element E_m , the pressure field P_{2m} may be assumed as

$$P_{2m}(x,y) = \underline{f}^T \underline{a} \quad (2.4.1)$$

where, $\underline{a}$ is a column vector of J constants and $\underline{f}$ is a column vector of J interpolation functions defining the pressure within the element. The constants $\underline{a}$ are related to the unknown nodal pressures and are determined by evaluating 2.4.1 at the nodes.

Thus,

$$\underline{P}_{2m} = F \underline{a} \quad (2.4.2)$$

where F is the matrix of interpolation functions evaluated at the nodes and $\underline{P}_{2m}$ is the column vector of nodal pressures.

From 2.4.2,

$$\underline{a} = F^{-1} \underline{P}_{2m} \quad (2.4.3)$$

and 2.4.1 may be expressed as

$$P_{2m}(x,y) = \underline{t}_m^T \underline{P}_{2m} \quad (2.4.4)$$

where

$$\underline{t}_m^T = \underline{f}^T F^{-1} \quad (2.4.5)$$

Augmenting $\underline{t}_m$ with zero elements such that the non-zero elements correspond to the nodes of element m , a new column vector $\underline{T}_m$ is defined. Equation 2.4.4 may now be put in the form

$$\underline{P}_{2m} = \underline{T}_m^T \underline{P}_2 \quad (2.4.6)$$

where $\underline{P}_2$ is now the column vector of nodal pressures for the entire finite element assemblage.

A column vector of pressure gradients $\underline{g}_m$ may be conveniently defined as

$$\underline{g}_m = \underline{B}_m \underline{P}_2 \quad (2.4.7)$$

where,

$$\underline{B}_m = \langle \underline{T}_{m,x}^T \quad \underline{T}_{m,y}^T \rangle^T \quad (2.4.8)$$

The variational expression 2.3.6 may now be expressed in matrix form as

$$\delta\phi\{P\} = \sum_{m=1}^m \int_{R_m} \delta P_2^T \left\{ [B_m^T C_m B_m + B_m^T D_m M_m + \beta T_m T_m^T] P_2 + B_m^T E_m + \alpha T_m \right\} dA_m + \delta P_2^T \int_{C_{2m}} T_m \gamma ds_m \quad (2.4.9)$$

where,

$$M_m = \begin{bmatrix} T_m^T \\ T_m^T \\ T_m^T \end{bmatrix}$$

$$C_m \equiv \text{DIAG} (P_1 f_2, P_1 f_2)$$

$$D_m \equiv \text{DIAG} (P_{1,x} f_2 + u_2, P_{1,y} f_2 + v_2)$$

$$E_m \equiv \langle (P_{1,x} f_1 + u_1) P_1, (P_{1,y} f_1 + v_1) P_1 \rangle^T$$

$$\gamma \equiv (Q_1/2) + (Q_2/3)$$

$$u_1 \equiv \underline{d}_1 \cdot \underline{i}_x$$

$$u_2 \equiv \underline{d}_2 \cdot \underline{i}_x$$

$$v_1 \equiv \underline{d}_1 \cdot \underline{i}_y$$

$$v_2 \equiv \underline{d}_2 \cdot \underline{i}_y \quad (2.4.10)$$

where $\underline{i}_x$ and $\underline{i}_y$ are unit vectors in the x and y directions respectively, and f_1, f_2, α and β are defined by equations 2.3.7.

Since δP_2 is an independent variation, from 2.4.9,

$$\begin{aligned} & \sum_{m=1}^m \int_{R_m} [B_m^T (C_m B_m + D_m M_m) + \beta T_m T_m^T] P_2 dA_m \\ &= - \sum_{m=1}^m \left[\int_{R_m} (B_m^T E_m + \alpha T_m) dA_m + \int_{C_{2m}} T_m \gamma ds_m \right] \quad (2.4.11) \end{aligned}$$

In 2.4.11, the line integrals are evaluated only for those elements that adjoin the boundary B.

2.5 Triangular Elements

Referring to Figure 2.5.1(a) let L_i , $i = 1$ to 3, be linear functions in x and y such that

$$L_i = a_i + b_i x + c_i y \quad (2.5.1)$$

where L_i equals 1 at node i and zero at the remaining two nodes. The constants a_i , b_i , c_i may now be explicitly stated as

$$\begin{aligned} a_1 &= (x_3 y_2 - x_2 y_3) / 2A \\ b_1 &= (y_3 - y_2) / 2A \\ c_1 &= (x_2 - x_3) / 2A \end{aligned} \quad (2.5.2)$$

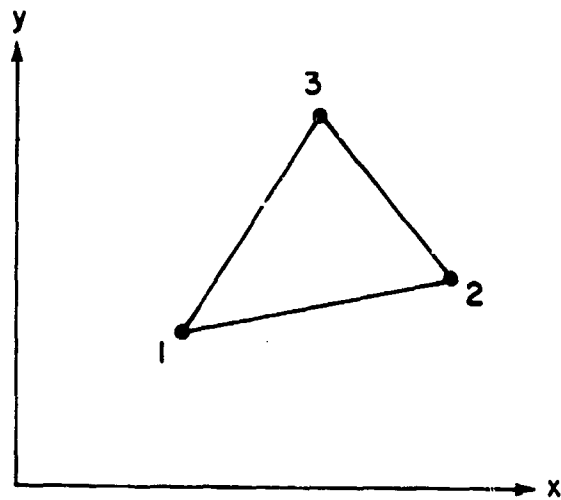
where A is the area of the element and the subscripts indicate element nodes. By a permutation of the indices, the remaining constants may be deduced from 2.5.2 in a straight forward manner. Thus, the column vector $\tilde{T}_m$ for the triangular element becomes

$$\tilde{T}_m^T = \langle L_1, L_2, L_3 \rangle \quad (2.5.3)$$

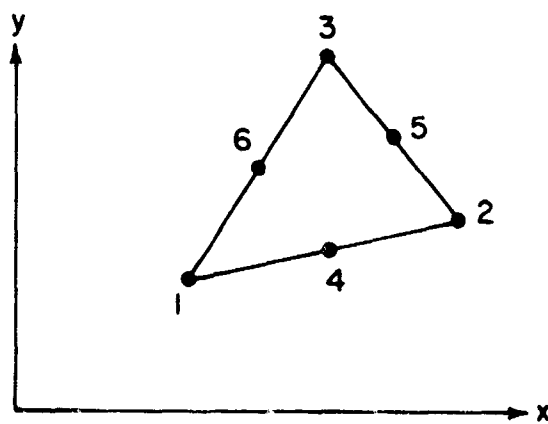
While this element allows almost a constant gradient within it, a refined element may be set up with additional nodes. Thus, referring to Figure 2.5.1(b),

$$\tilde{T}_m^T = \langle L_1(2L_1 - 1), L_2(2L_2 - 1), L_3(2L_3 - 1), 4L_1L_2, 4L_2L_3, 4L_3L_1 \rangle \quad (2.5.4)$$

where now the mid-side nodes come into effect. This element allows a linear variation of the gradient within it.



(a) THREE POINT TRIANGLE



(b) SIX POINT TRIANGLE

Figure 2.5.1 Triangular Elements

The required spatial integration over the element may be performed by a three point rule. Denoting a point X_1 in the triangle by its three area coordinates,

$$\int_{\Delta} F \, dA = \sum_{i=1}^3 W_i F(X_i) \quad (2.5.5)$$

where the weights W_i and the X_i in terms of the triangular coordinates are given by [5]

$$\begin{aligned} X_1 &= (2/3, 1/6, 1/6); W_1 = 1/3 \\ X_2 &= (1/6, 2/3, 1/6); W_2 = 1/3 \\ X_3 &= (1/6, 1/6, 2/3); W_3 = 1/3 \end{aligned} \quad (2.5.6)$$

It is also possible to evaluate the integrals in a closed form by resorting to the formula [6]

$$\int_{\Delta} L_1^i L_2^j L_3^k \, dA = [i! j! k! / (n+2)!] 2A \quad (2.5.7)$$

where

$$n = i + j + k \quad (2.5.8)$$

and i, j, k are non-negative integers.

While 2.5.7 is exact for polynomials of all degrees, the numerical integration rule 2.5.5 is exact for polynomials up to the second degree. Since three point element utilizes polynomials of the first degree and the six point element, polynomials of the second degree, the numerical rule 2.5.5 is completely adequate for the present purposes.

2.6 Quadrilateral Elements

Referring to Figure 2.6.1(a), two coordinates (ξ, η) , associated with a quadrilateral element are determined such that

$$\begin{aligned}\xi &= 1 \text{ on side } 23 \\ \xi &= -1 \text{ on side } 14 \\ \eta &= -1 \text{ on side } 12 \\ \eta &= 1 \text{ on side } 43\end{aligned}$$

The interpolation functions $N_i(\xi, \eta)$ are defined such that

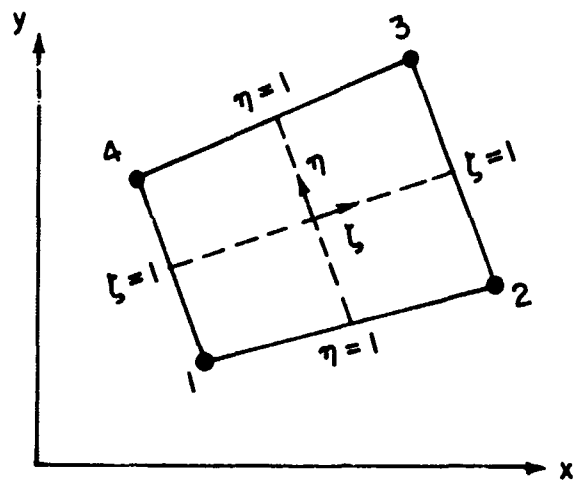
$$\begin{aligned}x &= \sum_{i=1}^J N_i x_i \\ y &= \sum_{i=1}^J N_i y_i\end{aligned}\tag{2.6.1}$$

where J designates the number of nodes in the element and i refers to the individual nodes in the element. With this definition of N_i , the functional form of the pressure P over the element may be expressed as

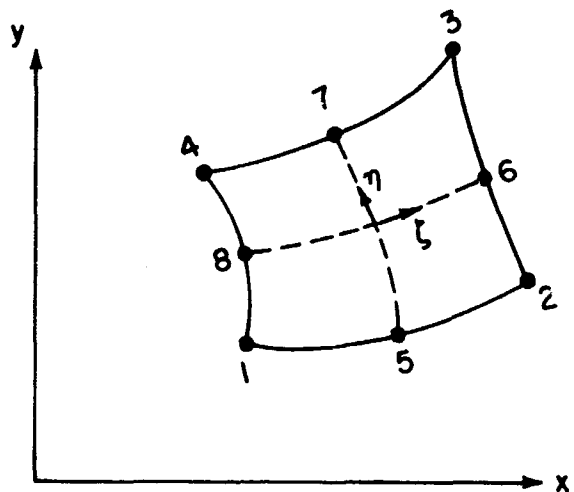
$$P = \sum_{i=1}^J N_i P_i\tag{2.6.2}$$

For a four-point quadrilateral, the functions N_i are given by [3]

$$\begin{aligned}N_1 &= (1 - \xi)(1 - \eta)/4 \\ N_2 &= (1 + \xi)(1 - \eta)/4 \\ N_3 &= (1 + \xi)(1 + \eta)/4 \\ N_4 &= (1 - \xi)(1 + \eta)/4\end{aligned}\tag{2.6.3}$$



(a) FOUR POINT QUADRILATERAL



(b) EIGHT POINT QUADRILATERAL

Figure 2.6.1 Quadrilateral Elements

and the column vector T_m for the four point quadrilateral is defined by

$$T_m^T = \langle N_1, N_2, N_3, N_4 \rangle \quad (2.6.4)$$

Referring to Figure 2.6.1(b), for the eight point quadrilateral, the function N_i for $i = 1, 4$ are defined by

$$N_i = [(1 + \xi_o)(1 + \eta_o) - (1 - \xi^2)(1 + \eta_o) - (1 + \xi_o)(1 - \eta^2)]/4 \quad (2.6.5)$$

where,

$$\begin{aligned} \xi_o &= \xi \xi_i \\ \eta_o &= \eta \eta_i \end{aligned} \quad (2.6.6)$$

with ξ_i and η_i being equal to either +1 or -1 depending on the node i .

The interpolation functions N_i for mid-side nodes are defined by [7]

$$N_i = (1 - \xi^2)(1 + \eta_o)/2 \text{ for } i = 5 \text{ and } 7 \quad (2.6.7)$$

and

$$N_i = (1 + \xi_o)(1 - \eta^2)/2 \text{ for } i = 6 \text{ and } 8 \quad (2.6.8)$$

The column vector T_m for the eight point quadrilateral is defined by

$$T_m^T = \langle N_1, N_2, N_3, N_4, N_5, N_6, N_7, N_8 \rangle \quad (2.6.9)$$

Inasmuch as the interpolation functions are defined in terms of ξ and η , a relation between the derivatives in the (ξ, η) and (x, y) coordinate systems is required and this is given by

$$\begin{Bmatrix} N_{i,\xi} \\ N_{i,\eta} \end{Bmatrix} = [J] \begin{Bmatrix} N_{i,x} \\ N_{i,y} \end{Bmatrix} \quad (2.6.10)$$

where, J is the Jacobian matrix given by

$$J = \begin{bmatrix} N_{1,\xi} & N_{2,\xi} & \cdots \\ N_{1,\eta} & N_{2,\eta} & \cdots \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \end{bmatrix} \quad (2.6.11)$$

From 2.6.10, the derivatives in (x,y) system are given by

$$N_{i,x} = \begin{bmatrix} 1 & 0 \end{bmatrix} J^{-1} \begin{Bmatrix} N_{i,\xi} \\ N_{i,\eta} \end{Bmatrix} \quad (2.6.12)$$

and

$$N_{i,y} = \begin{bmatrix} 0 & 1 \end{bmatrix} J^{-1} \begin{Bmatrix} N_{i,\xi} \\ N_{i,\eta} \end{Bmatrix} \quad (2.6.13)$$

Since the (ξ, η) coordinates range over -1 to 1, it is convenient to integrate in that system. Thus, the substitution

$$dx dy = \text{Det } [J] d\xi d\eta \quad (2.6.14)$$

and change of limits to -1 and +1 allows numerical integration in the Gaussian coordinates ξ, η . A Gaussian quadrature scheme, based on a two point rule in each coordinate direction is given by

$$\int F(\xi, \eta) d\xi d\eta = \sum_{i=1}^2 \sum_{j=1}^2 F(\xi_i, \eta_j) \quad (2.6.15)$$

where ξ_i and η_j are given by ± 0.57735027 . This rule is exact for all polynomials up to the second degree.

2.7 Boundary Conditions

For those nodes that lie on the boundary segment C_1 , pressure boundary conditions apply. Suppose that node i is on C_1 and the pressure at node i is a function of time given by $\bar{P}_i(T)$. Since equations 2.4.11 have the incremental pressure P_2 as the unknown, the incremental boundary condition must be deduced. Thus if ΔT is the time interval and T_1 is the time at which the boundary condition is to be applied, then at node i ,

$$P_{2i} = \bar{P}_i(T_1 + \Delta T) - \bar{P}_i(T_1) \quad (2.7.1)$$

If $\bar{P}_i$ is a constant, then P_{2i} is clearly equal to zero. To apply the boundary condition to the system of equations, let

$$[K] \underline{P}_2 = \underline{R} \quad (2.7.2)$$

where K is the assembled matrix and $\underline{R}$ is the vector of right hand sides. If the i th node is to be a pressure prescribed node, then, we modify 2.7.2 in this manner

$$\begin{aligned} R_j - K_{ji} P_{2i} &\rightarrow R_j \quad j \neq i \\ 0 &\rightarrow K_{ji}, K_{ij} \quad \text{for } i \neq j \\ 1 &\rightarrow K_{ii} \\ P_{2i} &\rightarrow R_i \end{aligned} \quad (2.7.3)$$

where the subscripted quantities are matrix elements of 2.7.2.

In a similar fashion, for each of the finite elements that has one or more sides on the boundary segment C_2 , the prescribed flow is

partitioned between the nodes on that side and simply added into the corresponding elements of the matrix K . Where symmetry or other considerations imply zero flow, Q_1 and Q_2 are set to zero.

2.8 Solution Scheme

The set of equations 2.4.11 for the incremental pressure P_2 constitute a banded, linear algebraic system that can be solved by Gaussian Elimination. Subsequent to solution, the incremental pressure vector is accumulated at each step to yield the pressure field at that time.

3. PROGRAM DESCRIPTION

3.1 Introduction

The computer program FELUB, (an acronym derived from Finite Element LUBrication Analysis), is a general purpose program written in Fortran IV for use with medium core-size computers. In general, the program can be used for analyzing any bearing, seal or pump type device in which the physical behavior is adequately predicted by Reynolds' equation.

In developing the program, while the temptation to achieve total generality was great, some restraint had to be exercised, because, at least with computer programs, generality is synonymous with maximality of input. Moreover, generality implies several options in the flow of information, and options in turn imply CPU consuming decisions. Thus, the final version of the program includes the following options which were judged to represent a good balance between generality, ease of input preparation and computation cost.

- a. Fluid Films
 - Incompressible
 - Isothermal Compressible
- b. Fluid Mechanics
 - Hydro-dynamic
 - Hydrostatic
 - Hybrid
- c. Temporal Description
 - Steady-State
 - Transient
- d. Spatial Description
 - Cartesian

- Polar
- Cylindrical
- Conical

The output from the program consists of a table of pressures within the device analyzed, the components of pressure integrated over the entire film in three mutually perpendicular directions, and flow across specified paths. In addition, input parameters, either read in, or internally generated, are also listed in the output printout to facilitate input checking.

3.2 Major Program Variables

COMMON/NAME/

TITLE(18)	Job Title
KONTRL(6)	Branch Control
" (1) = 1	Manual Input for Grid
" (1) = 2	Automatic Input for Grid
" (2) = 1	Steady-State
" (2) = 2	Transient
" (3) = 1	Compressible
" (3) = 2	Incompressible
" (4) = 1	Hydrostatic or Hydrodynamic
" (4) = 2	Hydrostatic and Hydrodynamic (Hybrid)
" (5) = 1	Cartesian (X,Y)
" (5) = 2	Polar (R, θ)
" (5) = 3	Cylindrical ($\theta, 2$), R
" (5) = 4	Conical (R, θ), α
" (6) = 1	Plots required
" (6) = 2	Plots not required
NLOOPS	Number of paths over which flow is to be computed.
TRUNC	Needed only for steady state, compressible case. Stops computation when average change in pressure is less than TRUNC.

COMMON/MESH/

NNODES	Number of nodes in assemblage.
NPANS	Number of elements in assemblage.
NPJTS(I,J)	Table of node numbers for elements. In J = 9 is stored the number of nodes in the element. I is the element number.
ITHIK(I)	Thickness klue set up by user to indicate if the Ith element is in a step or a groove. Not needed when film thickness is continuous.
CØØRD(J,I)	Coordinates of the Ith element. After call to TRANF, these are cartesian.

COMMON/FIELD/

X(j)	Element nodal coordinates x, relative.
Y(J)	Element nodal coordinates y, relative.
KLUE	Number of nodes in element.
IFILM	Thickness clue for element obtained from ITHIK.
PSI(J)	Element nodal pressures from previous step.
D(L,M)	Element matrix.
RR(L)	Element right hand side vector.
SIG	$\text{Sigma}/(2\Delta T)$
XA	Absolute coordinate x of node 1 in element.
YA	Absolute coordinate y of node 2 in element.
FMIN	Typical film thickness.
ØMEGA	Typical frequency.

COMMON/EQUATN/

A(I,J)	Augmented matrix of equations.
NEQ	Number of equations.

NBAND	Half bandwidth.
NAUG	Column location of right hand side in A.
NBAND2	Bandwidth.
PSTAR(I)	Dimensionless pressure at the I-th node.
COMMON/BOUNDR/	
NBC	Number of nodes at which pressure boundary conditions are specified.
NBCNDS(I)	Array of node numbers at which boundary conditions are specified.
P1(I)	Specified pressure at the nodes. P1(I) is in one-to-one correspondance with NBCNDS(I).
COMMON/TIMES/	
TIME	Current time.
DT	Time increment.
KSTEP	Current step number.
VKSLAM	Viscosity.
ALPRAD	Fixed radius in cylindrical coordinates or half-cone angle in conical coordinates.
PAMB	Ambient pressure.
AU1	Maximum speed in 1-direction.
AU2	Maximum speed in 2-direction.
DPLA	Fractional increase in speed.
PLA	Current fraction of maximum speed.

3.3 Subroutine Descriptions

ENTRY FELUB is the main routine which determines the overall computational sequence. A general flow chart for FELUB is given in

Figure 3.2.1. With a call to INPUT, required data for a problem is assembled. If a transformation of coordinates is required, TRANF is called. Next, influence coefficients for load computation are established by a call to INFSYS. Required arrays are initialized and an incremental loop on NSTEPS is entered. A call to SPEED establishes operating speed at that instant of time. The finite element equation for the step are assembled by a call to SYSTEM. These are then solved by a call to SOLVE. Results are accumulated and displayed by a call to OUTPUT. After all incremental steps are completed, if ISENT has been set to zero, program execution stops; otherwise control returns to ENTRY point.

SUBROUTINE CØNTRL is called by INPUT once for each problem. CØNTRL reads the two command cards that follow the title card in the input deck and analyzes them to fill the vector KØNTRL (6). In the first command card, it detects the first character of the first, fourth and sixth words respectively. It fills KØNTRL (6) in the following manner.

1st Command Card

<u>Word No.</u>	<u>Character No.</u>	<u>Character</u>	<u>KØNTRL(I)</u>
1	1	M	KØNTRL(1) = 1
1	1	A	KØNTRL(1) = 2
4	1	S	KØNTRL(2) = 1
4	1	T	KØNTRL(2) = 2
5	1	C	KØNTRL(3) = 1
5	1	I	KØNTRL(3) = 2

2nd Command Card

<u>Word No.</u>	<u>Character No.</u>	<u>Character</u>	<u>KØNTRL(I)</u>
1	1	S	KØNTRL(4) = 1
1	1	H	KØNTRL(4) = 2
4	5	E	KØNTRL(5) = 1

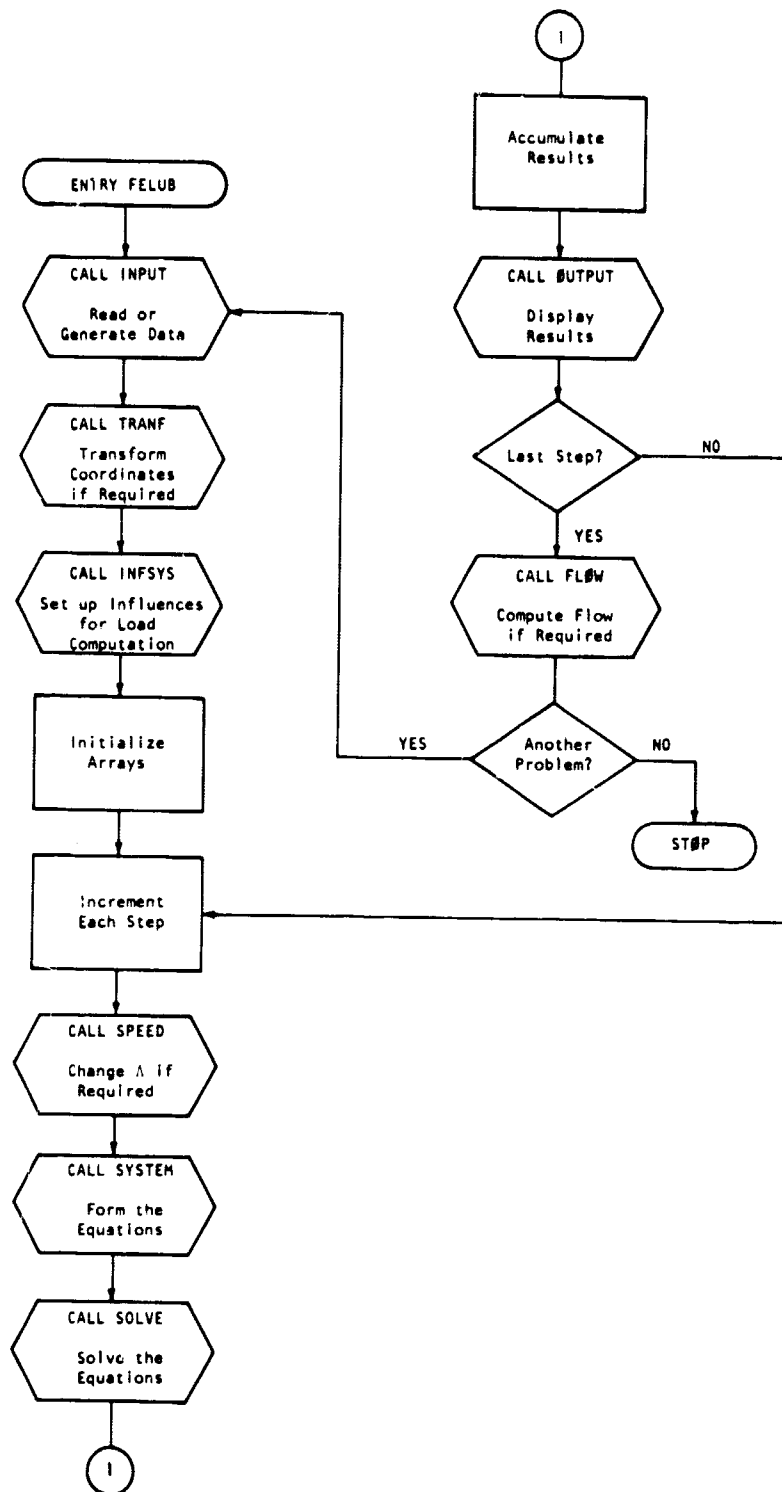


Figure 3.2.1 General Flow Chart for FELUB

<u>Word No.</u>	<u>Character No.</u>	<u>Character</u>	<u>KØNTRL(I)</u>
4	5	R	KØNTRL(5) = 2
4	5	N	KØNTRL(5) = 3
4	5	C	KØNTRL(5) = 4
6	1	P	KØNTRL(6) = 1
6	1	N	KØNTRL(6) = 2

For the purposes of this routine, words are defined as groups of non-blank characters separated by one or more blank fields. The labelled common /FIELD/ is used in this routine as scratch storage and not for transfer of information to other routines.

SUBROUTINE INPUT is called by FELUB once for each problem. It assembles the input data for each problem from the input deck. Execution control is set up by a call to CØNTRL. One of the two options is exercised for grid generation: If a user supplied routine AUTGRD is available, it is called. Otherwise MANGRD is called. These routines supply nodal coordinates and a table of node numbers and type for each finite element in the assemblage. In addition, boundary conditions for the problem are established.

The input parameters are put in a dimensionless form and the bandwidth for the problem is determined. All the input quantities are displayed via the printer.

This routine uses labelled common /NAME/, /MESH/, /FIELD/, /EQUATN/, /BØUNDR/ and /TIMES/. Directly or by external references, all the commons except /FIELD/ and /EQUATN/ are filled by this routine. In the latter two, the variables NEQ, NBAND, NAUG, NBAND2, SIG, HMIN and ØMEGA are also filled.

SUBROUTINE MANGRD is called by INPUT once for each problem. It generates grid information and boundary conditions. Special features are provided for minimizing input when four point quadrilateral

elements are used. In reading the sequential, nodal coordinate data, omitted information is generated on the basis of equal division on a straight line connecting the first and last nodes bounding the missing information. Similarly, in reading the sequential, element data, for missing elements, the last element nodes are incremented by 1 or 2 as specified. Where missing information cannot be generated in a consistent fashion, an Error Stop is made.

For each element, in addition to the nodal data, information regarding the manner in which the film thickness is to be determined is read in. This is particularly useful, indeed necessary, in step bearings where coordinate information is insufficient to determine film thickness at boundaries of grooves and lands.

This routine fills labelled common /MESH/ and /BOUND/. In addition, in /TIMES/, the variable ALPRAD is set for problems in which cylindrical or conical coordinates are utilized.

SUBROUTINE TRANF is called by FELUB once for each problem in which the coordinate system is not a cartesian system. Depending on the coordinate system used, it takes one of three branches to transform the coordinates into a cartesian system. In the case of cylindrical and conical systems, the developed surface is used in addition. The transformations are as follows:

a) Polar Coordinates

Input: R and θ for each node N in $COORD(I,N)$, $I=1,2$

Output: $X = COORD(1,N) = R \cos \theta$

$Y = COORD(2,N) = R \sin \theta$

b) Cylindrical Coordinates

Input: θ_d and Z for each node N in $COORD(I,N)$, $I=1,2$ and radius R in ALPRAD

Output: $X = COORD(1,N) = R \theta_r$

$Y = COORD(2,N) = Z$

Input θ_d is in degrees. Subroutine converts it to θ_r in radians.

c) Conical Coordinates

Input: R and θ_d for each node N in CØØRD(I,N) I=1,2 and half-cone angle α_d in ALPRAD

Output: $X = CØØRD(1,N) = (R \cos(\sin \alpha_r \theta_r)) / \sin \alpha_r$

$Y = CØØRD(2,N) = (R \sin(\sin \alpha_r \theta_r)) / \sin \alpha_r$

Input θ_d and α_d are in degrees. Subroutine converts them to θ_r and α_r in radians.

Labelled commons /TIMES/ and /MESH/ are used in this routine. Argument NCØRD is input to the routine and identifies coordinate type as follows:

NCØRD = 2 polar
 = 3 cylindrical
 = 4 conical

If NCØRD = 1, this routine is not called.

SUBROUTINE SYSTEM is called by FELUB once for each increment or step in each problem. SYSTEM begins by initializing the system matrix A to zero and entering a loop on all the elements in the assemblage. For each element, it fills the labelled common /FIELD/ with element data and calls wither FTCF3 or FTCF4 depending on whether the element is a triangle or a quadrilateral. These routines return the element matrix and right hand side vector in the variables D and RR respectively in /FIELD/, which are then added into the final system A. The column NAUG in A(N, NAUG) is the accumulated right hand side and A may be called the augmented matrix of the system. If an element with number of nodes other than 3, 4, 6 or 8 is encountered, this routine terminates with an error stop. After the entire system is assembled, boundary conditions are applied and control is returned to FELUB.

Labelled commons /NAME/, /FIELD/, /MESH/, /EQUATN/, /BØUNDR/ and /TIMES/ are used by this routine.

SUBROUTINE FTCF3 is called by SYSTEM. It generates the element matrix and right hand side for triangular elements with three or

six nodes. Appropriate interpolation functions and their derivatives are obtained by a call to TXT. A three point rule is used to integrate over the area of the element. Depending on the coordinate systems used and whether the problem is compressible or incompressible, alternative computational procedures are utilized in generating the element matrix and right hand side. Labelled common /NAME/, /TIMES/ and /FIELD/ are used by this routine.

SUBROUTINE FTCTF4 is called by SYSTEM for quadrilateral elements with four or eight nodes. Interpolation functions and their derivatives are obtained by a call to DXD. A two point rule giving rise to four terms is used to integrate over the area of the element. This routine is similar to FTCTF3 in all other respects.

SUBROUTINE INFSYS called by FELUB, once for each problem, generates three influence vectors for load computation in each of the three coordinate directions. Scalar product of these vectors with the vector of nodal pressures yields the load components. The routine begins by initializing FORMOM to zero and entering a loop on all the elements. For each element, at each node, unit pressure is assumed while the remaining nodes in the element are assumed to be at zero pressure. This pressure field is integrated over the element and the components are accumulated in FORMOM. The computation is repeated for each node in the element. The integration is done by a call to INGR3 for triangular elements and INGR4 for quadrilaterals. If an element with number of nodes other than 3, 4, 6 or 8 is encountered, this routine terminates with an error stop.

This routine uses labelled commons /MESH/, /FIELD/ and /INFLNC/.

Subroutine INGR3 is called by INFSYS once for each triangular element. It generates influence coefficients for load computation by integrating a pressure field of unity at one node and zero at the remaining and integrating over the area of the element. A three point

integration rule is used. Interpolation functions for the pressure distribution are obtained by a call to TXT.

This routine uses labelled commons /NAME/, /TIMES/ and /FIELD/.

SUBROUTINE INGRT4 is called by INFSYS once for each quadrilateral element. It is similar to INGRT3 in its function. A three point rule giving rise to nine terms is used in the integration. A call to DXD provides the interpolation functions for determining the pressure distribution.

This routine uses labelled commons /NAME/, /TIMES/ and /FIELD/.

SUBROUTINE OUTPUT is called by FELUB once for each step in all cases except steady-state problems in which it is called only after the final step. It computes load components and outputs them via the printer along with the nodal pressures.

Labelled commons /NAME/, /MESH/, /EQUATN/, /TIMES/ and /INFLNC/ are used.

SUBROUTINE SOLVE is called by FELUB once for each step. It solves the banded system of equations of the augmented matrix A, stored in the labelled common /EQUATN/ by Gaussian Elimination. If, during forward elimination, a diagonal term is found to be zero, this routine terminates with an error stop. The solution is returned in the NAUG column of A(N,NAUG).

Labelled common /EQUATN/ is used by this routine.

SUBROUTINE FLOW called by FELUB computes flow over prescribed paths. It calls FTCT3 or FTCT4, depending on whether triangular or quadrilateral elements are involved. For each element, the element matrix is assembled and with the known solution vector, the flow across the specified edge is obtained by appropriate multiplication of the element matrix and nodal pressures for the element.

This routine uses /EQUATN/ for scratch storage. The other labelled commons are /FIELD/, /MESH/ and /TIMES/.

SUBROUTINE TXT is called by FTCF3 and INGRT3. It returns the interpolation functions and their derivatives for three and six point triangles.

SUBROUTINE DXD is called by FTCF4 and INGRT4. It returns the interpolation functions and their derivatives for four and eight point quadrilaterals.

SUBROUTINE SPEED is a user supplied routine. Its input is the variable TIME representing time. This routine returns speed of the bearing at any instant of time and the change in speed during the next increment of time. These values are returned as a fraction of the maximum speed expected. Thus PLA equals the fraction corresponding to the current value and DPLA equals incremental fraction. The version of this routine supplied with the program may be used for all applications except those in which the speed is to be changed during the computation.

SUBROUTINE FILM is a user supplied routine. Its inputs are X1, X2, TIME, IFILM where X1 and X2 are coordinates of a location in the bearing in the untransformed coordinates, TIME is time and IFILM is a clue indicating whether the location is in a land region or groove region. The routine returns H which is the film thickness at time = TIME, and HDOT, which is the rate of change of film thickness.

3.4 Input Preparation

A natural starting point for the analysis of any bearing, seal or viscosity pump by the FELUB program is to make a list of the parametric conditions for which the device is to be analyzed. This may include such things as viscosity, speed, etc. At this point, if the geometry of the device is to be modified several times, while the basic configuration is kept unchanged, a decision should be made regarding the possibility of preparing a AUTGRD routine for automatically generating the mesh layout for the device under consideration.

A drawing of the developed view of the film can be particularly useful in making a judgement regarding regions in which rapid changes of the gradient of pressure may occur. In such regions, naturally, one would wish to employ a finer mesh than elsewhere. In addition, a drawing may aid one in delineating sections across which flow computation might be desirable.

Supposing that one has assembled all the pertinent information and subdivided the fluid film into an assemblage of finite elements. The next task is to number the nodes of the elements in such a fashion that in the entire assemblage, the maximum of the difference between the largest and smallest nodal numbers appearing in any element is as small as possible. While this objective is not a crucial consideration in so far as solution accuracy is concerned, it has considerable bearing on minimizing the computation time.

At this point one is ready to prepare input and the input map given in Table 3.4 may be consulted. A description of each of the card types and variables within each card type follows.

Card Type A

This card contains alpha-numeric information describing the analysis. This information is printed as a title on each page of the output. If the information is centered on the card, it will appear centered on a 128 column printout.

Card Type B

This card is essentially a command card that selects options in the analysis. This card is analyzed in the program as free field input. Words are assumed to be groups of non-blank characters separated by blanks. The first word can begin anywhere on the card and words must be separated by one or more blanks. The first word is allowed to be one of the two words: MANUAL or AUTOMATIC. The word AUTOMATIC is to be used only when the user supplies the AUTGRD routine. The second and

Table 3.4
INPUT MAP FOR 'FELUB', (VERSION 3, REV. 1)

<u>CARD TYPE</u>	<u>COLUMNS</u>	<u>VARIABLE</u>	<u>DEFINITION</u>	<u>FORMAT*</u>
A	1-80	TITLE	Job Title	A
B	1-80	NZ	Type of Analysis	Z
{MANUAL AUTOMATIC} Δ INPUT Δ FOR Δ {STEADY-STATE TRANSIENT} Δ {COMPRESSIBLE INCOMPRESSIBLE} Δ LUBRICATION				
C	1-80	NZ	Type of bearing and output	Z
{STANDARD HYBRID} Δ BEARING Δ IN Δ {CARTESIAN POLAR CYLINDRICAL CONICAL} Δ COORDINATES Δ {PLOTS NO PLOTS} Δ REQUIRED				
D	1-6	NNODES	Number of nodes	I
	7-12	NPANS	Number of elements	I
	13-18	NBC	Number of pressure prescribed nodes	I
	19-30	ALPRAD	If card type C indicates • Cartesian - ALPRAD is not needed • Polar - ALPRAD is not needed • Cylindrical - ALPRAD is radius • Conical - ALPRAD is half cone angle	E
E	5-6	ILINE	Connector	I
	7-12	N	Node number	I
	13-24	COORD(1,N)	Coordinate 1 for node N • (X, R, θ, R)	E

*I-Format requires integer right justified in field

E-Format can accept a decimal number anywhere in field, or an exponential number with exponent (base 10) right justified in field, or integer right justified in field

A-Format requires Alphanumeric Data. All zeros may be input as blank

Z-Free field with blanks as separators. One of the words in each curl bracket is mandatory

Table 3.4

INPUT MAP FOR 'FELUB', (VERSION 3, REV. 1) (Cont'd)

<u>CARD TYPE</u>	<u>COLUMNS</u>	<u>VARIABLE</u>	<u>DEFINITION</u>	<u>FORMAT</u>
	25-36	CØØRD(2,N)	Coordinate 2 for node N • (Y, θ , Z, θ)	E
	37-42	NMID	Starting node in previous list	I

Repeat card type E until all nodes are generated. Last card is for N = NNØDES. Coordinates 1 and 2 are (X,Y) for Cartesian, (R, θ) for Polar, (θ ,Z) for Cylindrical, (R, θ) for Conical.

F	6	ILINE	Element connector	I
	7-12	N	Element number	I
	13-18	NPJTS(1,N)	1st nodal number in element	I
	19-24	NPJTS(2,N)	2nd " "	I
	25-30	NPJTS(3,N)	3rd " "	I
	31-36	NPJTS(4,N)	4th " "	I
	37-42	NPJTS(5,N)	5th " "	I
	43-48	NPJTS(6,N)	6th " "	I
	49-54	NPJTS(7,N)	7th " "	I
	55-60	NPJTS(8,N)	8th " "	I
	61-66	NPJTS(9,N)	Number of nodes in element	I
	67-72	JH	Stepper for quadrilaterals	I
	73-78	ITHIK(N)	Clue indicating whether on land or groove	I

Repeat card type F until all elements are generated. ILINE and JH are useful only when quadrilateral elements are used.

G	1-6	NBCNDS(1)	Node number N1	I
	7-18	P1(1)	Prescribed pressure at N1	E
	19-24	NBCNDS(2)	Node number N2	I
	25-36	P1(2)	Prescribed pressure at N2	E

Table 3.4

INPUT MAP FOR 'FELUB', (VERSION 3, REV. 1) (Cont'd)

<u>CARD TYPE</u>	<u>COLUMNS</u>	<u>VARIABLE</u>	<u>DEFINITION</u>	<u>FORMAT</u>
	37-42	NBCNDS(3)	Node number N3	I
	43-54	P1(3)	Prescribed pressure at N3	E
	55-60	NBCNDS(4)	Node number N4	I
	61-72	P1(4)	Prescribed pressure at N4	E

Repeat card type G until all NBC nodes are listed

H	1-9	BISLAM	Viscosity	E
	10-18	HMIN	Typical film thickness	E
	19-27	WIDTH	Typical Bearing dimension	E
	28-36	OMEGA	Typical frequency	E
	37-45	U1	Typical speed in 1-direction	E
	46-54	U2	Typical speed in 2-direction	E
	55-63	PAMB	Ambient pressure	E
I	1-6	NSTEP	Number of computational steps	I
	7-18	DT	Time increment	E
	19-30	TIME	Initial time	E
	31-42	TRUNC	Truncation	E
	43-48	NLØØPS	Number of paths over which flow is to be completed	I

If NLØØPS = 0, go to card type L

J	1-6	N	Flow loop number	I
	7-12	NELM	Number of elements in loop	I
K	1-6	IELM(1)	Element number	I
	7-12	IED(1)	Edge number	I
	13-18	IELM(2)	Element number	I
	19-24	IED(2)	Edge number	I

Table 3.4

INPUT MAP FOR 'FELUB', (VERSION 3, REV. 1) (Cont'd)

<u>CARD TYPE</u>	<u>COLUMNS</u>	<u>VARIABLE</u>	<u>DEFINITION</u>	<u>FORMAT</u>
	25-30	IELM(3)	Element number	I
	31-36	IED(3)	Edge number	I
	37-42	IELM(4)	Element number	I
	43-48	IED(4)	Edge number	I
	49-54	IELM(5)	Element number	I
	55-60	IED(5)	Edge number	I
	61-66	IELM(6)	Element number	I
	67-72	IED(6)	Edge number	I

Repeat card type K until NELM elements are listed. Then repeat card types J and K until NLOOPS sets are listed.

L	1-6	ISENT	Sentinel to signal last problem = 0 another problem follows ≠ 0 last problem	I
---	-----	-------	--	---

third words are not significant. The fourth word must be one of the allowed two and the fifth word must also be one of the allowed two. Usage of these words is self-explanatory, the remaining words in card type B are of no significance.

Card Type C

This card is also a command card and usage is similar to card type B. The word HYBRID is assumed to denote a bearing in which both hydrostatic and hydrodynamic lubrication take place. Anything other than HYBRID is assumed to be called STANDARD.

Card Type D

The variables in this card type are self explanatory. Figure 3.4.1(a) indicates definition of ALPRAD for cylindrical coordinate system and Figure 3.4.1(b), indicates the same for a conical system.

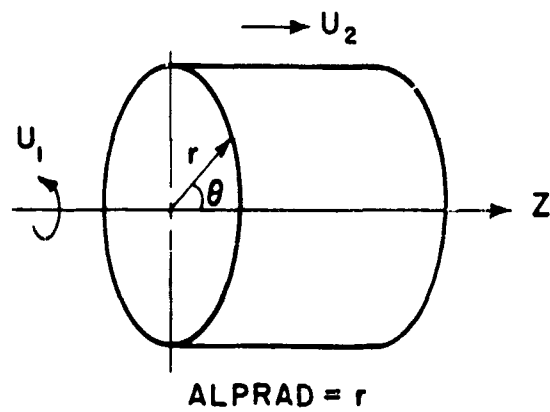
Card Type E

This card type inputs the nodal coordinates, these being assumed to be given in the coordinate system specified in card type C. Meaning of coordinates 1 and 2 is assumed to be given by

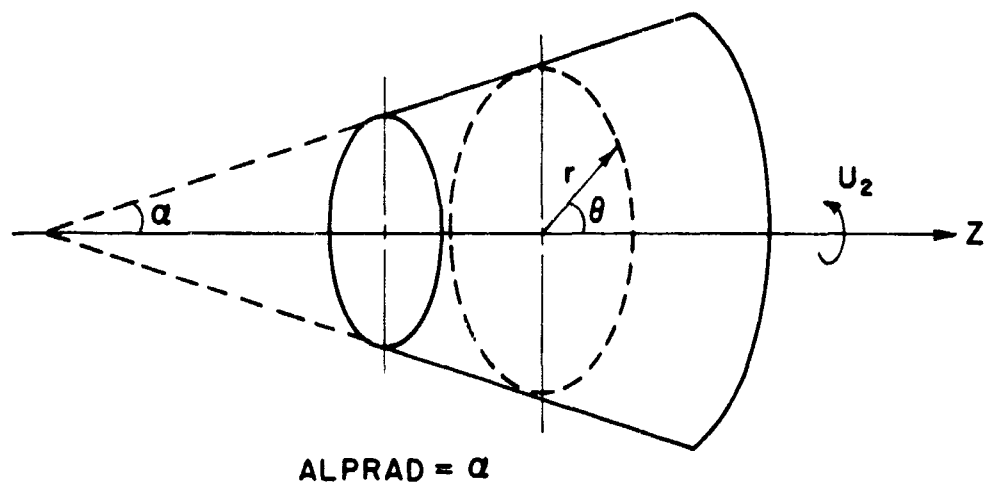
	<u>COORDINATE 1</u>	<u>COORDINATE 2</u>
Cartesian	X	Y
Polar	R	θ
Cylindrical	θ	Z
Conical	R	θ

This card type allows certain flexibility in use to minimize the number of cards required to describe nodal coordinates.

Ignoring ILINE and NMID for the moment, the nodal input is read sequentially with increasing nodal numbers. If node N_1 is read in and the next card has node N_2 , where N_2 is more than one removed from N_1 , then the program assumes that equal increments in coordinates 1



(a) CYLINDRICAL SYSTEM



(b) CONICAL SYSTEM

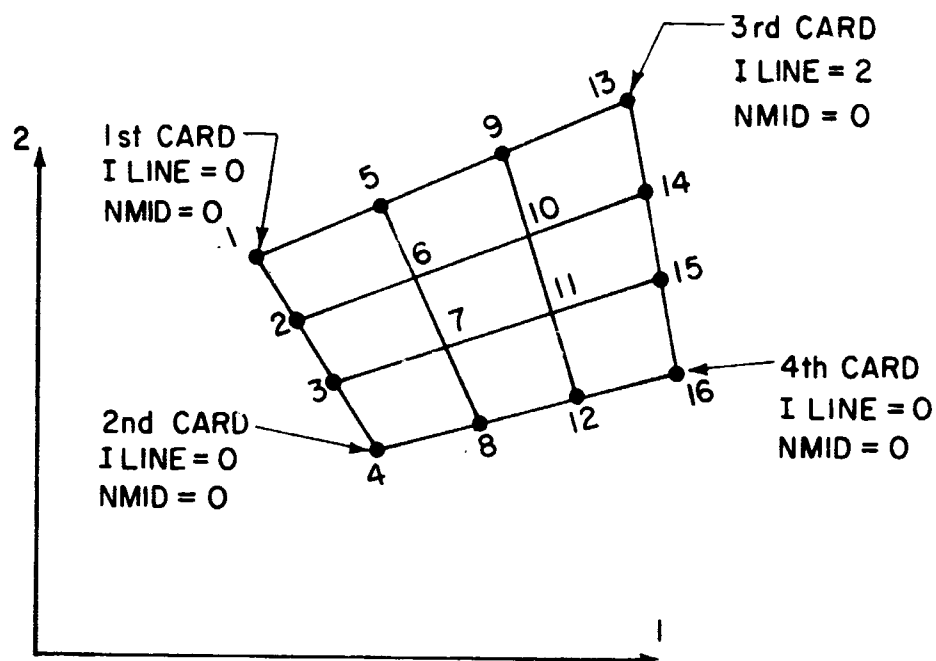
Figure 3.4.1 Cylindrical and Conical Coordinate Systems

and 2 are to be provided between nodes N_1 and N_2 , the incrementing being done in the coordinate system specified in card type C.

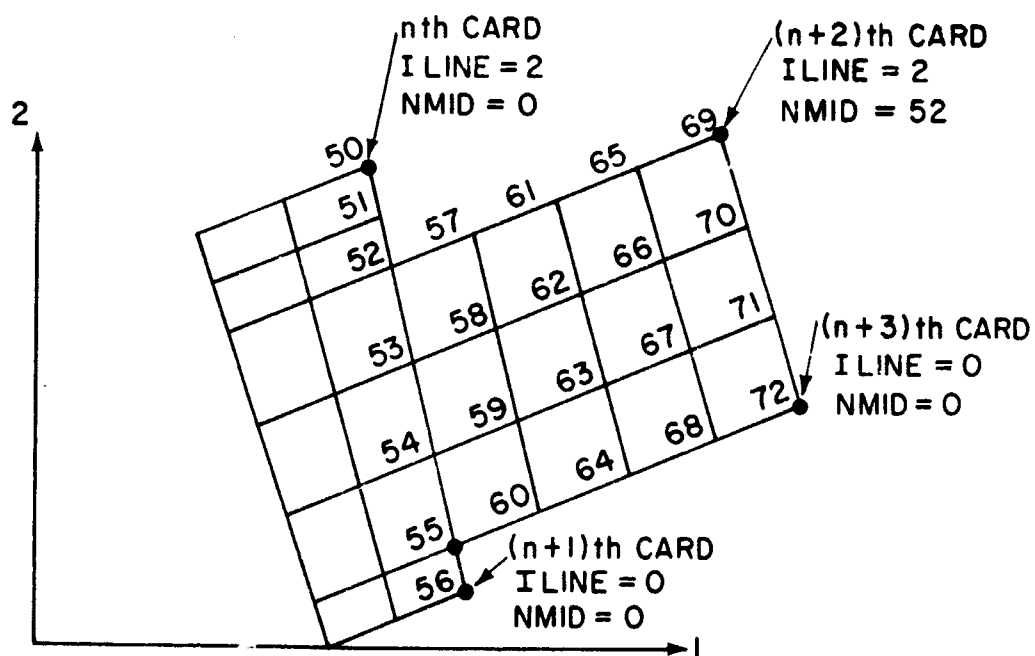
If every node is included in the input deck, then they need not be arranged in sequence.

ILINE allows generation of coordinates for a mesh of nodes. Its use is best illustrated by considering an example: suppose a quadrilateral domain with a 3×3 grid shown in Fig. 3.4.2(a) is to be generated. In card type E, there need be only four cards as follows: First, a line of nodes 1 to 4 giving their coordinates. ILINE and NMID are set to zero on these cards. Next, a card for node 13 with coordinates, ILINE = 2 and NMID = 0 is read in. Finally, a card for node 16 with coordinates, ILINE = 0, NMID = 0 is read in. The program can then generate the coordinates for all the remaining nodes under the assumption that equal increments are used in each coordinate direction. In words, use of ILINE may be defined as follows: If N_1 is the last node read into core and if the next card has ILINE = m then the set of m nodes to be read in beginning with the node that has ILINE associated with it is to be connected to the corresponding set already in core by a mesh. The set of m nodes read under ILINE control need not be sequential, between each set of nodes that are not sequential, equal increments in coordinates will be assumed.

While ILINE allows a mesh connection between two sets of nodes when there can be no ambiguity in such a connection, NMID allows other cases where such ambiguity arises to be taken care of. Referring to Figure 3.4.2(b), suppose the line of nodes 50 through 56 is already generated. In order to generate the remaining nodes 57 through 72, two cards are necessary, one each for node 69 and node 72. ILINE = 2 is specified on the card for node 69 to indicate that a mesh connection is required and NMID = 52 on the same card to indicate that the connection begins at node 52 in the nodal set already generated.



(a) ARC OF I LINE ILLUSTRATED



(b) USE OF NMID ILLUSTRATED

Figure 3.4.2 Use of ILINE and NMID

The mesh generation feature is intended primarily for use with four point quadrilateral elements. When other elements are used, mesh generation feature of card type E must be used with caution.

Card Type F

This card type inputs the nodal numbers for each element in the assemblage. The sequence in which the nodal numbers are to be listed is shown in Figure 2.5.1, page 2-11 for triangular elements and in Figure 2.5.2, page 2-14 for quadrilateral elements. Element node number 1 can be selected to be any of the vertices, but the sequence for the remaining must follow the indicated sense. A convenient way to keep the sense same throughout, is to look at the element in the direction of the inward normal and pick the sense of rotation of 1-direction into the 2-direction.

Ignoring ILINE and JH for the moment, this card type lists the element number N, the number of nodes in the element and the node numbers. In addition, ITHIK is a integer clue selected by the user to indicate whether the element is in a groove or land. With the understanding that the film thickness is continuous within any element, this clue can be utilized in the user supplied routine FILM to select any of several function types used to describe the film thickness as a function of coordinates.

By means of JH and ILINE this card type allows certain flexibility in use to minimize the number of cards required to describe four point quadrilateral elements. First, the element input is read sequentially with increasing element numbers. If two element numbers, more than 1 removed from each other are read in, then the program fills in the missing element node numbers by incrementing the corresponding previous one in storage by one or two depending on whether JH is zero or non-zero, respectively.

The use of ILINE is slightly different from the previous card type. Here it indicates that the next element number is to be read in and connected to the previous set of elements in memory. The card immediately following the card with non-zero ILINE need not list the node numbers and ILINE is significant only as being either equal to zero or not equal to zero. For internally generated elements, the program sets ITHIK to be equal to the last one read in.

Card Type G

This card type lists the boundary conditions. Description of the variables in the input map is sufficiently self-explanatory.

Card Type H

This card type specifies the problem parameters. Where a typical frequency does not exist, it may be set equal to 1. U1 and U2 are maximum speeds in the 1 and 2 directions of the coordinate system selected, and have the following meaning:

- Cartesian
 - U1 - max. lineal speed in x-direction (LT^{-1})
 - U2 - max. lineal speed in y-direction (LT^{-1})
- Polar
 - U1 - not defined
 - U2 - max. angular velocity ($Rad.T^{-1}$)
- Cylindrical
 - U1 - max. angular velocity ($Rad.T^{-1}$)
 - U2 - max. lineal speed in z-direction (LT^{-1})
- Conical
 - U1 - not defined
 - U2 - max. angular velocity ($Rad.T^{-1}$)

The remaining variables in this card type are self-explanatory.



Card Type I

The variables for this card type are self-explanatory.

Card Type J

A flow loop is defined as a continuous line joining several nodes along element edges. The line is assumed not to cross any of the elements. The elements to be listed are assumed to be on one side or the other of the line as one travels along the line. The sign of the computed flow is determined by which side is selected. If NLØØPS is set to zero in card type I, card type J should be omitted.

Card Type K

Edge number is defined by the element nodes that it connects as follows:

Element Type	Edge 1	Edge 2	Edge 3	Edge 4
3PT triangle	1-2	2-3	3-4	---
6PT triangle	1-2	2-3	3-4	---
4PT quadrilateral	1-2	2-3	3-4	4-1
8PT quadrilateral	1-2	2-3	3-4	4-1

If NLØØPS in card type I is set to zero, this card type should be omitted.

Card Type L

This is a sentinel indicating whether more problems follow.

The user in addition to an input deck must supply a subroutine that defines film thickness as a function of coordinates and time. Calling sequence is

CALL FILM (x1, x2, TIME, IFILM, H, HDØT)

x1, x2, TIME, and IFILM are inputs to the subroutine and, H and HDØT are outputs of the routine. x1 and x2 are coordinates in the user specified system in card type C. TIME is current time and IFILM is the thickness clue as defined by the user in card type F.

If effects of changing the bearing speed are to be studied, the SPEED routine also must be supplied by the user. The calling sequence is

CALL SPEED (TIME, PLA, DPLA)

where TIME is current time, PLA is the current fraction of maximum speed specified in card type H and DPLA is the expected change in PLA during the time increment DT specified in card type I.

If the user wants to automate mesh generation for particular bearing configurations, subroutines may be generated for each of them. The calling sequence is

CALL AUTGRD

The routine must fill the COMMON/MESH/, COMMON/BOUNDR/ and ALPRAD in COMMON/TIMES/. It may use /EQUATN/ as scratch storage.

3.5 Validation

The transient lubrication problem defined by equations 2.1.5, has an exact, though trivial, one-dimensional solution. While the solution is of little consequence, in so far as engineering applications are concerned, it is useful for validation and studies of the effects of discretization and truncation.

Consider the initial and boundary conditions give by

$$P(x,0) = (\sigma + \Lambda) e^{2\alpha x} / 2\alpha \quad 3.5.1$$

and

$$P(0,T) = (\sigma + \Lambda) e^{2\alpha T} / 2\alpha \quad 3.5.2$$

$$P(1,T) = (\sigma + \Lambda) e^{2\alpha(1+T)} / 2\alpha \quad 3.5.3$$

where x is the one dimensional space variable ranging over $0 \leq x \leq 1$, T is time ranging over $0 \leq T$ and α is a non-negative quantity. If the film thickness H is given by

$$H = e^{-\alpha(x+T)} \quad 3.5.4$$

then

$$P = (\sigma + \Lambda) e^{2\alpha(x+T)} / 2\alpha \quad 3.5.5$$

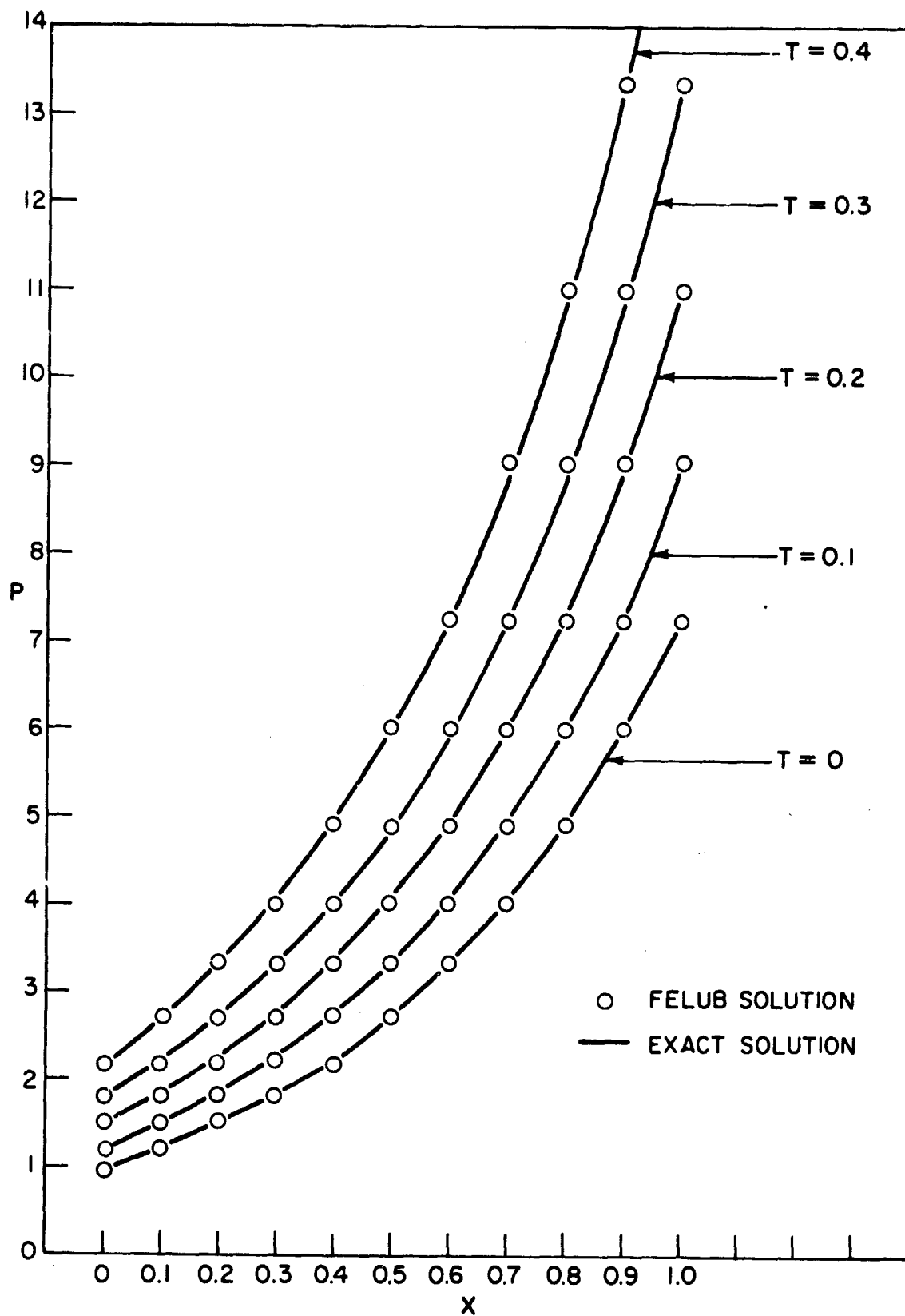
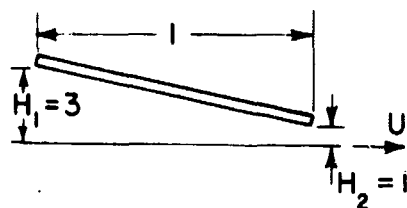


Figure 3.5.1 Comparison of Transient Solutions, Computed and Exact



$$\Lambda = 10$$

$$h_1/h_2 = 3$$

$$P_{MAX} = 1.40296$$

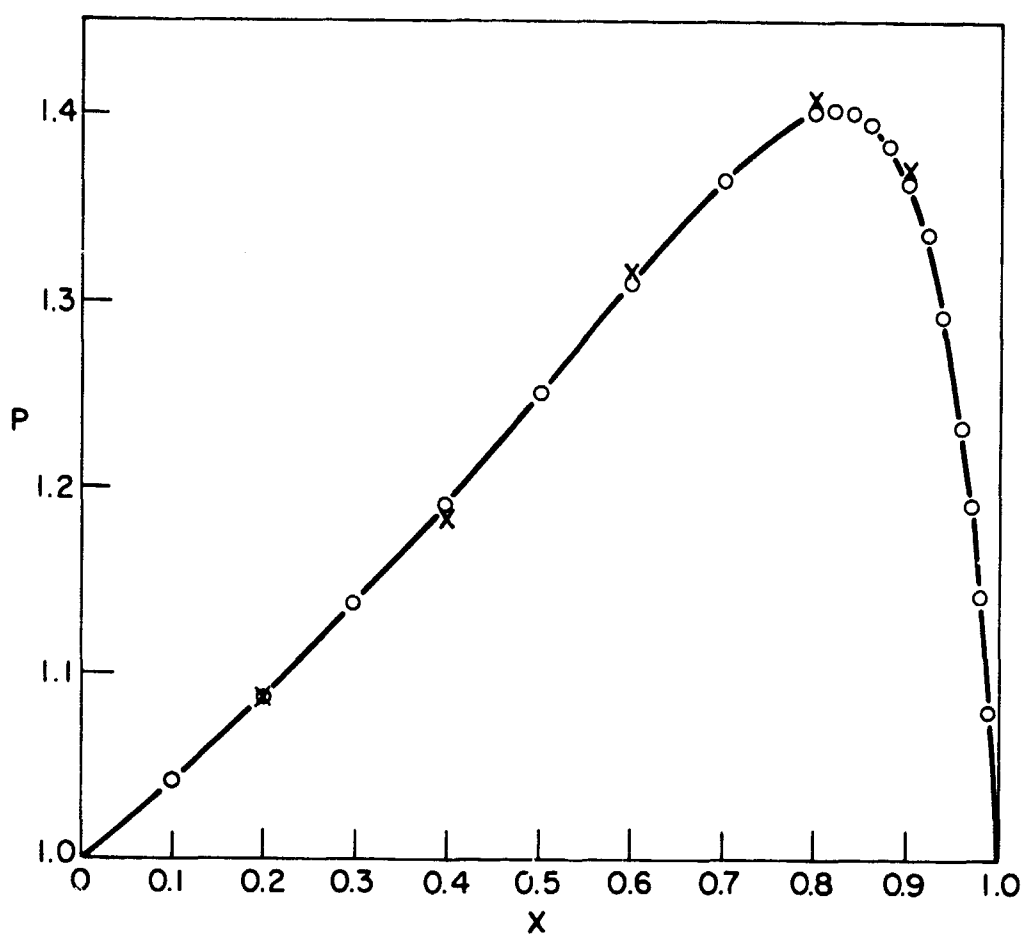


Figure 3.5.2 Steady State Solution for One-Dimensional Slider Bearing, $\Lambda = 10$

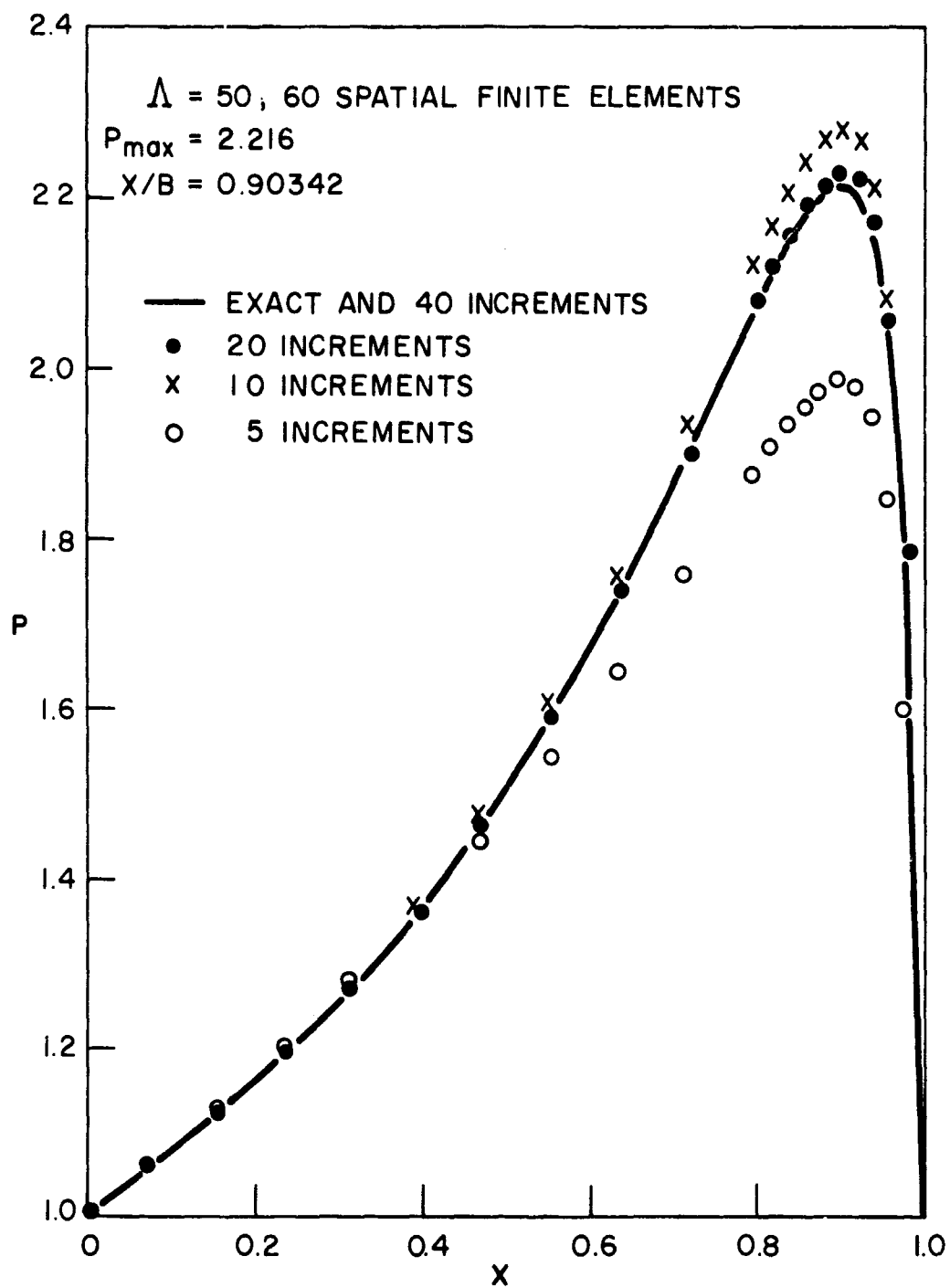


Figure 3.5.3 Steady State Solution for One-Dimensional Slider Bearing, $\Lambda = 50$

is a solution to equation 2.1.5, which can be readily verified by substituting equation 3.5.5 into 2.1.5.

Referring to Fig. 3.5.1, solution for a problem with 4 spatial increments and $\alpha = 1$, $\Lambda = 1$, $\sigma = 1$, obtained by FELUB is illustrated. With $\Delta T = .01$, the computed solution is found to differ negligibly from the exact solution given by equation 3.5.5.

Fig. 3.5.2 illustrates the steady state solution for a one dimensional slider bearing obtained with 20 finite elements, Λ being kept constant at 10. An exact solution for this case is given by Harrison [8]. With $\Delta T = -.1$, and 15 steps, the computed pressure profile is found to differ negligibly from the exact solution. This may be compared with a previous solution [3] which required 20 steps for obtaining the same accuracy. The presence of $\sigma(PH)_T$ in the present solution seems to be the major factor in obtaining the accelerated convergence.

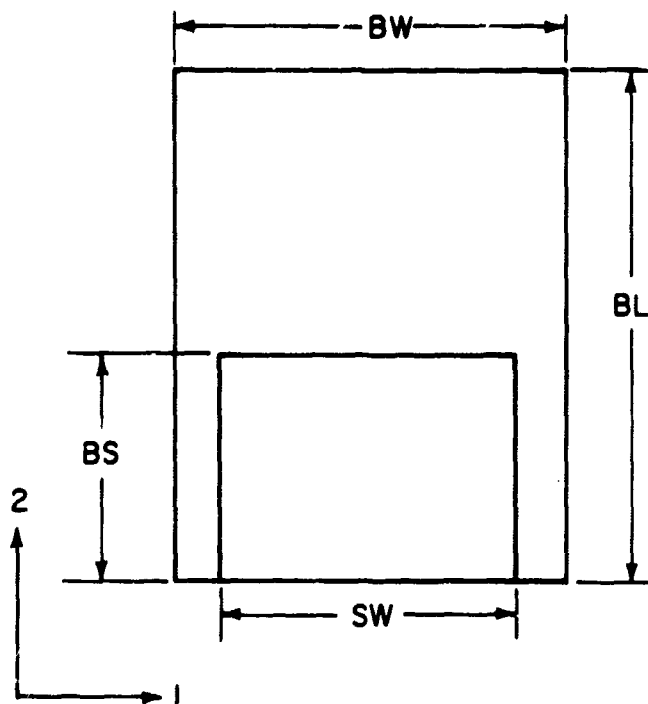
Referring to Fig. 3.5.3, solution for the same slider bearing but with $\Lambda = 50$ and 60 finite elements is shown. With 20 increments, the solution is within the requirements of engineering accuracy. The solutions shown for 5 and 10 increments are seen, respectively, to be below and above the true solution. This is a stationary rather than extremal. In the incompressible case, which is an extremal, the solution always converge from below.

3.6 Shrouded Step Bearing

As an example of a problem with complex geometry, the steady-state pressure distribution for the shrouded step bearing is considered.

An AUTGRD routine was prepared to generate the grid for both rectangular and polar configurations of the bearing. This routine supplies card types D, E, F and G of the input maps given in Table 3.4.

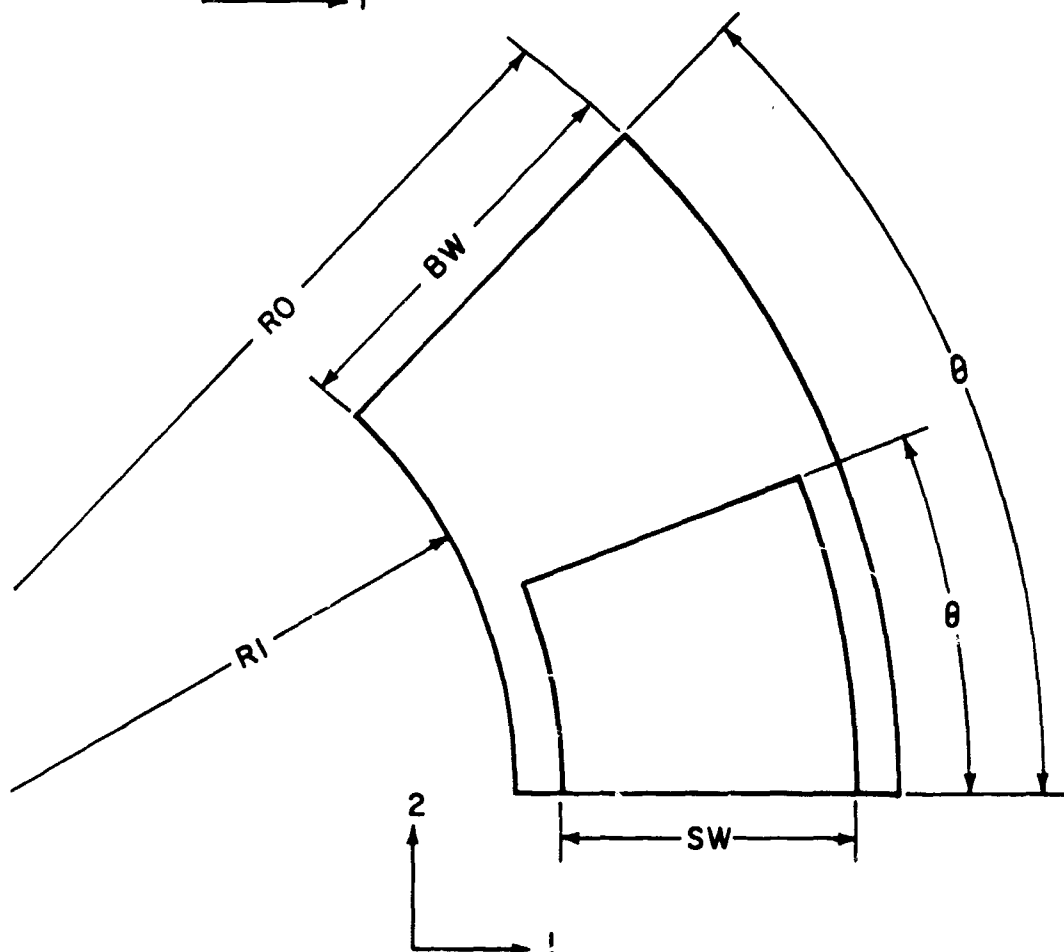
Definition of the variables used in the routine are shown in Fig. 3.6.1.



$$\frac{BW}{BL} = BWOBL$$

$$\frac{SW}{BW} = SWOBW$$

$$\frac{BS}{BL} = BS OBL$$



$$\frac{RI}{RO} \equiv RIORO$$

$$\frac{\theta_s}{\theta} \equiv BS OBL$$

$$\frac{SW}{BW} \equiv SWOBW$$

Figure 3.6.1 Shrouded Step Bearing Configuration

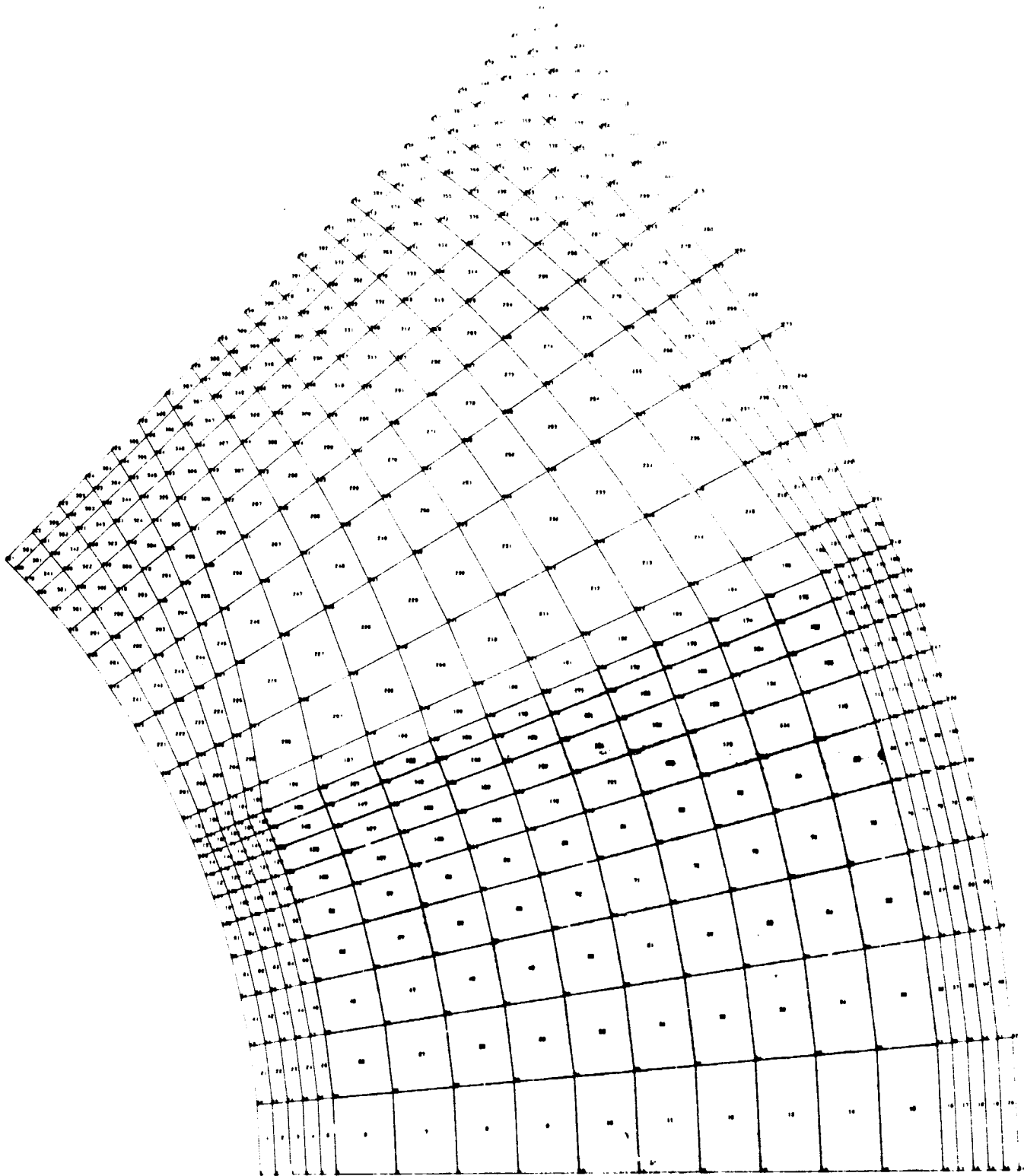


Figure 3.6.2 Polar, Shrouded Step Bearing Mesh

EOLDOUT FRAME 1

241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809	810	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837	838	839	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864	865	866	867	868	869	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959	960	961	962	963	964	965	966	967	968	969	970	971	972	973	974	975	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991	992	993	994	995	996	997	998	999	1000
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------

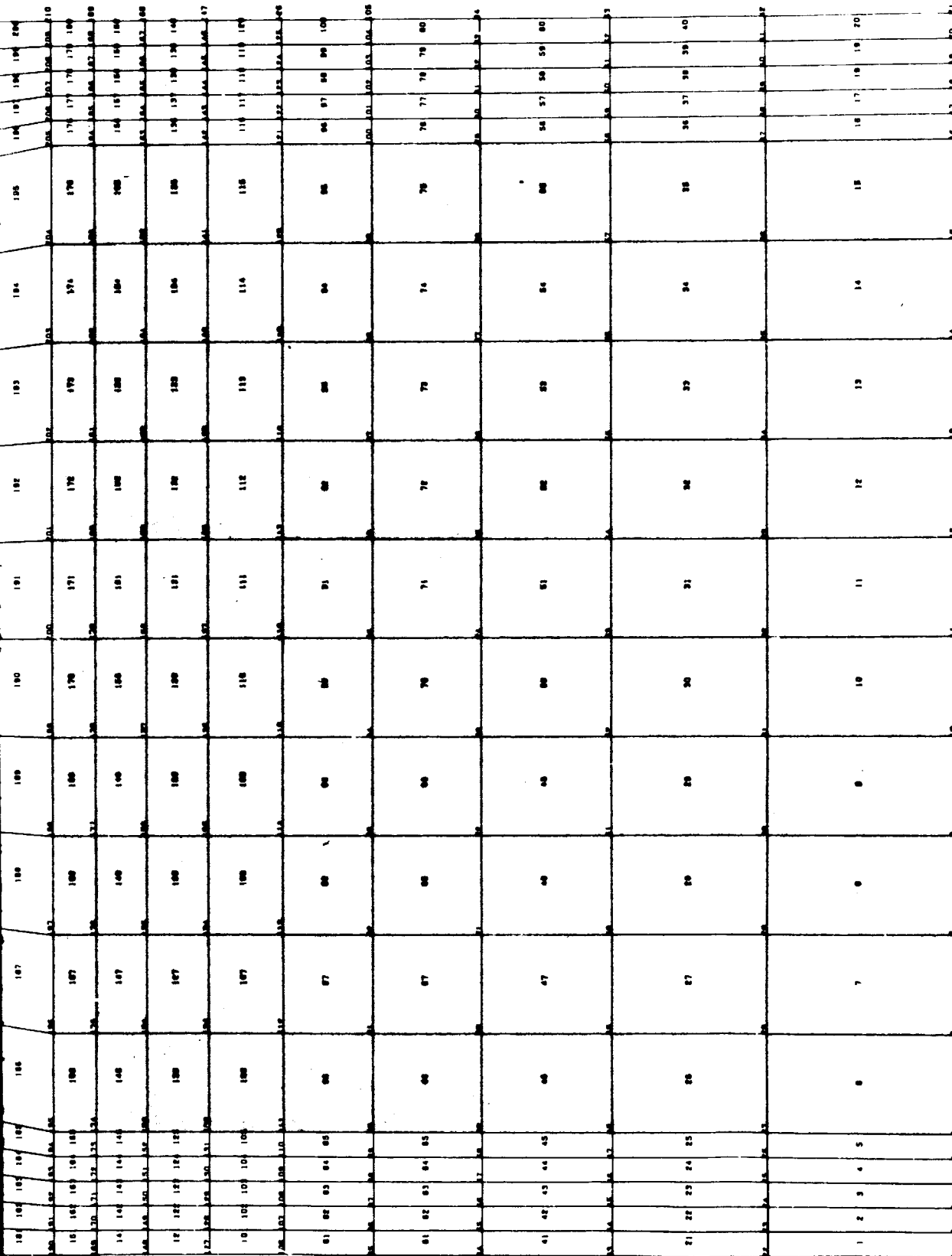


Figure 3.6.3 Rectangular Shrouded Step Bearing Mesh

CALCØMP plots of the mesh layout for typical cases in polar and rectangular configurations are shown in Fig. 3.6.2 and Fig. 3.6.3 respectively.

The particular case of a pad with the geometry shown in Fig. 3.6.4 was analyzed for an application with the velocity field shown in Fig. 3.6.5. The resulting steady-state solution obtained by FELUB is shown in Fig. 3.6.6 as isobars. The plot was made by a CALCØMP, 'GPCP' package.



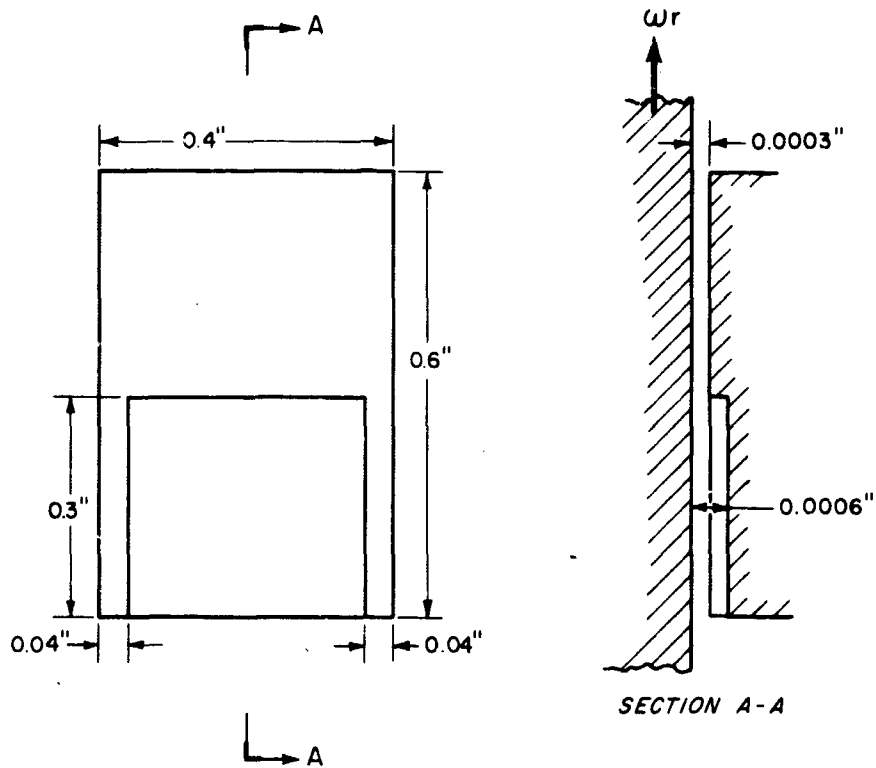


Figure 3.6.4 Pad Geometry

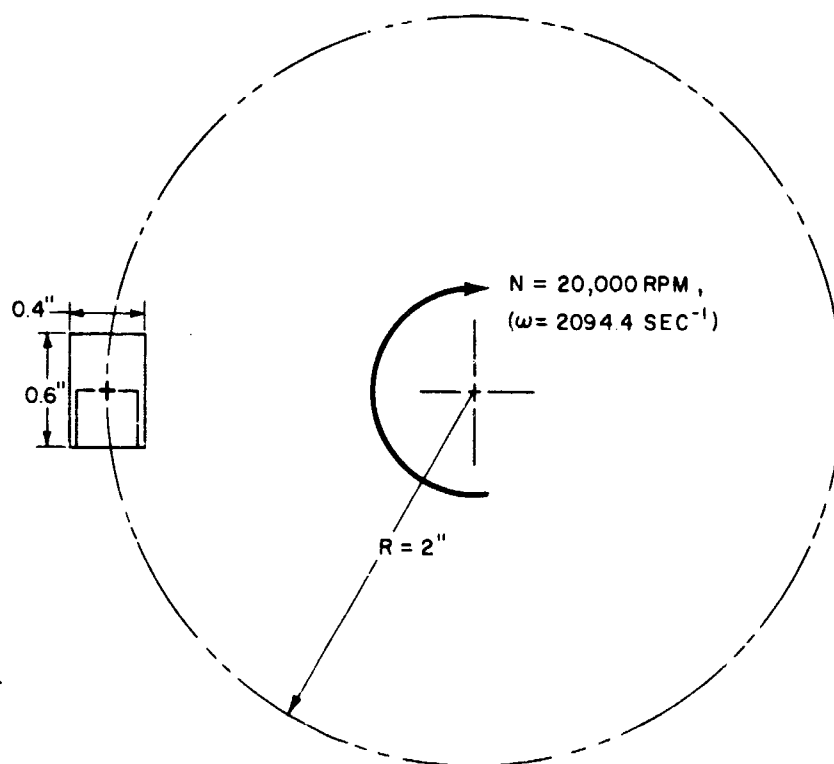


Figure 3.6.5 Polar Velocity Field

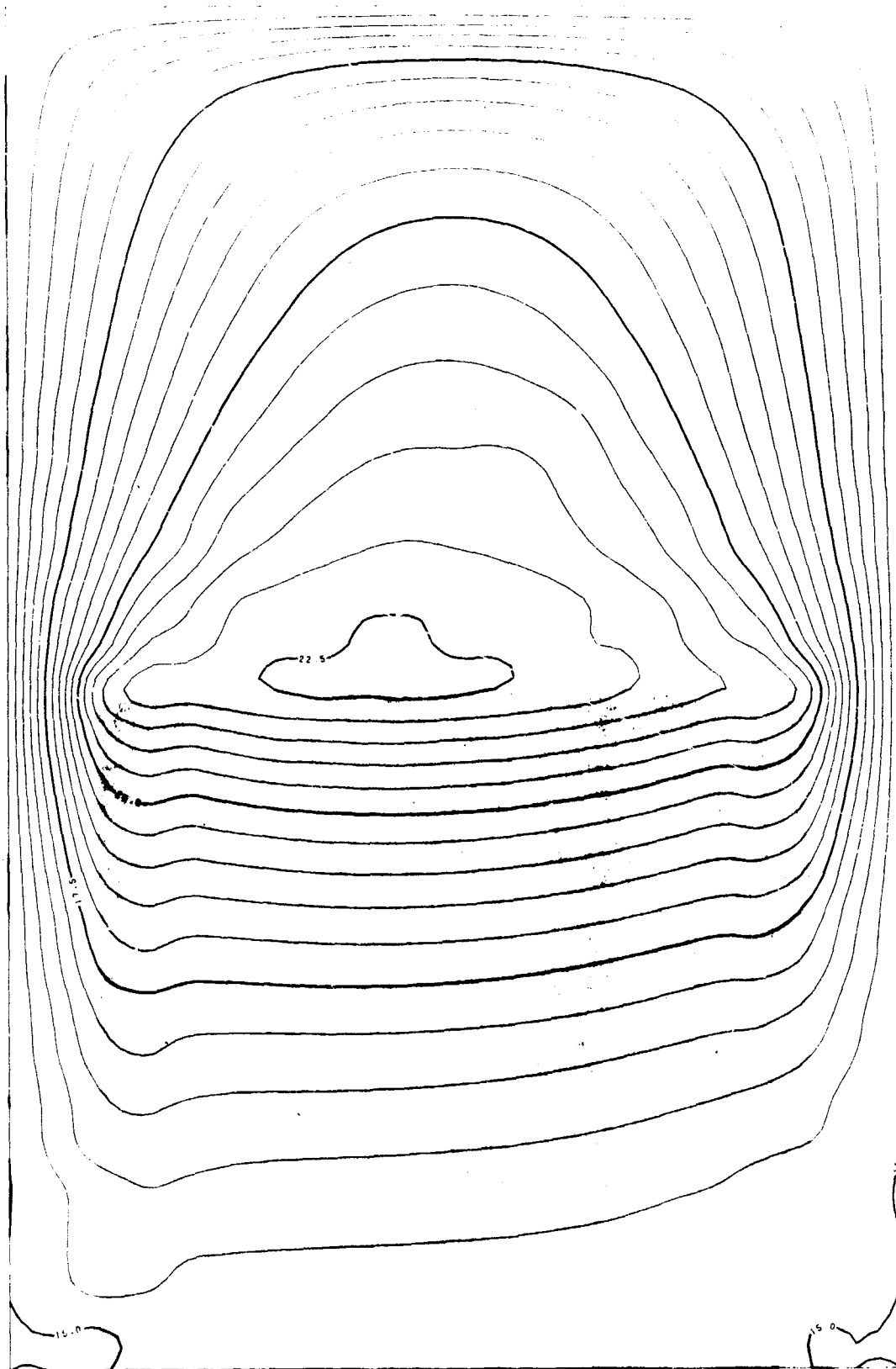


Figure 3.6.6 Steady State Pressure Distribution for Rectangular Shrouded Step Bearing

4. CONCLUSIONS

The overall objective of developing a general purpose, finite element lubrication program has been accomplished in the main.

Validity of the transient, variational formulation is established.

The preliminary experience with the program indicates that the transient procedure is extremely stable.

The transient procedure is a more economical way to solve the steady-state problem than that reported in reference [3].

5. REFERENCES

1. Castelli, V, and Pirvics, J., "Review of Numerical Methods in Gas Bearing Analysis," J. Lub. Tech., Trans. ASME, Series F, 90, 1968, pp. 777-792.
2. Reddi, M. M., "Finite Element Solution of the Incompressible Lubrication Problem," J. Lub. Tech., Trans ASME, 91, 1969, pp. 524-533.
3. Reddi, M. M. and T. Y. Chu, "Finite Element Solution of the Steady State Compressible Lubrication Problem," J. Lub. Tech., Trans ASME, 92, 1970, 495-503.
4. Michael, W. A., "Approximate Methods for Time-Dependent Gas-Film Lubrication Problems," J. App. Mech., Trans. ASME, 1963, pp. 509-517.
5. Hammer, P. C. and A. H. Stroud, "Numerical Evaluation of Multiple Integrals II," MTAC, V. 11, 1957, pp. 59-67.
6. Stricklin, J. A., "Integration of Area Coordinates in Matrix Structural Analysis," AIAA Journal, 6, 1968, p. 2023.
7. Ergatoudis, I., B. M. Irons, and O. C. Zickukiewicz, "Curved, Isoparametric, 'Quadrilateral' Elements for Finite Element Analysis," Int. J. Solids Structures, 1968, 4, pp. 31-42.
8. Gross, A. A., Gas Film Lubrication, Wiley, New York, 1962, pp. 54-83.

