1

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NATIONAL LIBRARY
OTTAWA

BIBLIOTHÈQUE NATIONALE
OTTAWA

NAME OF AUTHOR... REDA NAGUIB TADROS.....

TITLE OF THESIS... A Finite Element - Galerkin Model for Nonlinear
... Fluid Flow Problems.....

UNIVERSITY... Carleton University, Ottawa, Ontario.....

DEGREE FOR WHICH THESIS WAS PRESENTED... Ph.D.....

YEAR THIS DEGREE GRANTED... 1977.....

Permission is hereby granted to THE NATIONAL LIBRARY
OF CANADA to microfilm this thesis and to lend or sell copies
of the film.

The author reserves other publication rights, and
neither the thesis nor extensive extracts from it may be
printed or otherwise reproduced without the author's
written permission.

(Signed).....

PERMANENT ADDRESS

149 Coulange Street Apt. 307

Longueuil, Quebec

DATED... September 30, 1977

NL-91 (10-68)

Supervisor... James Kirkhope

National Library of Canada

Cataloguing Branch
Canadian Theses Division

Ottawa, Canada
K1A 0N4

Bibliothèque nationale du Canada

Direction du catalogage
Division des thèses canadiennes

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us a poor photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

**THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED**

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de mauvaise qualité.

Les documents qui ont déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

**LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE**

A FINITE ELEMENT - GALERKIN MODEL
FOR NONLINEAR FLUID FLOW PROBLEMS

by

Reda N. Tadros

B. Sc. (Aeron. & Mech. Eng.), Cairo University, Cairo, 1970

M. Sc. (Aeron. & Mech. Eng.), Cairo University, Cairo, 1973

A thesis submitted to the Faculty of Graduate Studies
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Mechanical Engineering.

Carleton University

Ottawa, Ontario

September, 1977

(C)

Reda N. Tadros

1977

The undersigned hereby recommend to the Faculty of
Graduate Studies, Carleton University, acceptance of this thesis,
"A Finite Element - Galerkin Model for Nonlinear Fluid Flow
Problems", submitted by Reda N. Tadros, in partial fulfillment
of the requirements for the degree of Doctor of Philosophy in
Mechanical Engineering.

Thesis Supervisor

Chairman, Mechanical and
Aeronautical Engineering

ABSTRACT

A Finite Element Galerkin Model (F.E.G.M.) has been developed and examined for solution of the two-dimensional, viscous, incompressible, steady and unsteady, non-linear flow equations.

Formulation of the equations of the model has been performed for both the primitive flow variables velocity and pressure (u, p), and for the stream function variable (ψ). Various finite elements have been used, having quadratic and quintic functions for the variables, and the method has been examined thoroughly for both steady and unsteady boundary layer flows by solving the resulting non-linear equations without referring to any linearization procedures. For comparison purposes, a detailed examination of the errors involved in linearization procedures has been carried out.

As a result of these investigations, novel methods have been derived which decrease substantially required computer core storage; allow choice of grid to obtain maximum efficiency and accuracy with minimum number of nodal variables; avoid the use of a large global matrix for the model; and, finally, improve the accuracy of results obtained when using non-compatible elements.

Results are presented for the Blasius steady state problem, with suction and blowing, and for steady wedge and stagnation flow problems. The transient problem of a suddenly accelerated plate in viscous flow is presented in some detail, and the effects of element formulation, grid choice, and time step on the conver-

gence and accuracy of the solution are examined thoroughly.

The results obtained indicate the high accuracy and efficiency of the F.E.G.M., when compared with alternative Finite Element and Finite Difference methods.

ACKNOWLEDGEMENTS

The acquisition of a doctoral degree is an undertaking requiring a large investment in time, thought, and effort. It is also an effort which requires the support and kindness of many other individuals and organizations. The author wishes to take this opportunity to express his sincere thanks to those persons whose help made his success possible.

First, the author expresses his deep sense of gratitude, and sincere appreciation to Professor J. Kirkhope for his valuable guidance and the many suggestions throughout the course of this investigation, and for his keenness to guide the research work. Thanks are extended to the staff of the Department of Mechanical and Aeronautical Engineering. Especially, to Professor H.I.H. Saravanamutto whose constant support is deeply appreciated; to Professors R.J. Kind and A.N. Abdelhamid whose supports and advices made the effort possible; to Professor W.J. Rainbird who suggested the test cases; and to Professor W.H. Bowes whose suggestions were quite appreciated. Special gratitude also goes to Professors E. Hughes and F. Fiala of the Mathematics Department who taught the author highly advanced mathematics.

The author is grateful to Dr. G.R. Cowper, National Research Council of Canada, for providing information about the high precision element used in this research. Finally, the author wishes to express his sincerest gratitude to his wife for her help and patient understanding during the preparation of this thesis.

This work was partially supported by the National Research
Council of Canada through Operating Grant A-7283.

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT	i
ACKNOWLEDGEMENTS	iii
TABLE OF CONTENTS	v
LIST OF FIGURES	ix
LIST OF TABLES	xi
LIST OF PRINCIPAL SYMBOLS	xiii
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 LITERATURE SURVEY	10
2.1 NUMERICAL SOLUTIONS OF THE NAVIER-STOKES EQUATIONS	10
2.2 FINITE ELEMENTS AND THE FLUID FLOW PROBLEM	11
CHAPTER 3 FINITE ELEMENT-GALERKIN MODEL FOR FLUID FLOW	17
3.1 THE FINITE ELEMENT CONCEPT	17
3.2 METHODS USED TO DETERMINE THE NODAL PARAMETERS	18
3.3 THE GALERKIN FORMULATION	23
3.4 THE FINITE ELEMENT - GALERKIN MODEL	26
3.5 THE FINITE ELEMENT SHAPE FUNCTION	29
3.6 METHOD OF ASSEMBLING THE RESIDUALS	34
CHAPTER 4 NUMERICAL SOLUTION OF THE DISCRETIZED EQUATIONS OF MOTION	39

	<u>Page</u>
4.1 THE FORM OF THE DISCRETIZED EQUATIONS	39
4.2 SOLUTION TECHNIQUE FOR STEADY FLOW	43
4.3 SOLUTION TECHNIQUE FOR UNSTEADY FLOW	48
CHAPTER 5 APPLICATION TO STEADY FLOW PROBLEMS	51
5.1 INTRODUCTION	51
5.2 FINITE ELEMENTS USED IN THE ANALYSIS	52
5.3 THE FLOW OVER A FLAT PLATE	54
5.3.1 ANALYSIS	54
5.3.2 RESULTS AND DISCUSSION	58
5.4 SUCTION, BLOWING, STAGNATION AND WEDGE FLOWS	59
5.4.1 ANALYSIS	59
5.4.2 RESULTS FOR SUCTION AND BLOWING	61
5.4.3 RESULTS FOR WEDGE AND STAGNATION	62
CHAPTER 6 APPLICATION TO UNSTEADY VISCOUS FLOW USING PRIMITIVE FLOW VARIABLES	64
6.1 INTRODUCTION	64
6.2 A REVIEW OF THE TRANSIENT FLOW OVER A SUDDENLY ACCELERATED PLATE	65
6.3 THE GOVERNING EQUATIONS	68
6.4 FINITE ELEMENT DISCRETIZATION	70
6.5 THE TIME MARCHING SCHEME	73
6.6 RESULTS AND DISCUSSION	75
6.7 DETAILS RELATED TO THE CHOICE OF GRIDS	78

	<u>Page</u>
CHAPTER 7 APPLICATION TO UNSTEADY VISCOUS FLOW USING STREAM FUNCTION FORMULATION	82
7.1 INTRODUCTION	82
7.2 THE GOVERNING EQUATIONS	83
7.3 FINITE ELEMENT DISCRETIZATION	87
7.4 THE TIME MARCHING SCHEME	97
7.5 RESULTS AND DISCUSSION	98
CHAPTER 8 ERRORS INVOLVED IN THE LINEARIZATION OF THE EQUATIONS OF MOTION	101
8.1 INTRODUCTION	101
8.2 METHODS OF LINEARIZATION	102
8.3 ANALYSIS FOR STEADY STATE FLOW	104
8.4 ANALYSIS FOR UNSTEADY FLOW	106
8.4.1 LINEARIZATION USING SMALL PERTURBATIONS	106
8.4.2 LINEARIZATION OF THE CONVECTIVE TERMS BY SERIES SOLUTION	113
8.5 RESULTS AND DISCUSSION	117
CHAPTER 9 CONCLUSIONS AND RECOMMENDATIONS	119
REFERENCES	127
FIGURES	138
TABLES	165
APPENDIX A ELEMENTAL MATRICES FOR STEADY FLOW	203
A.1 THIRD ORDER ELEMENT	203
A.2 QUINTIC ORDER ELEMENT	207

	<u>Page</u>
APPENDIX B ELEMENTAL MATRICES FOR UNSTEADY FLOW	217
B.1 QUADRATIC TRIANGULAR ELEMENT	217
B.1.1 AREA COORDINATES	217
B.1.2 RESIDUAL EVALUATION	220
B.2 QUINTIC TRIANGULAR ELEMENT	228
APPENDIX C DETAILS OF COMPUTER PROGRAMS	239
C.1 INTRODUCTION	239
C.2 FORTRAN PROGRAMS FOR THE STEADY STATE FLOWS	240
C.3 FORTRAN PROGRAMS FOR THE UNSTEADY STATE FLOWS	241
C.3.1 QUADRATIC ELEMENT AND PRIMITIVE VARIABLES	241
C.3.2 QUINTIC ELEMENT AND STREAM FUNCTION FORMULATION	242
C.4 LISTING OF PROGRAMS	243
C.4.1 STEADY FLOW	243
C.4.2 UNSTEADY FLOW WITH PRIMITIVE VARIABLES	269
C.4.3 UNSTEADY FLOW WITH STREAM FUNCTION FORMULATION	302

LIST OF FIGURESPage

1. Division of a Domain V into Finite Elements or Subdomains V_m .	138
2. Elements Based on Linear Strain Assumption.	139
3. Elements Based on Potential-Velocity Formulation.	140
4. Finite Elements Idealization of Cylinder Flow Problems.	141
5. Velocity Distribution for Different Elements after the Fourth Iterate.	142
6. Initial Iterate f_0 with the First two Iterates f_1 and f_2 for the Fifth Order Element versus Blasius.	143
7. Flow past a Wedge.	144
8. Finite Element-Velocity Distribution in the Boundary Layer of a Flat Plate with Suction and Discharge.	145
9. Finite Element-Velocity Distribution in the Boundary Layer of Wedges and Stagnation Flows.	146
10. Finite Element Meshes.	147
11. Profiles of the Velocity Component u (20 Elements)	148
12. Surface Shear (20 Elements).	149
13. Decay of the Velocity Component u with Time (20 Elements).	150
14. Nodal Order of the Quadratic Element.	151
15. Area Coordinates	152
16. Modified Quadratic Element Solution.	153
17. Comparison of Different Elements and Different Grids for the Suddenly Accelerated Plate.	154
18. Convergence of Different Elements' Grid to the Accurate Solution.	155
19. Time Marching Results for the Stream Function ψ	156
20. Time Marching Results for ψ_x .	157

	<u>Page</u>
21. Time Marching Results for ψ_{xx} .	158
22. Time Marching Results for ψ_{xy} .	159
23. Time Marching Results for ψ_{yy} .	160
24. Linearized versus Exact Solution for the Steady Flow over a Flat Plate.	161
25. Flow Chart for the Steady Flow.	162
26. Flow Chart for the Unsteady Flow - Primitive Variables Formulation.	163
27. Flow Chart for the Unsteady Flow - Stream Function Formulation.	164

LIST OF TABLES

	<u>Page</u>
1. Steady Flow - F.E.M. Applications.	165
2. Unsteady Flow - F.E.M. Applications.	168
3. Finite Element - Galerkin Results for the Blasius Problem.	170
4. Finite Element Solution for Suction Flow, $C = .5$	171
5. Finite Element Solution for Suction Flow, $C = 1.5$	172
6. Finite Element Solution for Suction Flow, $C = 3$	173
7. Finite Element Solution for Suction Flow, $C = 5$	174
8. Finite Element Solution for Blowing Flow, $C = -.5$	175
9. Finite Element Solution for Stagnation Flow ($\beta = 1$)	176
10. Finite Element Solution for Wedge Flow ($\beta = .5$)	177
11. Finite Element Solution for Wedge and Stagnation Flows.	178
12. Velocity u at $T = 1.2$ (20 Elements).	179
13. Choice of Time Steps - Quadratic Element.	180
14. Velocity Component u (T) - (20 Elements).	181
15. Comparison between two Grids for values of $u(1.2)$.	182
16. Surface Shear (20 Elements).	183
17. Residuals of the Continuity Equation and the Predicted Relative Errors Associated with the Nodal Values at $T = 1$ (Rayleigh).	184
18. Integrals Associated with Area Coordinates.	185
19. Element Matrix $[D_b]$ for the Continuity Equation.	187
20. Transformation Matrix $[TT]$.	188
21. Element Matrix $[AD_2]$.	189
22. Element Matrix $[AD_2]$.	189
23. Time Steps for the Quintic Element.	190

	<u>Page</u>
24. Comparison between Different Grids and Different Elements for u at $T = 1.2$.	191
25. Comparison between Different Grids and Different Elements for u and Wall Shear at $T = 2$	192
26. Comparison between the Quintic and Quadratic Elements for u and Wall Shear at $T = 3$.	193
27. Comparison of the Computer Time for Different Grids of Different Elements.	194
28. Time Marching Results for ψ (12 Elements).	195
29. Time Marching Results for ψ_x (12 Elements).	196
30. Time Marching Results for ψ_y (12 Elements).	197
31. Time Marching Results for ψ_{xx} (12 Elements).	198
32. Time Marching Results for ψ_{xy} (12 Elements).	199
33. Time Marching Results for ψ_{yy} (12 Elements).	200
34. Linearized versus Exact Solution for the Steady Flow over a Flat Plate.	201
35. Error Estimate in the Linearization of the Suddenly Accelerated Plate Problem.	202

LIST OF PRINCIPAL SYMBOLS

A_m	- area of the element m .
$c_i(t)$	- unknown parameters in the series solution.
e_o	- tolerance allowed in the solution.
e_r	- error associated with the solution.
$f^{(a)}(x_j, t)$	- approximate function for the dependent variable.
$f(x_j, t)$	- actual function for the dependent variable.
f^e	- nodal variables.
f^{e*}	- approximate nodal variables.
f_i	- the i^{th} shape function for the dependent variable.
$F(x_j, f, t)$	- left hand side of the differential equation to be solved.
I	- integral of the weighted residuals over the whole domain.
I_e	- integral of the weighted residuals over the finite element domain.
$J(f^e)$	- Jacobian of the matrix for the nodal variables f_e .
K	- number of nodes in a grid.
M	- total number of elements.
M_d	- order of the high derivative in the approximate solution.
M_e	- number of elements connected to the node e .
M_p	- order of the polynomial used for the approximation.
n	- number of nodal variables in the element.
n_e	- total number of nodal variables at the node e .
$\bar{n}_e$	- number of unknown nodal variables at the node e .
n_s	- size of the global matrix.
$N(x_j)$	- trial or shape function

- N_b - total number of known nodal variables in a grid.
- N_f - total number of degrees of freedom in a grid.
- N_t - total number of equations to be solved.
- p - hydrostatic pressure
- $\bar{p}$ - p/ρ
- $q(x_j)$ - positive weighting function associated with least square methods.
- Q_j - j^{th} component of body force per unit mass.
- $R(x_j, t)$ - residual associated with the differential equation after substituting the approximate solution.
- R_e - Reynolds number referred to a reference length.
- R_x - Reynolds number referred to the x coordinate.
- S_{ij} - components of stress tensor
- S_{Tij} - components of Reynolds stress
- t - dimensionless time
- τ - time independent variable
- T - t/x
- u - dimensionless stream wise velocity component.
- u - streamwise velocity component.
- u_j - x_j component of velocity.
- U - a reference velocity.
- U_B - Blasius velocity.
- U_R - Rayleigh velocity.
- $U(x)$ - streamwise potential velocity.
- U_∞ - undisturbed free stream velocity.
- v - dimensionless transversal velocity component.

- v - transversal velocity component.
- V_m - finite element or subdomain.
- $w(x_j)$ - weighting function.
- x - dimensionless streamwise cartesian coordinate.
- x^* - streamwise cartesian coordinate.
- x_j - vector representing the spatial variables in the flow field.
- y - dimensionless transversal cartesian coordinate.
- y^* - transversal cartesian coordinate.
- Y - $y/\sqrt{x}$
- β - wedge angle factor
- ρ - density
- μ - molecular viscosity
- ν - μ/ρ
- ϵ_m - eddy viscosity
- ω - vorticity
- ψ - dimensionless stream function
- ψ - stream function
- θ - $\psi/\sqrt{x}$
- χ - shape function
- ξ
- η } - area coordinates
- τ }

CHAPTER 1

INTRODUCTION

The advent of electronic computation has greatly accelerated the development of sophisticated numerical techniques for the study of fluid flow during the last two decades. Solutions have been obtained for a large variety of problems of practical engineering interest, and with increasing capability of computers this thrust can be expected to continue.

Nevertheless, there are still many problems areas in fluid mechanics in which the conventional numerical schemes have either failed to penetrate or are too inefficient to warrant serious considerations. Quantitative solutions to many of these problems including in particular those derived from the modern nonlinear theories of fluids, will require the development and use of more powerful approximate methods of analysis. The outlook, however, is encouraging, for within the last seven years or so, sufficient evidence has been presented to indicate that the relatively new concept of Finite Elements might well be the most attractive method for the formulation and numerical solution of the most general flow problems. Use of finite element approximation effectively frees the analyst from traditional difficulties associated with irregular geometries, multi-connected domains, and mixed boundary conditions.

The Finite Difference Method (F.D.M.) has been widely used in many numerical solutions, and the results obtained have demonstrated the accuracy and efficiency of the method. However, the applications have been limited due to complexities in the developed computational procedures in some problems that require mesh refinement, either at complex boundaries, or at complex boundary conditions. A considerable amount of work has been done for the potential flow problems, that require Laplace's, Poisson's and Helmholtz's equations as the equations of motion. But this is not the case for the viscous flow which involves often the solution of a system of time-dependent nonlinear partial differential equations with time-and-space dependent boundary conditions. There is no known general solution of the proper form of the Navier-Stokes equations. Appreciable work has been done for special types of flow problems, where the convective terms in the equations are neglected with respect to the viscous terms, hence the source of nonlinearity is eliminated; such problems exist for creeping and lubrication flows. Linearization of the partial differential equations is found in a wide variety of circumstances, but until recently there has been no method of linearization for solving a general fluid flow problem. Many restrictions are imposed on such methods when solutions are required for a highly nonlinear problem. For example, in a flow problem which possesses a large

Reynolds number, the convective terms in the equations of motion are the dominant elements affecting the behaviour of the flow, and modifications of these terms, by using linearization procedures, must be carefully assessed. Besides, if the flow is transient, the errors due to linearization can be accumulated after each time step, and one must accept the fact that the results obtained are inexact and be aware to what extent this linearization can be used. It will be shown later that, besides the errors involved in linearization, many linearization schemes do not succeed in fulfilling the necessary boundary conditions and stability of the flow field.

The F.D.M. has been widely used to solve the system of equations using different kinds of linearization procedures, and many errors have been reported in different occasions. The magnitude of these inaccuracies differs according to the type of problem and the importance of the nonlinear terms in controlling the flow behaviour. Similar inaccurate results have also been reported too in numerical schemes other than the F.D.M., when the equations are linearized.

The successful development of the Finite Element Method (F.E.M.) in solid mechanics over the past fifteen years, has inspired workers to extend its application to fluid mechanics. Besides its ability to accommodate the arbitrary region of interest without the penalty of special treatment or attention to boundary points (as is the case in the F.D.M.); the F.E.M. can handle linear and nonlinear equations accurately. In addition, the restrictions existing in the F.D.M. in choosing the time step for the numerical integration and

the instabilities involved due to a wrong choice, do not appear to be critical when using the F.E.M.. A further advantage of the F.E.M. is that refinement of the network does not lead to numerical or computational difficulties.

Intensive development of the F.E.M. in the areas of fluid flow and heat transfer has already taken place during the past decade, and several texts and bibliographies are now available which emphasize considerable progress. It is quite fair to say that for potential flow problems the lead of the F.E.M., over other numerical methods, has already been established. The outlook is encouraging for similar success with viscous flow problems. A fundamental difference between finite element modules of fluid motion and the motion of solids is that in problems of fluid dynamics finite elements represent spatial rather than material sub-regions of the continuum. Thus, instead of representing finite elements of a fluid material, the elements represent sub-regions in the space through which the fluid moves (i.e. the Eulerian description of motion). Finite element models of velocity field over an element are specified in terms of velocities at nodes in space rather than of nodes. The application of this method is essentially a representation of a continuum by discrete finite elements and setting up governing equations in terms of discrete displacements, velocities, accelerations etc. at node points which interconnect elements. Availability of variational formulations for some of the basic solid mechanics problems encouraged researchers to take advantage of such formulations in applying

the F.E.M. to fluid flow. It should be emphasized however, that variational formulations need not be the only basis for F.E.M. applications. Several attempts to formulate variational principles for non-linear systems have been examined. The variational formulations are found to lack the advantages of genuine variational principles, chiefly because the variational integral is not stationary or because no variational integral exists. The corresponding variational methods of approximation are shown to be equivalent to the more straight forward Galerkin method or another closely related version of the method of weighted residuals.

The F.E.M. is based on the use of series expansions within sub-domains or elements V_m , into which the domain of interest, V , is divided (Fig. 1). For example approximate representation $f^{(a)}(x_j, t)$ of any function $f(x_j, t)$ will be written in terms of unknown parameters $c_i(t)$, within each element: $f^{(a)}(x_j, t) = \sum_{i=1}^l N_i(x_j) c_i(t)$, where there are l unknown parameters and $N_i(x_j)$ are the shape functions. The fact that the unknown parameters $c_i(t)$ may include derivatives at nodes could be implicitly assumed. These unknown parameters are determined generally in three ways: (a) Minimization of a Functional, (b) Method of Weighted Residuals, or (c) Direct Method. These methods are well illustrated in chapter 3.

The solution of the set of equations obtained by the preceding analysis can be effected in three ways⁶²:

- (i) Direct coupling method. In this method all the finite element equations are assembled directly into a single matrix. The method is extremely simple to apply but the bandwidth of the resulting matrix is rather large.
- (ii) Uncouple method. The method can be used when each of the equations in the system can be solved independently. When this cannot be done it is sometimes possible to eliminate or modify one set of equations (e.g. the continuity equation typically) at element shape function level by using different techniques. The remaining equations can then be uncoupled and solved independently. The bandwidth is always smaller than in the direct coupling method but the accuracy is usually less due to the satisfaction of an equation at element rather than domain level.
- (iii) Iterative method. When boundary conditions relating to one variable are not available it is not possible to use the two former methods. For such a situation the procedure used is to assume starting values for this variable and iterate until self-consistent fields are obtained.

The accuracy of the finite element technique is obviously dependent on the type of the shape function selected for each variable. For example when dealing with variational integrals and stream function, we see that the stream function

has derivatives up to the second order and therefore continuity of ψ and its first derivatives between elements is needed to guarantee convergence. The finite element method becomes more attractive if accuracy can be improved while reducing the necessary amount of input data. This could be achieved by using relatively few high order elements which suffice to model problems with high accuracy. On the other hand, a large number of simple elements is clearly advantageous where a complicated boundary shape needs to be modelled.

In handling the time dependent problems, the finite element grid can be either fixed or moving with time. In the latter case the shape function has to be time dependent; this will lead to complication in the computing problem for elements in a general space domain (higher than one dimension in space). In that case the nodes of each element will move with time, hence the boundary conditions will be difficult to apply and more computer storage will be required. Hence it will be preferable to use a grid fixed in time and to specify the variables in terms of their values at nodes in space rather than of nodes.

Lm

In the present research work, a suitable numerical scheme for the solution of the nonlinear partial differential equations associated with steady and unsteady viscous incompressible flow, has been developed and tested carefully. The numerical scheme uses the Finite Element-Galerkin's formulation to discretize the equations without referring to any linearization procedures. An important feature of the developed scheme is the way by which the contribution of each element or subdomain to the residual over the whole domain has been introduced in the nonlinear algebraic equations. Instead of performing the summation of many elemental matrices to form the so called global matrix, the contributions of the elements that share a specified nodal point are added when the nonlinear equation at that node is required to be solved. This is done by identifying the elements by number which are connected to those nodes where the equations are to be solved. By doing this, a column vector will substitute for the two-dimensional global matrix, and hence the computer core required will be reduced by a ratio of $100 - \frac{100}{\sqrt{n_s}}$ where n_s is the size of the two dimension matrix (e.g. for a 100 by 100 matrix, the saving in the core size is 99%). The foregoing scheme has been tested carefully versus other Finite Element and Finite Difference schemes to show its superiority, efficiency and accuracy.

A comprehensive survey with important comments, mainly on

the use of the F.E.M. in fluid flow problems, is presented in Chapter 2. The Finite Element-Galerkin's scheme used is elaborated upon in Chapter 3. By using this scheme, the shape function is used as a weighting function for minimization of the error. The error will be distributed over the whole domain. This distribution is such that, on average, the partial differential equations are obeyed. The numerical solution of the discretized equations of motion are presented in Chapter 4 where a critical assessment of the accuracy and efficiency of the method of solution of the nonlinear equations has been done to show that the Brown's method is the suitable scheme to be used. The Finite Element-Galerkin's scheme has been tested thoroughly for different kind of elements (varying between quadratic and quintic) with different grids. Two schemes are used for the flow variables, velocity-pressure formulation and stream function formulation. Tests have been performed for the steady and unsteady boundary layer equations which are shown in Chapters 5, 6 and 7. Qualitative and quantitative results have been obtained to emphasize the sort and magnitude of error that results due to linearization procedures, together with the instabilities associated with such linearization. This is presented for the steady and unsteady cases in Chapter 8. Important conclusions and recommendations for further research are presented in Chapter 9.

CHAPTER 2

LITERATURE SURVEY2.1 NUMERICAL SOLUTIONS OF THE NAVIER-STOKES EQUATIONS

The Navier-Stokes equations governing the flow of fluids, are known to have applications to a wide range of engineering problems. Complete solution of these equations is very difficult to obtain even with the present computer facilities. The inadequacy of most of the classical analytical methods of mathematical analysis for the treatment of irregular geometries, unsteady flow, shock waves and nonlinear flow phenomena has forced analysts and scientists to seek and investigate new numerical techniques to solve the problem. The Finite Difference Method (F.D.M.) has traditionally been the popular tool for numerical solutions to special problems.¹⁻¹⁴ However, many difficulties have appeared in nonlinear steady and unsteady flow problems. This is mainly due to the importance of the nonlinear convective terms in the equations, any alterations to these terms by different methods of linearization will affect the behaviour of the flow field to an extent proportional to the importance of these terms in the equations.^{4,5} Many restrictions have been imposed in the F.D.M. on the time step for numerical integration of the equations, and different types of instabilities have been found in the literature.^{15,16} Also, the F.D.M. has been found inadequate to handle different problems associated

with complex boundary conditions and irregularities in the flow field. Refinement of the F.D.M. to handle such complex situations appears to be extremely difficult and even impossible.

2.2 FINITE ELEMENTS AND THE FLUID FLOW PROBLEM

The great success of the F.E.M. in solid mechanics¹⁷ has encouraged researchers and scientists to apply the method for fluid flow problems. Successful development of the F.E.M. in different areas of fluid flow and heat transfer has already taken place.¹⁸⁻²⁰ Several selected applications of the above mentioned techniques of using the F.E.M. for flow problems are shown and discussed for the steady and unsteady flow in Tables 1 and 2 respectively. The tables show the wide area of applicability of the F.E.M. in fluid flow problems. It can be safely stated that the F.E.M. satisfies the requirements of generality, flexibility and efficiency. Also, by using isoparametric elements^{21,22} it is possible to analyse the flow in regions of quite arbitrary shape with improved accuracy in the results²³, because of the close conformity to the true geometry of the problem.

From the preceeding Tables and from the wide range of application of the F.E.M. to fluid flow problems, the author would like to emphasize the following conclusions:

1. In general potential flow problems, involving Laplace's, Poisson's and Helmholtz's equations, much work has been done. The F.E.M. has been

assessed for an adequate number of test cases. A general and fruitful discussion of the potential flow problems has been presented by many authors.²⁴⁻²⁹

2. Without the restriction to potential flows, many attempts have been made to solve special type of problems (see Tables 1 & 2. Miscellaneous). It is important also to mention that an appreciable amount of work has been done for the problems of creeping and lubrication flows,³⁰⁻³⁴ with satisfactory results obtained.
3. Difficulties exist (which is also the case for any numerical method, other than F.E.M.) for any type of flow problems far from the inviscid case. For these types of flow, one has to study carefully the kind of formulation necessary to handle the nonlinearity in the equations. Several attempts have been made to formulate restricted variational principles for nonlinear special type of problems. In the author's opinion these types of derived functions are based on the idea of successive linearization procedures, which may give problems for large Reynolds number flows. The straight forward Galerkin's approach should be used in such situations. Bratanow et al.⁷¹⁻⁷⁴ have used a variational principle. Such functional has been obtained by linearizing the equations of motion using a

perturbation technique by assuming that vorticity field is not sensitive to incremental changes in velocity. The validity of this approach does not exist at large Reynolds number. As was noted in (Réf. 35), poor results were obtained by using a first order perturbation method in the Prandtl-Meyer expansion problem to linearize the equation. The Galerkin's approach will be appropriate for that problem. In a recent report,¹²⁸ the primitive dependent variables have been used together with a restricted variational principle to analyse the transient flow in a square cavity. The nonlinear convective terms have been linearized by using their values at the preceding time step, so that these terms will be at the right hand side of the equation of motion as known numbers at each time step iteration. Their method works for low Reynolds numbers up to 400. They reported that as the Reynolds number increases, their method never has a solution and fails to converge, this is because the nonlinear effect dominates the calculations at high Reynolds numbers, and the method whereby these terms were absorbed into the load vectors in the right hand side of the equation of motion was just not adequate. In the

present research, the errors involved in the linearization procedures have been calculated and solutions are obtained for steady as well as for unsteady flow problems, both with and without linearization procedures.

4. Regarding element selection and accuracy, most authors prefer using fully conforming elements, claiming that these types of element have been found superior in structural applications with regard to convergence. As we see from the preceding tables, certain plate bending elements have been used for flow problems. For example Olson^{36,37} has used the high precision triangular plate bending element³⁸. Skiba et al.³⁹ have used the rectangular plate-bending elements of Bogner, Fox and Schmit.⁴⁰ Argyris⁴²⁻⁴⁸ has developed different types of element (Fig. 2, 3) and has used some of them in flow analysis. Also isoparametric elements are used in the solution of heat conduction problems⁴⁹. In the solution of time dependent problems, finite elements in the time domain can also be used⁵¹ in which the shape of the element is a function of time; this will lead to complication in the computing problem for elements in a general space domain (higher than

one dimension in space); thus a grid fixed with time is preferred.

5. The truncation error of the simple triangular finite element has been assessed by Zienkiewicz and Cheung⁵² in their study of the Laplace and Poisson linear equations. By taking the element nodal points equally spaced in both independent variables, it could be shown that, for a specific pattern, the resultant algebraic system is identical to that of the F.D.M. formulation of the same differential equation. To the knowledge of the author, the truncation error for higher order finite elements has not been compared with the error involved in the F.D.M. for general fluid flow problems.
6. The accuracy of the F.E.M. is obviously dependent on the type of the shape function used for the formulation of each variable. The F.E.M. becomes more efficient if accuracy can be improved while reducing the necessary amount of input data. This could be achieved by using relatively few high order elements which suffice to model problems with high accuracy. On the other hand, a large number of simple elements is clearly advantageous where a complicated boundary shape needs to be

modelled. Hence efficiency of the scheme will depend on the type of problem to be solved. Area coordinates are often used to give easier formulation for integrations over the elemental domain.

CHAPTER 3

FINITE ELEMENT GALERKIN MODEL FOR FLUID FLOW

3.1 THE FINITE ELEMENT CONCEPT

The standard method of solution for a set of equations assumes that the true solution f of a dependent variable can be well approximated by a selected trial solution $f^{(a)}$, which when substituted into the governing equations allows the determination of some parameters that would make the trial solution the best possible fit to the true solution. The necessary boundary conditions are taken into consideration by requiring that the trial solution be selected from an appropriate function or functions which satisfy the relevant equations on the boundary. A common class of trial functions is the finite series, which may be linear, power, polynomial or other form, defined directly on the independent variables (space variables) x_j themselves or on functions (e.g. trigonometric) of these. The approximate solution can be written in a finite series as follows:

$$f^{(a)}(x_j, t) = \sum_{i=1}^l N_i(x_j) c_i(t) \quad , \quad (3.1)$$

where N_i is the spatial trial function and c_i are usually instantaneous constants specified at the time t . It will be shown in a later section that the finite element approximation is a particular case of the class of approximation (3.1) in which l is chosen equal to the total number of nodes multiplied by the number of nodal variables of each node, the c_i are the

nodal variables of the system (in number, l) and the N_i are the shape function composed of a selected functional description over a finite element or a subdomain and is zero everywhere else. Certain conditions also have to be satisfied by the N_i to obtain the necessary continuity of the $f^{(a)}$ (and its derivatives if necessary) at the nodes, and across the interelement boundaries. Thus, the finite element approximation can be used in any solution procedure in which the more general approximation (3.1) is used. It is important to note that the so-called shape function is assumed constant with respect to the time t . However, shape functions can also be used in time domain, this will lead to complication in the computing problem for elements, since the finite element grid will be moving in time, hence the boundary conditions will be difficult to apply and more computer storage will be required. It will be preferable to use a grid fixed in time and to specify the variables at nodes in space rather than of nodes.

3.2 METHODS USED TO DETERMINE THE NODAL PARAMETERS

The unknown parameters c_i which may include derivatives at nodes, are determined generally in three ways:

- (1) Minimization of a Functional (Variational Principle) whose associated Euler equations are the partial differential equations required to be solved. This functional is called a variational functional. The variation or the differentiation of this functional with respect to other

parameters is equal to some fixed value, which can be and usually is chosen to be zero.^{60,86} Any auxiliary conditions that must be satisfied as constraints when the variation or differentiation is taken must be fully specified. The functional whose variation vanishes is said to be stationary relative to those functions with respect to which the variation is taken, and also stationary relative to any constraints that are imposed on the system, like boundary conditions. This stationary property of an integral functional implies by the calculus of variations one or more "Euler equations" and "natural boundary and initial conditions". When these match the equations of motion and the specified boundary and initial conditions, then the variational formulation is indeed an equivalent description of the system concerned. The variational method is only applicable to those problems for which a variational principle exists⁶⁰ and has been found, a common situation only when the system of equations is linear and self-adjoint. An important advantage exists for the variational principle which is the elimination of an integration in the variational functional, since the differentiation of the integral will be used to obtain the system of linear algebraic equations. Several attempts to formulate variational principles for nonlinear systems have been examined⁸⁵ but conclude that the principle is not adequate and

does not exist for these problems, and a Galerkin method or another closely related version of the method of weighted residuals, has to be applied.

- (ii) The Method of Weighted Residuals (M.W.R.) in which the error resulting from the substitution of the approximate or trial solution $f^{(a)}$ into the governing differential equation, is minimized to satisfy the field equations. The manner in which the governing equations are satisfied determines the particular subclass of the M.W.R. used. The well known subclasses of the residual methods are Collocation, Sub-Domain, Galerkin, and Least Squares. Of the various possibilities the Galerkin method is the most successful method used. Consider the differential equation

$$F(x_j, f, t) = 0 \quad (3.2)$$

where x_j , $j=1,2,\dots$ are the spatial independent variables, t is the time variable, and f are the dependent variables (including their derivatives if necessary) for which the solution is required. Assume the trial solution $f^{(a)}$ which has to satisfy the Equation (3.2) and the necessary boundary and initial conditions. The residual of the Equation (3.2) will be defined as

$$R(x_j, t) = F(x_j, f^{(a)}, t) \quad (3.3)$$

The approximate solution $f^{(a)}$ is obtained from Equation (3.1). The magnitude of these residuals is a measure of

the extent to which the assumed solution form $f^{(a)}$, satisfies the original differential Equation (3.2) and the boundary and initial conditions. As the number, L , of the approximating functions, N_i , is increased, it is anticipated that these residuals will diminish. The exact solution occurs when they each vanish. As an approximation to this ideal, set the weighted integrals of these residuals equal to zero.

$$I_k = \int_V w_k(x_j) R(x_j, t) dV = 0 \quad k=1, 2, 3, \dots, L \quad (3.4)$$

where L is the total number of the unknown coefficients c_k , V is the bounded region of interest and w_k are the weighting functions. Equations (3.4) represent L distinct ordinary differential or algebraic (either type can be nonlinear) equations for the L unknowns $c_k(t)$. The choice of different weighting and approximation functions corresponds to various criteria in M.W.R.

In the Collocation method, the weighting functions are the Dirac delta functions

$$w_k = \delta(x_{jk} - x_j) \quad (3.5)$$

This implies that the differential Equation (3.2), has to be exactly satisfied at the L collocation points x_{jk} . One aspect of the collocation method is that the integral in Equations (3.4) does not have to be evaluated. Usually the points x_{jk} are the roots of an L -th order polynomial⁸⁶.

and all terms in the field equation are evaluated in terms of the function at the collocation points. For linear problems, it can be proven that the orthogonal collocation method is equivalent to the Galerkin method, and hence the variational method when applicable. Most applications of orthogonal collocation have been for trial functions defined over the whole domain and not on finite elements.

In the Subdomain method, the weighting functions are

$$w_k = \begin{cases} 1 & x_j \text{ in } V_m \\ 0 & x_j \text{ not in } V_m \end{cases} \quad (3.6)$$

where the V_m are subregions of V .

In the method of Least Squares the square of the residual is used in the integral Equations (3.4) to get

$$I_k = \int_V q(x_j) R^2(x_j, t) dV \quad k=1, 2, 3, \dots, L \quad (3.7)$$

where $q(x_j)$ is a positive weighting function. Then the parameters c_k are determined by setting

$$\frac{\partial I_k}{\partial c_k} = 0 \quad k=1, 2, \dots, L \quad (3.8)$$

The relation between $q(x_j)$ and $w_k(x_j)$ can thus be expressed as

$$w_k(x_j) = q(x_j) \frac{\partial R(x_j, t)}{\partial c_k} \quad k=1, 2, \dots, L \quad (3.9)$$

It is noted that, unlike the Galerkin formulation, the Least Squares method will usually require a higher order interpolation function. This may also lead to the re-

quirement of higher order continuity between elements. Another practical disadvantage of the Least Squares method is that it usually requires more integration points than the Galerkin method. Thus it would require more computing time, especially for the three dimensional or two dimensional unsteady problems.

In the Galerkin method, the weighting functions, w_k , are the approximating trial functions $N_k(x_j)$, this will be elaborated in the following section. Many authors have claimed that the Galerkin criterion is the best weighted residual method⁸⁷, and this will be also discussed in the coming section. Whether one uses a Variational Principle, Galerkin, Collocation, Subdomain or Least Squares method, the concept of the residual is useful as an aid in locating elements when piecewise trial functions are used. Hence, in places where the residual is large, a refinement of the finite element grid is required and additional elements have to be added.

- (iii) The Direct Method in which the trial solutions $f^{(a)}$ are substituted directly into an integral equation.⁶² The integral equation typically consists of the energy integrated over the whole domain. This method is of a limited use.

3.3 THE GALERKIN FORMULATION

As mentioned in the foregoing discussion, the Galerkin

method can be applied to any problem, whether or not it is linear and self-adjoint, however, the variational method is only applicable to those problems for which a variational principle exists and has been found, a common situation only when the system of equations is linear and self-adjoint. Many authors have attempted to derive quasi-variational formulations and restricted variational principles, these formulations depart considerably from the classical calculus of variations. To attain these formulations, they have had to distort and depose the definitions from the calculus of variations in order to make progress. Here, in the present research, the author holds the view, with other writers⁸⁵, that the variational formulations obtained in this way are unjustifiably called principles and are all too frequently misinterpreted or misrepresented or even misunderstood. Whenever the variational principle is not applicable to a certain problem, the straight-forward Galerkin procedure is preferable.

As mentioned in the preceeding section, the approximating trial functions N_k are used as the weighting functions in the Galerkin procedure. Hence the integral in Equation (3.4) will be

$$I_k = \int_V N_k(x_j) R(x_j, t) dV = 0 \quad k=1, 2, 3, \dots, L \quad (3.10)$$

The only unknowns in Equations (3.10) are the parameters c_k in $R(x_j, t)$. Certain of these parameters can be evaluated by requiring the series solution to satisfy the applied boundary conditions, the remaining L unknowns are to be obtained by

minimizing the Equations (3.10) which are L equations in L unknowns. These equations will be in general a system of nonlinear ordinary differential equations and are integrated at each time step to get the c_k parameters. The procedure is to be repeated at the following time step until a steady state solution is reached. The concept behind the criterion of equating the integral in Equations (3.10) to zero is as follows: The N_k 's are members of a complete set and R 's are piecewise continuous functions; a fundamental property of a complete set is that each member of the set (N_k) is orthogonal to a continuous function (R) only if that function is identically zero; therefore, if a finite number of members of a complete set are used as approximating functions and each member is made orthogonal to the residuals, then in the limit $L \rightarrow \infty$ the residuals are identically zero. Even though in actual practice the residual is made orthogonal to only a modest number of members, the method bears computational resemblance to quite powerful analytic techniques in the theory of differential equations.

The Galerkin method cannot be easily applied to problems with complicated domains in that it is not feasible to derive approximating functions applicable over the total region and which also satisfy all boundary conditions. However, the use of Galerkin's method within the context of a finite element scheme allows one to use simple approximating functions within small regions and to replace boundary conditions with simple interface continuity conditions for interior elements by expressing the approximating solutions in terms of nodal

values of the basic unknowns rather than in terms of arbitrary parameters c_k . This Finite Element-Galerkin model will be the subject of the following section.

3.4 THE FINITE ELEMENT-GALERKIN MODEL

The principle of the finite element approximation is based on the division of the domain into sub-domains or finite elements (Fig.1), with the dependent variables f_m being approximately represented in a piece-wise fashion across the domain, element by element, in terms of the independent variables x_j and the "nodal values" of f_m (and its derivatives, if required). Since Equation (3.2) is valid at a point, it is similarly valid in an arbitrary subdomain V_m of V , with M subdomains we have

$$V = \sum_{m=1}^M V_m \quad m=1, 2, \dots, M \quad (3.11)$$

hence a local approximation similar to Equation (3.1) can be written for the subdomain V_m as follows:

$$f_m^{(a)}(x_j, t) = \sum_{k=1}^n N_{mk}(x_j) c_{mk} \quad m=1, 2, \dots, M \quad (3.12)$$

where M is the number of subdomains or finite elements, which means that the subscript m constrains the range of Equation (3.12) to the subdomain V_m only. It is assumed that n is the number of nodal variables in V_m . Similarly the residual Equation (3.4) is written in the subdomain V_m as follows

$$I_{mk} = \int_{V_m} w_{mk}(x_j) R_m(x_j, t) dV_m = 0 \quad k=1, 2, \dots, n \quad (3.13)$$

where the unknown nodal variables are included in the residual

R_m given by an equation similar to (3.3) as follows:

$$R_m(x_j, t) = F_m(x_j, f_m^{(a)}, t) \quad (3.14)$$

The method by which the residuals are added and the appropriate boundary conditions will be emphasized in section 6.

The approximate variable f_m can be written in a matrix in the form

$$f_m^{(a)}(x_j, t) = \begin{bmatrix} N_{m1} & N_{m2} & \dots & N_{mk} & \dots & N_{mn} \end{bmatrix} \begin{Bmatrix} c_{m1}^e \\ \vdots \\ c_{mk}^e \\ \vdots \\ c_{mn}^e \end{Bmatrix} \quad (3.15)$$

where the superscript e denotes the variables of the element at the nodes, this means that their location are specified any where within the element or subdomain V_m . Each node can have as many variables as required by the approximate function N_m . Hence the total number of variables for the element is n . It is important to understand that the number n can be equal or greater than the number of nodes. E.g. the number of terms in a polynomial chosen as the approximating function is given by

$$n = (M_p + 1)(M_p + 2) / 2 \quad (3.16)$$

where M_p is the order of the polynomial used. For the linear polynomial, n is equal to 3, hence 3 nodal variables exist and if a triangular element is chosen, the variables are usually

located at the corners of the triangle. By increasing the order of the polynomial, the number n increases and one can specify more nodes in the element, or, maintaining the same number of nodes, one can include derivatives of the variables, or, one can both increase the number of nodes and use derivatives of the variables at nodes. More explanation will be given on the approximating functions in the next section. From here on, the superscript (a) over the variables f will be removed and it is understood that the variable used is an approximating one, hence Equation (3.15) can be written in the form

$$f_m(x_j, t) = \left[N_m \right] \{ c_m^e \} \quad (3.17)$$

Here the time t is included in the nodal variable c_m^e and not in the approximating spatial function N_m for the reasons specified before. Equation (3.17) is the mathematical statement of the finite element description of f_m across the domain V_m , element-by-element. It is obvious that the finite element method is a particular class of discretization procedure by which the original governing equations having infinite degrees of freedom are transformed into approximation equations with finite degrees of freedom. Some applications of the finite element-Galerkin method are shown in Tables 1 and 2 with important comments. Since the parameters c_m^e consist of values of the function f_m and possibly its derivatives at the nodes, Equation (3.17) can be written in the form

$$f_m(x_j, t) = [N_m] \{f_m^e\} \quad (3.18)$$

It is understood that f_m^e are the nodal values of the function f_m and possibly its derivatives if necessary.

3.5 THE FINITE ELEMENT SHAPE FUNCTION

Once the Finite Element formulation procedure has been decided, it is necessary to decide on the trial function N_m within the element. Many choices are possible, ranging from spline functions, which result in a solution which is highly continuous over the whole domain but has only one unknown coefficient c_m per element, to piecewise polynomials, which can be constructed to have varying continuity properties over the domain and several coefficients c_m per element, including derivatives if required. An approximation $f_m^{(a)}$ to a function f_m is, in general, an "interpolation" relation containing functions of the independent variables x_j and functions of f_m and its derivatives at selected points, called nodes, in the domain. If the highest order derivative occurring in the approximation is of order M_d , the approximation is said to be of $(M_d+1)^{th}$ or M_p order. Thus the generalized Lagrangian shape functions which contain only values of the f_m themselves at nodal points (i.e. zeroth order derivatives of f_m) are first order approximations. The generalized Hermitian shape functions are of second (first derivatives exist) or higher order. Continuity of the functions f_m across interelement boundaries as well as at nodes is normally a requirement which must be satisfied.

Also, if derivatives of order greater than first order appear in the solution procedure, and if the elements are to be "conformable", then continuity of appropriate derivatives is also required both at the interelement boundaries and at the nodes. By these requirements of continuity to obtain a conformable element, restrictions occur on the element shape, the number and location of nodal points, and the form of the generalized Lagrangian or Hermitian function used. The solution procedure must converge to the true solution as element size decreases. This implies that the shape function representations of f_m (and its derivatives if required) across an element converge towards the true values as element size decreases. This will be true if the shape functions are sufficiently flexible for any linear variation of f_m and its derivatives to be represented across the element, since such element-wise linear distributions themselves in the limit will tend to the true solution. This will be possible if the shape functions are such that "constant" values of f_m and its derivatives be representable across the element. It is possible if simple element formulations are used that not all of the interface continuity conditions are satisfied. Because of the approximate nature of the finite element method, and because of the possibility of successive refinements in the numerical modeling of the region into consideration, it has been shown in many instances that acceptable results are still obtained despite the lack of satisfaction of all the interface

continuity conditions. This means that non-conforming elements have shown acceptable behaviour in different problems. It is worthy to note also that some of the non-conforming type of elements show better accuracy than the conforming ones for practical element sizes⁸⁸. As mentioned previously, it is possible to derive more complicated element formulations which satisfy a greater number of the foregoing continuity conditions through definition of additional element nodes or of additional nodal variables, e.g., nodal derivatives of the variables. Generally many authors prefer using fully conforming elements, claiming that these kinds of element have been found superior with regard to convergence. Derived boundary conditions can also be used to place restrictions on the approximating functions admitted, and sometimes improvement of results.^{89,90} Usually, several sets of approximating functions are admissible and it is not possible to choose one as the "best". Selecting approximating functions remains somewhat dependent on the user's intuition and experience. It is important to note that the different kind of elements used in structural analysis are applicable to fluid flow problems. This is obvious, because the velocity in the flow equation can have an approximating function similar to the displacement in a structural problem. As mentioned previously, there are no rules to force a unique shape function for a specified problem, many shape functions can be used for the same problem leading to different levels of

accuracy for the solution. Not only that, but, the grid used in a structural problem can be used in a similar flow problem.⁵³

Therefore, we find in the literature (Tables 1 and 2), a wide range of structural elements that are used in many flow problems. For example Olson^{36,37} has used the high precision triangular plate bending element³⁸. Skiba et al.³⁹ have used the rectangular plate-bending elements of Bogner, Fox and Schmit⁴⁰. By using such elements, the integration of the distribution functions are simplified. Mason⁴¹ has studied the use of these elements with three (deflection and slopes), four (added twist) and six (added curvatures) generalized coordinates per node, in plate vibration problems. He recommends the element with four generalized coordinates per node, thus giving the best combination of accuracy and economy in solution. Argyris⁴²⁻⁴⁶ developed linear strain elements TRIM6 for membrane problems, TET10 for three dimensional continua, curvilinear two dimension TRIC and three dimension TEC elements^{47,48} (Fig. 2). For ducted flows Argyris has made use of the TRIAC4 and HERMES4 elements (Fig. 3). These element types are used in conjunction with the potential function and the velocities as unknowns. Also isoparametric elements are used in two and three dimensional flow in a time-stepping solution for heat conduction problems⁴⁹. Thompson²³, reports greatly improved results using isoparametric elements instead of triangular elements to model flow over a curved surface. Taylor⁵⁰ has discussed different types of element and

interpolation functions used. He noticed that modified serendipity quadrilateral elements (are constructed from a series of terms that are products of m^{th} order Lagrange interpolation functions with first order Lagrange interpolation functions) are much more efficient computationally than either the Lagrangian elements or triangular elements. This allows for use of low order elements in zones of relatively small rates of change in the dependent variables and higher order elements in areas of large rates of change. The serendipity class of elements have been shown to produce incomplete polynomials for all orders of elements above three. Finite elements in the time domain could also be used⁵¹ in which the shape of the element is a function of time; this will lead to complication in the computing problem for elements in a general space domain (higher than one dimension in space).

The F.E.M. becomes more efficient if accuracy can be improved while reducing the necessary amount of input data. This could be achieved by using relatively few high order elements which suffice to model problems with high accuracy. On the other hand, a large number of simple elements is clearly advantageous where a complicated boundary shape needs to be modelled. Hence efficiency of the scheme will depend on the type of problem required to solve. Area coordinates are usually used to give easier formulation of the equations.

In the present research a variety of elements of

different orders will be used to test the accuracy of each element. In the steady boundary layer equation, the partial differential equation will be transformed, by changing the dependent and independent variables, to an ordinary differential equation with one independent variable, and thus a one dimensional type element will be used. Two types have been chosen, a third order element with two nodal variables at each node, and a quintic element with three nodal variables at each node. Details of the behaviour of these two elements will be presented in Chapter 5. In the unsteady boundary layer equations, the partial differential equations are presented first with the primitive variables u and v . A two dimensional triangular element with mid-side node, having a total of six nodes each node possessing two nodal variables u and v , is used. The two components of velocity are represented over the triangle element by a quadratic function, and the total number of degrees of freedom is twelve. This is presented in Chapter 6. Finally, the formulation of the partial differential equations is done with the stream function ψ as the dependent variable, resulting in one equation in ψ . For this a high order precision triangular element³⁸ is used with six degrees of freedom at each of the corner nodes making a total of eighteen degrees of freedom, and will be presented in Chapter 7.

3.6 METHOD OF ASSEMBLING THE RESIDUALS

In Galerkin's procedure, the residual equation,

weighted by modes of the solution, vanishes in the mean over the region of interest. Thus Galerkin's method penalizes the equation error according to the behaviour of the solution. Substituting the shape functions in Equation (3.13) we get the weighted residuals over the finite element or subdomain V_m in the form

$$I_{mk} = \int_{V_m} N_{mk}(x_j) R_m(x_j, t) dV_m = 0 \quad k=1, 2, \dots, n \quad (3.19)$$

where the subscript m constrains the range of Equations (3.19) to the subdomain V_m only, and n is the number of nodal variables in the subdomain. For a number of elements equal to M we will have M equations of the type of Equations (3.19) to get a total of M by n equations. This number Mn is actually the summation of the nodal variables of the separate elements M . Due to the formation of the grid, elements share common nodal variables of other elements at their boundaries, hence the number of variables Mn has to be reduced. A better way to define the number of equations necessary to solve the problem is by summing the nodal variables of the grid. If we designate the number of variables at a certain node e by n_e , this number actually represents the "degrees of freedom" of the node e . Let us assume that the grid constituting the domain V is composed of K nodes, hence the total number of degrees of freedom of these nodes is given by

$$N_f = \sum_{e=1}^K n_e \quad (3.20)$$

Certain of the N_f unknown variables can be evaluated by requiring the solution in Equation (3.18) to satisfy the applied boundary conditions. Hence we are left with a number of equations N_t given by

$$N_t = N_f - N_b \quad (3.21)$$

where N_b is the number of the known nodal variables. This means that some of the N_f equations will be eliminated to yield N_t equations in N_t unknowns. The usual way to obtain the N_t equations is as follows:

- (i) Derive the element equation by substituting in the differential equation the approximating function in the element domain. Each equation will consist of a number of square matrices and column vectors representing the nodal variables, and possibly some constant terms.
- (ii) These sets of matrices will be derived for each of the total number of elements, M .
- (iii) Summation of these matrices using Boolean algebra will lead to the global matrices and column vectors composed from the total number N_f of the nodal variables.
- (iv) A number N_b of equations will be eliminated to end up with N_t equations in N_t nodal variables as unknowns.

In the present procedure, no global matrices need be obtained. Thus, instead of performing the summation of many elemental matrices to form the global matrices, the contri-

butions of the elements that share a specified nodal point are added when the equation at that node (one of the N_t equations) is required. This is done by identifying those elements which are connected at a common node. The computer program will call the node, set up the equations at this node, and for each equation the contributions coming from those elements that share the node are added. The program will stop adding contributions when it reads an element with a number zero (details of the computer program are given in Appendix C). The various contributions are obtained from Equations (3.19). By doing this, we will be dealing always with a column vector of size N_t instead of dealing with a global matrix of size n_s equal to N_t^2 , hence the computer core storage required will be reduced by a ratio of $(100 - 100/\sqrt{n_s})$, e.g. for N_t equal 100, the saving in core storage will be 99%. At a node e the residuals are expressed in the form

$$I_{ek} = \sum_{m=1}^{M_e} \left[\int_{V_m} N_{mk} R_m(x_j, t) dV_m \right] = 0 \quad k=1, 2, \dots, \bar{n}_e \quad (3.22)$$

where $\bar{n}_e$ is the number of unknown nodal variables at the node e , hence

$$\bar{n}_e \leq n_e \quad (3.23)$$

M_e is the number of elements that share and contribute to the residual at node e . If $\bar{n}_e$ is equal to zero, this means that all the nodal variables are known at the node e , then the

Equation (3.22) will not be performed for this node. The resulting equations are either algebraic or ordinary differential equations. Details of the types of equations obtained, and the method of solution are given in the following Chapter 4.

CHAPTER 4

NUMERICAL SOLUTION OF THE DISCRETIZEDEQUATIONS OF MOTION4.1 THE FORM OF THE DISCRETIZED EQUATIONS

The finite element-Galerkin formulation will be illustrated by modelling the governing differential equations of flow. In a domain V , the Navier-Stokes equations are (a comma denotes the gradient of a function)

$$\rho a_j - \rho Q_j - S_{ij,i} = 0 \quad (4.1)$$

where a_j is the x_j component of acceleration,

$$a_j = \frac{du_j}{dt} = \frac{\partial u_j}{\partial t} + u_i u_{j,i} \quad (4.2)$$

ρ is the density, u_j is the x_j component of velocity, x_j is the space coordinate, Q_j is the j^{th} component of body force, and S_{ij} is the stress tensor defined for Newtonian fluids as

$$S_{ij} = -p \delta_{ij} + \mu (u_{i,j} + u_{j,i}) + S_{Tij} \quad (4.3)$$

where p is the hydrostatic pressure, μ is the molecular viscosity, δ_{ij} is the Kronecker delta and S_{Tij} is the turbulent stress tensor. Associated with the Navier-Stokes equations, there exist the continuity equation and the equation of conservation of energy. As an example, discretization will be done on the Navier-Stokes equations only; similar procedures can be followed for any type of equation. If we consider the case of laminar flow, then $S_{Tij} = 0$, and u_j , p denote actual velocities and pressure. If turbulent flow were considered, then S_{Tij} should be determined by empirical means and u_j , p will stand for average velocities and pressure. As an example, an approximation of S_{Tij} is

$$S_{Tij} = \rho \epsilon_m (u_{i,j} + u_{j,i}) \quad (4.4)$$

where ϵ_m is the "eddy viscosity"⁹⁰. If the constitutive Equations (4.2, 3) are substituted into Equation (4.1), the governing equations can be written as

$$\frac{\partial u_j}{\partial t} + u_i u_{j,i} + p_{,j} - \nu u_{j,ii} - \frac{1}{\rho} S_{Tij,i} - Q_j = 0 \quad (4.5)$$

where ν is the kinematic viscosity, and $p = p/\rho$.

Equations (4.5) constitute a set of nonlinear partial differential equations in the unknown u_j , p and with which must be associated initial and boundary conditions. The important step now is to know the form of the discretized equation and to choose a suitable numerical scheme for the solution. The laminar flow case will be taken as an example. Equations (4.5) are then written in the form

$$\left(\frac{\partial u_j}{\partial t} + u_i u_{j,i} + p_{,j} - \nu u_{j,ii} - Q_j \right) = 0 \quad (4.6)$$

Hence Equation (4.6) represents the j^{th} component of the incompressible, laminar Navier-Stokes equations.

Let the element interpolation functions be defined by the equation

$$u_j^{(a)} = N_{jk} u_{jk}^e \quad k=1, 2, 3, \dots, n \quad (4.7)$$

summation over the n repeated indices k is understood, n is the number of nodal parameters associated with the approximate variable $u_j^{(a)}$ in the element V_m ; N_j are the shape functions associated with the velocity component u_j ; u_j^e are the n nodal parameters.

The subscript m is omitted from Equation (4.7) but it is understood that the equation is written for the element V_m . For the pressure we have

$$p^{(a)} = P_k p_k^e \quad k=1, 2, 3, \dots, n \quad (4.8)$$

Summation over the the n repeated indices k is understood, n is the number of nodal parameters associated with the approximate variable $p^{(a)}$ in the element V_m ; P are the shape functions associated with the pressure p ; p^e are the n nodal parameters. Equations (4.7,8) emphasize the fact that we can use mixed interpolation functions for different variables and we are not restricted to one shape function only. This fact is important, because in practice, the variables in a differential equation have different order of differentiation, and it is necessary to have an interpolation function of an order proportional to the order of differentiation of the variable to maintain accuracy of the solution (see chapter 3). Since the body force Q is for most problems is prescribed, it does not require element formulation, although variation of this term can be accommodated in the above manner. Substitute the Equations (4.7,8) into Equation (4.6) to get the j^{th} component of the Navier-Stokes equations as follows;

$$\begin{aligned} \frac{du_j^e}{dt} N_{jk} + u_{ik}^e N_{ik}^T N_{jk,i} u_{jk}^e + P_{k,j} p_k^e \\ - \nu N_{jk,ii} u_{jk}^e - Q_j = 0 \end{aligned} \quad (4.9)$$

Note that u_j^e are defined at the nodes, hence they are only functions

of time in the element and not functions of space, and so the partial derivative with respect to time will be a total derivative as shown.

Equation (4.9) represents the residual of the Navier-Stokes equations which is similar to Equation (3.14). Applying the Galerkin's procedure Equation (4.9) will be weighted by the shape functions N_j to get the weighted residual over the domain V_m , as follows for the l^{th} equation.

$$I_{ml} = \int_{V_m} N_{jl} \left\{ \frac{du^e}{dt} N_{jk} + u^e_{ik} N^T_{ik} N_{jk,i} u^e_{jk} + p_{k,j} p^e_k - \nu N_{jk,ii} u^e_{jk} - Q_j \right\} dV_m = 0 \quad (4.10)$$

Equation (4.10) can be written in the final form as

$$I_m = A \frac{du^e}{dt} + \bar{B} u^e_j + C p^e + D u^e_j - E = 0 \quad (4.11)$$

where A, C, D and E are constant known matrices of integration, $\bar{B}$ is a matrix function of u^e_i which are unknown, and so nonlinear terms will occur in $\bar{B} u^e_j$. Following the same procedure of assembling residuals (section 3.6) from different elements, a summation will be done on Equation (4.11) to get

$$I_l = \sum_{m=1}^{M_e} I_{ml} \quad l = 1, 2, \dots, L \quad (4.12)$$

where M_e is as before the number of elements that contribute to the residual of the l^{th} equation, and L is the total number of equations required to obtain the L unknown parameters. Equations (4.12) represent a set of first order nonlinear ordinary differential equations in time t, the solution of which will be discussed in

the next two sections.

4.2 SOLUTION TECHNIQUE FOR STEADY FLOW

In the steady flow condition, Equations (4.11) will constitute a set of nonlinear algebraic equations. To obtain numerical solutions to these equations, one can find many different techniques in the literature, each with its advantages and disadvantages.

Besides adequacy for use in a digital computer, the chosen numerical scheme should provide high order of accuracy, fast and successful convergence, and require minimum computer core storage and time.

Several books detail the different methods used for the treatment of the nonlinear equations.⁹¹⁻⁹⁵ Two important concepts are the

basis of different numerical methods; the direct solution method, and the iterative method. The problem at hand is to solve a system of L nonlinear algebraic equations for the L unknown nodal variables f^e . From (4.11) these equations are written in the form

$$I_l(f_1^e, f_2^e, \dots, f_l, \dots, f_L^e) = 0 \quad l=1, 2, \dots, L, \quad (4.13)$$

where a nonlinear relation of the variables exists in the L equations.

A method for solving the system (4.13) will be said to be "direct" if it reduces the problem to the solution of an equivalent single equation

$$G(c) = 0 \quad (4.14)$$

in one unknown. That is, relationships

$$\left. \begin{aligned} f_l^e &= f_l^e(c), \quad l=1, 2, \dots, L \\ c &= c(f_1^e, f_2^e, \dots, f_l^e, \dots, f_L^e) \end{aligned} \right\} \quad (4.15)$$

exist such that if $c = c^*$ is a solution of (4.14), then

$$\left. \begin{aligned} f_1^{e*} &= f_1^e(c^*), f_2^{e*} = f_2^e(c^*), \dots, \\ f_l^{e*} &= f_l^e(c^*), \dots, f_L^{e*} = f_L^e(c^*) \end{aligned} \right\} \quad (4.16)$$

satisfy (4.13), and also

$$c^* = c(f_1^{e*}, f_2^{e*}, \dots, f_l^{e*}, \dots, f_L^{e*}) \quad (4.17)$$

is a solution of (4.14) for any solution - vector f_e of (4.13).

The most obvious direct method is "elimination". Suppose, for example, that the L^{th} equation of (4.13) may be solved explicitly for f_L^e ,

$$f_L^e = H_L(f_1^e, f_2^e, \dots, f_l^e, \dots, f_{L-1}^e) \quad (4.18)$$

then this result may be substituted into the first $L-1$ equations to obtain the smaller system

$$I_l^{(1)}(f_1^e, f_2^e, \dots, f_l^e, \dots, f_{L-1}^e) = 0 \quad l=1, 2, \dots, L-1 \quad (4.19)$$

If this process can be continued, then one eventually arrives at the single scalar equation

$$I_1^{(L-1)}(f_1^e) = 0 \quad (4.20)$$

in which the unknown f_1^e plays the part of c in (4.14).

In the case of linear systems, elimination presents a practical method for the solution; the intermediate systems and the final Equation (4.20), are all linear, and there is no complexity. For

nonlinear systems, elimination may increase complexity to the point of being useless from a practical standpoint. In addition, there is the possibility that the solution of a given equation for some unknown, as indicated by (4.18), is not unique, so that one must consider a number of reduced systems to preserve equivalence. In general, elimination does not appear to be a promising method for nonlinear systems. However, in a given situation, its use should be investigated, as even a partial elimination of selected unknowns might result in a helpful decrease in the size of the system without a drastic increase in complexity.

The methods which have proved most satisfactory for high-speed computers are "iterative" schemes based upon successive improvement of some initial approximation. A wide range of iterative schemes exist, and among those having the greatest utility for solving equations on automatic computers are the interval-halving, secant, Newton-Raphson, Lin-Bairstow, and Miller methods. The method of interval halving is a useful procedure for finding real roots of Equations (4.13) when a high precision is not required. It has proved especially useful for transcendental equations where the roots are known to be real and positive. In general the secant method can be used where the interval-halving method is used and may be more desirable because of faster convergence. If high precision is required for the roots of Equation (4.13), one of the higher-order methods is recommended. The Lin-Bairstow method is usually used for finding the roots of a polynomial equation, when the

coefficients are real. The Miller method is similar in concept to the secant method with an additional high convergence procedure. The Newton-Raphson method is widely accepted as one of the best methods for solving for the roots. The excellent results that are generally obtained with this method and the simple computational routine justify its popularity. The method applies for complex roots as well as for real roots, and the iterations converge rapidly provided the initial estimate for a root is close enough. Some other methods under the name of "Newton-like" methods are variants of Newton's method, and they provide highly effective schemes in practical solutions. The most successful Newton-like method was derived by Brown⁹⁶. The concept of the Newton-Raphson method and the Newton-like Brown's method will be illustrated. To obtain the unknown nodal values in the L Equations(4.13), we follow this procedure:

a) Choose a starting nodal vector f_k^e which represents a well chosen guess for the actual vector f^e ; k stands for the first approximation, or the first iterate.

b) Define the derivative I_{lj} by

$$I_{lj}(f^e) = \frac{\partial I_l}{\partial f_j^e}, \quad (4.21)$$

where I_l is defined as in (4.13).

c) Construct the matrix $J(f^e)$ as follows;

$$J_{lj}(f^e) = I_{lj}(f^e), \quad 1 \leq l \leq L \text{ and } 1 \leq j \leq L \quad (4.22)$$

Hence the determinant of $J(f^e)$ is the Jacobian of the system (4.13), evaluated for the vector f^e .

d) Define the vector $\bar{I}(f^e)$ by

$$\bar{I}(f^e) = [I_1(f^e) \dots I_L(f^e)]^T \quad (4.23)$$

e) The approximation for the next iterate will be obtained from

$$f_{k+1}^e = f_k^e + d_k \quad (4.24)$$

where d_k is obtained from the solution of a set of simultaneous linear equations defined by

$$J(f_k^e) d_k = -\bar{I}(f_k^e) \quad (4.25)$$

and the set of linear equations is solved using the numerous techniques available, e.g. Gaussian elimination.

f) Check possible convergence to a root by the test

$$|d_{lk}| < e_0, \quad l = 1, 2, \dots, L \quad (4.26)$$

where e_0 is a small positive number, its magnitude define the accuracy required from the system. The Newton-Raphson method is called a quadratically convergent method because the error in the iterate is proportional to the square of the difference between two iterates. The method requires $L^2 + L$ function evaluations per iterative step.

The Newton-Raphson method seems to be impractical when a large set of equations is to be solved; this is because we are

required to compute the Jacobian of the matrix, which are the elements of Equation (4.21), and hence a laborious computation procedure is necessary. To overcome this difficulty, a Newton-like method has been devised⁹⁶, which approximates the partial derivatives in (4.21) by the corresponding "difference quotients". The method reports very good results, particularly if the computation of derivatives requires more time than the corresponding function evaluations. The procedure is also combined with an elimination procedure (as shown in the direct methods) to obtain a rapidly convergent iteration method. This scheme has been used in the present research and is compared with the regular Newton-Raphson scheme to prove that it is a preferable scheme. In the same manner, the new method is a quadratically convergent and the number of function evaluations per iterative step is $L^2/2 + 3L/2$.

4.3 SOLUTION TECHNIQUE FOR UNSTEADY FLOW

There are two broad classes of numerical solution schemes for initial value problems⁹⁷. One class is the explicit method wherein the highest order time derivative at time t_m is expressible as a function of the solutions and their derivative at previous time steps $t_i \leq t_{m-1}$. In this scheme the Equation (4.11) is written in the form

$$\Delta \frac{du_i^e}{dt} = - (\bar{B} u_j^e + C p^e + D u_j^e - E)$$

hence

$$\frac{du_j^e}{dt} = -A^{-1} (\bar{B} u_j^e + C p^e + D u_j^e - E) \quad (4.27)$$

The right hand side of the equation is evaluated at previous time $t_i \leq t_{m-1}$ and the solution at time t_m can be obtained explicitly from (4.27). Another class is the implicit method where the right hand side is also function of time t_m and previous time, hence Equation (4.27) cannot be applied. Many authors⁹⁸⁻¹⁰⁰ have reported that the implicit methods give solutions with better accuracy and less computations than the explicit schemes, and generally they are more popular in actual computation on high-speed digital computers. One further advantage of using an implicit method rather than an explicit method is that the implicit methods can be shown to be absolutely convergent^{97,101,102}. Within the implicit method different approaches exist to solve the problem, three of which are commonly used¹⁰⁰: (a) Forward difference schemes, (b) Backward difference schemes and (c) Mid-difference schemes. Forward differences are sometimes useful when nonlinear terms can be avoided by its use, but the method is found to be too unstable for a general application. Backward difference methods are the most stable but usually less accurate than mid-difference methods. The Crank-Nicholson method is a mid-difference implicit scheme and has shown great accuracy without appreciable restrictions on the time step. This method has been applied in the present research, so that Equation (4.27) takes the form

$$A \left[(u_j^e)_{t+\Delta t} - (u_j^e)_t \right] / \Delta t = -\frac{1}{2} \left[(\bar{B} u_j^e + C p^e + D u_j^e - E)_{t+\Delta t} + (\bar{B} u_j^e + C p^e + D u_j^e - E)_t \right]$$

or,

$$A \left[(u_j^e)_{t+\Delta t} - (u_j^e)_t \right] + \frac{\Delta t}{2} \left[(\bar{B} u_j^e + C p^e + D u_j^e - E)_{t+\Delta t} + (\bar{B} u_j^e + C p^e + D u_j^e - E)_t \right] = 0 \quad (4.28)$$

We are now in a same situation as the steady flow case, because Equation (4.28) is similar in form to Equation (4.13). The solution can be marched out in time, but within the marching process, one must solve an "equivalent" steady state problem at each time step. Furthermore, if $\bar{B}$ is a nonlinear matrix, as it is in the problem at hand, an iterative solution is required at each time step. A convergence acceleration method similar to the ones described in the preceeding section for steady-state flow problems should be used at each time-step in the marching process for the transient solution. The Newton-like scheme discussed in the preceeding section will be used in the present research to solve the nonlinear algebraic Equations (4.28).

CHAPTER 5

APPLICATION TO STEADY FLOW PROBLEMS5.1 INTRODUCTION

In this chapter a first assessment of the Finite Element-Galerkin model developed in the preceeding chapters will be performed. To do this, comparisons are made of the results obtained for several test cases having steady nonlinear viscous flow, and for which exact or alternative numerical solutions exist in the literature. To present a valuable assessment of the model, the boundary layer equations for an unbounded flow motion over different configurations will be used.

In section 5.2, two kinds of element are presented and compared. The first element is a third order polynomial and the second one is a quintic polynomial.

In section 5.3, the model is assessed, using the two previous elements discussed in section 5.2, and presenting the boundary layer ~~flow~~ over a flat plate. Different numbers of elements have been used to test the convergence of the method. It is shown that six elements of the quintic type over the boundary layer thickness are sufficient to obtain excellent accuracy of the results. The method is compared with previous work by Lynn¹⁰³, who has used twenty uniform length elements in the boundary layer region, and has used the Least Squares error criterion with linearization procedures.

In section 5.4, the model is applied to the suction and blowing flow over a flat plate. The quintic element is used with six elements to represent the boundary layer region. Results are compared with existing numerical solutions.

Finally, in section 5.5, wedge and stagnation two dimensional flow problems are solved using the model, and are compared with existing numerical results.

A paper partly based on the results of this chapter is recently published¹³³ by the author and his supervisor.

5.2 FINITE ELEMENTS USED IN THE ANALYSIS

In the solution of the steady boundary layer flow, the partial differential equations are transformed to an ordinary differential equation in one independent variable. The finite element formulation of this equation will require a one dimensional element type to construct the necessary grid. The highest order derivative involved in the residual equation will be shown to be a third order one. Two elements have been chosen to discretize the flow, a third order element and a quintic element. The flow variable can be written as a function of the independent variable Y as follows:

$$f_m(Y) = \sum_k N_k(Y) c_{mk} \quad k=1,2,\dots,n \quad (5.1)$$

where n , for one dimensional elements is related to the polynomial order M by

$$n = M + 1 \quad (5.2)$$

and the subscript m constrains the range of Equation (5.1) to the subdomain V_m only. The trial or shape function $N(Y)$ is a polynomial given by

$$[N(Y)] = [1, Y, Y^2, \dots, Y^M] \quad (5.3)$$

and the parameters c_k are written in the form

$$\{c\}_m^T = \{c_0, c_1, \dots, c_M\}_m \quad (5.4)$$

Now the unknown parameters c_k are replaced by the unknown variables $f_m(Y)$ at nodal points, i.e. by f^e to get for each subdomain m the following

$$\{c\}_m = [B]_m \{f^e\}_m \quad (5.5)$$

where $[B]_m$ is the inverse of the matrix resulting from the substitution in $N(Y)$ by the nodal coordinates. Substituting Equation (5.5) into Equation (5.1) to get

$$f_m(Y) = [N(Y)] [B]_m \{f^e\}_m \quad (5.6)$$

For the third order element, four unknown variables exist.

Since two nodes exist in the one dimensional element, then two nodal variables are at each node. The only unknown variable in the equation is the function f , hence f^e and f^e are the nodal variables dealing with at each node.

For the quintic element, six unknown variables exist, which are distributed three at each node, hence f^e , f^e and f^e are the variables at each node. The adequacy of each of these elements to

the specified problem depends on the highest order derivative in the equation of motion. As mentioned before, if the highest order derivative is M_d , it is preferable to have continuity of nodal variables up to an order of $(M_d - 1)$. In the boundary layer equation which will be developed later, M_d is equal to three, hence the quintic element is expected to have superior accuracy than the third order element. This will be demonstrated in the next section.

5.3 THE FLOW OVER A FLAT PLATE

In this section the author presents the results obtained for the simple case of boundary layer solution over a flat plate - the Blasius problem.

5.3.1 ANALYSIS

The equation to be solved is Blasius' nonlinear ordinary differential equation⁹⁰,

$$ff'' + 2f''' = 0 \quad (5.7)$$

with the boundary conditions

$$Y=0 : f=0, f'=0$$

and

$$Y=\infty : f' \rightarrow 1 \quad (5.8)$$

where f' stands for the non-dimensional fluid velocity u/u_∞ , Y is the non-dimensional variable $y \sqrt{U_\infty / \nu x}$. Similarity is assured by using Y as the only non-dimensional length variable. Here U_∞ is the undisturbed free stream velocity, ν is the kinematic

viscosity, x and y are co-ordinate axes in the directions along the plate and normal to the plate respectively. The prime in Equations (5.7) and (5.8) stands for the differentiation with respect to the non-dimensional variable Y . Equation (5.8) implies that f'' goes to zero when Y goes to infinity, this derived boundary condition is necessary when using an element with f'' specified as a nodal variable, as the case in the quintic element.

By applying the Galerkin criteria, the weighting functions are

$$\frac{\partial f_m}{\partial \{f^e\}_m} = [B]_m^T [N(Y)]^T \quad (5.9)$$

Hence, for each subdomain m , the residual for the Equation (5.7) is given by Equation (3.19) as:

$$I_m = [B]_m^T \left[\int_0^{s_m} [N(Y)]^T \{f^e\}_m^T [B]_m^T [N(Y)]^T [N''(Y)] dY \right] [B]_m \{f^e\}_m + 2 [B]_m^T \left[\int_0^{s_m} [N(Y)]^T [N''(Y)] dY \right] [B]_m \{f^e\}_m \quad (5.10)$$

where s_m is the length of element m and the prime stands for the differentiation with respect to the non-dimensional variable Y .

The final step is to sum the residuals for the whole domain V . As mentioned in the previous chapters, summation is done only over elements that contribute in the residuals at nodal point. Hence two elements will contribute to the residuals at each node, except for the first and last nodes, where the residuals are due to con-

tributions from the first and last element, respectively. Hence the total residual over the domain is given by Equation (3.22)

as

$$I = \sum_{m=1}^{M_e} I_m = 0 \quad (5.11)$$

where M_e is the number of elements that share and contribute in the residuals at node e . Equation (5.11) is to be applied at each node where the nodal variables are to be found. It will be applied in N_t equations in the whole domain, where N_t is given by Equation (3.21) and is equal to the total number of degrees of freedom N_f in the domain V , minus the known boundary conditions N_b . Equation (5.10) can be written in the form

$$I_m = [G_s]_m \{f^e\}_m + [D_s]_m \{f^e\}_m \quad (5.12)$$

It is obvious that the matrix $[G_s]_m$ is a function of the unknown nodal variables f^e , hence nonlinearity exists in the first term in Equation (5.12). The matrices $[G_s]_m$ and $[D_s]_m$ are developed in Appendix A, for the third order and quintic order elements. The N_t nonlinear equations (5.11), in N_t unknowns f^e have been solved by two different methods. The first method is the Newton-Raphson method and the second method is Brown's method. As described in Section (4.2), the second method has been found more appropriate, since the Jacobian matrix does not need to be supplied. In each of these methods, initial values have to be supplied for the unknown variables f^e . It is known that f is zero at the wall and reaches

the value of l at large Y (which is taken as 10 in this analysis); this suggests a linear distribution for f along Y . For f the initial values are obtained from f as

$$\frac{\Delta f}{\Delta Y} = f \quad (5.13)$$

with f zero at the wall. For f , the important value is the $f(0)$, which is related to the shear stress at the wall and is a maximum there and decreases as Y increases. To get an initial value at $Y=0$, consider an order of magnitude analysis. The dimensionless shear stress can be expressed as⁹⁰

$$f(Y) = \frac{\bar{x}}{\sqrt{R_{ex}}} \frac{\partial \bar{u}}{\partial \bar{y}} \quad (5.14)$$

where $\bar{x} = x/L$, $\bar{y} = y/L$, $\bar{u} = u/U_\infty$ and $R_{ex} = x U_\infty / \nu$.

From the exact solution of the suddenly accelerated plane wall⁹⁰, we find that

$$\delta = 4 \sqrt{\nu t} \quad (5.15)$$

where δ is the boundary layer thickness; t is the time. For our case an approximate δ is obtained by replacing t by x/U_∞ to get

$$\frac{\bar{\delta}}{\bar{x}} = \frac{4}{\sqrt{R_{ex}}} \quad (5.16)$$

$$\text{where } \bar{\delta} = \frac{\delta}{L}$$

From an order of magnitude analysis we can show that

$$\frac{\partial \bar{u}}{\partial \bar{y}} = 0 \left(\frac{1}{\delta} \right) \quad (5.17)$$

Substituting Equations (5.16) and (5.17) into (5.14) we get

$$f''(Y) = 0 \left(\frac{1}{4} \right) \quad (5.18)$$

This value was used as an initial guess at $Y=0$, with linear variation to reach a value of zero at $Y=10$.

5.3.2 RESULTS AND DISCUSSION

The foregoing analysis has been performed for the two different elements. The first is a third order element, hence continuity exists for f and f' but not for f'' . This means that discontinuity exists for the shear stress. Results obtained using six such elements are given in Table 3 and Figure 5, and are of poor accuracy presumably because of this noncompatibility of the shear stress or f'' . By increasing the number of elements or changing the grid shape, negligible improvement was found using this element.*

When a fifth order element was used, much improved results are obtained, even for two elements in the boundary layer. Increasing the number of elements to six, the Blasius results are obtained with an extremely good accuracy. The foregoing results are tabulated in Table 3, after four iterates, and are illustrated in Figure 5. Figure 6 shows the convergence to the Blasius results by increasing the number of iterates. The percentage error in f'' reported by Lynn¹⁰³, where twenty quintic elements were used together with a Least Squares linearization procedure, is 0.008 per cent at $Y = 1.2$.

*The situation can be solved either by reducing the order of f , by an integration by part, or by using a quintic element. The last approach has been used to compare the results with Lynn¹⁰³.

In this work, using six quintic elements, the error is 0.0015 per cent after four iterations, which is also found to be the maximum error in any f'' as shown in Table 3. Computer time on a Xerox SIGMA 9 machine at Carleton University is around three seconds for the grid composed of six elements with the quintic order element. Computer time is with all algebra performed in double precision. The number of nonlinear equations solved for that case is 17 compared to 122 linear equations used by Lynn¹⁰³. These results emphasize the fact that better accuracy is obtained using the developed Finite Element-Galerkin model and solving the nonlinear equations without referring to linearization procedures. Also, the use of a few high order elements, can assure better accuracy than if one use large number of lower order elements.

5.4 SUCTION, BLOWING, STAGNATION AND WEDGE FLOWS

Further assessment of the method is presented for the case of suction, blowing, stagnation, and wedge flows. The quintic order element has been used with different grid sizes, together with Brown's method for the solution of the resulting nonlinear equations.

5.4.1 ANALYSIS

An important property, arising from the solution of boundary layer equations, is the similarity condition. Similar solutions are defined as those for which the component u of the velocity in the direction x along the surface has the property that two velocity profiles $u(x,y)$ located at different coordinates x differ only by

a scale factor in u and y , where y is the direction perpendicular to the surface. Therefore, in the case of such "similar" solutions the velocity profiles $u(x,y)$ at all values of x can be made congruent if they are plotted in coordinates which have been made dimensionless with reference to the scale factors. To reach this similarity property, it was found that the velocity distribution of the potential flow is proportional to a power of the length of arc, measured along the wall from the stagnation point. Such potential flows occur, in fact, in the cases of two- and three-dimensional stagnation flows, and the wedge flow. The equation for the potential flow is written as

$$U(x) = A x^r \quad (5.1)$$

where A is a constant. The relationship between the exponent r and the wedge angle factor β (Figure 7) is given by

$$r = \frac{\beta}{2 - \beta} \quad (5.2)$$

The resulting ordinary differential equation is presented for the dimensionless stream function f in the form

$$f''' + ff'' + \beta(1-f^2) = 0 \quad (5.2)$$

with the boundary conditions

$$y=0 : f=0$$

$$y \rightarrow \infty : f \rightarrow 1$$

(5.2)

For an impermeable surface an additional boundary condition exists,

$$Y=0 : f=0 \quad (5.23)$$

For a permeable surface, the suction or blowing velocity at $Y=0$, $v_o(x)$ which is perpendicular to the surface, is proportional to $x^{(r-1)/2}$, and the particular case for a flat plate, when β is equal to zero, is given by

$$v_o(x) = -\frac{1}{2} C \sqrt{\frac{v U_\infty}{x}}, \quad (5.24)$$

for suction $C > 0$,

for blowing $C < 0$

This boundary condition at $Y=0$ will be given by

$$Y=0 : f=C \quad (5.25)$$

Thus, Equations (5.22) and (5.23) will be the boundary conditions for an impermeable wall, and Equations (5.22) and (5.25) are used for a permeable wall. It is noted also from Figure 7, that stagnation flow exists when $\beta = 1$.

5.4.2. RESULTS FOR SUCTION AND BLOWING

The suction or blowing on a flat plate is governed by Equation (5.21) for $\beta = 0$,

$$f''' + f'' = 0 \quad (5.26)$$

and boundary conditions given by Equations (5.22) and (5.25), the nondimensional variable Y is given by $Y=(y/2) \sqrt{U/vx}$. The elemental residual is similar to Equation (5.12) and written as

$$I_m = \left[G_s \right]_m \{f^e\} + \frac{1}{2} \left[D_s \right]_m \{f^e\} \quad (5.27)$$

where matrices $[G_s]_m$ and $[D_s]_m$ are presented in Appendix A. Initial guesses for the nodal variables are obtained from Blasius solution (zero suction or blowing), as a start for the iteration. The same procedure for the solution of the equations used in Section (5.3) is adopted here. An average of three iterates are required for convergence with a maximum tolerance in the residuals of 0.0004 and an agreement of four decimal digits in the last two iterates. Results are obtained for values of C in Equation (5.24), $C = .5, 1.5, 3, 5$ and $-.5$. These results are presented in Tables 4 to 8, and illustrated on Figure 8. They are compared with the existing numerical solutions of Emmons et al.¹⁰⁴ The average deviation between the present analysis and the existing solutions¹⁰⁴ is zero per cent (2 dec.) for f and f' , and .3 per cent for f'' . The average computer time on the Sigma 9 machine is three seconds with all algebra performed in double precision. Again the results emphasize the high level of accuracy and efficiency of the developed model.

5.4.3 RESULTS FOR WEDGE AND STAGNATION

The flow will be represented by Equation (5.21) together with the boundary conditions specified by Equations (5.22) and (5.23). Wedge flow is given by variable angles as shown in Figure 7. Stagnation flow exists when $\beta = 1$. The nondimensional variable Y is given by $Y = (U/\nu x (2-\beta))^{\frac{1}{2}} y$. The elemental residual is given by

$$I_m = [G_s]_m \{f^0\} + \frac{1}{2} [D_s]_m \{f^0\} + [W_s]_m \quad (5.28)$$

where $[G_s]_m$ and $[D_s]_m$ are defined in Equation (5.12), and

$[W_s]_m$ is obtained from

$$[W_s]_m = \beta [B]_m^T \left[\int_0^s [N(Y)]^T (1 - \{f^e\}_m^T [B]_m^T [N(Y)]^T [N(Y)] [B]_m \{f^e\}_m) dY \right] \quad (5.29)$$

The matrix $[W_s]_m$ is illustrated in Appendix A for the quintic order element. Initial guesses for the nodal variables are obtained from the flat plate solution. The procedures for the assembly of residuals and solution of the nonlinear algebraic equations are the same as in Section (5.3). Results are obtained for the wedge coefficient β equal to 1, .5, .2, 0, -.14, and -.19. An average of three iterates are found to be necessary to attain the accuracy mentioned in Section (5.4.2), with an average time of three seconds on the Sigma 9 computer. Results emphasize the fast convergence, accuracy and efficiency of the method and are well illustrated in Tables 9 to 11, and Figure 9. They are compared with existing numerical solutions.^{105,106} The average difference between the present results and the above mentioned results^{105,106} is: zero per cent for f , zero per cent for f' and .05 per cent for f'' .

CHAPTER 6

APPLICATION TO UNSTEADY VISCOUS FLOW
USING PRIMITIVE FLOW VARIABLES6.1 INTRODUCTION

In this chapter, the Finite Element-Galerkin model developed previously will be tested thoroughly in a transient flow problem. The problem concerned is that of a suddenly accelerated plate in a viscous flow. Since 1951 several authors have attempted to solve this complex initial and boundary value problem, and an adequate solution had not been reached until 1969, when Hall¹⁰⁷ presented his Finite Difference solution. His work has been considered by different authors as a nearly complete solution; this is due to the difficulties encountered by Hall when marching in time to attain the well known steady state Blasius solution. This was because the equations are linearized and the error is accumulative after each time step. It has been recommended by many authors that this problem is an excellent test case for different numerical solutions used. In the present work this problem has been solved using the developed F.E.G. model, and reaches the steady state solution with extremely good accuracy. The present results also satisfy, for the first time, the exponential decay to the Blasius solution described by Stewartson¹⁰⁸. In addition to its mathematical and numerical interest, the present problem is of practical importance in compressible flows. For example, the flow over a semi-

infinite plate following the passage of a shock-wave along the plate, and the problem of boundary-layer flow on the wall of a shock tube, are in fact problems of the same type as the present case. In the following the primitive variables u and v are used, and a comparison with a stream function formulation will be made in the next chapter.

In section 6.2, a review of the status and numerical difficulties encountered in the problem of a suddenly accelerated plate is given. In section 6.3 the governing equations are presented. The finite element discretization and the time marching scheme are detailed in sections 6.4 and 6.5 respectively, followed by the results and discussion in section 6.6. A comprehensive analysis concerning a comparison of different grids and the associated errors is presented in section 6.7.

A paper partly based on the results of this chapter is recently published¹³⁴ by the author and his supervisor.

6.2 A REVIEW OF THE TRANSIENT FLOW OVER A SUDDENLY ACCELERATED PLATE

The problem is concerned with analysis of the incompressible viscous flow in the vicinity of a suddenly accelerated semi-infinite flat plate, which has started to move from rest parallel to its length. The plate is assumed to be of zero thickness with an infinitely sharp leading edge. The mean flow direction is parallel to the plane of the plate and normal to the leading edge. The mathematical formulation of this problem leads to a well known leading edge singularity, where the boundary layer equations are not

satisfied. A second type of singularity, which seems to be rather "weak"¹⁰⁹, in common with other mathematical essential singularities, exists. This singularity appears to be at $T = Ut^*/x^* = 1$, where U , x^* and t^* are the reference plate velocity, the distance measured from the leading edge and the time measured from the start of the motion respectively. Since the analysis by Stewartson¹⁰⁸ on this type of singularity, the problem has been attacked by numerous investigators¹¹⁰⁻¹¹⁴ but without significant results. Farn and Arpaci¹¹⁵ have developed a numerical scheme for the integration of the unsteady two-dimensional boundary layer equations and presented calculations for the case of small oscillations about a steady state flow. Dwyer¹¹⁶ has also developed a numerical scheme of integration for the suddenly accelerated flat plate problem and using a very fine mesh ($\Delta x^* = 0.001$, $\Delta t^* = 0.005$). He reported no difficulty at all when integrating through the point $T = 1$, and concluded that the singular behaviour encountered in analytical solutions should be attributed to the use of the Oseen approximation. These conclusions were also supported by Yoshizawa¹¹⁷. Hall^{107,118} has presented an alternative method based on a mixed explicit-implicit finite difference scheme and has obtained probably the first complete solution for the transient flow over a segment of the impulsively started semi-infinite flat plate problem. His method uses an iteration scheme to improve the initial conditions of the problem, which is based on a similarity property of the flow. His choice for the grid is $\Delta x = \Delta x^*/L = 0.0625$, $\Delta t = U \Delta t^*/L = .125$

initially then was increased after $T = 2$. For the vertical dimension he chose $\Delta y = \Delta y^* Re^{1/2}/L = 0.4\sqrt{2}$ initially then decreasing after each time interval. The terms L , Re , y^* and y stand for the reference length, the reference Reynolds number, the dimensional vertical distance from the wall and its nondimensional parameter (the Blasius variable) respectively. The nonlinear equation for the streamwise velocity u is linearized at each calculation by using the previously calculated values of u . The cycles are repeated until successive values differ by a maximum tolerance of 0.04%. Koob and Abbott^{119,120} have studied the effect of the diffusion term in the flow direction in order to handle the region of singularity at the leading edge. The Reynolds number appears in this work as a parameter, and when increased, the solution drives closer to the singular case. Tani and Yu¹²¹ have presented a method based on the use of the integrated form of the boundary layer equations along with the assumption of one-parameter family of velocity profiles. The transition from the small-time to the large-time solution is accomplished by way of series solution beginning with a fractional power. Dennis¹²² has performed his calculations in the y, T plane by imposing the proper boundary conditions on the flow, i.e. the Rayleigh solution at $T = 1$ and the Blasius solution at a large value of T . He found that the essential singularity defined by Stewartson¹⁰⁸ at $T = 1$ exists. Reddy et al.¹²³ have developed a transformation scheme to allow for the initial conditions to be satisfied as a part of the solution of the transformed boundary

layer equation. The transformed equations are solved by a finite difference scheme known as 'box scheme'. The mesh sizes are

$$\Delta t = 0.025, \quad \Delta x = 0.025 \text{ and } \Delta y = 0.25.$$

6.3 THE GOVERNING EQUATIONS

The unsteady two-dimensional boundary layer equations for a uniform free stream can be written in non-dimensional form as

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (6.1)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \frac{\partial^2 u}{\partial y^2} \quad (6.2)$$

where,

$$\begin{aligned} x &= x^*/L & y &= y^* Re^{1/2}/L & t &= Ut^*/L \\ u &= u^*/U & v &= y^* Re^{1/2}/U & Re &= UL/\nu \end{aligned} \quad (6.3)$$

here u^* and v^* are the dimensional velocities in the x and y directions respectively, and ν is the kinematic viscosity. The boundary conditions for the surface and for the outer boundary layer edge, are

$$\begin{aligned} y = 0: & \quad u = v = 0 \\ y \rightarrow \infty: & \quad u \rightarrow 1 \end{aligned} \quad (6.4)$$

Since the disturbance is convected within the boundary layer by the fluid velocity, hence the flow at a certain station x will not feel the existence of the leading edge for a time t given by $t \leq x$. The solution for $t \leq x$ is then the Rayleigh solution which may be obtained by setting the derivatives with respect to x in Equations

(6.1) and (6.2) equal to zero. Thus we have,

$$T \leq 1 : u = U_R(\zeta) = \frac{2}{\sqrt{\pi}} \int_0^\zeta e^{-\zeta^2} d\zeta \quad (6.5)$$

where

$$\zeta = \frac{y}{2\sqrt{t}} = \frac{y^*}{2\sqrt{v/t}}$$

is the Rayleigh variable. For the other limit, when $t/x \rightarrow \infty$ with x fixed, the Blasius solution exists which may be obtained by setting the derivative with respect to t in Equation (6.2) equal to zero to get,

$$T \rightarrow \infty : u = U_B(Y) \quad (6.6)$$

where

$$Y = \frac{y}{\sqrt{x}} = \frac{y^*}{x^*} \sqrt{\frac{Ux^*}{v}}$$

is the Blasius variable. The number of independent variables can be reduced from three (x, y, t) to two (Y, T) by using $V = v \sqrt{x}$; thus equations (6.1) and (6.2) become

$$Y \frac{\partial u}{\partial y} + T \frac{\partial u}{\partial T} - \frac{\partial V}{\partial Y} = 0 \quad (6.7)$$

$$(1 - Tu) \frac{\partial u}{\partial T} - \left(\frac{Yu}{2} - V\right) \frac{\partial u}{\partial Y} = -\frac{\partial^2 u}{\partial Y^2} \quad (6.8)$$

Equations (6.7) and (6.8) admit a solution for the variables u and V in terms of T and Y alone. However, the solution of these two equations will be difficult due to the Stewartson's singularity, at $T = 1$.

In the present analysis, we attempt to solve Equations (6.1) and (6.2) with the boundary conditions given by Equations

(6.4) and (6.5). This means that Rayleigh solution will be imposed whenever $T \leq 1$.

6.4 FINITE ELEMENT DISCRETIZATION

A twelve degree of freedom element with mid-side nodes is used here. Each of the variables u and v over the element is a piecewise quadratic polynomial. Let the element shape function for the variables u and v be defined by the equation

$$\begin{aligned} \chi = \chi_m(\xi, \eta, \tau) &= [\chi_1 \chi_2 \chi_3 \chi_4 \chi_5 \chi_6] \\ &= [2\xi^2 - \xi \quad 2\eta^2 - \eta \quad 2\tau^2 - \tau \quad 4\xi\eta \quad 4\eta\tau \quad 4\xi\tau] \end{aligned} \quad (6.9)$$

where ξ , η and τ are the area coordinates of the element (see App.B.1.1.) the subscript m constrains the range of equation (6.9) to the element or subdomain m only. Hence u and v are defined by

$$u_m^* = [\chi] \{u^e\}_m = \sum_i \chi_i u_{im} \quad (6.10)$$

$$v_m^* = [\chi] \{v^e\}_m = \sum_i \chi_i v_{im} \quad (6.11)$$

where u_m^* and v_m^* are the approximations for u and v respectively $\{u^e\}_m$ and $\{v^e\}_m$ are the nodal vectors for u and v respectively.

Substitution of (6.10) and (6.11) into (6.1) and applying the Galerkin process in the usual manner yields a set of equations of which the k^{th} equation is

$$\sum_{m=1}^M \left(\iint_{A_m} \chi_k \sum_{j=1}^6 (X_{j,x} u_{jm} + X_{j,y} v_{jm}) dA_m \right) = 0 \quad (6.12)$$

where summation is over all the M elements which contain node k .

A_m is the area associated with the element m , a comma denotes partial differentiation with respect to the following suffix and ℓ is the number of nodal parameters in each element. The Crank-Nicholson time stepping scheme is used to integrate equation (6.2) after the finite element discretization and yields a set of equations of which the k^{th} nonlinear equation is

$$\sum_{m=1}^M \left(A_m \int \chi_k \left[\sum_{j=1}^{\ell} \chi_j (u_{jm} - \bar{u}_{jm}) + \frac{\Delta t}{2} (A_u \sum_{j=1}^{\ell} \chi_{j,x} u_{jm} + \bar{A}_u \sum_{j=1}^{\ell} \chi_{j,x} \bar{u}_{jm} + A_v \sum_{j=1}^{\ell} \chi_{j,y} u_{jm} + \bar{A}_v \sum_{j=1}^{\ell} \chi_{j,y} \bar{u}_{jm} - \sum_{j=1}^{\ell} \chi_{j,yy} (u_{jm} + \bar{u}_{jm})) \right] dA_m \right) = 0 \quad (6.13)$$

where

$$A_u = \sum_{i=1}^{\ell} \chi_i u_{im}, \quad \bar{A}_u = \sum_{i=1}^{\ell} \chi_i \bar{u}_{im} \quad (6.14)$$

$$A_v = \sum_{i=1}^{\ell} \chi_i v_{im}, \quad \bar{A}_v = \sum_{i=1}^{\ell} \chi_i \bar{v}_{im}$$

The bar denotes values obtained from the preceding time step and Δt is the time step which is discussed in the next section. Details of the integration of Equations (6.12, 13) and derivation of the elemental matrices are presented in Appendix B.1. The left hand side of each of the equations (6.12) and (6.13) is called a residual. These residuals are added at each node from the elements which contain that node and the computer program is arranged to sum these residuals without referring to a global matrix for the complete

mesh. This means there is no concern about the bandwidth of the matrix, the numbering of the nodes, and reduces the computer storage considerably. Brown's method⁹⁶ is used to solve the set of equations (6.12) and (6.13). A solution is accepted if either the maximum error in the residuals is less than 0.0001, or if two successive iterates agree in the first four decimal digits. Results show that the average difference of nodal variables between the last two iterates is 0.001%. Whenever a boundary condition exists at a node k , the k^{th} equation is deleted so that the total number of equations is equal to the number of unknowns. The predictors at each time step are obtained from a linear interpolation between the Rayleigh solution at $T = 1$ and the Blasius solution assumed at $T = 10$.

Two different grids are used in this analysis as shown in Figure 10. The first grid, which is the coarser one, is composed of 10 elements in the vertical y direction and of one element width in the streamwise x direction. The second grid is composed of 20 elements in the y direction with one element width in the x direction. For each grid, the nodes at $x = 1$ and $x = 1.5$ are located such that the similarity condition $Y_1 = Y_3$ is satisfied. This means that the y locations at $x = 1.5$ are $\sqrt{1.5}$ times the locations at $x = 1$. The maximum height y chosen for the two grids is the same and equal to $8.5 \sqrt{1.5}$ at $x = 1.5$ and the element width in the x direction is the same for the two grids and equal to 0.5. The vertical mesh size at $x = 1$ for the fine grid is $\Delta y = 0.5$.

and increases to 1.5 far from the wall, for the coarse grid this size is doubled.

6.5 THE TIME MARCHING SCHEME

As mentioned before in Chapter 4, the Crank-Nicholson scheme¹⁰⁰⁻¹⁰² which is a mid-difference scheme will be used in the time marching scheme.

No instabilities have been found at all due to the time step size Δt , as is common in the finite difference methods. Table (12) shows results obtained at $T = 1.2$. In case (a) the time step is 0.025, hence results are obtained after eight time steps. In Case (b) the time step is 0.2, hence results are obtained in the first time step, and despite this large time step, the results are very close to case (a). The choice of the time steps is made so that all the variables at mid-nodes at $x = 1.25$ and at the edge $x = 1.5$ can be obtained from the variables at $x = 1.0$ at the preceding time steps and this is illustrated in Table (13). This choice reduces the number of unknowns to a third of the total nodal variables. At a certain time t , the similarity parameter T has the value of $T_1 = t/1$ at $x = 1$, $T_2 = t/1.25$ at $x = 1.25$ and $T_3 = t/1.5$ at $x = 1.5$ where $T_1 > T_2 > T_3$. As the time increases, the time step is chosen so that T_2 and T_3 take values that T_1 has taken before. From the similarity parameters this means that values for u and V at $x = 1.25$ and $x = 1.5$ can be defined from the variables at $x = 1$ having the same T , if and only if the second similarity parameter Y is satisfied. To satisfy

this, the finite element grid is chosen so that $Y_1 = Y_3$. Thus the two similarity parameters are satisfied at $x = 1$ and $x = 1.5$. Since the nodes at $x = 1.25$ are mid-way located, Y_2 will not match Y_1 . To overcome this difficulty, the same quadratic polynomial is used along $x = 1$ to get values for the variables located at Y_2 whenever the variables are known at Y_1 . The number of unknowns is given by Equation (3.21) as

$$N_t = 2(m+1) - 3 \quad (6.15)$$

where m is the total number of elements in the grid. Hence for a 10 elements grid we have 19 unknowns, and for a 20 elements grid we have 39 unknowns. These unknowns are located at the left edge of the grids shown in Figure 10, which are at nodes 1, 2, 3, ... etc. existing at $x = 1$. Other nodal values at $x = 1.25$ and $x = 1.5$ are obtained, as mentioned before, from preceding time step solutions. The three boundary conditions in (6.15) are obtained from Equation (6.4). Before proceeding the time marching scheme to reach the steady state solution, the residuals are checked at $T = 1$, where the solution is given by Equation (6.5). The residuals were within the value of the tolerance allowed only at mid-nodes, which are at locations 2, 4, ...etc. By a careful check of the partial residual obtained from each term in Equation (6.13), it has been found (see Appendix B.1) that the last term, which represents the variation of the shear stress across the boundary layer, does not contribute at all to the residuals at corner nodes. To cure this situation, a quadratic polynomial has been fitted on

the edge $x = 1$, passing by nodes 1, 2 and 4, from which the unknown at node 3 will be obtained, the procedure is then repeated to get a total of $(m/2-1)$ unknowns at nodes 3, 5, 7, ...etc. An estimation of the deviation involved in this assumption is obtained by writing the quadratic function along the edge $x = 1$ as

$$f = a_0 + a_1 s + a_2 s^2 \quad (6.16)$$

where f is the value of the unknown (u or v), and s is the length along the edge. The deviation at node 3 is given by

$$e_3 = f_1 \left(-1 - s_1^2/s^* (s_1+s_2) + 2 s_1/(s_1+s_2) \right) + f_2 \left(2 - 2s_1/(s_1+s_2) \right) - f_3 + f_4 s_1^2/s^* (s_1+s_2) \quad (6.17)$$

where s_1 and s_2 are the lengths of the side edges 1-2-3 and 3-4-5 respectively (see Figure 10), and s^* is given by

$$s^* = s_1 + s_2/2 \quad (6.18)$$

The deviation has been calculated at Rayleigh and Blasius situations, to get a maximum value of .18 per cent. By this assumption, the residuals at the corner nodes have been reduced to attain the tolerance allowed at each node, which is 0.0001. If this modification is not done, the results will be oscillating around the exact solution (Figure 16).

6.6 RESULTS AND DISCUSSION

Selected results for the velocity component $u(T)$ are shown in Tables 14 and 15 and Figure 11. Table 14 and Figure 11 illustrate

the results obtained for the fine grid composed of twenty elements. It is concluded from these results that the transition from Rayleigh solution to Blasius solution is terminated at $T = 6$, this means that the plate has moved a distance equal to six times the distance from the station into consideration to the leading edge of the plate. From the practical point of view, the transition is almost complete at $T = 4$, where the average difference between $T = 4$ and Blasius solution is .25%. Table 15 presents a comparison between the nodal velocities for two different grids at $T = 1.2$; grid (a) which is composed of twenty elements, and grid (b) composed of ten elements. In spite of its coarseness, the grid (b) shows results very close to grid (a), with an average difference of 1%. When compared to the finite difference method used by Hall 107,118, the fine twenty elements, grid shows an average difference of .25%, and it is noted that the largest difference between the two methods (.5%) is near $Y = 2$. In fact Hall claimed that the highest difference between his iterates existed at $Y = 1.6\sqrt{2}$. This may indicate that the present finite element method is more accurate than the explicit implicit finite difference method used by Hall. In addition the present analysis demonstrated no instabilities in spite of the large time steps used, which may be a further advantage over finite difference methods. Another feature of the present analysis is that at large times $T \geq 4.8$, the difference between the nodal values of u at successive time intervals is decaying exponentially with a factor of $0.0294 T^3$.

(Figure 13), Stewartson¹⁰⁸ has found that a decaying factor of $0.03T^3$ exists at $T > 2.65$. Hall concluded that his method is not able to predict this decay, claiming that the errors involved in his computation were larger than the exponential decay of Stewartson. Table 16 and Figure 12 show the results for the wall shear stress at different time intervals to reach Blasius value at large time. The computer time taken on a CPU SIGMA 9 Xerox machine was .55 minutes for the coarse grid composed of ten elements, and 5.5 minutes for the fine grid using twenty elements with all algebraic in double precision. The time mentioned by Hall was 8 minutes on an IBM 7904. It is also noted that the accuracy of Hall's solution, who solved 1280 linearized equations at each time step, is similar to solving 19 equations in the coarse grid. The number of time steps used by Hall is 64 versus 43 time steps in the present analysis. It is important to note that the number of time steps can be reduced drastically if the triangular element does not possess mid-nodes, hence the computation at $x = 1.25$ will not be needed, this will be emphasized in the next chapter when using the high precision 18 degrees of freedom element, with 6 variables at each node. In conclusion, the foregoing results emphasize the excellent accuracy obtained using the Galerkin's finite element procedure to solve the nonlinear equations associated with the suddenly accelerated plate problem. Since in most viscous flow problems we are concerned with the solution of nonlinear equations, it is anticipated that the foregoing analysis will be the appropriate one to use in such cases.

2

6.7 DETAILS RELATED TO THE CHOICE OF GRIDS

It is very important in any numerical work to set a successful grid size. Also, it will be of importance to minimize the computer time as much as possible. One way to decrease the computation time, is to choose a coarse grid, but we have to make sure that the results obtained by this grid are of practical significance. To have a clear idea about the different meshes used, one has to solve the problem and find the nodal values, then, by comparing these values to the exact ones (assuming that an exact solution exists), a clear idea will be obtained about which grid to use. An alternative way to do this analysis, is to substitute the exact nodal values and calculate the residuals for the different meshes. Then, changing these exact values by a small factor and compute again the residuals. By this way, we save computer time (since the exact solutions are supplied directly, hence there is no need to solve the equations), and also we can predict the error in the nodal values associated with each grid. This alternative has been used by the author to study the mesh used in the problem of a suddenly accelerated plate in a viscous flow. Since, at $T = 1$, the solution is known by the Rayleigh solution, exact nodal values exist at this time. The method will be used to get the residuals for the continuity equation.

At the left side of the grids (FIG. 10), the mid-node is always node 6 in the associated element, and the corner nodes are always either node 1 or node 3 in the associated element. Besides,

the contribution to the residual at the mid-nodes is obviously from one element only, for the corner nodes contributions are added from two or three elements. From Appendix B.1, the integrated residual at a mid-node, associated with an element m can be written in the form

$$\begin{aligned}
 I = & b_1(m) \cdot u(NV1)/10 + c_1(m) \cdot v(NV1)/10 - b_2(m) \cdot u(NV2)/30 \\
 & - c_2(m) \cdot v(NV2)/30 + b_3(m) \cdot u(NV3)/10 + c_3(m) \cdot v(NV3)/10 \\
 & + 4(b_2(m) + b_1(m)/2) \cdot u(NV4)/15 + 4(c_2(m) + c_1(m)/2) \cdot \\
 & v(NV4)/15 + 4(b_2(m) + b_3(m)/2) \cdot u(NV5)/15 + 4(c_2(m) + \\
 & c_3(m)/2) \cdot v(NV5)/15 + 4(b_1(m) + b_3(m)) \cdot u(NV6)/15 + \\
 & 4(c_1(m) + c_3(m)) \cdot v(NV6)/15.
 \end{aligned} \tag{6.19}$$

where NV1, NV2 and NV3 are the corner nodes of the element as shown in Figure 14, starting from the lower left corner and then proceeding in an anti-clockwise direction. NV4, NV5 and NV6 are the mid-nodes, starting by the node between NV1 and NV2, then proceeding in the same direction as the corner nodes. The following equalities exist,

$$\begin{aligned}
 b_1(m) &= (Y(NV2) - Y(NV3))_m \\
 b_2(m) &= (Y(NV3) - Y(NV1))_m \\
 b_3(m) &= (Y(NV1) - Y(NV2))_m \\
 c_1(m) &= (x(NV3) - x(NV2))_m \\
 c_2(m) &= (x(NV1) - x(NV3))_m \\
 c_3(m) &= (x(NV2) - x(NV1))_m
 \end{aligned} \tag{6.20}$$

Equations (6.19) and (6.20) emphasize the linear relation between the residual and the nodal values, this is because the original equation is linear. Assuming now that the mesh dimensions are

fixed, substitute the exact nodal values and calculate the residual. If the residual is not zero (or less than a small tolerance), this means that the nodal values have to be changed from their exact values to suit this mesh. Hence one can expect that if one solves the problem to find the nodal values, these values will not be the same as the exact ones. The question now is how much is the error in the nodal values for a certain residual. This can be predicted by varying the exact nodal values by a certain factor say ± 0.0002 , and watch the effect on the residual. Then, by choosing another factor and get the associated change in the residual. By this way an order of magnitude of the average error which will be associated with the nodal values is obtained. This analysis has been carried out, and computer results are shown in Table 17 for three grids using ten, fourteen and twenty elements.

It is clear from Table 17 that the residuals, and hence the error in the nodal values, decrease as the grid becomes more refined. It is seen that the exact nodal values are satisfied nearly exactly with the twenty elements' grid. The maximum errors in the nodal values associated with the ten, fourteen and twenty element grids are 3%, 2% and .04% respectively. The problem at hand is very sensitive to any variation in the nodal values; thus if one wish to study the exponential decay in the difference between the nodal values at time intervals larger than 5, a very accurate method has to be used since this difference is of the order of $e^{-0.03T^3}$, i.e. less than .15%. The failure in satisfying this

exponential decay has been reported by Hall in his complex explicit-implicit finite difference method, and he mentioned in his analysis that his accuracy is not enough to predict this variation. The twenty elements' grid has been used in this analysis and excellent results have been achieved. To satisfy his curiosity, the author has solved the problem using ten elements and then compared the results with the twenty elements' case at $T = 1.2$. This is shown in Table 15. This table shows that the method used by the author in the foregoing analysis to predict the errors associated with the nodal values is extremely successful. This is shown by comparing Table 15 and Table 17. In Table 17, the average and maximum error in the nodal values are predicted to be 1% and 3% respectively. Table 15 shows that the actual value for these errors are 1% and 3% respectively. It is also of interest to note that the difference between the ten and twenty elements at $T = 1.2$ is nearly the same between Hall's results and the twenty elements' results at $T = 1.5$, and with the same signs. This fact emphasizes that the ten element model will lead to the same accuracy used by Hall. The computer time is shown to vary with the number of equations solved in a cubic manner. The time for ten elements is found to be one tenth of the time of twenty elements. The errors associated with the linearization procedure of this problem is further emphasized in Chapter 8.

CHAPTER 7

APPLICATION TO UNSTEADY VISCOUS FLOW
USING STREAM FUNCTION FORMULATION7.1 INTRODUCTION

In this chapter the higher order precision, 18 degrees of freedom element, developed by Cowper et al.^{38,124} for plate bending problems, will be used to represent the stream function formulation of the unsteady viscous flow. The relative merits of using either the primitive variables or the stream function variable depends on the problem under consideration and the accuracy required from the solution. In formulating the primitive variables model, the critical point is the choice of interpolations for the velocity and pressure. Hood and Taylor⁶² noted that mixed interpolation functions should be used, with the velocity interpolation having one order higher than that for pressure so that consistently accurate results would be obtained for both velocity and pressure. By using the stream function formulation the pressure variable will be eliminated from the equations and there results only one equation of a fourth-order type, with the stream function as the unique variable in the equation. The disadvantage of this fourth-order equation is that the interpolation function needs a higher order compatibility and thus is more complicated in its construction. On the other hand, when a higher order compatibility exists for the different variables, discontinuities between the elements' edges can be eliminated which will

automatically ensure better accuracy and convergence of the solution. Another feature of the high precision element is that all the nodal variables are lumped at the vertices of the triangle without any reference to mid-edge nodes (which have the inconvenience of making the programming more complicated). Also, if the high order derivatives of the variables are necessary as boundary conditions of the problem involved, high precision elements are preferred.

The test case performed in the preceding chapter will be chosen to compare the results for the quadratic element used previously and the quintic high precision element mentioned in the foregoing discussion. In section 7.2 the governing equations in stream function formulation are presented. In section 7.3 the finite element discretization is given in detail, followed by the time marching scheme in section 7.4. The results obtained for the test case of the suddenly accelerated plate, together with comparison with previous results, and discussion are presented in section 7.5.

7.2 THE GOVERNING EQUATIONS

The unsteady two-dimensional boundary layer equations are

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (7.1)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} - \nu \frac{\partial^2 u}{\partial y^2} = 0 \quad (7.2)$$

where u and v are the dimensional velocities in the streamwise x

and transversal y^* directions respectively, and ν is the kinematic viscosity. Now introduce the stream function ψ^* which is related to u^* and v^* as follows

$$u^* = \frac{\partial \psi^*}{\partial y^*} \quad (7.3)$$

$$v^* = - \frac{\partial \psi^*}{\partial x^*} \quad (7.4)$$

By substituting Equations (7.3) and (7.4) into (7.1), the continuity equation is satisfied automatically in a pointwise manner and we are only left with Equation (7.2) written in the form

$$\psi^*_{y^*t^*} + \psi^*_{y^*} \psi^*_{y^*x^*} - \psi^*_{x^*} \psi^*_{y^*y^*} - \nu \psi^*_{y^*y^*y^*} = 0 \quad (7.5)$$

where differentiations of the stream function with respect to the independent variables t^* , x^* and y^* were written in the form

$$\psi^*_{y^*} = \frac{\partial \psi^*}{\partial y^*}, \quad \psi^*_{y^*y^*} = \frac{\partial^2 \psi^*}{\partial y^{*2}}, \quad \dots \text{etc.} \quad (7.6)$$

Now assume a non-dimensional form of the stream function

$$\psi = (\psi^*/UL) \sqrt{Re} \quad (7.7)$$

where L and Re are reference length and Reynolds number respectively, and U is a reference velocity. Using the preceding non-dimensional form of the different parameters shown in Equation (6.3):

$$\begin{aligned} x &= x^*/L & y &= y^*Re^{1/2}/L & t &= \dot{U}t^*/L \\ u &= u^*/U & v &= v^*Re^{1/2}/U & Re &= UL/\nu \end{aligned} \quad (6.3)$$

It can be shown that Equation (7.5) can be written as

$$\psi_{yt} + \psi_y \psi_{yx} - \psi_x \psi_{yy} - \psi_{yyy} = 0 \quad (7.8)$$

It is noted that Equation (7.8) can be obtained directly from Equation (6.2) by using the relations;

$$u = \frac{\partial \psi}{\partial y} \quad (7.9)$$

and

$$v = - \frac{\partial \psi}{\partial x} \quad (7.10)$$

The boundary conditions for the surface and for the outer boundary layer edge, are

$$\begin{aligned} y = 0 & : \psi = \text{constant} \\ y = 0 & : \psi_y = - \psi_x = 0 \\ y \rightarrow \infty & : \psi_y \rightarrow 1 \end{aligned} \quad (7.11)$$

The time conditions are given by

$$T \leq 1 : \psi_y = U_R(\zeta) = \frac{2}{\sqrt{\pi}} \int_0^\zeta e^{-\zeta^2} d\zeta \quad (7.12)$$

which is similar to Equation (6.5). Here the Rayleigh variable is given by

$$\zeta = \frac{y}{2\sqrt{t}} = \frac{Y}{2\sqrt{v t}} \quad (7.13)$$

T is as before a time parameter given by $T = Ut/x = t/x$ and U_R is the Rayleigh velocity. It has been shown previously in Chapter 6 that there exist two independent variables to which all the other variables are related. These two independent variables were Y and T, where $Y = y/x^{1/2}$. By a similar approach, we can reduce the number of independent variables in Equation (7.8) to two instead of three, and hence the similarity will be illustrated. It is noted from

Equations (6.7) and (6.8) that the dependent variables u and V

($V = v x^{\frac{1}{2}}$) are function of only two independent variables T and Y .

This fact guides us to assume another non-dimensional stream function in the form

$$\theta = \frac{\psi}{\sqrt{x}} \quad (7.14)$$

Hence Equation (7.8) is written as follows

$$\theta_{YYY} + \theta_{YY} (\theta/2 - T \theta_T) + \theta_{YT} (T \theta_Y - 1) = 0 \quad (7.15)$$

and the differentiations of the different independent variables are related by

$$\frac{\partial}{\partial y} = \frac{\partial}{\partial Y} \cdot \frac{\partial Y}{\partial y} = \frac{1}{\sqrt{x}} \frac{\partial}{\partial Y} \quad (7.16)$$

$$\begin{aligned} \frac{\partial}{\partial x} &= \frac{\partial}{\partial Y} \cdot \frac{\partial Y}{\partial x} + \frac{\partial}{\partial T} \cdot \frac{\partial T}{\partial x} \\ &= -\frac{1}{2} y x^{-3/2} \frac{\partial}{\partial Y} - \frac{t}{x^2} \frac{\partial}{\partial T} \end{aligned} \quad (7.17)$$

Also, the following equalities exist

$$u = \psi_y = \theta_y \quad (7.18)$$

$$\begin{aligned} V &= v x^{\frac{1}{2}} \\ &= -\psi_x x^{\frac{1}{2}} \\ &= -x^{\frac{1}{2}} \frac{\partial}{\partial x} (x^{\frac{1}{2}} \theta) = -x^{\frac{1}{2}} (\theta x^{-\frac{1}{2}}/2 + x^{\frac{1}{2}} \frac{\partial \theta}{\partial x}) \\ &= -x^{\frac{1}{2}} (\theta x^{-\frac{1}{2}}/2 + x^{\frac{1}{2}} (-\frac{1}{2} y x^{-3/2} \frac{\partial \theta}{\partial Y} - \frac{t}{x^2} \frac{\partial \theta}{\partial T})) \\ &= -\frac{\theta}{2} + \frac{Y}{2} \frac{\partial \theta}{\partial Y} + T \frac{\partial \theta}{\partial T} \end{aligned} \quad (7.19)$$

With these relations the following dependent variables are similar and function of T and Y only:

$$\frac{\psi}{\sqrt{x}} = f_1(Y, T) \quad (7.20.1)$$

$$\psi_x \sqrt{x} = f_2(Y, T) \quad (7.20.2)$$

$$\psi_y = f_3(Y, T) \quad (7.20.3)$$

$$\psi_{xy} x = f_4(Y, T) \quad (7.20.4)$$

$$\psi_{yy} x^{\frac{3}{2}} = f_5(Y, T) \quad (7.20.5)$$

$$\psi_{xx} x^{\frac{3}{2}} = f_6(Y, T) \quad (7.20.6)$$

This means that if Y and T are the same in two different situations (where x , y and t are not the same), similarity exists between the variables in Equations (7.20) in the two situations. This property will lead to the choice of the time step required to save computer time, and this will be illustrated in section 7.4.

7.3 FINITE ELEMENT DISCRETIZATION

A fully conforming high precision¹²⁴ finite element of triangular shape is used in the discretization. This element was developed for plate bending problems³⁸ and tested in steady viscous flow problems³⁶. In fluid flow problems, the stream function is used as the displacement is used in structural problems. Also velocities and velocity gradients replace slopes and curvatures. According to this concept, the properties of this element are as follows: The stream function is represented by a fifth order polynomial. Six deflection parameters, namely the stream function

and its first and second derivatives, are specified at each vertex, yielding a total of 18 degrees of freedom for the element. The variation of the stream function along any edge is then a quintic polynomial in the edgewise co-ordinate. The six coefficients of the latter polynomial are uniquely determined by three quantities (stream function, its edgewise gradient, and its edgewise second derivative) at each of the terminal vertices. Since a general polynomial in two variables depends on $(M_p + 1)(M_p + 2)/2$ constants, where M_p is the order of the polynomial used, hence the quintic polynomial used here is dependent on 21 constants. Thus three additional conditions may be satisfied. This can be done either by adding 3 mid-edge nodes and specifying one variable at each node, or by restraining the variation of certain variables to be a complete polynomial of a specified order on the edge of the element. Other possibilities exist; for example by using a node in the centroid of the element and specifying three variables there. In the element used here¹²⁴, the variation of the gradient of the stream function normal to an edge (velocity parallel to the edge) is restricted to be a cubic function of the edgewise coordinate. This assures continuity of stream function and its gradients (velocities) between two elements that have a common edge. The three additional conditions do not affect terms of fourth or lower degree in the polynomial. The element is therefore capable of representing an arbitrary quadratic variation of vorticity, which is also specified explicitly at the nodal points by

$$\omega = -\nabla^2 \psi = \tau(\psi_{xx} + \psi_{yy}) \quad (7.21)$$

As mentioned previously, area coordinates have computational advantages and will be used. The definition of area coordinates is given in Appendix B.1.1. One of the important advantages of the use of area coordinates is that it makes the formulation of the shape functions easy to write, and since the basic shape functions are directly related to the local generalized stream functions, the inversion of a transformation matrix is avoided. Another advantage of area coordinates is that once the shape functions associated with one vertex are known, the shape functions for the remaining two vertices can be obtained by cyclic permutation of the area coordinates.

The first six shape functions $f_i(\xi, \eta, \tau)$, where ξ , η and τ are the area coordinates as specified in Appendix B.1.1, are chosen as follows:¹²⁵

$$f_1 = 10 \xi^3 - 15 \xi^4 + 6 \xi^5 \quad (7.22.1)$$

$$f_2 = \eta (4 \xi^3 - 3 \xi^4) \quad (7.22.2)$$

$$f_3 = \tau (4 \xi^3 - 3 \xi^4) \quad (7.22.3)$$

$$f_4 = \xi^3 \eta^2 \quad (7.22.4)$$

$$f_5 = \xi^3 \eta \tau \quad (7.22.5)$$

$$f_6 = \xi^3 \tau^2 \quad (7.22.6)$$

All these functions, as well as their first and second derivatives, vanish at vertices 2 and 3. The shape functions f_7 to f_{12} and

f_{13} to f_{18} are obtained from (7.22) by cyclic permutation of ξ , η and τ . In order to derive the quintic polynomial, three additional shape functions are required. These are chosen as

$$f_{19} = \xi^2 \eta^2 \tau^2 \quad (7.23.1)$$

$$f_{20} = \xi^2 \eta \tau^2 \quad (7.23.2)$$

$$f_{21} = \xi^2 \eta^2 \tau \quad (7.23.3)$$

Each of these functions, together with its first and second derivatives, vanishes at all three vertices. Each function has, however, a non-zero normal gradient on one of the three sides of the triangle.

The stream function within the element is represented by a linear combination of the shape functions, thus

$$\psi = \sum_{i=1}^{21} a_i f_i \quad (7.24)$$

Due to the restrictions on the normal gradients along each edge to be a cubic polynomial, values for a_{19} , a_{20} and a_{21} can be expressed as function of a_i ($i=1, 2, \dots, 18$), and Equation (7.24) is written as follows

$$\psi = \begin{bmatrix} f_1 & f_2 & \dots & f_{21} \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} \quad (7.25)$$

where X_1 is a column vector composed of $a_1, a_2, \dots, a_{18}$, and X_2 is a column vector composed of a_{19}, a_{20} and a_{21} . The relation between X_1 and X_2 is

$$\{X_2\} = [C] \{X_1\} \quad (7.26)$$

where $[C]$ is a 3×18 matrix containing the constants of proportionality. The nodal variables at vertex 1 are :

$$\psi_1, \psi_{\eta_1}, \psi_{\tau_1}, \psi_{\eta\eta_1}, \psi_{\eta\tau_1}, \text{ and } \psi_{\tau\tau_1}, \text{ at vertex 1} \quad (7.27.1)$$

and the nodal variables at the 2 other vertices are obtained by cyclic permutation of ξ , η , and τ to get:

$$\psi_2, \psi_{\tau_2}, \psi_{\xi_2}, \psi_{\tau\tau_2}, \psi_{\tau\xi_2} \text{ and } \psi_{\xi\xi_2}, \text{ at vertex 2} \quad (7.27.2)$$

and

$$\psi_3, \psi_{\xi_3}, \psi_{\eta_3}, \psi_{\xi\xi_3}, \psi_{\xi\eta_3} \text{ and } \psi_{\eta\eta_3}, \text{ at vertex 3} \quad (7.27.3)$$

However in the global coordinates the nodal variables are specified at the vertex i by

$$\psi_i, \psi_{x_i}, \psi_{y_i}, \psi_{xx_i}, \psi_{xy_i}, \text{ and } \psi_{yy_i}, \text{ at vertex } i \quad (7.28)$$

The relations between the 2 systems of variables (7.27 and 7.28) can be written in the form

$$\{\psi^e\}_{\xi, \eta, \tau} = [TT]_{x, y} \{\psi^e\} \quad (7.29)$$

or

$$\{\psi^e\} = [TT] \{\psi^e\} \quad (7.30)$$

where ψ^e , ψ^e stand for the nodal variables for element e , in the x - y and ξ , η , τ coordinates respectively.

To get the elements of the matrix $[TT]$ we need to know the relations between the derivatives in the two coordinate systems. Using Equations (B.8) and (B.9), the following can be written

$$\psi_x = \frac{1}{2A} (b_1 \psi_\xi + b_2 \psi_\eta + b_3 \psi_\tau) \quad (7.31)$$

$$\psi_y = \frac{1}{2A} (c_1 \psi_\xi + c_2 \psi_\eta + c_3 \psi_\tau) \quad (7.32)$$

$$\begin{aligned} \psi_{xx} = \frac{1}{4A^2} (b_1^2 \psi_{\xi\xi} + 2b_1b_2 \psi_{\xi\eta} + 2b_1b_3 \psi_{\xi\tau} \\ + b_2^2 \psi_{\eta\eta} + 2b_2b_3 \psi_{\eta\tau} + b_3^2 \psi_{\tau\tau}) \end{aligned} \quad (7.33)$$

$$\begin{aligned} \psi_{yy} = \frac{1}{4A^2} (c_1^2 \psi_{\xi\xi} + 2c_1c_2 \psi_{\xi\eta} + 2c_1c_3 \psi_{\xi\tau} \\ + c_2^2 \psi_{\eta\eta} + 2c_2c_3 \psi_{\eta\tau} + c_3^2 \psi_{\tau\tau}) \end{aligned} \quad (7.34)$$

and

$$\begin{aligned} \psi_{xy} = \frac{1}{4A^2} (b_1c_1 \psi_{\xi\xi} + (b_1c_2 + b_2c_1) \psi_{\xi\eta} + (b_1c_3 + b_3c_1) \psi_{\xi\tau} \\ + b_2c_2 \psi_{\eta\eta} + (b_2c_3 + b_3c_2) \psi_{\eta\tau} + b_3c_3 \psi_{\tau\tau}) \end{aligned} \quad (7.35)$$

The inverse relations will formulate the elements of matrix $[TT]$.

For example at vertex 1, we have

$$\psi_{x_1} = \frac{1}{2A} (b_2 \psi_{\eta_1} + b_3 \psi_{\tau_1}) \quad (7.36)$$

$$\psi_{y_1} = \frac{1}{2A} (c_2 \psi_{\eta_1} + c_3 \psi_{\tau_1}) \quad (7.37)$$

From Equations (7.36) and (7.37) we obtain

$$\psi_{\tau_1} = b_2 \psi_{y_1} - c_2 \psi_{x_1} \quad (7.38.1)$$

and

$$\psi_{\eta_1} = c_3 \psi_{x_1} - b_3 \psi_{y_1} \quad (7.38.2)$$

Similarly, Equations (7.33), (7.34) and (7.35) are written for vertex

1 and solved for $\psi_{\eta\eta_1}$, $\psi_{\eta\tau_1}$ and $\psi_{\tau\tau_1}$ to get

$$\psi_{\eta\eta_1} = c_3^2 \psi_{xx_1} + b_3^2 \psi_{yy_1} - 2 c_3 b_3 \psi_{xy_1} \quad (7.38.3)$$

$$\psi_{\eta\tau_1} = -c_3 c_2 \psi_{xx_1} + (c_3 b_2 + b_3 c_2) \psi_{xy_1} - b_3 b_2 \psi_{yy_1} \quad (7.38.4)$$

$$\psi_{\tau\tau_1} = c_2^2 \psi_{xx_1} + b_2^2 \psi_{yy_1} - 2 b_2 c_2 \psi_{xy_1} \quad (7.38.5)$$

By cyclic permutation of the area coordinates, the relations at the other two vertices are obtained to get the matrix $\begin{bmatrix} TT \end{bmatrix}$ as shown in Table 20. From Equation (7.24), the constants a_i are written in the form

$$\{a\} = \begin{bmatrix} AD \end{bmatrix} \{\psi e^{\sim}\} \quad (7.39)$$

where the matrix $\begin{bmatrix} AD \end{bmatrix}$ is obtained by substitution of the area coordinates at the three vertices for the stream function ψ .

Noting the form of the parameters a_i in Equation (7.25), thus Equation (7.39) can be written in the form

$$\{X_1\} = \begin{bmatrix} AD_1 \end{bmatrix} \{\psi e^{\sim}\} \quad (7.40.1)$$

$$\{X_2\} = \begin{bmatrix} AD_2 \end{bmatrix} \{\psi e^{\sim}\} \quad (7.40.2)$$

From Equations (7.22) and (7.25) we can obtain the following relations at vertex 1:

$$\psi_1 = a_1 \quad (7.41.1)$$

$$\psi_{\eta 1} = a_2 \quad (7.41.2)$$

$$\psi_{\tau 1} = a_3 \quad (7.41.3)$$

$$\psi_{\eta\eta_1} = 2 a_4 \quad (7.41.4)$$

$$\psi_{\eta\tau_1} = a_5 \quad (7.41.5)$$

$$\psi_{\tau\tau_1} = 2 a_6 \quad (7.41.6)$$

Similar relations are obtained at vertices 2 and 3 to get the diagonal matrix $[AD_1]$, whose first three elements are ones, the fourth, fifth and sixth elements are half, one, and half respectively; and this set is repeated twice along the diagonal to constitute the 18 elements of the diagonal matrix. The 3×18 matrix $[AD_2]$ is obtained from the restriction on the normal stream function derivative to be a cubic polynomial¹²⁵. Let us write the matrix $[AD_2]$ in the form

$$[AD_2] = [AR][AD_2^{\sim}] + [AS][AD_2^{\sim\sim}] \quad (7.42)$$

where $[AR]$ and $[AS]$ are 3×3 diagonal matrices whose elements are, a

$$AR(1,1) = (c_1^2 + b_1^2 + c_2^2 + b_2^2 - c_3^2 - b_3^2)/2(c_1^2 + b_1^2) \quad (7.43.1)$$

$$AR(2,2) = (-c_1^2 - b_1^2 + c_2^2 + b_2^2 + c_3^2 + b_3^2)/2(c_2^2 + b_2^2) \quad (7.43.2)$$

$$AR(3,3) = (c_1^2 + b_1^2 - c_2^2 - b_2^2 + c_3^2 + b_3^2)/2(c_3^2 + b_3^2) \quad (7.43.3)$$

and

$$AS(1,1) = (c_1^2 + b_1^2 - c_2^2 - b_2^2 + c_3^2 + b_3^2)/2(c_1^2 + b_1^2) \quad (7.44.1)$$

$$AS(2,2) = (c_1^2 + b_1^2 + c_2^2 + b_2^2 - c_3^2 - b_3^2)/2(c_2^2 + b_2^2) \quad (7.44.2)$$

$$AS(3,3) = (-c_1^2 - b_1^2 + c_2^2 + b_2^2 + c_3^2 + b_3^2)/2(c_3^2 + b_3^2) \quad (7.44.3)$$

The matrices $[AD_2^{\sim}]$ and $[AD_2^{\sim\sim}]$ are then as shown on Table 21, and 22 respectively. From Equations (7.30) and (7.39), the parameters a_i are given by

$$\begin{aligned} \{a\} &= \begin{bmatrix} AD \\ TT \end{bmatrix} \begin{bmatrix} \psi^e \end{bmatrix} \\ 21 \times 1 & \quad 21 \times 18 \quad 18 \times 18 \quad 18 \times 1 \\ &= \begin{bmatrix} B \end{bmatrix} \begin{bmatrix} \psi^e \end{bmatrix} \\ & \quad 21 \times 18 \quad 18 \times 1 \end{aligned} \quad (7.45)$$

Thus Equation (7.25) can be written in the form

$$\psi = \begin{bmatrix} f_1 & f_2 & \dots & f_{21} \end{bmatrix} \begin{bmatrix} B \end{bmatrix} \{\psi^e\} \quad (7.46)$$

or

$$\begin{aligned} \psi &= \begin{bmatrix} f_1 & f_2 & \dots & f_{21} \end{bmatrix} \{a\} \\ &\equiv \begin{bmatrix} f \end{bmatrix} \{a\} \end{aligned} \quad (7.47)$$

In Galerkin's formulation, the weighting function is obtained by

$$\begin{aligned} w &= \frac{\partial \psi}{\partial \psi^e} = \begin{bmatrix} B \end{bmatrix}^T \begin{bmatrix} f_1 & f_2 & \dots & f_{21} \end{bmatrix}^T \\ &\equiv \begin{bmatrix} B \end{bmatrix}^T \begin{bmatrix} f \end{bmatrix}^T \end{aligned} \quad (7.48)$$

Using (7.47) and (7.48), and applying the Crank-Nicholson time stepping scheme (Chapter 4), the Galerkin formulation for the element m residual is obtained as follows;

$$\begin{aligned} I_m &= \begin{bmatrix} B \end{bmatrix}^T \iint_A \begin{bmatrix} f \end{bmatrix}^T \left(\begin{bmatrix} f_y \end{bmatrix} \{a\} - \begin{bmatrix} f_y \end{bmatrix} \{\bar{a}\} + \frac{\Delta t}{2} \right. \\ &\quad \left[\left(\{a\}^T \begin{bmatrix} f_y \end{bmatrix}^T \begin{bmatrix} f_{yx} \end{bmatrix} \{a\} + \{\bar{a}\}^T \begin{bmatrix} f_y \end{bmatrix}^T \begin{bmatrix} f_{yx} \end{bmatrix} \{\bar{a}\} \right) \right. \\ &\quad \left. - \left(\{a\}^T \begin{bmatrix} f_x \end{bmatrix}^T \begin{bmatrix} f_{yy} \end{bmatrix} \{a\} + \{\bar{a}\}^T \begin{bmatrix} f_x \end{bmatrix}^T \begin{bmatrix} f_{yy} \end{bmatrix} \{\bar{a}\} \right) \right. \\ &\quad \left. \left. - \left(\begin{bmatrix} f_{yyy} \end{bmatrix} \{a\} + \begin{bmatrix} f_{yyy} \end{bmatrix} \{\bar{a}\} \right) \right] \right) dA \end{aligned} \quad (7.49)$$

where the bar over the column vector $\{a\}$ denotes values of the parameters a_i at the previous time step. After performing the integrations, Equation (7.49) is written in the form

$$\begin{aligned} I_m &= \begin{bmatrix} B \end{bmatrix}^T \left(\begin{bmatrix} RI \end{bmatrix} \{a - \bar{a}\} + \frac{\Delta t}{2} \left(\begin{bmatrix} RJ \end{bmatrix} \{a\} + \begin{bmatrix} RJO \end{bmatrix} \{\bar{a}\} - \right. \right. \\ &\quad \left. \left. \begin{bmatrix} RK \end{bmatrix} \{a\} - \begin{bmatrix} RKO \end{bmatrix} \{\bar{a}\} - \begin{bmatrix} RL \end{bmatrix} \{a + \bar{a}\} \right) \right) \end{aligned} \quad (7.50)$$

The elemental matrices in Equation (7.50) are detailed in Appendices B.2 and C. The residuals obtained from Equation (7.50) are added at each node from the elements which contain that node, and the computer program is arranged to sum these residuals without referring to a global matrix for the complete mesh. Thus no concern is given by the bandwidth of such a matrix, and the manner by which the nodes are numbered. As mentioned previously, when a boundary condition on a nodal variable exists at node k , the corresponding k^{th} equation is eliminated so that the total number of equations is equal to the number of unknowns. The predictors at each time step are obtained from previous time step, and at the start the Rayleigh solution is known as $T \leq 1$.

Three different grids are used in this analysis to test the convergence of the method. The first grid, which is the coarsest one, is composed of 6 elements in the vertical y direction and of one element width in the streamwise x direction. The second grid and the third grids are composed of 10 elements and 12 elements respectively in the y direction, and one element width in the streamwise direction. The element width in the streamwise direction is the same for the three grids and is equal to .5. The maximum height y chosen for the three grids is identical and equal to $8.5(1.5)^{\frac{1}{2}}$ at $x = 1.5$, as in Chapter 6 where the quadratic element was used. Thus, the only difference between these three grids, and the quadratic element's grids, is the number of elements in the y direction which is here 6, 10, and 12, and previously 10, 14 and 20 with the

quadratic element. Similar to the quadratic element, the nodes at $x = 1$ and $x = 1.5$ are located such that the similarity condition $Y_{(x=1)} = Y_{(x=1.5)}$ is satisfied. This means that the y locations at $x = 1.5$ are $(1.5)^{\frac{1}{2}}$ times the locations at $x = 1$. The vertical mesh size at $x = 1$ for the fine grid is $\Delta y = .5$ and increases to 2.25 far from the wall, and for the coarse 6 elements grid the mesh size is $\Delta y = 1.5$ at the wall and increase to 4 far from the wall. The different nodes' locations are shown in Tables 24 to 26.

7.4 THE TIME MARCHING SCHEME

The Crank-Nicholson scheme (Chapter 4) is used for the time marching scheme as mentioned previously. The choice of the time steps follows the same concept mentioned in Section 6.5; thus it is made so that all the variables at the edge $x = 1.5$ can be obtained from the variables at $x = 1.0$ at the preceding time steps and this is illustrated in Table 23. The Table shows that the number of intervals required to reach the time $T = 7.5$ is equal to 14 intervals. It is important to note that ~~43 intervals were re-~~quired previously with the quadratic element; thus increase was due to the existence of the mid-nodes at $x = 1.25$, and the number of intervals is reduced by one third in the present high precision element. As the time increases, the time step is such that T_2 (T at $x = 1.5$) takes values that T_1 has taken before. From the similarity concept described in section 7.2, all the variables in Equations (7.20) can be defined at the location $x = 1.5$ from the corresponding variables at the preceding time steps, and at

the location $x = 1$. For example, at $t = 2$, $T_1 = t/x_1 = 2$ and $T_2 = t/x_2 = 4/3$, thus if the variables are calculated previously at a time $t = 4/3$ ($T_1 = 4/3$), the variables at $x = 1.5$ when $t = 2$ are now known and will not be treated as unknowns. At the wall the only unknown is ψ_{yy} and all the other variables are zeros. At the boundary layer edge ψ , ψ_x and ψ_{xx} are unknowns, thus the total number of unknowns is $N_t = 6(m/2 + 1) - 8$, where m is the total number of elements in the grid. The values of the second order derivatives* of ψ (ψ_{xy} and ψ_{xx} at wall ; ψ_{xy} and ψ_{yy} at the edge) applied at the boundaries, are derived from Equation (7.11) and they must not be considered as an over specification on the boundary conditions^{62,129}, their existence is necessary only when using a high order element with these derivatives specified as nodal variables. Thus, for 6, 10 and 12 elements, the number of unknowns is 16, 28, and 34 respectively. Before continuing the time marching scheme to reach the steady state solution, the residuals were checked at $T = 1$, where the solution is given by Rayleigh. By increasing the number of elements the residuals decrease and reach the maximum tolerance allowed of 0.0001 with 12 elements in the grid; this proves that the convergence of the model is satisfied.

7.5 RESULTS AND DISCUSSION

The solution has been obtained for different grids, composed from 6 and 12 elements. Comparisons of the different grids for the two kind of elements (quadratic and quintic) are presented in Tables 24 to 26 and illustrated in Figure 17. The

* It is important to note that compatibility of 2nd derivatives exist at nodes only, and not at elemental interfaces.

results show that 12 quintic elements are adequate for obtaining an accuracy similar to the 20 quadratic elements used before. Considering that the solution obtained using 20 quadratic elements or 12 quintic elements is the accurate solution, Figure 18 shows the convergence to that solution for different grids of the above mentioned elements. The results emphasize the high accuracy obtained using the quintic element, together with its fast convergence to the accurate solution. The computer time used for the two different elements with different grids is illustrated in Table 27. The time used in the solution with the quintic element is of the same order of magnitude of the quadratic element, but slightly higher. Although the number of time intervals is reduced to 14 instead of 43 (if using quadratic element), much of the computer time is spent in the generation of the elemental matrices for the quintic element. For the quadratic element, most of the elemental matrices are hand calculated and inputted to the system due to the simplicity existing in their formulation, but this is not the case for the quintic element which requires many algebraic computer manipulations. The time marching results for all the nodal variables are presented in Tables 28 to 33 and illustrated in Figures 19 to 23. In conclusion, the foregoing results emphasize the excellent accuracy obtained using the Galerkin's finite element model developed in this research to solve the nonlinear equations associated with the unsteady viscous flow problem. The high precision finite element¹²⁴ used shows excellent accuracy and rapid convergence, but needed much algebraic

manipulation, which required much computer time.

CHAPTER 8

ERRORS INVOLVED IN THE LINEARIZATION
OF THE EQUATIONS OF MOTION8.1 INTRODUCTION

Linearization of the partial differential equations are found in a wide variety of circumstances, but until recently there has been no method of linearization for solving a general fluid flow problem. Many restrictions are imposed on such methods regarding their applicability for a given problem. Of course, by linearizing the equations it will be easy to solve them, but we must accept the fact that the results obtained are inexact and be aware to what extent this linearized procedure could be used. The most important fluid flow parameter that imposes many restrictions on the concept of linearization is the Reynolds number. For slow viscous flow problems, the inertial forces are neglected and the resulting equations are linear. As the Reynolds number increases, the convective terms become very important. Since these convective terms are the source of nonlinearity in the equations, any alteration in their values by using linearization will be a serious problem which has to be well studied. Since the boundary layer equations are based on the assumption of high Reynolds number, one has to pose the question: to what extent can a linear theory describe effects which are basically nonlinear? It has been reported by Rott et al.¹²⁶ that a linearized perturbation solution is possible for laminar steady

boundary layer problems which belong to the Falkner and Skan family of similarity solutions. Such assumption is valid only for very small changes in the free stream velocity.

In the following sections, the problem of linearization will be studied in detail for the class of steady and unsteady boundary layer problems. The test cases performed in the previous chapters will be taken as an example. In section 8.2, the different types of linearization are discussed. In section 8.3 an application of a type of linearization procedure is done for the steady boundary layer flow over a flat plate. In section 8.4 the unsteady transient flow over the suddenly accelerated plate will be linearized by different procedures and a detailed analysis of the errors involved in such linearizations is performed, and then followed by a discussion of the results in section 8.5.

8.2 METHODS OF LINEARIZATION

Two different approaches are usually used to linearize the equations of motion. The first approach is to assume a linear perturbation solution for the unknown variables in the equations, where the perturbation is taken from a known solution for the variables, and then proceeding to get the higher order solutions. The assumption used in this approach is based on very small perturbations so that we can neglect some terms in the equations with respect to other terms.

The second approach is to linearize the convective terms in the equation of motion using a series solution, such as Taylor's

expansion or others. This linearization can be done either on the original equations before applying any numerical schemes for the solution, or after presenting the equations in a discretized form. This approach is the most common one used in numerical methods.

Linearization by the first approach could fail due to different factors, such as instabilities of the solution or a violation of the boundary conditions required. However, in the second approach, numerical methods when used to solve the problem will always have a solution, and the question now is to what extent these results represent the actual nonlinear problem. In fact, it is known that the time for any bulk property of the fluid, such as vorticity or momentum, to diffuse through a distance x_0 is of the order of x_0^2/ν , which is called the diffusion time. The Reynolds number can be expressed as

$$R_e = \frac{Ux_0}{\nu} = \frac{U/x_0}{\nu/x_0^2} \quad (8.1)$$

$$= \frac{\text{Rate of convection by } U \text{ through a distance } x_0}{\text{Rate of diffusion through a distance } x_0}$$

Hence as the Reynolds number increases, the convective terms in the equations of motion are predominant and are of great importance in judging the behaviour of the flow. Besides, if the problem at hand is transient and is solved by a numerical method at different time intervals, the errors due to the linearization of the convective terms will accumulate step by step during the integration with respect to time.

The foregoing concepts will be further emphasized by presenting an analysis of the order of magnitude of the errors involved in the linearization procedures. This will be performed for steady and unsteady flow.

8.3 ANALYSIS FOR STEADY STATE FLOW

The test case performed in section 5.3 will be used to compare the results for the linear and the actual nonlinear case. The equation to be solved is Balsius' nonlinear ordinary differential equation (Equation 5.7) which is;

$$f f'' + 2 f''' = 0 \quad (5.7)$$

with the boundary conditions shown in Equation (5.8). The nonlinear term in Equation (5.7) will be linearized after the discretization of the equation using Galerkin's finite element model. The linearized equation at the $(k+1)$ th iterates will be written in the form

$$f_k f_{k+1}'' + 2 f_{k+1}''' = 0 \quad (8.2)$$

The error involved in this linearization is

$$\begin{aligned} e_r &= (f_{k+1}' f_{k+1}'' + 2 f_{k+1}''') - (f_k f_{k+1}'' + 2 f_{k+1}''') \\ &= (f_{k+1} - f_k) f_{k+1}'' \end{aligned} \quad (8.3)$$

Hence, for each subdomain m , the residual for the Equation (8.2) is given in the form (see Equation (5.10))

$$\begin{aligned} I_m &= [B]_m^T \left[\int_0^{s_m} [N(Y)]^T \{f_k^e\}^T [B]_m^T [N(Y)]^T [N''(Y)] \right. \\ &\quad \left. dY \right] \cdot [B]_m \{f_{k+1}^e\} + 2 [B]_m^T \left[\int_0^{s_m} [N(Y)]^T \right. \end{aligned}$$

$$\left[N^{(Y)} \right] dY \cdot \left[B \right]_m \{f_{k+1}^e\}_m \quad (8.4)$$

which is similar to Equation (5.12) in the form

$$I_m = \left[G_{s_k} \right]_m \{f_{k+1}^e\}_m + \left[D_s \right] \{f_{k+1}^e\}_m \quad (8.5)$$

The procedure used to sum the residuals and the steps of solution are the same as in section 5.3.

The results are tabulated in Table 34 and compared with the present (which is also exact) solution. They are also illustrated in Figure 24. The computer time needed is found to be 2.2 seconds. The actual nonlinear case needs 3 seconds, which shows that the computer time in the linearized case is reduced by 25% approximately. However, the results show an average error of 17.8% which is obviously extremely high with respect to .91% obtained with a similar number of elements (6) and solving the nonlinear case. It is also noted that the error is quite high when compared to 2 elements in the boundary layer and solving the nonlinear equation (see Table 3). In fact Lynn^{103,127} has used 20 elements in the boundary layer and solved 122 linear equations, using the same element and applying a Least Squares method to solve the linearized equation. Lynn's error is greater than the present method when compared to the exact solution. This emphasizes that the finite element- Galerkin model is found to be extremely accurate and efficient in the nonlinear boundary layer equations and that the results obtained by the linearization procedure are quite inaccurate.

8.4 ANALYSIS FOR UNSTEADY FLOW

The test case for the suddenly accelerated plate, performed previously in chapters 6 and 7, will be studied here to get an order of magnitude of the errors involved due to linearization procedures. The two types of linearization discussed in section 8.2 will be applied in the following sub-sections.

8.4.1 LINEARIZATION USING SMALL PERTURBATIONS

Consider the boundary layer equations for the suddenly accelerated flat plate as shown in Equations (6.1) to (6.6). To introduce a linearization scheme by assuming very small perturbations in the variables, two domains have to be considered. The first domain exists immediately after the start of motion, where the flow at a certain station x will not feel the existence of the leading edge and the solution will be a perturbation from the Rayleigh solution (infinite flat plate impulsively started from rest). In this domain the variables are function of y and t only and the flow is represented by the independent variable ζ , given by Equation (6.5), which is the Rayleigh's variable. The second domain exists at large times, when the transition region is terminated and the flow approaches the steady state Blasius solution; the solution then will be a perturbation from the Blasius case. In this domain the variables are function of y and x only and the flow is presented by the independent variable Y , given by Equation (6.6), which is the Blasius' variable. Further details regarding the attempts made to solve such a problem are given in section 6.2. In this analysis,

the linearization at large time will be considered assuming small perturbations from the Blasius solution. For this case, it is better to formulate the equations by introducing a stream function θ , so that the number of dependent variables will be reduced to one instead of two. The independent variables are Y and T . The reduction is performed in section 7.2 with the resulting equation

$$\theta_{YYY} + \theta_{YY} (\theta/2 - T \theta_T) + \theta_{YT} (T \theta_Y - 1) = 0 \quad (7.15)$$

The corresponding boundary conditions are:

$$\begin{aligned} Y = 0 & : \quad \theta = \frac{\partial \theta}{\partial Y} = 0 \\ Y \rightarrow \infty & : \quad \frac{\partial \theta}{\partial Y} \rightarrow 1 \end{aligned} \quad (8.6)$$

Assuming that $\theta_0(Y)$ is the Blasius steady state solution, the equation for $\theta(Y, T)$ can be written at large times in the following perturbation form

$$\theta(Y, T) = \theta_0(Y) + \epsilon \theta_1(Y) \mu(T) \quad (8.7)$$

where ϵ is a small number compared to unity and representing the deviation from the steady state solution, θ_1 is of the same order of magnitude as θ_0 , and μ represents the time variation. The boundary conditions on θ_0 are the Blasius conditions:

$$\begin{aligned} Y = 0 & : \quad \theta_0 = \frac{d\theta_0}{dY} = 0 \\ Y \rightarrow \infty & : \quad \frac{d\theta_0}{dY} \rightarrow 1 \end{aligned} \quad (8.8)$$

and the boundary conditions on θ_1 are given by

$$\begin{aligned}
 Y = 0 & : \quad \theta_1 = \frac{d\theta_1}{dY} = 0 \\
 Y \rightarrow \infty & : \quad \frac{d\theta_1}{dY} \rightarrow 0
 \end{aligned} \quad (8.9)$$

Also μ goes to zero as T tends to infinity. Substituting from (8.7) into (7.15) to get

$$\begin{aligned}
 & \frac{\partial^3}{\partial Y^3} (\theta_0 + \epsilon \theta_1 \mu) + \frac{1}{2} (\theta_0 + \epsilon \theta_1 \mu) \frac{\partial^2}{\partial Y^2} (\theta_0 + \epsilon \theta_1 \mu) \\
 & + (T \frac{\partial}{\partial Y} (\theta_0 + \epsilon \theta_1 \mu) - 1) \cdot \frac{\partial^2}{\partial Y \partial T} (\theta_0 + \epsilon \theta_1 \mu) \\
 & - T \frac{\partial^2}{\partial Y^2} (\theta_0 + \epsilon \theta_1 \mu) \frac{\partial}{\partial T} (\theta_0 + \epsilon \theta_1 \mu) = 0
 \end{aligned} \quad (8.10)$$

Noting that θ_0 and θ_1 are functions of Y only, thus the partial differential equation will be transformed to an ordinary differential equation in the form;

$$\begin{aligned}
 & \theta_0'''' + \frac{1}{2} \theta_0 \theta_0'' + \epsilon (\mu \theta_1'''' + \frac{1}{2} \mu \theta_0 \theta_1'' + \frac{1}{2} \mu \theta_1 \theta_0'') \\
 & + T \theta_0' \theta_1' \frac{d\mu}{dT} - \theta_1' \frac{d\mu}{dT} - \theta_1' T \theta_0'' \frac{d\mu}{dT} + \epsilon^2 (\frac{1}{2} \theta_1 \mu^2 \theta_1'' \\
 & + \mu T \theta_1'^2 \frac{d\mu}{dT} - \mu T \theta_1 \theta_1'' \frac{d\mu}{dT}) = 0
 \end{aligned} \quad (8.11)$$

Neglecting the terms in ϵ^2 , which are obviously the nonlinear terms we get;

$$\begin{aligned}
 & \theta_0'''' + \frac{1}{2} \theta_0 \theta_0'' + \epsilon (\mu \theta_1'''' + \frac{1}{2} \mu \theta_0 \theta_1'' + \frac{1}{2} \mu \theta_1 \theta_0'') \\
 & + T \theta_0' \theta_1' \frac{d\mu}{dT} - \theta_1' \frac{d\mu}{dT} - \theta_1' T \theta_0'' \frac{d\mu}{dT} = 0
 \end{aligned} \quad (8.12)$$

Since the first order solution is the Blasius one given by

$$2 \theta_0'''' + \theta_0 \theta_0'' = 0 \quad (8.13)$$

thus Equation (8.12) can be further reduced to reach the form

$$\mu (\theta_1^{''''} + \frac{1}{2} \theta_0 \theta_1^{''} + \frac{1}{2} \theta_1 \theta_0^{''}) + \frac{d\mu}{dT} (T \theta_0' \theta_1' - \theta_1' - \theta_1 T \theta_0'') = 0 \quad (8.14)$$

which is subjected to the boundary conditions given by Equation (8.9). Now the solution proceeds by assuming different forms for $\mu(T)$. First assume it in the form

$$\mu(T) = e^{\lambda T} \quad (8.15)$$

where λ is an arbitrary complex number. Substituting (8.15) into (8.14) gives;

$$\theta_1^{''''} + \frac{1}{2} \theta_0 \theta_1^{''} + \frac{1}{2} \theta_1 \theta_0^{''} + \lambda (T \theta_0' \theta_1' - \theta_1' - \theta_1 T \theta_0'') = 0 \quad (8.16)$$

At large times, the terms associated with T possess the dominating effect, thus from Equation (8.16) we have

$$\theta_1' - \frac{\theta_0''}{\theta_0'} \theta_1 = 0 \quad (8.17)$$

But it is known from Blasius' solution that :

$$\begin{aligned} \theta_0' (0) &= 0, & \theta_0' (\infty) &= 1 \\ \theta_0'' (\infty) &= 0 & \text{and } \theta_0'' (0) &= .332 \end{aligned} \quad (8.18)$$

thus Equation (8.17) will violate the necessary boundary conditions on θ_1 given by Equation (8.9). This is obvious when we substitute the values in Equation (8.18) into Equation (8.17), therefore the assumption made in Equation (8.15) is not adequate. Now assume $\mu(T)$ in the form;

$$\mu(T) = T^\lambda \quad (8.19)$$

Substituting Equation (8.19) into Equation (8.14) gives;

$$T^\lambda (\theta_1^{''''} + \frac{1}{2} \theta_0 \theta_1^{''} + \frac{1}{2} \theta_1 \theta_0^{''}) + \lambda T^{\lambda-1} (T \theta_0' \theta_1' - \theta_1' \theta_0' - \theta_1 T \theta_0'') = 0$$

or

$$\theta_1^{''''} + \frac{1}{2} \theta_0 \theta_1^{''} + \frac{1}{2} \theta_1 \theta_0^{''} + \lambda (\theta_0' \theta_1' - \theta_1 \theta_0'') - \lambda \frac{\theta_1'}{T} = 0. \quad (8.20)$$

At large time, the last term in Equation (8.20) is neglected to get

$$\theta_1^{''''} + \frac{1}{2} \theta_0 \theta_1^{''} + \frac{1}{2} \theta_1 \theta_0^{''} + \lambda (\theta_0' \theta_1' - \theta_1 \theta_0'') = 0 \quad (8.21)$$

thus λ can be written in the form

$$\lambda = \frac{\theta_1^{''''} + \frac{1}{2} (\theta_0 \theta_1^{''} + \theta_1 \theta_0'')}{\theta_1 \theta_0' - \theta_0' \theta_1'} \quad (8.22)$$

To know the sign of λ it is required to square the terms in Equation (8.22) composed from θ_1 and its derivatives. To do that the following steps will be performed. First multiply both the nominator and denominator of Equation (8.22) by $\theta_0' \theta_1'$ to get

$$\lambda = \frac{\theta_1^{''''} \theta_0' \theta_1' + \frac{1}{2} (\theta_0 \theta_0' \theta_1' \theta_1^{''} + \theta_0' \theta_0'' \theta_1 \theta_1')}{\theta_1 \theta_1' \theta_0' \theta_0'' - \theta_0'^2 \theta_1'^2} \quad (8.23)$$

Since the boundary conditions for θ_1 and its derivatives are known at $Y=0$ and $Y=\infty$, multiply both the numerator and denominator of Equation (8.23) by J , given by

$$J = e^{0 \int_Y \frac{\theta_0}{2} dY} \quad (8.24)$$

Now integrate from $Y=0$ to $Y=\infty$ to obtain λ in the form

$$\lambda = \frac{I_1}{I_2} \quad (8.25)$$

where I_1 and I_2 are written in the form

$$I_1 = I_{11} + I_{12} + I_{13} \quad (8.26)$$

$$I_2 = I_{21} + I_{22}$$

I_{11} is given by

$$I_{11} = \int_0^{\infty} \theta_1'''' \theta_0' \theta_1' J dY,$$

integrating by parts to get

$$\begin{aligned} I_{11} &= \left[\theta_1''' \theta_0' \theta_1' J \right]_0^{\infty} - \int_0^{\infty} \theta_1''' (\theta_0''' \theta_1' J + \theta_0'' \theta_1'' J \\ &\quad + \theta_0' \theta_1' \frac{\theta_0}{2} J) \\ &= 0 - \int_0^{\infty} (\theta_1''' \theta_0''' \theta_1' J + \theta_1'''^2 \theta_0'' J + \theta_0' \theta_1' \theta_1''' \frac{\theta_0}{2} \\ &\quad J) dY \end{aligned} \quad (8.27.1)$$

I_{12} is given by

$$I_{12} = \frac{1}{2} \int_0^{\infty} (\theta_0' \theta_0'' \theta_1' \theta_1'' J) dY \quad (8.27.2)$$

Add Equations (8.27.1) and (8.27.2) to get

$$I_{11} + I_{12} = - \int_0^{\infty} (\theta_1''' \theta_0''' \theta_1' + \theta_1'''^2 \theta_0'') J dY \quad (8.27.3)$$

Further integration of the first term in bracket by parts results in

$$\begin{aligned} I_A &= - \int_0^{\infty} \theta_1''' \theta_0''' \theta_1' J dY \\ &= - \left[\theta_1'''^2 \theta_0''' J \right]_0^{\infty} + \int_0^{\infty} \theta_1''' (\theta_1''' \theta_0''' J + \theta_0''' \theta_1'' J + \theta_0'' \theta_1' \frac{\theta_0}{2} J) dY \\ &= 0 - I_A + \int_0^{\infty} \theta_1'''^2 (\theta_0''' + \frac{\theta_0''}{2} \theta_0') J dY \end{aligned}$$

Since the term in bracket is Blasius equation and is equal to zero, thus $I_A = 0$ and Equation (8.27.3) will be in the form:

$$I_{11} + I_{12} = - \int_0^\infty \theta_1'^2 \theta_0' J dY \quad (8.27.4)$$

I_{13} is given by

$$I_{13} = \frac{1}{2} \int_0^\infty (\theta_0' \theta_0'' \theta_1' \theta_1' J) dY,$$

integrating by parts to get

$$\begin{aligned} I_{13} &= \left| \frac{1}{2} \theta_1'^2 \theta_0' \theta_0'' J \right|_0^\infty - \frac{1}{2} \int_0^\infty \theta_1' (\theta_0' \theta_0'' \theta_1' J \\ &\quad + \theta_0'^2 \theta_1' J + \theta_0' \theta_0'' \theta_1' J + \theta_0' \theta_0'' \theta_1' \frac{\theta_0}{2} J) dY \\ &= - I_3 - \frac{1}{2} \int_0^\infty \theta_1'^2 (\theta_0'^2 + \theta_0' \theta_0'' + \theta_0' \theta_0'' \frac{\theta_0}{2}) J dY \end{aligned}$$

or

$$I_{13} = - \frac{1}{2} \int_0^\infty \theta_1'^2 \theta_0'^2 J dY \quad (8.27.5)$$

Adding Equations (8.27.4) and (8.27.5) we get;

$$I_1 = - \int_0^\infty (\theta_1'^2 \theta_0' + \frac{1}{2} \theta_1'^2 \theta_0'^2) J dY \quad (8.28)$$

The equation for I_{21} is given by

$$\begin{aligned} I_{21} &= \int_0^\infty (\theta_1' \theta_1' \theta_0' \theta_0'') J dY \\ &= 2 I_{13} \end{aligned} \quad (8.29.1)$$

and for I_{22} we have

$$I_{22} = - \int_0^\infty (\theta_0'^2 \theta_1'^2) J dY \quad (8.29.2)$$

Adding Equations (8.29.1) and (8.29.2) we get

$$I_2 = - \frac{1}{2} \int_0^\infty (\theta_1'^2 \theta_0'^2 + 2 \theta_0'^2 \theta_1'^2) J dY \quad (8.30)$$

Substituting from (8.28) and (8.30) into (8.25) gives;

$$\lambda = \frac{\int_0^\infty (\theta_1'^{-2} \theta_0' + \frac{1}{2} \theta_1'^2 \theta_0'^{-2}) J dY}{\int_0^\infty (\theta_0'^{-2} \theta_1'^2 + \frac{1}{2} \theta_1'^2 \theta_0'^{-2}) J dY} \quad (8.31)$$

Noting that θ_0' is always greater or equal to zero, thus λ will always be positive. This means that, when T goes to infinity the perturbation will go to infinity too, hence the linearization of the type defined in Equation (8.19) is not appropriate and the solution will be unstable at large times.

It has been reported by Stewartson¹⁰⁸ that the theoretical solution at large time is proportional to f given by

$$f(Y, T) = \theta_0 + Y T \theta_0' g(YT, \frac{1}{T}) \exp(-\alpha T^3) \quad (8.32)$$

where g is a Bessel function of order $1/3$ and α is a constant.

Noting the form of the Equation (8.32), we realize why we have not been able to obtain a solution in the foregoing analysis. Hence we conclude that the kind of linearization based on small perturbation for the unknown variable, has no meaning for this problem.

8.4.2 LINEARIZATION OF THE CONVECTIVE TERMS BY SERIES SOLUTION

In contrast to the type of linearization in section 8.4.1, the linearization of the convective terms always possesses a solution. This is because we are solving numerically the problem with linear equations instead of the nonlinear ones. Taking the problem of the suddenly accelerated plate, we will investigate the order of magnitude of the error involved by the linearization procedure. Following the work of Hall^{107,118}, the flow variables

are known at the start of the motion (Rayleigh solution). At a certain time step $(j+1)$, the convective terms are linearized with respect to the preceding time step (j) as follows

$$(u \frac{\partial u}{\partial x})_{j+1} = (u)_j (\frac{\partial u}{\partial x})_{j+1} + (u)_{j+1} (\frac{\partial u}{\partial x})_j - (u)_j (\frac{\partial u}{\partial x})_j \quad (8.33)$$

and

$$(v \frac{\partial u}{\partial y})_{j+1} = (v)_j (\frac{\partial u}{\partial y})_{j+1} + (v)_{j+1} (\frac{\partial u}{\partial y})_j - (v)_j (\frac{\partial u}{\partial y})_j \quad (8.34)$$

Hence the error involved in the linearization in equation (8.33) is

$$\begin{aligned} E &= (u)_{j+1} (\frac{\partial u}{\partial x})_{j+1} - (u)_j (\frac{\partial u}{\partial x})_{j+1} - (u)_{j+1} (\frac{\partial u}{\partial x})_j \\ &\quad + (u)_j (\frac{\partial u}{\partial x})_j \\ &= \left[(u)_{j+1} - (u)_j \right] \left[(\frac{\partial u}{\partial x})_{j+1} - (\frac{\partial u}{\partial x})_j \right] \end{aligned} \quad (8.35)$$

Now using the finite element representation we have

$$u_m = [X]_m \{u^e\}_m \quad (8.36)$$

where $[X]$ is the spacial shape function, u^e are the nodal values of u , and m is the element or subdomain delt with. The convective term becomes

$$\begin{aligned} (u \frac{\partial u}{\partial x})_m &= \{u^e\}_m^T [X]_m^T [X_x]_m \{u^e\}_m \\ &= \{u^e\}_m^T [K_x]_m \{u^e\}_m \end{aligned} \quad (8.37)$$

where

$$K_x = \frac{\partial X}{\partial x}$$

Thus Equation (8.35) becomes for the element m

$$E = (\{u^e\}_{m,j+1}^T - \{u^e\}_{m,j}^T) [K_x]_m (\{u^e\}_{m,j+1} - \{u^e\}_{m,j}) \quad (8.38)$$

Note that $[K_x] = [X^T][X]$ is not time dependent. Furthermore assume the difference between the nodal values at the two successive time steps as given by

$$\{\Delta u^e\}_{m,j+1,j} = \{u^e\}_{m,j+1} - \{u^e\}_{m,j} \quad (8.39)$$

then Equation (8.38) becomes

$$E = \{\Delta u^e\}_{m,j+1,j}^T [K_x]_m \{\Delta u^e\}_{m,j+1,j} \quad (8.40)$$

Following the Galerkin's procedure, this error is weighted and distributed over the element to get the variation in the residual as

$$(\Delta Re)_{m,j+1,j} = \iint_A [X]_m^T (\{\Delta u^e\}_{m,j+1,j}^T [K_x]_m \{\Delta u^e\}_{m,j+1,j}) dA_e \quad (8.41)$$

* The contributions from different elements at each node are added together to get

$$(\Delta Re)_{j+1,j} = \sum_{m=1}^M (\Delta Re)_{m,j+1,j} \quad (8.42)$$

where the summation is done for the elements that contribute at the nodal point of interest. Similar procedures can be followed for the convective term in Equation (8.34).

It is noted from Equation (8.41) that the error due to linearization is proportional to $(\Delta u^e)^2$. This means that if the

time step is very small, the expected error will be decreased. However the error will be accumulated after each time step, since the variable at the time step $(j+2)$ is dependent on the variable at the time step $(j+1)$ which itself depends on the variable at the time step (j) and so on. Table 35 gives an order of magnitude analysis for the relative value $(\Delta u)_{j+1,j}^2 / (u)_{j+1}^2$. Table (35.a) is for the time step $\Delta T = .125$ between $T_j = 1.875$ and $T_{j+1} = 2$, which is the same time step followed by Hall in this range. Nodal values are presented from the finite element analysis done in chapter 6. Since the nodal values at $T = 1.875$ are obtained by the finite element method, hence the error accumulated using linearization between $T = 1$ and $T = 1.875$ does not exist, and the error shown on the table is only due to the linearization between $T = 1.875$ and $T = 2$. In the last column the percentage difference between finite element and finite difference analysis^{107,118} is listed. Table (35.b) is similar but for a time step of $\Delta T = .25$ between $T_j = 2.75$ and $T_{j+1} = 3$. It is noted how the relative error $(\Delta u)_{j+1,j}^2 / (u)_{j+1}^2$ increases by increasing the time step, also the error presented here does not contain accumulation error from previous time steps. It is important to note that the error accumulation is not linearly added since the actual equations are nonlinear. A detailed discussion of the actual error between the finite element method and the finite difference method is presented in section 6.6, where it can be noted that the difference between the two methods is greater than the error predicted here (due to error accumulation).

This difference was found to be around .5%, which is not too large due to the fact that Hall has used an enormous number of equations (1280 equations) to solve the problem. In this research only 19 equations are required to obtain a similar accuracy.

8.5 RESULTS AND DISCUSSION

From the foregoing analysis we conclude that linearization procedures are not always successful for a given problem. The type of linearization based on small perturbation does not necessarily succeed in solving the problem, especially if the situation requires an abrupt change in the flow field, and mainly this is the case for transient flows. The test case for the suddenly accelerated plate is one of these problems to which the linearization procedure noted above will fail to reach a solution. The other alternative of linearization is to linearize the convective terms in the partial differential equation, but at large Reynolds number, these convective terms have the dominant effect on the flow behaviour and the author believes it is best to solve the nonlinear equation itself without referring to linearization. To decrease the error involved in the linearization of the convective terms, the mesh size has to be reduced. It was shown in the solution of the steady boundary layer equation on a flat plate, that with 6 element in the vertical direction, and using Galerkin's finite element procedure to solve the nonlinear equation, we have reached better accuracy than by using 20 elements and linearizing the equation^{103,127}. By doing that, 17 equations in the present research using the finite element

Galerkin model are required to obtain better accuracy than the other finite element Least Squares method^{103,127} which solved 122 linear equations. It was also shown in the solution for the suddenly accelerated flat plate, that the theoretical decay of the velocity at large time mentioned by Stewartson¹⁰⁸ was found to exist using the Finite element Galerkin's procedure, and that 19 equations for the flow variables are needed to obtain a similar accuracy to the finite difference method which used 1280 equations.^{107,118}

CHAPTER 9

CONCLUSIONS AND RECOMMENDATION

The objective of this research was to develop a Finite Element Model to obtain numerical solutions of two-dimensional, viscous, incompressible, steady and unsteady, nonlinear flow problems. Such solutions were obtained using a Galerkin's approach to the equations of motion without referring to any linearization procedures. The method uses third order and quintic elements for the steady state flow case. For the unsteady flow, the method uses the quadratic element to discretize the flow based on the primitive variables u_j and p , and then the quintic high precision element to discretize the flow based on the stream function formulation.

The Finite Element Galerkin Model (F.E.G.M.) has been tested thoroughly for both steady and unsteady boundary layer flows. In the steady case, the flow has been analyzed for a flat plate (Blasius problem), together with suction and blowing, and also for wedge and stagnation problems. In the unsteady case, the important problem of a suddenly accelerated flat plate has been solved. A detailed examination of the errors involved in linearization procedures for steady and unsteady flows has been investigated. As a background to this work the author has also provided a comprehensive review on Finite Difference and Finite Element methods in fluid flow problems, a detailed examination of numerical methods used to solve the nonlinear algebraic equations, after the

discretization procedure, has been carried out, and discussion of the methods necessary for the time marching scheme in unsteady flow problems.

During the investigation, the following important, and novel, techniques have been evolved:

(1) A new method by which the residuals are calculated at each node without referring to a global matrix. Thus, instead of performing the summation of many elemental matrices to form the so called global matrix, the contributions of the elements that share a specified nodal point are added when the nonlinear equation at this node is required to be solved. By doing this, a column vector replaces the two-dimensional global matrix, and hence the computer core required will be reduced considerably.

(2) A new technique by which the errors involved in the nodal variables for a specified grid, and the associated residuals can be related. This new technique allows the choice of the best grid required to get the maximum accuracy and efficiency together with the minimum number of nodal variables. The technique has been found to be extremely accurate and efficient for time consuming unsteady flow problems.

(3) A method by which one can estimate the errors involved in different linearization procedures, has been presented both in theoretical and numerical forms. The method illustrates the different types of instabilities and errors obtained when the equations are

linearized, and shows how far the linearized solution will deviate from the actual nonlinear solution. It indicates that certain linearization procedures cannot possibly work.

(4) A new approach has been derived to take profit from the similarity behaviour in boundary layer flows. This approach can decrease the number of the unknown nodal variables involved, and thus, the computer time considerably. The novel concept in this approach is to choose an accurate orientation for the elements in the grid to profit from similarity relations.

(5) A method has been devised by which low order elements can still be used to discretize the equations of motion containing high order derivatives. It has been noted that when the compatibility of high order derivatives is not fulfilled by the element chosen; the results will show an oscillating behaviour around the expected true solution. By using the foregoing new method, the situation has been avoided.

(6) To the author's knowledge, the nonlinear unsteady viscous flow equations have been solved for the first time without referring to different kinds of linearization procedures or restrictions on the solution. The developed scheme is adequate to solve both linear and nonlinear equations with an extremely good accuracy.

The significant advantages of these developments are considered to be as follows;

(1) By making use of the developed F.E.G.M., high accuracy and efficiency are obtained without referring to any linearization

procedure.

(2) The method by which the global matrix is avoided has led to considerable savings in computer storage, and corresponding savings in computer time. The computer storage is decreased by a ratio of $(100 - \frac{100}{\sqrt{n_s}})$, where n_s is the size of the global matrix.

(3) By making use of the method by which the error in the nodal values can be estimated from the residuals, clear guidance is obtained on grid choice and its adequacy for the specified problem. This leads to an efficient, accurate solution, with a minimum number of nodal variables, and allows testing of the grid before starting the solution procedure.

(4) The method by which the similarities existing in boundary layer flows have been taken into consideration, has decreased the number of unknown nodal variables to a third of the original number, and reducing the corresponding computer time to one ninth of the actual time.

The accuracy and efficiency of the F.E.G.M. developed have been critically examined by comparison with other Finite Difference and Finite Element methods. The results obtained demonstrate the reliability, superiority and potential of the F.E.G.M. Some of the important results are listed below:

(1) Very few elements are required to obtain a high rate of convergence and accuracy for steady and unsteady problems. Reviewing as follows;

(a) For the steady state Blasius problem, a grid composed of 6 elements in the F.E.G.M. gives better accuracy than other finite element methods that use a grid of 20 elements of the same order and apply a linearization procedure.

(b) For the unsteady flow over the suddenly accelerated flat plate problem, 19 equations are required to obtain better accuracy when compared with the existing finite difference method which uses 1280 equations at each time step. The F.E.G.M. has been able to define, for the first time, the exponential decay at large time specified by Stewartson. In addition, the present analysis demonstrated no instabilities despite of the large time steps used, which may be a further advantage over finite difference methods.

(2) Linearization procedures are not always successful and sometimes impossible for a given problem. The error involved in the linearization procedures has been found to be too large to describe the non-linear problem with an acceptable accuracy. Linearization is not recommended for boundary layer transient flow due to the accumulation of errors at each time step, and thus a misleading steady state solution.

(3) The conversion from the primitive variables u_j - p solution to stream function ψ formulation will lead to the solution of one equation instead of two, but with high order derivatives. This will necessitate the use of a high order element, such as the quintic element. Results show that the two different kinds of formulation

will lead to a comparable accuracy, with the computer time slightly higher in the ψ formulation. The only disadvantage when using the high precision element is the tedious algebraic manipulations involved which increase the computer time considerably. It possesses the advantage of having nearly all the flow variables (e.g. stream function, velocities, vorticity, etc..) defined explicitly as nodal variables, thus obtaining a full compatibility of the flow variables at each node, a potential of the high precision element which may be of value in more general fluid flow problems than those treated in this work.

It is the author's opinion, that the foregoing F.E.G.M. can easily be extended to include more sophisticated flow problems, other than boundary layer equations; for example involving compressibility effects. The first logical extension of the model is to incorporate the necessary modifications for the solution of a suddenly accelerated airfoil in a viscous flow field and using the Navier-Stokes equations. This will involve a time step marching scheme to reach the steady state solution. Further work would study the flow over oscillating airfoils, and compare with the recent work of Bratanow⁷¹⁻⁷⁴ who used linearization procedures. From the experience gained in this research, and from the work of Bratanow (who has used 508 elements, third order for stream function, and linear for vorticity, thus 4 degrees of freedom per node), the number of elements required using the F.E.G.M. are expected to be 240 elements of the high precision type, thus the stream function,

velocities and vorticity will be defined explicitly at each node. By using this element one will solve one equation of a fourth order type. Another alternative is to use the vorticity transport equation (second order) and the Poisson's equation for stream function, and with the same elements used by Bratanow. In this last alternative the velocities will be obtained from the stream function nodal variables ($u = \psi_y$ and $v = -\psi_x$) and thus they are quadratic polynomials, which is consistent with the assumption of linear vorticity ($\omega = u_y - v_x$); this is not the case in Bratanow's work, who assumed linear velocities obtained from the perturbation of the vorticity. By using the foregoing vorticity - stream function formulation, the two equations are solved simultaneously at each time step and the time marching scheme is applied on the vorticity equation alone. Since the experience gained in this research is with the high precision element, the computer time necessary to solve the equations using such element is expected to be of the same order of magnitude of the time mentioned by Bratanow (15.5 minutes on a Univac 1108 computer with 65K core storage), and using the same computer (Univac 1108). For the suddenly accelerated airfoil problem, if the airfoil is symmetrical around the uniform flow coming from infinity, only half of the grid is required for the solution. Since the axes of the grid are fixed with the airfoil, thus moving with the airfoil by the same speed; it will be preferable to use a triangular element with one edge curved and lying on the airfoil section, this will be more accurate in satisfying the

boundary conditions on the airfoil surface, and will reduce the number of elements in the chordwise direction of the airfoil.

The F.E.G.M. could also be applied to heat transfer problems. Indeed, the suddenly heated plate will have similar solution to the suddenly accelerated plate. Similarly to solve the wave equation, a double integration with respect to time can be performed on the discretized equations using a modified version of the time marching scheme described previously.

REFERENCES

1. Cheng, S.I. and Rimón, Y., "Numerical Solution of a Uniform Flow over a Sphere at Intermediate Reynolds Numbers, "The Physics of Fluids, Vol.12, No. 5, May 1969, p. 949.
2. Harlow, F.H. and Amsden, A.A., "Fluid Dynamics: An Introductory Text," Rept, LA-4100, Feb. 1970, Los Alamos Scientific Lab., Los Alamos N. Mex.
3. Roache, P. J. and Mueller, T.J. "Numerical Solutions of Laminar Separated Flows, "AIAA Journal, Vol. 8, No. 3, March 1970, p. 530.
4. Thoman, D.C. and Szewczyk, A.A., "Time-Dependent Viscous Flow Over a Circular Cylinder," The Physics of Fluids Supplement II, Vol. 12, No. 12, Dec. 1969, p.76.
5. Fromm, J.E., "A Method for Computing Nonsteady, Incompressible, Viscous Fluid Flows," Los Alamos Scientific Lab. Report LA-2910, Sept. 1963.
6. Son, J.S. and Hanratty, T.J., "Numerical Solution for the Flow Around a Cylinder at Reynolds Numbers of 40, 200, and 500," J. Fluid Mech., Vol. 35, Part 2, Feb. 1969, pp. 369-386.
7. Sykes, D.M., "The Supersonic and Low-Speed Flows Past Circular Cylinders of Finite Length Supported at One End", J. Fluid Mech., Vol. 12, Mar. 1962, pp. 367-387.
8. Takami, H. and Keller, H.B., "Steady Two-Dimensional Viscous Flow of an Incompressible Fluid Past a Circular Cylinder," Phys. of Fluids Suppl. II, Vol. 12, 1969, pp. 51-56.
9. Achenbach, E., "Distribution of Local Pressure and Skin Friction Around a Circular Cylinder in Cross-Flow Up to $Re = 5 \times 10^6$," J. Fluid Mech., Vol. 34, Part 4, Dec. 1968
10. Hess, J.L., "Calculation of Potential flow about Arbitrary Three-Dimensional Lifting Bodies", Report No. MDC J5679-01, Douglas Aircraft Co., 1972.
11. Morino, L., "Unsteady Compressible Potential Flow Around Lifting bodies: General Theory", AIAA paper No. 73-196.
12. Morino, L. and Kuo C.C., "Unsteady Subsonic Compressible Flow Around Finite Thickness Wings," AIAA paper No. 73-313.

13. Giesieng, J.P., Kalman, T.P., and Rodden, W.P., "Subsonic Steady and Oscillatory Aerodynamics for Multiple Interfering Wings and Bodies," *Journal of Aircraft*, Vol. 9, No. 10, Oct. 1972, pp. 693-702.
14. Thompson, J.F. Jr., Shanks, S.P., and Wu, J.C., "Numerical Solution of the Three-Dimensional Navier-Stokes Equations in Integro-Differential Form: Flow about a Finite Body", *Proceedings of the AIAA Computational Fluid Dynamics Conf.*, Palm Springs, Calif., July 19-20, 1973.
15. Davis, G. de Vahl, "Laminar Natural Convection in a Rectangular Cavity", *New York University Report No. F-67-2*, N.Y. 1967.
16. Davis, G. de Vahl, "Laminar Natural Convection in an Enclosed Rectangular Cavity", *Int. J. Heat & Mass Transfer*, Vol. 11, 1968.
17. Zienkiewicz, O.C. and Cheung, Y.K., "The Finite Element Method in Structural and Continuum Mechanics", 2nd ed., McGraw-Hill, New York, 1967.
18. Norrie, D.H. and de Vries, G., "The Finite Element Method", Academic Press New York and London, 1973.
19. Norrie, D.H. and de Vries, G., "A Finite Element Bibliography", second edition, May 1975, Mechanical Engineering Dept. University of Calgary, Rept. No. 57, 58, 59.
20. Whiteman, J.R., "A Bibliography for Finite Elements", Academic Press, London, New York, San Francisco, 1975.
21. Jordon, W.B., "The Plane Isoparametric Structural Element", General Electric Company, KAPL-M-7112, Schenectady, N.Y., Feb. 1970.
22. Zienkiewicz, O.C., "Isoparametric Element Forms in F.E. Analysis", NATO advanced study institute on F.E. methods in continuum mechanics, Lisbon, 1971.
23. Thompson, D.S., "Flow Through a Cascade of Aerofoils", F.E.M. in Flow problems, *Int. Symp.* Jan. 1974. Swansea, United Kingdom.
24. de Vries, G., and Norrie, D.H., "The Application of the Finite Element Technique to Potential Flow Problems", *Trans. ASME appl. Mech. Div.*, Paper No. 71-APM-22, 798-802 (1971).
25. Argyris, J.H., Mareczek, G. and Scharpf, D.W., "Two and Three Dimensional Flow Using Finite Elements", *Aero. J. Roy. Aero. Soc.* Vol. 73, pp. 961-964, November 1969.

26. de Vries, G., and Norrie, D.H., "The Application of the Finite Element Technique to Potential Flow Problems: Part I, "Mechanical Eng. Rep. No. 7, Univ. of Calgary, July 1969.
27. de Vries, G., and Norrie, D.H., "The Application of the Finite Element Technique to Potential Flow Problems: Part II", Mechanical Eng. Rep. No. 8, Univ. of Calgary, Aug. 1969.
28. de Vries, G., and Norrie, D.H., "The Application of the Finite Element Technique to Potential Flow Problems: Part III, Unsteady Flows," Mechanical Eng. Rep. No. 9, University of Calgary.
29. Shen, S.F., "An Aerodynamicist Looks at the F.E.M., "F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.
30. Tong, P. "The Finite Element Method for Fluid Flow," Paper US 5-4, Japan-U.S. Seminar on Matrix Methods of Structural Analysis and Design, Tokyo, Japan, August, 1969.
31. Atkinson, B., Brockleband, M.P. Card, C.C.H. and Smith, J.M. "Low Reynolds Number Developing Flow", A.I.Ch.E.J., Vol. 15, pp. 548-553, July, 1969.
32. Reddi, M.M., "Finite Element Solution of the Incompressible Lubrication Problem", J. Lub. Tech., Trans. A.S.M.E., Vol. 91, Ser. F. No. 3, pp. 524-533, July 1969.
33. Argyris, J.H. and Scharpf, D., "The Incompressible Lubrication Problem", Aero. J. Roy. Aero. Soc., Vol. 73, pp. 1044-1046, Dec, 1969.
34. Oden, J.T. and Sornoggi, D., "Finite Element Applications in Fluid Dynamics," J. Eng. Mech. Div., Proc. A.S.C.E., Vol. 95, No. EM3, pp. 821-826, June 1969.
35. Leonard, J.W., "Finite Element Analysis of Perturbed Compressible Flow", Int. J. Num. Meth. Eng., Vol. 4, 1972, pp. 123-132.
36. Olson, M.D., "A Variational Finite Element Method for Two-Dimensional Steady Viscous Flow", McGill Univ. Eng. Inst. of Canada, Conf. on Finite Element Methods in Civil Eng., McGill Univ., Montreal, Quebec, 1972.
37. Olson, M.D., "Variational Finite Element Methods for Two-Dimensional and Axisymmetric Navier-Stokes Equations", F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.

38. Cowper, G.R., Kosko, E., Lindberg, B.M., and Olson, M.D., "Static and Dynamic Applications of a High Precision Triangular Plate Bending Element", AIAA, Journal 7, 1957-1965, (1969).
39. Skiba, E., Unny, T.E. and Weaver, D.S., "A Finite Element Solution for a Class of Two-Dimensional Viscous Fluid Dynamics Problems", Computer Aided Engineering, edited by G.M.L. Gladwell, Univ. of Waterloo, Waterloo, Ontario, Canada, 1970, p. 493.
40. Bogner, F.K., Fox, R.L. and Schmit, L.A., "The Generation of Inter-element-Compatible Stiffness and Mass Matrices by the use of Interpolation Formulae", Proc. Conf. on Matrix Methods in Structural Mechanics, Wright-Patterson Air Force Base, Ohio, U.S.A., 1965.
41. Mason, V., "Rectangular Finite Elements for Analysis of Plate Vibrations", J. Sound Vib. 7, pt. 3, pp. 437-448, 1968.
42. Argyris, J.H. : Recent Advances in Matrix Method of Structural Analysis, (Pergamon Press) 1964.
43. Argyris, J.H., "The Impact of the Digital Computer on Engineering Sciences, Part I, "Aeronautical Journal of the Royal Aeronautical Society, Vol. 74, January 1970, pp. 13-41.
44. Argyris, J.H. and Fried, I., "The LUMINA Element for the Matrix Displacement Method, " AJRAES, Vol. 72, No. 690, June 1968.
45. Argyris, J.H., Fried, I. and Scharpf, D.W., "The TET 20 and TEA Elements for the Matrix Displacement Method," AJRAES, Vol. 72, No. 691, July 1968.
46. Argyris, J.H., Fried, I. and Scharpf, D.W., "The HERMES 8 Element for the Matrix Displacement Method," AJRAES, Vol. 72, No. 691, July 1968.
47. Argyris, J.H. and Scharpf, D.W., "A Sequel to TN13: The Curved Tetrahedral and Triangular Elements TEC and TRIC for the Matrix Displacement Method, Part I: Small displacements", AJRAES, Vol. 73, No. 697, January 1969.
48. Argyris, J.H. and Scharpf, K.W., "A Sequel to TN13: The Curved tetrahedral and Triangular Elements TEC and TRIC for the Matrix Displacement Method, Part II: Large Displacements," AJRAES, Vol. 73, No. 697, January 1969.
49. Zienkiewicz, O.C. and Parekh, C.J., "Transient Field Problems: Two-Dimensional and Three-Dimensional Analysis by Isoparametric Finite Elements", Int. Journal for Num. Methods in Eng. Vol. 2, 61-71 (1970).

50. Taylor, R.L., "On Completeness of Shape Functions for Finite Element Analysis", Int. J. Num. Methods in Engin., Vol. 4, pp. 17-22, 1972.
51. Oden, J.T., "A General Theory of Finite Elements. II. Applications", Int. J. of Numerical Methods in Engineering, Vol. 1, 247-259 (1969).
52. Zienkiewicz, O.C., and Cheung, Y.K. "Finite Elements in the Solution of Field Problems," The Engineer, 220, 507-510 (1965).
53. Martin, H.C. "Finite Element Analysis of Fluid Flows", Proc. 2nd Conf. Matrix Methods in Structural Mechanics, Wright-Patterson Air Force Base, Ohio, 1968, AFFDL-TR-68-150.
54. Doctors, L.J. "An Application of the Finite Element Technique to Boundary Value Problems of Potential Flow", Int. J. of Numerical Methods in Engineering, Vol. 2, pp. 243-252, 1970.
55. Kyuichiro Washizu, "Finite Element Technique in Lifting Surface Problems", F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.
56. Bowley, W.W., and Prince, J.F. "Finite Element Analysis of General Fluid Flow Problems", AIAA Paper No. 71-602 AIAA 4th Fluid and Plasma Dynamics Conference, June 21-23, 1971.
57. Carey, G.F., "Perturbation-Variational Solution of Compressible Flows", F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.
58. Lee-Tzong Chen, Emil O. Suciu and Luigi Morino, "A Finite Element Method for Potential Aerodynamics Around Complex Configurations", AIAA paper, No. 74-107, 1974.
59. Tadros, R.N., "Second Order Unsteady Incompressible Aerodynamic Loads and Application on Flutter Speed", M.Sc. Thesis, Cairo University, 1973.
60. Finlayson, B.A., "Existence of Variational Principles for the Navier-Stokes Equations", The Physics of Fluids, Vol. 15, No. 6, 1972, pp. 963-967.
61. Tonti, E. "Variational Formulation of Nonlinear Differential Equations I and II, "Bull. Acad. Roy. Belg. (Class. Sci.), Vol. 5, No. 55, 1969, pp. 137-167, 262-278.
62. Hood, P. and Taylor, C., "Navier-Stokes Equations using Mixed Interpolation", F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.

63. Oden, J.T. and Wellford, L.C. Jr., "Analysis of Flow of Viscous Fluids by the Finite-Element Method," AIAA J. Vol. 10 No. 12, Nov. 1972.
64. Glansdorff, P. and Prigogine, I., "On a General Evolution Criterion in Macroscopic Physics," Physica, 30, 351-374, 1964.
65. Zienkiewicz, O.C., "The Finite Element Method in Engineering Science", (McGraw-Hill, New York, 1971), Chaps. 3, 15 and 16.
66. Morino, L. "A Finite-Element Formulation for Subsonic Flows Around Complex Configurations", NASA-CR-138142, Dec. 1973.
67. Mariappa and Smith, G.C.C., "Finite Element Approach to the Integrated Potential Formulation of General Unsteady Supersonic Aerodynamics", NASA-CR-112296, 1973.
68. Jones, W.P., "Supersonic Theory for Oscillating Wings of any Planform", A.R.C. R & M 2655, June 1948.
69. Woodcock, D.L. and Lawrence, A.J., "Flutter Research in the United Kingdom," 1969-1971 R.A.E. Tech. Memo. Structures 813, Sept. 1971.
70. Healet, M.A., Lomax, H. and Jones, A.L., "Volterra's Solution of the Wave Equation as Applied to Three Dimensional Supersonic Airfoil Problems", NACA Report 889, 1947.
71. Bratanow, T. Ecer, A., and Kobiske, M. "Finite Element Analysis of Unsteady Incompressible Flow Around an Oscillating Obstacle of Arbitrary Shape," AIAA 11th Annual Meeting, Washington, D.C. Jan. 1973.
72. Bratanow, T. and Ecer, A., "Computational Considerations in Application of the Finite Element Method for Analysis of Unsteady Flow Around Airfoils," presented at AIAA Comput. Fluid Dyn. Conf. Palm Springs, Calif., July 1973.
73. Bratanow, T. and Ecer, A., "Finite Element Analysis and Computer Graphics Visualization of Flow Around Pitching and Plunging Airfoils", NASA, CR-2249, Univ. of Wisconsin, Milwaukee, Sept., 1973.
74. Bratanow, T. and Ecer, A., "Analysis of Moving Body Problems in Aerodynamics", F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.
75. Oden, J.T., "Finite Element Analogue of Navier-Stokes Equations", J. Eng. Mech. Div., Proc. A.S.C.E., Vol. 96, No. EM4, pp.529-534, August 1970.

76. Silvester, P. "High Order Polynomial Triangular Finite Elements for Potential Problems," Int. J. Eng. Sci. Vol. 7, pp. 849-861.
77. Grandall, S., "Engineering Analysis," McGraw-Hill, New York, 1956.
78. Szabo, P.A. and Lee, G.C., "Derivation of the Stiffness Matrix for Plates by Galerkin Method", Int. J. Num. Meth. in Engng. 1, 1969.
79. Irons, B.M., "Engineering Applications of Numerical Integration in the Stiffness Method", AIAA J. 4, 2035-2037 (1966).
80. Ergatoudis, J., Irons, B.M. and Zienkiewicz, O.C., "Curved, Isoparametric, Quadrilateral Elements for Finite Element Analysis", Int. J. Solids Structures, 4, 31-42 (1968).
81. Ergatoudis, J., Irons, B.M. and Zienkiewicz, O.C., "Three Dimensional Analysis of Dams and Their Foundations", Symp. of Arch. Dams, Inst. Civ. Eng., London, 37-50, 1968.
82. King, I.P. Norton, W.R. and Iceman, K.R., "A F.E. Model for Two-Dimensional Flow", F.E.M. in Flow Problems, Int. Symp. on F.E.M. in Flow Problems, Jan. 1974, Swansea, United Kingdom.
83. Cheng, R.T., "Numerical Solution of the Navier-Stokes Equations by the Finite Element Method", The physics of Fluids, Vol. 15, No. 12, pp. 2098-2105, December 1972.
84. Baker, A.J. "Finite Element Theory for Viscous Fluid Dynamics", Bell Aerospace Co. Report No. 9500-920189, August 1970.
85. Finlayson, B.A., and Scriven, L.E., "On the Search for Variational Principles", Int. J. Heat Mass Transfer, Vol. 10, pp. 799-821, 1967.
86. Finlayson, B.A., "The Method of Weighted Residuals and Variational Principles", Academic Press, New York, 1972.
87. Akin, J.E., and Sen Gupta, S.R., "Finite Element Application of Least Squares Techniques", in Variational Methods in Engineering, Vol. I, Proceeding of an International Conference, University of Southampton, 25th September, pp. 5/82, 1972.
88. Bazeley, G.P., Cheung, Y.K., Irons, B.M. and Zienkiewicz, O.C., "Triangular Elements in Plate Bending-Conforming and Non-Conforming Solutions", Proceedings of Conference on Matrix Methods and Structural Mechanics, AFFDL-TR-66-80, Oct. 26-28, 1965, Wright-Patterson Air Force Base, Ohio, pp. 547-576.

89. Duncan, W.J., "Galerkin's Method in Mechanics and Differential Equations", Gt. Brit. Aero. Res. Council Rept. and Memo. (1937), reprinted in Gt. Brit. Air Ministry Aero. Res. Comm. Tech. Rept. 1, pp. 484-516, 1937.
90. Schlichting, H., "Boundary Layer Theory", 6th ed., McGraw-Hill, (1968).
91. Krasnosel'skii, M.A., Vainikko, G.M., Zabreiko, P.P., Rutitskii, Ya. B., and Stetsenko, V. Ya, "Approximate Solution of Operator Equations", Tr. by D. Louvish, Wolters-Noordhoff, 1972.
92. Ortega, J.M., and Rheinboldt, W.C., "Iterative Solution of Non-linear Equations in Several Variables, Academic Press., 1970.
93. Ostrowski, A.M. "Solution of Equations and Systems of Equations", Academic Press, 1966.
94. Rall, L.B., "Computational Solution of Nonlinear Operator Equations", Willey, 1969.
95. Rabinowitz, P., (ed.), "Numerical Methods for Nonlinear Algebraic Equations", Gordon and Breach, 1970.
96. Brown, K.M., "A Quadratically Convergent Newton-like Method Based Upon Gaussian Elimination", SIAM Journal on Numerical Analysis, 6(4), pp. 560-569, 1969.
97. Peaceman, D.W., and Rachford, H.H., Jr., "The Numerical Solution of Parabolic and Elliptic Differential Equations," Journal SIAM, Vol. 3, No. 1, March, 1955.
98. Ralston, A., and Wilf, H.S., "Mathematical Methods for Digital Computers", John Wiley & Sons, Inc., 1966.
99. Rosenberg, D.U.V., "Methods for the Numerical Solution of Partial Differential Equations, " Modern Analytic and Computational Methods in Science and Mathematics, 16, 1975
100. Gear, C.W. "Numerical Initial Value Problems in Ordinary Differential Equations," Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.
101. Thompson, J.F., "Computer Experimentations with an Implicit Numerical Solution of the Navier-Stokes Equations for an Oscillatory Body," AIAA Paper No. 69-105, AIAA Seventh Aerospace Sciences Meeting, New York, Jan. 1969.
102. Isaacson, E.K., and Keller, H.R., "Analysis of Numerical Methods," John Wiley and Sons, Inc.; N.Y., 1966.

103. Lynn, P.P., "Least Squares Finite Element Analysis of Laminar Boundary Layer Flows," Int. J. Num. Methods in Eng., Vol. 8, p.p. 865-876, 1974.
104. Emmons, H.W., and Leigh, D.C., "Tabulation of the Blasius Function with Blowing and Suction," A.R.C., C.P. 157, 1954.
105. Hartree, D.R., "On an Equation Occurring in Falkner and Skan's Approximate Treatment of the Equations of the Boundary Layer", Proc. Cambr. Phil. Soc. 33, Part II, pp. 223-239, 1937.
106. Rosenhead, L., (ed.), "Laminar Boundary Layers, " Clarendon Press, Oxford, 1963.
107. Hall, M.G., "The Boundary Layer over an Impulsively Started Flat Plate", Proc. Roy. Soc. A. 310, pp. 401-414, 1969.
108. Stewartson, K., "On the Impulsive Motion of a Flat Plate in a Viscous Fluid", Quart. Appl. Math. and Mech. 4, pp. 182-198, 1951.
109. Stuart, J.T., "Unsteady Boundary Layer", in Recent Research on Unsteady Boundary Layers, E.A. Eichelbrenner (ed.), Quebec, 1, pp. 886-980, 1971.
110. Oudart, A., "Mise en Prégome de la Couche Limite de la Plaque Plane dans l'Impulsion Brusque à partir du Repos", Rech. Aero. 31, 7-12, 1953.
111. Cheng, S. - I., and Elliott, D., "The Unsteady Laminar Boundary Layer on a Flat Plate", Trans. Am. Soc. Mech. Eng. 79, pp. 725-733, 1957.
112. Cheng, S. - I., "Some Aspects of Unsteady Laminar Boundary Layer Flow", Quart. Appl. Math. 14, pp. 337-352, 1957.
113. Lam, S.H., and Crocco, L., "Note on the Shock-Induced Unsteady Laminar Boundary Layer on a Semi-Infinite Flat Plate", J. Aerospace Sci. 26, pp. 54-55, 1959.
114. Smith, S.H., "The Impulsive Motion of a Wedge in a Viscous Fluid", Z. Angew. Math. Phys. 18, pp. 508-522, 1967.
115. Farn, C.L.S., and Arpaci, "On the Numerical Solution of Unsteady Laminar Boundary Layers", AIAA J. 4, pp. 730-732, 1966.

116. Dwyer, H.A., "Calculation of Unsteady Leading-Edge Boundary Layers," AIAA J. 6, pp. 2447-2448, 1968.
117. Yoshizawa, A., "Unsteady Viscous Flows Past a Semi-Infinite Flat Plate," J. Phys. Soc. Japan 27, pp. 472-477, 1969.
118. Hall, M.G., "A Numerical Method for Calculating Unsteady Two-Dimensional Laminar Boundary Layer", Ing. Arch. 38, pp. 97-106, 1969.
119. Koob, S.J., and Abbott, D.E., "Investigation of a Method for the General Analysis of Time Dependent Two-Dimensional Laminar Boundary Layers", J. of Basic Eng. 90 D, pp. 4-11, 1968.
120. Koob, S.J., and Abbott, D.E., "The Case for Extended Boundary-Layer Equations", in Fluid Dynamics of Unsteady, Three-Dimensional and Separated Flows, F.J. Marshall (ed.), pp. 458-463, 1971.
121. Tani, I. and Yu, N.J., "Unsteady Boundary Layers over a Flat Plate Started from Rest," in Recent Research on Unsteady Boundary Layers, E.A. Eichelbrenner (ed.) Quebec, 1, pp. 886-980, 1971.
122. Dennis, S.C.R., "The Motion of a Viscous Fluid Past an Impulsively Started Semi-Infinite Flat Plate," J. of the Inst. of Math. and its App. 10, pp. 105-117, 1972.
123. Reddy, K.C., Sickles, W.L., and Yalamanchili, Rao, "Computation of Unsteady Boundary Layers", in Unsteady Aerodynamics, R.B. Kinney (ed.), Univ. of Arizona, pp. 221-244, 1975.
124. Cowper, G.R. et al., "A High Precision Triangular Plate Bending Element", Aeronautical Rept. LR-514, National Research Council of Canada, 1968.
125. Cowper, G.R., "The High Precision Triangular Finite Element for Plate Bending: An Alternative Formulation", LM, National Aeronautical Establishment, ST-152, Canada, 1972.
126. Rott, N., and Rosenzweig, M.L., "On the Response of the Laminar Boundary Layer to Small Fluctuations of the Free Stream Velocity", J. Aeronautical Sciences, pp. 741-747, 1960.
127. Lynn, P.P., and Alani, K., "Efficient Least Squares Finite Elements for Two-Dimensional Laminar Boundary Layer Analysis", Int. J. Num. Meth. Eng., Vol. 10, pp. 809-825, 1976.

128. Tuann, S.-Y., and Olson, M.D., "A Study of Various Finite Element Solution Methods for the Navier-Stokes Equations", The Univ. of British Columbia, Structural Research Series, Report No. 14, May, 1976.
129. Taylor, C., and Hood, P., "A Numerical Solution of the Navier-Stokes Equations Using the Finite Element Technique", Int. J. Computers and Fluids, Vol.1, pp. 73-100, 1973.
130. Silvester, P., "Newton-Cotes Quadrature Formulae for N-Dimensional Simplexes," Proc. 2nd Can. Congr. Appl. Mech., Waterloo, Canada, 1969.
131. Irons, B.M. "Engineering Applications of Numerical Integration in Stiffness Methods", AIAA J. 4, pp. 2035-2037, 1966.
132. Hammer, P.C., Marlowe, U.J., and Stroud, A.H., "Numerical Integration over Simplexes and Cones", Math. Tabl. Natn. Res. Coun. Wash., 10, pp. 130-137, 1956.
133. Tadros, Reda N., and Kirkhope, J., "Galerkin-Finite Element Formulation in Viscous Flow", Int. J. Computers and Fluids (in Press).
134. Tadros, Reda N., and Kirkhope, J., "Unsteady Boundary Layer over an Impulsively Started Flat Plate Using Finite Elements", Int. J. Computers and Fluids (in Press).

Fig.1. Division of a Domain V into Finite Elements or Subdomains V^e .

Plane Elements

TRIM 6

Three Dimensional Elements

TET 10

TRIC 6

TEC 10

Fig.2. Elements Based on Linear Strain Assumption.

TRIAC 4 **HERMES 4**

- Nodal Point with Prescribed Potential and Velocities.
- △ Nodal Point with Prescribed Potential.
- x Nodal Point Defining Geometry.

Fig.3. Elements Based on Potential-Velocity Formulation.

Fig.4. Finite Elements Idealization of Cylinder Flow Problem.

Fig.5. Velocity Distribution for Different Elements after the Fourth Iterate.

Fig.6. Initial Iterate f_0 with the First two Iterates f_1 and f_2 for the Fifth Order Element versus Blasius.

Fig.7. Flow Past a Wedge.

Fig. 8. Finite Element-Velocity Distribution in the Boundary Layer of a Flat Plate with Suction and Discharge.

Fig. 9. Finite Element-Velocity Distribution in the Boundary Layer of Wedges and Stagnation Flows.

Fig. 10. Finite Element Meshes.

Fig. 11. Profiles of the Velocity Component u (20 Elements)

Fig. 12. Surface Shear (20 Elements)

Fig. 13. Decay of the Velocity Component u (20 Elements).

Fig. 14. Nodal Order of the Quadratic Element.

Fig. 15. Area Coordinates.

Fig. 16. Modified Quadratic Element Solution.

Fig.18. Convergence of Different Elements'Grids to the Accurate Solution.

Fig. 19. Time Marching Results for the Stream Function ψ

Fig. 20. Time Marching Results for ψ_x

Fig. 21. Time Marching Results for ψ_{xx}

Fig. 22: Time Marching Results for Ψ_{xy}

Fig.23. Time Marching Results for Ψ_{yy}

Fig.24. Linearized versus Exact Solution for the Steady Flow over a Flat Plate.

Fig. 25. Flow Chart for the Steady Flow.

Fig. 26. Flow Chart for the Unsteady Flow-Primitive Variables Formulation.

Fig. 27. Flow Chart for the Unsteady Flow-Stream Function Formulation.

TABLE 1 - STEADY FLOW - F.E.M. APPLICATIONS.

Type of the Steady Flow	Ref. No.	Method Used to Find Variables & Problem Involved	Type of Element (if any)	Comparison with Other Results (if any)	Comments
-1- INCOMPRESSIBLE NON-VISCOUS	53	Variational principle (flow around a cylinder between parallel walls)	Linear triangular	Results are compared with the singularity method. The F.E.M. results are undoubtedly closer, to the true results, than the singularity method. Similar results are obtained by Doctors ⁵⁴ who assessed the accuracy of the F.E.M.	The same idealization of the flow has been previously used for analyzing the corresponding flat sheet with a circular cutout (Fig. 4). Therefore, considerable input data was already available for undertaking the fluid problem.
	55	weighted residuals (2-dim-flat plate wing)	Equal straight line segments	Comparison with analytical solution give good agreement.	The F.E.M. is used here to solve the integral equation for the lift coefficient over the wing.
	23	Variational principle (two-dim cascades)	Triangular and isoparametric	Results are compared with exact solutions, experimental measurements and other numerical methods (singularity). It is noted that some of singularity methods are more efficient than the F.E.M. as regards computer time and storage, but calculate no results away from the aerofoil surface.	The superiority of the F.E.M. or F.D.M. depends on the type of problem. It is thought that as the interacting bodies increase, the F.E.M. will be the more effective approach. See also the F.E.M. analysis for cascades ⁵⁶ .
-2- COMPRESSIBLE NON-VISCOUS	57	variational principle (Rayleigh's prob.)	Linear triangular	Rayleigh's problem of fully infinite compressible flow about a cylinder compares well with exact solution.	Divergence manipulation and Gauss divergence theorem reduce the integrand derivative order in the variational functional.
	58				See Table II-1-
	35	Galerkin's method (Prandtl-Meyer expansion)	Third order triangular element	The Prandtl-Meyer expansion problem is compared with exact results. As the angle change increases, the percentage of error increases.	Bad results exist for higher change in angles, this is due to the assumption of small perturbation (which is valid for small changes in angle). By using this assumption the non-linear system of equations is linearized. Better results could be obtained by retaining the non-linear terms and use iteration methods, or by using a second order perturbation ⁵⁹ .

TABLE 1 (Cont'd)

Type of the Steady Flow	Ref. No.	Method Used to Find Variables & Problem Involved	Type of Element (if any)	Comparison with Other Results (if any)	Comments
-3- INCOMPRESSIBLE VISCIOUS	60	Variational Principle		NONE	Frechet differentials are used to study the existence of a variational principle for nonlinear operators. See also 61.
	62	Galerkin's Method	Mixed Power Elements	Slider bearing problem and the flow around a cylinder show acceptable agreement with analytical solutions. Accurate velocity fields are obtained but the associated pressures were inaccurate.	The inaccuracy is due to using the same interpolation polynomial for both velocity and pressure fields. In that case mixed interpolation must be used for the variables.
	36 & 37	Variational Principle	Eighteen degrees $\Delta(\psi, \psi_x, \psi_y, \psi_{xx}, \psi_{yy}, \psi_{xy}$ at each node)	Fully developed parallel flow, circulatory flow in a square cavity, channel entrance flow and flow over a circular cylinder; compare well with existing values.	The situation of non-convergent solution at higher Re normally encountered in F.D.M. was never experienced here.
"	128	Restricted Variational principle	Cubic-linear near u, p formulation & 18 deg. of freedom elem. 36, 37	Fully developed plane Poiseuille flow and circulatory flow in a square cavity.	The mixed cubic-linear interpolation element for the primitive variables gives a simple but practical results (except for coarse grids), when compared with the 18-degrees of freedom element.

TABLE 1 (Cont'd)

Type of the Steady Flow	Ref. No.	Method Used to Find Variable & Problem Involved	Type of Element (if any)	Comparison with Other Results (if any)	Comments
-4- COMPRESSIBLE VISCOUS	51 & 63	Weighted Residuals	Different orders triangular elements	Finite Element formulation for compressible flow are presented without practical problems.	The pressure and velocity fields are given by the same quadratic polynomials.
	39	Variational Principle (slot with temperature gradient)	Sixteen Degrees Rectangular	Rectangular slot with a horizontal temperature gradient gives better results than the F.D.M. used by Davis ⁶ . F.F.M. is more powerful at high Re than the F.D.M. (instabilities)	The variational functional used here was proposed by Glansdorff & Prigogine ⁶⁴ . In the F.D.M., Davis was unable to satisfy the non-slip condition at the wall.
-5- <u>MISCELLANEOUS</u> HEAT CONDUCTION AND TORSION OF PRISMATIC SECTIONS	52	Variational Principle	Linear Triangular	Application to torsion of equilateral triangle and rectangular section for which both exact and F.D.M. solutions exist. Same average error as F.D.M.	Better accuracy will exist if we use F.E.M. instead of F.D.M. for the case of bimetallic bar with a central hole. The diffusion equation is also treated similarly ⁶⁵ .

TABLE 2 - UNSTEADY FLOW - F.E.M. APPLICATIONS.

Type of the Unsteady Flow	Ref. No.	Method Used to Find Variables and Problem Involved	Type of Element (if any)	Comparator With Other Results (if any)	Comments
-1- INCOMPRESSIBLE NON-VISCOUS	58 & 66	Collocation method (aerodynamic around complex configurations)	Structural Quadrilateral elements	Results are presented for steady and oscillatory, subsonic and supersonic, wing and wing-body combination. Comparison was done with singularity methods and experimental results. For the oscillatory cases the results are not accurate as in the steady case.	The F.E.M. is used here to model the surface of the aircraft and its wake by using small quadrilateral elements. The unknown (velocity potential) is assumed to be constant within each element. Hence the integral equation relating the unknown potential on the surface of the body to the known downwash, reduces to a system of algebraic equations. The accuracy of the method is not assessed. Better results could be obtained if the assumption of constant velocity potential is removed.
-2- COMPRESSIBLE NON-VISCOUS	58				See Table II-1-
	67	(Planar rectangular wing)	Structural Quadrilateral and triangular elements	Results on a planar rectangular wing are presented and compared with the linearized three-dimensional lifting surface theory in supersonic flow. ⁶⁸⁻⁷⁰ Agreement is limited.	This paper combines the analytical formulation of the integrated potential approach, with numerical solution approaches using an arbitrary finite element mesh. The author reports that the source of inaccuracy is in the formulation of improper integrals.
-3- INCOMP. VISCOUS	71 To 74	Variational principle (oscillating airfoil & circular cylinder)	Triangular elements (third order for stream function & linear for pressure)	Results for the steady pressure distribution over a circular cylinder in viscous flow agree well with F.D.M.	Non-linearities were treated by the aid of a perturbation technique by assuming that vorticity field is not sensitive to incremental changes in velocity. The validity of this approach was not assessed, since there is no comparison with experimental results. Higher computation time is involved which increases by increasing the Regnold's number.

TABLE 2 (Cont'd)

Type of the Unsteady Flow	Ref. No.	Method Used to Find Variables and Problem Involved	Type of Element (if any)	Comparison With Other Results (if any)	Comments
	128	Restricted Variational principle.	Cubic-linear u, v, p elem.	Transient flow in a square cavity.	A segregated (iteration) u, v, p method has been used, and found to be of limited scope in terms of Reynolds number (less than 400). This was due to the linearization of the convective terms.
-4- COMPRESSIBLE VISCOUS	75			None	Finite element analogue for the proper Navier Stokes equations are presented but without a practical numerical method
-5- MISCELLANEOUS	76	Variational Principle (Helmholtz's equation)	Different orders of triangular elements (1st to 4th)	The free vibration modes of an isosceles right triangular, flexible membrane were calculated, % error for different elements is acceptable.	Few high-order elements suffice to model problems with high accuracy, the only disadvantage is the greater difficulty in approximating the complicated boundary shapes.
	49	Galerkin's method (Transient heat conduction)	Curved 2-dim. 77, 78 & 3-dim. 79-81 isoparametric elements	Circular opening with a heat transfer boundary heat loss from a cube and turbine-blade temp. variation give excellent agreement with exact solution.	Initial values of the function are known at $t = 0$ (or assumed) and a recurrence process is used to find the solution at subsequent times.
	82	Galerkin's method (Two-dim. incomp. stratified flow)	eighteen degrees of freedom Δ with Mid-Nodes	Comparison is done between the observed and F.E. values for the flow near a submerged weir. The turbulent exchange coeffs. require calibration.	The degrees of freedom are two velocities at each vertex and pressure + density & 2 velocities at mid. pts. A new approach to turbulent flow is required for better results.
	83	Variational principle (viscous incomp. transient flow-internal passage (Poiseuille flow))	Linear triangular	For different contraction ratios, similar results to the F.D.M. are obtained.	The over-all truncation error of the F.E.M. is found to be of the same order of magnitude as that of F.D.M. Similar analysis was done using Galerkin's method ⁸⁴ for transient flow in the channel entrance.

TABLE 3. FINITE ELEMENT - GALERKIN RESULTS FOR THE
BLASIUS PROBLEM

Y	Blasius			Third Order (6 Elements)			(2 Elements)			Fifth Order Finite Element (4 Elements)						(6 Elements)		
	f	f'	f''	f	f'	f''	f	f'	f''	f	f'	f''	f	f'	f''	f	f'	f''
0	0	0	.33206	0	0	0	0	0	0	.31076	0	0	.33015	0	0	0	0	.33201
.8	.10611	.26471	.32739	.10789	.26919													
1.2	.23795	.39378	.31659															
2	.65003	.62977	.26675	.53207	.43374								.64969	.62965	.26696	.23795	.39377	.31656
2.4	.92230	.72899	.22809															
3.4	1.74698	.90177	.11788	1.50718	.91154											.92231	.72899	.22791
3.6	1.92954	.92333	.09809															
4	2.30576	.95552	.06424													1.92953	.92333	.09834
5	3.28329	.99155	.01591	3.03354	.98102					2.29687	.95512	.06033	2.30526	.95548	.06452			
6	4.27964	.99898	.00240													3.28329	.99154	.01589
6.4	4.67938	.99961	.00098	4.42130	.99707								4.27913	.99897	.00264			
8	6.27923	1.0000	.00001	6.01962	1.0000					.00001	6.26961	1.0000	.00001	6.27873	1.0000	.00001	6.27924	1.0000
Average Error%			11.		7.		.27	.04	6.24	.02	.01	2.77	.00	.00	.91			

TABLE 4. FINITE ELEMENT SOLUTION FOR
SUCTION FLOW $C = .5$

Y	Theoretical Results ¹⁰⁴			Finite Elements		
	f	f'	f''	f	f'	f''
0	.5	0	2.0913	.5	0	2.0862
.4	.6563	.7543	1.6763	.6563	.7543	1.6787
.8	1.0798	1.3309	1.1936	1.0798	1.3309	1.1910
1.2	1.6940	1.7058	.6088	1.6939	1.7058	.6912
1.8	2.804	1.9459	.1799	2.8048	1.9459	.1788
2.8	4.7906	1.9992	.0040	4.7906	1.9992	.0034
4.0	7.1905	2.0	0.	7.1904	2.0	0.
Average Error %				.0000	0.	.31

TABLE 5. FINITE ELEMENT SOLUTION FOR

SUCTION FLOW $C = 1.5$

Y	Theoretical Results ¹⁰⁴			Finite Elements		
	f	f'	f''	f	f'	f''
0.	1.5	0.	3.7817	1.5	0.	3.7777
.4	1.7494	1.1291	2.0048	1.7494	1.1291	2.0049
.8	2.3279	1.6896	.8934	2.3279	1.6896	.8926
1.2	3.0564	1.9140	.3053	3.0564	1.9140	.3061
1.7	4.0372	1.9883	.0519	4.0372	1.9883	.0508
2.5	5.6348	1.9998	.0010	5.6348	1.9998	.0009
3.5	7.6348	2.	0.	7.6348	2.	0.
Average Error %				0.	0.	.5

TABLE 6. FINITE ELEMENT SOLUTION FOR

SUCTION FLOW $C = 3.0$

Y	Theoretical Results ¹⁰⁴			Finite Elements		
	f	f'	f''	f	f'	f''
0.	3.	0.	6.5289	3.	0.	6.5258
.2	3.1079	.9803	3.5562	3.1079	.9803	3.5568
.4	3.3622	1.5067	1.8653	3.3622	1.5067	1.8643
.6	3.6936	1.7759	.9220	3.6936	1.7759	.924
1.	4.4509	1.9625	.1813	4.4509	1.9625	.1795
1.6	5.6438	1.9985	.0088	5.6438	1.9985	.0079
2.5	7.4436	2.	0.	7.4436	2.	0.
Average Error %				0.	0.	.2

TABLE 7. FINITE ELEMENT SOLUTION FOR
SUCTION FLOW $C = 5$.

Y	Theoretical Results ¹⁰⁴			Finite Elements		
	f	f'	f''	f	f'	f''
0.	5.	0.	10.3596	5.	0.	10.3505
.2	5.1523	1.3070	3.7698	5.1523	1.3070	3.7667
.4	5.4685	1.7741	1.3054	5.4685	1.7741	1.3054
.6	5.8420	1.9314	.4215	5.8420	1.9314	.4202
.9	6.4327	1.9900	.0669	6.4327	1.9900	.0663
1.3	7.2314	1.9994	.0044	7.2314	1.9994	.0040
2.	8.6313	2.	0.	8.6313	2.	0.
Average Error %				0.	0.	.3

3

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

TABLE 8. FINITE ELEMENT SOLUTION FOR

BLOWING FLOW $C = -.5$

Y	Theoretical Results ¹⁰⁴			Finite Elements		
	f	f'	f''	f	f'	f''
0.	-.5	0.	.6579	-.5	0.	.6579
.6	-.3691	.4572	.8658	-.3691	.4572	.8660
1.2	.0703	1.0176	.9631	.0703	1.0176	.9627
1.8	.8463	1.5469	.7433	.8463	1.5469	.7431
2.6	2.2632	1.9234	.2186	2.2632	1.9234	.2201
3.6	4.2405	1.9982	.0085	4.2405	1.9982	.0074
5.	7.0401	2.	0.	7.0401	2.	0.
Average Error %				0.	0.	.15

TABLE 9 - FINITE ELEMENT SOLUTION FOR
STAGNATION FLOW ($\beta = 1$)

Y	Theoretical Results ¹⁰⁶			Finite Elements		
	f	f'	f''	f	f'	f''
0.	0.	0.	1.2326	0.	0.	1.2336
.8	.3124	.6859	.5251	.3124	.6859	.5254
1.6	.9798	.9324	.1474	.9798	.9324	.1474
2.4	1.7553	.9906	.0260	1.7553	.9906	.0259
3.2	2.5523	.9992	.0028	2.5523	.9992	.0028
4.0	3.3521	.9999	.0001	3.3521	.9999	.0001
4.8	4.1521	1.	0.	4.1521	1.	0.
Average Error %				0.	0.	.01

TABLE 10 - FINITE ELEMENT SOLUTION FOR
WEDGE FLOW ($\beta = .5$)

Y	Theoretical Results ¹⁰⁶			Finite Elements		
	f	f'	f''	f	f'	f''
0	0	0	.9278	0	0	.9277
.8	.2544	.5833	.5347	.2544	.5833	.5348
1.6	.8550	.8761	.2192	.8550	.8761	.2192
2.4	1.6045	.9761	.0564	1.6045	.9761	.0564
3.2	2.3963	.9972	.0085	2.3963	.9972	.0084
4.	3.1955	.9998	.0007	3.1955	.9998	.00078
4.8	—	1.	0.	3.9954	1.	0.
Average Error %				0.	0.	.1

TABLE 12. Velocity u at $T = 1.2$ (20 Elements)

Y	T = 1.2		Y	T = 1.2		T = 1.2
	Case (a)	Case (b)		Case (a)	Case (b)	
0	0	0	2.375	.88295	.87827	
0.25	.12431	.12494	2.75	.93566	.93040	
0.50	.24573	.24705	3.125	.96658	.96161	
0.75	.36425	.36633	3.5	.98653	.98246	
1.00	.47425	.47627	4.0	.99605	.99415	
1.25	.57571	.57686	4.5	.99802	.99708	
1.50	.66533	.66521	5.0	.99901	.99854	
1.75	.74313	.74130	5.5	.99951	.99927	
2.0	.80843	.80522	6.25	1.0000	1.0000	

TABLE 13. Choice of Time Steps - Quadratic Element

t ($T1 = t$)	Δt (to the 8th fig.)	T2	T3	$^*(T1 = t)$	Δt	T2	T3
1.00663296	.00663296	<1	<1	128/75	.06826667	512/375	256/225
1.024	.01736704	<1	<1	16/9	.07111111	64/45	32/27
1.048576	.024576	<1	<1	1.92	.14222222	1.536	1.28
16/15	.01809067	<1	<1	1.96608	.04608	1.572864	1.31072
2048/1875	.0256	<1	<1	2.0	.03392	1.6	4/3
256/225	.04551111	<1	<1	2.048	.048	1.6384	512/375
32/27	.04740741	<1	<1	32/15	.08533333	128/75	64/45
1.2	.01481481	<1	<1	20/9	.08888889	16/9	40/27
1.2288	.0288	<1	<1	2.4	.17777778	1.92	1.6
				2.4576	.0576	1.96608	1.6384
1.2582912	.0294912	1.00663296	<1	2.56	.1024	2.048	128/75
1.28	.0217088	1.024	<1	8/3	.10666667	32/15	16/9
1.31072	.03072	1.048576	<1	3.0	.33333333	2.4	2
4/3	.02261333	16/15	<1	3.072	.072	2.4576	2.048
512/375	.032	2048/1875	<1	3.2	.128	2.56	32/15
64/45	.05688889	256/225	<1	10/3	.13333333	8/3	20/9
40/27	.05925926	32/27	<1	3.84	.50666667	3.072	2.56
1.5	.01857852	1.2	1	4.0	.16	3.2	8/3
				4.8	.8	3.84	3.2
1.536	.036	1.2288	1.024	5.0	.2	4.0	10/3
1.572864	.036864	1.2582912	1.048576	6.0	1.0	4.8	4.0
1.6	.027136	1.28	16/15	7.5	1.5	6.0	5.0
1.6384	.0384	1.31072	2048/1875				

TABLE 14. Velocity Component u(T) -
(20 elements)

Y	u(1) (RAYLEIGH)	u(1.5)	u(2)	u(3)	u(4)	u(6)	u(7.5)	u(∞) (BLASIUS)
0.0	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
0.25	.14032	.11598	.10031	.08720	.08361	.08301	.08300	.08300
0.50	.27633	.23136	.19880	.17372	.16688	.16594	.16593	.16590
0.75	.40411	.34613	.29548	.25956	.24980	.24833	.24833	.24833
1.0	.52050	.46564	.38757	.34323	.33130	.32980	.32979	.32979
1.25	.62323	.55507	.47507	.42475	.41137	.40958	.40958	.40958
1.50	.71116	.64088	.55675	.50248	.48835	.48681	.48681	.48681
1.75	.78406	.72308	.63079	.57642	.56226	.56054	.56054	.56054
2.0	.84270	.77307	.69786	.64481	.63104	.62978	.62978	.62977
2.375	.90692	.85304	.78429	.73698	.72463	.72328	.72327	.72327
2.75	.94817	.91390	.85301	.81335	.80302	.80220	.80219	.80219
3.125	.97287	.94609	.90403	.87390	.86620	.86537	.86537	.86536
3.5	.98667	.96396	.94168	.92008	.91482	.91307	.91307	.91306
4.0	.99532	.98182	.97110	.95929	.95699	.95553	.95552	.95552
4.5	.99854	.99091	.98887	.98369	.98308	.97954	.97953	.97953
5.0	.99959	.99546	.99500	.99329	.99307	.99156	.99156	.99155
5.5	.99990	.99973	.99872	.99869	.99863	.99690	.99690	.99689
6.25	1.0000	1.0000	.99980	.99978	.99952	.99945	.99945	.99945
7.0	-	-	.99990	.99989	.99989	.99989	.99989	.99992
7.75	-	-	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

TABLE 15. Comparison Between Two Grids
For Values of $u(1.2)$

(a) 20 Elements

(b) 10 Elements

Y	(a)	(b)
.50	.24705	.24139
1.00	.47627	.46172
1.50	.66521	.66097
2.00	.80522	.81054
2.75	.93566	.94172
3.50	.98653	.97086
4.50	.99802	.98543
5.50	.99951	.99272
7.00	1.0000	.99636
Average Difference	-	1.%

TABLE 16. Surface Shear
(20 Elements)

T	Computed by Finite Element	Rayleigh	Blasius
1	0.56419	0.56419	
1.5	0.46511	0.46066	
2	0.40487	0.39894	
3	0.35017	0.32573	0.33206
4	0.33512	0.28209	0.33206
6	0.33220		0.33206
7.5	0.33216		0.33206

TABLE 17.

Residuals of the Continuity Equation
And the Predicted Relative Errors Associated
With the Nodal Values at $T = 1$. (Rayleigh)

Y	10 Elements		14 Elements		20 Elements	
	Residuals	Predicted error in u	Residuals	Predicted error in u	Residuals	Predicted error in u
.25			.002573	.01949	.00000	.0000
.50	.007881	.0260	.000444	.00413	-.00001	.0001
.75					.00000	.0000
1.00	.003101	.0170	.005953	.0160	.00000	.0000
1.25					.00000	.0000
1.50	.003792	.0104	.002874	.0079	.00000	.0000
1.75					.00000	.0000
2.00	.002505	.0152			.00002	.0004
2.125			.003162	.0068		
2.375					.00000	.0000
2.75	.001565	.0029	.002452	.00366	-.00002	.0002
3.125					.00001	.0002
3.375			.000840	.0018		
3.50	.001088	.0046			.00000	.0000
4.00			-.000209	.0011	.00001	.00005
4.50	.000202	.0003			-.00002	.00008
4.75			.000025	.0001		
5.00					.00000	.00000
5.50	-.000130	.0004	-.000024	.0001		
6.25			.000003	.0000		
7.00	.000018	.0000	.00000			
7.75						
8.50	-.000006	.0003				
Average error %		1. %		.06 %		.006 %
Maximum error %	3. %		2. %		.04 %	

TABLE 18 - INTEGRALS ASSOCIATED WITH THE AREA COORDINATES

P_t	q	m	n	p	P_t	q	m	n	p	P_t	q	m	n	p
0	0	0	0	2	6	4	2	0	840	8	4	4	0	6300
1	1	0	0	6		3	3	0	1120		5	2	1	15120
2	2	0	0	12		4	1	1	1680		4	3	1	25200
	1	1	0	24		3	2	1	3360		4	2	2	37800
3	3	0	0	20		2	2	2	5040		3	3	2	50400
	2	1	0	60	7	7	0	0	72	9	9	0	0	110
	1	1	1	120		6	1	0	504		8	1	0	990
4	4	0	0	30		5	2	0	1512		7	2	0	3960
	3	1	0	120		4	3	0	2520		7	1	1	7920
	2	2	0	180		5	1	1	3024		6	3	0	9240
5	2	1	1	360		4	2	1	7560		5	4	0	13860
	5	0	0	42	8	3	3	1	10080		6	2	1	27720
	4	1	0	210		3	2	2	15120		5	3	1	55440
	3	2	0	420		8	0	0	90		4	4	1	69300
	3	1	1	840		7	1	0	720		5	2	2	83160
6	2	2	1	1260		6	2	0	2520		4	3	2	138600
	6	0	0	56		6	1	1	5040		3	3	3	184800
	5	1	0	336		5	3	0	5040					

Table 18 (Continue)

P_t	q	m	n	P	P_t	q	m	n	P	P_t	q	m	n	P
10	10			132	11	9	2		8580	12	10	1	1	24024
	9	1		1320		9	1	1	17160		9	3		40040
	8	2		5940		8	3		25740		9	2	1	120120
	8	1	1	11880		8	2	1	77220		8	4		90090
	7	3		15840		7	4		51480		8	3	1	360360
	7	2	1	47520		7	3	1	205920		8	2	2	540540
	6	4		27720		7	2	2	308880		7	5		144144
	6	3	1	110880		6	5		72072		7	4	1	720720
	6	2	2	166320		6	4	1	360360		7	3	2	1441440
	5	5		33264		6	3	2	720720		6	4	2	2522520
11	5	4	1	166320	12	5	4	2	1081080		6	3	3	3363360
	5	3	2	332640		5	2	3	1441440		5	4	3	5045040
	4	4	2	415800		4	4	3	1801800		4	4	4	6306300
	4	3	3	1108800		12			182					
	11			156		11	1		2184					
11	10	1		1716		10	2		12012					

TABLE 19 - ELEMENT MATRIX $[D_b]$ FOR THE CONTINUITY EQUATION

$\frac{b_1}{15}$	$\frac{-b_2}{30}$	$\frac{-b_3}{30}$	$\frac{1}{15} (b_2 - \frac{b_1}{2})$	$\frac{-1}{30} (b_3 + b_2)$	$\frac{1}{15} (b_3 - \frac{b_1}{2})$
$\frac{-b_1}{30}$	$\frac{b_2}{15}$	$\frac{-b_3}{30}$	$\frac{1}{15} (b_1 - \frac{b_2}{2})$	$\frac{1}{15} (b_3 + \frac{b_2}{2})$	$\frac{-1}{30} (b_3 + b_1)$
$\frac{-b_1}{30}$	$\frac{-b_2}{30}$	$\frac{b_3}{15}$	$\frac{-1}{30} (b_1 + b_2)$	$\frac{1}{15} (b_2 - \frac{b_3}{2})$	$\frac{1}{15} (b_1 - \frac{b_3}{2})$
$\frac{b_1}{10}$	$\frac{b_2}{30}$	$\frac{-b_3}{30}$	$\frac{4}{15} (b_1 + b_2)$	$\frac{4}{15} (b_3 + \frac{b_2}{2})$	$\frac{4}{15} (b_3 + \frac{b_1}{2})$
$\frac{-b_1}{30}$	$\frac{b_2}{10}$	$\frac{b_3}{10}$	$\frac{4}{15} (b_1 + \frac{b_2}{2})$	$\frac{4}{15} (b_2 + b_3)$	$\frac{4}{15} (b_1 + \frac{b_3}{2})$
$\frac{b_1}{10}$	$\frac{-b_2}{30}$	$\frac{b_3}{10}$	$\frac{4}{15} (b_2 + \frac{b_1}{2})$	$\frac{4}{15} (b_2 + \frac{b_3}{2})$	$\frac{4}{15} (b_1 + b_3)$

$$[D_b]_m$$

Table 21. Element Matrix $[AD_2^I]$

$$[AD_2^I] = \begin{bmatrix} 3 & -3 & 1 & -1 & 30 & 12 & 3 & 1.5 & 1 & 3 & -3 & 1 & -1 \\ 30 & 12 & 3 & 1.5 & 1 & 3 & -3 & 1 & -1 & 30 & 12 & 3 & 1.5 & 1 \end{bmatrix}$$

Table 22. Element Matrix $[AD_2^{II}]$

$$[AD_2^{II}] = \begin{bmatrix} 30 & 3 & 12 & 1 & 1.5 & -3 & 3 & -1 & 1 & 30 & 3 & 12 & 1 & 1.5 \\ -3 & 3 & -1 & 1 & 1 & -30 & 3 & 12 & 1 & 1.5 & -3 & 3 & -1 & 1 \end{bmatrix}$$

Table 23. Time Steps for the Quintic Element

T_1 (At $x = 1$)	T_2 (At $x = 1.5$)
32/27	1
4/3	1
40/27	1
1.5	1
16/9	32/27
2	4/3
20/9	40/27
8/3	16/9
3	2
10/3	20/9
4	8/3
5	10/3
6	4
7.5	5

Table 24. Comparison Between Different Grids and Different Elements
for u at $T = 1.2$

Y	(a) 10 Quadratic (19 unknowns)	(b) 6 Quintic (16 unknowns)	(c) 10 Quintic (28 unknowns)	(d) 12 Quintic (34 unknowns)	(e) 20 Quadratic (39 unknowns)
.5				.24574	.24573
1.5	.66097	.66125	.66433	.66535	.66533
2.75			.93449	.93569	.93566
4.5	.98543	.98640	.99727	.99800	.99802
6.25			.99970	1	1
8.5		1	1	1	1
Average Difference from (e) %	.97	.9	.1	0	-

Table 25. Comparison Between Different Grids and Different Elements
for u and Wall Shear at $T = 2$

(A) Velocity u

(B) Shear at Wall

(A)				
Y	(a) 6 Quintic	(b) 12 Quintic	(c) 20 Quadratic	
.5		.19878	.19880	
1.5	.55130	.55614	.55615	
2.75		.85298	.85301	
4.5	.98072	.98888	.98887	
6.25		.99981	.99980	
8.5	1	1	1	
Aver. Diff. % from (c)	.85	0	-	

(B)			
Y	(a)	(b)	(c)
0	.38712	.40486	.40487
Difference % from (c)	4.4	0	-

Table 26. Comparison Between the Quintic and Quadratic
Elements for u and Wall Shear at $T = 3$

(A) Velocity u

(B) Shear at Wall

(A)		
Y	(a) 12 Quintic	(b) 20 Quadratic
.5	.17370	.17372
1.5	.50245	.50248
2.75	.81333	.81335
4.5	.98368	.98369
6.25	.99977	.99978
8.5	1	1
Aver. Diff. % from (b)	0	-

(B)		
Y	(a)	(b)
0	.35015	.35017
Difference % from (b)	.01	-

Table 27. Comparison of the Computer Time for Different Grids of Different Elements.

Test Case	Element Type	Number of Elements	Number of Unknowns	Aver. Devia. of Velocity from (d) %	Computer Time per Inter. (sec.)	Number of Time Inter-vals	Total Computer Time (sec.)
a	Quadratic	10	19	.97	.74	43	33
b	Quintic	6	16	.88	5.6	14	78
c	Quintic	12	34	0	27.5	14	385
d	Quadratic	20	39	-	7.4	43	330

* Executive Time on Fortran Load And Go System on SIGMA 9 Computer at Carleton University.

Table 28. Time Marching Results for ψ
(12 Elements)

T Y	RAYLEIGH			BLASIUŠ		
	1.2	2	3	1.2	2	3
0	0	0	0	0	0	0
.5	.06979	.06790	.05093	.06036	.04150	.04150
1.5	.58126	.56719	.44050	.51089	.37015	.37015
2.75	1.64942	1.61880	1.34360	1.49650	1.19066	1.19066
4.5	3.37214	3.33333	2.98410	3.17810	2.79009	2.79009
6.25	5.12162	5.08390	4.74370	4.93270	4.52946	4.52946
8.5	7.37162	7.32870	6.94240	7.15700	6.77923	6.77923

Table 29. Time Marching Results for ψ_x
(12 Elements)

T Y	RAYLEIGH			BLASIUS		
	0	1.2	2	3	4	5
0	0	0	0	0	0	0
.5	0	-.00099	-.00492	-.01831	-.02073	
1.5	0	-.00543	-.02718	-.14138	-.18003	
2.75	0	-.01202	-.05990	-.38192	-.50768	
4.5	0	-.01679	-.07990	-.67498	-.80889	
6.25	0	-.01809	-.09042	-.76642	-.85850	
8.5	0	-.01847	-.09234	-.78093	-.86039	

Table 30. Time Marching Results for ψ_y
(12 Elements)

T Y	RAYLEIGH				BLASIUS			
	1.2	2	3		1.2	2	3	
0	0	0	0	0	0	0	0	0
.5	.27633	.24574	.19878	.17370	.16590			
1.5	.71116	.66535	.55614	.50245	.48681			
2.75	.94817	.93569	.85298	.81333	.80219			
4.5	.99854	.99800	.98888	.98368	.97953			
6.25	1	1	.99981	.99977	.99945			
8.5	1	1	1	1	1			

Table 31. Time Marching Results for ψ_{xx}
(12 Elements)

T Y	RAYLEIGH			BLASIUS		
	1.2	2	3	1.2	2	3
0	0	0	0	0	0	0
.5	0	.00621	.01242	.03105		
1.5	0	.01041	.10409	.26023		
2.75	0	.02450	.24496	.61244		
4.5	0	.02306	.23059	.57647		
6.25	0	.01791	.17913	.44302		
8.5	0	.01611	.16106	.43019		

Table 32. Time Marching Results for ψ_{xy}
(12 Elements)

T Y	RAYLEIGH			BLASIUS		
	1.2	2	3	1.2	2	3
0	0	0	0	0	0	0
.5	-.00157	-.00784	-.01567	-.08273		
1.5	-.00756	-.03783	-.07567	-.22695		
2.75	-.01316	-.06579	-.13159	-.26079		
4.5	-.00341	-.01705	-.03409	-.07646		
6.25	-.00041	-.00205	-.00410	-.00943		
8.5	0	0	0	0		

Table 33. Time Marching Results for ψ_y
(12 Elements)

T \ Y	RAYLEIGH			BLASIUS		
	1.2	2	3			
0	.56420	.40486	.35015			.33206
.5	.53000	.39600	.34820			.33093
1.5	.32145	.31370	.30686			.30259
2.75	.08517	.15373	.18063			.18967
4.5	.00355	.01460	.02320			.03398
6.25	0	.00094	.00117			.00340
8.5	0	0	0			0

Table 34. Linearized Versus Exact Solution for
the Steady Flow over a Flat Plate

Y	Present and Exact Solution			Linearized Solution		
	f	f'	f''	f	f'	f''
0	0	0	.33201	0	0	.30200
1.2	.23795	.39377	.31656	.22661	.37544	.31058
2.4	.92231	.72899	.22791	.88589	.70727	.22576
3.6	1.92953	.92333	.09834	1.86932	.90772	.10865
5	3.28329	.99154	.01589	3.21103	.98842	.02075
6.4	4.67938	.99961	.00093	4.60548	.99941	.00143
8	6.27924	1	0	6.20526	1	0
Average Error%	-	-	-	2.8	1.9	17.8

Table 35. Error Estimate in the Linearization of
the Suddenly Accelerated Plate Problem.

(a) $T_j = 1.875$ and $T_{j+1} = 2$

(b) $T_j = 2.75$ and $T_{j+1} = 3$

Case	Y	u_{j+1}	u_j	$\frac{(\Delta u)_{j+1,j}^2}{(u)_{j+1}^2} \%$	$\frac{u_{F.E.} - u_{F.D.}}{u_{F.E.}} \%$
(a)	1	.38757	.39971	.098	.13
	2	.69786	.71199	.041	.063
	4	.97110	.97466	.001	.070
	5	.99500	.99597	.000	.030
(b)	1	.34323	.35428	.100	.100
	2	.64481	.65975	.053	.060
	4	.95929	.96244	.001	.076
	5	.99329	.99342	.000	.090

APPENDIX A

ELEMENTAL MATRICES FOR STEADY FLOW

The elemental matrices are developed using exact integration over the domain of interest. For the present case the integration is performed over the length of the element given by s_m .

A.1 THIRD ORDER ELEMENT

$$[N(Y)] = [1 \ Y \ Y^2 \ Y^3] \quad (A.1)$$

and

$$\{f^e\}_m = [f_1 \ f_1' \ f_2 \ f_2']_m^T \quad (A.2)$$

where the subscripts 1 and 2 refer to node number 1 and node number 2 respectively. From Equation (5.1) we have:

$$f_m(Y) = [1 \ Y \ Y^2 \ Y^3] \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{Bmatrix}_m \quad (A.3)$$

Substituting in Equation (A.3) the nodal coordinates zero and s we obtain the nodal variables given in Equation (A.2) as follows;

$$\begin{pmatrix} f_1 \\ f_1' \\ f_2 \\ f_2' \end{pmatrix}_m = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & s & s^2 & s^3 \\ 0 & 1 & 2s & 3s^2 \end{bmatrix}_m \begin{pmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{pmatrix}_m \quad (\text{A.4})$$

From Equation (A.4) and Equation (5.5), the matrix $[B]_m$ is then;

$$[B]_m = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3/s^2 & -2/s & 3/s^2 - 1/s \\ 2/s^3 & 1/s^2 & -2/s^3 & 1/s^2 \end{bmatrix}_m \quad (\text{A.5})$$

The matrix $[G_s]_m$ in Equation (5.12) is obtained from Equation (5.10) as;

$$[G_s]_m = [B]_m^T \left[\int_0^{s_m} [N(Y)]^T \{f^e\}_m^T \cdot [B]_m^T [N(Y)]^T [N''(Y)] dY \right] [B]_m \quad (\text{A.6})$$

By exact integration of Equation (A.6) from zero to an element of length s (it is understood that s can vary from one element to another) we get a four by four matrix, $[G_s]_m$, whose components are:

$$G_s(1, 1) = 2(-3f_1/5s - 2f_1'/35 + f_2'/140) \quad (A.7.1)$$

$$G_s(1, 2) = 2(-17f_1/35 - 23sf_1'/420 - 9f_2'/140 + 2sf_2'/105) \quad (A.7.2)$$

$$G_s(1, 3) = -G_s(1, 1) \quad (A.7.3)$$

$$G_s(1, 4) = 2(-4f_1/35 - sf_1'/420 + 9f_2'/140 - sf_2'/84) \quad (A.7.4)$$

$$G_s(2, 1) = 2(-2f_1/35 - sf_1'/140 + f_2'/140) \quad (A.7.5)$$

$$G_s(2, 2) = 2(-23sf_1'/420 - s^2f_1'/120 - sf_2'/84 + s^2f_2'/280) \quad (A.7.6)$$

$$G_s(2, 3) = -G_s(2, 1) \quad (A.7.7)$$

$$G_s(2, 4) = 2(-sf_1'/420 + s^2f_1'/840 + 2sf_2'/105 - s^2f_2'/280) \quad (A.7.8)$$

$$G_s(3, 1) = 2(f_1'/140 + 3f_2'/5s - 2f_2'/35) \quad (A.7.9)$$

$$G_s(3, 2) = 2(-9f_1'/140 - sf_1'/84 + 4f_2'/35 - sf_2'/420) \quad (A.7.10)$$

$$G_s(3, 3) = -G_s(3, 1) \quad (A.7.11)$$

$$G_s(3, 4) = 2(9f_1'/140 + 2sf_1'/105 + 17f_2'/35 - 23sf_2'/420) \quad (A.7.12)$$

$$G_s(4, 1) = 2(f_1'/140 - 2f_2'/35 + sf_2'/140) \quad (A.7.13)$$

$$G_s(4, 2) = 2(2sf_1'/105 + s^2f_1'/280 - sf_2'/420 - s^2f_2'/840) \quad (A.7.14)$$

$$G_s(4, 3) = -G_s(4, 1) \quad (\text{A.7.15})$$

$$G_s(4, 4) = 2(-sf_1/84 - s^2f_1/280 - 23sf_2/420 + s^2f_2/120) \quad (\text{A.7.16})$$

The matrix $[D_s]_m$ in Equation (5.12) is obtained from Equation (5.10) as follows;

$$[D_s]_m = 2 [B]_m^T \left[\int_0^s [N(Y)] [N'(Y)] dY \right] [B]_m \quad (\text{A.8})$$

Effectuating the integrations in Equation (A.8) to get the following;

$$[D_s]_m = \begin{bmatrix} 12/s^2 & 6/s & -12/s^2 & 6/s \\ 2/s & 1 & -2/s & 1 \\ 12/s^2 & 6/s & -12/s^2 & 6/s \\ -2/s & -1 & 2/s & -1 \end{bmatrix}_m \quad (\text{A.9})$$

A.2 QUINTIC ORDER ELEMENT

$$[N(Y)] = \begin{bmatrix} 1 & Y & Y^2 & Y^3 & Y^4 & Y^5 \end{bmatrix} \quad (A.10)$$

and

$$\{f^e\}_m = \begin{bmatrix} f_1 & f_1' & f_1'' & f_2 & f_2' & f_2'' \end{bmatrix}_m^T \quad (A.11)$$

Following the same procedure as in Sec. (A.1), the matrix

 $[B]_m$ is:

$$[B]_m = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 0 \\ -10/s^3 & -6/s^2 & -3/2s & 10/s^3 & -4/s^2 & 1/2s \\ 15/s^4 & 8/s^3 & 3/2s^2 & -15/s^4 & 7/s^3 & -1/s^2 \\ -6/s^5 & -3/s^4 & -1/2s^3 & 6/s^5 & -3/s^4 & 1/2s^3 \end{bmatrix} \quad (A.12)$$

Substituting in Equation (A.6) we get the components of the matrix $[G_s]_m$ as follows;

$$G_s(1, 1) = 2 \left(-10 h14/s^3 + 30 h26/s^6 + 15 h15/s^4 - 6 h16/s^5 \right. \\ \left. + 36 h66/s^{10} + 213 h46/s^8 - 135 h36/s^7 - 144 h56/s^9 \right) \quad (A.13.1)$$

$$\begin{aligned}
 G_s(1, 2) = & 2 (-6 h14/s^2 + 18 h26/s^5 + 8 h15/s^3 - 3 h16/s^4 \\
 & + 18 h66/s^9 + 564 h46/5s^7 - 75 h36/s^6 \\
 & - 369 h56/5s^8)
 \end{aligned}
 \tag{A.13.2}$$

$$\begin{aligned}
 G_s(1, 3) = & 2 (h13/2 - h16/s^3 + 21 h26/4s^4 - 321 h36/20s^5 \\
 & - 3 h14/2s + 106 h46/5s^6 + 3 h15/2s^2 \\
 & - 129 h56/10s^7 + 3 h66/s^8)
 \end{aligned}
 \tag{A.13.3}$$

$$\begin{aligned}
 G_s(1, 6) = & 2 (h14/2s - 3 h26/2s^4 + 33 h36/4s^5 - 149 h46/10s^6 \\
 & - h15/s^2 + 111 h56/10s^7 + h16/2s^3 - 3 h66/s^8)
 \end{aligned}
 \tag{A.13.4}$$

$$\begin{aligned}
 G_s(2, 1) = & 2 (-5 h15/s^3 + 12 h26/s^5 - 78 h36/s^6 + 117 h46/s^7 \\
 & + 9 h16/s^4 - 75 h56/s^8 + 18 h66/s^9)
 \end{aligned}
 \tag{A.13.5}$$

$$\begin{aligned}
 G_s(2, 2) = & 2 (-3 h15/s^2 + 39 h26/5s^4 - 216 h36/5s^5 \\
 & + 309 h46/5s^6 + 24 h16/5s^3 - 192 h56/5s^7 \\
 & + 9 h66/s^8)
 \end{aligned}
 \tag{A.13.6}$$

$$\begin{aligned}
 G_s(2, 3) = & 2 (h14/6 + 3 h16/5s^2 + 13 h26/5s^3 - 183 h36/20s^4 \\
 & - 3 h15/4s + 231 h46/20s^5 - 67 h56/10s^6 \\
 & + 3 h66/2s^7)
 \end{aligned}
 \tag{A.13.7}$$

$$\begin{aligned}
 G_s(2, 6) = & 2 (h15/4s - 2 h26/5s^4 + 24 h36/5s^4 - 33 h46/4s^5 \\
 & - 3 h16/5s^2 + 29 h56/5s^6 - 3 h66/2s^7)
 \end{aligned}
 \tag{A.13.8}$$

$$G_s(3, 1) = 2 (-3 h16/2s^3 + 9 h26/s^4 - 21 h36/s^5 + 24 h46/s^6 - 27 h56/2s^7 + 3 h66/s^8) \quad (A.13.9)$$

$$G_s(3, 2) = 2 (-9 h16/10s^2 + 51 h26/10s^3 - 57 h36/5s^4 + 63 h46/5s^5 - 69 h56/10s^6 + 3 h66/2s^7) \quad (A.13.10)$$

$$G_s(3, 3) = 2 (h15/24 - 3 h16/10s + 6 h26/5s^2 - 23 h36/10s^3 + 93 h46/40s^4 - 6 h56/5s^5 + h66/4s^6) \quad (A.13.11)$$

$$G_s(3, 6) = 2 (3 h16/40s - 21 h26/40s^2 + 11 h36/8s^3 + 69 h46/40s^4 + 21 h56/20s^5 - h66/4s^6) \quad (A.13.12)$$

$$G_s(4, 1) = 2 (-30 h26/s^6 + 135 h36/s^7 - 213 h46/s^8 + 144 h56/s^9 - 36 h66/s^{10}) \quad (A.13.13)$$

$$G_s(4, 2) = 2 (-18 h26/s^5 + 75 h36/s^6 - 564 h46/5s^7 + 369 h56/5s^8 - 18 h66/s^9) \quad (A.13.14)$$

$$G_s(4, 3) = 2 (h16/2s^3 - 21 h26/4s^4 + 321 h36/20s^5 - 106 h46/5s^6 + 129 h56/10s^7 - 3 h66/s^8) \quad (A.13.15)$$

$$G_s(4, 6) = 2 (3 h_{26}/2s^4 - 33 h_{36}/4s^5 + 149 h_{46}/10s^6 - 111 h_{56}/10s^7 + 3 h_{66}/s^8) \quad (A.13.16)$$

$$G_s(5, 1) = 2 (12 h_{26}/s^5 - 57 h_{36}/s^6 + 96 h_{46}/s^7 - 69 h_{56}/s^8 + 18 h_{66}/s^9) \quad (A.13.17)$$

$$G_s(5, 2) = 2 (36 h_{26}/5s^4 - 159 h_{36}/5s^5 + 51 h_{46}/s^6 - 177 h_{56}/5s^7 + 9 h_{66}/s^8) \quad (A.13.18)$$

$$G_s(5, 3) = 2 (-h_{16}/5s^2 + 43 h_{26}/20s^3 - 69 h_{36}/10s^4 + 193 h_{46}/20s^5 - 31 h_{56}/5s^6 + 3 h_{66}/2s^7) \quad (A.13.19)$$

$$G_s(5, 6) = 2 (-3 h_{26}/5s^3 + 69 h_{36}/20s^4 - 133 h_{46}/20s^5 + 53 h_{56}/10s^6 - 3 h_{66}/2s^7) \quad (A.13.20)$$

$$G_s(6, 1) = 2 (-3 h_{26}/2s^4 + 15 h_{36}/2s^5 - 27 h_{46}/2s^6 + 21 h_{56}/2s^7 - 3 h_{66}/s^8) \quad (A.13.21)$$

$$G_s(6, 2) = 2 (-9 h_{26}/10s^3 + 21 h_{36}/5s^4 - 36 h_{46}/5s^5 + 27 h_{56}/5s^6 - 3 h_{66}/2s^7) \quad (A.13.22)$$

$$G_s(6, 3) = 2 (h_{16}/40s - 11 h_{26}/40s^2 + 37 h_{36}/40s^3 - 11 h_{46}/8s^4 + 19 h_{56}/20s^5 - h_{66}/4s^6) \quad (A.13.23)$$

$$G_s(6, 6) = 2(3 h_{26}/40s^2 - 9 h_{36}/20s^3 + 37 h_{46}/40s^4 - 4 h_{56}/5s^5 + h_{66}/4s^6) \quad (A.13.24)$$

$$G_s(i, 4) = -G_s(i, 1) \quad \text{for } i = 1, 2, \dots, 6 \quad (A.13.25)$$

$$G_s(i, 5) = s G_s(i, 1) - G_s(i, 2), \quad i = 1, 2, \dots, 6 \quad (A.13.26)$$

where:

$$h_{13} = s(f_1/2 + s f_1'/10 + s^2 f_1''/120 + f_2/2 - s f_2'/10 + s^2 f_2''/120) \quad (A.14)$$

$$h_{14} = s^2(3 f_1/7 + 4s f_1'/35 + 3s^2 f_1''/280 + 15 f_2/14 - 13s f_2'/70 + s^2 f_2''/70) \quad (A.15)$$

$$h_{15} = s^3(5 f_1/14 + 3s f_1'/28 + 3s^2 f_1''/280 + 23 f_2/14 - s f_2'/4 + s^2 f_2''/56) \quad (A.16)$$

$$h_{16} = s^4(25 f_1/84 + 2s f_1'/21 + 5s^2 f_1''/504 + 185 f_2/84 - 25s f_2'/84 + 5s^2 f_2''/252) \quad (A.17)$$

$$h_{26} = s^5(f_1/6 + s f_1'/18 + s^2 f_1''/168 + 11 f_2/6 - 2s f_2'/9 + s^2 f_2''/72) \quad (A.18)$$

$$h_{36} = s^6 (10 f_1/99 + 8s f_1'/231 + s^2 f_1''/264 + 155 f_2/99 - 17s f_2'/99 + s^2 f_2''/99) \quad (A.19)$$

$$h_{46} = s^7 (5 f_1/77 + s f_1'/44 + s^2 f_1''/396 + 15 f_2/11 - 3s f_2'/22 + s^2 f_2''/132) \quad (A.20)$$

$$h_{56} = s^8 (25 f_1/572 + 20s f_1'/1287 + s^2 f_1''/572 + 345 f_2/286 - 95s f_2'/858 + 5s f_2''/858) \quad (A.21)$$

$$h_{66} = s^9 (25 f_1/819 + s f_1'/91 + 5s^2 f_1''/4004 + 295 f_2/273 - 25s f_2'/273 + 5s^2 f_2''/1092) \quad (A.22)$$

Substituting Equations (A.10 - 12) into Equation (A.8)

to get the matrix $\begin{bmatrix} D_s \end{bmatrix}_m$ as follows;

$$\begin{bmatrix} D_s \end{bmatrix}_m = \begin{bmatrix} 0 & -18/7s & -16/7 & 0 & 18/7s & -2/7 \\ 18/7s & 1 & -9s/70 & -18/7s & 11/7 & -11s/70 \\ 2/7 & 9s/70 & 0 & -2/7 & 11s/70 & -s^2/70 \\ 0 & 18/7s & 2/7 & 0 & -18/7s & 16/7 \\ -18/7s & -11/7 & -11s/70 & 18/7s & -1 & -9s/70 \\ 2/7 & 11s/70 & s^2/70 & -2/7 & 9s/70 & 0 \end{bmatrix} \quad (A.23)$$

Substituting by Equations (A.10, 11, 12) into Equation (5.29)

for $\begin{bmatrix} W_s \end{bmatrix}_m$ we get;

$$\begin{bmatrix} W_s \end{bmatrix}_m = \beta \begin{bmatrix} B \end{bmatrix}_m^T \left(\begin{bmatrix} AA \end{bmatrix}_m - \begin{bmatrix} AB \end{bmatrix}_m \right) \quad (\text{A.24})$$

$6 \times 6 \qquad 6 \times 1 \qquad 6 \times 1$

where $\begin{bmatrix} AA \end{bmatrix}_m$ is given by:

$$\begin{bmatrix} AA \end{bmatrix}_m = \begin{bmatrix} s & s^2/2 & s^3/3 & s^4/4 & s^5/5 & s^6/6 \end{bmatrix}_m^T \quad (\text{A.25})$$

and

$\begin{bmatrix} AB \end{bmatrix}_m$ is written in the form:

$$\begin{bmatrix} AB \end{bmatrix}_m = \begin{bmatrix} SI \end{bmatrix}_m \begin{bmatrix} B \end{bmatrix}_m \begin{Bmatrix} f^e \end{Bmatrix}_m \quad (\text{A.26})$$

$6 \times 6 \qquad 6 \times 6 \qquad 6 \times 1$

$\begin{bmatrix} SI \end{bmatrix}_m$ has the value of:

$$[SI]_m = \begin{bmatrix} 0 & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ 0 & k_{13}/2 & 2k_{14}/3 & 3k_{15}/4 & 4k_{16}/5 & k_{26} \\ 0 & k_{14}/3 & k_{15}/2 & 3k_{16}/5 & 4k_{26}/5 & k_{36} \\ 0 & k_{15}/4 & 2k_{16}/5 & 3k_{26}/5 & 4k_{36}/5 & k_{46} \\ 0 & k_{16}/5 & 2k_{26}/5 & 3k_{36}/5 & 4k_{46}/5 & k_{56} \\ 0 & k_{26}/5 & 2k_{36}/5 & 3k_{46}/5 & 4k_{56}/5 & k_{66} \end{bmatrix} \quad (A.27)$$

Furthermore the values of k_{ij} in Equation (A.27) are given by:

$$k_{12} = s^2 (f_1'/s + f_1''/2 + a_1 + a_2 + a_3) \quad (A.28)$$

$$k_{13} = s^3 (f_1'/s + 2 f_1''/3 + 6 a_1/4 + 8 a_2/5 + 10 a_3/6) \quad (A.29)$$

$$k_{14} = s^4 (f_1'/s + 3 f_1''/4 + 9 a_1/5 + 12 a_2/6 + 15 a_3/7) \quad (A.30)$$

$$k_{15} = s^5 (f_1'/s + 4 f_1''/5 + 12 a_1/6 + 16 a_2/7 + 20 a_3/8) \quad (A.31)$$

$$k_{16} = s^6 (f_1'/s + 5 f_1''/6 + 15 a_1/7 + 20 a_2/8 + 25 a_3/9) \quad (A.32)$$

$$k_{26} = s^7 (5 f_1' / 6s + 5 f_1'' / 7 + 15 a_1 / 8 + 20 a_2 / 9 + 25 a_3 / 10) \quad (A.33)$$

$$k_{36} = s^8 (5 f_1' / 7s + 5 f_1'' / 8 + 15 a_1 / 9 + 20 a_2 / 10 + 25 a_3 / 11) \quad (A.34)$$

$$k_{46} = s^9 (5 f_1' / 8s + 5 f_1'' / 9 + 15 a_1 / 10 + 20 a_2 / 11 + 25 a_3 / 12) \quad (A.35)$$

$$k_{56} = s^{10} (5 f_1' / 9s + 5 f_1'' / 10 + 15 a_1 / 11 + 20 a_2 / 12 + 25 a_3 / 13) \quad (A.36)$$

$$k_{66} = s^{11} (5 f_1' / 10s + 5 f_1'' / 11 + 15 a_1 / 12 + 20 a_2 / 13 + 25 a_3 / 14) \quad (A.37)$$

and,

$$a_1 = -10 f_1 / s^2 - 6 f_1' / s - 3 f_1'' / 2 + 10 f_2 / s^2 - 4 f_2' / s + f_2'' / 2 \quad (A.38)$$

$$a_2 = 15 f_1 / s^2 + 8 f_1' / s + 3 f_1'' / 2 - 15 f_2 / s^2 + 7 f_2' / s - f_2'' \quad (A.39)$$

$$a_3 = -6 f_1 / s^2 - 3 f_1' / s - f_1'' / 2 + 6 f_2 / s^2 - 3 f_2' / s + f_2'' / 2 \quad (A.40)$$

Matrix manipulations have been done by Computer to get
the $\begin{bmatrix} W_s \end{bmatrix}_m$ matrix specified in Equation (A.24).

APPENDIX B

ELEMENTAL MATRICES FOR UNSTEADY FLOWB.1 QUADRATIC TRIANGULAR ELEMENT

To develop the different elemental matrices, area coordinates¹⁷ have been employed because of their usefulness and effectiveness for formulation of shape functions and in surface integration of work functions.

B.1.1 AREA COORDINATES

An alternative to specify the position of a point P inside a triangle (Figure 15) by rectangular cartesian coordinates, is to specify the position relative to the triangle itself. This can be done using the three areas A_1 , A_2 , A_3 :

$$\xi = A_1/A, \quad \eta = A_2/A, \quad \tau = A_3/A \quad (B.1)$$

The ξ , η and τ will be used as coordinates of the point P. It should be noted that:

$$\xi + \eta + \tau = 1 \quad (B.2)$$

The area coordinates are related to the rectangular cartesian coordinates by satisfying the conditions at the nodes:

$$\begin{bmatrix} 1 \\ x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} \begin{bmatrix} \xi \\ \eta \\ \tau \end{bmatrix} \quad (\text{B.3})$$

Inversion of the square matrix in Equation (B.3) leads to the definition of area coordinates in terms of the rectangular cartesian coordinates:

$$\begin{bmatrix} \xi \\ \eta \\ \tau \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \end{bmatrix} \quad (\text{B.4})$$

where:

$$a_q = x_m y_n - x_n y_m \quad q = 1, 2, 3 \quad (\text{B.5})$$

in which q, m, n denote the cyclic permutation of 1, 2, 3 and

$$b_q = y_m - y_n \quad q = 1, 2, 3 \quad (\text{B.6})$$

$$c_q = x_n - x_m \quad q = 1, 2, 3 \quad (\text{B.7})$$

To get a differentiation we take the derivatives of

Equation (B.4):

$$\begin{bmatrix} \frac{\partial \xi}{\partial x} \\ \frac{\partial \eta}{\partial x} \\ \frac{\partial \tau}{\partial x} \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (\text{B.8})$$

and

$$\begin{bmatrix} \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial y} \\ \frac{\partial \tau}{\partial y} \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad (\text{B.9})$$

To get an integration, a general form¹⁷ is given by:

$$\iint_A \xi^q \eta^m \tau^n dA = 2A \frac{q! m! n!}{(q + m + n + 2)!} = \frac{2A}{p} \quad (\text{B.10})$$

The foregoing integral is tabulated by the author in Table 18, for different values of P_t , where P_t is given by:

$$P_t = q + m + n \quad (\text{B.11})$$

B.1.2 RESIDUAL EVALUATION

Using the differentiation formulas in Equation (B.8) and (B.9), and carrying out the integrations as in Equation (B.10), and using the Table 18, then the continuity equation (Equation (6.12)) can be written for the element n in the form as;

$$\left[D_b \right]_m \left\{ u^e \right\}_m + \left[D_c \right]_m \left\{ v^e \right\}_m = 0 \quad (B.12)$$

The matrix $\left[D_b \right]_m$ is given in Table 19. The matrix

$\left[D_c \right]_m$ is similar to $\left[D_b \right]_m$ with c_1, c_2, c_3

replacing b_1, b_2 and b_3 respectively.

The momentum equation, Equation (6.13), can be written for the element m in the form as;

$$\begin{aligned} & \left[W_t \right]_m \cdot \left\{ u^e - \bar{u}^e \right\}_m + \frac{\Delta t}{2} \left(\left[K_u \right]_m \cdot \left\{ u^e \right\}_m \right. \\ & \quad \left. + \left[\bar{K}_u \right]_m \cdot \left\{ \bar{u}^e \right\}_m + \left[K_v \right]_m \cdot \left\{ u^e \right\}_m + \left[\bar{K}_v \right]_m \cdot \left\{ \bar{u}^e \right\}_m \right. \\ & \quad \left. - \left[G_u \right]_m \cdot \left\{ u^e + \bar{u}^e \right\}_m \right) \end{aligned} \quad (B.13)$$

Where $\left[W_t \right]_m$ is given by;

$$[W_t]_m = \frac{2A_m}{360} \begin{bmatrix} 6 & -1 & -1 & 0 & -4 & 0 \\ -1 & 6 & -1 & 0 & 0 & -4 \\ -1 & -1 & 6 & -4 & 0 & 0 \\ 0 & 0 & -4 & 32 & 16 & 16 \\ -4 & 0 & 0 & 16 & 32 & 16 \\ 0 & -4 & 0 & 16 & 16 & 32 \end{bmatrix} \quad (B.14)$$

and A_m is the area of the triangle m . The elements of the matrix $[K_u]_m$ are given as follows;

$$K_u(1, 1) = b_1 (13 u_1/120 - (u_2 + u_3)/280 + 2(u_4 + u_6)/105 + u_5/210) \quad (B.15.1)$$

$$K_u(1, 2) = b_2 (-(u_2 + 2u_1)/280 + (11 u_3 - 20 u_5)/2520 - 2(u_6 + 2 u_4)/315) \quad (B.15.2)$$

$$K_u(1, 3) = b_3 (-(u_3 + 2 u_1)/280 + (11 u_2 - 20 u_5)/2520 - 2(u_4 + 2 u_6)/315) \quad (B.15.3)$$

$$K_u(1, 4) = b_2 (11 u_1/35 - 2(u_2 + u_3)/315 + 2 u_4/105 - 2 u_5/315 + 2 u_6/105) + b_1(u_1/105 - 2 u_2/315 + u_3/630 - 4 u_4/315 - 2 u_5/105 - 2 u_6/315) \quad (B.15.4)$$

$$\begin{aligned}
 K_u(1, 5) &= b_3 (6(u_1 - 2u_5) - 4(u_2 + u_6 + 2u_4) \\
 &\quad + u_3)/630 \\
 &\quad + b_2 (6(u_1 - 2u_5) - 4(u_3 + u_4 + 2u_6) \\
 &\quad + u_2)/630
 \end{aligned} \tag{B.15.5}$$

$$\begin{aligned}
 K_u(1, 6) &= b_3 (99u_1 + 6(u_4 + u_6) - 2(u_2 + u_3 + u_5))/315 \\
 &\quad + b_1 (6(u_1 - 2u_5) - 4(u_3 + u_4 + 2u_6) \\
 &\quad + u_2)/630
 \end{aligned} \tag{B.15.6}$$

$$\begin{aligned}
 K_u(2, 1) &= b_1 (-(u_1 + 2u_2)/280 + (11u_3 - 20u_6)/2520 \\
 &\quad - 2(u_5 + 2u_4)/315)
 \end{aligned} \tag{B.15.7}$$

$$\begin{aligned}
 K_u(2, 2) &= b_2 (-(u_1 + u_3)/280 + 13u_2/120 \\
 &\quad + 2(u_4 + u_5)/105 + u_6/210)
 \end{aligned} \tag{B.15.8}$$

$$\begin{aligned}
 K_u(2, 3) &= b_3 (-(u_3 + 2u_2)/280 + (11u_1 - 20u_6)/2520 \\
 &\quad - 2(u_4 + 2u_5)/315)
 \end{aligned} \tag{B.15.9}$$

$$\begin{aligned}
 K_u(2, 4) &= b_2 (6(u_2 - 2u_6) - 4(u_1 + u_5 + 2u_4) \\
 &\quad + u_3)/630 \\
 &\quad + b_1 (6(u_4 + u_5) + 99u_2 - 2 \\
 &\quad \cdot (u_1 + u_3 + u_6))/315
 \end{aligned} \tag{B.15.10}$$

$$\begin{aligned}
 K_u (2, 5) = & b_3 (6(u_4 + u_5) + 99 u_2 - 2 (u_1 + u_3 + u_6))/315 \\
 & + b_2 (6(u_2 - 2 u_6) - 4 (u_3 + u_4 + 2 u_5) \\
 & + u_1)/630.
 \end{aligned}
 \tag{B.15.11}$$

$$\begin{aligned}
 K_u (2, 6) = & b_3 (6 (u_2 - 2 u_6) - 4 (u_1 + u_5 + 2 u_4) \\
 & + u_3)/630 \\
 & + b_1 (6 (u_2 - 2 u_6) - 4 (u_3 + u_4 + 2 u_5) \\
 & + u_1)/630
 \end{aligned}
 \tag{B.15.12}$$

$$\begin{aligned}
 K_u (3, 1) = & b_1 (-(u_1 + 2 u_3)/280 + (11 u_2 - 20 u_4)/2520 \\
 & - 2 (u_5 + 2 u_6)/315)
 \end{aligned}
 \tag{B.15.13}$$

$$\begin{aligned}
 K_u (3, 2) = & b_2 (-(u_2 + 2 u_3)/280 + (11 u_1 - 20 u_4)/2520 \\
 & - 2 (u_6 + 2 u_5)/315)
 \end{aligned}
 \tag{B.15.14}$$

$$\begin{aligned}
 K_u (3, 3) = & b_3 (13 u_3/120 - (u_1 + u_2)/280 \\
 & + 2 (u_5 + u_6)/105 + u_4/210)
 \end{aligned}
 \tag{B.15.15}$$

$$\begin{aligned}
 K_u (3, 4) = & b_2 (6(u_3 - 2 u_4) - 4 (u_1 + u_5 + 2 u_6) \\
 & + u_2)/630 \\
 & b_1 (6(u_3 - 2 u_4) - 4 (u_6 + u_2 + 2 u_5) \\
 & + u_1)/630
 \end{aligned}
 \tag{B.15.16}$$

$$\begin{aligned}
 K_u(3, 5) = & b_3 (6(u_3 - 2u_4) - 4(u_2 + u_6 + u_5) \\
 & + u_1)/630 \\
 & b_2 (99u_3 - 2(u_1 + u_2 + u_4) \\
 & + 6(u_5 + u_6))/315
 \end{aligned} \tag{B.15.17}$$

$$\begin{aligned}
 K_u(3, 6) = & b_3 (6(u_3 - 2u_4) - 4(u_1 + u_5 + 2u_6) \\
 & + u_2)/630 \\
 & + b_1 (99u_3 + 6(u_5 + u_6) - 2 \cdot \\
 & \cdot (u_1 + u_2 + u_4))/315
 \end{aligned} \tag{B.15.18}$$

$$\begin{aligned}
 K_u(4, 1) = & b_1 (20(u_6 + 2u_4) + 12u_1 - 5u_3 \\
 & + 4(u_5 - 2u_2))/630
 \end{aligned} \tag{B.15.19}$$

$$\begin{aligned}
 K_u(4, 2) = & b_2 (20(u_5 + u_4) + 12u_2 - 5u_3 \\
 & + 4(u_6 - 2u_1))/630
 \end{aligned} \tag{B.15.20}$$

$$\begin{aligned}
 K_u(4, 3) = & b_3 (-24u_4 + 3u_3 + 4(u_5 + u_6 - u_1 \\
 & - u_2))/630
 \end{aligned} \tag{B.15.21}$$

$$\begin{aligned}
 K_u(4, 4) = & b_2 (3(2u_1 - 2u_3 + 16u_4 + 8u_6) \\
 & + 4(4u_5 - u_2))/315 \\
 & + b_1 (3(2u_2 - 2u_3 + 16u_4 + 8u_5) + 4 \cdot \\
 & \cdot (4u_6 - u_1))/315
 \end{aligned} \tag{B.15.22}$$

$$\begin{aligned}
 K_u(4, 5) = & b_3 (3(2 u_2 - 2 u_3 + 16 u_4 + 8 u_5) \\
 & + 4 (4 u_6 - u_1))/315 \\
 & + b_2 (16(u_4 + u_5 + u_6) - 2(u_1 + u_2 + u_3))/315 \quad (B.15.23)
 \end{aligned}$$

$$\begin{aligned}
 K_u(4, 6) = & b_3 (3 (2 u_1 - 2 u_3 + 16 u_4 + 8 u_6) \\
 & + 4 (4 u_5 - u_2))/315 \\
 & + b_1 (16(u_4 + u_5 + u_6) - 2 \cdot \\
 & \cdot (u_1 + u_2 + u_3))/315 \quad (B.15.24)
 \end{aligned}$$

$$\begin{aligned}
 K_u(5, 1) = & b_1 (-24 u_5 + 3 u_1 + 4 \cdot \\
 & \cdot (u_6 + u_4 - u_2 - u_3))/630 \quad (B.15.25)
 \end{aligned}$$

$$\begin{aligned}
 K_u(5, 2) = & b_2 (20 (u_4 + 2 u_5) + 12 u_2 - 5 u_1 + 4 \cdot \\
 & \cdot (u_6 - 2 u_3))/630 \quad (B.15.26)
 \end{aligned}$$

$$\begin{aligned}
 K_u(5, 3) = & b_3 (20 (u_6 + 2 u_5) + 12 u_3 - 5 u_1 \\
 & + 4 (u_4 - 2 u_2))/630 \quad (B.15.27)
 \end{aligned}$$

$$\begin{aligned}
 K_u(5, 4) = & b_2 (16 (u_4 + u_5 + u_6) - 2 \cdot \\
 & \cdot (u_1 + u_2 + u_3))/315 \\
 & + b_1 (3 (-2 u_1 + 2 u_2 + 8 u_4 + 16 u_5) \\
 & + 4 (4 u_6 - u_3))/315 \quad (B.15.28)
 \end{aligned}$$

$$\begin{aligned}
 K_u(5, 5) = & b_3 (3 (-2 u_1 + 2 u_2 + 8 u_4 + 16 u_5) \\
 & + 4 (4 u_6 - u_3))/315 \\
 & + b_2 (3 (-2 u_1 + 2 u_3 + 16 u_5 + 8 u_6) \\
 & + 4 (4 u_4 - u_2))/315 \quad (B.15.29)
 \end{aligned}$$

$$\begin{aligned}
 K_u(5, 6) = & b_3 (16 (u_5 + u_6 + u_4) - 2 (u_1 + u_2 + u_3))/315 \\
 & + b_1 (3 (-2 u_1 + 2 u_3 + 16 u_5 + 8 u_6) \\
 & + 4 (4 u_4 - u_2))/315
 \end{aligned} \tag{B.15.30}$$

$$\begin{aligned}
 K_u(6, 1) = & b_1 (20 (u_4 + 2 u_6) + 12 u_1 - 5 u_2 \\
 & + 4 (u_5 - 2 u_3))/630
 \end{aligned} \tag{B.15.31}$$

$$\begin{aligned}
 K_u(6, 2) = & b_2 (-24 u_6 + 3 u_2 + 4 (u_4 + u_5 - u_3 \\
 & - u_1))/630
 \end{aligned} \tag{B.15.32}$$

$$\begin{aligned}
 K_u(6, 3) = & b_3 (20 (u_5 + 2 u_6) + 12 u_3 - 5 u_2 + 4 \\
 & (u_4 - 2 u_1))/630
 \end{aligned} \tag{B.15.33}$$

$$\begin{aligned}
 K_u(6, 4) = & b_2 (3 (2 u_1 - 2 u_2 + 8 u_4 + 16 u_6) \\
 & + 4 (4 u_5 - u_3))/315 \\
 & + b_1 (-2 (u_1 + u_2 + u_3) + 16 \cdot \\
 & (u_4 + u_5 + u_6))/315
 \end{aligned} \tag{B.15.34}$$

$$\begin{aligned}
 K_u(6, 5) = & b_3 (-2 (u_1 + u_2 + u_3) + 16 \cdot \\
 & (u_4 + u_5 + u_6))/315 + \\
 & b_2 (3 (-2 u_2 + 2 u_3 + 8 u_5 + 16 u_6) \\
 & + 4 (4 u_4 - u_1))/315
 \end{aligned} \tag{B.15.35}$$

$$\begin{aligned}
K_u(6, 6) = & b_3 (3(2u_1 - 2u_2 + 8u_4 + 16u_6) \\
& + 4(4u_5 - u_3))/315 + b_1 (3(-2u_2 + \\
& 2u_3 + 8u_5 + 16u_6) + 4(4u_4 - u_1)) \\
& / 315
\end{aligned}
\tag{B.15.36}$$

$[\bar{K}_u]_m$ in Equation (B.13) can be obtained from Equation (B.15) by putting $\bar{u}_i$ instead of u_i ($i = 1, 2, \dots, 6$), and the bar over the u_i denotes values at previous time step.

$[K_v]_m$ in Equation (B.13) can be obtained from Equation (B.15) by replacing all the u_i 's and b_i 's by v_i 's and c_i 's respectively. Similary $[\bar{K}_v]_m$ is obtained by using $\bar{v}_i$ instead of v_i .

The matrix $[G_u]_m$ is given by;

$$[G_u]_m = \frac{1}{3A} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ c_1^2 & c_2^2 & c_3^2 & 2c_1c_2 & 2c_2c_3 & 2c_3c_1 \\ c_1^2 & c_2^2 & c_3^2 & 2c_1c_2 & 2c_2c_3 & 2c_3c_1 \\ c_1^2 & c_2^2 & c_3^2 & 2c_1c_2 & 2c_2c_3 & 2c_3c_1 \end{bmatrix} \tag{B.16}$$

B.2 QUINTIC TRIANGULAR ELEMENT

The matrix $[RI]$ in equation (7.50) is given by

$$[RI] = \iint_A f^T f_y dA \quad (B.17)$$

Thus

$$RI(i,j) = \iint_A (f_i) \cdot (f_{yj}) dA, \quad i,j=1, 2, \dots, 21 \quad (B.18)$$

The values for $f(i)$ are given in Equations (7.22,23), and the differentiation with respect to y is as follows;

$$\frac{\partial}{\partial y} = \frac{1}{2A} (c_1 \frac{\partial}{\partial \xi} + c_2 \frac{\partial}{\partial \eta} + c_3 \frac{\partial}{\partial \tau}) \quad (B.19)$$

where c_1 , c_2 , and c_3 are obtained from Equation (B.9), thus the following equalities exist;

$$f_{y1} = \frac{1}{2A} (30 c_1 (\xi^2 - \xi)^2) \quad (B.20.1)$$

$$f_{y2} = \frac{1}{2A} (c_2 (4 \xi^3 - 3 \xi^4) + 12 c_1 \eta (\xi^2 - \xi^3)) \quad (B.20.2)$$

$$f_{y3} = \frac{1}{2A} (c_3 \xi^3 (4 - 3\xi) + 12 \xi^2 c_1 \tau (1 - \xi)) \quad (B.20.3)$$

$$f_{y4} = \frac{1}{2A} (3 c_1 \xi^2 \eta^2 + 2 c_2 \eta \xi^3) \quad (B.20.4)$$

$$f_{y5} = \frac{1}{2A} (c_3 \eta \xi^3 + c_2 \tau \xi^3 + 3 c_1 \xi^2 \eta \tau) \quad (B.20.5)$$

$$f_{y6} = \frac{1}{2A} (2 c_3 \tau \xi^3 + 3 c_1 \xi^2 \tau^2) \quad (B.20.6)$$

The values for f_{y7} to f_{y12} , and f_{y13} to f_{y18} are obtained from Equations (B.20) by a cyclic permutation of c_1 , c_2 , c_3 , ξ , η , and τ .

The values for f_{y19} , f_{y20} and f_{y21} are given as follows;

$$f_{y19} = \frac{1}{2A} (c_1 \eta^2 \tau^2 + 2 c_2 \xi \eta \tau^2 + 2 c_3 \xi \tau \eta^2) \quad (B.20.7)$$

$$f_{y20} = \frac{1}{2A} (2 c_1 \xi \eta \tau^2 + c_2 \xi^2 \tau^2 + 2 c_3 \eta \tau \xi^2) \quad (B.20.8)$$

$$f_{y21} = \frac{1}{2A} (2 c_2 \eta \tau \xi^2 + c_3 \eta^2 \xi^2 + 2 c_1 \tau \xi \eta^2) \quad (B.20.9)$$

Substituting Equations (7.22), (7.23), and (B.20) into Equation

(B.18) we get the matrix $[RI]$ in the form

$$[RI] = \begin{bmatrix} \boxed{AA} & \boxed{BB} & \boxed{CC} & L & M & N \\ \boxed{CC^{\sim}} & \boxed{AA^{\sim}} & \boxed{BB^{\sim}} & N^{\sim} & L^{\sim} & M^{\sim} \\ \boxed{BB^{\sim\sim}} & \boxed{CC^{\sim\sim}} & \boxed{AA^{\sim\sim}} & M^{\sim\sim} & N^{\sim\sim} & L^{\sim\sim} \\ \begin{matrix} O \\ P \\ Q \end{matrix} \begin{matrix} \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \end{matrix} & \begin{matrix} Q^{\sim} \\ O^{\sim} \\ P^{\sim} \end{matrix} \begin{matrix} \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \end{matrix} & \begin{matrix} P^{\sim\sim} \\ Q^{\sim\sim} \\ O^{\sim\sim} \end{matrix} \begin{matrix} \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \end{matrix} & \begin{matrix} e \\ h \\ g \end{matrix} \begin{matrix} \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \end{matrix} & \begin{matrix} g \\ e \\ h \end{matrix} \begin{matrix} \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \end{matrix} & \begin{matrix} h \\ g \\ e \end{matrix} \begin{matrix} \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \\ \underline{\hspace{1cm}} \end{matrix} \end{bmatrix} \quad (B.21)$$

where AA, BB, and CC are 6x6 matrices; L, M and N are 6x1 column vectors; O, P and Q are 1x6 row vectors; and e, g, and h are single terms. In the matrix (B.21) any term in $AA^{\sim}$ is obtained by a cyclic permutation of c's on a term of similar location in matrix AA, also $AA^{\sim\sim}$ can be obtained from $AA^{\sim}$ similarly. The same procedure is repeated for $BB^{\sim}$, $CC^{\sim}$, $L^{\sim}$, $M^{\sim}$, $N^{\sim}$, $O^{\sim}$, $P^{\sim}$, $Q^{\sim}$, $e^{\sim}$, $g^{\sim}$ and $h^{\sim}$, also $BB^{\sim\sim}$, $CC^{\sim\sim}$, $L^{\sim\sim}$, $M^{\sim\sim}$, $N^{\sim\sim}$, $O^{\sim\sim}$, $P^{\sim\sim}$, $Q^{\sim\sim}$, $e^{\sim\sim}$, $g^{\sim\sim}$, and $h^{\sim\sim}$. By doing this the third of the matrix $[RI]$ is evaluated exactly and the other two thirds are obtained by a cyclic permutation of the c's. The results for the first row in the matrix $[RI]$ are given below as an example, the exact integration has been performed for the ninth degree polynomial in Equation (B.18) and using the

developed formula in Table 18. The rest of the matrix is well illustrated in the computer program (Appendix C).

$$\begin{aligned}
 RI(1,1) &= .1958874459 c_1 & (B.22.1) \\
 RI(1,2) &= (.039177489 c_1 + .067316017 c_2) & (B.22.2) \\
 RI(1,3) &= (.067316017 c_3 + .039177489 c_1) & (B.22.3) \\
 RI(1,4) &= (.006529582 c_1 + .010137085 c_2) & (B.22.4) \\
 RI(1,5) &= (.0050685425 (c_2 + c_3) + .0032647907 c_1) & (B.22.5) \\
 RI(1,6) &= (.006529582 c_1 + .010137085 c_3) & (B.22.6) \\
 RI(1,7) &= .1082251082 c_2 & (B.22.7) \\
 RI(1,8) &= .0106060606 (c_2 + c_3) & (B.22.8) \\
 RI(1,9) &= (.0106060606 c_1 + .0326839826 c_2) & (B.22.9) \\
 RI(1,10) &= .0010101010 (c_2 + c_3) & (B.22.10) \\
 RI(1,11) &= (.0005050505 c_1 + .0016414141 (c_3 + c_2)) & (B.22.11) \\
 RI(1,12) &= (.0032828282 c_1 + .006529582 c_2) & (B.22.12) \\
 RI(1,13) &= .1082251082 c_3 & (B.22.13) \\
 RI(1,14) &= (.0106060606 c_1 + .0326839826 c_3) & (B.22.14) \\
 RI(1,15) &= .0106060606 (c_3 + c_2) & (B.22.15) \\
 RI(1,16) &= (.0032828282 c_1 + .006529582 c_3) & (B.22.16) \\
 RI(1,17) &= (.0005050505 c_1 + .0016414141 (c_2 + c_3)) & (B.22.17) \\
 RI(1,18) &= .0010101010 (c_2 + c_3) & (B.22.18) \\
 RI(1,19) &= (.0003367003 c_1 + .001094276 (c_2 + c_3)) & (B.22.19) \\
 RI(1,20) &= (.001094276 c_1 + .0021765273 (c_2 + c_3)) & (B.22.20) \\
 RI(1,21) &= (.001094276 c_1 + .0020765273 (c_2 + c_3)) & (B.22.21)
 \end{aligned}$$

To evaluate the matrices $[RI]$ and $[RJO]$ in Equation (7.50), we need to know the second derivative f_{yx} which is obtained using the

relations given by Equations (7.22,23,35) as follows;

$$f_{yx1} = 60 c_1 b_1 \xi (2 \xi^2 - 3 \xi + 1)/4 A^2 \quad (B.23.1)$$

$$f_{yx2} = (12 \xi^2 (1 - \xi) (b_2 c_1 + b_1 c_2) + 12 b_1 c_1 \eta \xi (2 - 3 \xi))/4 A^2 \quad (B.23.2)$$

$$f_{yx3} = (12 \xi^2 (1 - \xi) (b_3 c_1 + b_1 c_3) + 12 b_1 c_1 \tau \xi (2 - 3 \xi))/4 A^2 \quad (B.23.3)$$

$$f_{yx4} = (6 b_1 c_2 \eta \xi^2 + 6 b_1 c_1 \xi \eta^2 + 2 b_2 c_2 \xi^3 + 6 b_2 c_1 \eta \xi^2)/4 A^2 \quad (B.23.4)$$

$$f_{yx5} = (3 \eta \xi^2 (c_1 b_3 + c_3 b_1) + \xi^3 (b_3 c_2 + b_2 c_3) + 3 \tau \xi^2 (b_2 c_1 + b_1 c_2) + 6 b_1 c_1 \xi \eta \tau)/4 A^2 \quad (B.23.5)$$

$$f_{yx6} = (6 b_1 c_3 \tau \xi^2 + 2 c_3 b_3 \xi^3 + 6 c_1 b_3 \tau \xi^2 + 6 c_1 b_1 \xi \tau^2)/4 A^2 \quad (B.23.6)$$

The values for f_{yx7} to f_{yx12} , and f_{yx13} to f_{yx18} are obtained from Equations (B.23) by a cyclic permutation of $c_1, c_2, c_3; b_1, b_2, b_3; \xi, \eta$, and τ . The values for f_{yx19}, f_{yx20} and f_{yx21} are given as follows;

$$f_{yx19} = (2 \tau \eta^2 (c_1 b_3 + c_3 b_1) + 2 \eta \tau^2 (c_1 b_2 + c_2 b_1) + 2 c_2 b_2 \xi \tau^2 + 4 \xi \eta \tau (c_2 b_3 + c_3 b_2) + 2 c_3 b_3 \xi \eta^2)/4 A^2 \quad (B.23.7)$$

$$f_{yx20} = (2 \xi \tau^2 (c_2 b_1 + c_1 b_2) + 2 \tau \xi^2 (c_2 b_3 + c_3 b_2) + 2 c_3 b_3 \eta \xi^2 + 4 \xi \eta \tau (c_3 b_1 + c_1 b_3) + 2 c_1 b_1 \eta \tau^2)/4 A^2 \quad (B.23.8)$$

$$\begin{aligned}
f_{yx21} = & (2 \eta \xi^2 (c_3 b_2 + c_2 b_3) + 2 \xi \eta^2 (c_3 b_1 + c_1 b_3) \\
& + 2 c_1 b_1 \tau \eta^2 + 4 \xi \eta \tau (c_1 b_2 + c_2 b_1) \\
& + 2 c_2 b_2 \tau \xi^2) / 4 A^2
\end{aligned} \tag{B.23.9}$$

From Equation (7.50), we can write the element of $[RJ]$ as follows

$$RJ(i,k) = \iint_A f_i \left(\sum_{j=1}^{21} a_j f_{yj} \right) f_{yxk} dA, \quad i,k=1,2,\dots,21 \tag{B.24}$$

where $a(j)$'s are the unknown parameters, thus the column matrix $[RJ] \{a\}$ in Equation (7.50) is nonlinear in $\{a\}$. The amount of effort required in obtaining the integration in Equation (B.24) analytically is truly formidable. Many references¹³⁰⁻¹³² in the literature exist for numerical integration using weighting values for several sample points within the element. A 25 points formulae¹³¹ has been used which is based on Gauss Radau method. To check the accuracy of the numerical integration method, Equation (B.24) has been solved exactly for $RJ(1,1)$, which includes 21 integrations composed of complicated function. The exact integration has been performed for an element having one of the sides at $x = 1$, the relations there are as follows;

$$c_3 = -c_1 \tag{B.25}$$

$$c_2 = 0 \tag{B.26}$$

The error associated with the numerical integration has been found to be .09% which is adequate for the accuracy required. To obtain the exact solution, we substitute Equations (7.22,23) and Equations (B.20,23,25,26) in Equation (B.24). The results are as follows;

$$\begin{aligned}
 RJ(1,1) = & \frac{1}{4A^2} c_1^2 b_1 (-.2047952048 a_1 - .0409590408 a_2 \\
 & + .1931401932 a_3 - .0068265068 a_4 \\
 & + .0119047619 a_5 + .0238095238 a_6 \\
 & - .0198135175 (a_8 - a_9) - .003080253 a_{10} \\
 & - .0003330048 a_{11} + .003746244 a_{12} \\
 & + .0396269673 a_{14} - .0198135743 a_{15} \\
 & + .010572744 a_{16} - .0003330048 a_{17} \\
 & - .0003080253 a_{18} - .0002219940 a_{19} \\
 & + .003524244 (a_{20} + a_{21})) \quad (B.27)
 \end{aligned}$$

The numerical integration procedure is well illustrated in the computer program in Appendix C.

The matrix $[RJ0]$ in Equation (7.50) is similar to $[RJ]$, but with $\bar{a}_j$'s replacing a_j 's in Equation (B.24). From Equation (7.50), the elements of the matrix $[RK]$ can be written as follows;

$$RK(i,k) = \iint_A f_i \left(\sum_{j=1}^{21} a_j f_{xj} \right) f_{yyk} dA, \quad i,k=1,2, \dots, 21 \quad (B.28)$$

Using Equations (7.22,23,31), the derivatives f_{xj} are written as follows;

$$f_{x1} = (30 b_1 (\xi^2 - \xi^2)^2)/2 A \quad (B.29.1)$$

$$f_{x2} = (b_2 \xi^3 (4 - 3 \xi) + 12 \xi^2 b_1 \eta (1 - \xi))/2 A \quad (B.29.2)$$

$$f_{x3} = (b_3 \xi^3 (4 - 3 \xi) + 12 b_1 \xi^2 \tau (1 - \xi))/2 A \quad (B.29.3)$$

$$f_{x4} = (3 b_1 \xi^2 \eta^2 + 2 b_2 \eta \xi^3)/2 A \quad (B.29.4)$$

$$f_{x5} = (b_3 \eta \xi^3 + b_2 \tau \xi^3 + 3 b_1 \xi^2 \eta \tau)/2 A \quad (B.29.5)$$

$$f_{x6} = (2 b_3 \tau \xi^3 + 3 b_1 \xi^2 \tau^2) / 2 A \quad (B.29.6)$$

The values for f_{x7} to f_{x12} , and f_{x13} to f_{x18} are obtained from Equations (B.29) by a cyclic permutation of $b_1, b_2, b_3; \xi, \eta$, and τ . The values for f_{x19}, f_{x20} , and f_{x21} are given as follows;

$$f_{x19} = (b_1 \eta^2 \tau^2 + 2 b_2 \xi \eta \tau^2 + 2 b_3 \xi \tau \eta^2) / 2 A \quad (B.29.7)$$

$$f_{x20} = (2 b_1 \xi \eta \tau^2 + b_2 \xi^2 \tau^2 + 2 b_3 \eta \tau \xi^2) / 2 A \quad (B.29.8)$$

$$f_{x21} = (2 b_2 \eta \tau \xi^2 + b_3 \eta^2 \xi^2 + 2 b_1 \tau \xi \eta^2) / 2 A \quad (B.29.9)$$

Using Equations (7.22,23,34) the derivatives f_{yyk} in Equation (B.28) can be written as follows;

$$f_{yy1} = (60 c_1^2 \xi (2 \xi^2 - 3 \xi + 1)) / 4 A^2 \quad (B.30.1)$$

$$f_{yy2} = (24 c_1 c_2 \xi^2 (1 - \xi) + 12 c_1^2 \eta \xi (2 - 3 \xi)) / 4 A^2 \quad (B.30.2)$$

$$f_{yy3} = (24 c_1 c_3 \xi^2 (1 - \xi) + 12 c_1^2 \tau \xi (2 - 3 \xi)) / 4 A^2 \quad (B.30.3)$$

$$f_{yy4} = (12 c_1 c_2 \eta \xi^2 + 6 c_1^2 \xi \eta^2 + 2 c_2^2 \xi^2) / 4 A^2 \quad (B.30.4)$$

$$f_{yy5} = (6 c_1 c_3 \xi^2 \eta + 2 c_2 c_3 \xi^3 + 6 c_1 c_2 \tau \xi^2 + 6 c_1^2 \eta \tau \xi) / 4 A^2 \quad (B.30.5)$$

$$f_{yy6} = (12 c_1 c_3 \xi^2 \tau + 6 c_1^2 \xi \tau^2 + 2 c_3^2 \xi^3) / 4 A^2 \quad (B.30.6)$$

The values for f_{yy7} to f_{yy12} , and f_{yy13} to f_{yy18} are obtained from Equations (B.30) by a cyclic permutation of $c_1, c_2, c_3; \xi, \eta$, and τ . The values for f_{yy19}, f_{yy20} , and f_{yy21} are given as follows;

$$f_{yy19} = (4 c_1 c_3 \tau \eta^2 + 4 c_1 c_2 \eta \tau^2 + 2 c_2^2 \xi \tau^2 + 8 c_2 c_3 \xi \eta \tau + 2 c_3^2 \xi \eta^2) / 4 A^2 \quad (B.30.7)$$

$$\begin{aligned} f_{yy20} = & (2 c_1^2 \eta \tau^2 + 4 c_1 c_2 \xi \tau^2 + 8 c_1 c_3 \xi \eta \tau \\ & + 4 c_2 c_3 \tau \xi^2 + 2 c_3^2 \eta \xi^2) / 4 A^2 \end{aligned} \quad (B.30.8)$$

$$\begin{aligned} f_{yy21} = & (2 c_2^2 \tau \xi^2 + 4 c_2 c_3 \eta \xi^2 + 8 c_2 c_1 \xi \eta \tau \\ & + 4 c_3 c_1 \xi \eta^2 + 2 c_1^2 \tau \eta^2) / 4 A^2 \end{aligned} \quad (B.30.9)$$

Using Equations (7.22,23), Equations (B.25,29,30) and substitute in Equation (B.28) to obtain the matrix $[RK]$. The exact results for the first element are given below. The numerical solution shows a .11% deviation from the exact solution, which is acceptable;

$$\begin{aligned} RK(1,1) = & \frac{1}{4A^2} c_1^2 \left[\begin{aligned} & -.2047952048 b_1 a_1 \\ & - (.040959 b_1 + .2340992341 b_2) a_2 \\ & - (.040959 b_1 + .2340992341 b_3) a_3 \\ & - (.0068265 b_1 + .0306360306 b_2) a_4 \\ & - (.015318015 (b_2 + b_3) + .00341325 b_1) a_5 \\ & - (.0068265 b_1 + .0306360306 b_3) a_6 \\ & + (.019813506 (b_2 + b_3)) a_8 \\ & + (.019813506 (b_1 - b_2)) a_9 \\ & + (.00308025 (b_2 + b_3)) a_{10} \\ & + (.001540122 b_1 + .001873122 (b_2 + b_3)) a_{11} \\ & + (.003746244 b_1 - .0068265 b_2) a_{12} \\ & + (.019813506 (b_1 - b_3)) a_{14} \\ & + (.019813506 (b_2 + b_3)) a_{15} \\ & + (.003746244 b_1 - .0068265 b_3) a_{16} \\ & + (.001540122 b_1 + .001873122 (b_2 + b_3)) a_{17} \\ & + (.00308025 (b_2 + b_3)) a_{18} \end{aligned} \right] \end{aligned}$$

$$\begin{aligned}
& + (.00102675 b_1 + .001248748 (b_2 + b_3)) a_{19} \\
& + (.001248748 b_1 - .0022755 (b_2 + b_3))(a_{20} + a_{21}) \quad (B.31)
\end{aligned}$$

Equation (B.31) can be simplified further by using the relation

$$b_1 + b_2 + b_3 = 0 \quad (B.32)$$

The matrix $[RKO]$ in Equation (7.50) is similar to $[RK]$, but with $\bar{a}_j$'s replacing a_j 's in Equation (B.28).

From Equation (7.50), the elements of the matrix $[RL]$ can be written as follows;

$$RL(i,j) = \iint_A (f_i) (f_{yyyj}) dA, \quad i,j=1, 2, \dots, 21 \quad (B.33)$$

Using Equations (7.22,23,34) and Equation (B.19), the derivatives f_{yyyj} are written as follows;

$$f_{yyy1} = (60 c_1^3 (6\xi^2 - 6\xi + 1))/8 A^3 \quad (B.34.1)$$

$$f_{yyy2} = (36 c_1^2 c_2 (2\xi - 3\xi^2) + 24 c_1^3 \eta (1 - 3\xi))/8 A^3 \quad (B.34.2)$$

$$f_{yyy3} = (36 c_1^2 c_3 (2\xi - 3\xi^2) + 24 c_1^3 \tau (1 - 3\xi))/8 A^3 \quad (B.34.3)$$

$$f_{yyy4} = (36 c_1^2 c_2 \xi \eta + 18 c_1^2 c_2^2 \xi^2 + 6 c_1^3 \eta^2)/8 A^3 \quad (B.34.4)$$

$$\begin{aligned}
f_{yyy5} = & (18 c_1 c_2 c_3 \xi^2 + 18 c_1^2 c_3 \xi \eta + 18 c_1^2 c_2 \xi \tau \\
& + 6 c_1^3 \eta \tau)/8 A^3 \quad (B.34.5)
\end{aligned}$$

$$f_{yyy6} = (18 c_1 c_3^2 \xi^2 + 36 c_1^2 c_3 \xi \tau + 6 c_1^3 \tau^2)/8 A^3 \quad (B.34.6)$$

The values f_{yyy7} to f_{yyy12} , and f_{yyy13} to f_{yyy18} are obtained from Equations (B.34) by a cyclic permutation of $c_1, c_2, c_3; \xi, \eta$, and

τ . The values for f_{yyy19}, f_{yyy20} , and f_{yyy21} are given as follows;

$$f_{yyy19} = (6 c_1 c_3^2 \eta^2 + 24 c_1 c_2 c_3 \eta \tau + 6 c_1 c_2^2 \tau^2 + 12 c_2^2 c_3 \xi \tau + 12 c_2 c_3^2 \xi \eta) / 8 A^3 \quad (B.34.7)$$

$$f_{yyy20} = (6 c_1^2 c_2 \tau^2 + 12 c_1^2 c_3 \eta \tau + 24 c_1 c_2 c_3 \xi \tau + 12 c_1 c_3^2 \xi \eta + 6 c_2 c_3^2 \xi^2) / 8 A^3 \quad (B.34.8)$$

$$f_{yyy21} = (6 c_2^2 c_3 \xi^2 + 12 c_2^2 c_1 \tau \xi + 24 c_1 c_2 c_3 \eta \xi + 12 c_2 c_1^2 \eta \tau + 6 c_3 c_1^2 \eta^2) / 8 A^3 \quad (B.34.9)$$

Substituting Equations (7.22), (7.23), and (B.34) into Equation (B.33), we note that the matrix $[RL]$ will have similar construction as the matrix $[RI]$ in the form of Equation (B.21). Thus the third of the matrix $[RL]$ is evaluated exactly and the other two thirds are obtained by a cyclic permutation of the c 's as was done previously in matrix $[RI]$. The results for the first row in the matrix $[RL]$ are given below, and the rest of the matrix is well illustrated in Appendix C;

$$RL(1,1) = (-2.142857143 c_1^3) / 4 A^2 \quad (B.35.1)$$

$$RL(1,2) = (.2142857143 c_1^2 c_2 - .357142857 c_1^3) / 4 A^2 \quad (B.35.2)$$

$$RL(1,3) = (.2142857143 c_1^2 c_3 - .357142857 c_1^3) / 4 A^2 \quad (B.35.3)$$

$$RL(1,4) = (.535714286 c_1^2 c_2 + .964285714 c_1 c_2^2 + .0595238095 c_1^3) / 4 A^2 \quad (B.35.4)$$

$$RL(1,5) = (.0297619047 c_1^3 + .267857143 (c_1^2 c_3 + c_1^2 c_2) + .964285714 c_1 c_2 c_3) / 4 A^2 \quad (B.35.5)$$

$$RL(1,6) = (.964285714 c_1 c_3^2 + .535714286 c_1^2 c_3 + .0595238095 c_1^3) / 4 A^2 \quad (B.35.6)$$

$$RL(1,7) = (1.428571429 c_2^3)/4 A^2 \quad (B.35.7)$$

$$RL(1,8) = (1.071428571 c_2^2 c_3 + .357142857 c_2^3)/4 A^2 \quad (B.35.8)$$

$$RL(1,9) = (1.071428571 c_2^2 c_1 + .9285714286 c_2^3)/4 A^2 \quad (B.35.9)$$

$$RL(1,10) = (.0595238095 (c_2^3 + 3 c_2 c_3^2 + 3 c_2^2 c_3))/4 A^2 \quad (B.35.10)$$

$$RL(1,11) = (.0595238095 (3 c_1 c_2 c_3 + 1.5 c_2^2 c_1) + .0892857143 (c_2^3 + 3 c_2^2 c_3))/4 A^2 \quad (B.35.11)$$

$$RL(1,12) = (.1785714282 c_2 c_1^2 + .535714286 c_2^2 c_1 + .3214285713 c_2^3)/4 A^2 \quad (B.35.12)$$

$$RL(1,13) = (1.428571429 c_3^3)/4 A^2 \quad (B.35.13)$$

$$RL(1,14) = (1.071428571 c_3^2 c_1 + .9285714286 c_3^3)/4 A^2 \quad (B.35.14)$$

$$RL(1,15) = (1.071428571 c_3^2 c_2 + .357142857 c_3^3)/4 A^2 \quad (B.35.15)$$

$$RL(1,16) = (.535714286 c_3^2 c_1 + .1785714282 c_3 c_1^2 + .3214285713 c_3^3)/4 A^2 \quad (B.35.16)$$

$$RL(1,17) = (.0595238095 (3 c_1 c_2 c_3 + 1.5 c_3^2 c_1) + .0892857143 (c_3^3 + 3 c_3^2 c_2))/4 A^2 \quad (B.35.17)$$

$$RL(1,18) = (.0595238095 (c_3^3 + 3 c_3 c_2^2 + 3 c_3^2 c_2))/4 A^2 \quad (B.35.18)$$

$$RL(1,19) = (.0595238095 (c_1 c_3^2 + c_1 c_2^2 + 2 c_1 c_2 c_3 + 3 c_2^2 c_3 + 3 c_2 c_3^2))/4 A^2 \quad (B.35.19)$$

$$RL(1,20) = (.3214285713 c_2 c_3^2 + .0595238095 (c_1^2 c_2 + c_1^2 c_3 + 3 c_1 c_3^2 + 6 c_1 c_2 c_3))/4 A^2 \quad (B.35.20)$$

$$RL(1,21) = (.3214285713 c_2^2 c_3 + .0595238095 (c_3 c_1^2 + c_2 c_1^2 + 3 c_2^2 c_1 + 6 c_1 c_2 c_3))/4 A^2 \quad (B.35.21)$$

APPENDIX C

DETAILS OF COMPUTER PROGRAMSC.1 INTRODUCTION

For the numerical solution of steady and unsteady flows several FORTRAN programs were written. Although the equations required to solve are different for the steady and unsteady flow, the concept used in the programming procedures is the same. It will be noticed later in the other sections that three important subroutines and functions are used through all the programming, these are as follows;

- (i) Subroutine GRIDT, which has the role of defining the grid dimensions, nodes and elements connections, and calculates the area of each element.
- (ii) Subroutine ZSYST, which solve the nonlinear algebraic equations using Brown's method⁹⁶.
- (iii) Function SLVT, which computes the nonlinear algebraic equations required to be solved. The role of this function is to compute the residuals at each node; these residuals are obtained from contributions coming from different elements. The elements that contribute at a node and their properties are defined previously in subroutine GRIDT and transmitted to SLVT by a common statement. For the steady flow problems, the function SLVT defines the nonlinear equations completely, but for the unsteady flow problems, other subroutines are called by SLVT to help computing the nonlinear equations. In

any case, the function SLVT is called by ZSYST and then the iteration procedure starts.

Details of the different subroutines used in the steady and unsteady flow problems are given in sections C.2 and C.3 respectively, also the listing of these subroutines is presented in section C.4.

C.2 FORTTRAN PROGRAMS FOR THE STEADY STATE FLOWS

The steady flow programs are written for the following problems;

- (i) Flat plate - Blasius problem.
- (ii) Flat plate with suction or blowing.
- (iii) Wedges and stagnation flows.

The subroutines used in the preceeding cases are as follows;

case (i) : ZSYST1 and SLVT1

case (ii) : ZSYST1, and SLVT1

case (iii) : ZSYST1, and SLVT2 (which uses SUV1 as helper)

In cases (i) and (ii), the only difference is the boundary conditions applied at the surface. In (i) the surface is impermeable, and in (ii) the surface allows either suction or blowing of the flow, thus the boundary conditions read by the main program are different from case (i). In case (iii) an extra term has to be included in the ordinary differential equation and the computation of the nonlinear algebraic equations are performed in subroutine SLVT2. In all the three cases, the subroutine ZSYST1 is used without change. The

program is well illustrated in the flow chart given in Figure 25, where, the addition of contributions coming from different elements to compute the residual is well demonstrated. A listing of the program is presented in section C.4.1.

C.3 FORTTRAN PROGRAMS FOR THE UNSTEADY STATE FLOWS

The unsteady programs are written for a suddenly accelerated plate in a viscous flow. This is done for both the primitive variables and the stream function formulations. The same concept discussed in the steady state flow cases is followed here.

C.3.1 QUADRATIC ELEMENT AND PRIMITIVE VARIABLES

The subroutines used in this case are as follows;

- (i) GRIDT2 : To calculate the grid properties given the y location of nodes in the main program
- (ii) NODT : To define the elements to nodes connections
- (iii) ZSYT2 : To solve the nonlinear algebraic equations
- (iv) SLVT3 : To compute the nonlinear algebraic equations and add residuals.
- (v) SUVT : Used as helper for SLVT3 to calculate the nonlinear matrices.
- (vi) SOLV : To compute the nodal variables at $x = 1.25$ at $T = T_1$, as discussed in section 6.5 for the time marching scheme.

The flow chart of the program is illustrated in Figure 26, and also a listing of the program is presented in section C.4.2.

C.3.2 QUINTIC ELEMENT AND STREAM FUNCTION FORMULATION

The subroutines used in this case are as follows;

- (i) GRIDT3 : To calculate the grid properties and defines the element-node connections (does the role of (i) and (ii) in section C.3.1.
- (ii) TRANSF : Calculates the transfer matrix B in Equation (7.45) (called A in the program).
- (iii) RIELEM : Calculates the remaining two thirds of the matrices $\begin{bmatrix} RI \end{bmatrix}$, and $\begin{bmatrix} RL \end{bmatrix}$ in Equations (B.18) and (B.33) respectively and they are stored on disc.
- (iv) ZSYST3 : Solves the nonlinear algebraic equations, using Brown's method.
- (v) SLVT4 : Computes the nonlinear algebraic equations and adds contributions from different elements to the residuals at the nodes.
- (vi) INTE : Integrates the terms involved in the matrices $\begin{bmatrix} RJ \end{bmatrix}$ and $\begin{bmatrix} RK \end{bmatrix}$ in Equations (B.24) and (B.28) respectively.
- (vii) FUNC : Evaluates the integrands required in INTE.

The flow chart of the program is illustrated in Figure 27, and a listing is presented in section C.4.3.

C.4. LISTING OF PROGRAMSC.4.1. STEADY FLOW

```

C *****
C *
C *MAIN PROGRAM TO SOLVE THE FLAT, PLATE PROBLEM, SUCTION
C *BLOWING, WEDGE AND STAGNATION STEADY STATE FLOW
C *
C *****

EXTERNAL SLVT1
DOUBLE PRECISION X, EPS, S
1, SLVT1
  DIMENSION X(18), S(6)
  NR=17
  IN1=(NR+4)/3
  IN11=IN1-1
  JJ=0
21 JJ=JJ+1
  IF(JJ.GT.5) GOTO 130
  PRINT 100
  DO 20 I=1, IN11
    READ(5,25) S(I)
    WRITE(6,25) S(I)
20 CONTINUE
  IF(JJ.GT.1) GOTO 31
  DO 30 J=1, NR
    READ(5,25) X(J)
30 CONTINUE
31 READ(5,65) D, DD
  WRITE(6,67) D
  EPS=4.0D-4
  NSIG=5
  ITMAX=10

```

```

WRITE(6,66) (X(K),K=1,NR)
CALL ZSYST1(SLVT1,EPS,NSIG,NR,X,ITMAX,S,D,DD,IER)
WRITE(6,91)ITMAX,(X(I),I=1,NR)
GOTO 21

25 FORMAT(F10.5)
57 FORMAT(5X,E16.6)

65 FORMAT(2F10.5)
66 FORMAT(10X,*INITIAL GUESS TO XOLD(1---N)*,/,(/,4E16.6))
67 FORMAT(/,10X,*WEDGE COEFFICIENT  =*,E16.6)
91, FORMAT(5X,*ITMAX  =*,I3,/,5X,*      X(1)      X(N)*,/,(/,4E16.6)
1,/,10X,*SUCCESSFUL CONVERGENCE*)
100 FORMAT(1H1)
130 STOP
END

FUNCTION SLVT1(X,K,S,D,DD)
C *****
C
C TO EVALUATE THE RESIDUALS FOR FLAT PLATE
C SUCTION AND BLOWING FLOWS
C *****
C
DOUBLE PRECISION SLVT1,S,F,FD,FDD,H26,H36,H46,H56,H66,H16,H14,
1H15,H13,R1,R2,R3,R4,X
DIMENSION X(18),S(6),F(7),FD(7),FDD(7)
NR=17
IN1=(NR+4)/3
IN11=IN1-1
F(1)=0.00
FD(1)=0.00
FD(IN1)=0
FDD(IN1)=00
J=2
DO 5 N=2,IN1
F(N)=X(J)
J=J+3
5 CONTINUE
J=3

```

```

DO 10 N=2,IN11
  FD(N)=X(J)
  J=J+3
10 CONTINUE
  J=1
  DO 15 N=1,IN11
    FDD(N)=X(J)
    J=J+3
  15 CONTINUE
11 FORMAT(5X,"K  =",I3,5X,"RESIDUAL  =",E16.6)
  GOTO(3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1),K
  1 N=(K+4)/3
  SLVT1=0.
101 H26=F(N-1)/6.+S(N-1)*FD(N-1)/18.+FDD(N-1)*S(N-1)**2/168.
  1+11.*F(N)/6.-2.*S(N-1)*FD(N)/9.+FDD(N)*S(N-1)**2/72.
  H36=10.*F(N-1)/99.+8.*S(N-1)*FD(N-1)/231.+FDD(N-1)*S(N-1)**2/
  1264.+155.*F(N)/99.-17.*S(N-1)*FD(N)/99.+FDD(N)*S(N-1)**2/99.
  H46=5.*F(N-1)/77.+S(N-1)*FD(N-1)/44.+FDD(N-1)*S(N-1)**2/396.
  1+15.*F(N)/11.-3.*FD(N)*S(N-1)/22.+FDD(N)*S(N-1)**2/132.
  H56=25.*F(N-1)/572.+20.*S(N-1)*FD(N-1)/1287.+FDD(N-1)*S(N-1)**2
  1/572.+345.*F(N)/286.-95.*S(N-1)*FD(N)/858.+5.*FDD(N)*S(N-1)
  1**2/858.
  H66=25.*F(N-1)/858.+S(N-1)*FD(N-1)/91.+5.*FDD(N-1)*S(N-1)**2/
  14004.+295.*F(N)/273.-25.*S(N-1)*FD(N)/273.+5.*FDD(N)*S(N-1)**2/
  11092.
  H16=25.*F(N-1)/84.+2.*S(N-1)*FD(N-1)/21.+5.*FDD(N-1)*S(N-1)**2
  1/504.+185.*F(N)/84.-25.*S(N-1)*FD(N)/84.+5.*FDD(N)*S(N-1)**2
  1/252.
  H14=3.*F(N-1)/7.+4.*S(N-1)*FD(N-1)/35.+3.*FDD(N-1)*S(N-1)**2/280.
  1+15.*F(N)/14.-13.*S(N-1)*FD(N)/70.+FDD(N)*S(N-1)**2/70.

```



```

H15=5.*F(N-1)/14.+3.*S(N-1)*FD(N-1)/28.+3.*FDD(N-1)*S(N-1)**2
1/280.+23.*F(N)/14.-S(N-1)*FD(N)/4.+FDD(N)*S(N-1)**2/56.
H13=F(N-1)/2.+S(N-1)*FD(N-1)/10.+FDD(N-1)*S(N-1)**2/120.+F(N)/2.
1-S(N-1)*FD(N)/10.+FDD(N)*S(N-1)**2/120.
IF(N.GT.((K+4)/3))GOTO 12
R1=(-30.*H26+135.*H36-213.*H46+144.*H56-36.*H66)/S(N-1)
R2=-18.*H26+75.*H36-564.*H46/5.+369.*H56/5.-18.*H66
R3=(H16/2.-21.*H26/4.+321.*H36/20.-106.*H46/5.
1+129.*H56/10.-3.*H66)*S(N-1)
R4=(3.*H26/2.-33.*H36/4.+149.*H46/10.-111.*H56/10.+3.*H66)*S(N-1)
SLVT1=SLVT1+2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(9.*FD(N-1)/(7.*S(N-1))+FDD(N-1)/7.-9.*FD(N)/(7.*S(N-1))+8.*FDD(
1N)/7.))
IF(N.EQ.IN1)GOTO 111
N=N+1
GOTO 101
12 R1=(-10.*H14+30.*H26+15.*H15-6.*H16+36.*H66+213.*H46
1-135.*H36-144.*H56)/S(N-1)
R2=-6.*H14+18.*H26+8.*H15-3.*H16+18.*H66+564.*H46/5.-75.*H36
1-369.*H56/5.
R3=(H13/2.-H16+21.*H26/4.-321.*H36/20.-3.*H14/2.+106.*H46/5.
1+3.*H15/2.-12.9*H56+3.*H66)*S(N-1)
R4=(H14/2.-3.*H26/2.+33.*H36/4.-149.*H46/10.-H15+111.*H56/10.+
1H16/2.-3.*H66)*S(N-1)
SLVT1=2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(-9.*FD(N-1)/(7.*S(N-1))-8.*FDD(N-1)/7.+9.*FD(N)/(7.*S(N-1))
1-FDD(N)/7.))
111 CONTINUE
WRITE(108,11)K,SLVT1

```

RETURN

2 N=(K+3)/3

SLVT1=0.

201 H26=F(N-1)/6.+S(N-1)*FD(N-1)/18.+FDD(N-1)*S(N-1)**2/168.

1+11.*F(N)/6.-2.*S(N-1)*FD(N)/9.+FDD(N)*S(N-1)**2/72.

H36=10.*F(N-1)/99.+8.*S(N-1)*FD(N-1)/231.+FDD(N-1)*S(N-1)**2/
1264.+155.*F(N)/99.-17.*S(N-1)*FD(N)/99.+FDD(N)*S(N-1)**2/99.

H46=5.*F(N-1)/77.+S(N-1)*FD(N-1)/44.+FDD(N-1)*S(N-1)**2/396.
1+15.*F(N)/11.-3.*FD(N)*S(N-1)/22.+FDD(N)*S(N-1)**2/132.

H56=25.*F(N-1)/572.+20.*S(N-1)*FD(N-1)/1287.+FDD(N-1)*S(N-1)**2
1/572.+345.*F(N)/286.-95.*S(N-1)*FD(N)/858.+5.*FDD(N)*S(N-1)
1**2/858.

H66=25.*F(N-1)/819.+S(N-1)*FD(N-1)/91.+5.*FDD(N-1)*S(N-1)**2/
14004.+295.*F(N)/273.-25.*S(N-1)*FD(N)/273.+5.*FDD(N)*S(N-1)**2/
11092.

H16=25.*F(N-1)/84.+2.*S(N-1)*FD(N-1)/21.+5.*FDD(N-1)*S(N-1)**2
1/504.+185.*F(N)/84.-25.*S(N-1)*FD(N)/84.+5.*FDD(N)*S(N-1)**2
1/252.

H14=3.*F(N-1)/7.+4.*S(N-1)*FD(N-1)/35.+3.*FDD(N-1)*S(N-1)**2/280.
1+15.*F(N)/14.-13.*S(N-1)*FD(N)/70.+FDD(N)*S(N-1)**2/70.

H15=5.*F(N-1)/14.+3.*S(N-1)*FD(N-1)/28.+3.*FDD(N-1)*S(N-1)**2
1/280.+23.*F(N)/14.-S(N-1)*FD(N)/4.+FDD(N)*S(N-1)**2/56.

H13=F(N-1)/2.+S(N-1)*FD(N-1)/10.+FDD(N-1)*S(N-1)**2/120.+F(N)/2.
1-S(N-1)*FD(N)/10.+FDD(N)*S(N-1)**2/120.

IF(N.GT.((K+3)/3))GOTO 202

R1=12.*H26-57.*H36+96.*H46-69.*H56+18.*H66

R2=(36.*H26/5.-159.*H36/5.+51.*H46-177.*H56/5.+9.*H66)*S(N-1)

R3=(-H16/5.+43.*H26/20.-69.*H36/10.+193.*H46/20.

1-31.*H56/5.+3.*H66/2.)*S(N-1)**2

R4=(-3.*H26/5.+69.*H36/20.-133.*H46/20.+53.*H56/10.-3.*H66/2.)

```

1*S(N-1)**2
      SLVT1=2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(-9.*F(N-1)/(7.*S(N-1))-11.*FD(N-1)/14.-11.*S(N-1)*FDD(N-1)/140.
1+9.*F(N)/(7.*S(N-1))-FD(N)/2.-9.*S(N-1)*FDD(N)/140.))
      N=N+1
      GOTO 201
202 R1=-5.*H15+12.*H26-78.*H36+117.*H46+9.*H16-75.*H56+18.*H66
      R2=(-3.*H15+39.*H26/5.-216.*H36/5.+309.*H46/5.
1+24.*H16/5.-192.*H56/5.+9.*H66)*S(N-1)
      R3=(H14/6.+3.*H16/5.+13.*H26/5.-183.*H36/20.-3.*H15/4.
1+231.*H46/20.-67.*H56/10.+3.*H66/2.)*S(N-1)**2
      R4=(H15/4.-2.*H26/5.+24.*H36/5.-33.*H46/4.
1-3.*H16/5.+29.*H56/5.-3.*H66/2.)*S(N-1)**2
      SLVT1=SLVT1+2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(9.*F(N-1)/(7.*S(N-1))+FD(N-1)/2.-9.*FDD(N-1)*S(N-1)/140.
1-9.*F(N)/(7.*S(N-1))+11.*FD(N)/14.-11.*S(N-1)*FDD(N)/140.))
      WRITE(108,11) K, SLVT1
      RETURN
3 N=(K+2)/3
      SLVT1=0.
      IF(N.EQ.1)GOTO 303
301 H26=F(N-1)/6.+S(N-1)*FD(N-1)/18.+FDD(N-1)*S(N-1)**2/168.
1+11.*F(N)/6.-2.*S(N-1)*FD(N)/9.+FDD(N)*S(N-1)**2/72.
      H36=10.*F(N-1)/99.+8.*S(N-1)*FD(N-1)/231.+FDD(N-1)*S(N-1)**2/
1264.+155.*F(N)/99.-17.*S(N-1)*FD(N)/99.+FDD(N)*S(N-1)**2/99.
      H46=5.*F(N-1)/77.+S(N-1)*FD(N-1)/44.+FDD(N-1)*S(N-1)**2/396.
1+15.*F(N)/11.-3.*FD(N)*S(N-1)/22.+FDD(N)*S(N-1)**2/132.
      H56=25.*F(N-1)/572.+20.*S(N-1)*FD(N-1)/1287.+FDD(N-1)*S(N-1)**2
1/572.+345.*F(N)/286.-95.*S(N-1)*FD(N)/858.+5.*FDD(N)*S(N-1)

```

1**2/858.

H66=25.*F(N-1)/819.+S(N-1)*FD(N-1)/91.+5.*FDD(N-1)*S(N-1)**2/
14004.+295.*F(N)/273.-25.*S(N-1)*FD(N)/273.+5.*FDD(N)*S(N-1)**2/
11092.

H16=25.*F(N-1)/84.+2.*S(N-1)*FD(N-1)/21.+5.*FDD(N-1)*S(N-1)**2
1/504.+185.*F(N)/84.-25.*S(N-1)*FD(N)/84.+5.*FDD(N)*S(N-1)**2
1/252.

H14=3.*F(N-1)/7.+4.*S(N-1)*FD(N-1)/35.+3.*FDD(N-1)*S(N-1)**2/280.
1+15.*F(N)/14.-13.*S(N-1)*FD(N)/70.+FDD(N)*S(N-1)**2/70.

H15=5.*F(N-1)/14.+3.*S(N-1)*FD(N-1)/28.+3.*FDD(N-1)*S(N-1)**2
1/280.+23.*F(N)/14.-S(N-1)*FD(N)/4.+FDD(N)*S(N-1)**2/56.

H13=F(N-1)/2.+S(N-1)*FD(N-1)/10.+FDD(N-1)*S(N-1)**2/120.+F(N)/2.
1-S(N-1)*FD(N)/10.+FDD(N)*S(N-1)**2/120.

IF(N.GT.((K+2)/3)) GOTO 302

R1=(-3.*H26/2.+15.*H36/2.-27.*H46/2.+21.*H56/2.-3.*H66)*S(N-1)

R2=(-9.*H26/10.+21.*H36/5.-36.*H46/5.+27.*H56/5.

1-3.*H66/2.)*S(N-1)**2

R3=(H16/40.-11.*H26/40.+37.*H36/40.-11.*H46/8.+19.*H56/20.

1-H66/4.)*S(N-1)**3

R4=(3.*H26/40.-9.*H36/20.+37.*H46/40.-4.*H56/5.+H66/4.)*S(N-1)

1**3

SLVT1=2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+

1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+

1(F(N-1)/7.+11.*S(N-1)*FD(N-1)/140.+FDD(N-1)*S(N-1)**2/140.

1-F(N)/7.+9.*S(N-1)*FD(N)/140.))

303 N=N+1

GOTO 301

302 R1=(-3.*H16/2.+9.*H26-21.*H36+24.*H46-27.*H56/2.+3.*H66)*S(N-1)

R2=(-9.*H16/10.+51.*H26/10.-57.*H36/5.+63.*H46/5.-69.*H56/10.

1+3.*H66/2.)*S(N-1)**2

```

R3=(H15/24.-3.*H16/10.+6.*H26/5.-23.*H36/10.+93.*H46/40.
1-6.*H56/5.+H66/4.)*S(N-1)**3
R4=(3.*H16/40.-21.*H26/40.+11.*H36/8.-69.*H46/40.+
121.*H56/20.-H66/4.)*S(N-1)**3
SLVT1=SLVT1+2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(F(N-1)/7.+9.*S(N-1)*FD(N-1)/140.-F(N)/7.+11.*S(N-1)*FD(N)/140.
1-FDD(N)*S(N-1)**2/140.))
307 CONTINUE
WRITE(108,11)K,SLVT1
RETURN
END
DOUBLE PRECISION FUNCTION SLVT2(X,K,S,D,DD)
C *****
C TO EVALUATE THE RESIDUALS FOR FLATE PLATE /
C WEDGE AND STAGNATION FLOWS
C *****
DIMENSION X(18),S(6),F(7),FD(7),FDD(7)
DOUBLE PRECISION F,S,FD,FDD,H26,H36,H46,H56,H66,H16,H14,
1H15,H13,R1,R2,R3,R4,X
1,SUM
NR=17
IN1=(NR+4)/3
IN11=IN1-1
F(1)=0.00
FD(1)=0.00
FD(IN1)=1.00
FDD(IN1)=00
J=2
DO 5 N=2,IN1
F(N)=X(J)

```

```

      J=J+3
5  CONTINUE
      J=3
      DO 10 N=2,IN11
        FD(N)=X(J)
        J=J+3
10  CONTINUE
      J=1
      DO 15 N=1,IN11
        FDD(N)=X(J)
        J=J+3
15  CONTINUE
      GOTO(3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1),K
      1 N=(K+4)/3
      SLVT2=0.
101 H26=F(N-1)/6.+S(N-1)*FD(N-1)/18.+FDD(N-1)*S(N-1)**2/168.
    1+11.*F(N)/6.-2.*S(N-1)*FD(N)/9.+FDD(N)*S(N-1)**2/72.
      H36=10.*F(N-1)/99.+8.*S(N-1)*FD(N-1)/231.+FDD(N-1)*S(N-1)**2/
1264.+155.*F(N)/99.-17.*S(N-1)*FD(N)/99.+FDD(N)*S(N-1)**2/99.
      H46=5.*F(N-1)/77.+S(N-1)*FD(N-1)/44.+FDD(N-1)*S(N-1)**2/396.
    1+15.*F(N)/11.-3.*FD(N)*S(N-1)/22.+FDD(N)*S(N-1)**2/132.
      H56=25.*F(N-1)/572.+20.*S(N-1)*FD(N-1)/1287.+FDD(N-1)*S(N-1)**2
1/572.+345.*F(N)/286.-95.*S(N-1)*FD(N)/858.+5.*FDD(N)*S(N-1)
1**2/858.
      H66=25.*F(N-1)/819.+S(N-1)*FD(N-1)/91.+5.*FDD(N-1)*S(N-1)**2/
14004.+295.*F(N)/273.-25.*S(N-1)*FD(N)/273.+5.*FDD(N)*S(N-1)**2/
11092.
      H16=25.*F(N-1)/84.+2.*S(N-1)*FD(N-1)/21.+5.*FDD(N-1)*S(N-1)**2
1/504.+185.*F(N)/84.-25.*S(N-1)*FD(N)/84.+5.*FDD(N)*S(N-1)**2
1/252.

```

H14=3.*F(N-1)/7.+4.*S(N-1)*FD(N-1)/35.+3.*FDD(N-1)*S(N-1)**2/280.
1+15.*F(N)/14.-13.*S(N-1)*FD(N)/70.+FDD(N)*S(N-1)**2/70.

H15=5.*F(N-1)/14.+3.*S(N-1)*FD(N-1)/28.+3.*FDD(N-1)*S(N-1)**2
1/280.+23.*F(N)/14.-S(N-1)*FD(N)/4.+FDD(N)*S(N-1)**2/56.

H13=F(N-1)/2.+S(N-1)*FD(N-1)/10.+FDD(N-1)*S(N-1)**2/120.+F(N)/2.
1-S(N-1)*FD(N)/10.+FDD(N)*S(N-1)**2/120.

IF(N.GT.((K+4)/3))GOTO 12

R1=(-30.*H26+135.*H36-213.*H46+144.*H56-36.*H66)/S(N-1)

R2=-18.*H26+75.*H36-564.*H46/5.+369.*H56/5.-18.*H66

R3=(H16/2.-21.*H26/4.+321.*H36/20.-106.*H46/5.

1+129.*H56/10.-3.*H66)*S(N-1)

R4=(3.*H26/2.-33.*H36/4.+149.*H46/10.-111.*H56/10.+3.*H66)*S(N-
11)

SLVT2=2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(9.*FD(N-1)/(7.*S(N-1))+FDD(N-1)/7.-9.*FD(N)/(7.*S(N-1))+8.*FDD(
1N)/7.)/2.)

KK=4

CALL SUV1(N, KK, S, F, FD, FDD, O, SUM)

SLVT2=SLVT2+SUM

IF(N.EQ.IN1)GOTO 111

N=N+1

GOTO 101

12 R1=(-10.*H14+30.*H26+15.*H15-6.*H16+36.*H66+213.*H46
1-135.*H36-144.*H56)/S(N-1)

R2=-6.*H14+18.*H26+8.*H15-3.*H16+18.*H66+564.*H46/5.-75.*H36
1-369.*H56/5.

R3=(H13/2.-H16+21.*H26/4.-321.*H36/20.-3.*H14/2.+106.*H46/5.
1+3.*H15/2.-12.9*H56+3.*H66)*S(N-1)

R4=(H14/2.-3.*H26/2.+33.*H36/4.-149.*H46/10.-H15+111.*H56/10.+
1H16/2.-3.*H66)*S(N+1)

```

      SLVT2=SLVT2+2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(-9.*FD(N-1)/(7.*S(N-1))-8.*FDD(N-1)/7.+9.*FD(N)/(7.*S(N-1))
1-FDD(N)/7.)/2.)
      KK=1
      CALL SUV1(N,KK,S,F,FD,FDD,D,SUM)
      SLVT2=SLVT2+SUM
111 CONTINUE
      RETURN
      2 N=(K+3)/3
      SLVT2=0.
201 H26=F(N-1)/6.+S(N-1)*FD(N-1)/18.+FDD(N-1)*S(N-1)**2/168.
1+11.*F(N)/6.-2.*S(N-1)*FD(N)/9.+FDD(N)*S(N-1)**2/72.
      H36=10.*F(N-1)/99.+8.*S(N-1)*FD(N-1)/231.+FDD(N-1)*S(N-1)**2/
1264.+155.*F(N)/99.-17.*S(N-1)*FD(N)/99.+FDD(N)*S(N-1)**2/99.
      H46=5.*F(N-1)/77.+S(N-1)*FD(N-1)/44.+FDD(N-1)*S(N-1)**2/396.
1+15.*F(N)/11.-3.*FD(N)*S(N-1)/22.+FDD(N)*S(N-1)**2/132.
      H56=25.*F(N-1)/572.+20.*S(N-1)*FD(N-1)/1287.+FDD(N-1)*S(N-1)**2
1/572.+345.*F(N)/286.-95.*S(N-1)*FD(N)/858.+5.*FDD(N)*S(N-1)
1**2/858.
      H66=25.*F(N-1)/819.+S(N-1)*FD(N-1)/91.+5.*FDD(N-1)*S(N-1)**2/
14004.+295.*F(N)/273.-25.*S(N-1)*FD(N)/273.+5.*FDD(N)*S(N-1)**2/
11092.
      H16=25.*F(N-1)/84.+2.*S(N-1)*FD(N-1)/21.+5.*FDD(N-1)*S(N-1)**2
1/504.+185.*F(N)/84.-25.*S(N-1)*FD(N)/84.+5.*FDD(N)*S(N-1)**2
1/252.
      H14=3.*F(N-1)/7.+4.*S(N-1)*FD(N-1)/35.+3.*FDD(N-1)*S(N-1)**2/280.
1+15.*F(N)/14.-13.*S(N-1)*FD(N)/70.+FDD(N)*S(N-1)**2/70.
      H15=5.*F(N-1)/14.+3.*S(N-1)*FD(N-1)/28.+3.*FDD(N-1)*S(N-1)**2
1/280.+23.*F(N)/14.-S(N-1)*FD(N)/4.+FDD(N)*S(N-1)**2/56.

```


H13=F(N-1)/2.+S(N-1)*FD(N-1)/10.+FDD(N-1)*S(N-1)**2/120.+F(N)/2.
1-S(N-1)*FD(N)/10.+FDD(N)*S(N-1)**2/120.

IF(N.GT.((K+3)/3))GOTO 202

R1=12.*H26-57.*H36+96.*H46-69.*H56+18.*H66

R2=(36.*H26/5.-159.*H36/5.+51.*H46-177.*H56/5.+9.*H66)*S(N-1)

R3=(-H16/5.+43.*H26/20.-69.*H36/10.+193.*H46/20.

1-31.*H56/5.+3.*H66/2.)*S(N-1)**2

R4=(-3.*H26/5.+69.*H36/20.-133.*H46/20.+53.*H56/10.-3.*H66/2.)

1*S(N-1)**2

SLVT2=2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+

1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+

1(-9.*F(N-1)/(7.*S(N-1))-11.*FD(N-1)/14.-11.*S(N-1)*FDD(N-1)/140.

1+9.*F(N)/(7.*S(N-1))-FD(N)/2.-9.*S(N-1)*FDD(N)/140.)/2.)

KK=5

CALL SUV1(N,KK,S,F,FD,FDD,D,SUM)

SLVT2=SLVT2+SUM

N=N+1

GOTO 201

202 R1=-5.*H15+12.*H26-78.*H36+117.*H46+9.*H16-75.*H56+18.*H66

R2=(-3.*H15+39.*H26/5.-216.*H36/5.+309.*H46/5.

1+24.*H16/5.-192.*H56/5.+9.*H66)*S(N-1)

R3=(H14/6.+3.*H16/5.+13.*H26/5.-183.*H36/20.-3.*H15/4.

1+231.*H46/20.-67.*H56/10.+3.*H66/2.)*S(N-1)**2

R4=(H15/4.-2.*H26/5.+24.*H36/5.-33.*H46/4.

1-3.*H16/5.+29.*H56/5.-3.*H66/2.)*S(N-1)**2

SLVT2=SLVT2+2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+

1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+

1(9.*F(N-1)/(7.*S(N-1))+FD(N-1)/2.-9.*FDD(N-1)*S(N-1)/140.

1-9.*F(N)/(7.*S(N-1))+11.*FD(N)/14.-11.*S(N-1)*FDD(N)/140.)/2.

KK=2

CALL SUV1(N,KK,S,F,FD,FDD,D,SUM)

SLVT2=SLVT2+SUM

RETURN

3 N=(K+2)/3

SLVT2=0.

IF(N.EQ.1)GOTO 303

301 H26=F(N-1)/6.+S(N-1)*FD(N-1)/18.+FDD(N-1)*S(N-1)**2/168.

1+11.*F(N)/6.-2.*S(N-1)*FD(N)/9.+FDD(N)*S(N-1)**2/72.

H36=10.*F(N-1)/99.+8.*S(N-1)*FD(N-1)/231.+FDD(N-1)*S(N-1)**2/
1264.+155.*F(N)/99.-17.*S(N-1)*FD(N)/99.+FDD(N)*S(N-1)**2/99.

H46=5.*F(N-1)/77.+S(N-1)*FD(N-1)/44.+FDD(N-1)*S(N-1)**2/396.
1+15.*F(N)/11.-3.*FD(N)*S(N-1)/22.+FDD(N)*S(N-1)**2/132.

H56=25.*F(N-1)/572.+20.*S(N-1)*FD(N-1)/1287.+FDD(N-1)*S(N-1)**2/
1/572.+345.*F(N)/286.-95.*S(N-1)*FD(N)/858.+5.*FDD(N)*S(N-1)
1**2/858.

H66=25.*F(N-1)/819.+S(N-1)*FD(N-1)/91.+5.*FDD(N-1)*S(N-1)**2/
14004.+295.*F(N)/273.-25.*S(N-1)*FD(N)/273.+5.*FDD(N)*S(N-1)**2/
11092.

H16=25.*F(N-1)/84.+2.*S(N-1)*FD(N-1)/21.+5.*FDD(N-1)*S(N-1)**2/
1/504.+185.*F(N)/84.-25.*S(N-1)*FD(N)/84.+5.*FDD(N)*S(N-1)**2
1/252.

H14=3.*F(N-1)/7.+4.*S(N-1)*FD(N-1)/35.+3.*FDD(N-1)*S(N-1)**2/280.
1+15.*F(N)/14.-13.*S(N-1)*FD(N)/70.+FDD(N)*S(N-1)**2/70.

H15=5.*F(N-1)/14.+3.*S(N-1)*FD(N-1)/28.+3.*FDD(N-1)*S(N-1)**2/
1/280.+23.*F(N)/14.-S(N-1)*FD(N)/4.+FDD(N)*S(N-1)**2/56.

H13=F(N-1)/2.+S(N-1)*FD(N-1)/10.+FDD(N-1)*S(N-1)**2/120.+F(N)/2.
1-S(N-1)*FD(N)/10.+FDD(N)*S(N-1)**2/120.

IF(N.GT.((K+2)/3)) GOTO 302

R1=(-3.*H26/2.+15.*H36/2.-27.*H46/2.+21.*H56/2.-3.*H66)*S(N-1)

R2=(-9.*H26/10.+21.*H36/5.-36.*H46/5.+27.*H56/5.

1-3.*H66/2.)*S(N-1)**2

```

R3=(H16/40.-11.*H26/40.+37.*H36/40.-11.*H46/8.+19.*H56/20.
1-H66/4.)*S(N-1)**3
R4=(3.*H26/40.-9.*H36/20.+37.*H46/40.-4.*H56/5.+H66/4.)*S(N-1)
1**3

```

```

SLVT2=2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+
1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+
1(F(N-1)/7.+11.*S(N-1)*FD(N-1)/140.+FDD(N-1)*S(N-1)**2/140.
1-F(N)/7.+9.*S(N-1)*FD(N)/140.)/2.)

```

```

KK=6

```

```

CALL SUV1(N, KK, S, F, FD, FDD, D, SUM)

```

```

SLVT2=SLVT2+SUM

```

```

303 N=N+1

```

```

GOTO 301

```

```

302 R1=(-3.*H16/2.+9.*H26-21.*H36+24.*H46-27.*H56/2.+3.*H66)*S(N-1)

```

```

R2=(-9.*H16/10.+51.*H26/10.-57.*H36/5.+63.*H46/5.-69.*H56/10.

```

```

1+3.*H66/2.)*S(N-1)**2

```

```

R3=(H15/24.-3.*H16/10.+6.*H26/5.-23.*H36/10.+93.*H46/40.

```

```

1-6.*H56/5.+H66/4.)*S(N-1)**3

```

```

R4=(3.*H16/40.-21.*H26/40.+11.*H36/8.-69.*H46/40.+

```

```

121.*H56/20.-H66/4.)*S(N-1)**3

```

```

SLVT2=SLVT2+2.*(F(N-1)*R1+FD(N-1)*R2+FDD(N-1)*R3-F(N)*R1+

```

```

1FD(N)*(S(N-1)*R1-R2)+FDD(N)*R4+

```

```

1(F(N-1)/7.+9.*S(N-1)*FD(N-1)/140.-F(N)/7.+11.*S(N-1)*FD(N)/140.

```

```

1-FDD(N)*S(N-1)**2/140.)/2.)

```

```

KK=3

```

```

CALL SUV1(N, KK, S, F, FD, FDD, D, SUM)

```

```

SLVT2=SLVT2+SUM

```

```

307 RETURN

```

```

END

```

```

SUBROUTINE ZSYST1 (F, EPS, NSIG, N, X, ITMAX, S, D, DD, IER)

```

```

*****

```

```

C
C   SOLVE THE NONLINEAR ALGEBRAIC EQ. USING BROWN S METHOD
C   FOR STEADY STATE PLATE , WEDGES , STAGNATION ,
C   SUCTION AND BLOWING FLOWS
C   MODIFIED VERSION OF THE IMSL SUBROUTINE ZSYSTEM
C
C   *****
C   DOUBLE PRECISION   EPS,DELTA,XTEMP,PREC,ETA, TOP,RELCON,F,FMAX
1   PT1,P2,HOLD,H,FPLUS,DERMAX,TEST,X,WA,E,
2   ZERO,PM1
C   1,S.
C   DIMENSION          X(1),WA(224),S(6)
C   DATA              PREC,DELTA/5.0-14,5.0-10/
C   DATA              ZERO,PM1,PT1,P2/0.00,.100,.000100,.00200/
C   IER=0
C   N2 = N+N
C   RELCON=10.0+0**(-NSIG)
C   JTEST = 1
C   IERROR=0
C   IPART=((N+2)*(N-1))/2
C   ITMP=IPART+N
C   LKSUB=ITMP+N
C   DO 130 M = 1, ITMAX
C       IQUIT=0
C       FMAX=ZERO
C       M1 = M-1
C       K1 = LKSUB + 1
C       KMIN = LKSUB + N
C       XTEMP = ZERO
C       DO 5 J = K1,KMIN

```

```

      XTEMP = XTEMP+1.00
      WA(J) = XTEMP
5     CONTINUE
      K = 1
10    IF(K .LE. 1) GO TO 30
      KMIN = K-1
      KK = 1
15    DO 25 K1=1,KMIN
        ISUB=K-K1
        NM=((ISUB-1)*(N2 -ISUB))/2
        LIM=N-ISUB
        KPOINT = WA(LKSUB+ISUB)+PM1
        ISUB1 = ISUB-1
        X(KPOINT)=ZERO
        DO 20 L1=1,LIM
          JS1=ISUB1+L1
          LKJSUB=LKSUB + JS1 +1
          IJ=NM+JS1
          JPOINT= WA(LKJSUB) + PM1
          X(KPOINT)=X(KPOINT) + WA(IJ)*X(JPOINT)
20      CONTINUE
        X(KPOINT)=X(KPOINT) + WA(NM+N)
25    CONTINUE
      IF(KK.EQ.1)GOTO 30
      IF(KK.EQ.2)GOTO 45
      IF(KK.EQ.3)GOTO 105
30    E = F(X,K,S,D,DD)
      FMAX= DMAX1(FMAX,DABS(E))
      IF(DABS(E) .GE. EPS) GO TO 35
      IQUIT=IQUIT+1
      IF(IQUIT .EQ. N) GO TO 140

```

```

35   I = K
40   IP=IPART+I
      ITEM = WA(LKSUB+I) + PM1
      HOLD = X(ITEM)
      ETA=.001D+00*DABS(HOLD)
      IF(DABS(HOLD) .LT. PREC) ETA=DELTA
      H = DMIN1(FMAX,ETA)
      IF(H .LT. PREC) H=PREC
      X(ITEM)=HOLD+H
      IF(K.LE.1)GOTO 45
      KK = 2
      GO TO 15
45   FPLUS = F(X,K,S,D,DD)
      TOP=FPLUS-E
      IF(DABS(TOP) .GE. PREC) GO TO 50
      WA(IP)=ZERO
      GO TO 55
50   WA(IP)=TOP/H
55   X(ITEM)=HOLD
      I = I + 1
      IF (I .LE. N) GO TO 40
      IF (K .LT. N) GO TO 60
      IP=IPART+N
      IF(DABS(WA(IP)) .EQ. ZERO) GO TO 80
      X(ITEM) = -E/ WA(IP) + X(ITEM)
      GO TO 100
60   KL=LKSUB+K
      LOOK= WA(KL) + PM1
      KNAX=LOOK
      IP=IPART+K

```

```

DERMAX=DABS( WA(IP))
KPLUS = K+1
DO 65 I = KPLUS,N
    TEST=DABS( WA(IPART+I))
    IF(TEST .LE. DERMAL) GO TO 65
    DERMAL = TEST
    KMAX=I
65  CONTINUE
    IF(LOOK .EQ. KMAX) GO TO 75
    LKMAX=LKSUB+KMAX
    WA(KL)=WA(LKMAX)
    WA(LKMAX)=LOOK
    IP=IPART+KMAX
    XTEMP= WA(IP)
    IPK=IPART+K
    WA(IP)=WA(IPK)
    WA(IPK)=XTEMP
    IF(K .LT. 2) GO TO 75
    KMIN=K-1
    I1 = 0
    DO 70 I=1,KMIN
        L=((I1 )*(N2 -I))/2-1
        J=L+KMAX
        XTEMP= WA(J)
        JJ=L+K
        WA(J)=WA(JJ)
        WA(JJ)=XTEMP
        I1 = I
70  CONTINUE
75  IF (DABS(WA(IPART+K)).NE. ZERO) GO TO 90
80  IF(IERROR .EQ. 1) GO TO 135

```

```

      DO 85 I=1,N
        X(I) = 0.900*X(I) + .1234500
85    CONTINUE
      IERROR=1
      GO TO 105
90    L=((K-1)*(N2-K))/2
      KN=L+N
      I1 = L-1
      WA(KN)=ZERO
      IPK=IPART+K
      DO 95 J = KPLUS,N
        JSUB= WA(LKSUB+J) + PM1
        JJ=I1+J
        IPJ=IPART+J
        WA(JJ)=-WA(IPJ)/WA(IPK)
        WA(KN)=WA(KN)+WA(IPJ)*X(JSUB)
95    CONTINUE
      LK= WA(LKSUB+K) + PM1
      WA(KN)=(WA(KN)-E)/WA(IPK) + X(LK)
      K = K+1
      IF (K .LE. N) GO TO 10
100   IF( N .EQ. 1) GO TO 105
      KMIN=N-1
      KK = 3
      GO TO 15
105   IF (M .LE. 1) GO TO 120
      DO 110 I = 1,N
        IF(DABS(WA(ITMP+I)-X(I)) .GT. DABS(X(I))*RELCON) GO TO
110   CONTINUE
      JTEST = JTEST+1

```



```
      IF (JTEST-3)120,140,140
115    JTEST = 1
120    DO 125 I = 1,N
        WA(ITMP+I)=X(I)
125  CONTINUE
130  CONTINUE
      M=ITMAX
      IER = 129
      GO TO 240
135  IER = 130
140  FMAX=ZERO
      TEST=1.00+15
      IF (N.GT.1) GOTO 145
      WA(IPART+2) = F(X,1,S,D,DD)
      FMAX= DMAX1(FMAX,DABS(WA(IPART+2)))
      GO TO 155
145  DO 150 I=1,N
      IP=IPART+I
      WA(IP) = F(X,I,S,D,DD)
      FMAX=DMAX1(FMAX,DABS(WA(IP)))
150  CONTINUE
155  K=1
      DO 160 I=1,N
      WA(I)=X(I)
      IF(DABS(X(I)) .GT. P2) GO TO 160
      K=2
      WA(I)=ZERO
160  CONTINUE
      IF(K .EQ. 1) GO TO 195
      KK = 1
      GO TO 205
```

```
165 IF(FMAX .LT. TEST) GO TO 190
    DO 170 I=1,N
        X(I)= WA(I)
170 CONTINUE
    IF (N .GT. 1) GO TO 175
    WA(IPART+2) = WA(ITMP+2)
    GO TO 185
175 DO 180 I = 1,N
        WA(IPART+I) = WA(ITMP+I)
180 CONTINUE
185 FMAX=TEST
190 K=1
195 ITEST=0
    DO 200 I=1,N
        WA(I)=X(I)
        IF(DABS(X(I)) .LE. P2) GO TO 200
        L=X(I)+PT1
        J=X(I)-PT1
        IF(L .EQ. J) GO TO 200
        WA(I) =ISIGN(1,J)*MAX0(IABS(L),IABS(J))
        K=2
200 CONTINUE
    IF(K .EQ. 1) GO TO 235
    KK = 2
205 TEST=ZERO
    IF (N .GT. 1) GO TO 210
    WA(ITMP+2) = F(WA,1,S,D,DD)
    TEST = DMAX1(TEST,DABS(WA(ITMP+2)))
    GO TO 220
210 DO 215 I=1,N
```

```

      IT=ITMP+I
      WA(IT) = F(WA,I,S,D,DD)
      TEST=DMAX1(TEST,DABS(WA(IT)))
215  CONTINUE
220  IF(KK.EQ.1) GOTO 165
225  IF(FMAX .LT. TEST) GO TO 235
250  DO 230 I=1,N
      X(I)= WA(I)
230  CONTINUE
260  ITEST=1
235  IF(FMAX .LT. EPS .OR. TEST .LT. EPS) IER = 0
240  ITHAX=M1 + 1
9000 CONTINUE
270  IF(IER.EQ.0) GOTO 9005
      PRINT 88
      88 FORMAT(5X,'MATRIX IS ILL CONDITIONED')
9005 RETURN
      END
      SUBROUTINE SUV1(N,KK,S,F,FD,FDD,D,SUM)
C *****
C
C   WEDGE AND STAGNATION FLOWS
C   HELPER FOR SLVT2
C
C *****
C
      DOUBLE PRECISION F,FD,FDD, SUM,B,BT,AA,SK,SKK,A,S,C1,C2,C3,A
1A4,A5,FF,FFF,F4,A2,A3
      DIMENSION F(7),FD(7),FDD(7),S(6),B(6,6),BT(6,6),AA(6),SK(6,6)
1FF(6),FFF(6),F4(6),A(6),SKK(6,6)
      SUM=0.00
      DO 10 I=1,6

```

```

F4(I)=0.00
A(I)=0.00
AA(I)=0.00
FFF(I)=0.00
DO 5 J=1,6
SK(I,J)=0.00
SKK(I,J)=0.00
B(I,J)=0.00
BT(I,J)=0.00
5 CONTINUE
10 CONTINUE
B(1,1)=1.00
B(2,2)=1.00
B(3,3)=.500
B(4,1)=-10.00/(S(N-1)**3)
B(5,1)= 15.00/(S(N-1)**4)
B(6,1)=-6.00/(S(N-1)**5)
B(4,2)=-6.00/(S(N-1)**2)
B(5,2)= 8.00/(S(N-1)**3)
B(6,2)=-3.00/(S(N-1)**4)
B(4,3)= -1.500/S(N-1)
B(5,3)= 1.500/(S(N-1)**2)
B(6,3)= -.500/(S(N-1)**3)
B(4,4)= -B(4,1)
B(5,4)= -B(5,1)
B(6,4)= -B(6,1)
B(4,5)=B(4,1)*S(N-1)-B(4,2)
B(5,5)=B(5,1)*S(N-1)-B(5,2)
B(6,5)=B(6,1)*S(N-1)-B(6,2)
B(4,6)= .500/S(N-1)

```

B(5,6) = -1.00/(S(N-1)**2)

B(6,6) = .500/(S(N-1)**3)

DO 20 I=1,6

DO 15 J=1,6

BT(I,J)=B(J,I)

15 CONTINUE

20 CONTINUE

C1=(-F(N-1)+F(N))*10.00/S(N-1)**2 - (FD(N-1)*6.00+FD(N)*4.00)/
1S(N-1)+(-FDD(N-1)*3.00+FDD(N))*500

C2=(F(N-1)-F(N))*15.00/S(N-1)**2+(FD(N-1)*8.00+FD(N)*7.00)/S(N-1)
1+FDD(N-1)*1.500-FDD(N)

C3=(-F(N-1)+F(N))*6.00/S(N-1)**2-(FD(N-1)+FD(N))*3.00/S(N-1)+
1(-FDD(N-1)+FDD(N))*500

A1= FD(N-1)

A2= FDD(N-1)*S(N-1)

A3= C1*S(N-1)

A4= C2*S(N-1)

A5= C3*S(N-1)

SK(1,2)=(A1+A2*.500+A3+A4+A5)*S(N-1)

SK(1,3)= (A1+A2*2.00/3.00+A3*1.500+A4*8.00/5.00+A5*5.00/3.00)
1*S(N-1)**2

SK(1,4)= (A1+A2*3.00/4.00+A3*9.00/5.00+A4*2.00+A5*15.00/7.00)
1*S(N-1)**3

SK(1,5)= (A1+A2*4.00/5.00+A3*2.00+A4*16.00/7.00+A5*2.500)*S(N-1)
1**4

SK(1,6)= (A1+A2*5.00/6.00+A3*15.00/7.00+A4*2.500+A5*25.00/9.00)
1*S(N-1)**5

SK(2,2) = SK(1,3)*.500

SK(2,3) = SK(1,4)*2.00/3.00

SK(2,4) = SK(1,5)*.7500

SK(2,5) = SK(1,6)*.800

SK(2,6)=(A1*5.00/6.00+A2*5.00/7.00+A3*15.00/8.00+A4*20.00/9.00+A
1*2.500)*S(N-1)**6

SK(3,2) = SK(1,4)/3.00

SK(3,3) = SK(1,5)*.500

SK(3,4) = SK(1,6)*3.00/5.00

SK(3,5) = SK(2,6)*.800

SK(3,6)=(A1*5.00/7.00+A2*5.00/8.00+A3*15.00/9.00+A4*2.00+A5*25.0
1/11.00)*S(N-1)**7

SK(4,2) = SK(1,5)*.2500

SK(4,3) = SK(1,6)*.400

SK(4,4) = SK(2,6)*.600

SK(4,5) = SK(3,6)*.800

SK(4,6)=(A1*5.00/8.00+A2*5.00/9.00+A3*1.500+A4*20.00/11.00+A5*25.
100/12.00)*S(N-1)**8

SK(5,2) = SK(1,6)/5.00

SK(5,3) = SK(2,6)*.400

SK(5,4) = SK(3,6)*.600

SK(5,5) = SK(4,6)*.800

SK(5,6)=(A1*5.00/9.00+A2*.500+A3*15.00/11.00+A4*5.00/3.00+A5*25.
100/13.00)*S(N-1)**9

SK(6,2) = SK(2,6)*.200

SK(6,3) = SK(3,6)*.400

SK(6,4) = SK(4,6)*.600

SK(6,5) = SK(5,6)*.800

SK(6,6)=(A1*.500+A2*5.00/11.00+A3*1.2500+A4*20.00/13.00+A5*25.00
1/14.00)*S(N-1)**10

DO 55 I=1,6

DO 50 J=1,6

DO 45 K=1,6

45 SKK(I,J)=SKK(I,J)+SK(I,K)*B(K,J)

50 CONTINUE

55 CONTINUE

FF(1) = F(N-1)

FF(2) = FD(N-1)

FF(3) = FDD(N-1)

FF(4) = F(N)

FF(5) = FD(N)

FF(6) = FDD(N)

DO 65 I=1,6

DO 60 K=1,6

60 AA(I)=AA(I)+SKK(I,K)*FF(K)

65 CONTINUE

FFF(1)=S(N-1)-AA(1)

FFF(2)=S(N-1)**2/2.00-AA(2)

FFF(3)=S(N-1)**3/3.00-AA(3)

FFF(4)=S(N-1)**4/4.00-AA(4)

FFF(5)=S(N-1)**5/5.00-AA(5)

FFF(6)=S(N-1)**6/6.00-AA(6)

DO 75 I=1,6

DO 70 K=1,6

70 F4(I) = F4(I)+BT(I,K)*FFF(K)

75 CONTINUE

SUM = D*F4(KK)

RETURN

END

C.4.2. UNSTEADY FLOW WITH PRIMITIVE VARIABLES

```

C *****
C *
C *MAIN PROGRAM TO SOLVE THE SUDDENLY ACCELERATED PLATE *
C *USING PRIMITIVE VARIABLES AND QUADRATIC ELEMENT *
C *
C *****

ASSIGN F:105, (FILE,DAT), (IN)
ASSIGN F:106, (FILE,TEMP), (OUT), (SAVE)
EXTERNAL SLVT3
DOUBLE PRECISION XO,B1,B2,B3,C1,C2,C3,DELT,XB
1,SH
DOUBLE PRECISION X,EPS,UU,YY
1,DT,II
COMMON SLVT/ NN1(40),NN2(40),NN3(40),NN4(40),NN5(40),NN6(40),
1NEC1(105),NEC2(105),NEC3(105),NEC4(105),NEC5(105),NEC6(105),
1NEO1(105),NEO2(105),NEO3(105),NEO4(105),NEO5(105),NEO6(105)
1,YY(105)
DIMENSION B1(40),B2(40),B3(40),C1(40),C2(40),C3(40),DELT(40)
DIMENSION X(30),XO(41),XB(156),UU(42)
NRT=39
NR=NRT-1
NR1=NR/2
NF=30
NF1=10
NF2=NF-NF1
CALL GRIDT2
1      (DELT,NN1,NN2,NN3,NN4,NN5,NN6,B1,B2,B3,C1,C2,C3,YY)
CALL NODT(NEC1,NEC2,NEC3,NEC4,NEC5,NEC6,NEO1,NEO2,NEO3,NEO4,
1NEO5,NEO6)
DO 10 I=1,NRT
READ(105,20) XO(I)
10 CONTINUE

```



```

50 DO 15 I=1,NF2
    X(I)=0.000
15 CONTINUE
    DO 45 I=1,NF1
        KO=NF2+I
        READ(105,20) X(KO)
45 CONTINUE
        NRR=4*NRT
        DO 80 K=1,NRR
            READ(105,20) XB(K)
80 CONTINUE
            TI=1.00
            DT=0.02500
            TI=TI+DT
            WRITE(108,70) TI
            WRITE(108,30) ( X(K),K=1,NRT)
            EPS=1.0D-4
            NSIG=4
            ITMAX=10
            CALL ZSYST2(SLVT1,EPS,NSIG,NF,X,XO,ITMAX,IER,DELT,B1,B2,B3,
1C1,C2,C3,DT,XB,NF2)
            IF(IER.NE.0) GOTO 130
            CALL SOLV(VV,X,XO,UU)
            KM=NR1+3
            WRITE(108,41) (UU(I),I=KM,41)
            WRITE(108,40) ITMAX,(UU(I),I=1,20)
                SH=2.*(4.*UU(2)-UU(3))
            WRITE(108,43) SH
20 FORMAT(F10.7)
30 FORMAT(10X,*INITIAL GUESS      XOLD(1----N)*,/,(/,6E16.6))
40 FORMAT(5X,*ITMAX  =*,I3,/,5X,*      X(1)-----X(N)*,/,(/,6E16.6
1),/,10X,*SUCCESSFUL CONVERGENCE*)
41 FORMAT(10X,*U,S AT X=1.25 FOR THE SAME T*,/,5X,(/,6E16.6))
43 FORMAT(/,20X,*S H E A R  S T R E S S *,//,20X,E16.8,/)
70 FORMAT(/,15X,*TIME T =*,E16.8,/)
130 STOP

```

```

END
SUBROUTINE GRIDT2
1          (DELT, NN1, NN2, NN3, NN4, NN5, NN6, B1, B2, B3, C1, C2, C3
1, YY)
C          *****
C
C          FOR THE SUDDENLY ACCELERATED PLATE USING PRIMITIVE
C          VARIABLES AND QUADRATIC ELEMENT
C          *****
C          DIMENSION XX(105), YY(105), NN1(40), NN2(40), NN3(40), NN4(40), NN5(40),
1, NN6(40), B1(40), B2(40), B3(40), C1(40), C2(40), C3(40), DELT(40)
1, PA(21)
          DOUBLE PRECISION XX, B1, B2, B3, C1, C2, C3, DELT
1, P
1, YY, PA
          P=0.000
          L=1
          M=3
          N=5
          DO 5 I=1,9
          YY(N)=P
          YY(L)=P*SQRT(2.)
          YY(M)=P*SQRT(1.5)
          XX(L)=2.000
          XX(M)=1.500
          XX(N)=1.000
          L=L+5
          M=M+5
          N=N+5
5  P=P+.2500

```

4

OF/DE

4

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
L=46
M=48
N=50
P=2.37500
DO 6 I=1,4
  YY(N)=P
  YY(L)=P*SQRT(2.)
  YY(M)=P*SQRT(1.5)
  XX(L)=2.000
  XX(M)=1.500
  XX(N)=1.000
  L=L+5
  M=M+5
  N=N+5
6 P=P+.37500
  L=66
  M=68
  N=70
  P=4.000
  DO 7 I=1,4
    YY(N)=P
    YY(L)=P*SQRT(2.)
    YY(M)=P*SQRT(1.5)
    XX(L)=2.000
    XX(N)=1.000
    XX(M)=1.500
    L=L+5
    M=M+5
    N=N+5
7 P=P+.500
```

```
L=86
M=88
N=90
P=6.2500
DO 8 I=1,4
  YY(N)=P
  YY(L)=P*SQRT(2.)
  YY(M)=P*SQRT(1.5)
  XX(L)=2.000
  XX(N)=1.000
  XX(M)=1.500
  L=L+5
  M=M+5
  N=N+5
8 P=P+.7500
  L=2
  M=4
  DO 10 J=1,11
    XX(L)=(XX(L-1)+XX(L+1))/2.
    YY(L)=(YY(L-1)+YY(L+1))/2.
    XX(M)=(XX(M-1)+XX(M+1))/2.
    YY(M)=(YY(M-1)+YY(M+1))/2.
    PA(J)=YY(M)/DSORT(XX(M))
    L=L+10
10 M=M+10
  L=7
  M=9
  IK=11
  DO 15 KC=1,10
    XX(L)=(XX(L-6)+XX(L+6))/2.
    YY(L)=(YY(L-6)+YY(L+6))/2.
```

```
XX(M)=(XX(M-6)+XX(M+6))/2.  
YY(M)=(YY(M-6)+YY(M+6))/2.  
PA(IK+KC)=YY(M)/DSQRT(XX(M))  
L=L+10  
15 M=M+10  
WRITE(108,20)  
DO 25 NN=1,105  
WRITE(108,30) NN,XX(NN),YY(NN)  
25 CONTINUE  
L=0  
M=12  
DO 35 IC=1,10  
DO 40 LC=1,4  
N=L+LC  
IF(LC.EQ.2) GOTO 55  
IF(LC.EQ.4) GOTO 55  
NN1(N)=N+M  
NN2(N)=NN1(N)-12  
NN3(N)=NN1(N)-2  
NN4(N)=NN1(N)-6  
NN5(N)=NN1(N)-7  
NN6(N)=NN1(N)-1  
GOTO 40  
55 NN1(N)=N+M-11  
NN2(N)=NN1(N)-2  
NN3(N)=NN1(N)+10  
NN4(N)=NN1(N)-1  
NN5(N)=NN1(N)+4  
NN6(N)=NN1(N)+5  
40 CONTINUE
```

```

      M=M+6
35  L=L+4
      DO 60 NE=1,40
        NV1=NN1 (NE)
        NV2=NN2 (NE)
        NV3=NN3 (NE)
        B1 (NE)=YY (NV2) -YY (NV3)
        B2 (NE)=YY (NV3) -YY (NV1)
        B3 (NE)=YY (NV1) -YY (NV2)
        C1 (NE)=XX (NV3) -XX (NV2)
        C2 (NE)=XX (NV1) -XX (NV3)
        C3 (NE)=XX (NV2) -XX (NV1)
        DELT (NE)=(XX (NV2)*YY (NV3)-XX (NV3)*YY (NV2)-XX (NV1)*(YY (NV3)-
1YY (NV2))+YY (NV1)*(XX (NV3)-XX (NV2)))/2.
60  CONTINUE
      WRITE (108,45)
      DO 65 NE=1,40
        WRITE (108,50) NE,NN1 (NE),NN2 (NE),NN3 (NE),NN4 (NE),NN5 (NE),NN6 (NE),
1DELT (NE)
65  CONTINUE
      M=1
      DO 70 J=1,11
        YY (M)=PA (J)
70  M=M+10
      IK=11
      M=6
      DO 75 J=1,10
        YY (M)=PA (IK+J)
75  M=M+10
20  FORMAT (7X,*NODE*,16X,*X*,22X,*Y*,/)
30  FORMAT (5X,I5,9X,E16.8,9X,E16.8)

```

```
45 FORMAT(/,5X,*ELEM=*,5X,*NODES CONNECTED WITH*,17X,*AREA*)
```

```
50 FORMAT(5X,7I5,3X,E16.6)
```

```
RETURN
```

```
END
```

```
SUBROUTINE NODT(NEC1,NEC2,NEC3,NEC4,NEC5,NEC6,NEO1,NEO2,NEO3,  
1NEO4,NEO5,NEO6)
```

```
C *****
```

```
C
```

```
C TO SET NODES AND ELEMENT CONNECTIONS.
```

```
C
```

```
C *****
```

```
DIMENSION NEC1(105),NEC2(105),NEC3(105),NEC4(105),NEC5(105),  
1NEC6(105),NEO1(105),NEO2(105),NEO3(105),NEO4(105),NEO5(105),  
1NEO6(105)
```

```
DO 5 NCC=1,105
```

```
NEC1(NCC)=0
```

```
NEC2(NCC)=0
```

```
NEC3(NCC)=0
```

```
NEC4(NCC)=0
```

```
NEC5(NCC)=0
```

```
NEC6(NCC)=0
```

```
NEO1(NCC)=0
```

```
NEO2(NCC)=0
```

```
NEO3(NCC)=0
```

```
NEO4(NCC)=0
```

```
NEO5(NCC)=0
```

```
NEO6(NCC)=0
```

```
5 CONTINUE
```

```
NO=6
```

```
LL=1
```



```
DO 10 KC=1,10
  NEC1(NO)=LL
  NEO1(NO)=5
  NO=NO+10
10 LL=LL+4
  NO=10
  LL=4
  DO 15 KC=1,10
    NEC1(NO)=LL
    NEO1(NO)=6
    NO=NO+10
15 LL=LL+4
  NO=7
  LL=1
  DO 20 KC=1,10
    NEC1(NO)=LL
    NEO1(NO)=4
    NEC2(NO)=LL+1
    NEO2(NO)=5
    NEC1(NO+1)=LL+1
    NEO1(NO+1)=6
    NEC2(NO+1)=LL+2
    NEO2(NO+1)=5
    NO=NO+10
20 LL=LL+4
  NO=12
  LL=1
  DO 25 KC=1,9
    NEC1(NO)=LL
    NEO1(NO)=6
    NEC2(NO)=LL+5
```

NEO2(N0)=4
NEC1(N0+2)=LL+2
NEO1(N0+2)=6
NEC2(N0+2)=LL+7
NEO2(N0+2)=4
N0=N0+10

25 LL=LL+4
N0=9
LL=3
DO 30 KC=1,10
NEC1(N0)=LL
NEO1(N0)=4
NEC2(N0)=LL+1
NEO2(N0)=5
N0=N0+10

30 LL=LL+4
NEC1(1)=1
NEO1(1)=2
NEC2(1)=2
NEO2(1)=2
NEC1(2)=2
NEO1(2)=4
NEC1(3)=2
NEO1(3)=1
NEC2(3)=3
NEO2(3)=2
NEC3(3)=4
NEO3(3)=2
NEC1(4)=4
NEO1(4)=4

```
NEC1(5)=4
NEO1(5)=1
NO=15
LL=3
DO 35 KC=1,9
NEC1(NO)=LL
NEO1(NO)=1
NEC2(NO)=LL+5
NEO2(NO)=1
NEC3(NO)=LL+1
NEO3(NO)=3
NO=NO+10
35 LL=LL+4
NO=11
LL=1
DO 40 KC=1,9
NEC1(NO)=LL
NEO1(NO)=3
NEC2(NO)=LL+4
NEO2(NO)=2
NEC3(NO)=LL+5
NEO3(NO)=2
NO=NO+10
40 LL=LL+4
NO=13
LL=1
DO 45 KC=1,9
NEC1(NO)=LL
NEO1(NO)=1
NEC2(NO)=LL+5
NEO2(NO)=1
```

NEC3(N0)=LL+6

NE03(N0)=2

NEC4(N0)=LL+7

NE04(N0)=2

NEC5(N0)=LL+1

NE05(N0)=3

NEC6(N0)=LL+2

NE06(N0)=3

N0=N0+10

45 LL=LL+4

N1=1

N2=37

N3=100

NEC1(N3+1)=N2

NE01(N3+1)=N1+2

NEC1(N3+2)=N2

NE01(N3+2)=N1+5

NEC1(N3+4)=N2+2

NE01(N3+4)=N1+5

IO=N3+3

NEC1(IO)=N2

NE01(IO)=N1

NEC2(IO)=N2+1

NE02(IO)=N1+2

NEC3(IO)=N2+2

NE03(IO)=N1+2

II=N3+5

NEC1(II)=N2+2

NE01(II)=N1

NEC2(II)=N2+3

```

      NEO2(II)=N1+2
      WRITE(108,55)
      DO 50 KC=1,105
        WRITE(108,60) KC, NEC1(KC), NEO1(KC), NEC2(KC), NEO2(KC), NEC3(KC),
          1NEO3(KC), NEC4(KC), NEO4(KC), NEC5(KC), NEO5(KC), NEC6(KC), NEO6(KC)
50    CONTINUE
55    FORMAT(3X, *NODE= *, *ELEMENT=  + THE ORDER OF THE NODE IN THIS EL
      1MENT RESPEC. *)
60    FORMAT(3X, 13I4)
      RETURN
      END
      FUNCTION SLVT3(X,X0,K,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
C      *****
C
C      TO EVALUATE THE RESIDUALS
C      FOR THE SUDDENLY ACCELERATED PLATE USING PRIMITIVE
C      VARIABLES AND QUADRATIC ELEMENT
C
C      *****
      DOUBLE PRECISION SLVT3,X,SUM
      DOUBLE PRECISION X0,B1,B2,B3,C1,C2,C3,DELT,XB,UP,UPO,VP,VPO,V,U
1,SU,SUO,SV,SV0,TT,TDT,VO,UO,DA,DT
      DOUBLE PRECISION BB1,BB2,BB3,CC1,CC2,CC3
1,YY,A,B,C,D,E,A1,A2,A3
      COMMON/SLVT/ NN1(40),NN2(40),NN3(40),NN4(40),NN5(40),NN6(40),
1NEC1(105),NEC2(105),NEC3(105),NEC4(105),NEC5(105),NEC6(105),
1NEO1(105),NEO2(105),NEO3(105),NEO4(105),NEO5(105),NEO6(105)
1,YY(105)
      DIMENSION X(1),X0(1)
      DIMENSION B1(40),B2(40),B3(40),C1(40),C2(40),C3(40),DELT(40),
1VPO(6),TT(6),TDT(6),          NV(6),NVO(6),UP(6),UPO(6),VP(6),

```

```
1V(105),VO(105),U(105),UO(105),XB(156),SU(6),SUO(6),SV(6),SVO(6)
NF=19
NFF=2*NF
NF1=NF+1
NF2=10
NDF=4*NF
DO 71 KL=1,NF1
KC=5*KL+5
71 V(KC)=X(KL)
DO 10 KL=1,NF1
KC=KL+8
KN=KL*10-(KC-9)*5
10 VO(KN)=XO(KL)
DO 72 KL=1,NF2
KLL=KL+NF1
KN=10*KL
72 U(KN)=X(KLL)
DO 15 KL=1,NF
KLL=KL+NF1
KC=KL+23
KN=KL*10-(KC-24)*5
15 UO(KN)=XO(KLL)
DO 20 KB=1,5
UO(KB+100)=1.
U(KB+100)=1.
VO(KB)=0.
UO(KB)=0.
V(KB)=0.
U(KB)=0.
20 CONTINUE
```

```

      L1=8
      L2=9
      DO 75 KC=1,NF1
      NK1=NDF+NF1+KC
      V(L1)=XB(NK1)
      NK=NDF+KC
      V(L2)=XB(NK)
      NKK=NK+2*NF1
      VO(L2)=XB(NKK)
      NK2=NK1+2*NF1
      VO(L1)=XB(NK2)
      L1=L1+5
75  L2=L2+5
      L1=9
      M=0
      DO 85 KC=1,NF
      KCC=KC+M
      KCO=KCC+NFF
      U(L1)=XB(KCC)
      UO(L1)=XB(KCO)
85  L1=L1+5
      M=NF
      L2=8
      DO 90 KC=1,NF
      KCC=KC+M
      KCO=KCC+NFF
      U(L2)=XB(KCC)
      UO(L2)=XB(KCO)
90  L2=L2+5
      NF22=NF2-1
      DO 111 KC=1,NF22

```

```

KNN=KC*10+5
A=(U(KNN-10)-U(KNN-5))/(YY(KNN-10)**2-YY(KNN-5)**2)
B=(U(KNN-5)-U(KNN+5))/((YY(KNN-10)+YY(KNN-5))*(YY(KNN-5)-YY(KNN+
1)))
C=1.-(YY(KNN-5)+YY(KNN+5))/(YY(KNN-10)+YY(KNN-5))
D=(U(KNN-5)-U(KNN+5))/(YY(KNN-5)-YY(KNN+5))
E=YY(KNN-5)+YY(KNN+5)
A3=(A-B)/C
A2=D-A3*E
A1=U(KNN-10)-YY(KNN-10)*A2-(YY(KNN-10)**2)*A3
111 U(KNN)=A1+YY(KNN)*A2+A3*YY(KNN)**2
133 IF(K.LE.NF2) GOTO 35
    IF(K.GT.NF2) GOTO 30
30 NO=K*10-(K+9)*5
33 NV(1)=NEC1(NO)
    NV0(1)=NE01(NO)
    NV(2)=NEC2(NO)
    NV0(2)=NE02(NO)
    NV(3)=NEC3(NO)
    NV0(3)=NE03(NO)
    NV(4)=NEC4(NO)
    NV0(4)=NE04(NO)
    NV(5)=NEC5(NO)
    NV0(5)=NE05(NO)
    NV(6)=NEC6(NO)
    NV0(6)=NE06(NO)
    SUM=0.000
    DO 50 I=1,6
    N=NV(I)
    IF(N.E0.0) GOTO 51

```



```

KK=NVO(I)
NV1=NN1(N)
NV2=NN2(N)
NV3=NN3(N)
NV4=NN4(N)
NV5=NN5(N)
NV6=NN6(N)
      GOTO(4,17,8,17,17,16),KK
4  SUM=SUM+B1(N)*U(NV1)/15.+C1(N)*V(NV1)/15.-B2(N)*U(NV2)/30.-C2(N)
    1*V(NV2)/30.-B3(N)*U(NV3)/30.-C3(N)*V(NV3)/30.+(B2(N)-B1(N)/2.)
    1*U(NV4)/15.+(C2(N)-C1(N)/2.)*V(NV4)/15.-(B3(N)+B2(N))*U(NV5)/30.
    1-(C3(N)+C2(N))*V(NV5)/30.+(B3(N)-B1(N)/2.)*U(NV6)/15.+(C3(N)-C1
    1(N)/2.)*V(NV6)/15.
      GOTO 50
8  SUM=SUM-B1(N)*U(NV1)/30.-C1(N)*V(NV1)/30.-B2(N)*U(NV2)/30.-C2(N)
    1*V(NV2)/30.+B3(N)*U(NV3)/15.+C3(N)*V(NV3)/15.-(B1(N)+B2(N))
    1*U(NV4)/30.-(C1(N)+C2(N))*V(NV4)/30.+(B2(N)-B3(N)/2.)*U(NV5)/15.
    1+(C2(N)-C3(N)/2.)*V(NV5)/15.+(B1(N)-B3(N)/2.)*U(NV6)/15.+(C1(N)
    1-C3(N)/2.)*V(NV6)/15.
      GOTO 50
16 SUM=SUM+B1(N)*U(NV1)/10.+C1(N)*V(NV1)/10.-B2(N)*U(NV2)/30.-C2(N)
    1*V(NV2)/30.+B3(N)*U(NV3)/10.+C3(N)*V(NV3)/10.+4.*(B2(N)+B1(N)
    1/2.)*U(NV4)/15.+4.*(C2(N)+C1(N)/2.)*V(NV4)/15.+4.*(B2(N)+B3(N)/2.
    1)*U(NV5)/15.+4.*(C2(N)+C3(N)/2.)*V(NV5)/15.+4.*(B1(N)+B3(N))
    1*U(NV6)/15.+4.*(C1(N)+C3(N))*V(NV6)/15.
50 CONTINUE
51 SLVT3=SUM
17 RETURN
35 NO=K*10
    NV(1)=NEC1(NO)
    NVO(1)=NEO1(NO)

```

```
NV(2)=NEC2(N0)
NV0(2)=NE02(N0)
NV(3)=NEC3(N0)
NV0(3)=NE03(N0)
NV(4)=NEC4(N0)
NV0(4)=NE04(N0)
NV(5)=NEC5(N0)
NV0(5)=NE05(N0)
NV(6)=NEC6(N0)
NV0(6)=NE06(N0)
SUM=0.000
DO 55 I=1,6
  N=NV(I)
  IF(N.EQ.0) GOTO 56
  KK=NV0(I)
  NV1=NN1(N)
  NV2=NN2(N)
  NV3=NN3(N)
  NV4=NN4(N)
  NV5=NN5(N)
  NV6=NN6(N)
  UP(1)=U(NV1)
  UP(2)=U(NV2)
  UP(3)=U(NV3)
  UP(4)=U(NV4)
  UP(5)=U(NV5)
  UP(6)=U(NV6)
  VP(1)=V(NV1)
  VP(2)=V(NV2)
  VP(3)=V(NV3)
```

```
VP(4)=V(NV4)
VP(5)=V(NV5)
VP(6)=V(NV6)
UP0(1)=U0(NV1)
UP0(2)=UQ(NV2)
UP0(3)=U0(NV3)
UP0(4)=U0(NV4)
UP0(5)=U0(NV5)
UP0(6)=U0(NV6)
VPO(1)=V0(NV1)
VPO(2)=V0(NV2)
VPO(3)=V0(NV3)
VPO(4)=V0(NV4)
VPO(5)=V0(NV5)
VPO(6)=V0(NV6)
BB1=B1(N)
BB2=B2(N)
BB3=B3(N)
CC1=C1(N)
CC2=C2(N)
CC3=C3(N)
DA=DELT(N)
CALL SUVT(BB1,BB2,BB3,UP,KK,SU)
CALL SUVT(BB1,BB2,BB3,UP0,KK,SU0)
CALL SUVT(CC1,CC2,CC3,VP,KK,SV)
CALL SUVT(CC1,CC2,CC3,VPO,KK,SV0)
99 DO 65 IC=1,6
   TT(IC)=0.
65 TDT(IC)=0.
   GOTO(18,22,24,26,28,32),KK
18 TT(1)=DA/30.
```

```
TT(2)=-DA/180.  
TT(3)=TT(2)  
TT(5)=-DA/45.  
GOTO 70  
22 TT(1)=-DA/180.  
TT(2)=-6.*TT(1)  
TT(3)=TT(1)  
TT(6)=-DA/45.  
GOTO 70  
24 TT(1)=-DA/180.  
TT(2)=TT(1)  
TT(3)=-6.*TT(1)  
TT(4)=-DA/45.  
GOTO 70  
26 TT(3)=-DA/45.  
TT(4)=DA*8./45.  
TT(5)=TT(4)/2.  
TT(6)=TT(4)/2.  
34 TDT(1)=C1(N)**2/(3.*DA)  
TDT(2)=C2(N)**2/(3.*DA)  
TDT(3)=C3(N)**2/(3.*DA)  
TDT(4)=C1(N)*C2(N)*2./(3.*DA)  
TDT(5)=C2(N)*C3(N)*2./(3.*DA)  
TDT(6)=C3(N)*C1(N)*2./(3.*DA)  
GOTO 70  
28 TT(1)=-DA/45.  
TT(4)=-4.*TT(1)  
TT(5)=2.*TT(4)  
TT(6)=TT(4)  
GOTO 34
```

```

32 TT(2)=-DA/45.
   TT(4)=-4.*TT(2)
   TT(5)=TT(4)
   TT(6)=2.*TT(4)
   GOTO 34
70 CONTINUE
   DO 60 M=1,6
   SUM=SUM+TT(M)*(UP(M)-UPO(M))+DT*((SUC(SV(M))*UP(M)+(SUO(M)
1+SV0(M))*UPO(M)-TDT(M)*(UP(M)+UPO(M)))/2.
60 CONTINUE
55 CONTINUE
56 SLVT3=SUM
   RETURN
   END
   SUBROUTINE SUVT(P,Q,R,T,KK,A)
C *****
C
C   TO COMPUTE THE NONLINEAR TERMS IN THE EQUATION USING
C   EXACT INTEGRATION
C *****
C
   DIMENSION T(6),A(6)
   DOUBLE PRECISION E1,E2,E3,T,P,Q,R,A
   GOTO(1,7,3,7,7,6),KK
1  E1=99.*T(1)-2.*(T(2)+T(3)+T(5))+6.*(T(4)+T(6))
   E2=T(3)+6.*(T(1)-2.*T(5))-4.*(T(2)+2.*T(4)+T(6))
   A(1)=P*(13.*T(1)/120.-(T(2)+T(3))/280.+2.*(T(4)+T(6))/105.+T(5)
1/210.)
   A(2)=Q*(-(T(2)+2.*T(1))/280.+(11.*T(3)-20.*T(5))/2520.-2.*
1(T(6)+2.*T(4))/315.)
   A(3)=R*(-(T(3)+2.*T(1))/280.+(11.*T(2)-20.*T(5))/2520.-2.*

```

$$1(T(4)+2.*T(6))/315.)$$

$$A(4)=Q*E1/315.+P*E2/630.$$

$$E3=6.*(T(1)-2.*T(5))-4.*(T(3)+T(4)+2.*T(6))+T(2)$$

$$A(5)=R*E2/630.+Q*E3/630.$$

$$A(6)=R*E1/315.+P*E3/630.$$

RETURN

$$3 E1=6.*(T(3)-2.*T(4))-4.*(T(1)+T(5)+2.*T(6))+T(2)$$

$$E2=6.*(T(3)-2.*T(4))-4.*(T(6)+T(2)+2.*T(5))+T(1)$$

$$E3=99.*T(3)-2.*(T(1)+T(2)+T(4))+6.*(T(5)+T(6))$$

$$A(1)=P*(-(T(1)+2.*T(3))/280.+(11.*T(2)-20.*T(4))/2520.-2.*1(T(5)+2.*T(6))/315.)$$

$$A(2)=Q*(-(T(2)+2.*T(3))/280.+(11.*T(1)-20.*T(4))/2520.-2.*1(T(6)+2.*T(5))/315.)$$

$$A(3)=R*(-(T(1)+T(2))/280.+13.*T(3)/120.+2.*(T(5)+T(6))/105.1+T(4)/210.)$$

$$A(4)=Q*E1/630.+P*E2/630.$$

$$A(5)=R*E2/630.+Q*E3/315.$$

$$A(6)=R*E1/630.+P*E3/315.$$

RETURN

$$6 E1=3.*(2.*T(1)-2.*T(2)+8.*T(4)+16.*T(6))+4.*(4.*T(5)-T(3))$$

$$E2=-2.*(T(1)+T(2)+T(3))+16.*(T(4)+T(5)+T(6))$$

$$E3=3.*(-2.*T(2)+2.*T(3)+8.*T(5)+16.*T(6))+4.*(4.*T(4)-T(1))$$

$$A(1)=P*(20.*(T(4)+2.*T(6))+12.*T(1)-5.*T(2)+4.*(T(5)-2.*T(3)))/630.$$

$$A(2)=Q*(-24.*T(6)+3.*T(2)+4.*(T(4)+T(5)-T(3)-T(1)))/630.$$

$$A(3)=R*(20.*(T(5)+2.*T(6))+12.*T(3)-5.*T(2)+4.*(T(4)-2.*T(1)))/630.$$

$$A(4)=Q*E1/315.+P*E2/315.$$

$$A(5)=R*E2/315.+Q*E3/315.$$

$$A(6)=R*E1/315.+P*E3/315.$$

7 RETURN

END

SUBROUTINE SOLV(Y,X,XO,U)

C

C

C

TO SOLVE AT X=1.25

C

C

DOUBLE PRECISION Y,X,XO,U, A,B,C,D,E,F,G,H,A1,A2,A3,A4,A6,A8
1,V

DIMENSION Y(105),X(30),XO(41),U(42)

1,V(42)

NF1=10

NF2=2*NF1

U(1)=0.000

V(1)=0.000

U(21)=1.000

XO(39)=U(21)

KK=2

DO 25 I=1,NF1

KLL=I+NF2

KOO=KK+2*NF1-1

U(KK)=X(KLL)

XO(KOO)=U(KK)

25 KK=KK+2

DO 35 I=1,NF2

35 V(I+1)=X(I)

KC=1

NFF=NF1-1

DO 5 KN=1,NFF

KNN=KN*10+5

```

N=KC+2
K00=N+2*NF1-1
A=(U(N-2)-U(N-1))/(Y(KNN-10)**2-Y(KNN-5)**2)
B=(U(N-1)-U(N+1))/((Y(KNN-10)+Y(KNN-5))*(Y(KNN-5)-Y(KNN+5)))
C=1.-(Y(KNN-5)+Y(KNN+5))/(Y(KNN-10)+Y(KNN-5))
D=(U(N-1)-U(N+1))/(Y(KNN-5)-Y(KNN+5))
E=Y(KNN-5)+Y(KNN+5)
A3=(A-B)/C
A2=D-A3*E
A1=U(N-2)-Y(KNN-10)*A2-(Y(KNN-10)**2)*A3
U(N)=A1+Y(KNN)*A2+A3*Y(KNN)**2
X0(K00)=U(N)
5 KC=KC+2
U(22)=0.000
U(42)=1.00
V(22)=0.000
NP=NF2+2
NOF=2*NF1-1
DO 10 I=1,NOF
NN=NP+I
N=NN-NF2
NK2=N*5-14
KNN=N*5
A=(U(N-2)-U(N-1))/(Y(KNN-10)**2-Y(KNN-5)**2)
F=(V(N-2)-V(N-1))/(Y(KNN-10)**2-Y(KNN-5)**2)
B=(U(N-1)-U(N))/(Y(KNN-10)+Y(KNN-5))*(Y(KNN-5)-Y(KNN))
G=(V(N-1)-V(N))/(Y(KNN-10)+Y(KNN-5))*(Y(KNN-5)-Y(KNN))
C=1.-(Y(KNN-5)+Y(KNN))/(Y(KNN-10)+Y(KNN-5))
D=(U(N-1)-U(N))/(Y(KNN-5)-Y(KNN))
H=(V(N-1)-V(N))/(Y(KNN-5)-Y(KNN))

```



```

E=Y(KNN-5)+Y(KNN)
A3=(A-B)/C
A6=(F-G)/C
A2=D-A3*E
A4=H-A6*E
A1=U(N-2)-Y(KNN-10)*A2-(Y(KNN-10)**2)*A3
A8=V(N-2)-Y(KNN-10)*A4-(Y(KNN-10)**2)*A6
K=NN
U(K)=A1+Y(NK2+5)*A2+A3*Y(NK2+5)**2
V(K)=A8+Y(NK2+5)*A4+A6*Y(NK2+5)**2
10 CONTINUE
WRITE(108,42) ( U(I),I=1,42)
WRITE(108,42) ( V(I),I=1,42)
42 FORMAT(E16.8)
RETURN
END
SUBROUTINE ZSYST2(F,EPS,NSIG,N,X,X0,ITMAX,IER,DELT,B1,B2,B3,
1C1,C2,C3,DT,XB,NF2)
C *****
C TO SOLVE THE NONLINEAR EQUATIONS
C FOR THE SUDDENLY ACCELERATED PLATE USING PRIMITIVE
C VARIABLES AND QUADRATIC ELEMENT
C *****
DOUBLE PRECISION DT
DOUBLE PRECISION X0,B1,B2,B3,C1,C2,C3,DELT,XB
1,YV
DOUBLE PRECISION EPS,DELTA,XTEMP,PREC,ETA,TOP,RELCON,F,FMAX,
1 PT1,P2,HOLD,H,FPLUS,DERMAX,TEST,X,WA,E,
2 ZERO,PM1
COMMON/SLVT/ NN1(40),NN2(40),NN3(40),NN4(40),NN5(40),NN6(40),
1NEC1(105),NEC2(105),NEC3(105),NEC4(105),NEC5(105),NEC6(105),

```

```

1NE01(105),NE02(105),NE03(105),NE04(105),NE05(105),NE06(105)
1,YY(105)
  DIMENSION X(1),B1(40),B2(40),B3(40),C1(40),C2(40),C3(40),WA(584)
1DELTA(40),X0(1),XB(1)
  DATA          PREC,DELTA/5.0-14,5.0-10/
  DATA          ZERO,PM1,PT1,P2/0.00,.100,.000100,.00200/
  IER=0
  N2 = N+N
  RELCON=10.0+0**(-NSIG)
  JTEST = 1
  IERROR=0
  IPART=((N+2)*(N-1))/2
  ITMP=IPART+N
  LKSUB=ITMP+N
  DO 130 M = 1, ITMAX
    IQUIT=0
    FMAX=ZERO
    M1 = M-1
    K1 = LKSUB + 1
    KMIN = LKSUB + N
    XTEMP = ZERO
    DO 5 J = K1,KMIN
      XTEMP = XTEMP+1.00
      WA(J) = XTEMP
5    CONTINUE
    K = 1
10   IF(K .LE. 1) GO TO 30
    KMIN = K-1
    KK = 1
15   DO 25 K1=1,KMIN

```

```

      ISUB=K-K1
      MM=((ISUB-1)*(N2 -ISUB))/2
      LIM=N-ISUB
      KPOINT = WA(LKSUB+ISUB)+PM1
      ISUB1 = ISUB-1
      X(KPOINT)=ZERO
      DO 20 L1=1,LIM
          JS1=ISUB1+L1
          LKJSUB=LKSUB + JS1 +1
          IJ=MM+JS1
          JPOINT= WA(LKJSUB) + PM1
          X(KPOINT)=X(KPOINT) + WA(IJ)*X(JPOINT)
20      CONTINUE
          X(KPOINT)=X(KPOINT) + WA(MM+N)
25      CONTINUE
      GO TO (30,45,105), KK
30      E=F(X,X0,K,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
      FMAX= DMAX1(FMAX,DABS(E))
      IF(DABS(E) .GE. EPS) GO TO 35
      IQUIT=IQUIT+1
      IF(IQUIT .EQ. N) GO TO 140
35      I = K
40      IP=IPART+I
      ITEMP = WA(LKSUB+I) + PM1
      HOLD = X(ITEMP)
      ETA=.0010+00*DABS(HOLD)
      IF(DABS(HOLD) .LT. PREC) ETA=DELTA
      H = DMIN1(FMAX,ETA)
      IF(H .LT. PREC) H=PREC
      X (ITEMP)=HOLD+H
      IF (K .LE. 1) GO TO 45

```

```
      KK = 2
      GO TO 15
45    FPLUS=F(X,X0,K,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
      TOP=FPLUS-E
      IF (DABS(TOP) .GE. PREC) GO TO 50
      WA(IP)=ZERO
      GO TO 55
50    WA(IP)=TOP/H
55    X(ITEMP)=HOLD
      I = I + 1
      IF (I .LE. N) GO TO 40
      IF (K .LT. N) GO TO 60
      IP=IPART+N
      IF (DABS( WA(IP)) .EQ. ZERO) GO TO 80
      X(ITEMP) = -E/ WA(IP) + X(ITEMP)
      GO TO 100
60    KL=LKSUB+K
      LOOK= WA(KL) + PM1
      KMAX=LOOK
      IP=IPART+K
      DERMAY=DABS( WA(IP))
      KPLUS = K+1
      DO 65 I = KPLUS,N
          TEST=DABS( WA(IPART+I))
          IF (TEST .LE. DERMAY) GO TO 65
          DERMAY = TEST
          KMAX=I
65    CONTINUE
      IF (LOOK .EQ. KMAX) GO TO 75
      LKMAX=LKSUB+KMAX
```

```

WA(KL)=WA(LKMAX)
WA(LKMAX)=LOOK
IP=IPART+KMAX
XTEMP= WA(IP)
IPK=IPART+K
WA(IP)=WA(IPK)
WA(IPK)=XTEMP
IF(K .LT. 2) GO TO 75
KMIN=K-1
I1 = 0
DO 70 I=1,KMIN
  L=((I1 )*(N2 -I))/2-1
  J=L+KMAX
  XTEMP= WA(J)
  JJ=L+K
  WA(J)=WA(JJ)
  WA(JJ)=XTEMP
  I1 = I
70 CONTINUE
75 IF (DABS(WA(IPART+K)).NE. ZERO) GO TO 90
80 IF(IERROR .EQ. 1) GO TO 135
DO 85 I=1,N
  X(I) = 0.900*X(I) + .1234500
85 CONTINUE
IERROR=1
GO TO 105
90 L=((K-1)*(N2 -K))/2
KN=L+N
I1 = L-1
WA(KN)=ZERO
IPK=IPART+K

```

```

DO 95 J = KPLUS,N
  JSUB= WA(LKSUB+J) + PM1
  JJ=I1+J
  IPJ=IPART+J
  WA(JJ)=-WA(IPJ)/WA(IPK)
  WA(KN)=WA(KN)+WA(IPJ)*X(JSUB)
95  CONTINUE
  LK= WA(LKSUB+K) + PM1
  WA(KN)=(WA(KN)-E)/WA(IPK) + X(LK)
  K = K+1
  IF (K .LE. N) GO TO 10
100 IF ( N .EQ. 1) GO TO 105
  KMIN=N-1
  KK = 3
  GO TO 15
105 IF (M .LE. 1) GO TO 120
  DO 110 I = 1,N
    IF(DABS(WA(ITMP+I)-X(I)) .GT. DABS(X(I))*RELCON) GO TO 115
110  CONTINUE
  JTEST = JTEST+1
  IF (JTEST-3)120,140,140
115  JTEST = 1
120  DO 125 I = 1,N
    WA(ITMP+I)=X(I)
125  CONTINUE
130  CONTINUE
  M=ITMAX
  IER = 129
  GO TO 240
135 IER = 130

```

```
140 FMAX=ZERO
    TEST=1.00+15
    IF (N .GT. 1) GO TO 145
    WA(IPART+2)=F(X,X0,1,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
    FMAX = DMAX1(FMAX,DABS(WA(IPART+2)))
    GO TO 155
145 DO 150 I = 1,N
    IP=IPART+I
    WA(IP)=F(X,X0,I,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
    FMAX=DMAX1(FMAX,DABS(WA(IP)))
150 CONTINUE
155 K=1
    DO 160 I=1,N
    WA(I)=X(I)
    IF(DABS(X(I)) .GT. P2) GO TO 160
    K=2
    WA(I)=ZERO
160 CONTINUE
    IF(K .EQ. 1) GO TO 195
    KK = 1
    GO TO 205
165 IF(FMAX .LT. TEST) GO TO 190
    DO 170 I=1,N
    X(I)= WA(I)
170 CONTINUE
    IF (N .GT. 1) GO TO 175
    WA(IPART+2) = WA(ITMP+2)
    GO TO 185
175 DO 180 I = 1,N
    WA(IPART+I) = WA(ITMP+I)
180 CONTINUE
```

```
185 FMAX=TEST
190 K=1
195 ITEST=0
    DO 200 I=1,N
        WA(I)=X(I)
        IF(DABS(X(I)) .LE. P2) GO TO 200
        X(I)=X(I)+PT1
        J=X(I)-PT1
        IF(I .EQ. J) GO TO 200
        WA(I) =ISIGN(1,J)*MAX0(IABS(I),IABS(J))
        K=2
200 CONTINUE
    IF(K .EQ. 1) GO TO 235
    KK = 2
205 TEST=ZERO
    IF (N .GT. 1) GO TO 210
    WA(ITMP+2)=F(WA,X0,1,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
    TEST = DMAX1(TEST,DABS(WA(ITMP+2)))
    GO TO 220
210 DO 215 I=1,N
    IT=ITMP+I
    WA(IT)=F(WA,X0,I,B1,B2,B3,C1,C2,C3,DELT,DT,XB)
    TEST=DMAX1(TEST,DABS(WA(IT)))
215 CONTINUE
220 GO TO (165,225), KK
225 IF(FMAX .LT. TEST) GO TO 235
250 DO 230 I=1,N
    X(I)= WA(I)
230 CONTINUE
260 ITEST=1
```



```
235 IF(FMAX .LT. EPS .OR. TEST .LT. EPS) IER = 0
240 ITMAX=M1 + 1
9000 CONTINUE
270 IF(IER.EQ.0) GOTO 9005
    WRITE(108,88)
    88 FORMAT(5X,*MATRIX IS ILL CONDITIONED*)
9005 RETURN
END
```

C.4.3. UNSTEADY FLOW WITH STREAM FUNCTION FORMULATION

```

C *****
C *
C *MAIN PROGRAM TO SOLVE THE SUDDENLY ACCELERATED PLATE *
C *USING STREAM FUNCTION FORMULATION AND HIGH PRECISION *
C *18 DEGREES OF FREEDOM ELEMENT *
C *
C *****
C MAIN CALLS
  EXTERNAL SLVT4
  DOUBLE PRECISION X,EPS
  COMMON/SLVT/NN1(12),NN2(12),NN3(12),NEC1(14),NEC2(14),NEC3(14),
  1NEO1(14),NEO2(14),NEO3(14),
  1RIJST(18,108),B1(12),B2(12),B3(12),C1(12),C2(12),C3(12),AREA(12
  1,Q(12,378),RLST(18,108)
  1,PS(16),PSX(16),PSY(16),PSXX(16),PSXY(16),PSYY(16),PS0(16),
  1PSX0(16),PSY0(16),PSXX0(16),PSXY0(16),PSYY0(16),NE
  DIMENSION X(40),X0(40),XB(80),Y(7)
  1,A(21,18)
C NE IS THE NUMBER OF ELEMENT
  READ(105,5)NE
5  FORMAT(I5)
  NEE=3*NE/2
C NC IS THE NUMBER OF THE EDGE COORDINATES
  ICCU=0
  DO 30 I=1,NE
    NP=1
    NPP=12
    DO 25 K=1,32

```

```
      READ(104,35) (Q(I,K),K=NP,NPP)
      NP=NPP+1
      NPP=NPP+12
25  CONTINUE

30  CONTINUE
      DO 45 I=1,NEE
      NP=1
      NPP=12
      DO 40 K=1,9
      READ(104,35) (RIJST(I,J),J=NP,NPP)
      NP=NPP+1
      NPP=NPP+12
40  CONTINUE
45  CONTINUE
      DO 55 I=1,NEE
      NP=1
      NPP=12
      DO 50 K=1,9
      READ(104,35) (RLST(I,J),J=NP,NPP)
      NP=NPP+1
      NPP=NPP+12
50  CONTINUE
55  CONTINUE
      DO 66 I=1,NEE
      NP=1
      NPP=12
      DO 60 K=1,9
      READ (104,35) (RKST(I,J),J=NP,NPP)
      NP=NPP+1
      NPP=NPP+12
60  CONTINUE
66  CONTINUE
35  FORMAT(12E10.4)
      NE=10
```

```

      NC=(NE/2)+1
      READ(105,10) (Y(I),I=1,NC)
      WRITE(108,5) NE
      WRITE(108,10) (Y(I),I=1,NC)
10  FORMAT(7F10.5)
      CABL GRIDT3(AREA,NN1,NN2,NN3,NEC1,NEC2,NEC3,NEO1,NEO2,NEO3,B1,B
1B3,C1,C2,C3,NE,Y)
      NF=6*NC -8
      NR=NF
      NF1=NF+1
      NFF=2*NF
      READ(105,65) (X(I),I=1,NF)
      READ(105,65) (XO(I),I=1,NF)
      READ(105,65) (XB(I),I=1,NF)
      READ(105,65) (XB(I),I=NF1,NFF)
65  FORMAT(8F10.5)
      DO 11 I=1,2
      PS(I)=0.
      PSO(I)=0.
      PSX(I)=0.
      PSXO(I)=0.
      PSY(I)=0.
      PSYO(I)=0.
      PSXY(I)=0.
      PSXYO(I)=0.
      PSXX(I)=0.
      PSXXO(I)=0.
      NEF=I+NE
      PSYY(NEF)=0.
      PSYYO(NEF)=0.
      PSXY(NEF)=0.

```

```

    PSXYO(NEF)=0.
    PSY(NEF)=1.
    PSYO(NEF)=1.
11  CONTINUE
    PSYY(1)=X(1)
    NEN=NE+1
C   NEN IS THE NODE NUMBER AT LEFT EDGE
    PSXX(NEN)=X(NF)
    PSX(NEN)=X(NF-1)
    PS(NEN)=X(NF-2)
    PSYYO(1)=XO(1)
    PSXXO(NEN)=XO(NF)
    PSXO(NEN)=XO(NF-1)
    PSO(NEN)=XO(NF-2)
    PSYY(2)=XB(1)
    NEN1=NEN+1
C   NEN1 IS THE NODE NUMBER AT RIGHT EDGE
    PSXX(NEN1)=XB(NF)
    PSX(NEN1)=XB(NF-1)
    PS(NEN1)=XB(NF-2)
    PSYYO(2)=XB(NF+1)
    PSXXO(NEN1)=XB(2*NF)
    PSXO(NEN1)=XB(2*NF-1)
    PSO(NEN1)=XB(2*NF-2)
    NCC=NC-2
    NK=3
    NK1=4
    N1=2
    DO 75 I=1,NCC
    PS(NK)=X(N1)

```

```
PSO(NK)=XO(N1)
PS(NK1)=XB(N1)
N11=NF+N1
PSO(NK1)=XB(N11)
PSX(NK) = X(N1+1)
PSXO(NK)= XO(N1+1)
PSX(NK1)= XB(N1+1)
PSXO(NK1)=XB(N11+1)
PSY(NK) =X(N1+2)
PSYO(NK) =XO(N1+2)
PSY(NK1) =XB(N1+2)
PSYO(NK1)=XB(N11+2)
PSXX(NK) = X(N1+3)
PSXXO(NK) =XO(N1+3)
PSXX(NK1) =XB(N1+3)
PSXXO(NK1)=XB(N11+3)
PSXY(NK) =X(N1+4)
PSXYO(NK) =XO(N1+4)
PSXY(NK1) =XB(N1+4)
PSXYO(NK1)=XB(N11+4)
PSYY(NK) =X(N1+5)
PSYYO(NK) =XO(N1+5)
PSYY(NK1) =XB(N1+5)
PSYYO(NK1)=XB(N11+5)
NK=NK+2
NK1=NK1+2
N1=N1+6
75 CONTINUE
IF(ICCU.EQ.1) GOTO 68
NSIG=4
EPS=4.00-4
```

```

ITMAX=12
T=1.
DT=.2
T=T+DT
WRITE(108,98) T
98 FORMAT(10X,'TIME T =',E16.6)
CALL ZSYST3(SLVT4,EPS,NSIG,NR,X,X0,XB,DT,ITMAX,IER)
WRITE(108,67) ITMAX,(X(I),I=1,NR)
67 FORMAT(5X,'ITMAX =',I3,'/',5X,'X(1)-----X(NF)',/,(' ',8E13.
1,/,10X,'SUCCESFUL CONVERGENCE ',/))
WRITE(108,70)
70 FORMAT(/,2X,'NODE NO. ',2X,'Y ',2X,'PS ',2X,'PSX ',2X,
1X,'PSY ',2X,'PSXX ',2X,'PSXY ',2X,'PSYY ',/))
ICCU=1
GOTO 11
68 LM=1
DO 69 I=1,NC
WRITE(108,71) LM,Y(I),PS(LM),PSX(LM),PSY(LM),PSXX(LM),PSXY(LM),
1PSYY(LM)
LM=LM+2
69 CONTINUE
71 FORMAT(4X,I2,E10.4,6E12.5)
STOP
END

SUBROUTINE GRIDT3(AREA,NN1,NN2,NN3,NEC1,NEC2,NEC3,NEO1,NEO2,NEO
1B1,B2,B3,C1,C2,C3,NE,Y)
C *****
C
C DEFINE THE GRID PROPERTIES , THE ELEMENT-NODE CONNECTIONS.
C FOR THE SUDDENLY ACCELERATED PLATE USING STREAM
C FUNCTION FORMULATION AND QUINTIC ELEMENT
C *****
C
C DIMENSION XX(14),YY(14),NN1(12),NN2(12),NN3(12),B1(12),B2(12),
1B3(12),C1(12),C2(12),C3(12),AREA(12),Y(7)

```

```

DIMENSION NEC1(14),NEC2(14),NEC3(14),NEO1(14),NEO2(14),NEO3(14)
C  NE IS THE NUMBER OF ELEMENT
C  NC IS THE NUMBER OF THE EDGE COORDINATES
NC=(NE/2)+1
K=1
NCC= NC+NC
DO 15 I=1,NC
  YY(K)=Y(I)
  KK=K+1
  YY(KK)=YY(K)*SORT(1.5)
  XX(K)=1.
  XX(KK)=1.5
  K=K+2
15 CONTINUE
  WRITE( 3,25)
  DO 20 N=1,NCC
    WRITE( 3,21) N,XX(N),YY(N)
20 CONTINUE
21 FORMAT(5X,I5,9X,E16.8,9X,E16.8)
25 FORMAT(7X,*NODE*,16X,*X*,22X,*Y*,/)
NEE=NE/2
I1=1
I2=2
I3=3
J1=3
J2=2
J3=4
N=1
DO 30 I=1,NEE
  NN1(N)=I1
  NN2(N)=I2
  NN3(N)=I3
  K=N+1
  NN1(K)=J1
  NN2(K)=J2

```


NN3(K)=J3

N=N+2

I1=I1+2

I2=I2+2

I3=I3+2

J1=J1+2

J2=J2+2

J3=J3+2

30 CONTINUE

DO 35 I=1,NE

NV1=NN1(I)

NV2=NN2(I)

NV3=NN3(I)

B1(I)= YY(NV2)-YY(NV3)

B2(I)= YY(NV3)-YY(NV1)

B3(I)= YY(NV1)-YY(NV2)

C1(I)= XX(NV3)-XX(NV2)

C2(I)= XX(NV1)-XX(NV3)

C3(I)= XX(NV2)-XX(NV1)

AREA(I)=(XX(NV2)*YY(NV3)-XX(NV3)*YY(NV2)-XX(NV1)*(YY(NV3)-YY(NV2)
1+YY(NV1)*(XX(NV3)-XX(NV2)))/2.

35 CONTINUE

WRITE(3,40)

40 FORMAT(/,5X,*ELEMENT NO. *,5X,*NODES CONNECTED WITH IT*,15X,*AREA

DO 46 I=1,NE

WRITE(3,45)I,NN1(I),NN2(I),NN3(I),AREA(I)

45 FORMAT(5X,I5,2X,I5,5X,I5,5X,I5,6X,E16.6)

46 CONTINUE

DO 50 I=1,NCC

NEC1(I)=0

NEC2(I)=0

NEC3(I)=0

NEO1(I)=0

NEO2(I)=0

```

50 NEO3(I)=0
   NU=3
C   NU IS THE NODE NUMBER
   LE=1
C   LE IS THE ELEMENT NUMBER

   NEC1(1)=1
   NEO1(1)=1
   WRITE( 3,65)
   DO 55 I=1,NEE
   NEC1(NU)=NU-2
   NEO1(NU)=3
   NEC2(NU)=NU-1
   NEO2(NU)=1
   IF(I.EQ.NEE)GOTO 55
   NEC3(NU)=NU
   NEO3(NU)=1
   NU=NU+2
55 CONTINUE
   DO 60 I=1,NC
   WRITE( 3,70) LE,NEC1(LE),NEO1(LE),NEC2(LE),NEO2(LE),NEC3(LE),
1NEO3(LE)
60 LE=LE+2
65 FORMAT(3X,*NODE NO.*,*ELEMENT NO-THE ORDER OF THE NODE IN IT*)
70 FORMAT(3X,I3,6X,6I6)
   RETURN
   END
   SUBROUTINE TRANSF(K,CC1,CC2,CC3,BB1,BB2,BB3,A,NE,Q)
C   *****
C
C   CALCULATES THE TRANSFER MATRIX A
C
C   *****
   DIMENSION A(21,18),B(3),C(3)
1,Q(12,378)
   DO 5 I=1,21

```

```
DO 5 J=1,18
5 A(I,J)=0.
  IF(K.GT.1) GOTO 7
DO 6 I=1,12
DO 6 J=1,378
  Q(I,J)=0.
6 CONTINUE
7 CONTINUE
  L=1
  M=2
  N=3
  I=1
  J=1
  KKC=0
  II=19
  JJ=7
  KK=13
  B(1)=881
  B(2)=882
  B(3)=883
  C(1)=CC1
  C(2)=CC2
  C(3)=CC3
15 I1=I+1
  I2=I+2
  I3=I+3
  I4=I+4
  I5=I+5
  J1=J+1
  J2=J+2
  J3=J+3
```

```

J4=J+4
J5=J+5
JJ1=JJ+1
JJ2=JJ+2
JJ3=JJ+3
JJ4=JJ+4
JJ5=JJ+5
KK1=KK+1
KK2=KK+2
KK3=KK+3
KK4=KK+4
KK5=KK+5
A(I,J)=1.
A(I1,J1)=C(N)
A(I1,J2)=-B(N)
A(I2,J1)=-C(M)
A(I2,J2)=B(M)
A(I3,J3)=.5*C(N)**2
A(I3,J4)=-C(N)*B(N)
A(I3,J5)=.5*B(N)**2
A(I4,J3)=-C(M)*C(N)
A(I4,J4)=C(N)*B(M)+B(N)*C(M)
A(I4,J5)=-B(M)*B(N)
A(I5,J3)=.5*C(M)**2
A(I5,J4)=-B(M)*C(M)
A(I5,J5)=.5*B(M)**2
CR=(C(L)**2+B(L)**2+C(M)**2+B(M)**2-C(N)**2-B(N)**2)/(2.*C(L)**2
12.*B(L)**2)
CS=(C(L)**2+B(L)**2-C(M)**2-B(M)**2+C(N)**2+B(N)**2)/(2.*C(L)**2
12.*B(L)**2)

```

```

A(II,JJ)= 30.*CR
A(II,JJ1)=(12.*C(L)-3.*C(N))*CR -3.*CS*(C(L)+C(N))
A(II,JJ2)=(-12.*B(L)+3.*B(N))*CR+3.*CS*(B(L)+B(N))
A(II,JJ3)=(1.5*C(L)**2-C(L)*C(N))*CR-CS*(C(L)**2+C(L)*C(N))
A(II,JJ4)=(-3.*C(L)*B(L)+C(L)*B(N)+B(L)*C(N))*CR+CS*(2.*C(L)*B(L)
1+C(L)*B(N)+B(L)*C(N))
A(II,JJ5)=(1.5*B(L)**2-B(L)*B(N))*CR-CS*(B(L)**2+B(L)*B(N))
A(II,KK)=30.*CS
A(II,KK1)=3.*CR*(C(L)+C(M))+3.*CS*(C(M)-4.*C(L))
A(II,KK2)=-3.*CR*(B(L)+B(M))+3.*CS*(-B(M)+4.*B(L))
A(II,KK3)=-CR*(C(L)*C(M)+C(L)**2)+CS*(-C(L)*C(M)+1.5*C(L)**2)
A(II,KK4)=CR*(C(M)*B(L)+B(M)*C(L)+2.*B(L)*C(L))+CS*(C(M)*B(L)+
1B(M)*C(L)-3.*B(L)*C(L))
A(II,KK5)=-CR*(B(L)*B(M)+B(L)**2)+CS*(-B(L)*B(M)+1.5*B(L)**2)
KKC=KKC+1
IF(KKC.GT.2)GOTO 30
GOTO(20,25),KKC
20 I=7
J=7
L=2
M=3
N=1
II=20
JJ=13
KK=1
GOTO 15
25 I=13
J=13
L=3
M=1
N=2

```

```

      II=21
      JJ=1
      KK=7
      GOTO 15
30  CONTINUE
      M=1
      DO 65 I=1,21
      DO 60 J=1,18
      Q(K,M)=A(I,J)
60  M=M+1
65  CONTINUE
      RETURN
      END

```

```

      SUBROUTINE RIELEM(NK,P1,P2,P3,AREA,IOP,A,NE,R1,R2)

```

```

C      *****

```

```

C
C      CALCULATES THE REMAINING 2/3 OF RI AND RL BY CYCLIC
C      PERMUTATION AND STORE ON DISC THE NECESSARY PARTS
C      OF THE MATRICES AFTER REDUCTION TO 18*18 USING A

```

```

C      *****

```

```

      DIMENSION RI(21,21),RL(21,21),P(3)
      1,A(21,18),R1(18,108),R2(18,108),A(18,21),ATRI(18,21),ATRL(18,21)
      1,ATRIA(18,18),ATRLA(18,18)
      B=.067316017
      C=.0050685425
      D=.1082251082
      E=.0106060606
      F=.0326839826
      G=.0003367003

```

```
H=.001094276
O=.0021765273
PP=.0004689754
Z=.0004148629
Q=.00005772
R=.0001046176
S=.0001394901
T=.0007647907
U=.0106060606
V=.022077922
W=.0032647907
X=.0019480519
Y=.0001563251
BB=.000036075
AA=.000012025
CC=.0000162336
AB=.0595238095
AD=.0047619047
AE=.0003968253
DO 5 KC=1,21
DO 5 KD=1,21
RI(KC,KD)=0.
IF(IOP.EQ.1) GOTO 5
RL(KC,KD)=0.
5 CONTINUE
I=1
J=1
P(1)=P1
P(2)=P2
P(3)=P3
C P1=C1,P2=C2,P3=C3.
```

C IF IOP =1 ,P1=B1,P2=B2,P3=B3 AND RL WILL NOT BE RECALCULATED
 L=1
 M=2
 N=3
 KKC=0
 10 I1=I+1
 I2=I+2
 I3=I+3
 I4=I+4
 I5=I+5
 J1=J+1
 J2=J+2
 J3=J+3
 J4=J+4
 J5=J+5
 RI(I,J)=60.*W*P(L)
 RI(I,J1)=12.*W*P(L)+B*P(M)
 RI(I,J2)=B*P(N)+12.*W*P(L)
 RI(I,J3)=2.*W*P(L)+2.*C*P(M)
 RI(I,J4)=C*(P(M)+P(N))+W*P(L)
 RI(I,J5)=2.*W*P(L)+2.*C*P(N)
 RI(I1,J)=60.*PP*P(L)
 RI(I1,J1)=16.*PP*(P(L)+P(M))
 RI(I1,J2)=8.*PP*(P(L)+2.*P(N))
 RI(I1,J3)=3.*PP*P(L)+4.*Z*P(M)
 RI(I1,J4)=PP*P(L)+Z*(P(M)+2.*P(N))
 RI(I1,J5)=PP*P(L)+2.*Z*P(N)
 RI(I2,J)=RI(I1,J)
 RI(I2,J1)=8.*PP*(P(L)+2.*P(M))
 RI(I2,J2)=16.*PP*(P(L)+P(N))


```

RI(I2,J3)= PP*P(L)+2.*Z*P(M)
RI(I2,J4)= PP*P(L)+Z*(P(N)+2.*P(M))
RI(I2,J5)= 3.*PP*P(L)+4.*Z*P(N)
RI(I3,J) = 100.*BB*P(L)
RI(I3,J1)= BB*(30.*P(L)+23.*P(M))
RI(I3,J2)= BB*(10.*P(L)+23.*P(N))
RI(I3,J3)= 6.*BB*(P(L)+P(M))
RI(I3,J4)= BB*(1.5*(P(L)+2.*P(N))+P(M))
RI(I3,J5)= BB*(P(L)+2.*P(N))
RI(I4,J) = 50.*BB*P(L)
RI(I4,J1)= BB*(10.*P(L)+11.5*P(M))
RI(I4,J2)= BB*(10.*P(L)+11.5*P(N))
RI(I4,J3)= BB*(1.5*P(L)+2.*P(M))
RI(I4,J4)= BB*(P(L)+P(M)+P(N))
RI(I4,J5)= BB*(1.5*P(L)+2.*P(N))
RI(I5,J) = 100.*BB*P(L)
RI(I5,J1)= BB*(10.*P(L)+23.*P(M))
RI(I5,J2)= BB*(30.*P(L)+23.*P(N))
RI(I5,J3)= BB*(P(L)+2.*P(M))
RI(I5,J4)= BB*(P(N)+1.5*(P(L)+2.*P(M)))
RI(I5,J5)= 6.*BB*(P(L)+P(N))
IF(IOP.EQ.1) GOTO 14
RL(I,J) =-36.*AB*P(L)**3
RL(I,J1)=(3.6*P(M)*P(L)**2-6.*P(L)**3)*AB
RL(I,J2)=(3.6*P(N)*P(L)**2-6.*P(L)**3)*AB
RL(I,J3)=(9.*P(M)*P(L)**2+16.2*P(L)*P(M)**2+P(L)**3)*AB
RL(I,J4)=(16.2*P(L)*P(M)*P(N)+.5*P(L)**3+4.5*(P(M)+P(N))*P(L)**2)
1*AB
RL(I,J5)=(16.2*P(L)*P(N)**3+9.*P(M)*P(L)**2+P(L)**3)*AB
RL(I1,J) = -75.*AD*P(L)**3
RL(I1,J1)= (-P(L)+1.5*P(M))*12.*AD*P(L)**2

```

```

RL(I1,J2)= (-P(L)+3.*P(N))*6.*AD*P(L)**2
RL(I1,J3)= (22.5*P(L)*P(M)**2+ 3.*(P(L)+7.*P(M))*P(L)**2)*AD
RL(I1,J4)= (22.5*P(L)*P(M)*P(N)+ (10.5*P(N)+5.25*P(M)+P(L))*P(L)
1**2)*AD
RL(I1,J5)= (22.5*P(L)*P(N)**2+ (10.5*P(N)+P(L))*P(L)**2)*AD
RL(I2,J) = -75.*AD*P(L)**3
RL(I2,J1)= (6.*(3.*P(M)-P(L))*P(L)**2)*AD
RL(I2,J2)= (12.*(-P(L)+1.5*P(N))*P(L)**2)*AD
RL(I2,J3)=(22.5*P(L)*P(M)**2+(10.5*P(M)+P(L))*P(L)**2)*AD
RL(I2,J4)=(22.5*P(L)*P(M)*P(N)+(5.25*P(N)+10.5*P(M)+P(L))*P(L)**2
1 *AD
RL(I2,J5)=(22.5*P(L)*P(N)**2+(21.*P(N)+3.*P(L))*P(L)**2)*AD
RL(I3,J) = -120.*AE*P(L)**3
RL(I3,J1)= (36.*(P(M)-.5*P(L))*P(L)**2)*AE
RL(I3,J2)= (6.*(6.*P(N)-P(L))*P(L)**2)*AE
RL(I3,J3)=(30.*P(L)*P(M)**2+6.*(6.*P(M)+P(L))*P(L)**2)*AE
RL(I3,J4)=(30.*P(L)*P(M)*P(N)+(18.*P(N)+6.*P(M)+1.5*P(L))*P(L)**2
1 *AE
RL(I3,J5)=(30.*P(L)*P(N)**2+(12.*P(N)+P(L))*P(L)**2)*AE
RL(I4,J) = -60.*AE*P(L)**3
RL(I4,J1)= (6.*(-P(L)+3.*P(M))*P(L)**2)*AE
RL(I4,J2)= (6.*(-P(L)+3.*P(N))*P(L)**2)*AE
RL(I4,J3)=(15.*P(L)*P(M)**2+(12.*P(M)+1.5*P(L))*P(L)**2)*AE
RL(I4,J4)=(15.*P(L)*P(M)*P(N)+(6.*P(N)+6.*P(M)+P(L))*P(L)**2)*AE
RL(I4,J5)=(15.*P(L)*P(N)**2+(12.*P(N)+1.5*P(L))*P(L)**2)*AE
RL(I5,J) = -120.*AE*P(L)**3
RL(I5,J1)=(6.*(6.*P(M)-P(L))*P(L)**2)*AE
RL(I5,J2)=(36.*(P(N)-.5*P(L))*P(L)**2)*AE
RL(I5,J3)=(30.*P(L)*P(M)**2+(12.*P(M)+P(L))*P(L)**2)*AE
RL(I5,J4)=(30.*P(L)*P(M)*P(N)+(6.*P(N)+18.*P(M)+1.5*P(L))*P(L)**2

```

```
1 *AE
  RL(I5,J5)=(30.*P(L)*P(N)**2+(36.*P(N)+6.*P(L))*P(L)**2)*AE
14 KKC=KKC+1
  IF(KKC.GT.2)GOTO 25
  GOTO(15,20),KKC
15 L=2
  M=3
  N=1
  J=J+6
  I=I+6
  GOTO 10
20 L=3
  M=1
  N=2
  I=I+6
  J=J+6
  GOTO 10
25 I=1
  J=7
  L=1
  M=2
  N=3
  KKC=0
30 I1=I+1
  I2=I+2
  I3=I+3
  I4=I+4
  I5=I+5
  J1=J+1
  J2=J+2
  J3=J+3
```

```

J4=J+4
J5=J+5
RI(I,J) = D*P(M)
RI(I,J1)=E*(P(M)+P(N))
RI(I,J2)=E*P(L)+F*P(M)
RI(I,J3)=3.*G*(P(M)+P(N))
RI(I,J4)=1.5*G*P(L)+1.5*H*(P(M)+P(N))
RI(I,J5)=3.*H*P(L)+2.*W*P(M)
RI(I1,J) = V*P(M)
RI(I1,J1)= X*P(M)+47.*Q*P(N)
RI(I1,J2)= 47.*Q*P(L)+9.*T*P(M)
RI(I1,J3)= Q*(3.*P(M)+4.*P(N))
RI(I1,J4)= 2.*Q*P(L)+R*(3.*P(M)+4.*P(N))
RI(I1,J5)= 8.*R*P(L)+3.*PP*P(M)
RI(I2,J) = U*P(M)
RI(I2,J1)= T*(P(N)+2.*P(M))
RI(I2,J2)= T*P(L)+47.*Q*P(M)
RI(I2,J3)= Q*(3.*P(M)+2.*P(N))
RI(I2,J4)= Q*P(L)+R*(P(N)+2.*P(M))
RI(I2,J5)= 2.*R*P(L)+PP*P(M)
RI(I3,J) = 90.*BB*P(M)
RI(I3,J1)= BB*(7.2*P(M)+13.*P(N))
RI(I3,J2)= BB*(13.*P(L)+28.8*P(M))
RI(I3,J3)= BB*(.6*P(M)+P(N))
RI(I3,J4)= BB*(.5*P(L)+2.*P(N)+1.2*P(M))
RI(I3,J5)= 4.*BB*(P(L)+1.5*P(M))
RI(I4,J) = 31.5*BB*P(M)
RI(I4,J1)= BB*(2.9*P(N)+4.2*P(M))
RI(I4,J2)= BB*(2.9*P(L)+8.4*P(M))
RI(I4,J3)= CC*P(M)+.4*BB*P(N)

```

```

RI(I4,J4)= .2*BB*(P(L)+2.*P(N)+3.*P(M))
RI(I4,J5)= BB*(.8*P(L)+1.5*P(M))
RI(I5,J) = 28.*BB*P(M)
RI(I5,J1)= 1.6*BB*(P(N)+3.*P(M))
RI(I5,J2)= 1.6*BB*(P(L)+4.*P(M))
RI(I5,J3)= .3*BB*(P(N)+2.*P(M))
RI(I5,J4)= BB*(.15*(P(L)+4.*P(M))+.2*P(N))
RI(I5,J5)= BB*(P(M)+.4*P(L))
IF(IOP.EQ.1) GOTO 34
RL(I,J) =(24.*P(M)**3)*AB
RL(I,J1)=6.*(3.*P(N)+P(M))*AB*P(M)**2
RL(I,J2)=(18.*P(L)+15.6*P(M))*AB*P(M)**2
RL(I,J3)=(P(M)**3+3.*P(M)*P(N)**2+3.*P(N)*P(M)**2)*AB
RL(I,J4)=(3.*P(L)*P(M)*P(N)+1.5*P(L)*P(M)**2+1.5*P(M)**2*(P(M)+3.
1*P(N)))*AB
RL(I,J5)=(3.*P(M)*P(L)**2+(9.*P(L)+5.4*P(M))*P(M)**2)*AB
RL(I1,J) = -30.*AD*P(M)**3
RL(I1,J1)= (36.*P(N)+3.*P(M))*AD*P(M)**2
RL(I1,J2)= (36.*P(L)+9.*P(M))*AD*P(M)**2
RL(I1,J3)= 19.*P(M)*P(N)**2+(P(M)+6.*P(N))*P(M)**2)*AD
RL(I1,J4)=(9.*P(L)*P(M)*P(N)+(3.*P(L)+10.5*P(N)+1.75*P(M))*P(M)**2
1)*AD
RL(I1,J5)=(9.*P(M)*P(L)**2+(21.*P(L)+7.5*P(M))*P(M)**2)*AD
RL(I2,J) =(75.*P(M)**3)*AD
RL(I2,J1)=(18.*(P(M)+1.5*P(M))*P(M)**2)*AD
RL(I2,J2)=((27.*P(L)+30.*P(M))*P(M)**2)*AD
RL(I2,J3)=(3.*P(M)*P(N)**2+(2.*P(N)+P(M))*P(M)**2)*AD
RL(I2,J4)=(3.*P(L)*P(M)*P(N)+(3.*P(L)+5.25*P(N)+3.5*P(M))*P(M)**2
1)*AD
RL(I2,J5)=(3.*P(M)*P(L)**2+(10.5*P(L)+7.5*P(M))*P(M)**2)*AD
RL(I3,J) = -90.*AE*P(M)**3

```

```

RL(I3,J1)= 54.*AE*P(N)*P(M)**2
RL(I3,J2)= 54.*AE*P(L)*P(M)**2
RL(I3,J3)= (18.*P(M)*P(N)**2+(9.*P(N)+P(M))*P(M)**2)*AE
RL(I3,J4)= (18.*P(L)*P(M)*P(N)+(4.5*P(L)+18.*P(N)+2.*P(M))*P(M)**2
1)*AE
RL(I3,J5)=(18.*P(M)*P(L)**2+(36.*P(L)+9.*P(M))*P(M)**2)*AE
RL(I4,J1)=((27.*P(N)+6.*P(M))*P(M)**2)*AE
RL(I4,J2)=((27.*P(L)+12.*P(M))*P(M)**2)*AE
RL(I4,J3)=(6.*P(N)*P(M)**2+(4.5*P(M)+1.5*P(N))*P(N)**2)*AE
RL(I4,J4)=(4.5*P(L)*P(M)*P(N)+(3.*P(L)+6.*P(N)+2.*P(M))*P(M)**2)
1*AE
RL(I4,J5)=(4.5*P(M)*P(L)**2+(12.*P(L)+5.*P(M))*P(M)**2)*AE
RL(I5,J) = 150.*AE*P(M)**3
RL(I5,J1)= (36.*AE*(P(N)+P(M))*P(M)**2)
RL(I5,J2)= (AE*(36.*P(L)+48.*P(M))*P(M)**2)
RL(I5,J3)=(3.*P(M)*P(N)**2+(9.*P(N)+6.*P(M))*P(M)**2)*AE
RL(I5,J4)=(3.*P(L)*P(M)*P(L)+(4.5*P(L)+6.*P(N)+6.*P(M))*P(M)**2)
1 *AE
RL(I5,J5)=(3.*P(M)*P(L)**2+(12.*P(L)+10.*P(M))*P(M)**2)*AE
34 KKC=KKC+1
IF(KKC.GT.2)GOTO 45
GOTO(35,40),KKC
35 L=2
M=3
N=1
I=I+6
J=J+6
GOTO 30
40 L=3
M=1

```

```

N=2
I=I+6
J=1
GOTO 30
45 I=1
J=13
L=1
M=2
N=3
KKC=0
50 T1=I+1
I2=I+2
I3=I+3
I4=I+4
I5=I+5
J1=J+1
J2=J+2
J3=J+3
J4=J+4
J5=J+5
RI(I,J) = D*P(N)
RI(I,J1)= E*P(L)+F*P(N)
RI(I,J2)= E*(P(M)+P(N))
RI(I,J3)= 3.*H*P(L)+2.*H*P(N)
RI(I,J4)= 1.5*G*P(L)+1.5*H*(P(M)+P(N))
RI(I,J5)= 3.*G*(P(M)+P(N))
RI(I1,J) = U*P(N)
RI(I1,J1)= T*P(L)+47.*Q*P(N)
RI(I1,J2)= T*(P(M)+2.*P(N))
RI(I1,J3)= PP*P(N)+2.*R*P(L)
RI(I1,J4)= R*(P(M)+3.*P(N))+Q*P(L)

```

```

RI(I1,J5)= Q*(2.*P(M)+3.*P(N))
RI(I2,J) = V*P(N)
RI(I2,J1)= 9.*T*P(N)+47.*Q*P(L)
RI(I2,J2)= X*P(N)+47.*Q*P(M)
RI(I2,J3)= 3.*PP*P(N)+8.*R*P(L)
RI(I2,J4)= 2.*Q*P(L)+R*(4.*P(M)+3.*P(N))
RI(I2,J5)= Q*(4.*P(M)+3.*P(N))
RI(I3,J) = 28.*BB*P(N)
RI(I3,J1)= 1.6*BB*(P(L)+4.*P(N))
RI(I3,J2)= 1.6*BB*(P(M)+3.*P(N))
RI(I3,J3)= BB*(P(N)+.4*P(L))
RI(I3,J4)= BB*(.2*P(M)+.15*(P(L)+4.*P(N)))
RI(I3,J5)= .3*BB*(P(M)+2.*P(N))
RI(I4,J) = 31.5*BB*P(N)
RI(I4,J1)= BB*(8.4*P(N)+2.9*P(L))
RI(I4,J2)= BB*(2.9*P(M)+4.2*P(N))
RI(I4,J3)= BB*(1.5*P(N)+.8*P(L))
RI(I4,J4)= .2*BB*(P(L)+2.*P(M)+3.*P(N))
RI(I4,J5)= .4*BB* P(M)+CC*P(N)
RI(I5,J) = 90.*BB*P(N)
RI(I5,J1)= BB*(13.*P(L)+28.8*P(N))
RI(I5,J2)= BB*(13.*P(M)+7.2*P(N))
RI(I5,J3)= 4.*BB*(P(L)+1.5*P(N))
RI(I5,J4)= BB*(2.*P(M)+.5*P(L)+1.2*P(N))
RI(I5,J5)= BB*(P(M)+.6*P(N))
IF(IOP.EQ.1) GOTO 54
RL(I,J) = 24.*AB*P(N)**3
RL(I,J1)= (18.*P(L)+15.6*P(N))*AB*P(N)**2.
RL(I,J2)= (18.*P(M)+6.*P(N))*AB*P(N)**2
RL(I,J3)=(3.*P(N)*P(L)**2+(9.*P(L)+5.4*P(N))*P(N)**2)*AB

```


$RL(I, J4) = (3. * P(L) * P(M) * P(N) + 1.5 * (P(L) + P(N) + 3. * P(M)) * P(N) ** 2) * AB$
 $RL(I, J5) = (3. * P(N) * P(M) ** 2 + (P(N) + 3. * P(M)) * P(N) ** 2) * AB$
 $RL(I1, J) = 75. * AD * P(N) ** 3$
 $RL(I1, J1) = (27. * P(L) + 30. * P(N)) * AD * P(N) ** 2$
 $RL(I1, J2) = (27. * P(M) + 18. * P(N)) * AD * P(N) ** 2$
 $RL(I1, J3) = (3. * P(N) * P(L) ** 2 + (10.5 * P(L) + 7.5 * P(N)) * P(N) ** 2) * AD$
 $RL(I1, J4) = (3. * P(L) * P(M) * P(N) + (3. * P(L) + 5.25 * P(M) + 3.5 * P(N)) * P(N) ** 2) * AD$
 $RL(I1, J5) = (3. * P(N) * P(M) ** 2 + (6. * P(M) + 3. * P(N)) * P(N) ** 2) * AD$
 $RL(I2, J) = -30. * AD * P(N) ** 3$
 $RL(I2, J1) = (9. * (P(N) + 4. * P(L)) * P(N) ** 2) * AD$
 $RL(I2, J2) = (3. * (P(N) + 12. * P(M)) * P(N) ** 2) * AD$
 $RL(I2, J3) = (9. * P(N) * P(L) ** 2 + (21. * P(L) + 7.5 * P(N)) * P(N) ** 2) * AD$
 $RL(I2, J4) = (9. * P(L) * P(M) * P(N) + (3. * P(L) + 10.5 * P(M) + 1.75 * P(N)) * P(N) ** 2) * AD$
 $RL(I2, J5) = (9. * P(N) * P(M) ** 2 + (P(N) + 6. * P(M)) * P(N) ** 2) * AD$
 $RL(I3, J) = 150. * AE * P(N) ** 3$
 $RL(I3, J1) = (36. * P(L) + 48. * P(N)) * P(N) ** 2 * AE$
 $RL(I3, J2) = (36. * (P(M) + P(N)) * P(N) ** 2) * AE$
 $RL(I3, J3) = (3. * P(N) * P(L) ** 2 + (12. * P(L) + 10. * P(N)) * P(N) ** 2) * AE$
 $RL(I3, J4) = (3. * P(L) * P(M) * P(N) + (6. * P(M) + 4.5 * P(L) + 6. * P(N)) * P(N) ** 2) * AE$
 $RL(I3, J5) = (3. * P(N) * P(M) ** 2 + (9. * P(M) + 6. * P(N)) * P(N) ** 2) * AE$
 $RL(I4, J1) = ((27. * P(L) + 12. * P(N)) * P(N) ** 2) * AE$
 $RL(I4, J2) = ((27. * P(M) + 6. * P(N)) * P(N) ** 2) * AE$
 $RL(I4, J3) = (4.5 * P(N) * P(L) ** 2 + (12. * P(L) + 5. * P(N)) * P(N) ** 2) * AE$
 $RL(I4, J4) = (4.5 * P(L) * P(M) * P(N) + (6. * P(M) + 3. * P(L) + 2. * P(N)) * P(N) ** 2) * AE$
 $RL(I4, J5) = (4.5 * P(N) * P(M) ** 2 + (6. * P(M) + 1.5 * P(N)) * P(N) ** 2) * AE$
 $RL(I5, J) = -90. * AE * P(N) ** 3$
 $RL(I5, J1) = 54. * AE * P(L) * P(N) ** 2$

RL(I5,J2)= 54.*AE*P(M)*P(N)**2

RL(I5,J3)= (18.*P(N)*P(L)**2+(36.*P(L)+10.*P(N))*P(N)**2)*AE

RL(I5,J4)=(18.*P(L)*P(M)*P(N)+(18.*P(M)+4.5*P(L)+2.*P(N))*P(N)**2

1 *AE

RL(I5,J5)=(18.*P(N)*P(M)**2+(9.*P(M)+P(N))*P(N)**2)*AE

54 KKC=KKC+1

IF(KKC.GT.2)GOTO 65

GOTO(55,60),KKC

55 L=2

M=3

N=1

I=I+6

J=1

GOTO 50

60 L=3

M=1

N=2

I=I+6

J=J+6

GOTO 50

65 I=1

J=19

L=1

M=2

N=3

KKC=0

J1=J+1

J2=J+2

70 I3=I+3

I4=I+4

```

I5=I+5
I1=I+1
I2=I+2
RI(I,J) = G*P(L)+H*(P(N)+P(M))
RI(I,J1)= H*P(L)+O*(P(N)+P(M))
RI(I,J2)= RI(I,J1)
RI(I1,J) = O*P(L)+S*P(M)+2.*R*P(N)
RI(I1,J1)= S*P(L)+Y*(P(M)+2.*P(N))
RI(I1,J2)= 2.*Y*P(M)+PP*P(N)+2.*R*P(L)
RI(I2,J) = Q*P(L)+2.*R*P(M)+S*P(N)
RI(I2,J1)= 2.*R*P(L)+PP*P(M)+2.*Y*P(N)
RI(I2,J2)= Y*(P(N)+2.*P(M))+S*P(L)
RI(I3,J) = .2*BB*(P(L)+2.*P(M)+4.*P(N))
RI(I3,J1)= .2*BB*(5.*P(N)+2.*P(L))+AA*P(M)
RI(I3,J2)= .2*BB*(4.*P(L)+10.*P(N)+5.*P(M))
RI(I4,J) = BB*(.15*P(L)+.4*(P(M)+P(N)))
RI(I4,J1)= BB*(.4*P(L)+.5*P(M))+2.*AA*P(N)
RI(I4,J2)= 2.*AA*P(M)+BB*(.5*P(N)+.4*P(L))
RI(I5,J) = .2*BB*(P(L)+2.*P(N)+4.*P(M))
RI(I5,J1)= .2*BB*(4.*P(L)+10.*P(M)+5.*P(N))
RI(I5,J2)= .2*BB*(5.*P(M)+2.*P(L))+AA*P(N)
IF(IOP.EQ.1) GOTO 74
RL(I,J) =(2.*P(L)*P(M)*P(N)+(P(L)+3.*P(M))*P(N)**2+(P(L)+3.*P(N)
1*P(M)**2)*AB
RL(I,J1)=(5.4*P(M)*P(N)**2+(P(M)+P(N))*P(L)**2+3.*(P(N)**2+2.*P
1*P(N))*P(L))*AB
RL(I,J2)=(5.4*P(N)*P(M)**2+(P(M)+P(N))*P(L)**2+3.*(P(M)**2+2.*P
1*P(N))*P(L))*AB
RL(I1,J) =(4.*P(L)*P(M)*P(N)+(P(L)+3.5*P(N))*P(M)**2+(3.*P(L)+7
1*P(M))*P(N)**2)*AD
RL(I1,J1)=(7.*P(L)*P(M)*P(N)+(P(M)+2.*P(N))*P(L)**2+(7.5*P(M)+7

```

```

1*P(L))*P(N)**2)*AD
  RL(I1,J2)=(14.*P(L)*P(M)*P(N)+(3.5*P(L)+7.5*P(N))*P(M)**2+(2.*P(M)
1+3.*P(N))*P(L)**2)*AD
  RL(I2,J)=(4.*P(L)*P(M)*P(N)+(P(L)+3.5*P(M))*P(N)**2+(3.*P(L)+7.
1*P(N))*P(M)**2)*AD
  RL(I2,J1)=(14.*P(L)*P(M)*P(N)+(3.5*P(L)+7.5*P(M))*P(N)**2+(2.*P(M)
1+3.*P(M))*P(L)**2)*AD
  RL(I2,J2)=(7.*P(L)*P(M)*P(N)+(7.*P(L)+7.5*P(N))*P(M)**2+(P(N)+2.
1*P(M))*P(L)**2)*AD
  RL(I3,J)=(6.*P(L)*P(M)*P(N)+(6.*P(L)+12.*P(M))*P(N)**2+(P(L)+4.
1*P(N))*P(M)**2)*AE
  RL(I3,J1)=(8.*P(L)*P(M)*P(N)+(P(M)+3.*P(N))*P(L)**2+(10.*P(M)+12.
1*P(L))*P(N)**2)*AE
  RL(I3,J2)=(24.*P(L)*P(M)*P(N)+(10.*P(N)+4.*P(L))*P(M)**2+(6.*P(N)
1+3.*P(M))*P(L)**2)*AE
  RL(I4,J)=(4.*P(L)*P(M)*P(N)+(1.5*P(L)+4.*P(M))*P(N)**2+(1.5*P(L)
1+4.*P(N))*P(M)**2)*AE
  RL(I4,J1)=(8.*P(L)*P(M)*P(N)+(1.5*P(M)+2.*P(N))*P(L)**2+(5.*P(M)
1+4.*P(L))*P(N)**2)*AE
  RL(I4,J2)=(8.*P(L)*P(M)*P(N)+(5.*P(N)+4.*P(L))*P(M)**2+(1.5*P(N)
1+2.*P(M))*P(L)**2)*AE
  RL(I5,J)=(6.*P(L)*P(M)*P(N)+(P(L)+4.*P(M))*P(N)**2+(6.*P(L)+12.
1*P(N))*P(M)**2)*AE
  RL(I5,J1)=(24.*P(L)*P(M)*P(N)+(6.*P(M)+3.*P(N))*P(L)**2+(10.*P(M)
1+4.*P(L))*P(N)**2)*AE
  RL(I5,J2)=(8.*P(L)*P(M)*P(N)+(10.*P(N)+12.*P(L))*P(M)**2+(P(N)+3.
1*P(M))*P(L)**2)*AE
74 KKC=KKC+1
  IF(KKC.GT.2)GOTO 85
  GOTO(75,80),KKC

```

75 L=2
M=3
N=1
I=I+6
J=20
J1=21
J2=19
GOTO 70

80 L=3
M=1
N=2
I=I+6
J=21
J1=19
J2=20
GOTO 70

85 KC=21
KB=20
KA=19
I=19
J=1
L=1
~~M=2~~
N=3
KKC=0
K=7
KK=13

90 J1=J+1
J2=J+2
J3=J+3
J4=J+4

J5=J+5

K1=K+1

K2=K+2

K3=K+3

K4=K+4

K5=K+5

KK1=KK+1

KK2=KK+2

KK3=KK+3

KK4=KK+4

KK5=KK+5

RI(I,J) = 21.*BB*P(L)

RI(I,J1)= 4.2*BB*P(L)+5.8*AA*P(M)

RI(I,J2)= 4.2*BB*P(L)+5.8*AA*P(N)

RI(I,J3)= .2*BB*(3.*P(L)+2.*P(M))

RI(I,J4)= .2*BB*(P(M)+P(N))+CC*P(L)

RI(I,J5)= .2*BB*(3.*P(L)+2.*P(N))

RI(I,K) = 30.*BB*P(M)

RI(I,K1)= 13.*AA*P(N)+7.2*BB*P(M)

RI(I,K2)= 13.*AA*P(L)+4.8*BB*P(M)

RI(I,K3)= BB*(1.2*P(M)+P(N))

RI(I,K4)= BB*(.5*P(L)+.6*P(M))+AA*P(N)

RI(I,K5)= 2.*AA*P(L)+.6*BB*P(M)

RI(I,KK) = 30.*BB*P(N)

RI(I,KK1)= 4.8*BB*P(N)+13.*AA*P(L)

RI(I,KK2)= 7.2*BB*P(N)+13.*AA*P(M)

RI(I,KK3)= .6*BB*P(N)+2.*AA*P(L)

RI(I,KK4)= AA*P(M)+BB*(.5*P(L)+.6*P(N))

RI(I,KK5)= BB*(P(M)+1.2*P(N))

RI(I,KA) = .4*BB*(P(L)+P(M)+P(N))

```

RI(I,KB)=BB*(.4*P(L)+.2*P(M)+.3*P(N))
RI(I,KC) = BB*(.4*P(L)+.2*P(N)+.3*P(M))
IF(IOP.EQ.1) GOTO 94
RL(I,J1)= (18.*P(M)+6.*P(L))*AE*P(L)**2
RL(I,J2)= (18.*P(N)+6.*P(L))*AE*P(L)**2
RL(I,J3)= (3.*P(L)*P(M)**2+(6.*P(M)+2.*P(L))*P(L)**2)*AE
RL(I,J4)= (3.*P(L)*P(M)*P(N)+(3.*P(N)+3.*P(M)+1.5*P(L))*P(L)**2)
1 *AE
RL(I,J5)= (3.*P(L)*P(N)**2+(6.*P(N)+2.*P(L))*P(L)**2)*AE
RL(I,K) = -30.*AE*P(M)**3
RL(I,K1)= 18.*AE*P(N)*P(M)**2
RL(I,K2)= 18.*AE*P(L)*P(M)**2
RL(I,K3)= (6.*P(M)*P(N)**2+(9.*P(N)+2.*P(M))*P(M)**2)*AE
RL(I,K4)=(6.*P(L)*P(M)*P(N)+(4.5*P(L)+3.*P(N)+P(M))*P(M)**2)*AE
RL(I,K5)= (6.*P(M)*P(L)**2+(6.*P(L)+P(M))*P(M)**2)*AE
RL(I,KK) = -30.*AE*P(N)**3
RL(I,KK1)= 18.*AE*P(L)*P(N)**2
RL(I,KK2)= 18.*AE*P(M)*P(N)**2
RL(I,KK3)= (6.*P(N)*P(L)**2+(6.*P(L)+P(N))*P(N)**2)*AE
RL(I,KK4)=(6.*P(L)*P(M)*P(N)+(3.*P(M)+4.5*P(L)+P(N))*P(N)**2)*AE
RL(I,KK5)=(6.*P(N)*P(M)**2+(9.*P(M)+2.*P(N))*P(N)**2)*AE
RL(I,KA) =(3.*P(L)*P(M)*P(N)+(P(L)+P(M))*P(N)**2+(P(L)+P(N))
1 *P(M)**2)*2.*AE
RL(I,KB) =(4.*P(L)*P(M)*P(N)+(2.*P(M)+3.*P(N))*P(L)**2+(P(M)+2.
1 *P(L))*P(N)**2)*AE
RL(I,KC) =(4.*P(L)*P(M)*P(N)+(P(N)+2.*P(L))*P(M)**2+(2.*P(N)+3.
1 *P(M))*P(L)**2)*AE
94 KKC=KKC+1
IF(KKC.GT.2) GOTO 105
GOTO(95,100),KKC
95 L=2

```

0

M=3
N=1
I=I+1
J=7
K=13
KK=1
KA=20
KB=21
KC=19
GOTO 90

100 I=I+1
L=3
M=1
N=2
J=13
K=1
KK=7
KA=21
KB=19
KC=20
GOTO 90

105 CD=1./(4.*AREA**2)
IF(IOP.EQ.1) GOTO 115
DO 110 I=1,21
DO 110 J=1,21
RL(I,J)=RL(I,J)*CD

110 CONTINUE
115 CONTINUE
GOTO(120,125,120,125,120,125,120,125,120,125,120,125),NK

120 IC= 3*(NE-NK+1)/2


```

      GOTO 130
125 IC= 3*(NE-NK+2)/2 -2
130 CONTINUE
      KO=(3*NE/2) -IC+1
      DO 136 I=1,IC
      DO 135 J=1,108
      R1(KO,J)=0.
      IF(IOP.EQ.1) GOTO 136
      R2(KO,J)=0.
135 CONTINUE
      KO=KO+1
136 CONTINUE
      DO 140 I=1,18
      DO 140 J=1,21
140 AT(I,J)=A(J,I)
      DO 155 I=1,18
      DO 150 IC=1,21
      SUM1=0.
      SUM2=0.
      DO 145 J=1,21
      SUM1=SUM1+AT(I,J)*RI(J,IC)
      IF(IOP.EQ.1) GOTO 145
      SUM2=SUM2+AT(I,J)*RL(J,IC)
145 CONTINUE
      ATRI(I,IC)=SUM1
      IF(IOP.EQ.1) GOTO 150
      ATRL(I,IC)=SUM2
150 CONTINUE
155 CONTINUE
      DO 160 I=1,18
      DO 160 IC=1,18

```

```
SUM1=0.
SUM2=0.
DO 156 J=1,21
SUM1=SUM1+ATRI(I,J)*A(J,IC)
IF(IOP.EQ.1) GOTO 156
SUM2=SUM2+ATRL(I,J)*A(J,IC)
156 CONTINUE
ATRIA(I,IC)=SUM1
IF(IOP.EQ.1) GOTO 160
ATRLA(I,IC)=SUM2
160 CONTINUE
GOTO(165,170,165,170,165,170,165,170,165,170,165,170),NK
165 IC=3*(NE-NK+1)/2
K0=(3*NE/2)-IC+1
LP=1
DO 167 I=1,6
DO 166 J=1,18
R1(K0,LP)=ATRIA(I,J)
IF(IOP.EQ.1) GOTO 166
R2(K0,LP)=ATRLA(I,J)
166 LP=LP+1
167 CONTINUE
LP=1
II=13
K0=K0+1
DO 169 I=1,6
DO 168 J=1,18
R1(K0,LP)=ATRIA(II,J)
IF(IOP.EQ.1) GOTO 168
R2(K0,LP)=ATRLA(II,J)
```

```

168 LP=LP+1
    II=II+1
169 CONTINUE
    GOTO 175
170 IC=3*(NE-NK+2)/2 -2
    KO=(3*NE/2)-IC+1
    LP=1
    DO 172 I=1,6
    DO 171 J=1,18
    R1(KO,LP)=ATRIA(I,J)
    IF(IOP.EQ.1) GOTO 171
    R2(KO,LP)=ATRLA(I,J)
171 LP=LP+1
172 CONTINUE
175 KP= 3*NE/2
    RETURN
    END

```

```

DOUBLE PRECISION FUNCTION SLVT4(K,X,X0,XB,DT)

```

```

C

```

```

C

```

```

C

```

```

C

```

```

C

```

```

C

```

```

C

```

```

EVALUATES THE RESIDUAL AT NODE NO
FOR THE SUDDENLY ACCELERATED PLATE USING STREAM
FUNCTION FORMULATION AND QUINTIC ELEMENT

```

```

*****
DOUBLE PRECISION X

```

```

COMMON/SLVT/NN1(12),NN2(12),NN3(12),NEC1(14),NEC2(14),NEC3(14),
1NEO1(14),NEO2(14),NEO3(14),
1RIJST(18,108),B1(12),B2(12),B3(12),C1(12),C2(12),C3(12),AREA(12
1,Q(12,378),RLST(18,108)
1,PS(16),PSX(16),PSY(16),PSXX(16),PSXY(16),PSYY(16),PSO(16),

```

```

1PSX0(16),PSY0(16),PSXX0(16),PSXY0(16),PSYY0(16),NE
  DIMENSION X(40),X0(1),XB(1),
1PP(18),PPO(18),NV(3),NVO(3),
1RJRE(18,18),RKRE(18,18),RJORE(18,18),RKORE(18,18)
1,A(21,18)
  NC=(NE/2)+1
  NF=6*NC-8
  NEN=NE+1
  NCC=NC-2
  PSYY(1)=X(1)
  NK=3
  N1=2
  DO 5 I=1,NCC
    PS(NK)=X(N1)
    PSX(NK)=X(N1+1)
    PSY(NK)=X(N1+2)
    PSXX(NK)=X(N1+3)
    PSXY(NK)=X(N1+4)
    PSYY(NK)=X(N1+5)
    NK=NK+2
    N1=N1+6
5 CONTINUE
  PSXX(NEN)=X(NF)
  PSX(NEN)=X(NF-1)
  PS(NEN)=X(NF-2)
  NE4=NE/2
  NE5=NE+NCC
  NE6=NE5+NCC+1
  NE7=NE6+NCC
  IF(K.GT.1) GOTO 10

```

```
NO=1
LPC=6
GOTO 40
10 IF(K.GT.7) GOTO 15
NO=3
LPC=K-1
GOTO 40
15 IF(K.GT.13) GOTO 20
NO=5
LPC=K-7
GOTO 40
20 IF(K.GT.19) GOTO 25
NO=7
ISUB=NF-K+1
IF(ISUB.GT.3) GOTO 24
GOTO(21,22,23),ISUB
24 LPC=4
GOTO 40
22 LPC=2
GOTO 40
23 LPC=1
GOTO 40
24 LPC=K-13
GOTO 40
25 IF(K.GT.25) GOTO 30
NO=9
ISUB=NF-K+1
IF(ISUB.GT.3) GOTO 29
GOTO(26,27,28),ISUB
26 LPC=4
GOTO 40
```

```
27 LPC=2
   GOTO 40
28 LPC=1
   GOTO 40
29 LPC=K-19
   GOTO 40
30 IF(K.GT,31) GOTO 35
   NO=11
   ISUB=NF-K+1
   IF(ISUB.GT.3) GOTO 34
   GOTO(31,32,33),ISUB
31 LPC=4
   GOTO 40
32 LPC=2
   GOTO 40
33 LPC=1
   GOTO 40
34 LPC=K-25
   GOTO 40
35 NO=13
   ISUB=NF-K+1
   GOTO(36,37,38),ISUB
36 LPC=4
   GOTO 40
37 LPC=2
   GOTO 40
38 LPC=1
C   LPC IS THE ORDER OF THE UNKNOWN IN THE NODAL VARIABLES
40 NV(1)=NEC1(NO)
   NVO(1)=NEO1(NO)
```

```

NV(2) = NEC2(N0)
NV0(2) = NEO2(N0)
NV(3) = NEC3(N0)
NV0(3) = NEO3(N0)
SUM = 0.
DO 100 I = 1, 3
  N = NV(I)
  IF(N.EQ.0) GOTO 101
  KK = NV0(I)
C   KK IS THE ORDER OF THE NODE NO IN ELEMENT N
C   N IS THE ELEMENT FOR WHICH THE RESIDUAL WILL BE CALCULATED
C   NN1(N) IS THE FIRST NODE CONSTITUTING ELEMENT N
  NV1 = NN1(N)
  NV2 = NN2(N)
  NV3 = NN3(N)
C   PP IS THE VARIABLE -6 AT EACH NODE
  PP(1) = PS(NV1)
  PP(2) = PSX(NV1)
  PP(3) = PSY(NV1)
  PP(4) = PSXX(NV1)
  PP(5) = PSXY(NV1)
  PP(6) = PSYY(NV1)
  PP0(1) = PS0(NV1)
  PP0(2) = PSX0(NV1)
  PP0(3) = PSY0(NV1)
  PP0(4) = PSXX0(NV1)
  PP0(5) = PSXY0(NV1)
  PP0(6) = PSYY0(NV1)
  PP(7) = PS(NV2)
  PP(8) = PSX(NV2)
  PP(9) = PSY(NV2)

```

```
PP(10)=PSXX(NV2)
PP(11)=PSXY(NV2)
PP(12)=PSYY(NV2)
PP0(7) = PS0(NV2)
PP0(8) = PSX0(NV2)
PP0(9) = PSY0(NV2)
PP0(10)=PSXX0(NV2)
PP0(11)=PSXY0(NV2)
PP0(12)=PSYY0(NV2)
PP(13)= PS(NV3)
PP(14)= PSX(NV3)
PP(15)= PSY(NV3)
PP(16)=PSXX(NV3)
PP(17)=PSXY(NV3)
PP(18)=PSYY(NV3)
PP0(13)= PS0(NV3)
PP0(14)= PSX0(NV3)
PP0(15)= PSY0(NV3)
PP0(16)=PSXX0(NV3)
PP0(17)=PSXY0(NV3)
PP0(18)=PSYY0(NV3)
GOTO(45,48,45,48,45,48,45,48,45,48),N
45 IF(KK.EQ.3) GOTO 46
   KIN=(3*N-1)/2
   GOTO 50
46 KIN=(3*N-1)/2 +1
   GOTO 50
48 KIN=3*N/2
50 IIN=LPC*18 -17
   M=1
```



```

      DO 65 I00=1,21
      DO 61 J=1,18
      A(I00,J)=Q(N,M)
61  M=M+1
65  CONTINUE
      DO 200 I11=1,18
      DO 200 I12=1,18
      RJRE(I11,I12)=0.
      RKRE(I11,I12)=0.
      RJORE(I11,I12)=0.
      RKORE(I11,I12)=0.
200 CONTINUE
      CALL INTE(N,B1,B2,B3,C1,C2,C3,A,AREA,PP,PP0,RJRE,RJORE,
1RKRE,RKORE)
      LPKK=LPC*KK
      DO 60 K2=1,18
      SUM=SUM+ (RIJST(KIN,IIN)*(PP(K2)-PP0(K2))+(DT/2.)*(RJRE(LP
1KK,K2)*PP(K2)+RJORE(LP
1KK,K2)*PP0(K2)-RKRE(LP
1KK,K2)*PP(K2)-RKORE(LP
1KK,K2)*PP0(K2)-RLST(KIN,IIN)*(PP(K2)+PP0(K2))))
      IIN=IIN+1
60  CONTINUE
100 CONTINUE
101 SLVT4=SUM
      RETURN
      END
      SUBROUTINE INTE(N,B1,B2,B3,C1,C2,C3,A,AREA,PP,PP0,RJRE,RJORE,
1RKRE,RKORE)
C      *****
C
C      INTEGRATE THE TERMS F(I)*(SUM(PP(J)*FY(J)))*FYX(K) TO GET RJ
C      INTERCHANGE PP(J) BY PP0(J)

```

C INTEGRATE THE TERMS $F(I) * (SUM(PP(J) * FX(J))) * FYY(K)$ TO GET RK
 C REDUCE THE MATRICES RJ,RK TO RJRE,RKRE USING THE TRANSFER MAT.A

C

C

DIMENSION PP(18),PPO(18),A(21,18),RJRE(18,18),RJOE(18,18),
 1RKRE(18,18),RKORE(18,18),AT(18,21),RJO(21,21),RKO(21,21),ATRJ(18,
 221),ATRJO(18,21),ATRK(18,21),ATRK0(18,21)

DIMENSION C1(12),C2(12),C3(12),B1(12),B2(12),B3(12),AJ(7),H(7),
 1AI(7),AS(7),RJ(21,21),RK(21,21),F(21),FYX(21),FYY(21),Z(3)
 1,AREA(12)

C NN IS THE ELEMENT NO.

AJ(1)=0.0469100770

AJ(2)=0.2307653449

AJ(3)=0.5

AJ(4)=0.7692346551

AJ(5)=0.9530899230

H(1) =0.1184634425

H(2) =0.2393143353

H(3) =0.2844444444

H(4) =0.2393143353

H(5) =0.1184634425

AI(1)=0.198098571

AI(2)=0.1980134179

AI(3)=0.4379748102

AI(4)=0.6954642734

AI(5)=0.9014649142

AS(1)=0.1007941926

AS(2)=0.2084506672

AS(3)=0.2604633916

AS(4)=0.2426935942

```

AS(5)=0.1598203766
DO 95 IP=1,21
DO 95 JP=1,21
RJ(IP,JP)=0.
RK(IP,JP)=0.
RJO(IP,JP)=0.
RKO(IP,JP)=0.
95 CONTINUE
DO 100 IW=1,5
DO 100 JW=1,5
Z(1)=AI(IW)
Z(2)=AJ(JW)*(1.-Z(1))
Z(3)= 1.-Z(1)-Z(2)
W= AS(IW)*H(JW)*(1.-Z(1))
CALL FUNC(C1,C2,C3,B1,B2,B3,N,Z,CY,CX,CY0,CX0,F,FYX,FYY,PP,PP0,A)
CAD=4.*AREA(N)**2
DO 96 IP=1,21
DO 96 JP=1,21
RJ(IP,JP)=RJ(IP,JP)+ W*F(IP)*FYX(JP)*CY
RK(IP,JP)=RK(IP,JP)+ W*F(IP)*FYY(JP)*CX
RJO(IP,JP)=RJO(IP,JP)+ W*F(IP)*FYX(JP)*CY0
RKO(IP,JP)=RKO(IP,JP)+ W*F(IP)*FYY(JP)*CX0
96 CONTINUE
100 CONTINUE
DO 97 IP=1,21
DO 97 JP=1,21
RJ(IP,JP)= RJ(IP,JP)/CAD
RK(IP,JP)= RK(IP,JP)/CAD
RJO(IP,JP)=RJO(IP,JP)/CAD
RKO(IP,JP)=RKO(IP,JP)/CAD
97 CONTINUE

```

```
DO 140 I=1,18
DO 140 J=1,21
140 AT(I,J)=A(J,I)
DO 155 I=1,18
DO 150 IC=1,21
SUM1=0.
SUM2=0.
SUM3=0.
SUM4=0.
DO 145 J=1,21
SUM1= SUM1+AT(I,J)*RJ(J,IC)
SUM2= SUM2+AT(I,J)*RK(J,IC)
SUM3= SUM3+AT(I,J)*RJO(J,IC)
SUM4= SUM4+AT(I,J)*RKO(J,IC)
145 CONTINUE
ATRJ(I,IC)= SUM1
ATRK(I,IC)= SUM2
ATRJO(I,IC)=SUM3
ATRKO(I,IC)=SUM4
150 CONTINUE
155 CONTINUE
DO 160 I=1,18
DO 160 IC=1,18
SUM1=0.
SUM2=0.
SUM3=0.
SUM4=0.
DO 156 J=1,21
SUM1= SUM1+ATRJ(I,J)*A(J,IC)
SUM2= SUM2+ATRK(I,J)*A(J,IC)
```

```

SUM3= SUM3+ATRJO(I,J)*A(J,IC)
SUM4= SUM4+ATRK0(I,J)*A(J,IC)
156 CONTINUE
RJRE(I,IC)=SUM1
RKRE(I,IC)=SUM2
RJOE(I,IC)=SUM3
RKOE(I,IC)=SUM4
160 CONTINUE
RETURN
END
SUBROUTINE FUNC(C1,C2,C3,B1,B2,B3,NN,Z,CY,CX,CY0,CX0,F,FYX,
1FYY,PP,PP0,A)
C *****
C
C EVALUATES THE INTEGRANDS FOR RJ,RJO,RK,RK0
C *****
C
DIMENSION C1(12),C2(12),C3(12),B1(12),B2(12),B3(12),Z(3),F(21),
1FYX(21),FYY(21),PA(21),D(3),C(3),AL(21),
2PP(18),PP0(18),A(21,18),PPA(21),PPA0(21)
DO 170 I=1,21
SUM1=0.
SUM2=0.
DO 165 J=1,18
SUM1= SUM1+A(I,J)*PP(J)
SUM2= SUM2+A(I,J)*PP0(J)
165 CONTINUE
PPA(I)=SUM1
PPA0(I)=SUM2
170 CONTINUE
DO 42 IMM=1,2

```

```

      GOTO(1,2),IMM
1 DO 3 I=1,21
3 AL(I)=PPA(I)
  GOTO 6
2 DO 4 I=1,21
4 AL(I)=PPA0(I)
6 CONTINUE
  DO 40 IK=1,2
    GOTO(5,10),IK
5 D(1)=C1(NN)
  D(2)=C2(NN)
  D(3)=C3(NN)
  GOTO 11
10 D(1)=B1(NN)
  D(2)=B2(NN)
  D(3)=B3(NN)
11 C0=0.
  H1=AL(1)
  H2=AL(2)
  H3=AL(3)
  H4=AL(4)
  H5=AL(5)
  H6=AL(6)
  L=1
  M=2
  N=3
  IKK=0
20 C0=C0+H1*30.*D(L)*(Z(L)**2-Z(L))**2+(D(M)*(4.-3.*Z(L))*Z(L)**3+
  112.*D(L)*Z(M)*(1.-Z(L))*Z(L)**2)*H2+ H3*(D(N)*(4.-3.*Z(L))*Z(L)**3
  1+12.*D(L)*Z(N)*(1.-Z(L))*Z(L)**2)+ H4*(3.*D(L)*(Z(L)*Z(M))**2+ 2.

```

```

1*O(M)*Z(M)*Z(L)**3)+ H5*(D(N)*Z(M)*Z(L)**3+D(M)*Z(N)*Z(L)**3+3.*
1D(L)*Z(M)*Z(N)*Z(L)**2)+ H6*(2.*D(N)*Z(N)*Z(L)**3+3.*D(L)*(Z(L)*
1Z(N))**2)
  IKK=IKK+1
  H1=AL(7)
  H2=AL(8)
  H3=AL(9)
  H4=AL(10)
  H5=AL(11)
  H6=AL(12)
  L=2
  M=3
  N=1
  IF(IKK.GE.2)GOTO 21
  GOTO 20
21 IF(IKK.GT.2) GOTO 22
  H1=AL(13)
  H2=AL(14)
  H3=AL(15)
  H4=AL(16)
  H5=AL(17)
  H6=AL(18)
  L=3
  M=1
  N=2
  GOTO 20
22 C0=C0+AL(19)*(D(1)*(Z(2)*Z(3))**2+ 2.*D(2)*Z(1)*Z(2)*Z(3)**2+ 2.
1*D(3)*Z(1)*Z(3)*Z(2)**2)+ AL(20)*(2.*D(1)*Z(1)*Z(2)*Z(3)**2+D(2)*
1(Z(1)*Z(3))**2+ 2.*D(3)*Z(2)*Z(3)*Z(1)**2) + AL(21)*(2.*D(2)*Z(2)*
1Z(3)*Z(1)**2+ D(3)*(Z(2)*Z(1))**2 +2.*D(1)*Z(1)*Z(3)*Z(2)**2)
  GOTO(25,30),IK

```

```
C   CY IS SUM OF FYJ*AJ
C   CX IS SUM OF FXJ*AJ
25 S1=CO
   GOTO 40
30 S2=CO
40 CONTINUE
   GOTO(31,32),IMM
31 CX=S2
   CY=S1
   GOTO 33
32 CX0=S2
   CY0=S1
33 CONTINUE
42 CONTINUE
   L=1
   M=2
   N=3
   I=1
43 F(I)= 10.*Z(L)**3- 15.*Z(L)**4+ 6.*Z(L)**5
   F(I+1)= Z(M)*(4.*Z(L)**3-3.*Z(L)**4)
   F(I+2)= Z(N)*(4.*Z(L)**3-3.*Z(L)**4)
   F(I+3)= (Z(L)**3)*Z(M)**2
   F(I+4)= Z(M)*Z(N)*Z(L)**3
   F(I+5)= (Z(L)**3)*Z(N)**2
   II=I+1
   IF(II.EQ.2)GOTO 45
   IF(II.EQ.8)GOTO 50
   IF(II.EQ.14)GOTO 55
45 I=7
   L=2
```



```

M=3
N=1
GOTO 43
50 I=13
L=3
M=1
N=2
GOTO 43
55 F(19)= Z(1)*(Z(2)*Z(3))**2
F(20)= Z(2)*(Z(1)*Z(3))**2
F(21)= Z(3)*(Z(1)*Z(2))**2
DO 90 IK=1,2
GOTO(56,57),IK
56 D(1)=B1(NN)
D(2)=B2(NN)
D(3)=B3(NN)
GOTO 58
57 D(1)=C1(NN)
D(2)=C2(NN)
D(3)=C3(NN)
58 L=1
M=2
N=3
I=1
C PA IS EITHER FYX(1ST LOOP)OR FVY(2ND LOOP)
C(1)=C1(NN)
C(2)=C2(NN)
C(3)=C3(NN)
59 PA(I)= 60.*C(L)*D(L)*Z(L)*(2.*Z(L)**2-3.*Z(L)+1.)
PA(I+1)= 12.*(1.-Z(L))*(D(M)*C(L)+ D(L)*C(M))*Z(L)**2+ 12.*D(L)*
1C(L)*Z(L)*Z(M)*(2.-3.*Z(L))

```

```

PA(I+2) = 12.* (1.-Z(L)) * (D(N)*C(L) + D(L)*C(N)) * Z(L)**2 + 12.*D(L)*
1C(L)*Z(L)*Z(N)*(2.-3.*Z(L))
PA(I+3) = 6.*D(L)*C(M)*Z(M)*Z(L)**2 + 6.*D(L)*C(L)*Z(L)*Z(M)**2 + 2.*
1D(M)*C(M)*Z(L)**3 + 6.*D(M)*C(L)*Z(M)*Z(L)**2
PA(I+4) = 3.*Z(M)*(C(L)*D(N)+C(N)*D(L))*Z(L)**2 + (D(N)*C(M)+D(M)*
1C(N))*Z(L)**3 + 3.*Z(N)*(D(M)*C(L)+D(L)*C(M))*Z(L)**2 + 6.*D(L)*
1C(L)*Z(L)*Z(M)*Z(N)
PA(I+5) = 6.*D(L)*C(N)*Z(N)*Z(L)**2 + 2.*C(N)*D(N)*Z(L)**3 + 6.*C(L)
1D(N)*Z(N)*Z(L)**2 + 6.*C(L)*D(L)*Z(L)*Z(N)**2
II=I+1
IF(II.EQ.2)GOTO 60
IF(II.EQ.8)GOTO 61
IF(II.EQ.14)GOTO 62
60 I=7
L=2
M=3
N=1
GOTO 59
61 I=13
L=3
M=1
N=2
GOTO 59
62 I=19
L=1
M=2
N=3
63 PA(I) = 2.*Z(N)*(C(L)*D(N)+C(N)*D(L))*Z(M)**2 + 2.*Z(M)*(C(L)*D(M)+
1C(M)*D(L))*Z(N)**2 + 2.*C(M)*D(M)*Z(L)*Z(N)**2 + 4.*Z(L)*Z(M)*Z(N)*
1(C(M)*D(N)+C(N)*D(M)) + 2.*C(N)*D(N)*Z(L)*Z(M)**2

```

```
II=I+1
IF(II.EQ.20)GOTO 64
IF(II.EQ.21)GOTO 65
IF(II.EQ.22)GOTO 66
64 I=20
   L=2
   M=3
   N=1
   GOTO 63
65 I=21
   L=3
   M=1
   N=2
   GOTO 63
66 GOTO(67,68),IK
67 DO 70 IP=1,21
   FYX(IP)=PA(IP)
70 CONTINUE
   GOTO 90
68 DO 80 IP=1,21
   FYY(IP)=PA(IP)
80 CONTINUE
90 CONTINUE
   RETURN
   END
```

SUBROUTINE ZSYST3(F,EPS,NSIG,N,X,XO,XB,DT,ITMAX,IER)

C *****

C

C SOLVE THE NONLINEAR ALGEBRAIC EQ. USING BROWN S METHOD
C FOR THE SUDDENLY ACCELERATED PLATE USING STREAM
C FUNCTION FORMULATION AND QUINTIC ELEMENT
C MODIFIED VERSION OF THE IMSL SUBROUTINE ZSYSTM

C

C *****

DOUBLE PRECISION EPS,DELTA,XTEMP,PREC,ETA, TOP,RELCON,F,FMAX,
1 PT1,P2,HOLD,H,FPLUS,DERMAX,TEST,X,WA,E,
2 ZERO,PM1
COMMON/SLVT/NN1(12),NN2(12),NN3(12),NEC1(14),NEC2(14),NEC3(14),
1NEO1(14),NEO2(14),NEO3(14),
1RIJST(18,108),B1(12),B2(12),B3(12),C1(12),C2(12),C3(12),AREA(12)
1,Q(12,378),RLST(18,108)
1,PS(16),PSX(16),PSY(16),PSXX(16),PSXY(16),PSYY(16),PSO(16),
1PSXO(16),PSYO(16),PSXXO(16),PSXYO(16),PSYYO(16),NE
DIMENSION X(1),XO(1),XB(1),WA(730)
DATA PREC,DELTA/5.D-14,5.D-10/
DATA ZERO,PM1,PT1,P2/0.00,.100,.000100,.00200/
IER=0
N2 = N+N
RELCON=10.0+0**(-NSIG)
JTEST = 1
IERROR=0
IPART=((N+2)*(N-1))/2
ITMP=IPART+N
LKSUB=ITMP+N
DO 130 M = 1, ITMAX
IQUIT=0
FMAX=ZERO

```

      M1 = M-1
      K1 = LKSUB + 1
      KMIN = LKSUB + N
      XTEMP = ZERO
      DO 5 J = K1, KMIN
        XTEMP = XTEMP+1/DO
        WA(J) = XTEMP
5     CONTINUE
      K = 1
10    IF(K .LE. 1) GO TO 30
      KMIN = K-1
      KK = 1
15    DO 25 K1=1, KMIN
      ISUB=K-K1
      MM=((ISUB-1)*(N2 -ISUB))/2
      LIM=N-ISUB
      KPOINT = WA(LKSUB+ISUB)+PM1
      ISUB1 = ISUB-1
      X(KPOINT)=ZERO
      DO 20 L1=1, LIM
        JS1=ISUB1+L1
        LKJSUB=LKSUB + JS1 +1
        IJ=MM+JS1
        JPOINT= WA(LKJSUB) + PM1
        X(KPOINT)=X(KPOINT) + WA(IJ)*X(JPOINT)
20    CONTINUE
      X(KPOINT)=X(KPOINT) + WA(MM+N)
25    CONTINUE
      IF(KK.EQ.1)GOTO 30
      IF(KK.EQ.2)GOTO 45

```

```

      IF(KK.EQ.3)GOTO 105
30  E=F(K,X,X0,XB,DT)
      FMAX= DMAX1(FMAX,DABS(E))
      IF(DABS(E) .GE. EPS) GO TO 35
      IQUIT=IQUIT+1
      IF(IQUIT .EQ. N) GO TO 140
35  I = K
40  IP=IPART+I
      ITEMp = WA(LKSUB+I) + PM1
      HOLD = X(ITEMp)
      ETA=.001D+00*DABS(HOLD)
      IF(DABS(HOLD) .LT. PREC) ETA=DELTA
      H = DMIN1(FMAX,ETA)
      IF(H .LT. PREC) H=PREC
      X(ITEMp)=HOLD+H
      IF(K.LE.1)GOTO 45
      KK = 2
      GO TO 15
45  FPLUS=F(K,X,X0,XB,DT)
      TOP=FPLUS-E
      IF(DABS(TOP) .GE. PREC) GO TO 50
      WA(IP)=ZERO
      GO TO 55
50  WA(IP)=TOP/H
55  X(ITEMp)=HOLD
      I = I + 1
      IF (I .LE. N) GO TO 40
      IF (K .LT. N) GO TO 60
      IP=IPART+N
      IF(DABS(WA(IP)) .EQ. ZERO) GO TO 80
      X(ITEMp) = -E/ WA(IP) + X(ITEMp)

```

```

GO TO 100
60  KL=LKSUB+K
    LOOK= WA(KL) + PM1
    KMAX=LOOK
    IP=IPART+K
    DERM=OABS( WA(IP))
    KPLUS = K+1
    DO 65 I = KPLUS,N
        TEST=OABS( WA(IPART+I))
        IF(TEST .LE. DERM) GO TO 65
        DERM = TEST
        KMAX=I
65  CONTINUE
    IF(LOOK .EQ. KMAX) GO TO 75
    LKMAX=LKSUB+KMAX
    WA(KL)=WA(LKMAX)
    WA(LKMAX)=LOOK
    IP=IPART+KMAX
    XTEMP= WA(IP)
    IPK=IPART+K
    WA(IP)=WA(IPK)
    WA(IPK)=XTEMP
    IF(K .LT. 2) GO TO 75
    KMIN=K-1
    I1 = 0
    DO 70 I=1,KMIN
        L=((I1) + (N2 - I))/2 + 1
        J=L+KMAX
        XTEMP= WA(J)
        JJ=L+K

```

```

      WA(J)=WA(JJ)
      WA(JJ)=XTEMP
      I1 = I
70  CONTINUE
75  IF (DABS(WA(IPART+K)).NE. ZERO) GO TO 90
80  IF(IERROR .EQ. 1) GO TO 135
      DO 85 I=1,N
          X(I) = 0.900*X(I) + .1234500
85  CONTINUE
      IERROR=1
      GO TO 105
90  L=((K-1)*(N2 -K))/2
      KN=L+N
      I1 = L-1
      WA(KN)=ZERO
      IPK=IPART+K
      DO 95 J = KPLUS,N
          JSUB= WA(LKSUB+J) + PM1
          JJ=I1+J
          IPJ=IPART+J
          WA(JJ)=-WA(IPJ)/WA(IPK)
          WA(KN)=WA(KN)+WA(IPJ)*X(JSUB)
95  CONTINUE
      LK= WA(LKSUB+K) + PM1
      WA(KN)=(WA(KN)-E)/WA(IPK) + X(LK)
      K = K+1
      IF (K .LE. N) GO TO 10
100 IF( N .EQ. 1) GO TO 105
      KMIN=N-1
      KK = 3
      GO TO 15

```



```

105  IF (M .LE. 1) GO TO 120
      DO 110 I = 1,N
          IF (DABS(WA(ITMP+I))-X(I)) .GT. DABS(X(I))*RELCON) GO TO 11
110  CONTINUE
      JTEST = JTEST+1
      IF (JTEST-3)120,140,140
115  JTEST = 1
120  DO 125 I = 1,N
      WA(ITMP+I)=X(I)
125  CONTINUE
130  CONTINUE
      M=ITMAX
      IER = 129
      GO TO 240
135  IER = 130
140  FMAX=ZERO
      TEST=1.00+15
      IF (N.GT.1) GOTO 145
      WA(IPART+2)=F(1,X,X0,XB,DT)
      FMAX= DMAX1(FMAX,DABS(WA(IPART+2)))
      GOTO 155
145  DO 150 I=1,N
      IP=IPART+I
      WA(IP)=F(I,X,X0,XB,DT)
      FMAX=DMAX1(FMAX,DABS(WA(IP)))
150  CONTINUE
155  K=1
      DO 160 I=1,N
      WA(I)=X(I)
      IF (DABS(X(I)) .GT. P2) GO TO 160

```

```
      K=2
      WA(I)=ZERO
160 CONTINUE
      IF(K .EQ. 1) GO TO 195
      KK = 1
      GO TO 205
165 IF(FMAX .LT. TEST) GO TO 190
      DO 170 I=1,N
          X(I)= WA(I)
170 CONTINUE
      IF (N .GT. 1) GO TO 175
      WA(IPART+2) = WA(ITMP+2)
      GO TO 185
175 DO 180 I = 1,N
      WA(IPART+I) = WA(ITMP+I)
180 CONTINUE
185 FMAX=TEST
190 K=1
195 ITEST=0
      DO 200 I=1,N
          WA(I)=X(I)
          IF(DABS(X(I))) .LE. P2) GO TO 200
          L=X(I)+PT1
          J=X(I)-PT1
          IF(L .EQ. J) GO TO 200
          WA(I) =ISIGN(1,J)*MAX0(IABS(L),IABS(J))
          K=2
200 CONTINUE
      IF(K .EQ. 1) GO TO 235
      KK = 2
205 TEST=ZERO
```

```
      IF (N .GT. 1) GO TO 210
      WA(ITMP+2)=F(1,WA,XO,XB,DT)
      TEST = DMAX1(TEST,DABS(WA(ITMP+2)))
      GO TO 220
210  DO 215 I=1,N
      IT=ITMP+I
      WA(IT)=F(I,WA,XO,XB,DT)
      TEST=DMAX1(TEST,DABS(WA(IT)))
215  CONTINUE
220  IF(KK.EQ.1) GOTO 165
225  IF(FMAX .LT. TEST) GO TO 235
250  DO 230 I=1,N
      X(I)= WA(I)
230  CONTINUE
260  ITEST=1
235  IF(FMAX .LT. EPS .OR. TEST .LT. EPS) IER = 0
240  ITMAX=M1 + 1
9000 CONTINUE
270  IF(IER.EQ.0) GOTO 9005
      PRINT 88
      88 FORMAT(5X,*MATRIX IS ILL CONDITIONED*)
9005 RETURN
      END
```

END

5

08

78

FIN