CANADIAN THESES ON MICROFICHE

THÈSES CANADIENNES SUR MICROFICHE

National Library of Canada
Collections Development Branch

Canadian Theses on
Microfiche Service

Ottawa, Canada
K1A 0N4

Bibliothèque nationale du Canada
Direction du développement des collections

Service des thèses canadiennes
sur microfiche

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 an inferior 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.

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 qualité inférieure.

Les documents qui font 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.

THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED

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

Canada

NL 339 (r. 86/09)

FINITE ELEMENT IMPLEMENTATION OF SOME CONVENTIONAL GEOTECHNICAL PROBLEMS

by

Richard Wan

Submitted in partial fulfillment of the
requirements for the degree of
Master of Applied Science
under the supervision of
Dr. C.I. Papantonopoulos
Assistant Professor

Department of Civil Engineering
Faculty of Science and Engineering
University of Ottawa
Ottawa, Canada
July, 1985

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

To my parents

ACKNOWLEDGEMENTS

Sincere gratitude is due to Dr. C.I Papantonopoulos for his assistance during this project and there is also grateful appreciation of the generous help provided by Dr. B.H Fellenius.

The author is considerably indebted to A. Fusco for his help and encouragement throughout this project. His guidance has helped to bring this work to fruition.

Also, special appreciation is extended towards the staff of the Civil Engineering Department. The author is thankful to the National Research Council for having partially provided financial assistance (Grant NSERC A8478) in carrying out this research and to the School of Graduate Studies for the grant of an entrance scholarship.

SUMMARY

At present, the finite element method has the power to solve a very wide variety of problems described by quite different differential equations. When viewed from a finite element standpoint, all equilibrium problems encountered in geotechnical engineering, whether involving solids or fluids, merge to the same form of equation.

This study deals with the coding of some of the most common geotechnical problems, with emphasis on the refinement of existing numerical algorithms in order to overcome fundamental difficulties usually met in such analyses.

Chapter 1 depicts the objectives of the present study while the general overview of the different problems implemented with reference to governing differential equations and programming structure is presented in the second Chapter "Preliminaries".

Solving the Laplace's equation finds plentiful application in the areas of seepage and ground water flow. Chapter 3 treats the steady flow problem and describes a new algorithm employed to shift the nodes in an unconfined flow analysis. Its performance is then illustrated through examples.

Chapter 4 leads off with problems of transient nature and deals with the implementation of the uncoupled consolidation theory. The development of a computer code with results obtained are then presented.

Chapter 5 is concerned with stress analysis in a continuum. Material non-linearity is included with an introduction to the basic equations and their discretization in finite elements. A brief review of failure criteria and their inclusion in elasto-plasticity models is then done. However emphasis is laid on the actual refinements on the numerical algorithms used in the development of a computer code. The testing is done on several examples and results are discussed.

In Chapter 6, the coupling between solid and fluid phases is treated as an amalgamation of the respective phase formulations presented in Chapters 3 and 5. The numerical difficulties and errors usually encountered in this complex analysis are presented and their remedy proposed. Further extension is done to the more realistic elasto-plastic coupled theory to embrace the generalized Burland and linear elastic perfectly plastic models. Their responses are subsequently examined.

Any finite element program usually needs supporting programs for a comprehensive analysis of the numerical results. Chapter 7 describes the structure and programming organization with flowcharts involved in writing such programs.

Finally, Chapter 8 finishes up with the general conclusions. Details of all relevant subroutines developed in this research are attached in Appendices A, B, C, D.

GLOSSARY OF SYMBOLS

a_1, a_2, a_3	: vectorial components of the gradient of a stress function
a, b	: Henkel's pore pressure parameters
a, b, d	: Generalized Burland model coefficients
$\underset{\sim}{b}$	: current body forces
c	: cohesion
c_1, c_2, c_3, c_4	: coefficients
$\underset{\sim}{c}_F = C^e \underset{\sim}{\nabla}_\sigma F$	
$\underset{\sim}{c}_G = C^e \underset{\sim}{\nabla}_\sigma G$	
$\underset{\sim}{d}$	: nodal displacement vector
$d\bar{l} = l \, dG_\sigma$	: plastic multiplier
f	: function
$\underset{\sim}{f}$	: equivalent load vector at the nodes
$\underset{\sim}{g}$	: equivalent flow vector at the nodes
h	: size of the yield and bounding surfaces
h, h_o	: current and initial head pressure functions
$\underset{\sim}{h}$, $\underset{\sim}{h}_o$	: current and initial nodal head pressures
i, j, k	: indices
k	: hardening parameter
$l = 1/A$	: reciprocal of the plastic modulus
l_{ij}	: small Lagrange stain component
m	: mass
n	: degree of a function, ratio of ellipse's semi-axes

$\hat{n}_F$, $\hat{n}_G$	: unit gradient to F and G functions
p	: sinks and sources
p,p'	: total and effective mean stress tensors ($p = \sigma_{kk}/3$)
p_e	: equivalent preconsolidation pressure
$\underset{\sim}{q}$	: flow vector
q	: deviatoric stress invariant ($q = \sqrt{3}\, J_2^{\frac{1}{2}} = (\frac{3}{2}\, s_{ij}\, s_{ij})^{\frac{1}{2}}$)
r	: Generilized Burland model coefficient, reduction factor
s, $\bar{s}$	: shape functions
s_{ij}	: deviatoric stress components of the Cauchy stress tensor
t	: time
$\underset{\sim}{t}$	: traction force
u	: displacement function
$\bar{v}$, $\bar{v}_o$	: current and initial specific volumes
$\bar{v}_\chi$, $\bar{v}_\lambda$	: specific volume at p' = 1 KPa on the swelling and virgin lines
v	: current fluid velocity function
$\underset{\sim}{v}^*$	: "smoothed" nodal velocity
$\underset{\sim}{x}$	: spatial coordinate
z	: geometrical height
A	: plastic modulus
B	: plastic modulus, B-matrix
BE	: Big Element
C	: coupling matrix
C^α	: constitutive matrix for a state of the material, where α stands for elasticity, plasticity or elasto-plasticity
CS	: Critical State

CPU	: Central Processing Unit
D	: D-matrix in TERCON, damping matrix
DBE	: Dynamic Big Element
E	: Young's modulus
F	: potential function
FE	: Finite Element
FEM	: Finite Element Method
G	: shear modulus, yield function
I_1	: 1st stress invariant of the Cauchy stress tensor ($I_1 = \sigma_{kk}$)
J_2	: 2nd stress invariant of the deviatoric stress tensor ($J_2 = 1/2\ s_{ij}\ s_{ij}$)
J_3	: 3rd stress invariant of the deviatoric stress tensor ($J_3 = \det(s_{ij})$)
K	: permeability tensor
K_p	: normalized plastic modulus
K^α	: stiffness matrix of the solid phase for α state of the material, where α stands for elasticity, plasticity or elasto-plasticity
L	: small Lagrange strain tensor
M	: slope of the linear yield (failure) function in a p vs. q diagram, smoothing matrix
M_1, M_2	: slopes in the Generilized Burland model, smoothing submatrices
N	: q-intercept of the linear yield (failure) function in a p vs.q diagram
NCS	: Normally Consolidated State
OCS	: OverConsolidated State
$\tilde{P}$	: Permeability matrix
$\bar{P}$	: hydraulic "stiffness" matrix in TERCON

and SEEP

Q	: failure surface
S	: shape matrix
T	: Cauchy stress tensor
V	: current volume
X	: material coordinates
α, β	: coordinates of the centre of the Generalized Burland model
γ_f	: specific weight of the fluid
$\underset{\sim}{\delta}$	: space diagonal vector
δ_{ij}	: Kronecker delta
ε^{α}	: homologous vectors to the small Lagrange strain tensor
ε_v^{α}	: trace of the small Lagrange strain tensor ($\varepsilon_v = \varepsilon_{kk}$)
ε_s^{α}	: deviatoric invariant associated with the small Lagrange strain tensor ($\varepsilon_s = (\frac{3}{2} e_{ij} e_{ij})^{\frac{1}{2}}$)
θ	: stress invariant, similar to Lode's angle, associated with Cauchy stress tensor ($\theta = \frac{1}{3} \arcsin (\frac{-3\sqrt{3}}{2} J_3/J_2^{3/2})$)
Θ	: parameter in Θ-Wilson time marching scheme
λ	: Lame's constant, slope of virgin line in a v vs. ln p diagram
ν	: Poisson's ratio
ξ, η	: local coordinates
π, π^o, π^e	: current, initial and excess pore pressure functions
$\underset{\sim}{\pi}$, $\underset{\sim}{\pi}^o$, $\underset{\sim}{\pi}^e$	: current, initial and excess nodal pore pressures
ρ	: current density
s_{ij}, s'_{ij}	: total and effective Cauchy stress tensor components

$\underset{\sim}{s}$, $\underset{\sim}{s}'$	: homologous vectors to the total and effective Cauchy stress tensor
ϕ	: friction angle
χ	: slope of the swelling line in a v vs. ln p' space
$\underset{\sim}{\psi}$	: residual forces
F_i	: function
Ω	: current volume
Γ	: surface

Matrices are denoted by uppercase Latin characters and arrays by lowercase Latin characters with a wave sign below.

DICTIONARY OF SOME PROGRAM VARIABLES

BMATX(NSTRE,NEVAB)	element strain matrix at any point within the element
CARTD(NDIME,NNODE)	Cartesian shape function derivatives associated with nodes of current element
CESTIF(27,27)	coupled element stiffness matrix
CMATX(18,9)	coupling matrix
COOBI(NPOIN,NDIME)	coordinates of a Big Element nodal points
COORD(NPOIN,NDIME)	coordinates of nodal points
DMATX	elastic constitutive matrix
DTIME	time interval adopted
DVOLU	an infinitesimal element of volume at a Gauss point
EHIST(MTOTG,10)	array containing the stress history at each Gauss point for the Critical State model
ELOAD(MELEM,MEVAB)	nodal force for each element
ESTIF(MEVAB,MEVAB)	common element stiffness matrix
FIXED(IPOSN)	prescribed displacement or pore pressure values transferred from PRESC array in FRONT
HEAD(MTOTV)	vector of total head at Gauss points
H2	height of downstream water level in unconfined flow analysis
IOPT	option for the type of mesh to be generated IOPT= 1 standard 2 transition type 3 transformed triangles

LNOBI(NELEM,NNODE)	Big element node numbers listed for each BE
LOCFS(MDUMM)	array of nodes on free surface
MATNO(NELEM)	material set numbers for each element
MDUMM	maximum number of dummy coordinates in dynamic mesh generation
MELEM	maximum number of elements in the structure
MEVAB	maximum number of variables per element
MPOIN	maximum nodal points in the structure
MTOTV	maximum number of Gauss points in the structure
NDIME	number of coordinate components required to define each nodal point
NDMN	total number of domains or BE
NDOWN	number of downstream nodes in unconfined flow analysis
NDYN	total number of dynamic domains or DBE
NELEM	total number of elements in the structure
NEVAB	number of variables per element
NGAUS	number of Gauss rule adopted
NN	starting node number of a BE
NNODE	number of nodes per element
NPOIN	total nodal points in the structure
NPROP	number of material parameters required to define the characteristics of a material NPROP = 9 in ESEPA 13 TEPSA
NSPEC	number of special BE (encountered when an internal free surface is treated in steady unconfined flow)
NTIPE	parameter defining type of transition element pattern to be generated

NVFIX	total number of boundary points at which one or more degree of freedom are restrained
NXL	number of subdivision in X direction
NYL	number of subdivision in Y direction
PRESC(IVFIX,IDOFN)	indicates that the IDOFN th. degree of freedom of the IVFIX th. boundary node has a prescribed value provided the corresponding value of IFRE(IVFIX, IDOFN) = 1
PROPS(NMATS,NPROP)	material properties for each element set
PSTIF(MEVAB,MEVAB)	element hydraulic "stiffness" matrix
SHAPE(NNODE)	shape functions associated with each node of the current element sample
SLOPE	downstream slope in unconfined flow analysis
SMATX(8,4)	"Smoothing" matrix
STRIN(ISTRE,KGAUS)	initial stresses at each Gauss point
STRND(MPOIN,4)	equivalent stresses at the nodes
STRSG(ISTR1,KGAUS)	current Gauss point stresses
TIMEX	discretizing parameter in time marching scheme
VELCN(MPOIN,4)	fluid velocity component at nodal points
VELG(KGAUS,2)	fluid velocity component at Gauss points
XM, YM	external nodal coordinates of BE
XTIME	time step ratio or subdivision parameter in time marching scheme

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter I

INTRODUCTION

1.1 GENERAL

The analysis of a saturated soil usually comprises the determination of the pore pressure distribution associated with a pre-existing regime of steady flow and the calculation of the stresses produced by the application of a set of external forces.

For soils of low permeability, e.g. clays, the loading process causes a build up of excess pore pressures which dissipate with time thus giving origin to the so-called consolidation process. Though the basic equations governing the different problems are well established, very few analytical solutions are nowadays available for a rigorous investigation of real geotechnical structures.

In fact, the classical mathematical tools can solve the governing differential equations only for ideal soils with simple boundary conditions. That is why, until recent years, geotechnical engineers were forced to develop simplified approaches which unfortunately proved to be only applicable to very specific practical situations.

Nowadays the advent of powerful numerical techniques and computer facilities has greatly enhanced the rigorous

solution of these equations so that the last few years have witnessed a substantial amount of work in the development of engineering-oriented computer codes. Concurrently, several researchers have directed considerable efforts towards producing more concise and realistic constitutive equations since numerical techniques offer the possibility of solving highly non-linear differential equations.

In short, the rapid expansion of both computer and numerical facilities has had tremendous impetus to the solution of problems in engineering science and inexorably has opened a new era in analysis and design.

1.2 PROBLEM STATEMENT

Among the different numerical techniques nowadays available, the F.E.M is the most flexible and commonly used method in the solution of geotechnical engineering problems.

Because of the evolutionary nature of this technique, a number of ready available codes have become increasingly inefficient and unwidely. In the meantime, theoretical studies have pinpointed a number of fundamental flaws inherent in many popular computer codes, so that it has been found necessary to devise new algorithms in order to increase both efficiency and reliability of F.E. programs.

For example, in the analysis of unconfined flow, the usual algorithms are often cumbersome and consequently costly. On the other hand, the numerical implementation of

elasto-plastic problems is usually prone to inaccuracies in the determination of specific quantities needed for a correct stress analysis. Turning to the consolidation problem, a number of programs treating the two common theories (uncoupled and coupled) are available. While, the implementation of the uncoupled theory has been, so far, exclusively restricted to 1-D cases, recent studies have clearly revealed the numerical problems encountered in the usual way of implementing the coupled theory.

1.3 OBJECTIVE AND SCOPE OF STUDY

The goal of this research is to implement consistently in F.E. the following geotechnical problems:

1. Seepage under steady conditions for both confined and unconfined flow under anisotropic permeability conditions.
2. Elasto-plastic analysis of a highly permeable saturated medium.
3. Uncoupled non-linear consolidation in a soil with anisotropic permeability.
4. Coupled elasto-plastic consolidation in a soil with anisotropic permeability.

Plastic deformations are described by both linear perfectly plastic models and generalized Cam-clay model (Fusco, 1985). The first group of models comprises the classical failure criteria of Tresca, von Mises, Mohr-

Coulomb and Drucker-Prager, while the generalized Cam-clay model, incorporating both the Mohr-Coulomb and Drucker-Prager failure criteria, leads to the concept of Critical State Soil Mechanics.

The F.E. implementation wants to consistently treat the 2-D and axisymmetric cases by employing the isoparametric elements. Since all problems treated deal with symmetric stiffness matrices, the solution process is carried out by the powerful "Front Solver" to cut down computational effort. Moreover, emphasis is made on the improvement of numerical algorithms to remedy some of the inaccuracies common to many available F.E. codes.

This thesis is not intended to stand alone, but rather to be the numerical counterpart of a Ph.D thesis (Fusco 1985). Thus, for a comprehensive description of the underlying theory and derivations for this thesis, the reader is directed to the above cited reference.

Chapter II

PRELIMINARIES

2.1 INTRODUCTION

The stress analysis in soils is very similar in concept to that of any other engineering material unless saturated conditions are met. In this case the classical Cauchy's equations of equilibrium fail to adequately determine the state of stress in the medium.

In a saturated soil, the stresses produced by the application of a set of external load are transferred partly to the soil skeleton and partly to the fluid. For highly impermeable soils, an excess pore pressure is set up with respect to the initial hydrostatic conditions.

To take into account the role of the fluid component in a 2-phase medium, it is then necessary to have recourse to the principle of effective stress, in conjunction with the continuity of mass.

In Sect. 2.2, is presented a brief review of the equations governing the different aspects involved in the analysis of a saturated medium. Because of the complexity of these equations, few analytical solutions are available, as underlined in Sect. 2.3. Consequently this explains the increasing interest engaged in the development of computer

codes toward a general solution of the above-mentioned problem. The basic principles of the F.E.M. are thereby explained in Sect. 2.4.

2.2 BRIEF REVIEW

The equilibrium in a solid can be generally described by either the Cauchy's equilibrium equations (Cauchy 1827) or the principle of virtual work (Castigliano 1879).

$$\frac{\partial \sigma_{ij}}{\partial x_j} + \rho\, b_i = 0 \tag{2.1}$$

$$\int_\Omega \sigma_{ij}\, \delta\varepsilon_{ij}\, d\Omega = \int_\Gamma t_i \cdot \delta u_i\, d\Gamma + \int_\Omega \rho \cdot b_i \cdot \delta u_i\, d\Omega \tag{2.2}$$

where $\underset{\sim}{\sigma}$ is the Cauchy stress tensor, $\underset{\sim}{b}$ the body forces, ρ the density, $\underset{\sim}{t}$ the boundary traction forces, $\underset{\sim}{u}$ the displacements, Ω the domain of interest and Γ the contour of the boundary.

The solution of the above equations requires the determination of the constitutive law relating stress and strain and generally written as:

$$\underset{\sim}{\sigma} = C \cdot \underset{\sim}{\varepsilon} \tag{2.3}$$

where C is a second order tensor describing the material property.

In a saturated medium, the pore pressure participates in the equilibrium of the system and, in the assumption of an inviscid fluid, its contribution can be mathematically expressed by the principle of effective stress (after Terzaghi, 1943).

$$\sigma_{ij} = \sigma'_{ij} - \delta_{ij}\pi \tag{2.4a}$$

where

$$\underset{\sim}{\sigma}' = \underset{\sim}{\sigma}'_o + \underset{\sim}{\sigma}'(t) \tag{2.4b}$$

and

$$\pi = \pi^o + \pi^e \tag{2.4c}$$

$\underset{\sim}{\sigma}'_o$, $\underset{\sim}{\sigma}'$, π^o, π^e, and π are the initial and current effective stresses, the initial, excess and current pore pressures respectively.

The substitution of the above equations into either Eq. 2.1 or 2.2 clearly indicates that the solution of the

equilibrium involves the determination of two additional field variables π^o and π^e

To evaluate these two field variables, it is necessary to consider the law of conservation of mass, in this case, extended to a 2-phase medium. From the works of Green and Nadghi (1965), it is possible to prove that the continuity equation, applied to an incompressible fluid in the assumption of small deformation, takes the following form:

$$\text{div } K \text{ grad } h = \frac{d\varepsilon_v}{dt} \tag{2.5a}$$

where

$$h = z + \frac{\pi}{\gamma_f} = h_o + \frac{\pi^e}{\gamma_f} \tag{2.5b}$$

and

$$h_o = (z + \frac{\pi^o}{\gamma_f}) \tag{2.5c}$$

K is the fluid permeability matrix, h the total head, ε_v the volumetric change, γ_f the fluid unit weight and z the elevation of the geometrical point under consideration.

For soils at rest (zero volumetric change and excess pore pressure), the continuity equation is reduced to the well-known Laplace's equation which describes laminar flow under steady conditions:

$$\text{div } K \text{ grad } h_o = 0 \qquad (2.6)$$

If from the above condition of equilibrium a set of additional forces is applied, then excess pore pressures are built up and the soil undergoes volumetric changes with time. Under this condition, assuming known the initial distribution of total fluid head h_o , Eq. 2.5a can be rewritten as:

$$\text{div } K \text{ grad}\left(\frac{\pi^e}{\gamma_f}\right) = \frac{d\varepsilon_v}{dt} \qquad (2.7)$$

Rendulic, 1936, assumed that during the dissipation of the excess pore pressure π^e, the total stress does not change with time. It is then possible to deduce that the variation of effective normal pressure $\Delta p'$, is equal to that of the excess pore pressure, $\Delta\pi^e$. Since under the hypothesis of small deformation, $\Delta p'$ can be expressed in terms of the change in volumetric strain $\Delta\varepsilon_v$, $\Delta\pi^e$ results in:

$$\Delta\pi^e = \Delta p' = B \cdot \Delta\varepsilon_v \qquad (2.8a)$$

where B is the bulk modulus and may, under the hypothesis of linear elasticity and small deformation, assume the following forms:

1-D
$$B = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \qquad (2.8b)$$

2-D
$$B = \frac{E(1-\nu)}{2(1+\nu)(1-2\nu)} \qquad (2.8c)$$

Axisymmetry
$$B = \frac{E}{3(1-2\nu)} \qquad (2.8d)$$

The substitution of the above equation into Eq. 2.7 leads to the parabolic differential equation governing the uncoupled theory of consolidation.

$$\text{div } K \text{ grad } \frac{\pi^e}{\gamma_f} = \frac{1}{B}\,\dot{\pi}^e \qquad (2.9)$$

Its solution describes the consolidation process, once the initial excess pore pressures are known. For instance, these can be calculated in terms of the variation of the total stress invariants Δp and Δq, as proposed by Henkel (1960).

$$\pi^e = a\ \Delta p + b\ \Delta q \qquad (2.10)$$

where a and b are pore pressure parameters.

The uncoupled theory, though the most popular approach adopted in engineering practice to describe the consolidation process, suffers two major shortcomings. Firstly it requires the determination of initial excess pore pressures by means of a semi-empirical formula and secondly it does not take into account solid-fluid interaction, the key-point to the consolidation process.

Biot (1941a), instead, clearly pointed out that a rigorous solution of the consolidation process requires the simultaneous solution of the Cauchy's equilibrium and the continuity equations, Table 2.1. This approach is referred as the coupled consolidation theory.

TABLE 2.1

Field equations of Coupled Consolidation problem in the hypothesis of Linearized Elasticity

Cauchy's equation of equilibrium

3 equations

$$\frac{\partial \sigma'_{ij}}{\partial X_j} - \delta_{ij} \frac{\partial \pi^e}{\partial X_j} = 0 \qquad (2.11)$$

Generalized Hooke's law

6 equations

$$\sigma'_{ij} = \lambda \, 1_{kk} \, \delta_{ij} + 2\mu \, 1_{ij} \qquad (2.12)$$

λ and μ are the Lamé coefficients

Strain-Displacement relationship

6 equations

$$1_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} \right) \qquad (2.13)$$

1 Continuity eq. $\qquad 1 * K_{ij} \, \pi^e_{,ji} = \dot{u}_{k,k} \qquad (2.14)$

16 equations, for 6 effective stresses, 6 strains, 3 displacements and 1 excess pore pressure.

2.3 ANALYTICAL SOLUTIONS

2.3.1 Seepage

The mathematical solution of laminar flow described by the Laplace's equation can only be generally obtained for homogeneous, isotropic media in confined flow situations with simple boundary conditions.

A comprehensive description of these solutions is given by Harr (1962), and Aravin and Numerov (1965).

Instead, it is frequently met in geotechnical engineering, the so-called "unconfined" flow which necessitates the finding of a "free" surface not known a priori. In this situation engineers have always relied upon graphical construction methods (Casagrande, 1940 and Cedergren, 1967). On the other hand, some analytical solutions for specific cases have been proposed, (Shankin 1947, Nelson and Skornyakov 1949, and Polubarinova-Kochina, 1962), but are usually inconvenient to use.

2.3.2 Uncoupled Consolidation Theory

Closed-form solutions of Eq. 2.9 are unfortunately restricted to 1-D situations in an homogeneous, isotropic medium. Some examples of these solutions have been found by Taylor (1948) and Carslaw and Jaeger (1959), while simplified solutions were proposed by Rendulic (1936) and Terzaghi (1943), under the ideal assumptions previously mentioned.

2.3.3 Coupled Consolidation Theory

Analytical solutions of Biot's equations (Table 2.1) are rare and only exist in ideal cases. As a result, solutions are usually restricted to the case of a flexible footing load on a homogeneous, isotropic and linear elastic semi-infinite medium. These solutions have been published by Biot (1941b,c), Mandel (1957), Josselin de Joung (1957), Mcnamee and Gibson (1960a,b) and Cryer (1963).

2.4 THE FINITE ELEMENT METHOD

The finite element method is a technique for transforming partial differential equations into a set of algebraic equations. This is accomplished by discretizing the closed and bounded domain over which the equations are defined into a number of finite subdomains known as Finite Elements. Along the edges of the elements, are defined a number of "Nodal" points . Within each element, the field variable $u(\underset{\sim}{x},t)$ is approximated by means of a shape function whose order is determined by the number of element nodal points.

$$u(\underset{\sim}{x},t) = \underset{\sim}{s}(\underset{\sim}{x})\ \underset{\sim}{u}(t) \qquad (2.15)$$

where $\underset{\sim}{u}(t)$ contains the discrete values of u at the nodes.

There exist many types of shape functions, but the ones based on the isoparametric concept, (Ergatoudis et al. 1968)

have by now established their superiority. In these types of elements the function is mapped into a local system whose normalized coordinates (ξ, η, ζ) vary between -1 and 1. This mapping technique allows the use of the Gaussian quadrature formula for integration over the element, regardless its geometrical complexity in the real coordinate system.

Generally, by means of shape functions and mathematical procedures based on the variational principle, different kinds of differential equation can be conveniently discretized into a set of algebraic equations as follows:

$$M \ddot{\underset{\sim}{u}}(t) + D \dot{\underset{\sim}{u}}(t) + K \underset{\sim}{u}(t) = f(t) \qquad (2.16a)$$

where in general

$$M = \int_{\Omega} m \, S^T S \, d\Omega \qquad (2.16b)$$

$$D = \int_{\Omega} d \, S^T S \, d\Omega \qquad (2.16c)$$

$$K = \int_{\Omega} B^T C \, B \, d\Omega \qquad (2.16d)$$

$$\underset{\sim}{f}(t) = \int_{\Omega} S^T \underset{\sim}{a}\, d\Omega + \int_{\Gamma_2} S^T \underset{\sim}{b}\, d\Gamma_2 \qquad (2.16e)$$

$$S = [s_i] \qquad (2.16f)$$

$$B = [s_{i,j}] \qquad (2.16g)$$

and m, d, $\underset{\sim}{a}$, $\underset{\sim}{b}$ and C are dependent on the specific problem being analysed. The matrices S and B contain the shape function coefficients and their derivatives respectively.

The integration terms belonging to matrices M, D and K may be performed by the Gaussian quadrature formula if isoparametric elements are used. In general, calling $f(\underset{\sim}{x})$ any of the previously reported integrands, the numerical calculation can be performed as follows:

$$\int_{\Omega} f(\underset{\sim}{x})\, d\Omega = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} W_i\, W_j\, W_k\, f(\xi_i, \eta_j, \xi_k)\, \det J \qquad (2.17)$$

where W_i, W_j and W_k are the weighting factors, J the Jacobian matrix and n the number of Gauss sampling points.

To perform this integration, a number of standard numerical operations must be done and detailed explanations can be found in a wide range of publications and computer codes.

The discretization in time, instead, can be done by using numerous methods. For instance, in the absence of any second derivative with respect of time ($\ddot{u}(t)=o$), the θ-Wilson scheme appears to be the most general. The variable u(t) can be expressed as:

$$\underset{\sim}{u}(t) = (1-\theta)\ \underset{\sim}{u}_{n-1} + \theta\ \underset{\sim}{u}_{n} \qquad (2.18)$$

where $\underset{\sim}{u}_{n-1}$ and $\underset{\sim}{u}_{n}$ are the values of u at times t_{n-1} and t_n respectively for $t \in [t_{n-1},\ t_n]$. The various values θ can assume for the discretization of different types of functions are given in Table 2.2.

As a result, discretizing Eq 2.16a leads to the following set of algebraic linear equations:

$$\bar{K}_n\ \underset{\sim}{u}_{n} = \underset{\sim}{f}_{n-1} \qquad (2.19)$$

where $\bar{K}_n$ incorporates all the matrices dependent on $\underset{\sim}{u}_n$, and $\underset{\sim}{f}_{n-1}$ represents the array of known variables.

At this stage, the solution of Eq. 2.19 determines the current nodal values of the field variable u_n at time t_n.

Several techniques are available for the solution of the system of equations 2.19. In the particular case the matrix $\bar{K}_n$ is symmetric, the frontal solution (Irons, 1970) proves to be the most convenient method to use. In fact it combines both efficiency in solution time and economy in storage.

TABLE 2.2

Different time discretization parameters

function type	θ-value
linear	$\theta = \frac{1}{2}$
parabolic	$\theta = \frac{2}{3}$
logarithmic (Sandhu 1968)	$\theta = 1 + \frac{1}{\Delta t} - \frac{1}{\ell n(1+\Delta t)}$
logarithmic (Hwang et al. 1971)	$\theta = 1 + \frac{1}{\tau_o} - \frac{1}{\ell n(1+\tau_o)}$ $\tau_o = \frac{\Delta t}{\tau_{n-1}}$

2.5 STRUCTURE OF TYPICAL FINITE ELEMENT PROGRAMS DEVELOPED

Each finite element program is generally structured into four different stages of calculations which are: initialization, pre-solution, solution and post-solution (see Fig. 2.1).

In the initialization stage, the geometry and material parameters are entered in Subroutine INPUT whereas the setting up of the working variables is done by a number of subroutines.

The pre-solution stage normally includes the built up of the initial "load" vector $\underset{\sim}{f}_o$ and the calculation of the stiffness matrix $\bar{K}_n$ by means of Subroutines LOADPS and STIFFP respectively.

In the next stage of the program, assembly of each element "stiffness" and "load" is performed and subsequent solution of the current unknown is performed in Subroutine FRONT.

Some problems require the calculation of additional quantities on the basis of the current solution. This takes place in the post-solution stage which also includes the printout of all the requested results in Subroutine OUTPUT.

Finally, time dependent and/or non-linear analyses require an iterative process over the last three stages in accordance to Fig. 2.1.

The main differences between the various F.E. programs however lie in the setting up of the working parameters and

the eventual elaboration of results in the post-solution stage. These usually depend on the mathematical formulation of the problem currently under study. Turning to routines commonly employed, Subroutines INPUT and OUTPUT are rather trivial whereas LOADPS, STIFFP and FRONT are quite standard. It is clear that, for various problems the integrands in Eqs. 2.16b,c,d are different but only few modifications are required to adapt the corresponding subroutines to the current needs.

In this research four master F.E programs together with three supporting programs have been developed. This has led to the writing of codes each comprising of approximately 3000 statements and it is illusory to review through all the programming details. As a result, only primary subroutines are explained in contrast to others which are either standard or trivial in their explanation.

In writing the different programs developed in the next chapters, the structural organization of the standard subroutines such as LOADPS, STIFFP and FRONT were adapted from the program, PLANET, published by Owen and Hinton (1980). The programs were run on an AMDAHL 470 V8 machine at the University of Ottawa.

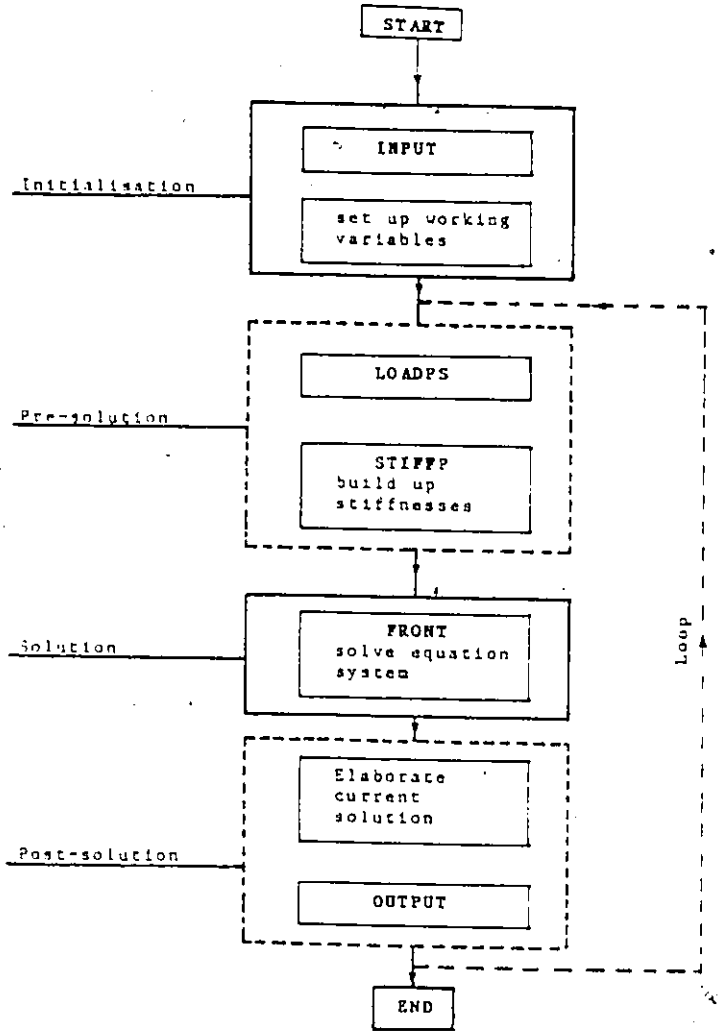


Fig. 2.1 Flowchart of a typical F.E. program developed

Chapter III

SEEPAGE ANALYSIS

3.1 INTRODUCTION

The basic governing equation describing steady flow is the well-known Laplace's equation as discussed in the previous chapter.

Past experience has shown that the solution of the confined flow by F.E. does not pose any particular problem since the domain of calculation is known together with a set of well defined boundary conditions. In return, the analysis of the unconfined flow is not that straightforward and requires an iterative process to locate the position of a phreatic or free surface not known a priori.

Two different procedures, herein distinguished as the conventional and residual methods, are available. The conventional method consists of solving the Laplace's equation assuming a trial free surface which is ultimately checked to satisfy the imposed boundary conditions. If these are not respected, then the free surface is updated for the next iteration until convergence is achieved. Normally, the trial free surface is assumed to be an impermeable boundary and the corresponding computed pore pressures are verified to satisfy the nullity condition.

This approach has been adopted by several authors such as Taylor and Brown (1967), Finn (1967), Newmann and Witherspoon (1970).

The second method was apparently introduced by Desai (1976) in which the necessity of updating the entire mesh is avoided. In this method, once a trial solution is found for a given set of boundary conditions, the trial free surface is located by the nodal points satisfying $\pi^o = 0$. This surface is subsequently adjusted and corrected by using the so-called residual approach.

Both approaches have advantages and disadvantages, but the former is considered to have more generality. However, Finn and Varoglu (1976) pointed out that usual algorithms employed in the first method were unable to control the relative size of the elements during an iteration, which resulted in numerical difficulties. They proposed to update the entire mesh for the location of the free surface by acting on each individual node, which reveals to be rather costly. Fusco (1985), has therefore suggested a new algorithm which updates only regions of space by the use of an efficient mesh generator. Furthermore, a special "smoothing" technique has been incorporated in order to calculate the velocity at the nodes.

3.2 F.E. DISCRETIZATION OF SEEPAGE EQUATIONS

The F.E. discretization of the Laplace's equation is obtained by approximating the main field variable, h_o, by means of convenient shape functions.

$$h_o = \underset{\sim}{s}\ \underset{\sim}{h}_o \tag{3.1}$$

$\underset{\sim}{h}_o$ is the array containing the discrete value of the total head at the nodes.

The equivalent system of equations describing seepage, Eq. 3.2a, can be derived in several ways but the Galerkin's method of weighted residuals proves to be the most convenient one.

$$P\ \underset{\sim}{h}_o = \underset{\sim}{g} \tag{3.2a}$$

where

$$P = \int_\Omega B^T K\ B\ d\Omega \tag{3.2b}$$

and

$$\underset{\sim}{g} = -\left\{\int_\Omega \bar{p}\ s_j\ d\Omega + \int \bar{q}\ s_j\ d\Gamma\right\}^T \tag{3.2c}$$

The possible sinks and sources $\bar{p}$ within the system, as well as flow conditions $\bar{q}$ are integrated in the equivalent flow vector $\underset{\sim}{g}$.

The B-matrix and the permeability tensor K which constitute the hydraulic "stiffness" P are given as:

$$B = [\frac{\partial s_i}{\partial x_j}] \tag{3.2d}$$

$$K = [K_{ij}] \tag{3.2e}$$

Usually the permeability coefficients can be at best determined along their principal directions. However, for a coordinate system not coinciding with the principal directions of flow, Fig. 3.1, a transformation is necessary, namely:

$$K = A^T \bar{K} A \tag{3.3a}$$

where $\bar{K}$ is the principal permeability tensor and A the transformation matrix.

In the common case of 2-D flow, this matrix simplifies to:

$$K = \begin{bmatrix} K_{11} & 0 \\ 0 & K_{22} \end{bmatrix} \tag{3.3b}$$

$$A = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{3.3c}$$

Furthermore, when parabolic isoparametric elements are employed, the integration of the first term of Eq. 3.2c can be reformulated in terms of local coordinate ξ for Gaussian quadrature calculations, Fig. 3.2.

$$\int_\Gamma \bar{q} \cdot s_j \, d\Gamma = \int_{-1}^{1} \bar{q} \, s_i \left[\left(\frac{\partial x}{\partial \xi}\right)^2 + \left(\frac{\partial y}{\partial \xi}\right)^2 \right]^{1/2} d\xi \tag{3.4a}$$

since

$$d\Gamma = \left[(dx)^2 + (dy)^2 \right]^{1/2} \tag{3.4b}$$

and

$$dx = \frac{\partial x}{\partial \xi} \cdot d\xi \tag{3.4c}$$

$$dy = \frac{\partial y}{\partial \xi} \cdot d\xi \qquad (3.4d)$$

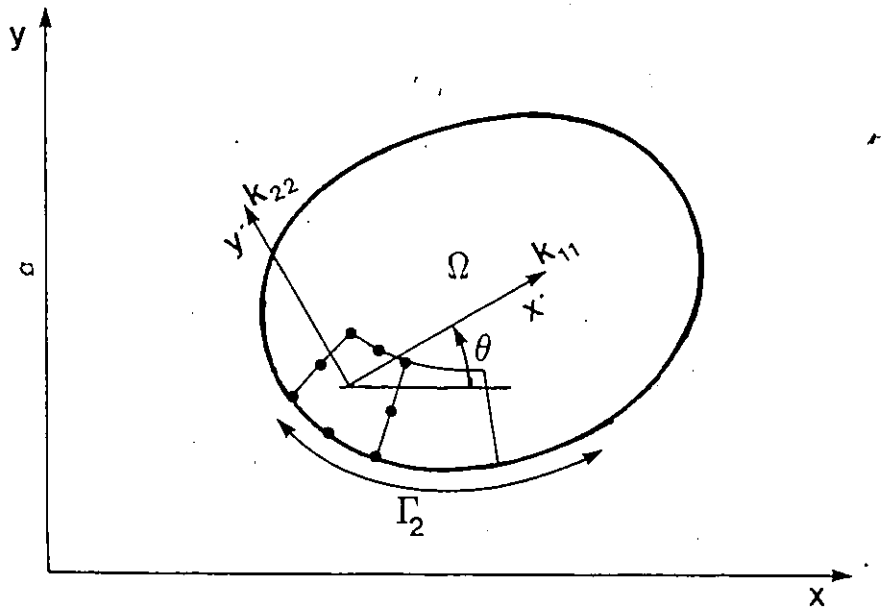


Fig. 3.1 Element with principal directions of permeability different from X and Y axes

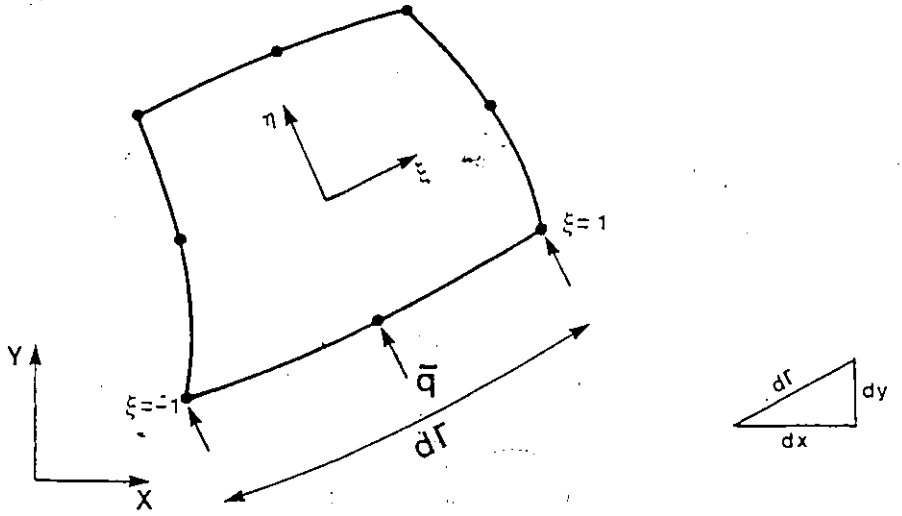


Fig. 3.2 Imposed flow conditions across boundary

3.3 CALCULATION OF THE VELOCITY

Once the nodal total heads are determined from the solution of Eq. 3.2a, the fluid velocity may be evaluated through Darcy's law:

$$\underset{\sim}{v} = - K \text{ grad } h_o = K \, B \, \underset{\sim}{h_o} \qquad (3.5)$$

where $\underset{\sim}{v}$ is the fluid velocity array.

Analogous to the theoretical study performed by Barlow (1976), the above expression gives the exact solutions only at the Gauss integration points. Instead, engineers are often interested in calculating the velocity at the nodes. Inspired by previous works (Hinton and Campbell 1974, Hinton et al. 1975 and Reed 1982), a numerical technique for extrapolating the exact value of the velocity at Gauss integration points to the nodal locations has been herein elaborated.

In the assumption of discretizing the total head function h_o by means of 4-or-8-node isoparametric elements, the resultant velocity may be either constant or linear within the element. In both cases, therefore, the values of velocity at any point inside the element belong to a plane which also contains the exact values at the Gauss integration points.

Similarly to Reed (1982), the interpolating function for velocity is assumed to take the following form:

$$v(\xi,\eta) = c_1 + c_2\xi + c_3\eta + c_4\xi\eta \qquad (3.6)$$

The coefficients c_1, c_2, c_3 and c_4 may be determined by replacing into Eq. 3.6 the values of the velocity obtained from Eq. 3.5 and the corresponding Gauss integration point locations (ξ, η).

$$\underset{\sim}{v}_{(i)} = M_1 / \underset{\sim}{c}_{(i)} \qquad (3.7a)$$

$$\underset{\sim}{c}_{(i)} = M_1^{-1} \underset{\sim}{v}_{(i)} \qquad (3.7b)$$

where $v_{(i)}$ is the ith. component of the velocity at the Gauss integration point, M_1 is a 4 x 4 matrix whose components are given in Table 3.1, and $\underset{\sim}{c}_{(i)}$ the array of coefficients.

The nodal values $v^*_{(i)}$, therefore, can be readily calculated since the coefficients $\underset{\sim}{c}_{(i)}$ are known, in fact

$$\underset{\sim}{v}^*_{(i)} = M_2 \, \underset{\sim}{C}_{(i)} = M_2 \, M_1^{-1} \, \underset{\sim}{v}_{(i)} = M \, \underset{\sim}{v}_{(i)} \tag{3.8}$$

where M is a 8 x 4 matrix whose coefficients are reported in Table 3.1.

For nodal points in common with m adjoining elements, the velocity is evaluated as a mean of the contribution of each element, Fig. 3.3.

$$\overline{\underset{\sim}{v}}^*_{(i)} = \frac{\sum_{e=1}^{m} \underset{\sim}{v}^*_{(i)e}}{m} \tag{3.9}$$

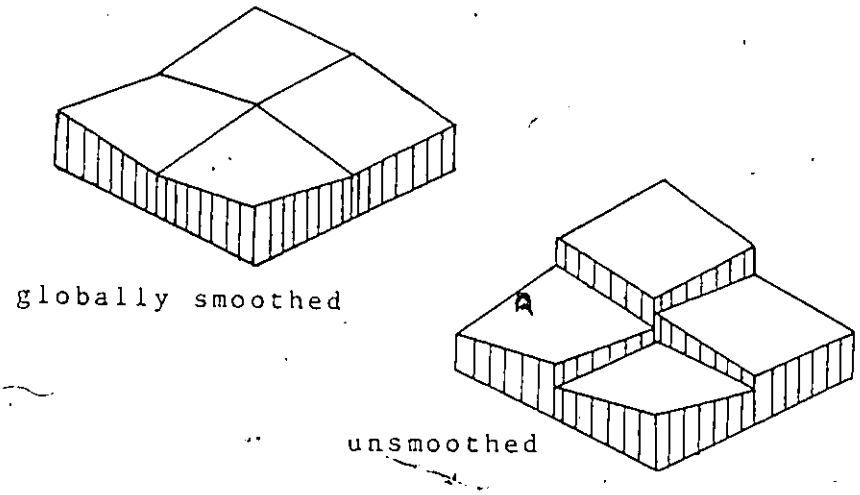


Fig. 3.3 Smoothing velocities over 4 adjoining elements

TABLE 3.1

Smoothing variables for 8-node isoparametric elements with 2x2 Gauss integration scheme

$$M_1 = \begin{bmatrix} 1 & -f & -f & f^2 \\ 1 & f & -f & -f^2 \\ 1 & f & f & f^2 \\ 1 & -f & f & -f^2 \end{bmatrix}$$

$$M_2 = \frac{1}{4} \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} ; \qquad M_2 M_1^{-1} = \frac{1}{2} \begin{bmatrix} -1 & 2e & -1 & 2d \\ a & b & b & a \\ 2d & -1 & 2e & -1 \\ a & a & b & b \\ -1 & 2a & -1 & 2b \\ b & a & a & b \\ 2e & -1 & 2d & -1 \\ b & b & a & a \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

where

$$a = \frac{1+\sqrt{3}}{2} \quad ; \quad b = \frac{1-\sqrt{3}}{2} \quad ; \quad d = \frac{2+\sqrt{3}}{2} \quad ; \quad e = \frac{2-\sqrt{3}}{2}$$

3.4 ALGORITHM

The solution of the confined flow does not entail any major difficulty as previously discussed in Sect. 3.2. On the other hand, in the analysis of the unconfined flow, an iterative process is required to locate the free surface and the hydraulic "stiffnesses" are recalculated at each iteration since the geometry is changing.

As previously reported, most of the known algorithms in the conventional methods are unable to keep a well graded distribution of elements during the iterations. Also, the method proposed by Finn and Varoglu (1976) appears to be very costly since all the nodes of the domain have to be updated.

The alternative method implemented in this present work is based on the definition of large zones of homogeneous materials by means of isoparametric elements denoted as Big Elements (BE). A number of those BE found below the free surface are made active and termed as Dynamic Big Elements (DBE) in contrast to the remaining inactive BE, Fig. 3.4. The BE are further discretized into a number of sub-elements specified by the user. This is accomplished by the use of an efficient automatic mesh generation program.

The prime advantage of this method is that after the first iteration in which both BE and DBE have been discretized, only part of the entire mesh, i.e. exclusively the DBE, are revised. In this situation, the updating

process is restricted to the nodal coordinates of the DBE and the use of the automatic mesh generation guarantees a good distribution of elements throughout the mesh. In addition, the sub-element "stiffnesses" of the BE, which are generally in greater number than the DBE, are kept unaltered thus resulting in a considerable save in computer time. Hence, due to the simplicity of this approach, complicated geometries can be treated; for example the analysis of flow across two different vertical layers, is easily tackled as shown in Fig. 3.4, in which the free surface is "discontinuous" (part B-E).

However, the major problem at this point is the establishment of an adequate law for updating the DBE nodes.

3.4.1 Method of shifting DBE nodes

In the present F.E. implementation, two basic types of updating procedures have been considered. The first one is concerned with the movement of the DBE nodes, with the only constraint of respecting the geometrical boundaries of the regions. The second type of movement has been elaborated to cope with the presence of a seepage surface within the medium.

In the first procedure, the nodes on the free surface are allowed to move only along the lateral edges of the DBE. This generally allows to respect both geometry and material properties constraints. The nodes move such that the

condition of zero pore pressure is satisfied on the free surface. The mid vertical side nodes are consequently updated in order to ensure a good distribution of sub-elements, Fig. 3.5. When the top edge nodes belong to a physical boundary, they are obviously forced to move along it, Fig. 3.6.

A seepage surface normally introduces a special algorithm for its simulation and consequently local modifications of the previous procedure are required to give rise to the second type of node movement. For convenience and easy presentation of fundamental ideas of the procedure, reference is made to Fig. 3.7. The DBE are numbered in a sequence such that the ones found below the free surface are considered as the first six DBE, the nodes on the free surface are updated according to the first procedure. Then the top nodes of DBE 7, 8 and 9 are adjusted following a geometrical regression law in order to guarantee a smooth transition.

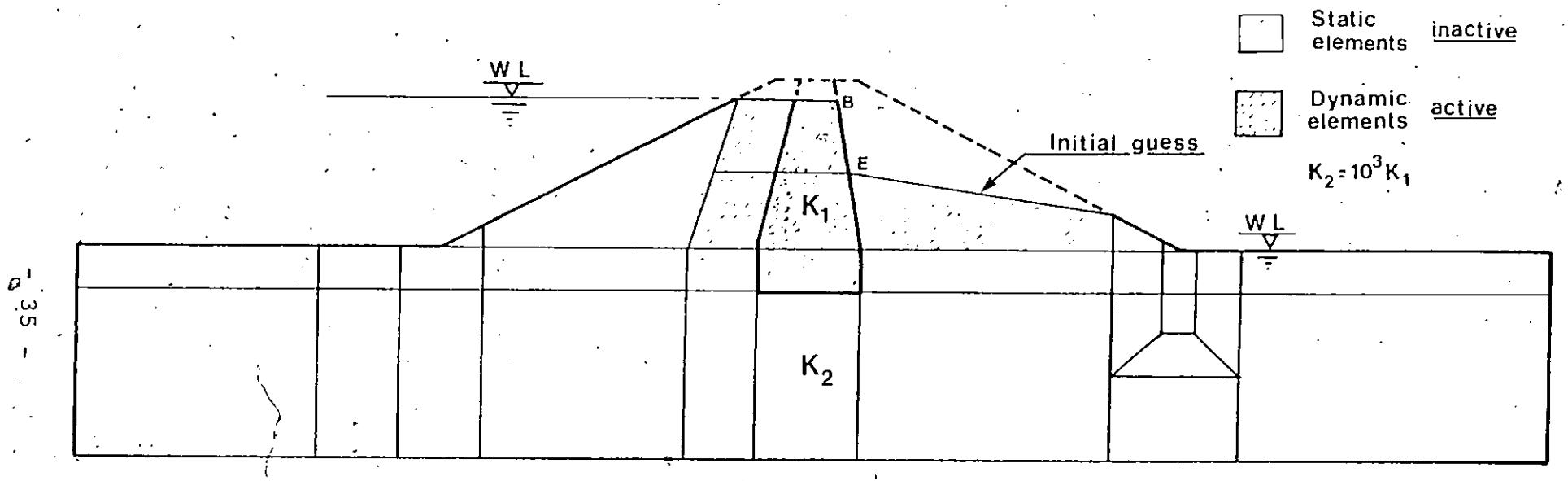


Fig. 3.4 Subdivision into active (DBE) and (BE) zones in a foundation dam

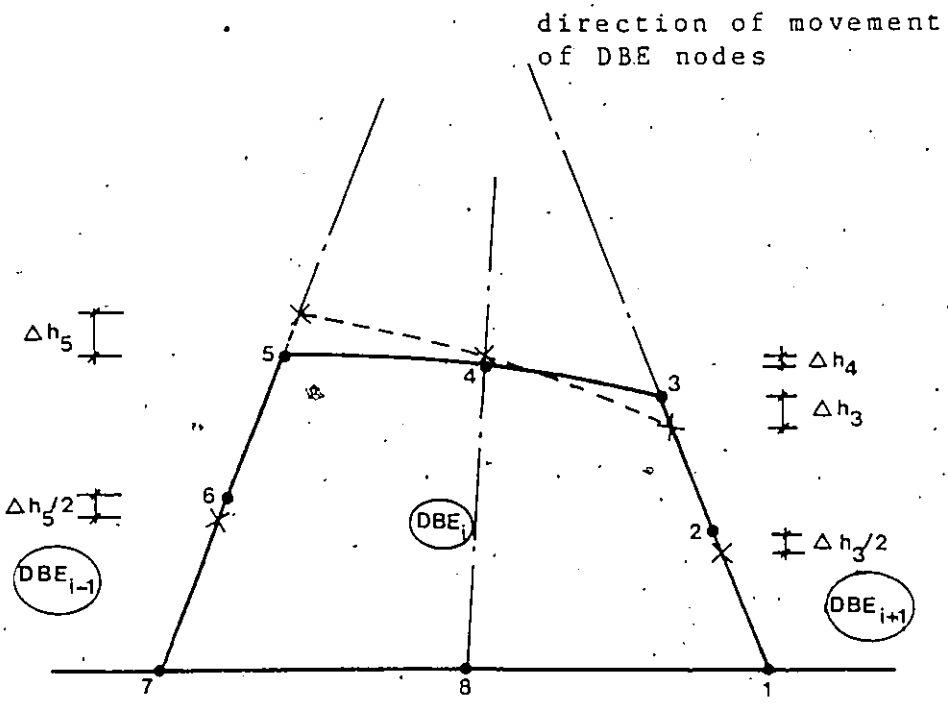


Fig. 3.5 Movement of nodes at boundary between 2 different materials

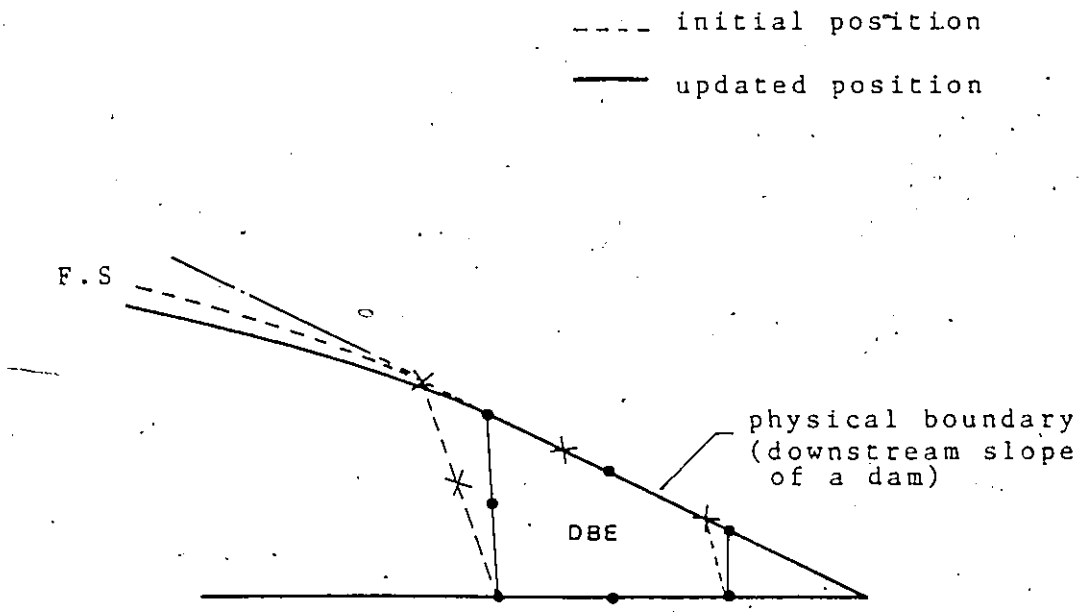


Fig. 3.6 Movement along a physical boundary (downstream slope of a dam)

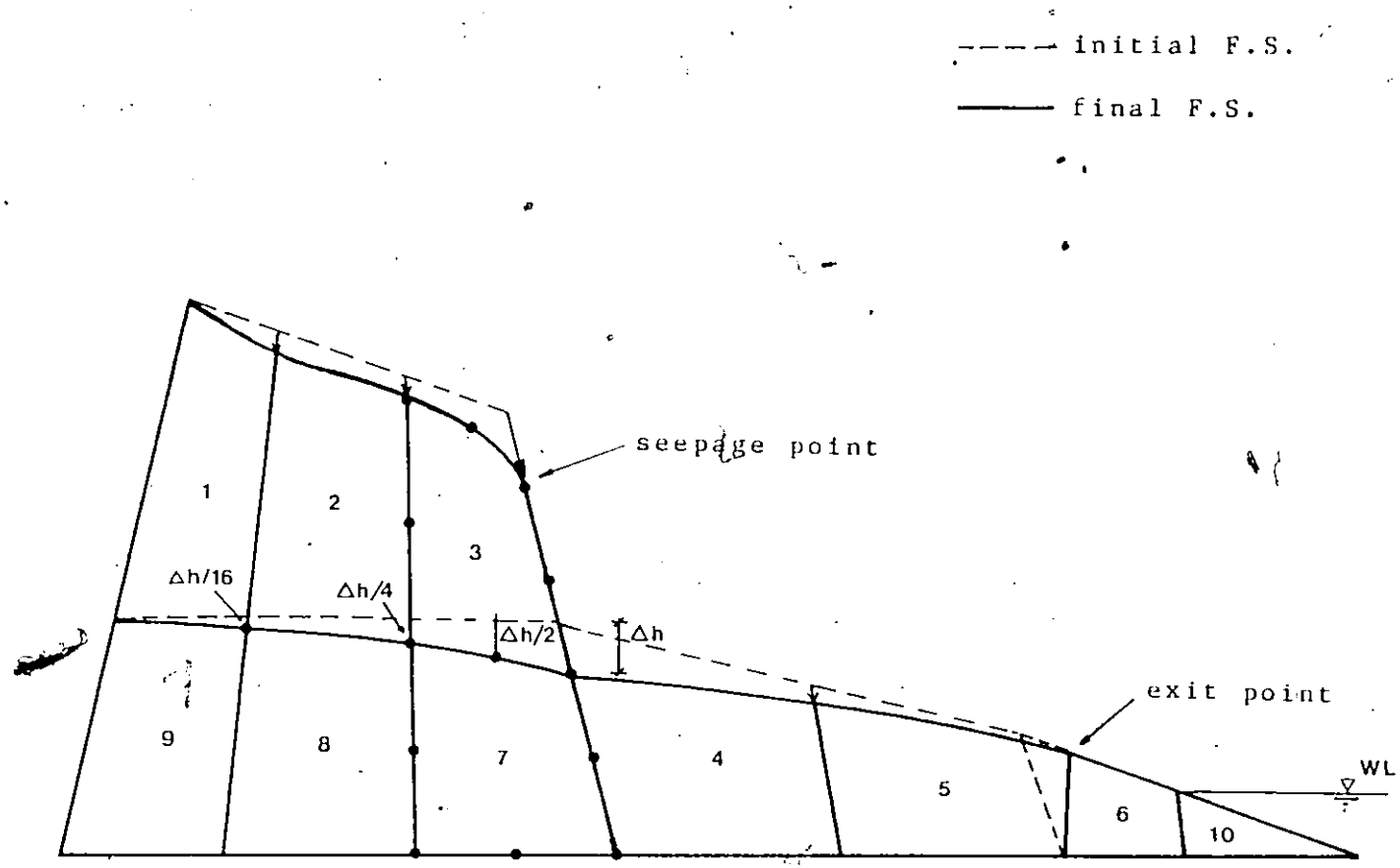


Fig. 3.7 Collapse of DBE during the iterative process

3.5 OVERALL STRUCTURE OF SEEP

The implementation of the seepage analysis was performed adopting a modular approach and is illustrated in the flowchart in Fig. 3.8.

After presetting the array dimensions and introducing the geometrical data in Subroutines DIMEN and INPUT respectively, the equivalent flow $\underset{\sim}{q}$ is calculated according to Eq. 3.2c in Subroutine LOADPS. Because of an eventual iterative process, a number of arrays are initialized in Subroutine INITIA.

The element hydraulic "stiffnesses" are then calculated in Subroutine STIFFP according to Eq. 3.2b after which the current solution is searched in Subroutine FRONT. For confined flow, the computation ends by the calculation of the velocity at Gauss integration points and at the nodes. These are done in Subroutines GAUVEL and NODVEL respectively.

In the case of unconfined flow, convergence of the iterative process is checked in Subroutine CONVER by verifying that all the total heads on the top nodes of the DBE are equal to their corresponding elevation. If convergence is reached, then the velocities at the Gauss integration and nodal points are calculated.

Instead, in the event convergence is not yet achieved, the free surface is moved and the dynamic region subsequently updated in Subroutines MOVE and MESHGEN

respectively. Finally, the appropriate head fixities for the new sub-element nodal points are adjusted where necessary in Subroutine HEDFIX. At this point, the program is ready to undertake a new solution.

Whenever the computations are complete, the numerical results are printed in Subroutine OUTPUT.

The only subroutines which show some distinction with respect to the others are MOVE, MESHGEN, HEDFIX and NODVEL. The others are either trivial or require minor changes with respect to available standard subroutines. It is for these reasons that the additional subroutines are explained in Appendix A, with the exception of MESHGEN, which is instead presented in Chapter 7.

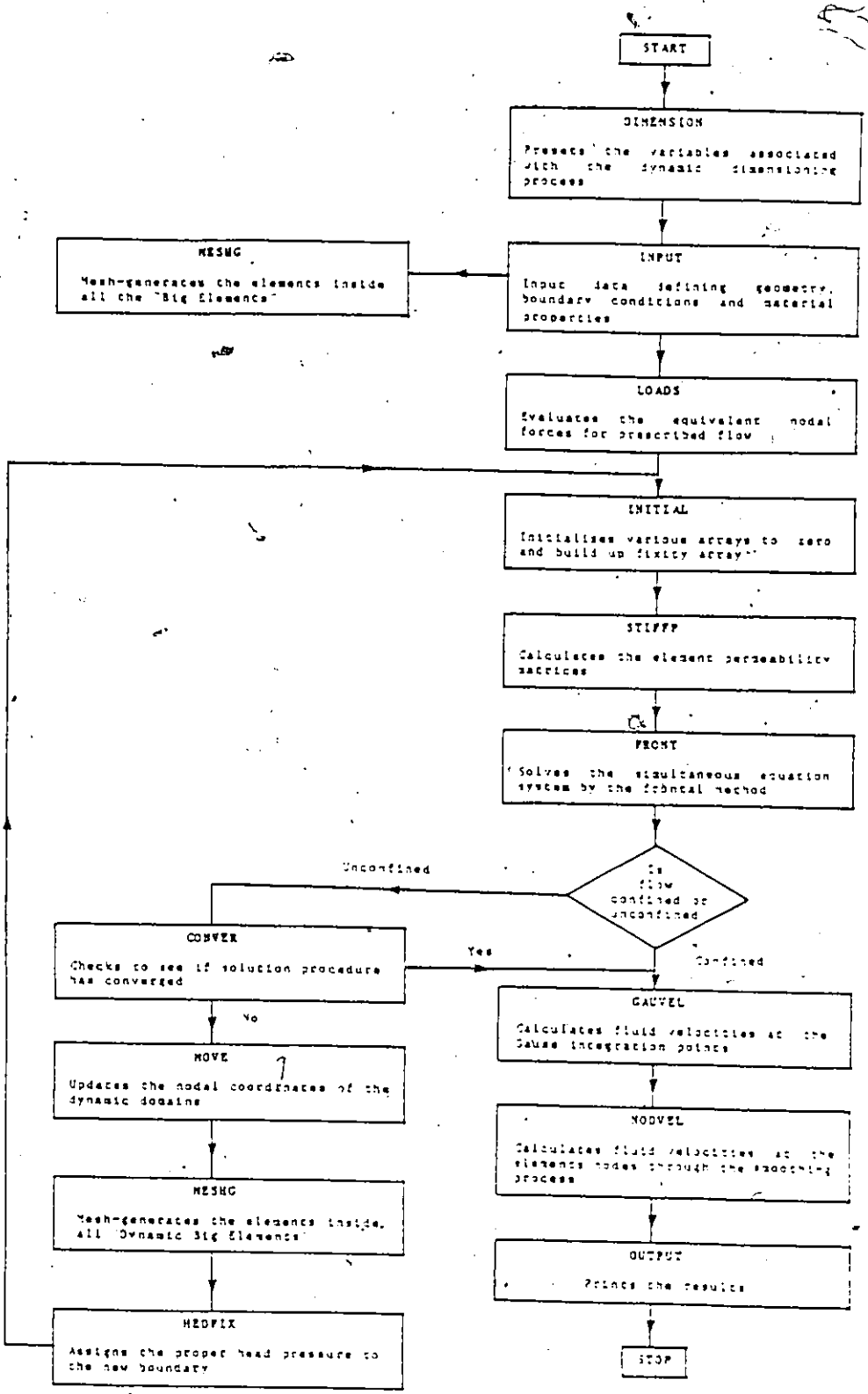


Fig. 3.8 Flowchart SEEP

3.6 NUMERICAL EXAMPLES

The first numerical example which typifies steady confined seepage is illustrated in Fig. 3.9a showing an impervious sheet-pile wall driven into a porous isotropic soil foundation. The heads at the upstream and downstream sides were assumed to be constant and equal to 10 and 2 respectively and the flow occurred only through the foundation soil.

Fig. 3.9b shows the plot of the velocity field as computed from the analysis. The program consumed only 3.5 CPU thus showing its efficiency. Unfortunately no closed form solution was available for comparison of results so that they can only be appreciated qualitatively.

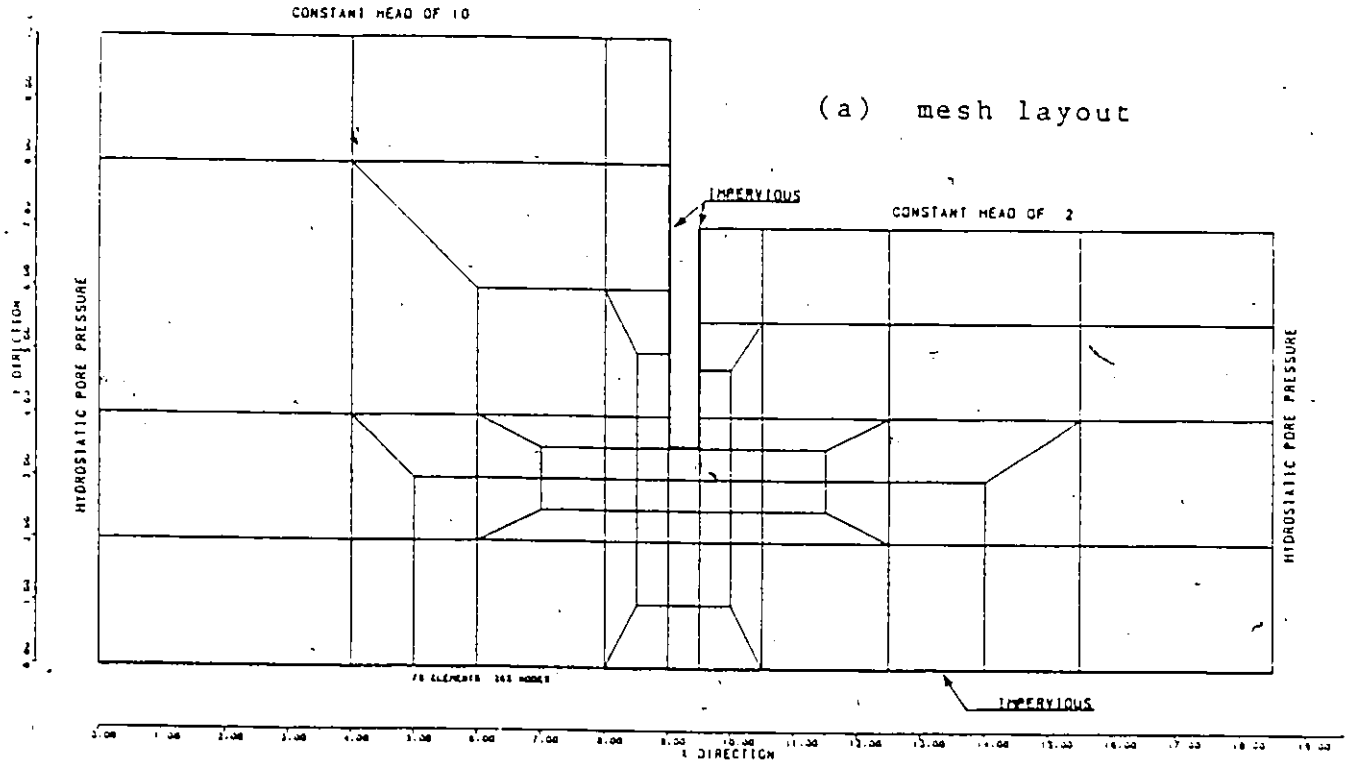
The second example considered was unconfined seepage through a homogeneous dam with vertical upstream and downstream faces for which analytical solutions exist, (Polubarinova-Kochina, 1962). With reference to Fig. 3.10a, the starting free surface was assumed to be horizontal. The analysis required 14 iterations and 3.3 CPU to converge to the true phreatic surface. Fig. 3.10b illustrates the corresponding velocity field.

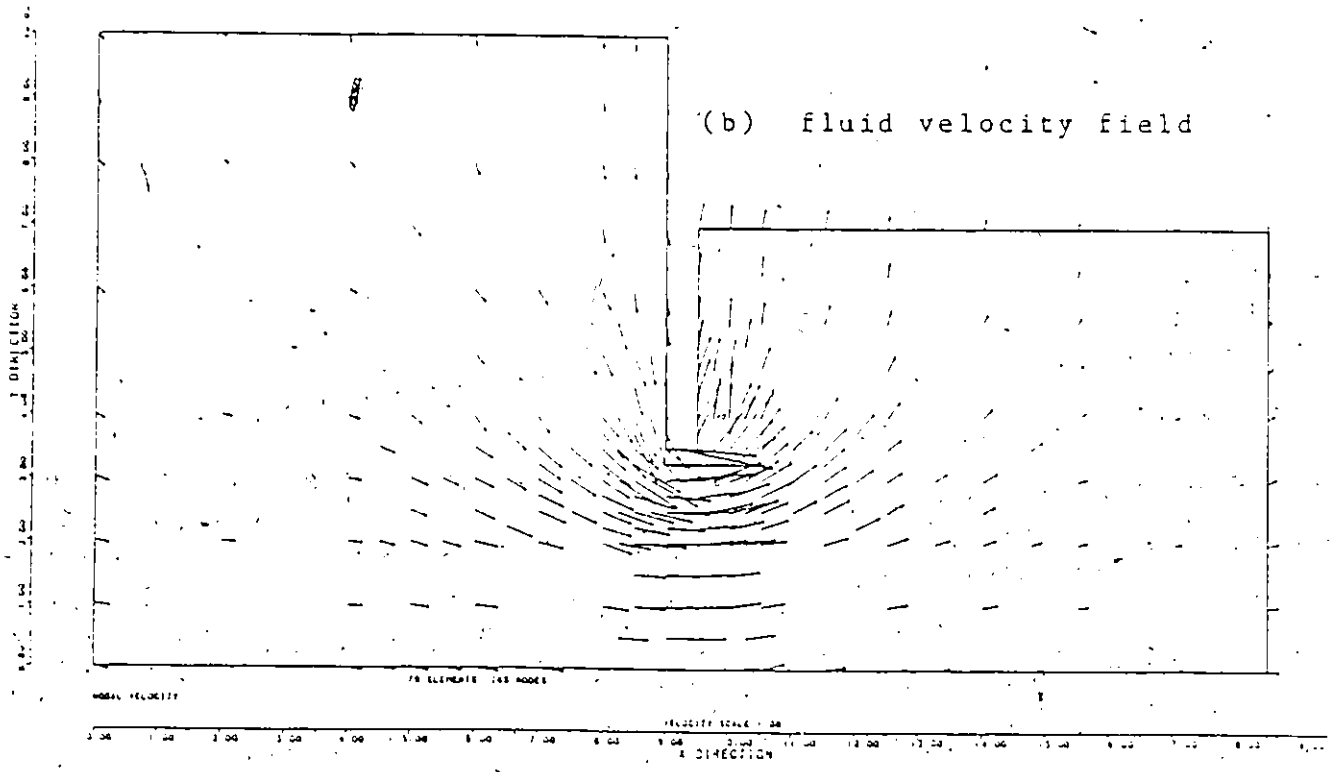
An exit head Hc of 1.44 was computed and complete agreement was found with the available analytical solution. This comparison is very favourable to the present finite element analysis. Also, cases of flow through an anisotropic homogeneous rectangular dam ($K_x = 0.1K_y$ and $K_x = 10K_y$) were

investigated showing no apparent problems in computation (see Figs. 3.11 and 3.12).

Finally the program was applied to the more general case of flow through a clay-core earth dam shown in Figs. 3.13. In this problem both the upstream and downstream boundaries had specified constant heads. The bottom of the foundation was considered impervious with water seeping through the pervious dam and its foundation. In the iterative procedure, an initial location of the free surface was estimated as shown in Fig. 3.13a and 35.7 CPU were spent during 8 cycles of iteration to converge to the correct free surface. The first iteration which comprises the computation of all the element stiffnesses consumed 7.3 CPU while the subsequent cycles only 3.7 CPU, showing a saving of about 50 % of the total time.

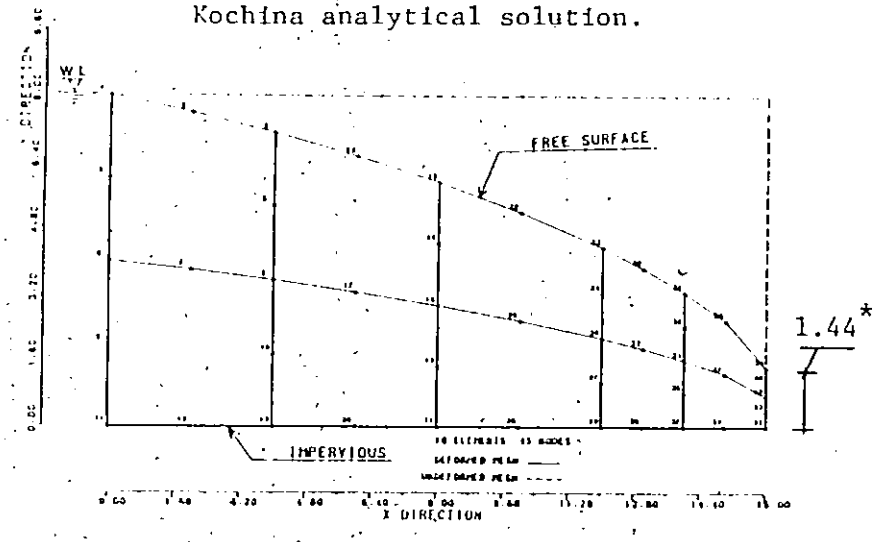
It is further noticed that the final mesh has preserved a uniform mesh distribution, a prerequisite for obtaining accurate results. Finally Fig. 3.13b shows the distribution of velocity in the dam and its foundation.





Figs. 3.9 Seepage below a sheet pile in a homogeneous soil

* coincides with Polubarinova-Kochina analytical solution.



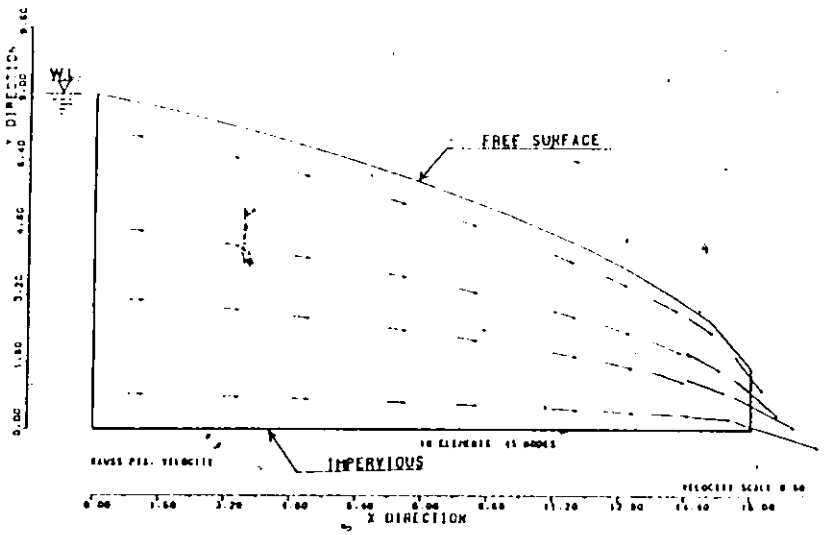


Fig. 3.10 Seepage through an isotropic rectangular dam ($K_x=K_y$): (a) final mesh, (b) fluid velocity field

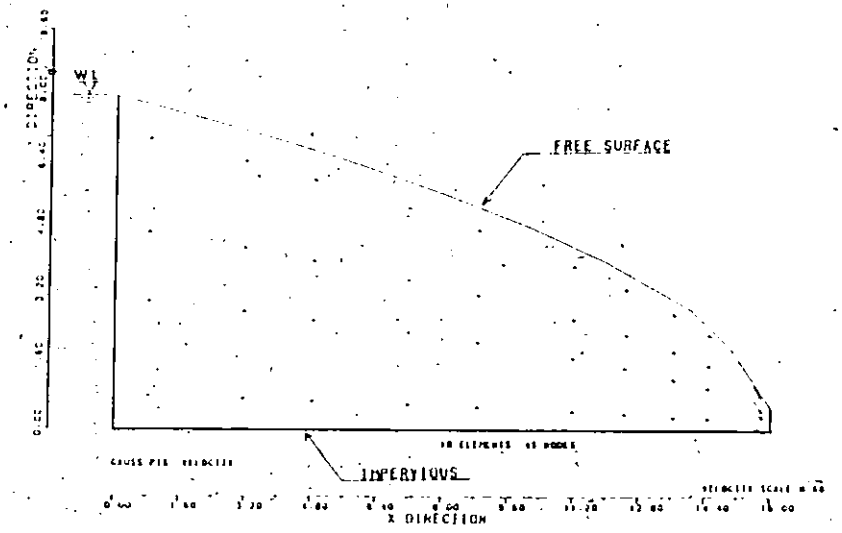


Fig. 3.11 Seepage through an anisotropic rectangular dam ($K_x=0.1K_y$): fluid velocity field

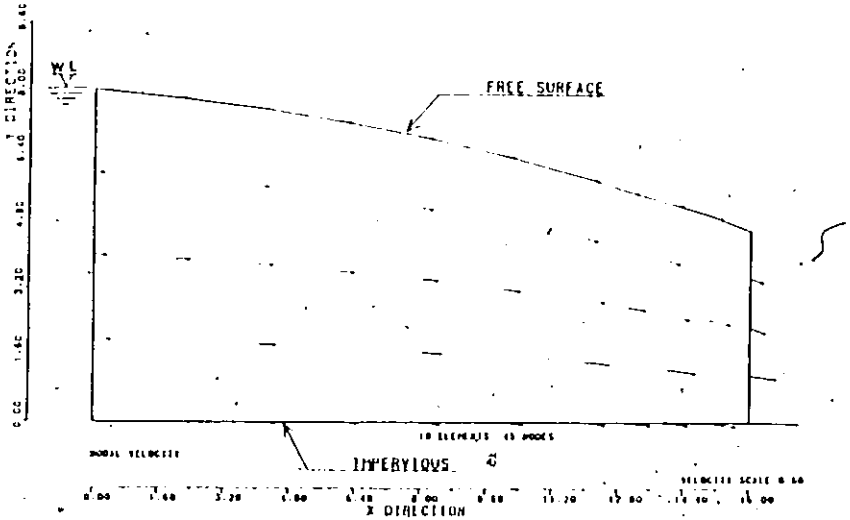


Fig. 3.12 Seepage through an anisotropic rectangular dam ($K_x=10\ K_y$): fluid velocity field

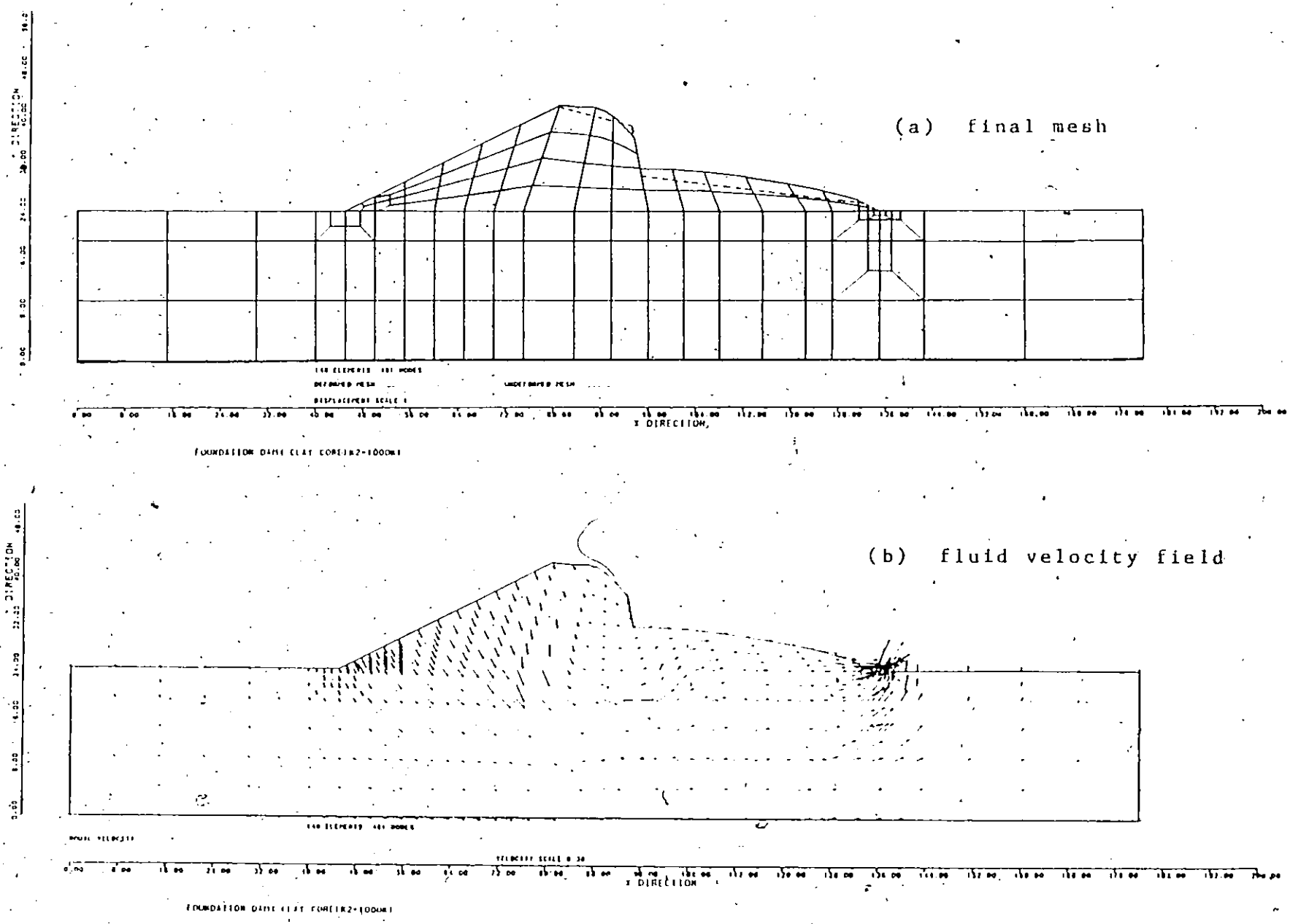


Fig. 3.13 Seepage through a heterogeneous dam

3.7 CONCLUSION

The new algorithm implemented for solving Laplace's equation over fixed and varying regions of definition can be considered very efficient and reliable. The fundamental difficulties inherent in such numerical solution have been overcome by the combined use of an effective mesh generator and a strategy for updating the working mesh.

The results from the unconfined flow analyses were compared with closed form solutions whenever possible. It was found that in the case of seepage through a homogeneous rectangular dam, the computed results completely agreed with the analytical solution (Polubarinova-Kochina, 1962). Similar results and accuracy were provided by the present program in the more complicated case of flow in a clay-core foundation dam, which introduces an internal seepage surface. Else, no particular numerical problems were encountered when treating anisotropic flow.

As regard computer processing time and solution convergence, the algorithm proved to be very economical requiring only a few cycles of iteration for the location of the free surface. This aspect of SEEP may have a positive implication for the analysis of more complicated and practical cases.

Another feature of the program lies in its capability of calculating the velocity of flow at the nodal points besides at the Gauss integration points. This is quite often more

useful in a practical flow analysis. The adaptation of the "smoothing" technique to the seepage problem has proved to be successful as reveal the results on nodal velocities. Also, SEEP is a prelude for the problems such as the uncoupled consolidation and effective stress analysis coming in the next chapters. In this latter case, SEEP provides the nodal values of pore pressures taken as input hydraulic loads for an effective stress analysis.

Finally, it is considered that the application of SEEP to the practical case of a geotechnical problem seems to pose no major difficulty.

Chapter IV

UNCOUPLED CONSOLIDATION THEORY

4.1 INTRODUCTION

The uncoupled consolidation theory leads to a parabolic differential equation similar to the one governing heat conduction problems.

In spite of the fact that a number of finite element programs have been implemented for the numerical analysis of the latter problem, very few have been elaborated for the uncoupled consolidation analysis. Among them, Desai and Johnson (1972) solved the 1-D problem for layered soils employing linear and parabolic 1-D elements. The analysis in this case required high computation time and accuracy was not always achieved, most probably due to the algorithm and the solution technique therein employed.

Gioda and Nova (1982) apparently extended to 2-D the previous works of Desai and Johnson but unfortunately, no detailed information on the type of element and on the results obtained could be found.

In this research, a new F.E. code for analysing the current problem in 2-D and axisymmetric conditions has been implemented on the basis of the novel algorithm recently formulated by Fusco (1985).

4.2 FINITE ELEMENT DISCRETIZATION OF UNCOUPLED CONSOLIDATION EQUATIONS

The basic F.E. equation governing the uncoupled consolidation theory, as previously reported, is

$$\text{div } K \text{ grad } \frac{\pi^e}{\gamma_f} = \frac{1}{B} \dot{\pi}^e \tag{4.1}$$

together with the following boundary conditions:

$$\pi^e = \pi_o^e \quad \text{in} \quad \Omega \quad \text{at} \quad t = 0, \tag{4.2}$$

$$\pi^e(t) = \bar{\pi}^e(t) \quad \text{on} \quad \Gamma_1 \tag{4.3}$$

$$\frac{K_n}{\gamma_f} \frac{\partial \pi^e}{\partial n}(\Gamma_2, t) = -\bar{q}_n \quad \text{on} \quad \Gamma_2 \tag{4.4}$$

n is the normal to the boundary Γ_2.

The field variable π^e is discretized by means of an appropriate shape function as follows:

$$\pi^e(t) = \underset{\sim}{\bar{s}}^T \underset{\sim}{\pi}^e \tag{4.5}$$

where $\underset{\sim}{\pi}^e$ contains the discrete nodal values of the excess pore pressures.

The straightforward Galerkin's method of weighted residuals applied to Eq. 4.1 leads to the following set of equations:

$$P\,\underset{\sim}{\pi}^e + D\,\dot{\underset{\sim}{\pi}}^e = \underset{\sim}{g} \qquad (4.6a)$$

where P and $\underset{\sim}{g}$ are respectively the same hydraulic "stiffness" matrix and equivalent flow vector previously defined in Chapter 3. The D-matrix is instead defined as:

$$D = \left[\int_\Omega \frac{\gamma_f}{B} s_i s_j \, d\Omega\right] \qquad (4.6b)$$

At this point it was proposed to discretize the time domain by the θ-Wilson method which can be conveniently expressed as:

$$\underset{\sim}{\pi}^e = \theta\,\underset{\sim n}{\pi}^e + (1-\theta)\underset{\sim n-1}{\pi}^e = \theta\,\Delta\underset{\sim n}{\pi}^e + \underset{\sim n-1}{\pi}^e \qquad (4.7)$$

$$\frac{d\underset{\sim}{\pi}^e}{dt} = \frac{\Delta\underset{\sim n}{\pi}^e}{\Delta t} \qquad (4.8)$$

Substituting the above expressions into Eq. 4.6a and rearranging the terms leads to the following set of linear equations:

$$\overline{P}\ \Delta\underset{\sim}{\pi}^{e}_{n} = \Delta g'_{n} \qquad (4.9a)$$

where

$$\overline{P} = (\theta P + \frac{D}{\Delta t}) \qquad (4.9b)$$

$$\Delta g'_{n} = (\theta \Delta g_{n} + g_{n-1}) - P\ \underset{\sim}{\pi}^{e}_{n} \qquad (4.9c)$$

Once the excess pore pressure is determined from solution of Eq. 4.9a, the associated fluid velocity may be calculated for that time increment by:

$$v_{n} = - K \text{ grad } h = - K\ B\ \underset{\sim}{h}_{n} \qquad (4.10a)$$

where

$$\underset{\sim}{h}_{n} = \underset{\sim}{h}_{o} + \frac{\pi^{e}_{n}}{\gamma_{f}} \qquad (4.10b)$$

with $\underset{\sim}{h}_o$ as the initial total head.

4.3 STRUCTURE OF TERCON (TERZAGHI CONSOLIDATION)

A computer program, TERCON, has been written in order to implement the uncoupled consolidation. Fig. 4.1 illustrates the corresponding flowchart.

The uncoupled consolidation theory governed by Eq. 4.6a introduces a second matrix D to be integrated, besides the hydraulic "stiffness" matrix P defined in Chapter 3. Also, matrix $\bar{P}$ and array $\Delta g'$ need to be updated at every time step. From the flowchart, it can be seen that due to the modular approach of programming, a number of subroutines belonging to the program SEEP are employed. This eventually requires only some occasional modifications to be done.

The variables associated with dynamic dimensioning process are preset in Subroutine DIMEN. The input data defining the mesh, boundary conditions, material properties and time marching scheme parameters are then introduced in Subroutine INPUT. Subroutine LOADPS reads the prescribed boundary flow conditions and is in common to SEEP. Since the process is iterative, some arrays need to be initialized for accumulation of data. This is done in Subroutine ZERO. Subroutine STIFFP differs from its counterpart in SEEP, in so far as a new matrix D has to be integrated. The time step interval is predicted by Subroutine TMARCH before the matrix $\bar{P}$ and array $\Delta g'$ are updated in Subroutine INCREM.

The fluid velocity is finally calculated according to Eqs. 4.10 in Subroutine VELSEE which slightly differs from the Subroutine GAUVEL in SEEP. In fact, one additional statement performing the sum in Eq. 4.10b, to take into account the initial total head, has to be added.

At selected time steps, the numerical results are printed in Subroutine OUTPUT. Finally Subroutine CONVER monitors the number of nodes which have completed the dissipation process.

Under the hypothesis of constant bulk modulus and coefficient of permeability, matrices P and D are calculated only at the beginning of the first time increment. However, for sake of generality those matrices can be updated in Subroutine STIFFP.

Subroutines TMARCH and INCREM are explained in details in Appendix B since they are the only completely new subroutines introduced in this program. In fact the time stepping can be done according to four different schemes. One can march in the time space by means of constant intervals in a logarithmic scale with θ either fixed or function of time as follow:

$$\Delta t_n = t_{n-1} (\exp r - 1) \qquad (4.11a)$$

$$\Delta t_n = \log_e (t_n / t_{n-1}) \qquad (4.11b)$$

$$t_n = t_{n-1} + \Delta t_n \qquad (4.11c)$$

$$\theta \geq 1/2 \quad \text{constant with time} \qquad (4.11d)$$

The above equations constitute the first two options in Subroutine TMARCH. The third option consists in dividing the time domain into a number of big intervals which are subsequently subdivided into finer intervals. The scheme proposed by Sandhu (1977) is given in Table 4.1. Finally the last option permits the use of a linear constant time interval.

TABLE 4.1

Decomposition of the time domain (after Sandhu et al. 1977)

Steps	Δt	time domain
10	0.01	0. , 0.1
10	0.1	0.1 , 1.1
10	1.0	1.1 , 11.1
9	10.0	11.1 , 101.1
10	100.0	101.1 , 1101.1
8	1000.0	1101.1 , 9101.1

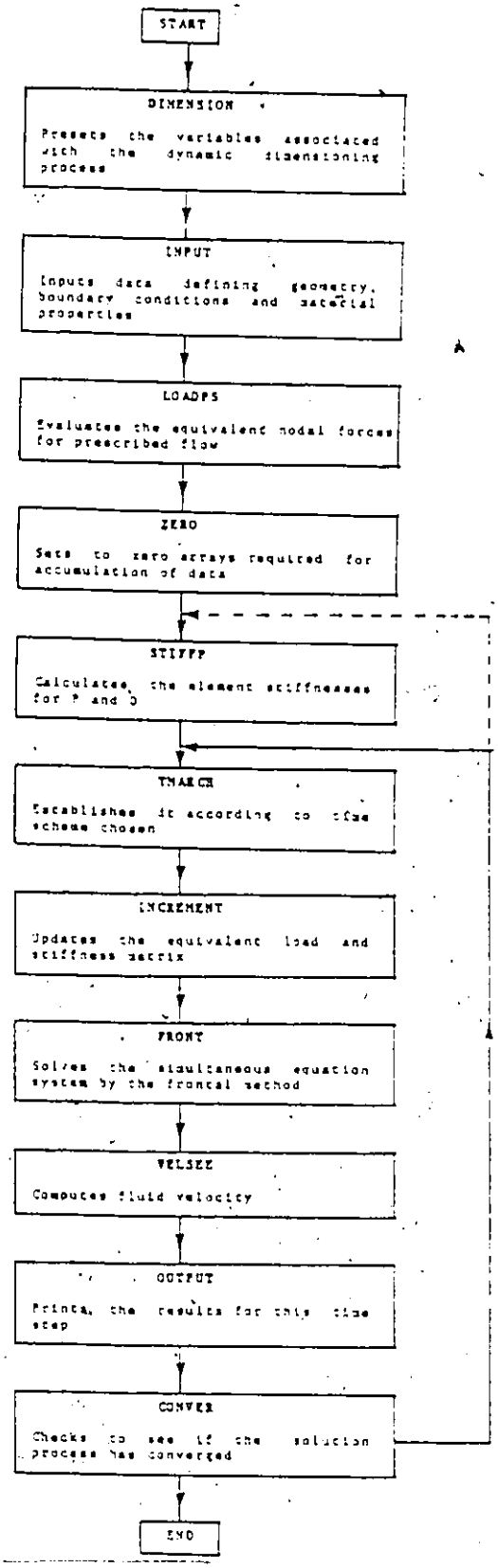


Fig. 4.1 Flowchart TERCON

4.4 NUMERICAL EXAMPLES

To illustrate the versatility and accuracy of TERCON, some simple examples were analysed in the cases of 1-D, axisymmetry and 2-D conditions. The corresponding mesh layouts are shown in Figs. 4.2a,b,c.

The numerical examples were carried out in the assumption of linear elasticity with constant bulk modulus during computation (see Eqs. 2.8). Table 4.2 summarizes the material parameters as well as the corresponding coefficients of consolidation C_v used throughout the numerical analysis. As regards time discretization, θ was assumed to be constant while the time interval was determined by Eq. 4.11a.

TABLE 4.2

Material parameters and coefficients of consolidation C_v for TERCON

Material properties

Young's modulus	E	=	10,000
Poisson ratio	ν	=	0.0
Permeability coefficient	$K_x = K_y = K$		4.0 E-06
Unit weight of fluid	γ_f	=	1.0

Coefficient of consolidation C_v

1-D consolidation

$$C_{v1} = \frac{K\ E\ (1-\nu)}{\gamma_f\ (1-\nu)\ (1-2\nu)}$$

2-D consolidation

$$C_{v2} = \frac{2\ G\ K}{\gamma_f} = \frac{E\ (1-\nu)\ K}{\gamma_f} \quad ; \quad G = \frac{E\ (1-\nu)}{2}$$

axisymmetric consolidation

$$C_{v3} = \frac{K}{\gamma_f\ B} = \frac{E\ (1-\nu)\ K}{3\ \gamma_f}$$

Time coefficient T_v

$$T_v = C_v\ t / H^2$$

where H is the drainage path. For the semi-infinite medium H is equal to the half width of the footing.

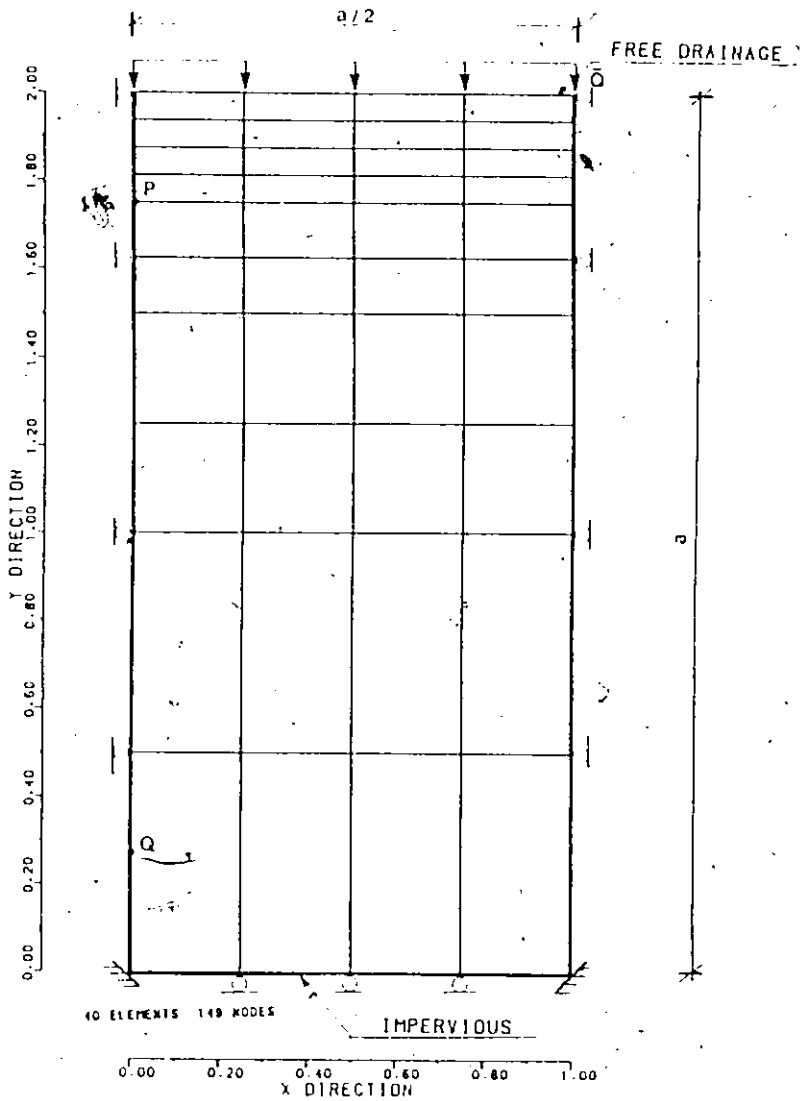


Fig. 4.2a. Mesh layout for 1-D conditions

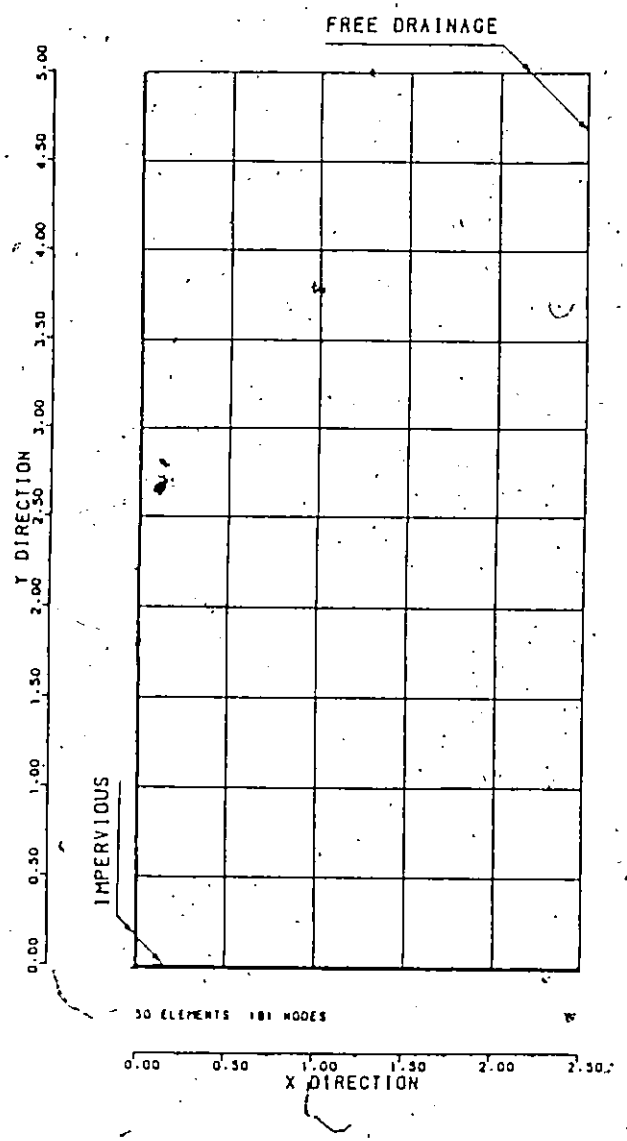


Fig. 4.2b Mesh layout for axisymmetric conditions

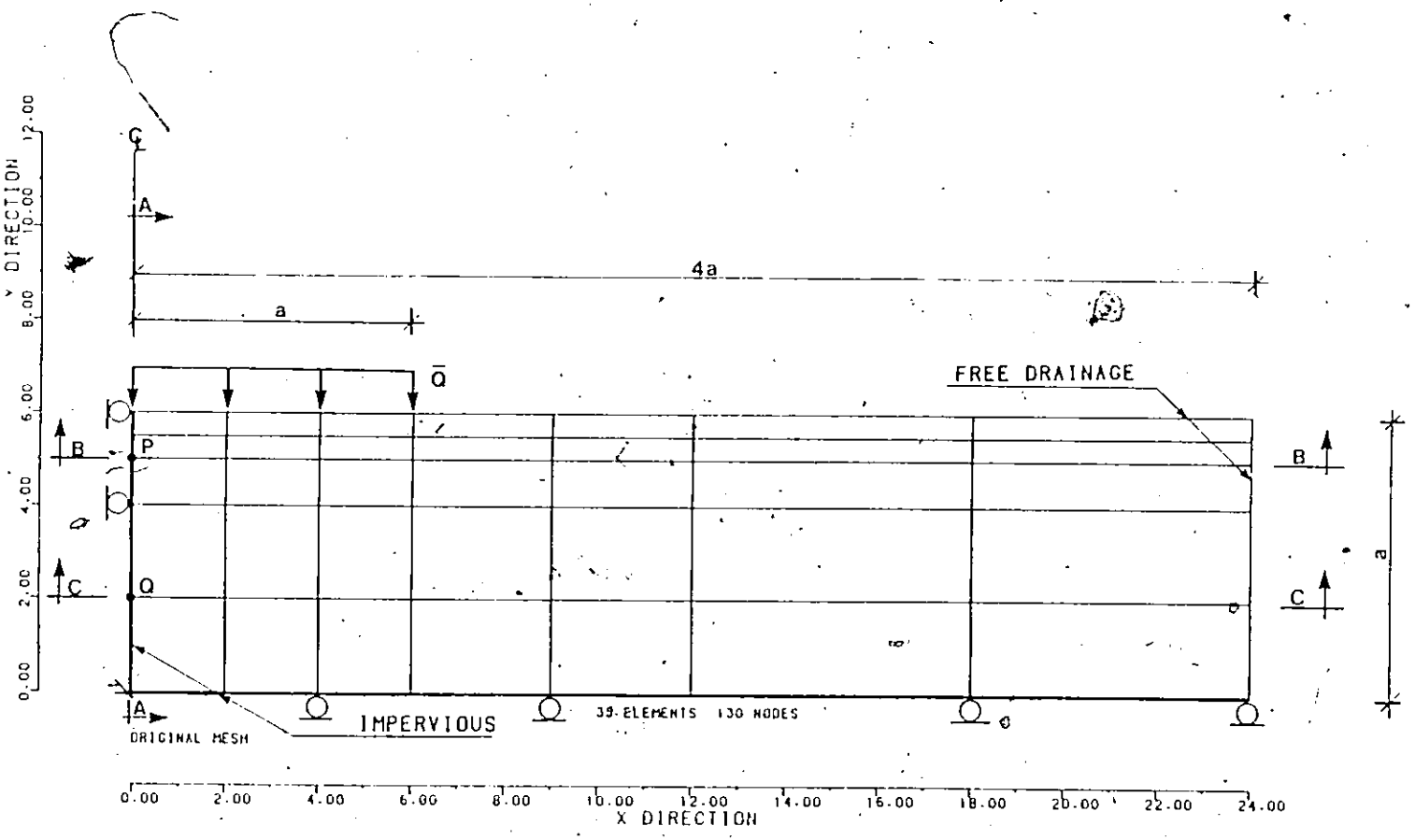
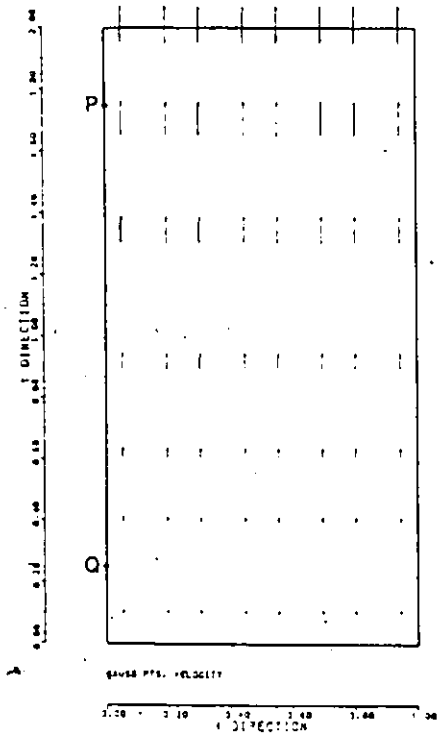


Fig. 4.2c Mesh layout for semi-infinite layer

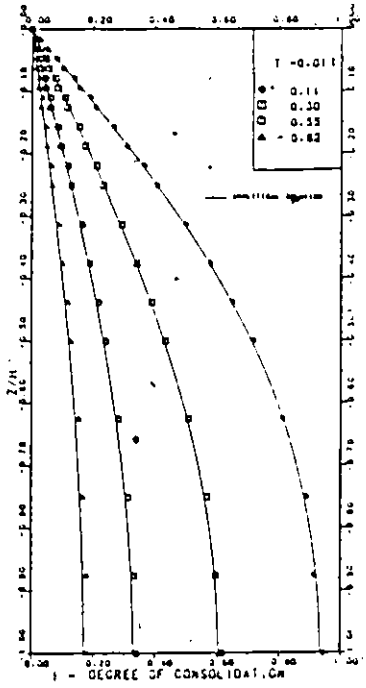
4.4.1 1-D consolidation

Fig. 4.3a shows the simplest case that TERCON can analyse. A soil sample closed both at its bottom and lateral surfaces with the only free-draining boundary as its top surface was considered. The flow regime is thereby illustrated at a particular time during the dissipation process.

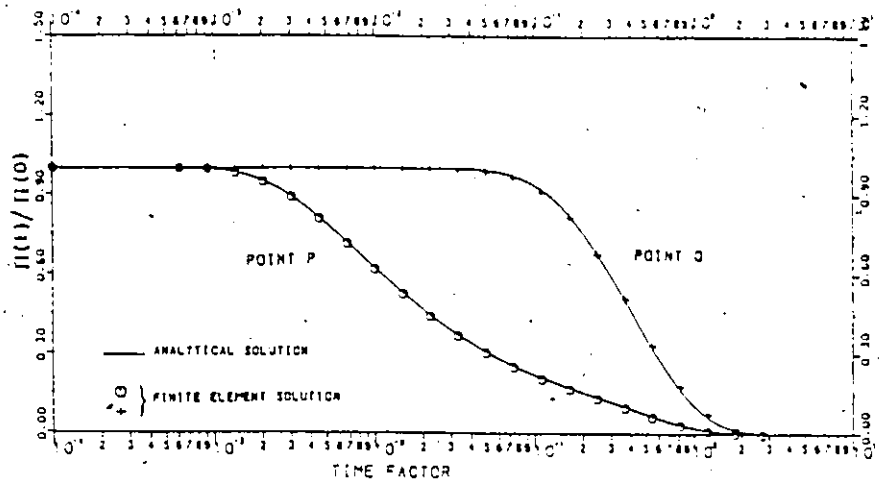
With reference to Figs. 4.3b,c which show respectively the isochrones at selected times and the pore pressure histories at points P and Q, complete agreement is obtained with the analytical solution of Taylor (1948). It is also worth to mention that in accordance to theoretical expectations, identical results were recovered in the axisymmetric case subjected to the very same boundary conditions as in 1-D. Both analyses required 38.1 CPU. The first time step which comprised the calculation of the matrices P and D, Eqs. 4.6a,b, consumed 1.46 CPU while only 0.86 CPU were needed for subsequent steps.



(a) fluid velocity field



(b) pore pressure isochrones (comparison with analytical solution)



(c) excess pore pressure vs. time (comparison with analytical solution)

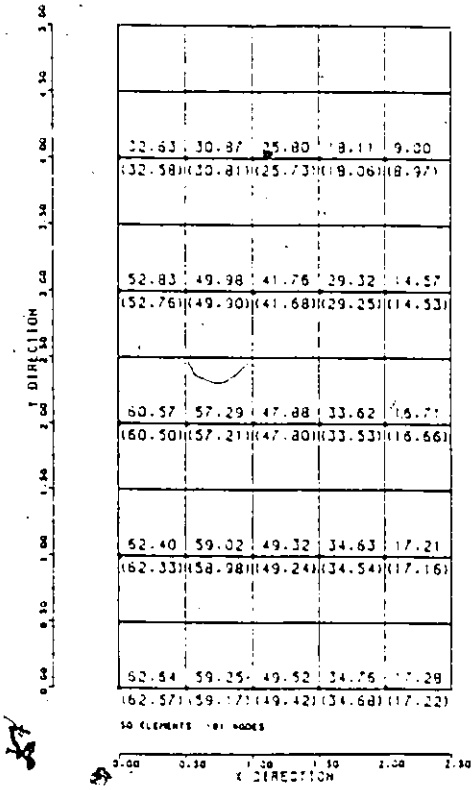
Fig. 4.3 1-D Uncoupled consolidation

4.4.2 Axisymmetric consolidation

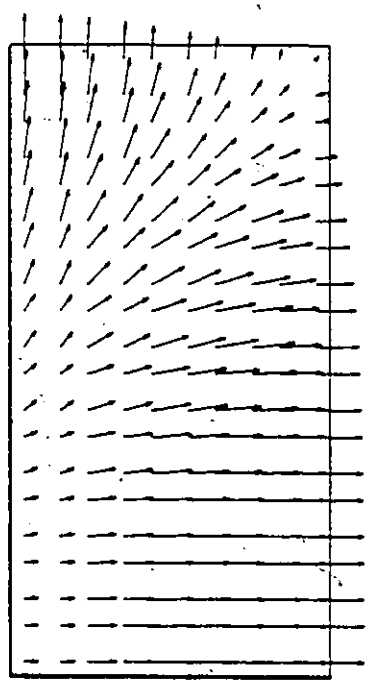
The example considered is a cylindrical soil sample draining on all its outer surfaces from an initial excess pore pressure of 100.0. The relevant hydraulic boundary conditions are given in Fig. 4.2b.

The F.E results were compared with the analytical solution of Carslaw and Jaeger (1959), and were found to be in very close agreement as Fig. 4.4a shows (the analytical values are in between parentheses). It was infact observed that the analytical computation demanded much more time than the numerical solution due to rather lengthy expansions of Bessel functions involved in.

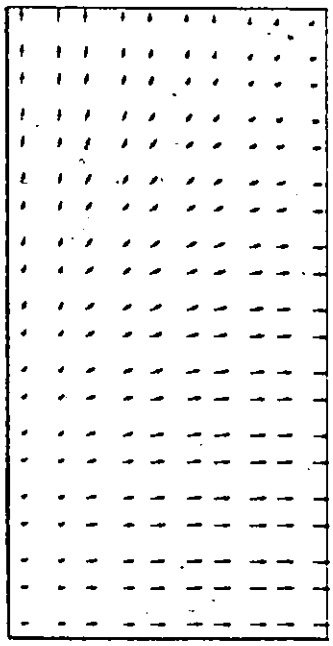
Finally Figs. 4.4b,c,d illustrate the progression of the fluid flow at different stages of consolidation.



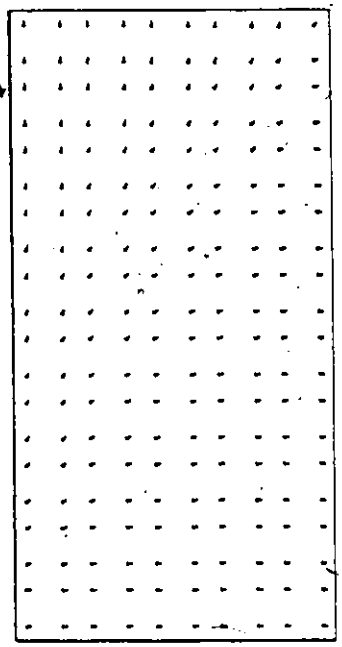
(a) excess pore pressure at t=1.0 (comparison with analytical solution)



(b) VELOCITY AT TIME 1.25



(c) VELOCITY AT TIME 2.50



(d) VELOCITY AT TIME 3.75

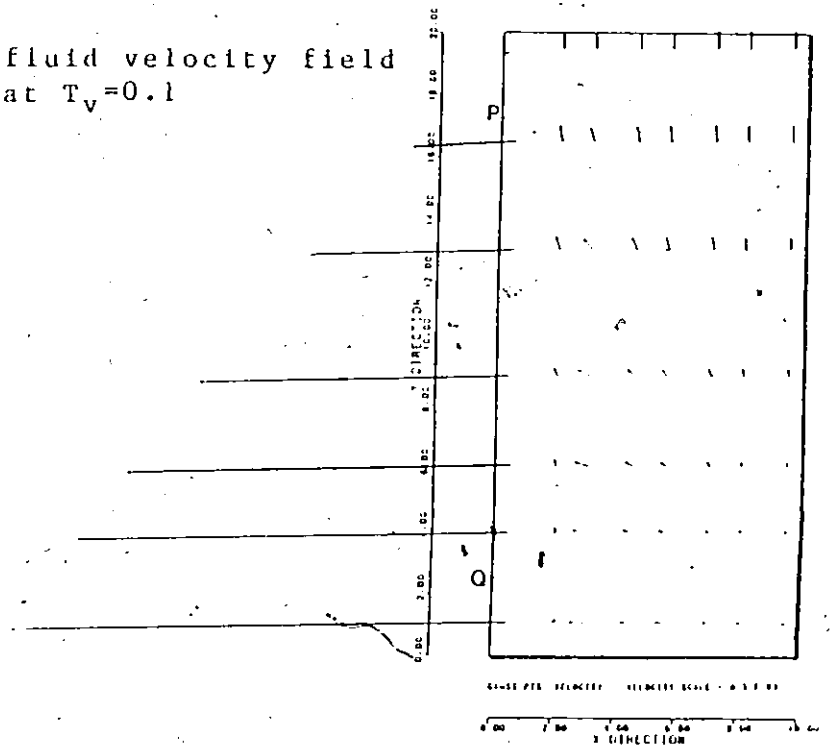
Fig. 4.4 Uncoupled consolidation in axial symmetry

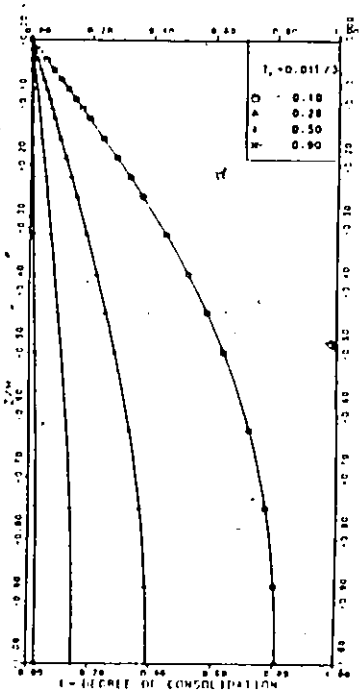
4.4.3 Sand drain simulation

One very interesting application of TERCON is the analysis of the performance of a vertical sand drain in a soil specimen. The mesh used is shown in Fig. 4.2a with its left lateral surface converted into the axis of symmetry and a free draining boundary.

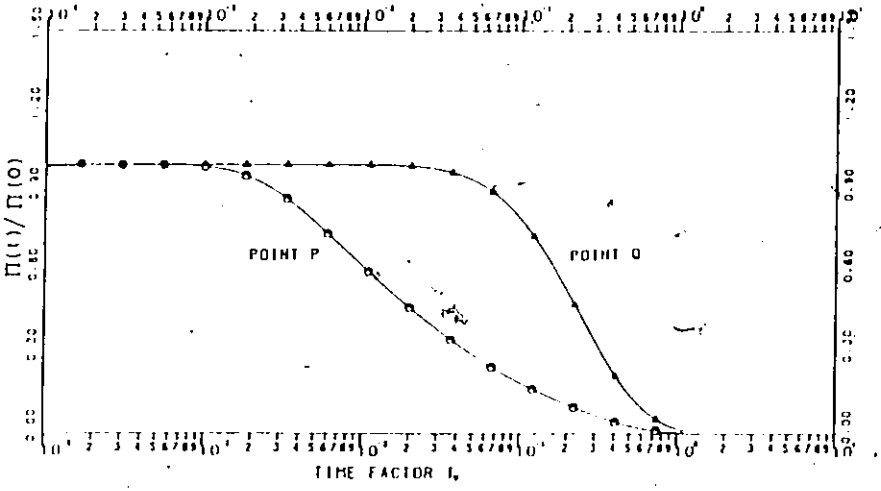
Two cases were examined; namely isotropic and anisotropic soil permeabilities. Figs. 4.5a and 4.6a illustrate different flow patterns inside the soil sample when considering anisotropy in permeabilities. In fact if the horizontal permeability is higher than the vertical one, flow tends to be more radial throughout the sample. Furthermore, it can be observed from the isochrones and pore pressure histories at points P and Q in Figs. 4.5b,c and 4.6b,c that the rate of consolidation is faster in anisotropic conditions.

(a) fluid velocity field at T_v=0.1





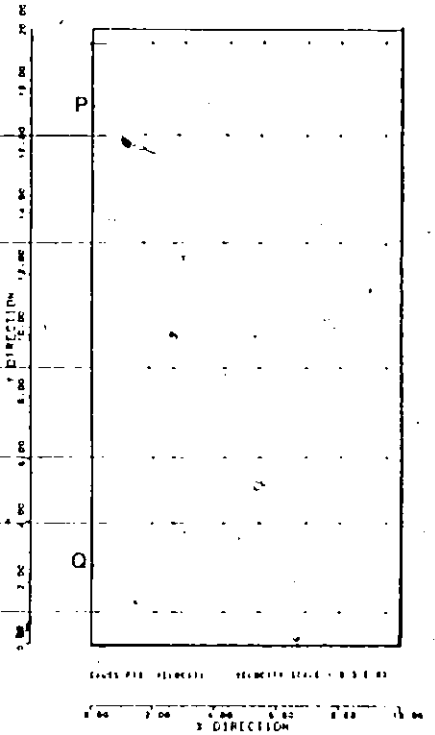
(b) excess pore pressure isochrones

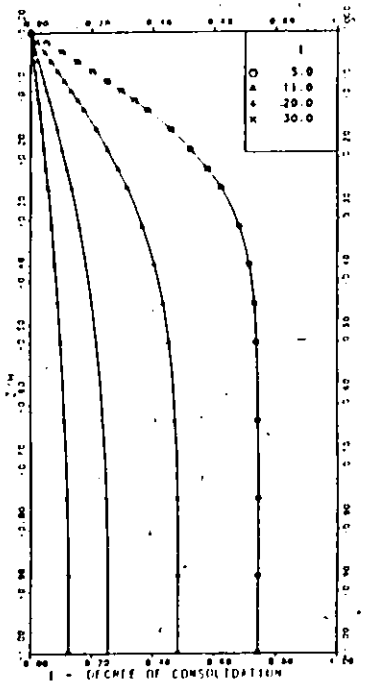


(c) excess pore pressure vs. time

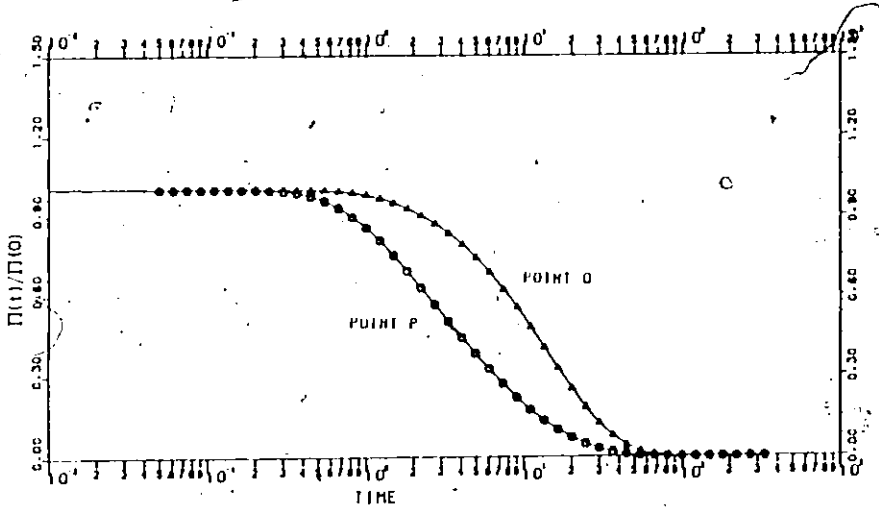
Fig. 4.5 Uncoupled consolidation in a sand drain (K_x=K_y)

(a) fluid velocity field at t=5.0





(b) excess pore pressure isochrones



(c) excess pore pressure vs. time

Fig. 4.6 Uncoupled consolidation in sand drain (K_x=10K_y)

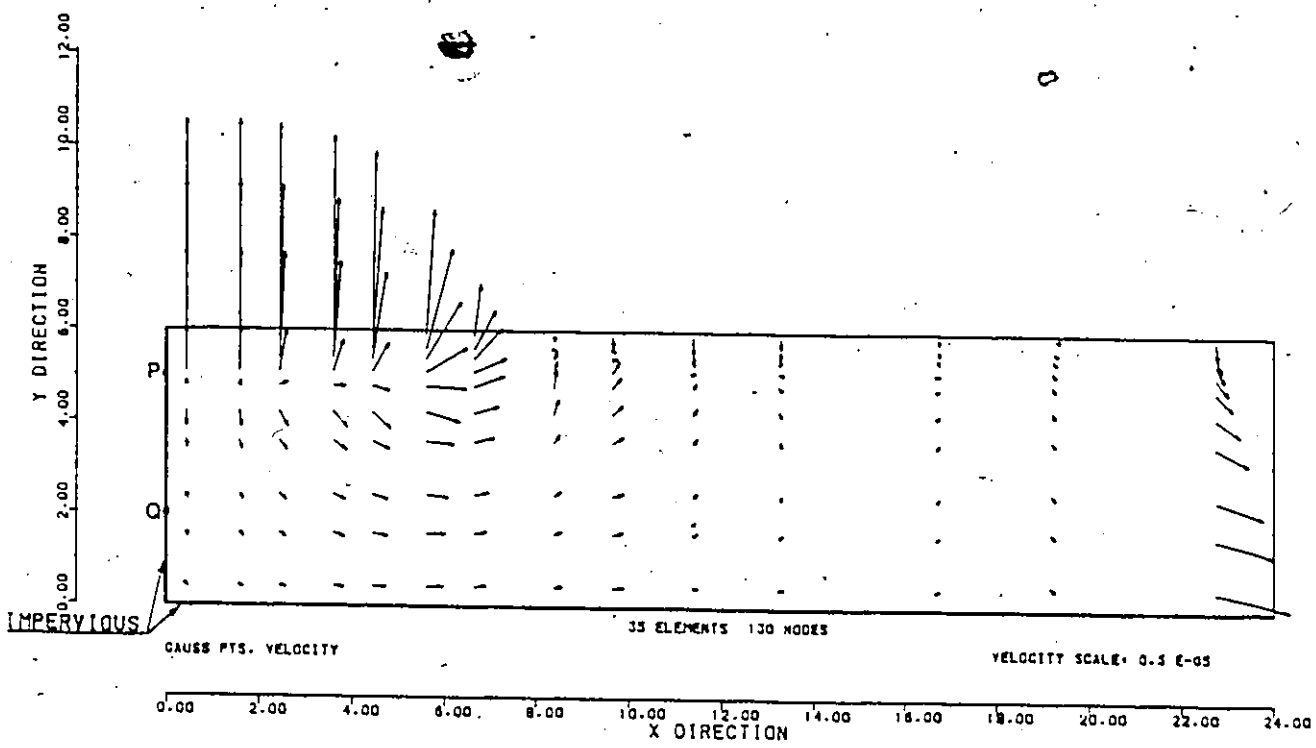
4.4.4 Uncoupled Consolidation in a semi-infinite layer

The practical case of consolidation in a semi-infinite layer due to a uniform strip load was analysed. The mesh layout with the relevant boundary conditions, and profiles are shown in Fig. 4.2c. Sandhu's time space decomposition was used with variable θ. The global "stiffness" matrix had to be updated only at the beginning of each big time interval.

Fig 4.7a illustrates the directions of flow at a stage of consolidation, while the excess pore pressure isochrones drawn at chosen sections are given in Figs. 4.7b to 4.7e.

The initial excess pore pressure distribution, a prerequisite for the analysis of the uncoupled consolidation, was calculated by Henkel's formula. In this case of small deformation and perfectly isotropic condition, the pore pressure parameters "a" and "b" are respectively 1 and 0. The stress invariants involved in the calculation of excess pore pressures were obtained by an elastic stress analysis under the loading conditions illustrated in Fig. 4.2c.

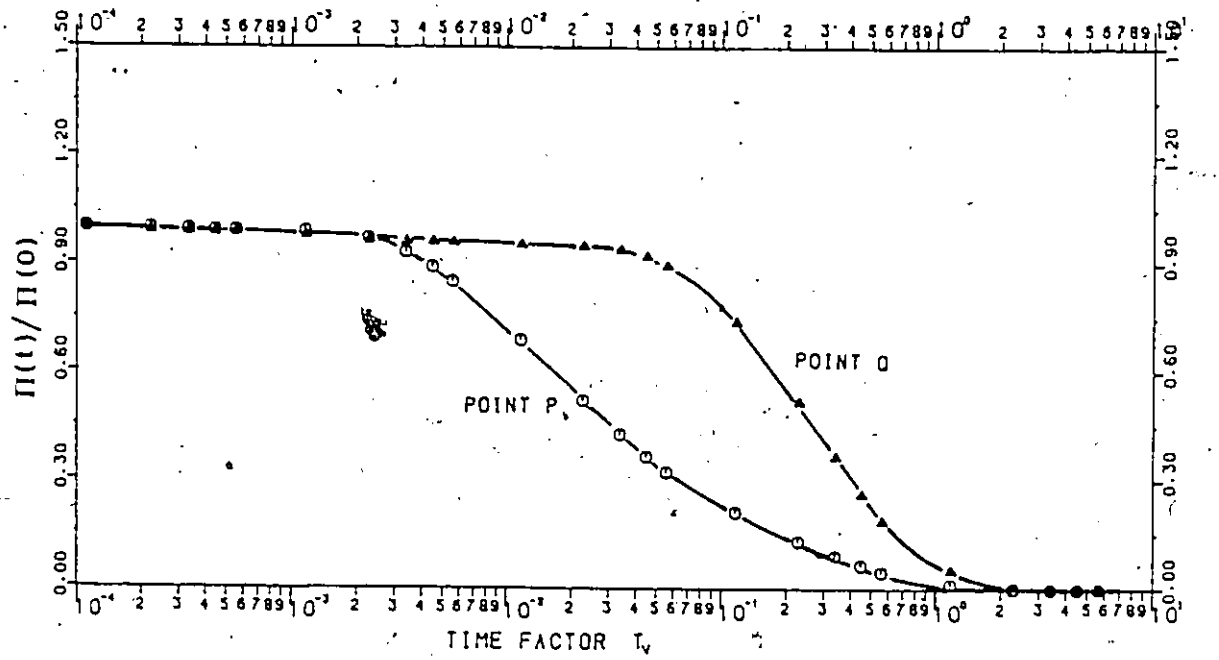
In contrast to the 1-D consolidation approach which crudely assumes a constant excess pore pressure equal in magnitude to the applied load, the present 2-D analysis predicts a slower dissipation process. In fact, 90% degree of consolidation were reached at t=763 in 1-D conditions or at a time factor of 0.42 in a 2-D situation. With reference to Fig. 4.7b, it is found that the dissipation is much slower in 2-D conditions.



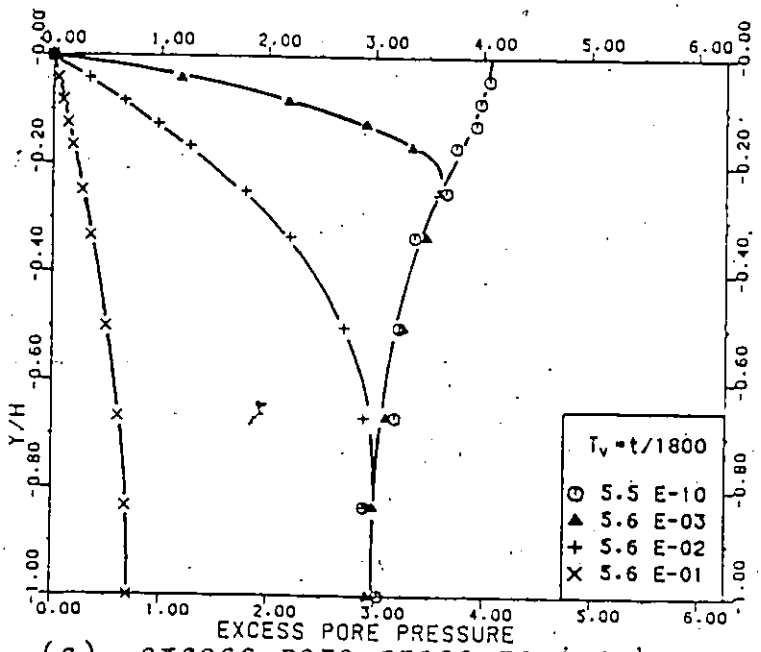
(a) fluid velocity field at T_v = 5.6 E-03

Fig. 4.7 Uncoupled consolidation in a semi-infinite layer

N.B The far right end fluid velocities are unexpectedly high due to boundary conditions.

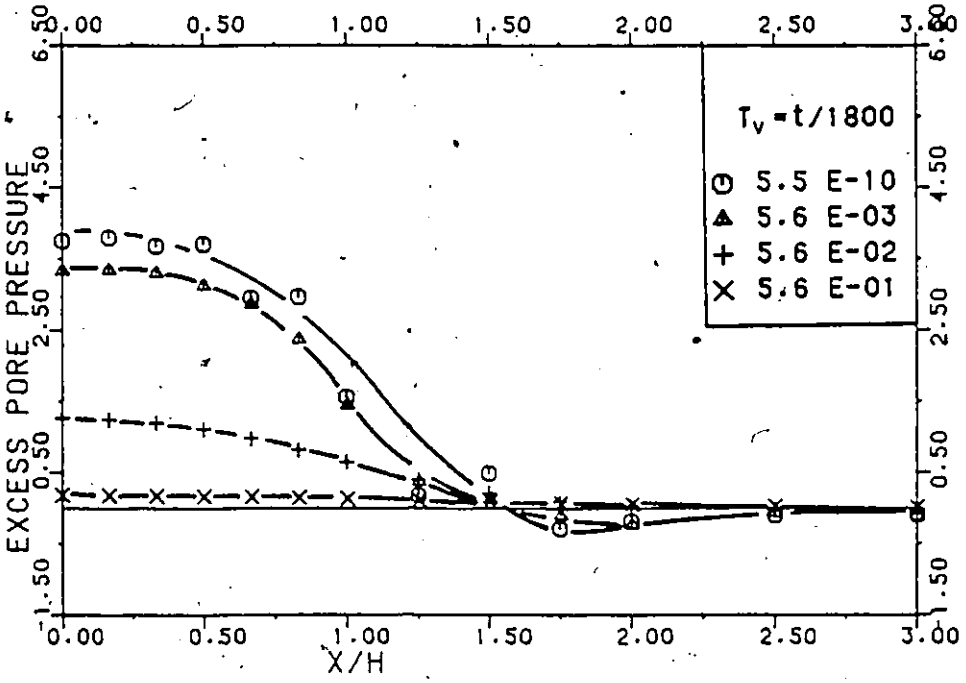


(b) excess pore pressure vs. time

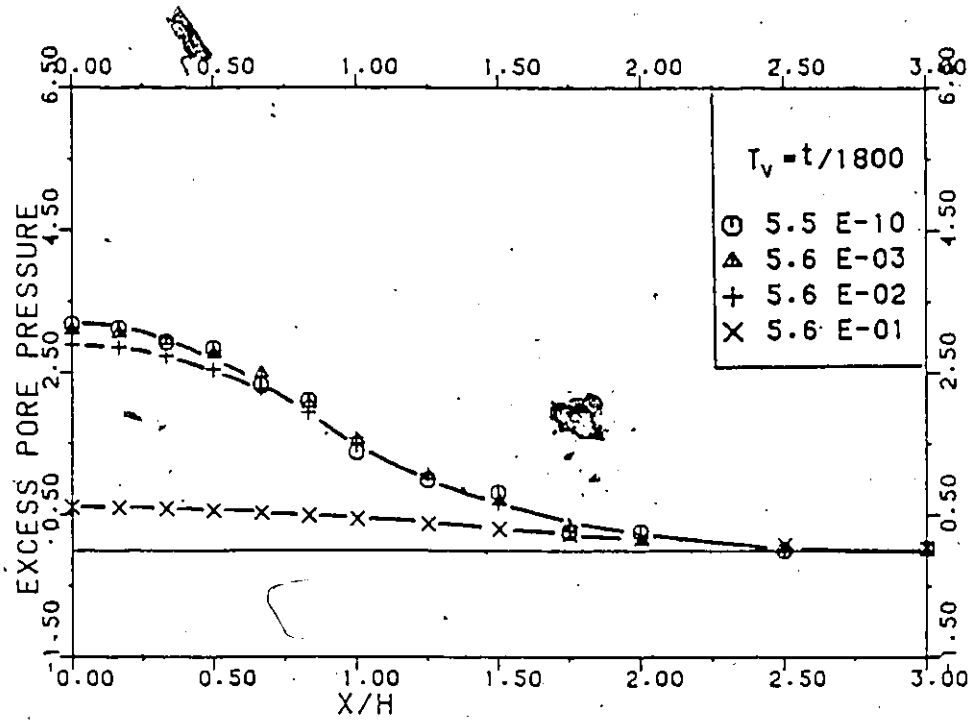


(c) excess pore pressure isochrones at section A-A

Fig. 4.7 Uncoupled consolidation in a semi-infinite layer



(d) excess pore pressure isochrones at section B-B



(e) excess pore pressure isochrones at section C-C

Fig. 4.7 Uncoupled consolidation in a semi-infinite layer

4.5 CONCLUSION

The results within the framework of one-dimensional and axisymmetric consolidation when compared with analytical solutions were in excellent agreement. However, it has to be mentioned that the time stepping scheme employed was rather costly since the "stiffness" matrix had to be updated at each time step due to a variable Δt and θ. On the other hand equally good results were obtained when using option 3 but in this case the computational time was drastically cut down. It was found that the proper choice of the time increment and the θ-value are both determinant factors for achieving non-oscillatory and accurate results. As a rule of thumb at least 60 time steps over the entire period of consolidation were needed to achieve good results. Furthermore, making θ variable with time, as Sandhu (1977) proposed, improved the stability of the solution.

The treatment of 2-D consolidation showed that the simplifications made in 1-D might bring a certain degree of error. The two cases of one- and two-dimensional uncoupled consolidation were compared and observations revealed a slower rate of dissipation in the former case. In conclusion it may be said that as far as the numerical implementation is concerned, TERCON is able to correctly solve the governing equations of the uncoupled consolidation theory. However, the extent to which it can be applied in practice is questionable since it does not take into account solid-fluid interaction.

Chapter V

ELASTO-PLASTICITY

5.1 INTRODUCTION

This chapter is concerned with the stress analysis of a fully saturated, highly permeable soil. As a result, the calculations are carried out in terms of effective stresses.

During the past fifteen years, in an attempt to describe the complicated yield and flow properties of geotechnical materials, several models have been published but their validity and application are still to be questioned.

Though theoretically attractive, the models lead to non-linear relationships which require special F.E. solution algorithms as for instance reported by Zienkiewicz and Pande 1971, or Argyris et al. 1972.

The theoretical developments will not be attempted, nor will their validity be discussed, this not being the main objective. Instead, the actual numerical response to the general boundary value problem will be examined.

Implementation of these models in a F.E. program is by itself a challenging task since a number of details have yet to be solved for a correct numerical analysis.

Two main classes of elasto-plastic models have been considered, namely:

1. Elastic perfectly plastic model in which the limit state can be described by either von Mises, Tresca, Mohr-Coulomb or Drucker-Prager failure criterion.
2. Generalized Burland model (FUSCO, 1985) which is an extension of the modified Cam-clay model.

Although the first class of elasto-plastic models may not adequately simulate a number of the soil skeleton features, on the other hand numerical solutions may eventually be checked against the theoretical expectations of the Limit Analysis theory.

These models, therefore, once implemented correctly, ensure a good foundation for later extension to the more elaborate generalized Burland model.

In a first stage an account of the different available failure criteria, along with their incorporation in both linear elastic perfectly plastic and cap models is presented.

In order to implement these models, a brief review of previous works in numerical algorithms pointing out the pertinent details is done.

Finally the last part of this chapter highlights the refinements brought in the solution procedures for a better accuracy and efficiency. The numerical results are then herein presented and discussed.

5.2 BASIC EQUATIONS IN ELASTO-PLASTICITY

Because strains depend upon the entire loading history of the material, plastic stress-strain relationships are commonly given in incremental form.

It is common to assume strain increments to be divisible into elastic and plastic components, so that:

$$d\underset{\sim}{\varepsilon} = d\underset{\sim}{\varepsilon}^{e} + d\underset{\sim}{\varepsilon}^{p} \qquad (5.1)$$

In the hypothesis of linear elasticity, the elastic strain is related to the stress by the well-known generalized Hooke's law (after Hooke 1678).

$$d\underset{\sim}{\varepsilon}^{e} = [C^{e}]^{-1}\, d\underset{\sim}{\sigma} \qquad (5.2)$$

where C^{e} is the elastic constitutive matrix, which for an isotropic material contains only two independent parameters, the Young's modulus E and the Poisson ratio υ.

Instead, the relationship between plastic strain and stress, in the case of small deformation, is based on the early mathematical formulations of Melan (1938).

$$d\underset{\sim}{\varepsilon}^{p} = \overline{d\ell}\, \frac{\partial F}{\partial \underset{\sim}{\sigma}} \qquad (5.3a)$$

where

$$\overline{dl} = \ell \, dG_\sigma = \frac{dG_\sigma}{A} \qquad \text{if } dG_\sigma > 0 \qquad (5.3b)$$

$$\overline{dl} = 0 \qquad \text{if } dG_\sigma \leq 0 \qquad (5.3c)$$

$$dG_\sigma = \overline{\nabla}_{\underset{\sim}{\sigma}} G \cdot d\underset{\sim}{\sigma} \qquad (5.3d)$$

$$A = \frac{1}{\ell} \qquad (5.3e)$$

where F and G are respectively the plastic potential and yield functions.

By considering all the above expressions, it is possible to prove that the elasto-plastic constitutive equation takes up the following form (see Yamada et al. 1968, or Zienkiewicz et al. 1969).

$$d\underset{\sim}{\sigma} = C^{ep} \, d\underset{\sim}{\varepsilon} = [\underset{\sim}{C}^e - C^p] \, d\underset{\sim}{\varepsilon} \qquad (5.4a)$$

where

$$C^P = \frac{\underset{\sim}{c}_F * \underset{\sim}{c}_G}{A + \underset{\sim}{c}_G^T \cdot \overline{\nabla}_\sigma F} \qquad (5.4b)$$

and

$$\underset{\sim}{c}_F = c^e \cdot \overline{\nabla}_\sigma F \; ; \qquad \underset{\sim}{c}_G = C^e \cdot \overline{\nabla}_\sigma G \qquad (5.4c)$$

* Diac product

When the plastic potential function F coincides with the yield function G, the material is said to follow an associated flow rule.

According to Drucker (1951), a stable work hardening material follows an associated flow rule and requires the potential function F to be convex.

In the particular case when the potential function is also quasi-isotropic, i.e. the function F may be equivalently described by the stress invariants I_1, J_2 and J_3, the gradient $\overline{\nabla}_\sigma F$ may be expressed as the sum of three individual components as follow:

$$\overline{\nabla}_\sigma F = \overline{\nabla}_p F + \overline{\nabla}_s F = \underset{\sim}{a}_1 c_1 + \underset{\sim}{a}_2 c_2 + \underset{\sim}{a}_3 c_3 \qquad (5.5)$$

where

$$\underset{\sim}{a}_1 = \frac{\partial I_1}{\partial \sigma_{ij}} = \delta_{ij} = \{1,1,1,0,0,0\}^T = \underset{\sim}{\delta} \qquad (5.6)$$

$$c_1 = \frac{1}{3}\frac{\partial F}{\partial p} = \frac{\partial F}{\partial I_1} \tag{5.7}$$

$$\underset{\sim}{a}_2 = \frac{\partial J_2^{1/2}}{\partial s_{ij}} - \frac{\delta_{ij}}{3}\frac{\partial J_2^{1/2}}{\partial s_{kk}} = \frac{\underset{\sim}{s}}{2J_2^{1/2}} \tag{5.8}$$

$$c_2 = \frac{\partial F}{\partial J_2^{1/2}} - \frac{tg3\theta}{J_2^{1/2}}\frac{\partial F}{\partial \theta} \tag{5.9}$$

$$\underset{\sim}{a}_3 = \frac{\partial J_3}{\partial s_{ij}} - \frac{\delta_{ij}}{3}\frac{\partial J_3}{\partial s_{kk}} = \frac{\partial J_3}{\partial s_{ij}} - \frac{\delta_{ij}}{3} J_2 \tag{5.10}$$

$$\frac{\partial J_3}{\partial \tilde{s}_{ij}} = \begin{bmatrix} (s_{22}s_{33}-s_{23}) & (s_{13}s_{23}-s_{33}s_{12}) & (s_{12}s_{23}-s_{22}s_{13}) \\ & (s_{11}s_{33}-s_{13}) & (s_{12}s_{13}-s_{11}s_{23}) \\ \text{symmetric} & & (s_{11}s_{22}-s_{12}) \end{bmatrix} \tag{5.11}$$

$$c_3 = -\frac{\sqrt{3}}{2J_2^{3/2}\cos 3\theta}\frac{\partial F}{\partial \theta} \tag{5.12}$$

coefficients c_1, c_2 and c_3 depend on the particular choice of the potential function and will be explicited in the following sections.

Furthermore, it is interesting to underline that the last three components of the vector $\underset{\sim}{a}_2$ in Eq. 5.8 have been changed and are half those reported by several other authors.

5.3 FINITE ELEMENT FORMULATION OF ELASTO-PLASTICITY

There are many approaches for reducing the differential equations of equilibrium into a suitable form for a finite element analysis. For instance, the variational formulation (Gurtin, 1964), the Galerkin's method of weighted residuals (Finlayson, 1972), or the principle of virtual work (Oden, 1982) are equivalently employed.

Apparently the most immediate derivation is achieved through the principle of virtual work whose basic equation is reported in Eq. 5.13. (Castigliano 1875).

According to the principle of effective stress (Terzaghi, 1943), the total stress may be decomposed the way shown in Eq. 5.14. In the assumption that displacement, and consequently strain, can be approximated according to Eqs. 5.15 and 5.16, the starting equation 5.13 reduces its discretized form, Eq. 5.17a.

Principle of virtual work

$$\int_\Omega \delta\underset{\sim}{\varepsilon}^* \cdot \underset{\sim}{\sigma}(t)\, d\Omega - \int_\Omega \delta\underset{\sim}{u}^* \cdot \underset{\sim}{b}\, d\Omega - \int_{\Gamma_t} \delta\underset{\sim}{d}^* \cdot \underset{\sim}{t}(t)\, d\Gamma = 0 \qquad (5.13)$$

Principle of effective stress

$$\underset{\sim}{\sigma}(t) = \underset{\sim}{\sigma}'(t) + \underset{\sim}{\sigma}'_o - \underset{\sim}{\delta}[\pi^e(t) + \pi_o] \qquad (5.14)$$

Finite element discretization

$$\underset{\sim}{u}^* = S \cdot \underset{\sim}{d} \qquad (5.15a)$$

$$\delta\underset{\sim}{u}^* = S \cdot \delta\underset{\sim}{d} \qquad (5.15b)$$

$$\underset{\sim}{\varepsilon}^* = B \cdot \underset{\sim}{d}^* \qquad (5.16a)$$

$$\delta\underset{\sim}{\varepsilon}^* = B \cdot \delta\underset{\sim}{d}^* \qquad (5.16b)$$

$$\underset{\sim}{\pi} = \bar{s} \cdot \underset{\sim}{\pi}_o \qquad (5.16c)$$

$$\int_\Omega B^T \underset{\sim}{\sigma}'(t) d\Omega - C \underset{\sim}{\pi}^e(t) = \underset{\sim}{f}'(t) \qquad (5.17a)$$

$$\underset{\sim}{f}'(t) = \underset{\sim}{f}(t) - \int B^T \underset{\sim}{\sigma}'_o \, d\Omega + C \underset{\sim}{\pi}^o \qquad (5.17b)$$

$$C = \int_\Omega (B^T \hat{\underset{\sim}{m}}) * \bar{s}^T d\Omega \qquad (5.17c)$$

$$\underset{\sim}{f}(t) = \int_{\Gamma_t} S^T \underset{\sim}{t} \, d\Gamma + \int_\Omega \rho \, S^T \underset{\sim}{b} \, d\Omega \qquad (5.17d)$$

σ tensor of the total stress

$\sigma'(t)$ tensor of the effective stress at instant t.

σ'_o tensor of the pre-existing effective stress

$\underset{\sim}{t}(t)$ vector of boundary traction forces changing with time

$\delta\underset{\sim}{u}^*$ vector of virtual internal displacements

$\delta\underset{\sim}{d}^*$ vector of virtual nodal displacements

$\delta\underset{\sim}{\varepsilon}^*$ vector of associated virtual strains

Ω	domain of interest
Γ_t	boundary along which traction forces act
S	shape function matrix for displacement discretization
$\bar{S}$	shape function matrix for pore pressure discretization
B	strain-displacement matrix
$\hat{\underset{\sim}{m}}$	$(1,1,1,0,0,0)^T$
$\underset{\sim}{\pi}_o(t)$	vector of initial pore pressures at time t
$\underset{\sim}{\pi}^e(t)$	vector of excess pore pressures at time t
$\underset{\sim}{f}(t)$	loading vector

Since equilibrium has to be respected at any stage, Eq. 5.17a may be alternatively rewritten in incremental form.

$$\int_\Omega B^T \, \Delta\underset{\sim}{\sigma}'(t) \, d\Omega - C \, \Delta\pi^e(t) = \Delta f'(t) \qquad (5.18)$$

For highly permeable saturated media the excess pore pressure $\underset{\sim}{\pi}^e$ and $\Delta\underset{\sim}{\pi}^e$ are equal to zero and the process is time independent.

Finally, by means of Eqs. 5.4a and 5.16 the equilibrium may be represented by the following system of linear equations:

$$K^{ep} \, \Delta\underset{\sim}{d} = \Delta\underset{\sim}{f} \qquad (5.19a)$$

where K^{ep}, the elasto-plastic stiffness matrix, is defined as

$$K^{ep} = \int B^T C^{ep} B \, d\Omega \qquad (5.19b)$$

and

$$C^{ep} = C^e - C^p \qquad (5.19c)$$

C^e and C^p have been previously defined in Sect. 5.2.

5.4 FAILURE CRITERIA

It is known that materials reach failure at a certain threshold value after which no further load can be sustained.

The envelope of all possible states of stress producing failure describes a surface in the stress space.

Coulomb (1776) was apparently the first to suggest a criterion for the definition of failure. He proposed that failure takes place when the shear stress τ reaches a critical value that depends on the internal friction angle ϕ and the normal stress σ_n, provided the latter is a compressive stress.

$$\tau = c + \sigma_n \tan\phi \qquad (5.20)$$

where the coefficient c is known as the cohesion and is a material constant.

For pure cohesive materials $\phi = 0$, Eq. 5.20 thus simplifies into

$$\tau = c \tag{5.21}$$

which is known as the Tresca failure criterion (Tresca, 1868).

Mohr (1882) interpreted the Coulomb criterion in terms of principal stresses and the geometrical representation of the complete Mohr-Coulomb failure surface in the principal stress space, is shown in Fig. 5.1a. This pyramidal surface is symmetric about the space diagonal and the normal section at any point is an irregular hexagon, Fig. 5.1b.

For pure cohesive material, the failure surface (Tresca) changes into a prism whose normal section at any point is a regular hexagon, Fig. 5.1b.

A convenient numerical representation of the Mohr-Coulomb failure surface was given by Nayak and Zienkiewicz (1972) in terms of stress invariants.

$$I_1 \frac{\sin\phi}{3} + J_2^{1/2}\left(\cos\theta - \frac{\sin\theta \sin\phi}{\sqrt{3}}\right) = c \cos\phi \tag{5.22}$$

where I_1 and J_2 are respectively the first stress invariant and the second deviatoric stress invariant. The parameter θ is defined as

$$\theta = \frac{1}{3} \arcsin \left(- \frac{3\sqrt{3}}{2} \frac{J_3}{J_2^{3/2}}\right) \qquad (5.23)$$

$$-\pi/6 \leqslant \theta \leqslant \pi/6 \quad \text{(refer to Fig. 5.1b)}$$

J_3 is the third stress invariant.

The 3-D stress representation of the Tresca yield criterion can be easily obtained by setting $\phi = 0$.

$$J_2^{1/2} \cos\theta = c \qquad (5.24)$$

Von Mises (1928) tried to alleviate the mathematical difficulties associated with the determination of the angle θ in Eq. 5.24, replacing the hexagonal prism by an inscribed circular cylinder, Fig. 5.1b. Thus, the von Mises failure criterion can be obtained by setting $\theta = 30°$ in Eq. 5.24.

$$J_2^{1/2} \frac{\sqrt{3}}{2} = c \qquad (5.25)$$

A generalized form of Eq. 5.25 was further proposed by Drucker and Prager (1952), Eq. 5.26. In this case the yield surface in the principal stress space is a right circular cone whose axis of revolution is the diagonal space (Fig. 5.1a).

$$\alpha I_1 + \beta J_2^{1/2} = k \tag{5.26}$$

In order to make the cone coincide with the outer or inner apexes of the Mohr-Coulomb hexagon at any section, the above parameters α, β and k should assume the following values:

$$\alpha = \frac{\sin\phi}{3} \tag{5.27}$$

depending on θ,

$$\alpha = \frac{\sqrt{3}}{2}(1 \mp \sin\phi) \tag{5.28}$$

$$k = c\cos\phi \tag{5.29}$$

It then follows that any linear failure criteria may be generally expressed by:

$$q + Mp = N \tag{5.30}$$

where p is the mean pressure (negative for compression), q the devaitoric stress, and M and N strength parameters defined in Table 5.1.

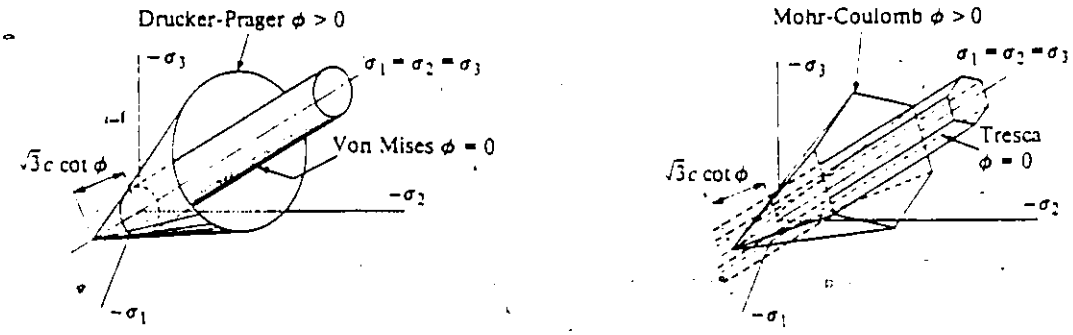


Fig. 5.1a Linear yield surfaces in principal stress space

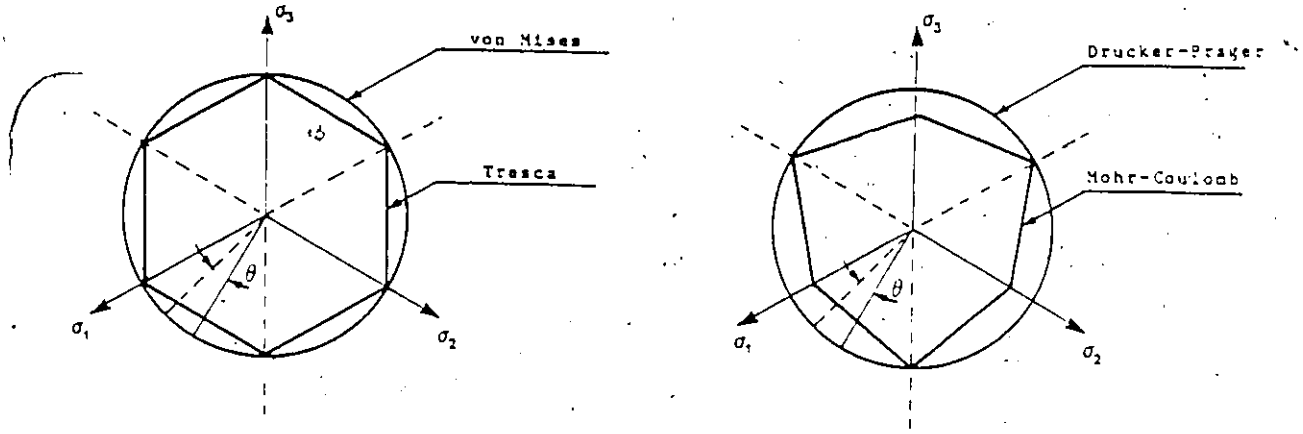


Fig. 5.1b Cross section of a linear yield function onto the octahedral plane

TABLE 5.1

Failure criteria coefficients

Type	N	M
Tresca	$\frac{3c}{\sqrt{3}\cos\theta}$	0
von Mises	$2c$	0
Mohr-Coulomb	$\frac{3c\cos\phi}{\sqrt{3}\cos\theta - \sin\theta\sin\phi}$	$\frac{3\sin\phi}{\sqrt{3}\cos\theta - \sin\theta\sin\phi}$
Drucker-Prager	$\frac{6c\cos\phi}{3 - \sin\phi}$	$\frac{6\sin\phi}{3 - \sin\phi}$

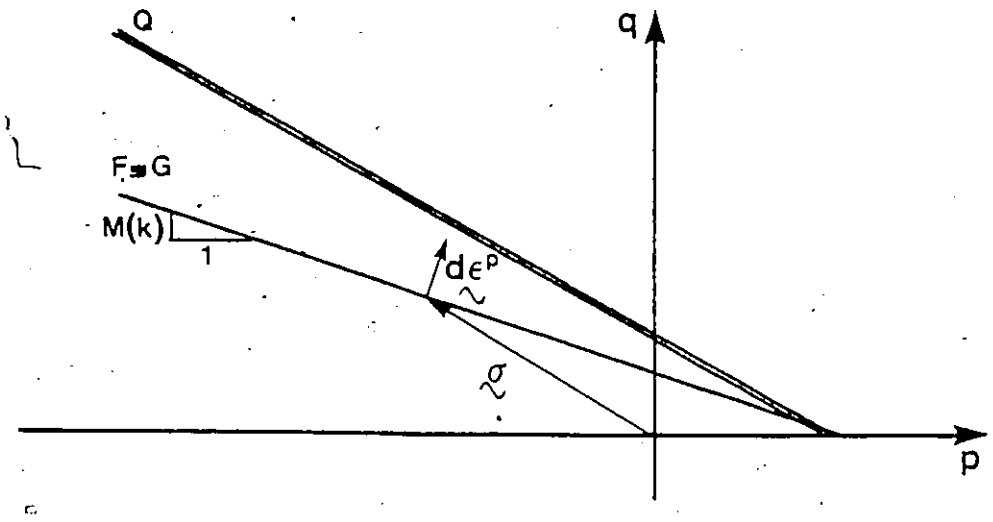


Fig. 5.2 Hardening of a plastic isotropic linear function

5.5 ELASTO-PLASTIC MODELS

It is often assumed that, for some types of materials, yield can be approximated by linear functions like those describing the failure criteria reported in Sect. 5.4.

As plastic deformation develops, the yield function G expands and eventually coincides with Q at failure, Fig. 5.2. The law of variation of G is usually assumed to be controlled by a single parameter k, and is represented by Eqs. 5.31a and 5.31b for cohesive and frictionless materials respectively.

$$G = q - M\,(d - p) = 0\,, \qquad M = M(k) \tag{5.31a}$$

$$G = q - N = 0\,, \qquad N = N(k). \tag{5.31b}$$

where d is the intercept with the p-axis.

There are two alternative ways to determine the plastic modulus A. Hill (1956) postulated the hardening parameter k to be equal to the total plastic work. Some other authors related k to the plastic deviatoric strain.

Apparently, these means of determining A is only restricted to isotropic functions. However, for perfectly plastic materials with associated flow rule; where F always

coincides with G, the plastic modulus A reduces to zero regardless the assumption on the hardening parameter k (Fusco, 1985). In fact, since $d\sigma$ in this case cannot leave the failure surface Q, it can be at most tangent to the surface G=Q and hence orthogonal to the gradient $\nabla_{\sigma} G$. Consequently as $dG_s = \nabla_{\sigma} G \cdot d\sigma$, A is also equal to zero from Eq. 5.3b.

For elastic perfectly plastic materials the values of M and N coincide with those reported in Table 5.1. Under this condition, it is possible to prove that the coefficients c_1, c_2, c_3, as defined in Eqs. 5.7, 5.9 and 5.12, are equal to those reported in Table 5.2.

At the corners of Tresca and Mohr-Coulomb yield surfaces, i.e. $\theta = +30°$, the flow vector a cannot be uniquely defined (refer to expressions of c_1 and c_2 in Table 5.2). Therefore a smooth surface is substituted for all values of $|\theta| \leq 29°$. In this region, the yield functions are evaluated explicitly with $\theta = 30°$, giving a circular surface over a small arc. This procedure results in the alternative expressions of c_1 and c_2 reported in Table 5.3.

TABLE 5.2

Constants defining flow vector in a form suitable for numerical analysis (linear elasto-plastic models)

	c_1	c_2	c_3
Tresca	0	$2\cos\theta[1+\operatorname{tg}\theta\operatorname{tg}3\theta]$	$\frac{\sqrt{3}}{J_2}\frac{\sin\theta}{\cos 3\theta}$
Von Mises	0	$\sqrt{3}$	0
Mohr-Coulomb	$\frac{\sin\phi}{3}$	$\cos\theta[(1+\operatorname{tg}\theta\operatorname{tg}3\theta)+\frac{\sin\phi}{\sqrt{3}}(\operatorname{tg}3\theta-\operatorname{tg}\theta)]$	$\frac{\sqrt{3}\sin\theta+\cos\theta\sin\phi}{2J_2\cos 3\theta}$
Drucker-Prager	$\frac{\sin\phi}{3}$	$\frac{\sqrt{3}}{2}(1-\sin\phi)$	0

TABLE 5.3

Rounding off the yield surface corners

for $|\theta| > 29^{\circ}$

yield criterion	c_1	c_2	c_3
Tresca	0	$\sqrt{3}$	0
Mohr-Coulomb	$\frac{\sin\phi}{3}$	$\frac{\sqrt{3}}{2}(1 \mp \sin\phi)$	0

5.6 CAP MODELS

Most soils undergo plastic deformation even under hydrostatic loading conditions. According to Roscoe et al. (1968), the soil at failure is not subjected to any further change in volume. Parry (1956 and 1960), and Bishop (1966) verified experimentally that the yield surface is of the Mohr-Coulomb type.

These observations lead to the following deductions:

1. The yield surface is of convex shape and intersects the hydrostatic axis.
2. The failure, or Critical State, is represented by the Mohr-Coulomb surface.
3. The gradient of the potential function lies on the deviatoric plane, i.e the plastic volume change is equal to zero ($d\varepsilon_v^p = 0$, in the hypothesis of small deformation.

It follows that the previous linear yield functions cannot adequately represent the soil plastic deformation.

5.6.1 Cam-clay models

Drucker et al. (1957, 1961, 1966) suggested that the soil might be modelled as an elasto-plastic hardening material whose yield surface has convex end caps which upon successive loading, does not reach the failure surface.

Roscoe el al. 1958, pursuing the early findings of Rendulic (1936), Hvorslev (1937) and Terzaghi (1943),

postulated the existence of a critical state and a "State Boundary Surface". Subsequent works of Roscoe and Poorooshab, Calladine and Roscoe, Schofield and Thurairajah, all in 1963 led to the establishment of a stress-strain theory of clay known nowadays as the Cam-clay model.

Finally Roscoe and Burland (1968) proposed a new form of the yield function representing an elliptical shape when mapped in the p-q diagram (see Fig. 5.3).

$$F = p^2 + \frac{q^2}{M^2} - pp_e = 0 \qquad (5.32)$$

where p, the mean pressure is taken as positive for compression and p_e is the value of p at the intersection of the current yield surface with the p-axis.

The above equation was derived from an hypothesis on plastic work dissipation (Burland 1965).

Since then, several alternative interpretations of the model have been proposed (Zienkiewicz and Naylor, 1971; Mroz et. al. 1978; Naylor and Pande, 1981), but the basic assumptions on the plastic deformation have always been kept. These are:

1. The plastic deformation occurs according to an associated flow rule in the normally consolidated state.

2. The variation of plastic specific volume between the two next surfaces depends only on the variation of the effective pressure p_e and specific volume $\bar{v}$.

$$d\varepsilon_v^p \simeq \frac{d\bar{v}^p}{\bar{v}} = \frac{\chi-\lambda}{\bar{v}} \frac{dp_e}{p_e} \qquad (5.33)$$

λ and χ are respectively the slope of the normal consolidation and overconsolidation lines.

5.6.2 Determination of the plastic modulus A

It can be proved the validity of the following equation relating the plastic volumetric change to the mean pressure p and the plastic modulus A.

$$d\varepsilon_v^p = \frac{dG_\sigma}{A} \frac{\partial F}{\partial p} \qquad (5.34)$$

Since the yield function G depends on the hardening parameter k, the plastic modulus A can be alternatively expressed as:

$$A = \frac{dG_\sigma}{d\varepsilon_v^p} \frac{\partial F}{\partial p} = - \frac{\partial G}{\partial k} \frac{dk}{d\varepsilon_v^p} \frac{\partial F}{\partial p} \qquad (5.35)$$

The fact that the hardening parameter k can be assimilated to p_e, a more general form to express A using Eqs. 5.33 and 5.35 is given by:

$$A = - \frac{\partial G}{\partial p_e} \frac{dp_e}{d\varepsilon^p} \frac{\partial F}{\partial p} = \frac{\bar{v}\, p_e}{\lambda - \chi} \frac{\partial G}{\partial p_e} \frac{\partial F}{\partial p} \qquad (5.36)$$

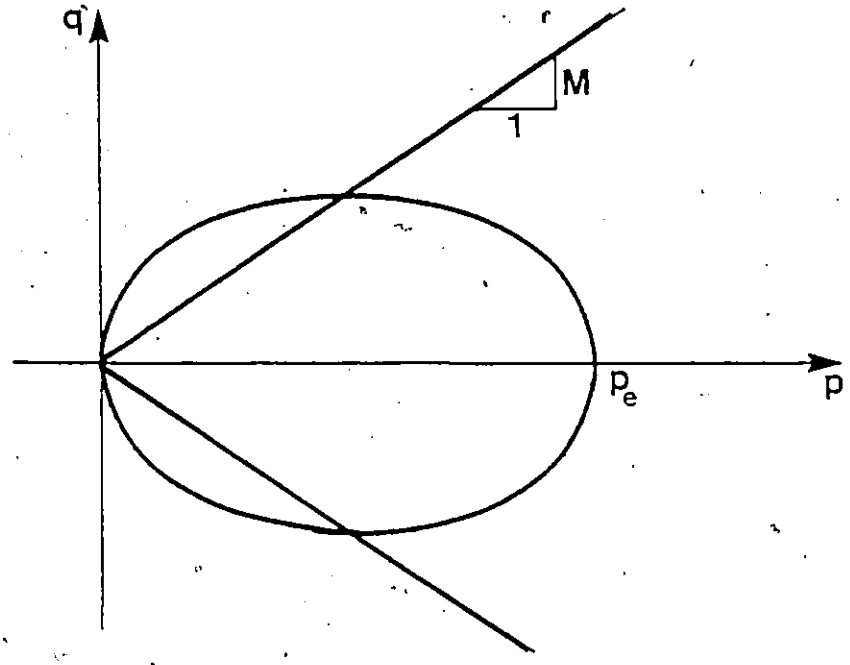


Fig. 5.3 Mapping of Burland's yield surface in p vs. q diagram

5.6.3 Generalized Burland model

The Burland model, or modified Cam-clay model, in its original form assumed a Drucker-Prager failure surface with vertex at the origin of the stress space.

The yield function represented by the ellipse given by Eq. 5.32 has been generalized and expanded according to many aspects (Fusco 1985). Firstly, the yield function has been alternatively expressed in terms of new geometrical parameters like α, β defining the centre of the ellipse, h the major semi-axis and n the ratio of the semi-axes.

$$F = (p + \beta)^2 + \frac{q^2}{n^2} - h^2 = 0 \qquad (5.37)$$

$$P_e = 2\beta = 2h \qquad (5.38)$$

$$n \equiv M \qquad (5.39)$$

Then, the Mohr-Coulomb failure criterion, together with non-zero cohesion have been combined to give more generality to the original model.

The inclusion of the Mohr-Coulomb failure surface in this model can be justified by the experimental findings of Parry (1956, 1960) and Bishop (1966).

The corresponding equations are summarized in Table 5.4 showing the introduction of new parameters and their dependence on the angle θ. The graphical representation of the yield function mapping in a p-q diagram is illustrated in Fig. 5.4. Subsequently, the plastic modulus A is formulated by means of the new generalized function.

It is obviously possible to recover the original equations of Burland model by setting $c = 0$ and $\theta = 30°$ in the equations found in Table 5.4.

However, it is noticed that the case $\alpha \neq 0$, when the ellipse is no more symmetric about the p-axis, has not yet a physical meaning, and has been introduced just for the sake of mathematical completeness.

The coefficients c_1, c_2 and c_3 needed to calculate the flow vector a are accordingly derived and reported in Table 5.5.

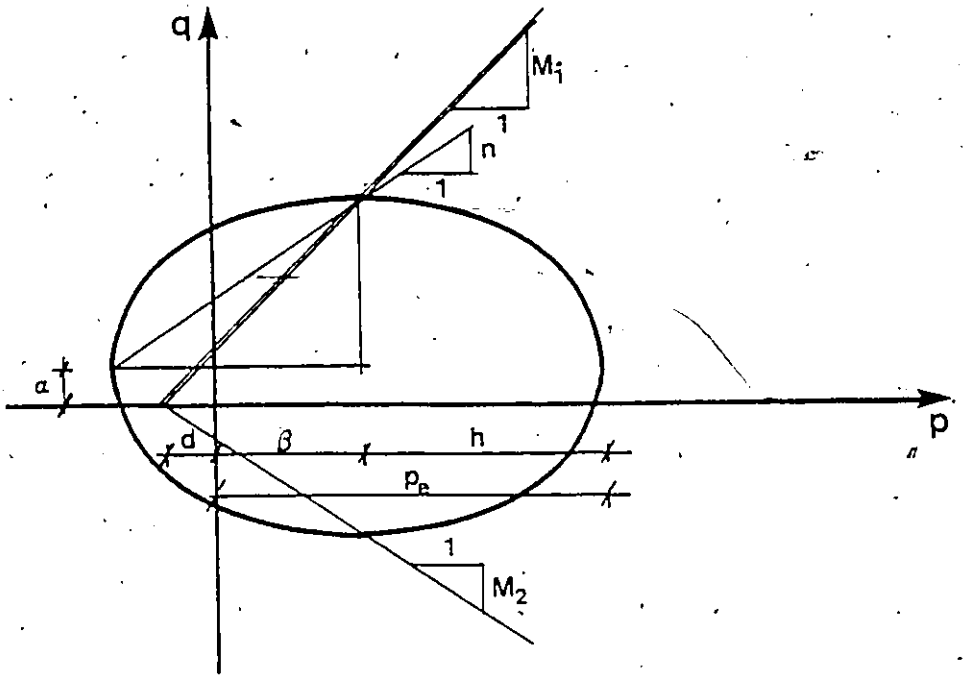


Fig. 5.4 Mapping of the Generalized Burland yield surface in p vs. q diagram

TABLE 5.4

Generalized Burland model

yield function

$$F = (p + \beta)^2 + \frac{(q - \alpha)^2}{n^2} - h^2 = 0 \qquad (5.40a)$$

where

$$\alpha = a\ h; \quad \beta = b\ h - d \qquad (5.40b)$$

$$n = r[2\ M_1 - (M_1 - M_2)_{\theta=30°}]\ ; \quad r = \left(\frac{n}{M_1 + M_2}\right)_{\theta=30°} \qquad (5.40c)$$

$$a = n\left(\frac{M_1 - M_2}{M_1 + M_2}\right)\ ; \quad b = \frac{2n}{M_1 + M_2} \qquad (5.40d)$$

$$d = \frac{N_i}{M_i} = c\ \mathrm{ctg}\phi \qquad i = 1, 2 \qquad (5.40e)$$

$$M_i = \frac{3\ \sin\phi_i}{\sqrt{3}\ \cos\theta - \sin\theta\ \sin\phi_i} \qquad i = 1, 2 \qquad (5.40f)$$

$$h = \frac{\bar{b} + (\bar{b}^2 - \bar{a}\bar{c})^{1/2}}{\bar{a}} \qquad \forall\ \bar{a} \neq 0 \qquad (5.40g)$$

$$h = \frac{\bar{c}}{2\bar{b}} \qquad \bar{a} = 0 \qquad (5.40h)$$

$$\bar{a} = b^2 + \left(\frac{a}{n}\right)^2 - 1 \qquad (5.40i)$$

$$\bar{b} = \frac{\bar{q}\, a}{n^2} - (\bar{p} - d)b \qquad (5.40j)$$

$$\bar{c} = \frac{\bar{q}^2}{n^2} + (\bar{p} - d) \qquad (5.40k)$$

$\bar{p}$, $\bar{q}$ stress invariant associated with the current stress

$$p_e = \beta + h$$

plastic modulus A

$$A = \frac{\bar{v}(h-g)}{\lambda-\chi} \frac{\partial F}{\partial h} \frac{\partial F}{\partial p} \qquad (5.41a)$$

where

$$g = \frac{d}{1+b} = \frac{c \cot\phi (M_1 + M_2)}{(M_1 + M_2 + 2n)} \qquad (5.41b)$$

$$\frac{\partial F}{\partial p} = 2(p + \beta) \qquad (5.41c)$$

$$\frac{\partial F}{\partial p_e} = 2(\bar{a}\, h - \bar{b}) \qquad (5.41d)$$

TABLE 5.5

Constants defining flow vector in a form suitable for numerical analysis (Generalized Burland model)

coupling yield criterion	c_1	c_2	c_3
Gen.Burland (Mohr-Coulomb)	$\frac{2}{3}(p+\beta)$	$\frac{2\sqrt{3}}{n^2}(q-\alpha)(1+Q)$	$\frac{(q-\alpha)^2}{q^3 n^2}\frac{9}{\cos 3\theta}T$
Gen.Burland (Drucker-Prager) for $\lvert\theta\rvert > 29^\circ$	$\frac{2}{3}(p+\beta)$	$\frac{2\sqrt{3}}{n^2}(q-\alpha)$	0
Gen.Burland	$\frac{2}{3}(p+\beta)$	$\frac{2\sqrt{3}}{\bar{n}^2}(q-\alpha)$	0

where

$$T = \frac{M_1\Theta_1 + M_2\Theta_2}{M_1 + M_2} \quad ; \quad \Theta_i = \frac{\sqrt{3}\sin\theta - \cos\theta\sin\phi_i}{\sqrt{3}\cos\theta - \sin\theta\sin\phi_i} \quad ; \quad Q = \left[\frac{(q-\alpha)}{q}\,\mathrm{tg}\,3\theta\right]^T$$

$$M_i = \frac{6\sin\theta}{3 \mp \sin\theta} \quad ; \quad \bar{n} = \text{in Table 5.4}$$

5.7 NUMERICAL ALGORITHMS IN ELASTO-PLASTICITY

Various algorithms are available for the solution of non-linear problems and detailed treatment of the methods may be found in a wide range of publications (Armen et al. 1972, Oden 1972, Argyris et al. 1972, Desai and Abel 1972, Tillerson et al. 1973, and Zienkiewicz 1977).

In an elasto-plastic analysis, the stiffness matrix is stress dependent and not unique function of displacement. As a result, the choice of the solution procedure is basically restricted to the so-called "Pure Incremental Procedure" and to the "Initial Load Methods".

The "Pure Incremental Procedure" is a piecewise linearization of the non-linear force-deformation curve as shown in Fig. 5.5. This process, besides being computationally costly, does not guarantee accuracy and may not converge for softening materials.

The "Initial Load Method" is comprised of two basic approaches namely the "Initial Strain" and "Initial Stress" methods.

The "Initial Strain" method (see Fig. 5.6), essentially consists in finding the additional plastic stress field which equilibrates the mobilized stress field calculated in linear elasticity (Desai and Abel 1972). It is to be noted that this method is manifestly inadequate for states of stress at failure, when the plastic modulus A becomes zero.

Instead the "Initial Stress" method has proved to be generally applicable and the main points are thereby illustrated in Fig. 5.7

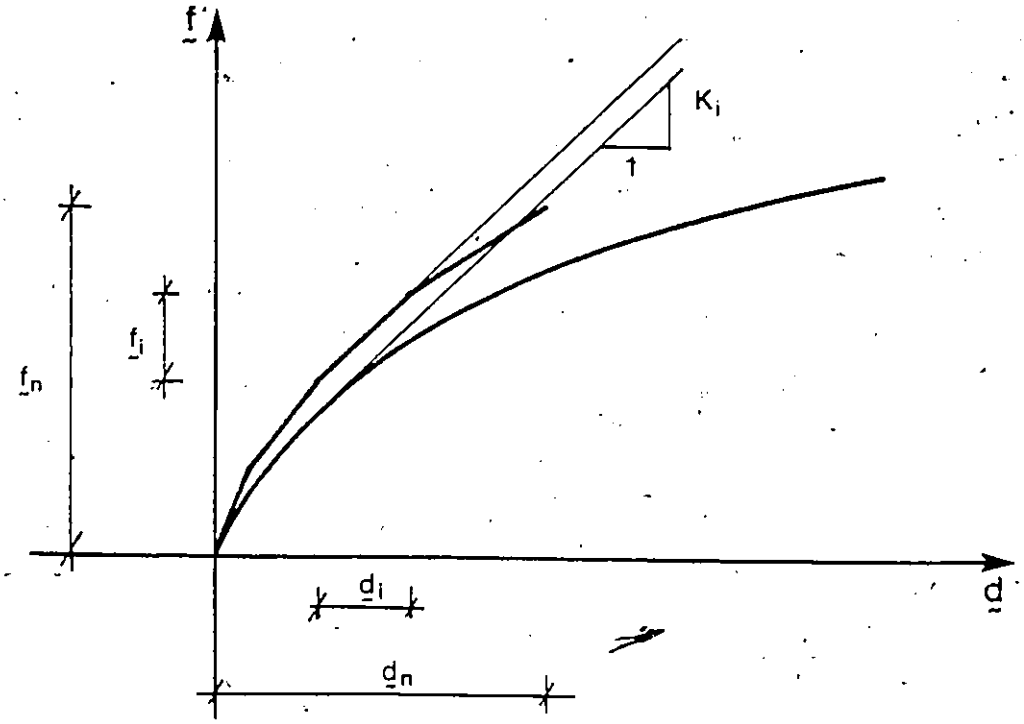
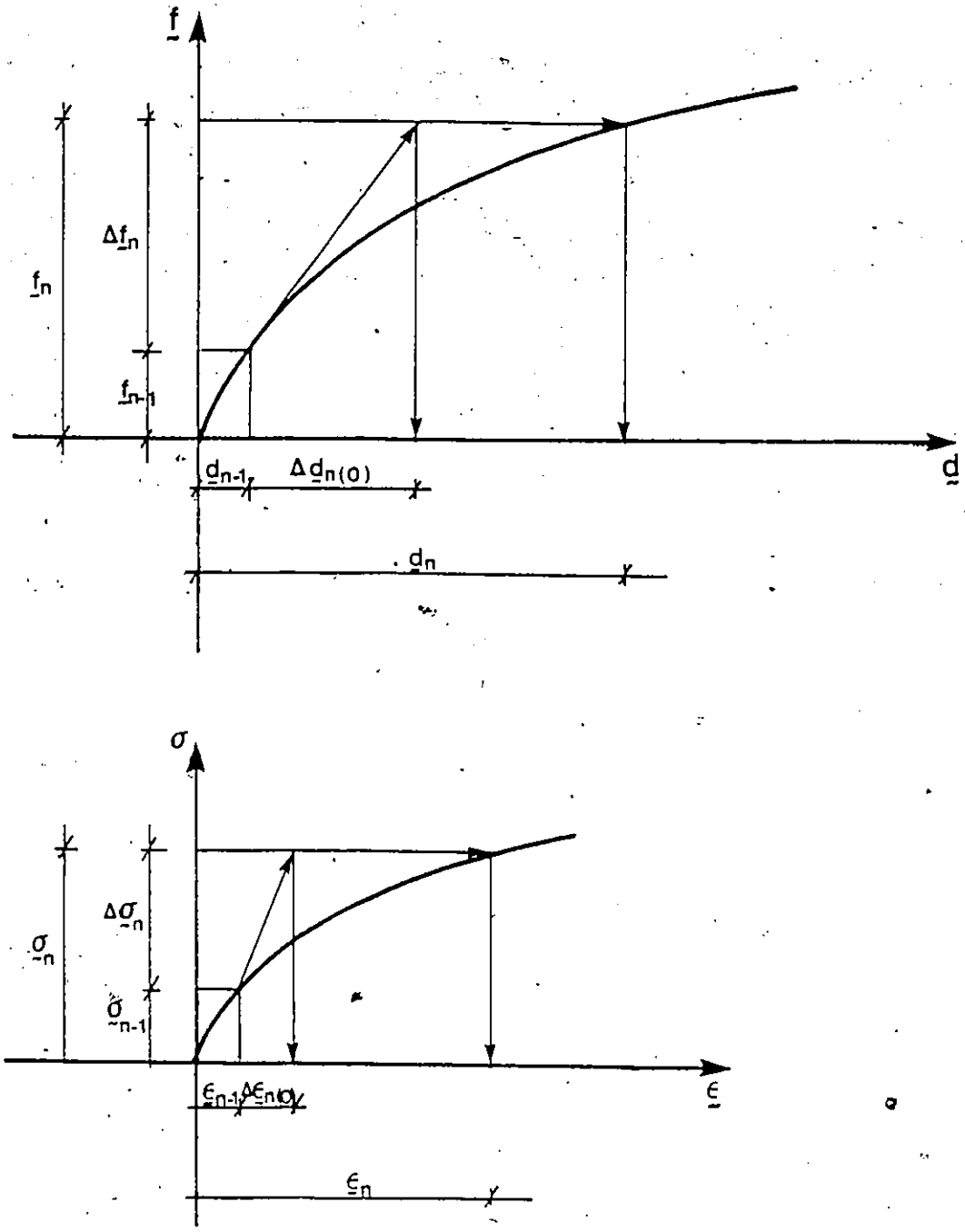


Figure 5.5: Pure Incremental Procedure



$$\Delta\underset{\sim}{\varepsilon}^{p}_{(i)} = \frac{\underset{\sim}{c}_{G(i)}^{T}\left[\Delta\underset{\sim}{\varepsilon}_{(i)} - \Delta\underset{\sim}{\varepsilon}^{p}_{(i-1)}\right]}{A}$$

Fig. 5.6 Initial Strain Method

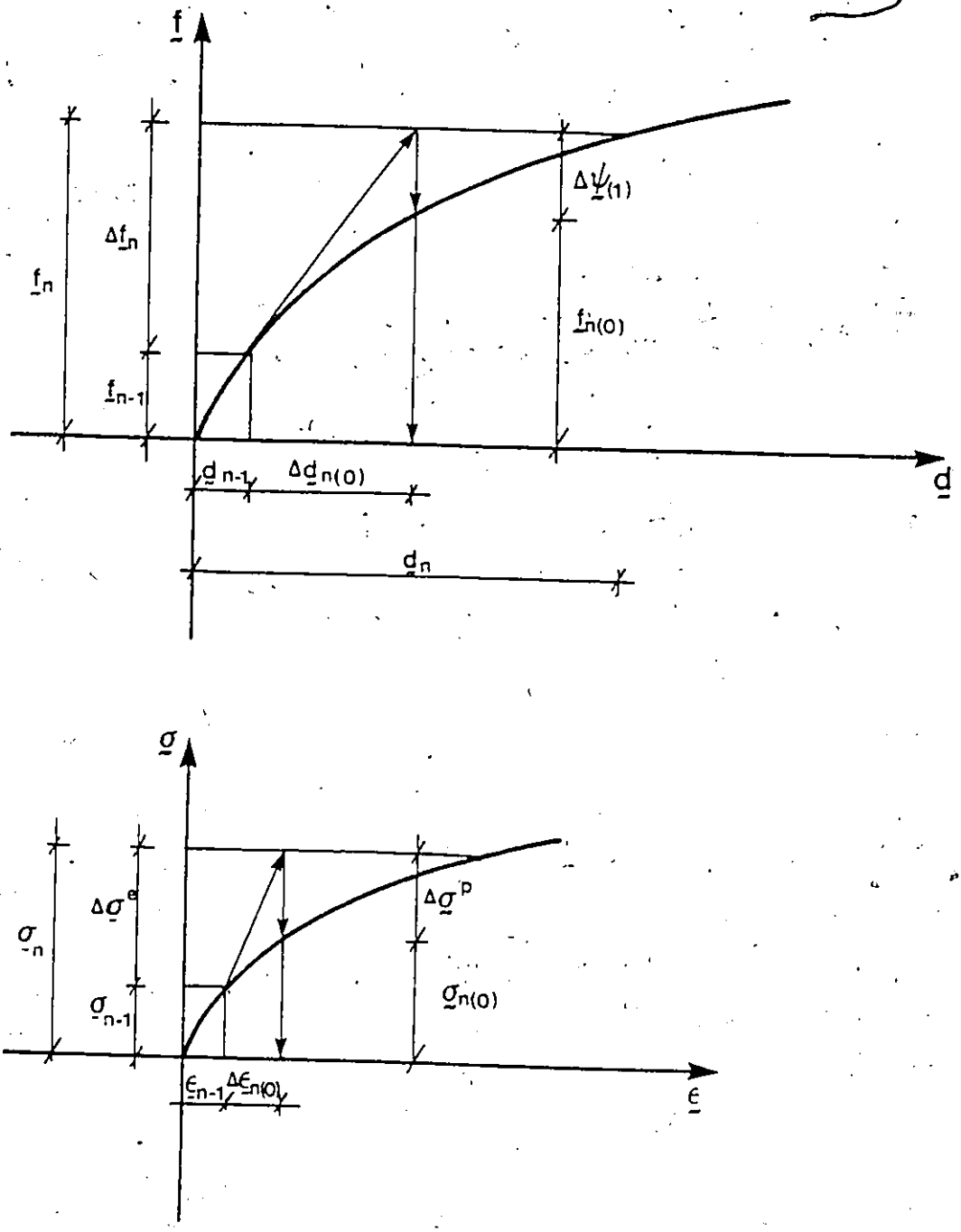


Fig. 5.7 Initial Stress Method

5.7.1 Initial Stress Method

In contrast to the "Initial Strain" method, the "Initial Stress" method searches for a stress field compatible with a given strain field, and the main steps involved in the solution process are summarized in Table 5.6.

In the first place, during a load increment, a purely elastic problem is solved determining an increment of displacement $\Delta\underset{\sim}{d}_o$. For that current $\Delta\underset{\sim}{d}_o$, the corresponding strain and consequently the compatible stress field $\sigma_{n(i-1)}$ are calculated in steps 2 to 6.

The load $\underset{\sim}{f}_{n(i-1)}$, equivalent to the balanced stress field $\underset{\sim}{\sigma}_{n(i-1)}$, is then determined as in step 7. However, this load will not be in general equal to the applied load $\underset{\sim}{f}_n$ due to the non-linearity of the problem. Accordingly a residual load $\Delta\underset{\sim}{\psi}_{n(i)}$ is calculated as in step 8.

At the second stage of computation, this residual load is redistributed and balanced by means of an iterative process described by steps 9 and steps 2 to 8.

During the next balancing cycle, the stiffness matrix can be either kept constant, Initial Stiffness Method "ISM" (Zienkiewicz et al. 1969), or be updated, Tangential Stiffness Method "TSM" (Owen and Hinton 1981) as illustrated in Figs. 5.8a,b. The TSM requires few iterations at the expense of high computational cost due to the recalculation of the stiffness matrix.

On the other hand in the ISM, the amount of calculations at every iteration is less, but more iterations are needed for convergence.

The selection of the method depends on the nature of the problem being analysed.

For materials exhibiting softening behaviour, the TSM cannot be employed since, by updating the stiffness, some diagonal terms become negative thus ill-conditioning the solution process.

In order to calculate the mobilized stress increment for the current strain increment, $\Delta\underset{\sim}{\varepsilon}_j$ shown in step 6, two different procedures are available. Zienkiewicz et al. (1969) calculated the mobilized elastic stress increment $\Delta\underset{\sim}{\bar{\sigma}}_j$ as the equilibrating stress due to the mobilized elastic stress increment $\Delta\underset{\sim}{\sigma}^e$, Eqs. 5.42.

Hermann et al. (1983) instead, assumed the elasto-plastic matrix C^{ep} to vary according to a θ-Wilson interpolation scheme between the two next steps.

$$\Delta\underset{\sim}{\bar{\sigma}}_j = [\theta C_j^{ep} + (1-\theta)C_{j-1}^{ep}]\Delta\underset{\sim}{\bar{\varepsilon}}_j \qquad (5.43)$$

This approach might give better accuracy but has the drawback of requiring the prediction of the elasto-plastic matrix C_j^{ep}.

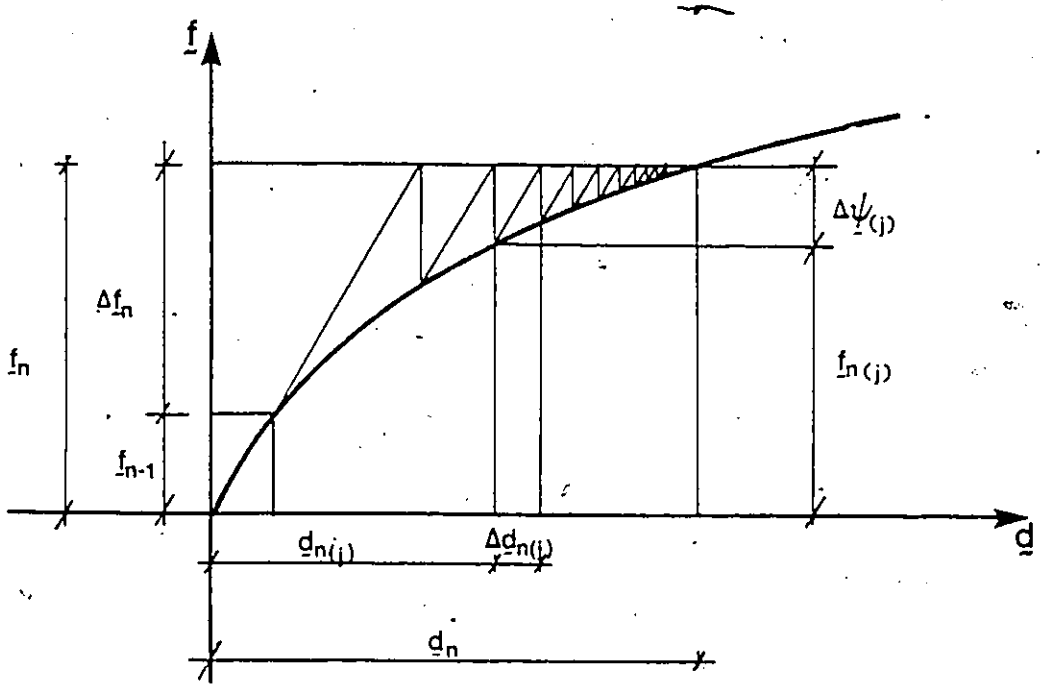


Fig. 5.8a Initial Stress Method with Initial Stiffness

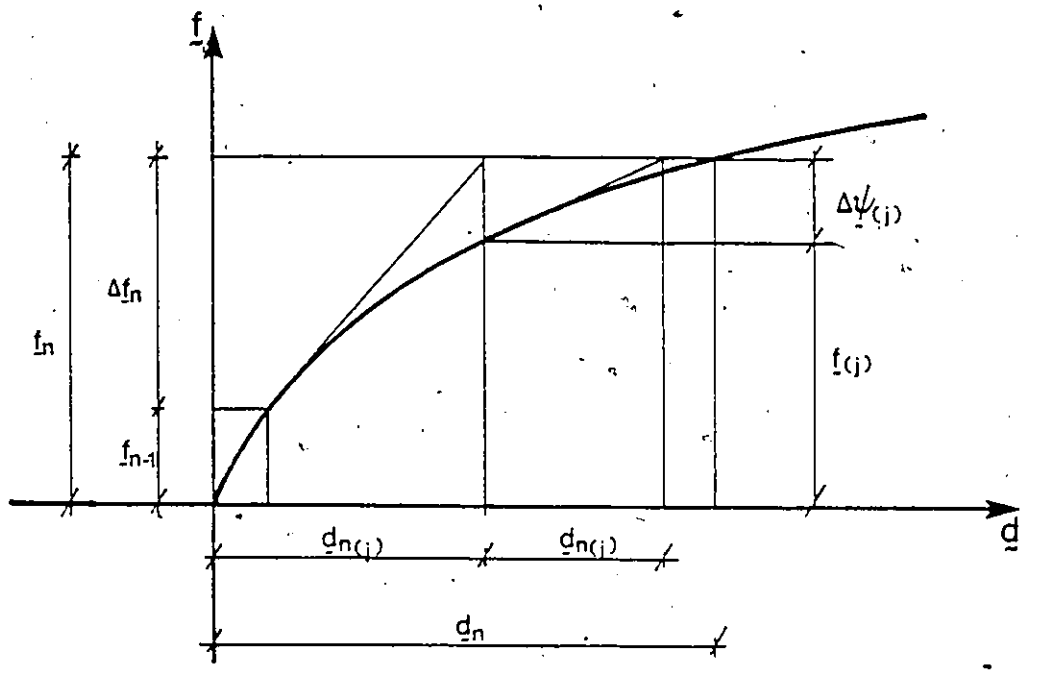


Fig. 5.8b Initial Stress Method with Tangential Stiffness

TABLE 5.6

Summary of steps followed in the Initial Stress Method

During a particular load increment $\Delta \underset{\sim}{f}_n$

1. Calculate the increment in displacement by solving
$$\Delta \underset{\sim}{d}_{n(o)} = (K_{n-1})^{-1} \Delta \underset{\sim}{f}_n$$

2. Compute corresponding strain increment as
$$\Delta \underset{\sim}{\varepsilon}_{n(o)} = B \cdot \Delta \underset{\sim}{d}_{n(o)}$$

3. Calculate the current stress level in the assumption of linear elasticity
$$\underset{\sim}{\sigma}_n = \underset{\sim}{\sigma}_{n-1} + C^e \Delta \underset{\sim}{\varepsilon}_{n(o)} = \underset{\sim}{\sigma}_{n-1} + \Delta \underset{\sim}{\sigma}$$

4. Check if the current stress lies inside the yield surface in which case, $\underset{\sim}{\sigma}_n$ is the actual stress mobilized in the continuum (structure) and is in equilibrium with the current level of applied forces $\underset{\sim}{f}_n$.

   If instead $\underset{\sim}{\sigma}_n$ falls outside the yield surface, plasticity develops and find the actual stress mobilized for the current displacement. $\underset{\sim}{d}_n = \underset{\sim}{d}_{n-1} + \Delta \underset{\sim}{d}_n$, in other words proceed to step 5.

5. Differentiate the stress increment contained inside the yield surface from the offset one.
$$\Delta \underset{\sim}{\sigma}^e = r\, \Delta \underset{\sim}{\sigma}^e + (1 - r) \Delta \underset{\sim}{\sigma}^e = r\, \Delta \underset{\sim}{\sigma}^e + s\, \Delta \underset{\sim}{\sigma}^e$$
$$\underset{\sim}{\sigma}_{n(o)} = \underset{\sim}{\sigma}_{n-1} + r \Delta \sigma^e$$
r: reduction factor

6. Compute the compatible stress with the current displacement (or strain increment $\Delta \underset{\sim}{\varepsilon}$)
$$\overline{\underset{\sim}{\sigma}}_{n(i-1)} = \overline{\underset{\sim}{\sigma}}_{n(o)} + \sum_{j=1}^{m} \Delta \overline{\underset{\sim}{\sigma}}_j$$
$\Delta \overline{\underset{\sim}{\sigma}}_j$ mobilized stress increment defined in Eq. 5.42

7. Compute the equivalent load to the $\overline{\sigma}_{n}$ stress

$$\overline{f}_{n(i-1)} = \int_{\Omega} B^{T} \overline{\sigma}_{n(i-1)} d\Omega$$

8. Calculate the residual load yet to be balanced

$$\Delta\psi_{n(i)} = f_{n} - \overline{f}_{n(i-1)}$$

9. Calculate the additional displacement for the residual load

$$\Delta d_{n(i)} = K^{-1} \Delta\psi_{n(i)}$$

Go to Step 2

Balance of excess stresses

$$\Delta\overline{q}_{j} = \Delta\overline{q}^{e} - (dl\ \varepsilon_{F})_{j} = \Delta\overline{q}^{e} - \Delta\overline{q}_{j}^{P} \qquad (5.42a)$$

where

$$\Delta\overline{q}^{e} = \frac{s}{m} \Delta q^{e} = \frac{s}{m} C^{e} \Delta\varepsilon_{n(j)} = C^{e} \Delta\overline{\varepsilon}_{j} \qquad (5.42b)$$

$$dl = \frac{\nabla_{\sigma} G^{T} \Delta\overline{q}^{e}}{A + \varepsilon_{G}^{T} \nabla_{E}} \qquad (5.42c)$$

thus:

$$\Delta\overline{\sigma}_{j} = C^{ep}_{j-1} \Delta\overline{\varepsilon}_{j} \qquad (5.42d)$$

m: imposed number of balancing cycles for a better calculation of $\Delta\sigma_{j}^{P}$.

5.7.2 Reduction factor

An important aspect of the elasto-plastic analysis concerns the calculation of the reduction factor r. In previous publications (Zienkiewicz 1972b, Owen and Hinton 1980), it was determined by simple ratio calculations.

Unfortunately this only holds for simple stress paths such as the conventional triaxial loading path and may lead to erroneous answers in a more general context.

5.8 EFFECTIVE STRESS ELASTO-PLASTIC ANALYSIS

The underlying formulation of the elasto-plastic implementation in ESEPA is based on Eq. 5.18, which takes into account the buoyant effect of the pore pressures in an effective stress analysis.

As far as constitutive relationships are concerned, the models described earlier have been consistently implemented, the main objective being focussed on the generalized Burland which provides the possibility to follow softening behaviour of materials.

In contrast to what is conventionally incorporated in elasto-plastic programs (Zienkiewicz 1970, Owen and Hinton 1980), a better evaluation of the reduction factor r along with a more correct calculation of the flow vector $\underset{\sim}{a}_2$, Eq. 5.8 are herein performed.

5.8.1 Structure of ESEPA (Effective Stress Elasto-Plastic Analysis)

The flow chart in Fig. 5.9 illustrates the overall structure of ESEPA.

In saturated soils, the buoyant effect of the initial pore pressures along with the pre-existing stress distribution, are taken into consideration by Subroutine LOAD, through calculation of the equivalent nodal forces given in Eq. 5.17a.

The initial position of the yield surface, a measure of the preconsolidation pressure at each Gauss integration point, is calculated in Subroutine PREVOS on the basis of the information supplied by the user.

The appropriate applied load increment is then computed in Subroutine INCREM. In order to identify the different alternatives in the solution algorithm e.g. ISM or TSM or a combination of both, an indicator is set up in Subroutine ALGOR.

The elasto-plastic stiffness matrix given in Eq. 5.19b is calculated in Subroutine STIFFP, before the solution process in Subroutine FRONT can start.

Subroutine RESIDU is the main core of the program, where elasto-plastic equilibrium is performed and its functions will be discussed in detail in the next section.

The residual forces ψ are calculated in Subroutine CONVER according to step 8 in Table 5.6. The convergence is then monitored by verifying if the norm of the residual forces, as calculated in Eq. 5.44, is less than a given tolerance.

$$\frac{\sum_{i=1}^{N} (\underset{\sim}{\psi}_i)^2}{\sum_{i=1}^{N} (\underset{\sim}{f}_i)^2} * 100 < \varepsilon \qquad (5.44)$$

where N denotes the total number of nodes and r the iteration number.

Finally the converged displacements, reactions and stresses are printed for each load increment in Subroutine OUTPUT.

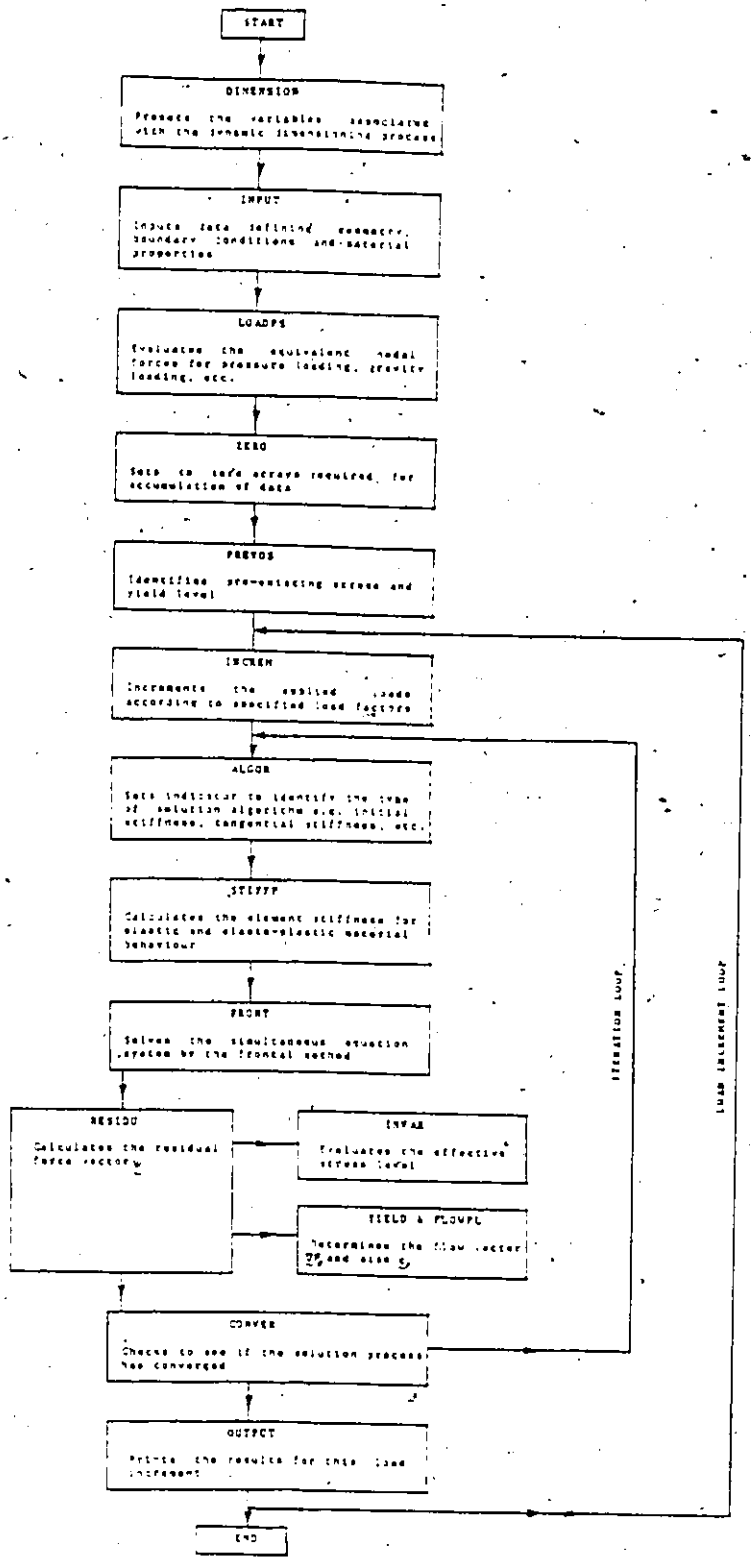


Fig. 5.9 Flowchart ESEPA

5.8.2 Subroutine RESIDU. (RESIDUal)

The function of this subroutine is to evaluate the nodal forces statically equivalent to the stress field satisfying elasto-plastic conditions, according to the "Initial Stress" method discussed in Sect. 5.7.1.

Fig. 5.10 illustrates the subprogram structure while the specific subroutine listing can be found in Appendix C.

Two main loops, one on the number of elements and the other on the number of integration points per element, review the mobilized stresses at all Gauss integration points.

At each Gauss integration point, the current stress $\underset{\sim}{\sigma}_{i(n)}$ is thereby determined after having calculated the incremental displacement $\Delta\underset{\sim}{d}_n$ and strain $\Delta\underset{\sim}{\varepsilon}_n$ in linear elasticity.

Subroutine PLASTC checks if the stress $\underset{\sim}{\sigma}_{i(n)}$ lies inside or outside the current yield surface.

If $\underset{\sim}{\sigma}_{i(n)}$ exceeds the elastic limit, then the intersection of the current stress increment $\Delta\underset{\sim}{\sigma}_e$ with the yield surface is determined in Subroutine FACTOR through a reduction factor r. Subsequently the actual level of stress $\underset{\sim}{\sigma}_{i(j)}$ is computed in Subroutine ELPLAS according to Eqs. 5.42.

For the case of linear elastic perfectly plastic models, the drift of the balanced stress $\underset{\sim}{\sigma}_{i(j)}$ is brought back on the yield surface by means of Subroutine BACK (refer to Sect. 5.8.5).

Finally the integration of the balanced stress gives the equivalent nodal forces.

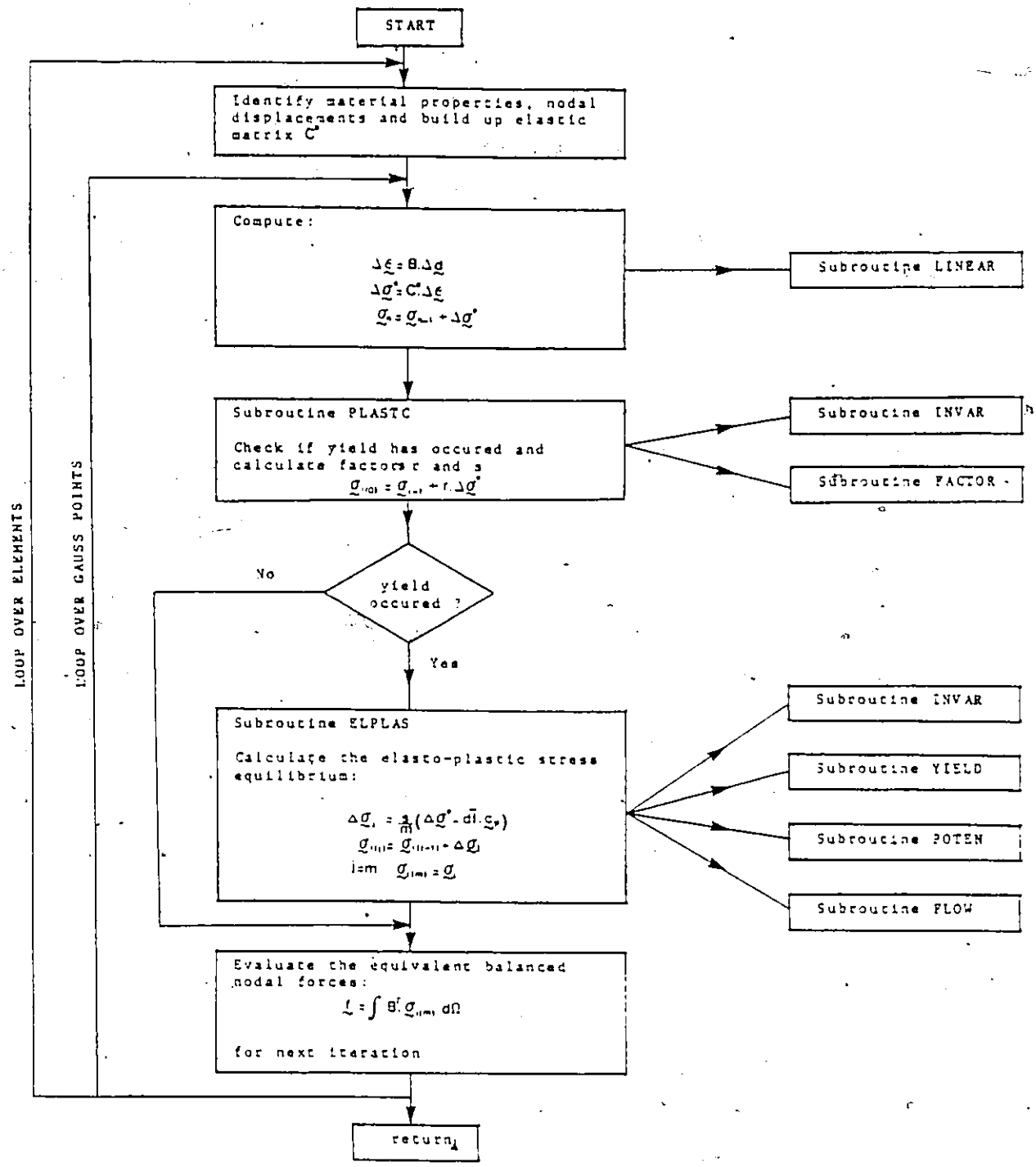


Fig. 5.10 Flowchart RESIDU

5.8.3 Subroutine PLASTC (PLASTiCity)

This subroutine determines at each Gauss integration point if the stress point $\underset{\sim}{\sigma}_n$ lies inside or outside the current yield surface. The flowchart is reported in Fig. 5.11 whereas programming details are found in Appendix C.

Plasticity can be detected by evaluating the location of the initial stress $\underset{\sim}{\sigma}_{n-1}$ and the incremented one $\underset{\sim}{\sigma}_n$ relative to the pre-existing yield surface (preconsolidation). This is accomplished by verifying the sign of the current yield function evaluated for the stresses $\underset{\sim}{\sigma}_{n-1}$ and $\underset{\sim}{\sigma}_n$.

a) Linear elastic perfectly plastic models

$$F = q + \bar{M}\, p - \bar{N} = 0 \qquad (5.45a)$$

b) Generalized Burland model

$$F = (p + \bar{\beta})^2 + \frac{(q - \bar{\alpha})^2}{\bar{n}^2} - \bar{h}^2 = 0 \qquad (5.45b)$$

$\bar{M}$, $\bar{N}$, $\bar{\alpha}$, $\bar{\beta}$, $\bar{n}$ and $\bar{h}$ are parameters previously defined and are characteristics of the previous yield surface.

If the stress $\underset{\sim}{\sigma}_n$ is located outside the pre-existing yield surface the reduction factor r is accordingly calculated in order to determine the amount of excess stress which has yet to be balanced.

Finally the number of balancing cycles needed in the stress reduction process is determined through an empirical relationship given by

$$\text{MSTEP} = \frac{\text{ESCUR}}{\text{DENOM}} * \text{CONSTANT} \qquad (5.46)$$

$$\text{DENOM} = \begin{cases} \text{PREYS} & \text{if PREYS} \neq 0 \\ \text{SIGYS} & \text{PREYS} = 0 \end{cases}$$

ESCUR gives a measure of the excess stress, while PREYS and SIGYS are measures related to the pre-existing and current yield surfaces respectively.

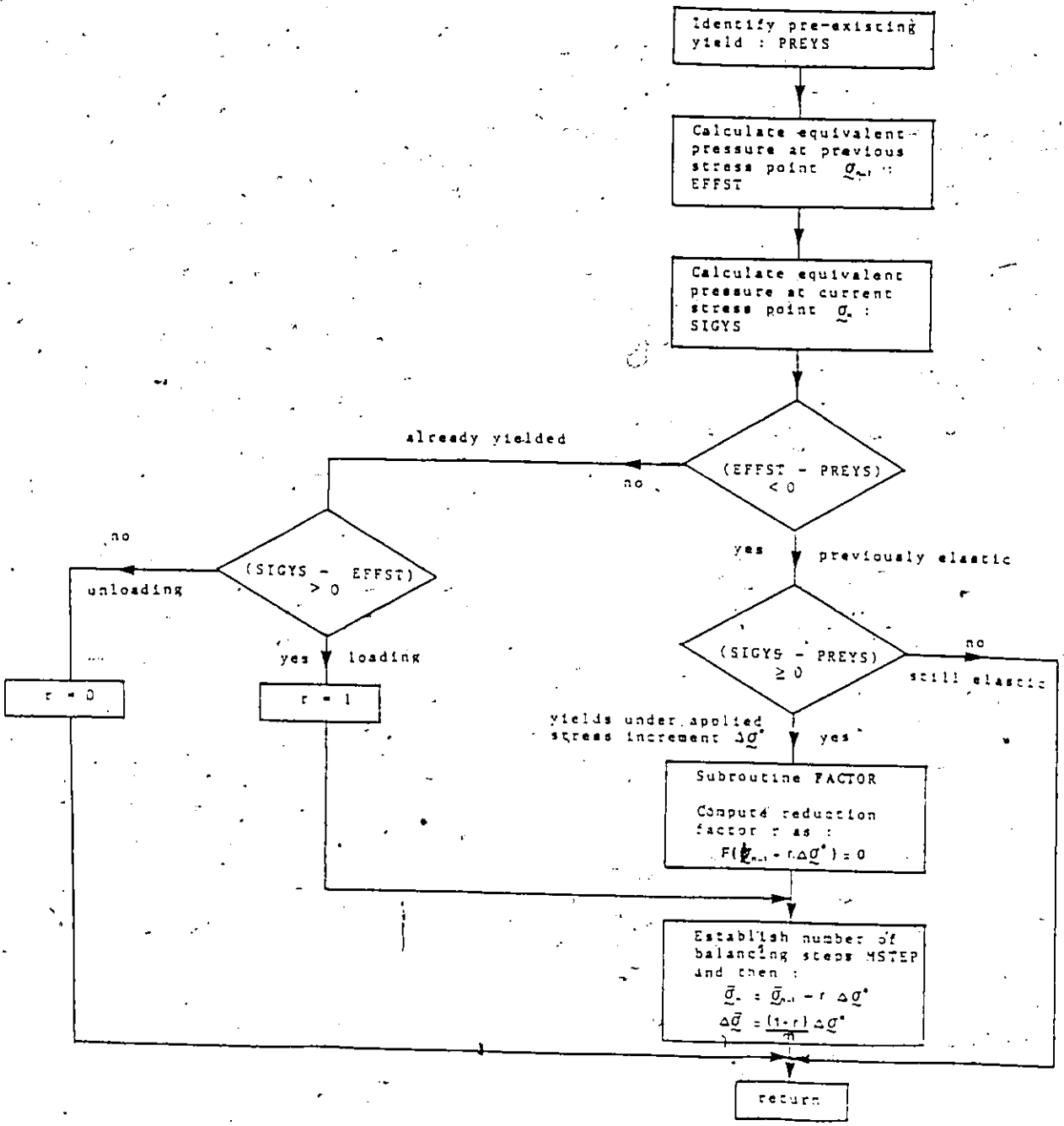


Fig. 5.11 Flowchart PLASTC

5.8.4 Calculation of reduction factor r

The procedure consists of scaling the stress increment vector on the yield surface by means of a reduction factor r as follows:

$$\bar{\underset{\sim}{\sigma}} = \underset{\sim}{\sigma} + r\, d\sigma \qquad (5.47a)$$

$$0 \leq r \leq 1 \qquad (5.47b)$$

$$G(\underset{\sim}{\sigma} + r d\underset{\sim}{\sigma}, k) = 0 \qquad (5.48)$$

A similar approach was adopted by Siriwardane et al. (1983) for some isotropic functions, and later extended to quasi-isotropic functions by Fusco (1985).

The invariants $\bar{I}_1, \bar{J}_2$ and $\bar{J}_3$ of the reduced stress $\bar{\underset{\sim}{\sigma}}$ can be expressed, after some mathematical manipulation, in terms of those of $\underset{\sim}{\sigma}$ and $d\underset{\sim}{\sigma}$ as reported in Table 5.7.

Subsequent substitution of the reduced invariants in the yield function, as shown in Table 5.8, permits a solution of the reduction factor r. For isotropic yield functions, r is

calculated by trivial solution of a quadratic equation of the type reported in the previous table, while for quasi-isotropic yield functions the solution requires an iterative process such as the "bisection" method. The reduced invariants are then checked to verify the yield function at every successive value of r. In this case the solution always contains some degree of error which may be artificially eliminated by eventually bringing back the reduced stress on the yield surface.

Therefore in this program, the search of r is performed within a certain accuracy in Subroutine BISECT which is explained in details in the next section.

TABLE 5.7

Determination of the reduction factor

$$\bar{\sigma}_{ij} = \sigma_{ij} + rd\sigma_{ij} \qquad \therefore \qquad \bar{s}_{ij} = s_{ij} + rds_{ij}$$

$$\bar{I}_1 = \bar{\sigma}_{ii} = I_1 + rdI_1$$

$$\bar{J}_2 = \frac{1}{2}\bar{s}_{ij}\bar{s}_{ij} = J_2 + 2r\,Z + r^2\,dJ_2$$

$$\bar{J}_3 = \det(\bar{s}_{ij}) = J_3 + r\,Y + r^2 W + r^3\,dJ_3$$

where

$$Z = \frac{1}{2}(s_{11}ds_{11}+s_{22}ds_{22}+s_{33}ds_{33}) + s_{12}ds_{12} + s_{13}ds_{13} + s_{23}ds_{23}$$

$$Y = A + B - C$$

$$A = s_{11}s_{22}d_{33} + s_{11}s_{33}ds_{22} + s_{22}s_{33}ds_{11}$$

$$B = 2(s_{12}s_{13}ds_{23} + s_{12}s_{23}ds_{13} + s_{13}s_{23}ds_{12})$$

$$C = 2(s_{11}s_{23}+s_{22}s_{13}ds_{13}+s_{33}s_{12}ds_{12}) + (s_{23}^2 ds_{11}+s_{13}^2 ds_{22}+s_{12}^2 ds_{33})$$

$$W = D + E - F$$

$$D = s_{11}ds_{22}ds_{33} + s_{22}ds_{11}ds_{33} + s_{33}ds_{11}ds_{33}$$

$$E = 2(s_{12}ds_{13}ds_{23} + s_{12}ds_{23}ds_{13} + s_{13}ds_{23}ds_{12})$$

$$F=2(s_{23}ds_{11}ds_{23}+s_{13}ds_{22}ds_{13}+s_{12}ds_{33}ds_{12})+(s_{11}ds_{23}^2+s_{22}ds_{13}^2+s_{23}ds_{12}^2)$$

For 2D problem

$$Z = \frac{1}{2}\left(s_{11}ds_{11} + s_{22}ds_{22} + s_{33}ds_{33}\right) + s_{12}ds_{12}$$

$$A = s_{11}s_{22}ds_{33} + s_{11}s_{33}ds_{22} + s_{22}s_{33}ds_{11}$$

$$B = 0$$

$$C = 2s_{33}s_{12}ds_{12} + s_{12}^{2}ds_{23}$$

$$D = s_{11}ds_{22}ds_{33} + s_{22}ds_{11}ds_{33} + s_{33}ds_{11}ds_{22}$$

$$E = 0$$

$$F = 2s_{12}ds_{33}ds_{12} + s_{33}ds_{12}^{2}$$

TABLE 5.8

Solution example of the reduction factor

$$\bar{p} = \frac{\bar{I}_1}{3} = p + r\,dp \quad ; \qquad d = \frac{N}{M} = c\ \mathrm{ctg}\ \phi$$

$$\bar{q} = \sqrt{3}(\bar{J}_2^{1/2} = (\bar{J}_2 + 2r\,Z + r^2 dJ_2)^{1/2}$$

$$\bar{\theta} = \frac{1}{3}\arcsin\left(-\frac{3\sqrt{3}}{2}\frac{\bar{J}_3}{\bar{J}_2^{3/2}}\right)$$

Linear yield function

1) $F = \bar{q} + M\,\bar{p} - N = 0$

Generalized Burland

2) $F = (\bar{p} + \beta)^2 + \frac{(\bar{q} - \alpha)^2}{n^2} - h^2 = 0$

3) $r^2 A + 2r\,B + C = 0$

Type	A	B	C
Linear, $M \neq 0$	$dq^2 - M^2 dp^2$	$3Z - M\,dp(Mp-N)$	$q^2 - (Mp-N)^2$
Burland, $\alpha = 0$	$dp^2 + \frac{dq^2}{n^2}$	$\frac{3Z}{n^2} + dp(p+\beta)$	$(p+\beta)^2 + \frac{q^2}{n^2} - h^2$

5.8.5 Subroutines ELPLAS (ELasto-PLAStic) and BACK

In Subroutine ELPLAS, the stress is balanced stepwise in order to reduce inevitable numerical errors.

At each step the yield function is updated and may contract or expand depending mainly on the value of the plastic modulus A.

In contrast to the linear elastic perfectly plastic models, the yield function in the generalized Burland model and the plastic modulus A may vary.

The value of A is controlled by the determination of the specific volume during stepwise balancing. Since the strain increment is considered to be constant during the balancing process, the specific volume is found as follow:

$$\bar{v}_{n(j)} = \bar{v}_{no} + \sum_{i=1}^{k} \Delta\bar{v} \qquad 1 \leq k \leq m \qquad (5.49a)$$

where

$$\bar{v}_{no} = 1 + r\ \Delta\varepsilon^{v}_{n(j)}\ \bar{v}_{n-1} \qquad (5.49b)$$

$$\Delta\bar{v} = \frac{s}{m}\ \Delta\varepsilon^{v}_{n(j)}\ \bar{v}_{n(j-1)} \qquad (5.49c)$$

$$\Delta\varepsilon^{v}_{n(j)} = (\Delta\varepsilon_{11} + \Delta\varepsilon_{22} + \Delta\varepsilon_{33})_{n(j)} \qquad (5.49d)$$

For linear elastic perfectly plastic models, the equilibrium performed by the only use of Eqs. 5.42 may result in an undesired drift of the final stress point from the failure surface, this due to the magnitude of the load increment, Fig. 5.12. Several methods have been proposed in order to remedy this shortcoming (Krieg et al. 1977, Schreyer et al. 1979, Owen and Hinton 1980).

Herein, it was chosen to relax the drifted stress along its deviatoric component on the yield surface. In that case it is possible to prove that, by decomposing the stress into mean and deviatoric parts, the reduction factor r can be obtained by simple ratio of the deviatoric stress $\bar{J}_2$ to J_2, provided that the mean pressure p and the angle θ are kept fixed, Fig. 5.13.

$$p_D = p_{D'} \; ; \quad \theta_D = \theta_{D'} \qquad (5.50a)$$

$$r = \left(\frac{\bar{J}_2}{J_2}\right)^{1/2} \qquad (5.50b)$$

where $\bar{J}_2$ is the second stress invariant associated with point D' which satisfies the yield function at mean pressure p_D and angle θ_D and J_2 is the value of the second invariant associated with point D.

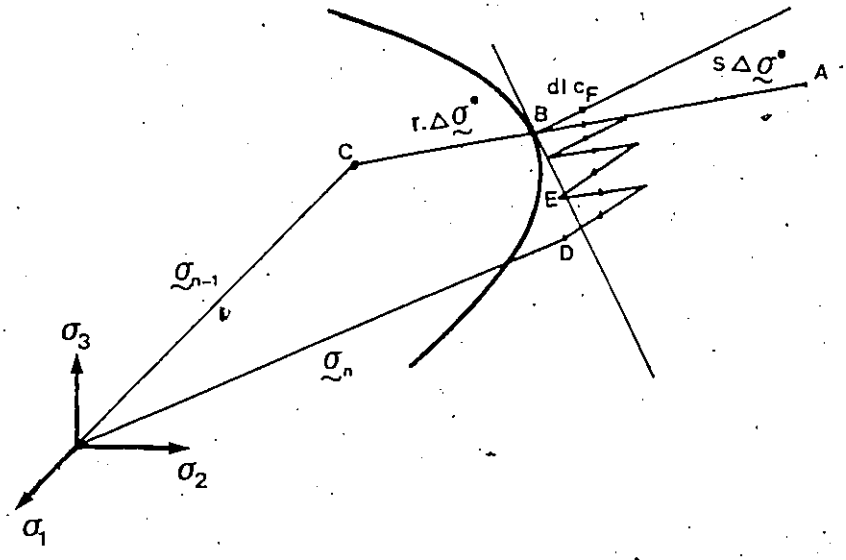


Fig. 5.12 Graphical representation of the stress balancing process

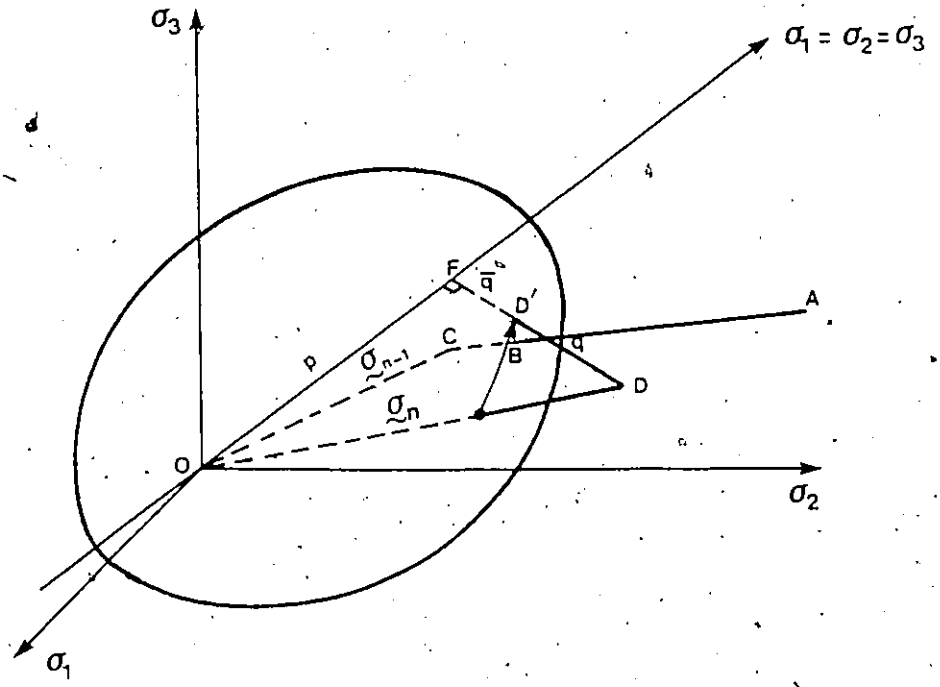


Fig. 5.13 Graphical representation of stress relaxation on the yield surface

5.9 NUMERICAL EXAMPLES

To demonstrate the capabilities of ESEPA, the basic elasto-plastic features were first tried out in the simple case of the triaxial test.

The one-element mesh together with the imposed boundary conditions are shown in Fig. 5.14. In the present example, an isotropic starting stress of $\sigma_r = \sigma_\theta = \sigma_z = -100.00$ was applied. The additional input parameters such as material properties are reported in Table 5.9 and were consistently used throughout the entire numerical analysis undertaken in this chapter and Chapter 6.

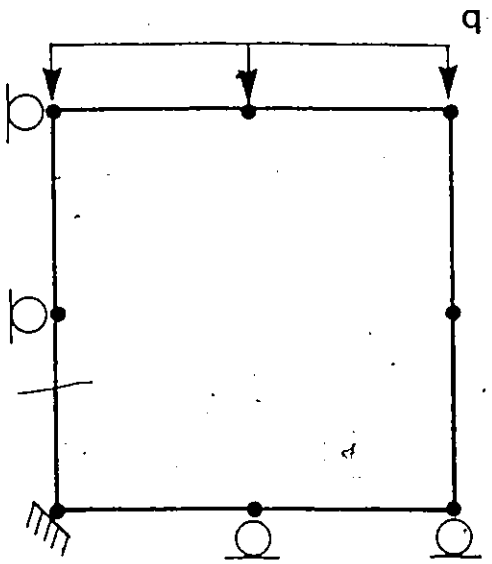


Fig. 5.14 One-element mesh layout

TABLE 5.9

Material properties used in stress and consolidation analyses

Elastic parameters:

$E = 10\ 000$ $\qquad \nu = 0$

Failure parameters:

von Mises and Tresca

a) $N = 94.8680, \quad M = 0 \quad \therefore \quad c = 47.4341, \quad \phi = 0$

b) $N = 100, \quad M = 0 \quad \therefore \quad c = 50, \quad \phi = 0$

For Drucker-Prager and Mohr-Coulomb

$N = 10, \quad M = 1 \quad \therefore \quad c = 4.7434, \quad \phi = 25.3769$

Generalized Burland Parameters:

$\lambda = 0.2 \qquad \chi = 0.02 \qquad \bar{v} = 2$

Permeability

$K = 4 \times 10^{-6}$

5.9.1 Linear elastic perfectly plastic failure criteria

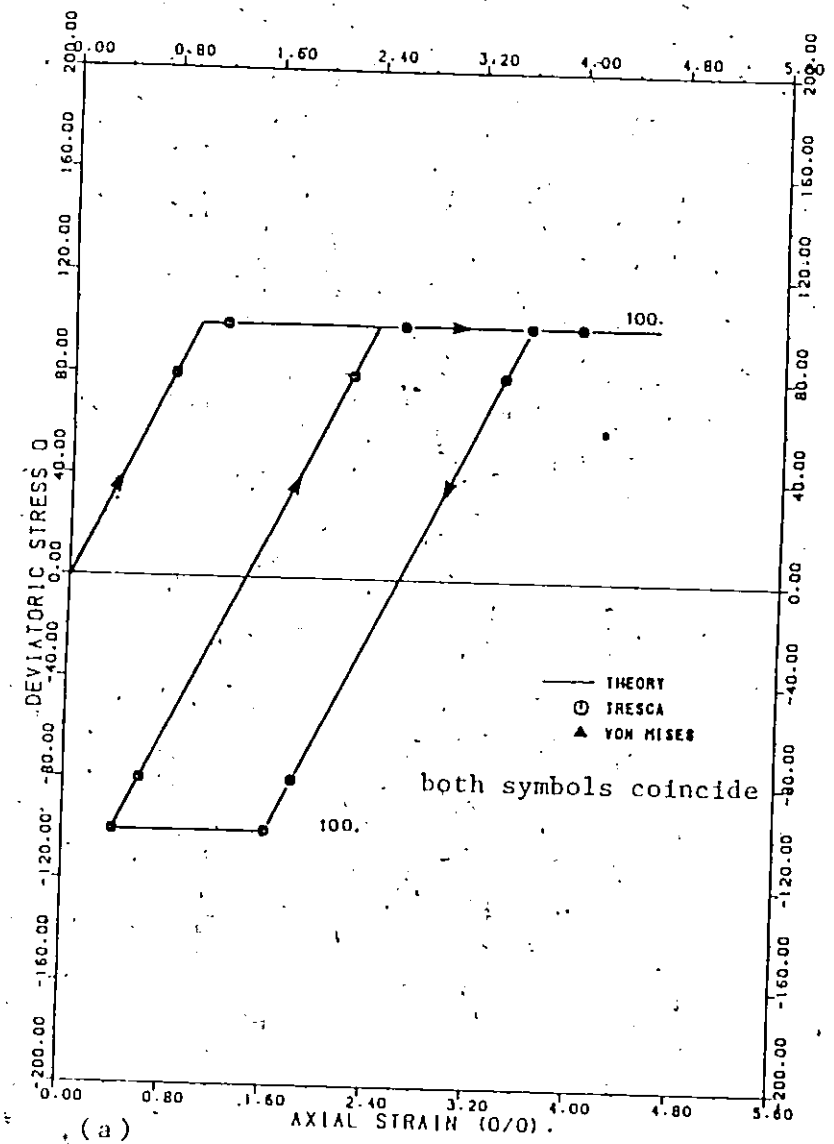
Figures 5.15a and 5.16a illustrate the numerical results obtained for the four common failure criteria in axisymmetric conditions. They are in complete agreement with the theory. In fact, the stress-strain curves revealed a linearly increasing section in the domain of recoverable strains and a perfectly flat plateau at failure. The Tresca and von Mises failure criteria coincided in both loading and unloading regimes however Mohr-Coulomb and Drucker-Prager criteria do not. In fact, as expected upon unloading, the Mohr-Coulomb failure criterion gives a lower yield stress than the Drucker-Prager criterion. Turning to figures 5.15b and 5.16b, unrealistically high plastic volumetric expansions are predicted especially for Mohr-Coulomb and Drucker-Prager failure criteria.

In plane strain conditions, the material responses were found to be quite different. The stress-strain curves either showed a flat plateau at failure (Tresca and Mohr-Coulomb) or continued to increase (Drucker-Prager and von Mises) as illustrated in Figs. 5.17 and 5.18.

To account for these basic post-yield behaviours exhibited by the two above sets of failure criteria, reference is made to the geometrical representation of their respective failure surfaces. In fact the limiting deviatoric stress q is function of the angle θ for both Mohr-Coulomb and Tresca failure criteria but not for the Drucker-Prager and von Mises criteria.

At first θ remains constant and equal to 30° until the stress point touches the failure surface. As further displacement is imposed, θ changes and the second stress σ_2 perpendicular to the plane is no longer equal to zero. Considering elasto-plastic equilibrium, the mobilized vertical stress σ_1 reaches a stable value although q decreases. This accounts for the flat part of the stress-strain curves for both Mohr-Coulomb and Tresca failure criteria. On the other hand, for Drucker-Prager and von Mises criteria, the variation of θ does not influence q so that σ_1 keeps on increasing. It is worth mentioning that, though the above results may appear evident and trivial, they confirm the accuracy and reliability of ESEPA.

The performance of ESEPA was next tested by comparing numerical results found in the literature, Desai and Siriwardane (1983). Fig. 5.19 shows the stress-strain curve obtained by the above author for a one-element mesh loaded in plane-strain conditions assuming a Drucker-Prager failure criterion. Plastic strains were first mobilized at a stress level of 1000 psi and failure ultimately occurred at 1732 psi, in agreement with the results obtained by Davidson and Chen (1974). In contrast to these results, ESEPA shows that for Mohr-Coulomb failure criterion plastic strains first appear at 1732 psi where the curve becomes perfectly flat as theoretically expected. Instead, the Drucker-Prager failure criterion curve continuously increases after the onset of yield.



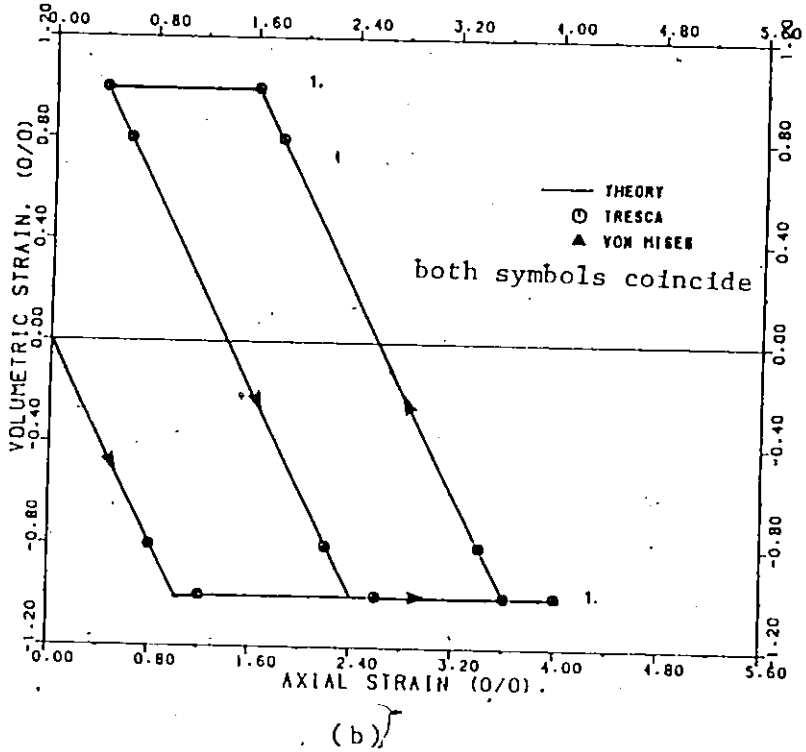


Fig. 5.15 Simulation of a triaxial test: comparison between Tresca and von Mises materials

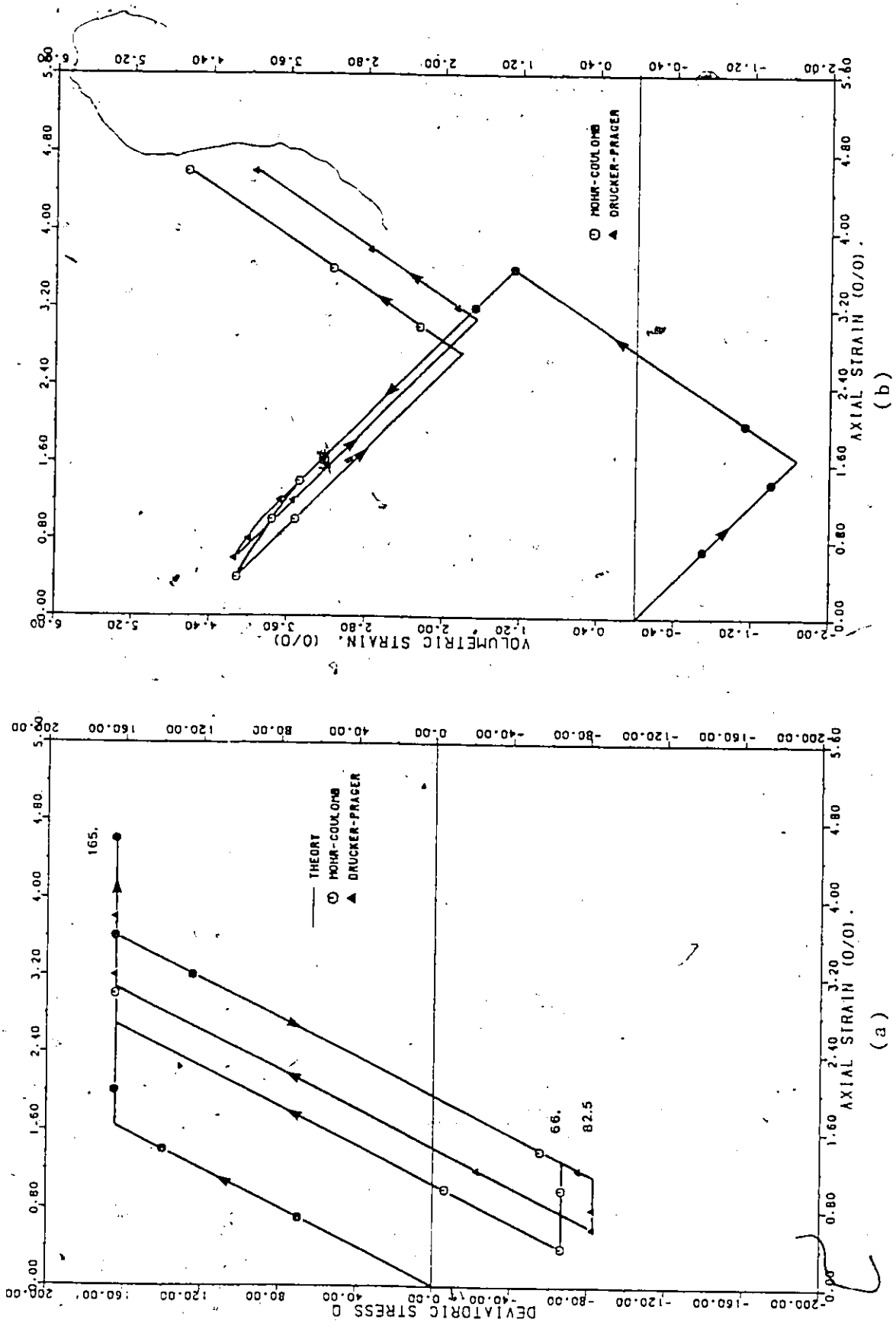


Fig. 5.16 Simulation of a triaxial test: comparison between Mohr-Coulomb and Drucker-Prager materials

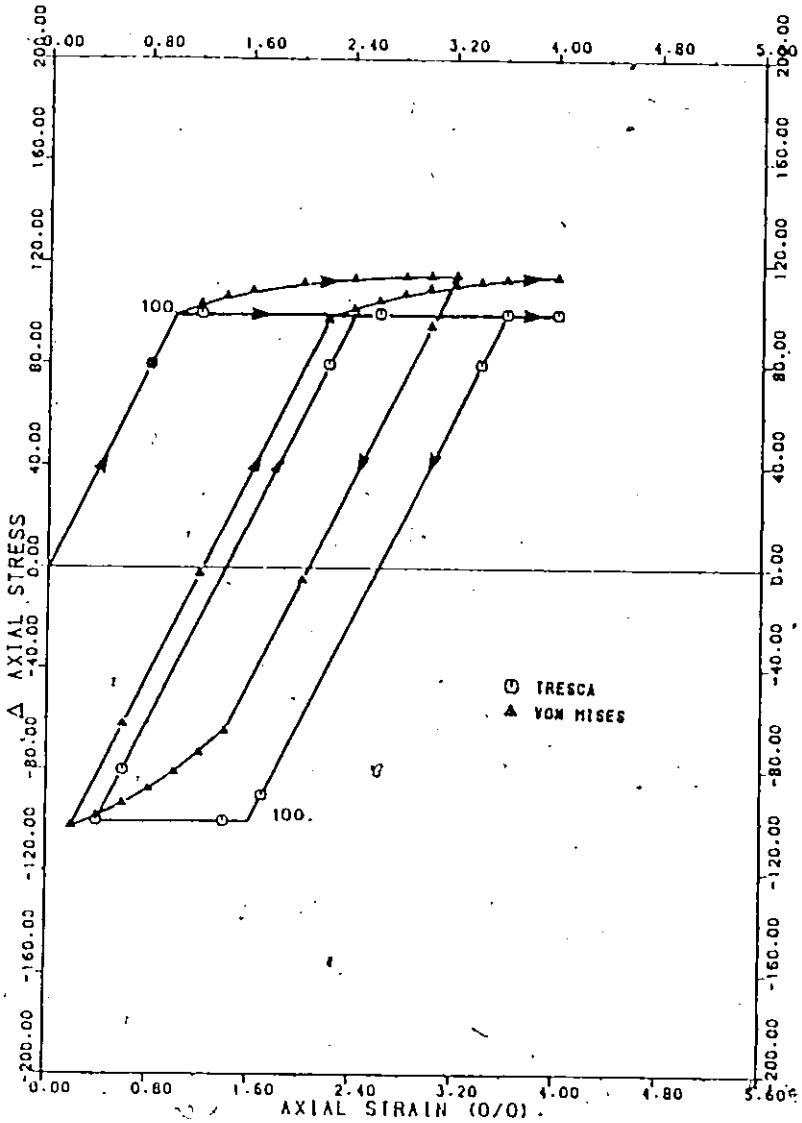


Fig. 5.17 Simple test in plane strain: comparison between Tresca and von Mises materials

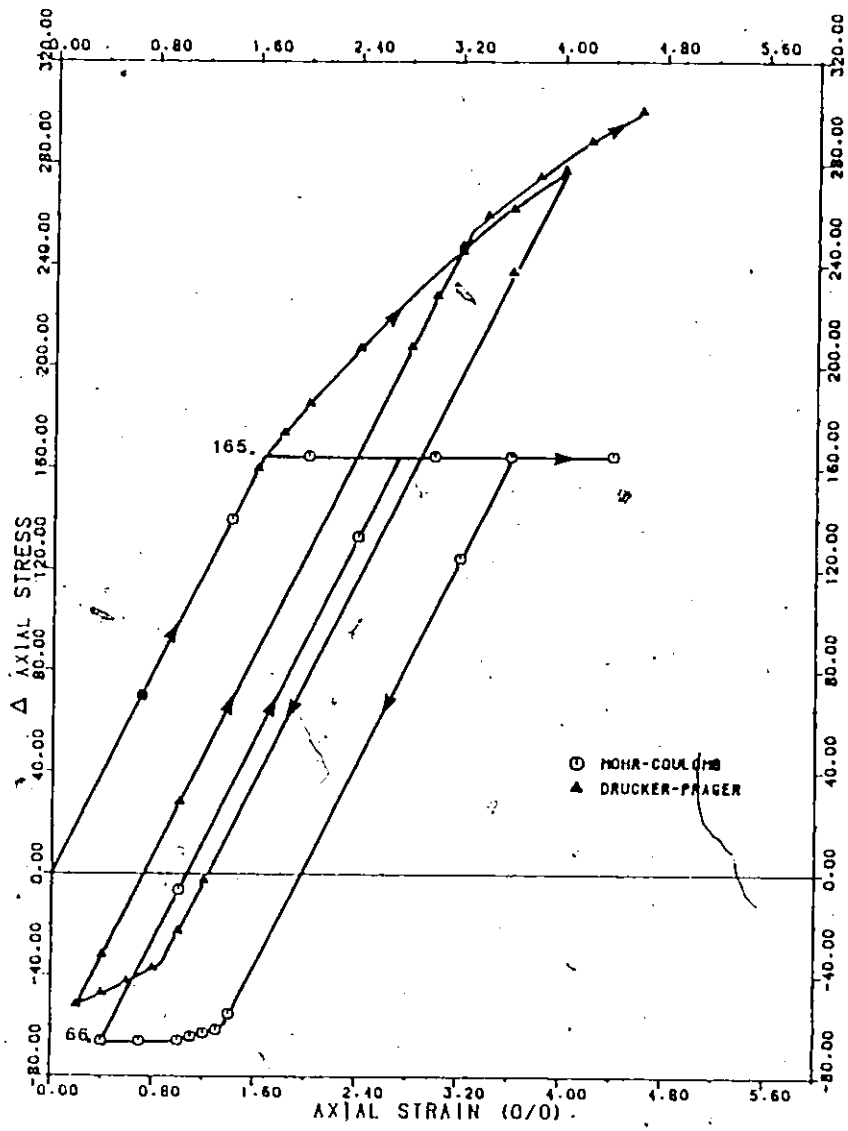


Fig. 5.18 Simple test in plane strain: comparison between Mohr-Coulomb and Drucker-Prager materials

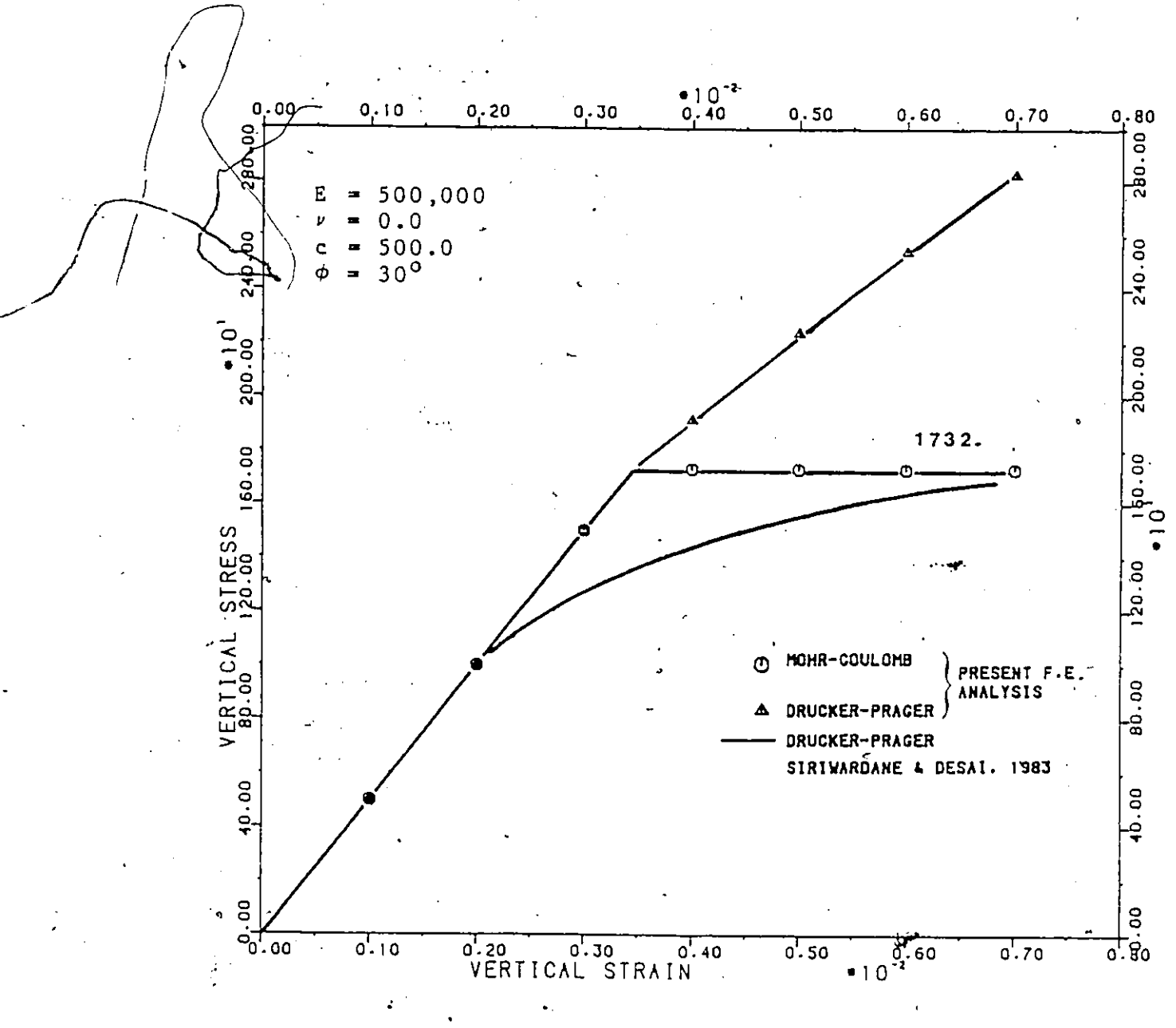


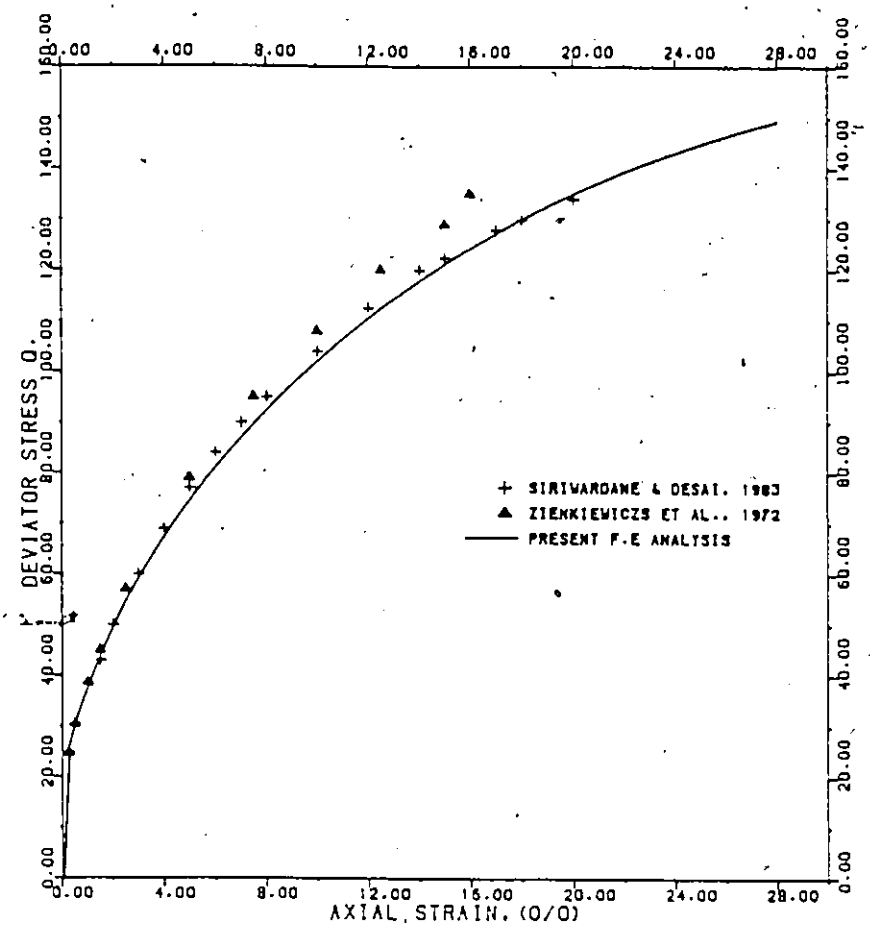
Fig. 5.19 Simple test in plane strain: comparison with numerical results by Siriwardane and Desai (1983)

5.9.2 Generalized Burland model

The capabilities of the program were demonstrated in the sophisticated generalized Burland model, herein coupled with a Mohr-Coulomb limiting surface.

Good agreement was obtained between the present analysis and the available numerical results found in the literature when the model was reduced to the original Burland model (see Fig. 5.20). The analysis was then extended to the case of an overconsolidated clay with a Mohr-Coulomb limiting failure surface. The characteristic softening behaviour in a highly overconsolidated state as well as dilation effects were simulated as shown in Fig. 5.21.

It is also interesting to notice that the model responses with Mohr-Coulomb and Drucker-Prager limiting surfaces coincided in the loading part of the stress-strain curve for normally consolidated states but differed in the unloading part. This is due to the different values M and N take in both compression and extension for the Mohr-Coulomb limiting surface (refer to Fig. 5.4).



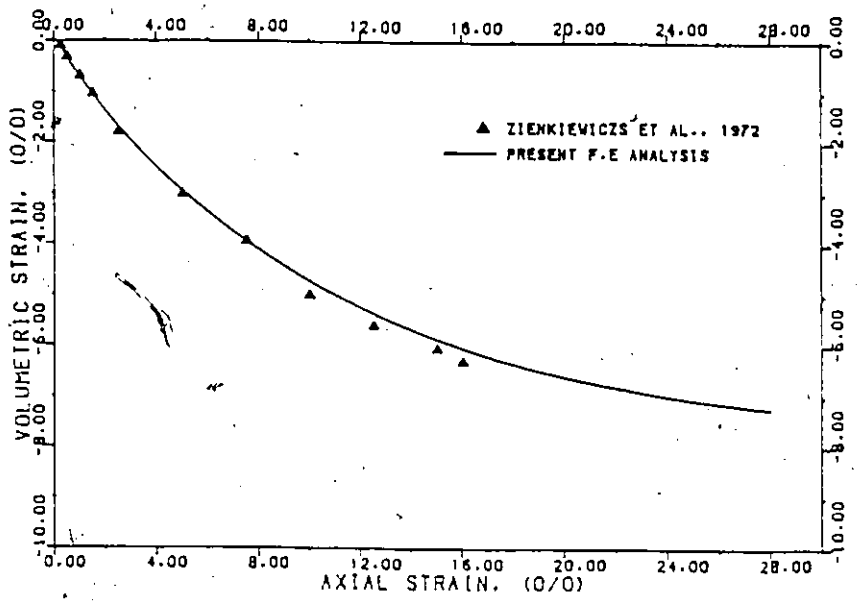


Fig. 5.20 Simulation of a triaxial test: Burland model (comparison with previous numerical results)

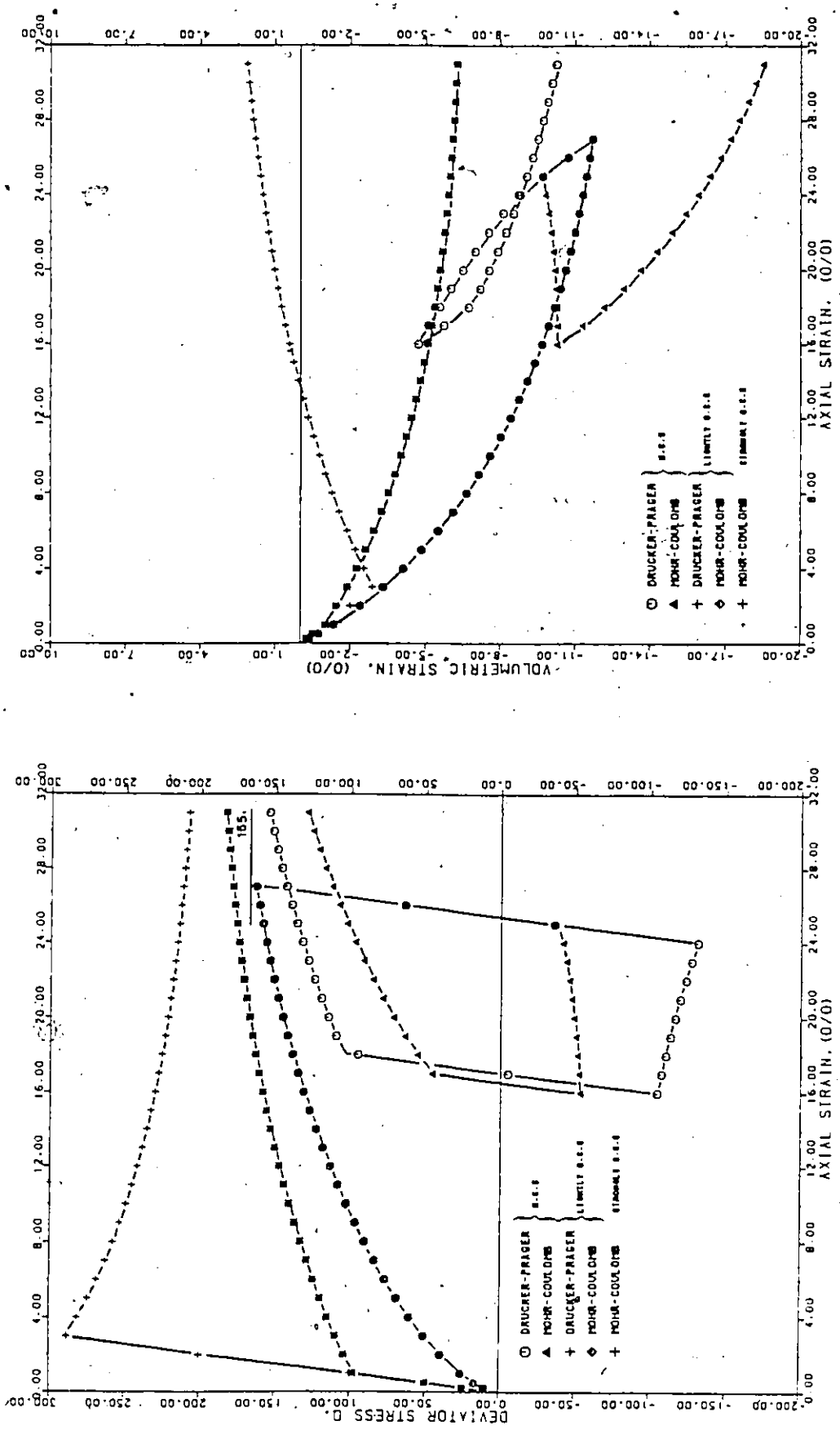


Fig. 5.21 Simulation of a traxial test: Critical Mohr-Coulomb and Drucker-Prager materials (NCS: p_e=100; slightly OCS: p_e=200; strongly OCS: p_e=600)

5.9.3 Influence of the type of algorithm

The numerical results were quite dependent on the algorithm adopted (TSM or ISM) as well as on the convergence tolerance.

With reference to Fig. 5.22 the ISM normally gives a higher response when compared with the TSM. Nonetheless, under reduced convergence tolerance ε, the former tends to the TSM with much more computational effort. These facts were revealed irrespective of imposing stress or strain to the structure.

However one shortcoming of the appealing TSM is that it cannot cope with cases involving strain softening as reported in Sect. 5.7.1.

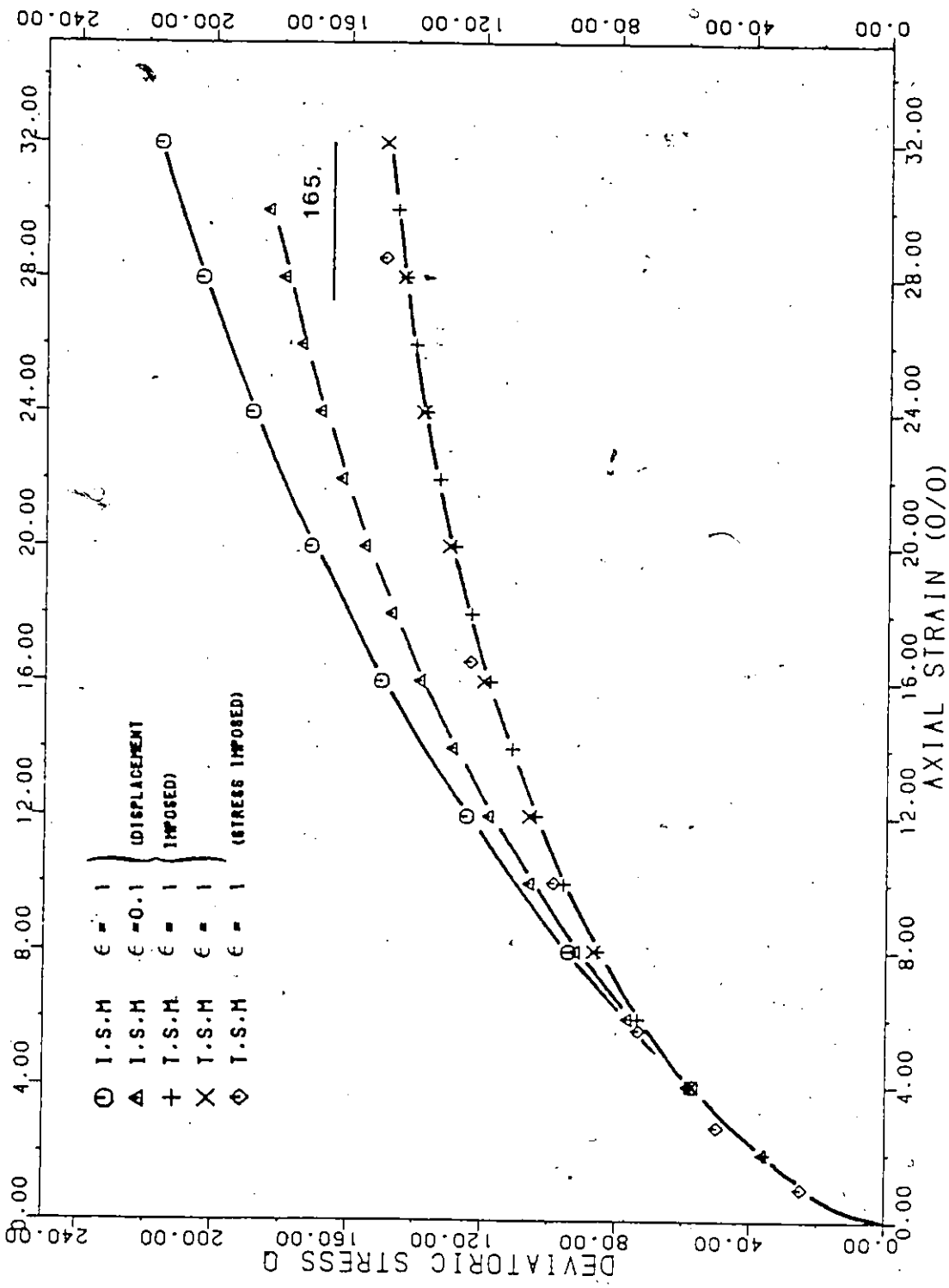


Fig. 5.22 Simulation of a triaxial test: Critical Mohr-Coulomb material (comparison between different algorithms)

5.9.4 Semi-infinite medium

The analysis refers to the boundary value problem of a uniform smooth and flexible strip load resting on a semi-infinite medium. Fig. 5.23 illustrates the mesh layout with the boundary and loading conditions while the material properties can be found in Table 5.9 as previously mentioned.

Before proceeding to the solution of the problem in elasto-plasticity, the equivalent linear elastic case was first tackled. At the same time the suitability of the mesh as well as the efficiency of the "smoothing" technique were investigated by confronting computed results with analytical values given by Poulos and Davis (1974).

The corresponding displacement and stress fields are illustrated in Figs. 5.24a,b respectively. With reference to Figs. 5.24c,e the computed stresses at the Gauss integration points agree closely with the analytical ones; so are the "smoothed" stresses at the nodes, Figs. 5.24d,f. It has to be emphasized that the correctness of the calculated nodal stresses depends on the degree of accuracy the Gauss integration point values are determined.

The transition to the more realistic non-linear analysis was then done in the case of a weightless material. The computed ultimate loads found in Tresca and Mohr-Coulomb failure criteria, Figs. 5.25 and 5.26 coincide with the theoretical values given by the Limit Analysis (Chen, 1975).

On the other hand the ultimate loads obtained by assuming von Mises and Drucker-Prager failure criteria were found to be higher than the Tresca and Mohr-Coulomb counterparts. This can be explained in the light of the previous results obtained in the simple one-element mesh tests.

The next step was to include the body weight in the form of initial in-situ stresses under isotropic conditions ($\sigma_x = \sigma_y = \sigma_z = \tau_{xy} = 0.0$). The unit weight was taken to be 20. KN/m3. As from Figs. 5.27 and 5.28, the ultimate loads are found to be substantially higher than those obtained in the absence of body weight, but these values still do not reach the theoretical Limit Analysis value of 376 KPa. for the Mohr-Coulomb failure criterion. Finally the comparison of the two failure criteria in Fig. 5.29 confirm the generally higher ultimate load predicted by the Drucker-Prager failure criterion.

Numerical results for the general boundary value problem were confronted with those published by Siriwardane and Desai, 1983. Figure 5.30a refers to the load-displacement curve obtained by the authors while Fig. 5.30b illustrates the present analysis.

Turning to the generalized Burland model, the initial in-situ stresses were set up in similar way as before, while the Critical State parameters were defined in a normally consolidated state ($p_e = p_o$). In this case, care had to be taken in the choice of algorithm. Both the TSM and ISM were

adopted and the analysis showed that the TSM could not adequately follow the load-displacement history of the material since some points in the medium undergo "softening" thus ill-conditioning the solution process. Figs. 5.31 and 5.32 show the model responses for "Critical" Mohr-Coulomb and Drucker-Prager materials respectively. Notice that the ISM curve tends towards the TSM curve in both cases when imposing a restrictive convergence tolerance ε.

It was found that the generalized Burland model predicts a much lower ultimate load for the same mobilized displacements compared with the linear elastic perfectly plastic models. The difference in failure mode is thought to be responsible for this phenomenon. With reference to Figs. 5.33 and 5.34 most of the displacements are mobilized beneath the foundation for the generalized Burland model while the conventional linear elastic perfectly plastic model predicts a more uniform spreading of displacement in the medium. Due to the concentration of displacements in a particular region of the medium the generalized Burland model may account for a "localized" mode of failure. The linear elastic perfectly plastic model, on the other hand, predicts a "generalized" mode of failure.

Finally to conclude, it is worth mentioning that the processing of the generalized Burland model required much more computational time than the conventional linear elastic perfectly plastic model (867 CPU compared with 264 CPU for a 63-element mesh made up of 222 nodes).

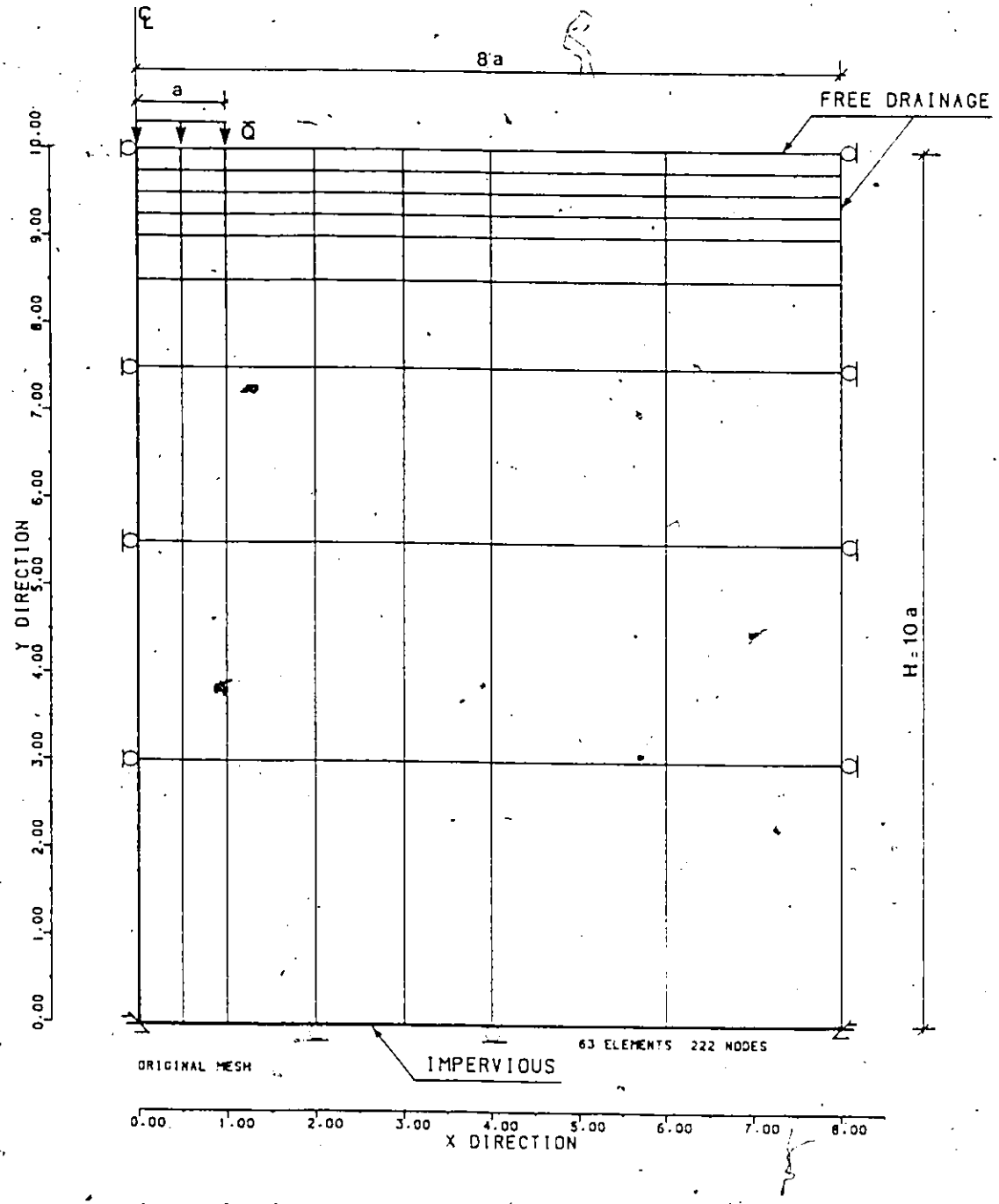
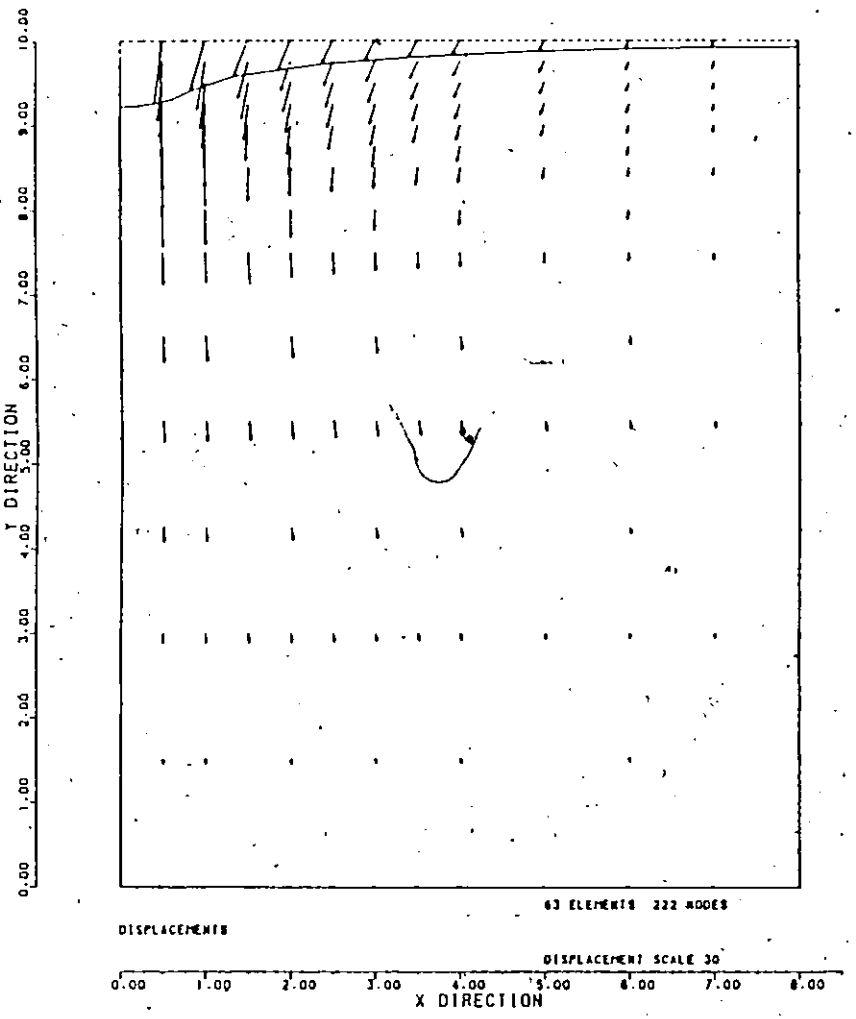
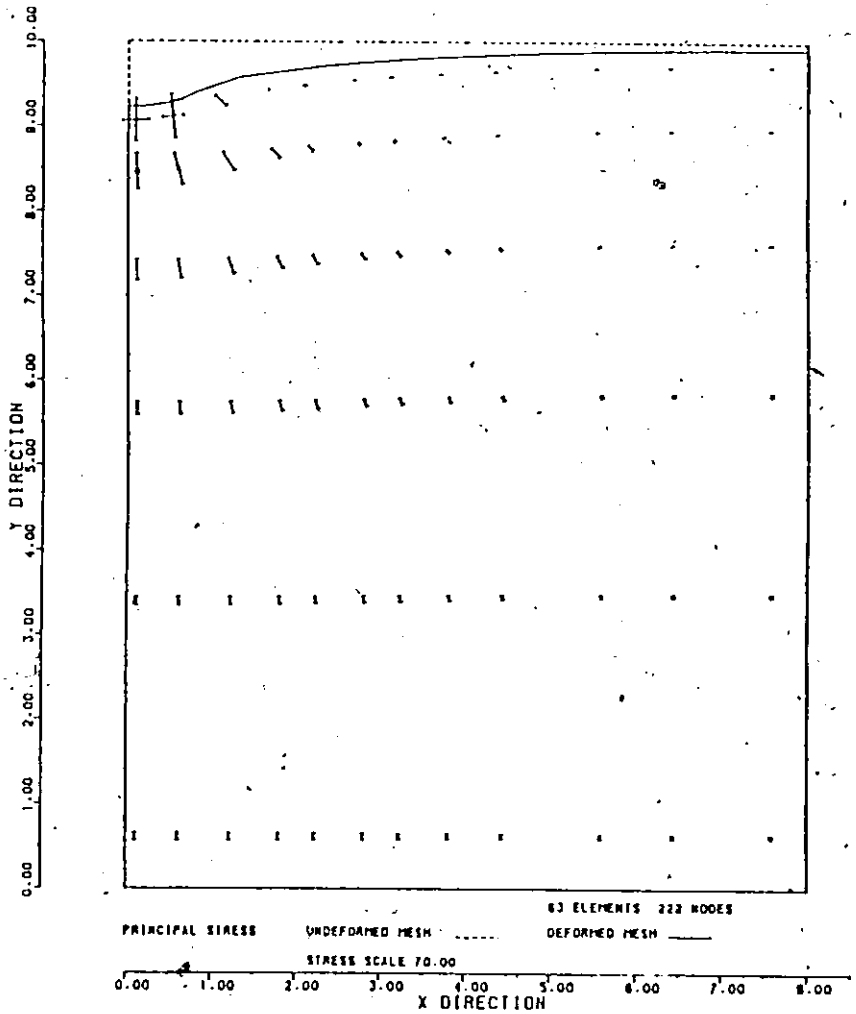


Fig. 5.23 Mesh layout for semi-infinite medium

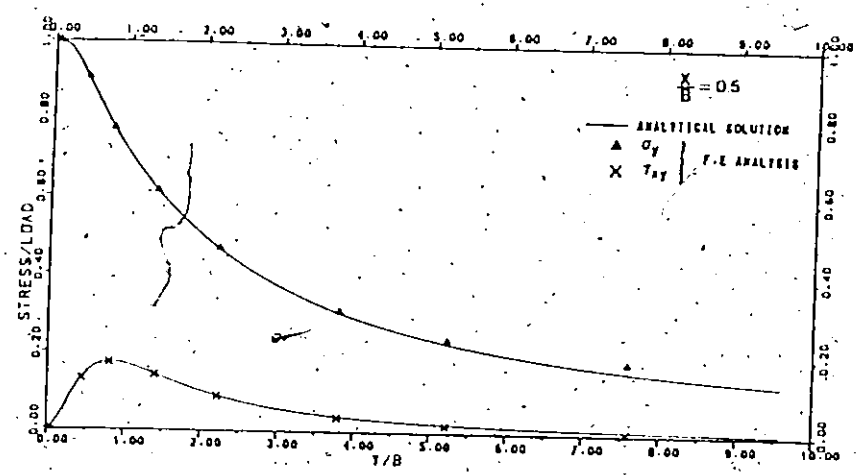


(a) displacement field

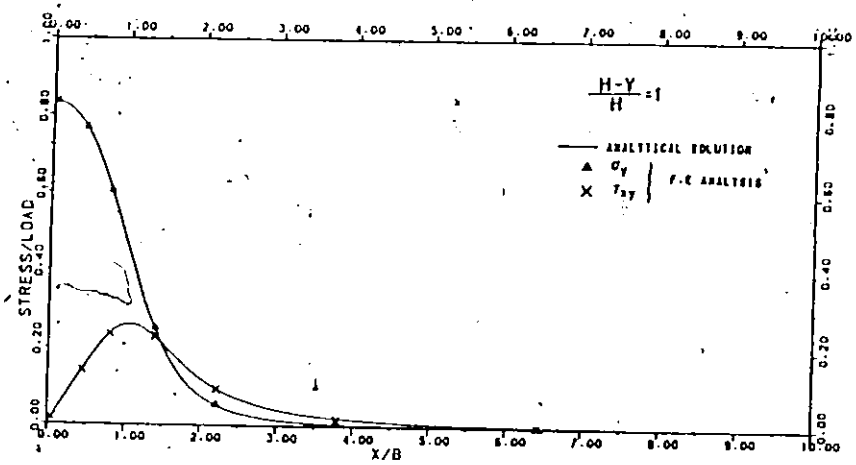


(b) principal stress field

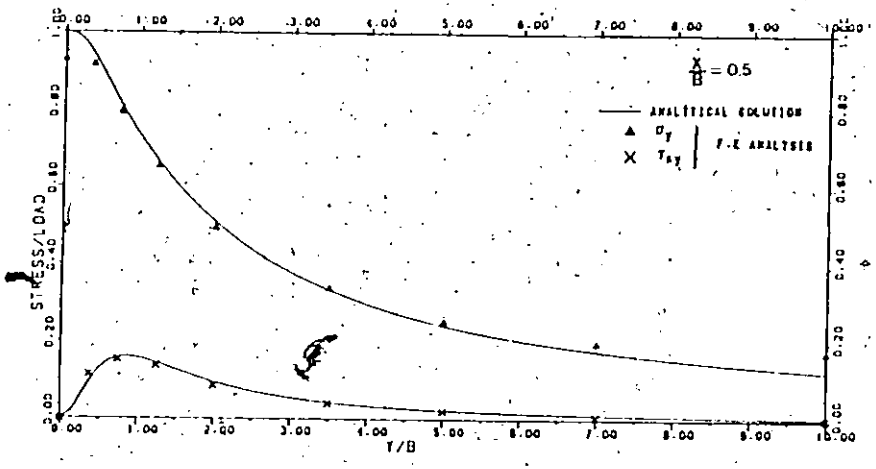
Fig. 5.24 Semi-infinite elastic medium



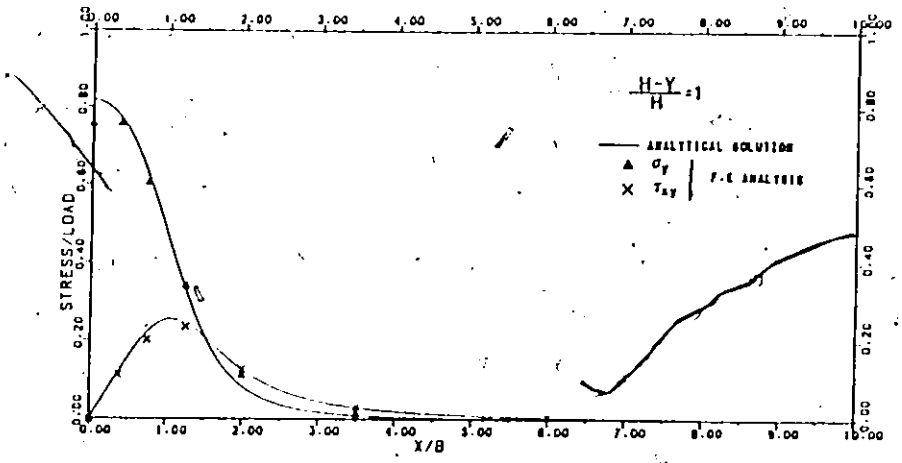
(c) comparison of stresses at Gauss points with analytical solution (vert. section)



(e) comparison of stresses at Gauss points with analytical solution (horiz. section)



(d) comparison of stresses at nodal points with analytical solution (vert. section)



(f) comparison of stresses at nodal points with analytical solution (horiz. section)

Fig. 5.24 Semi-infinite elastic medium

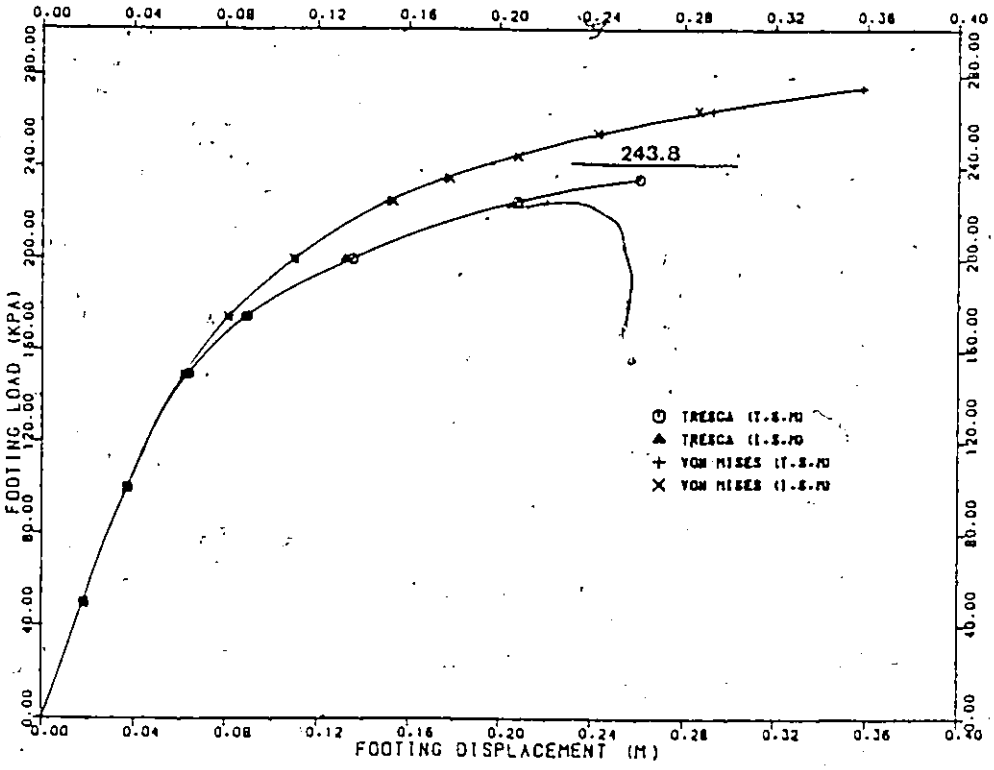


Fig. 5.25 Semi-infinite plastic medium: load displacement curves for weightless Tresca and von Mises materials

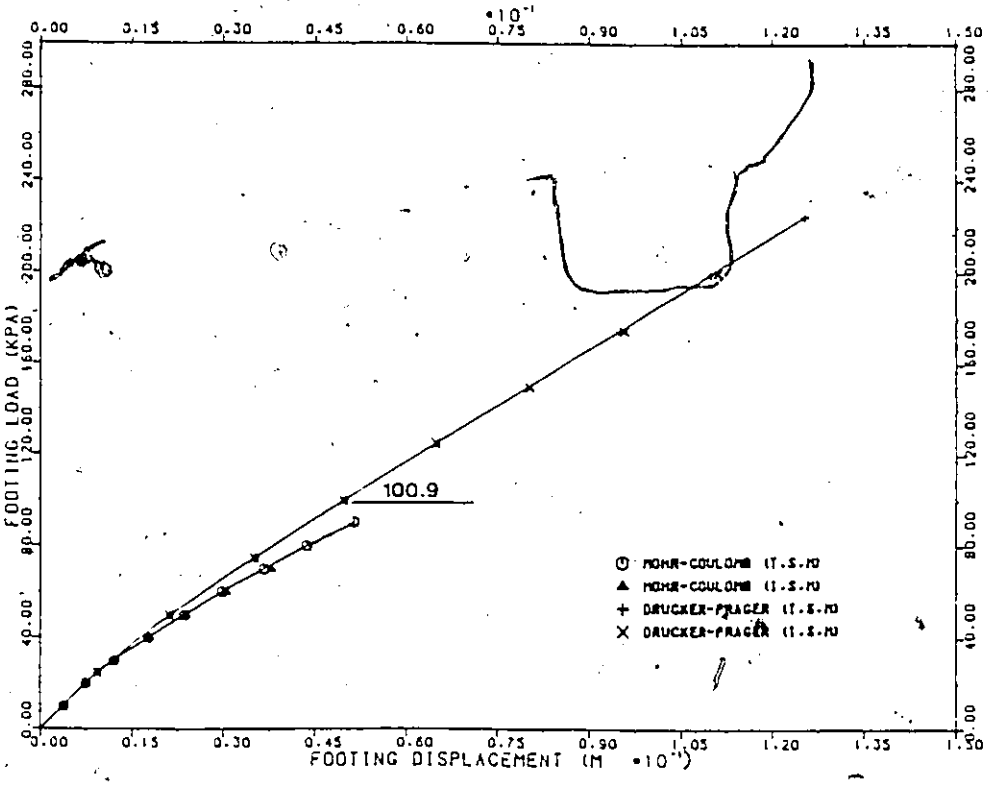


Fig. 5.26 Semi-infinite plastic medium: load displacement curves for weightless Mohr-Coulomb and Drucker-Prager materials.

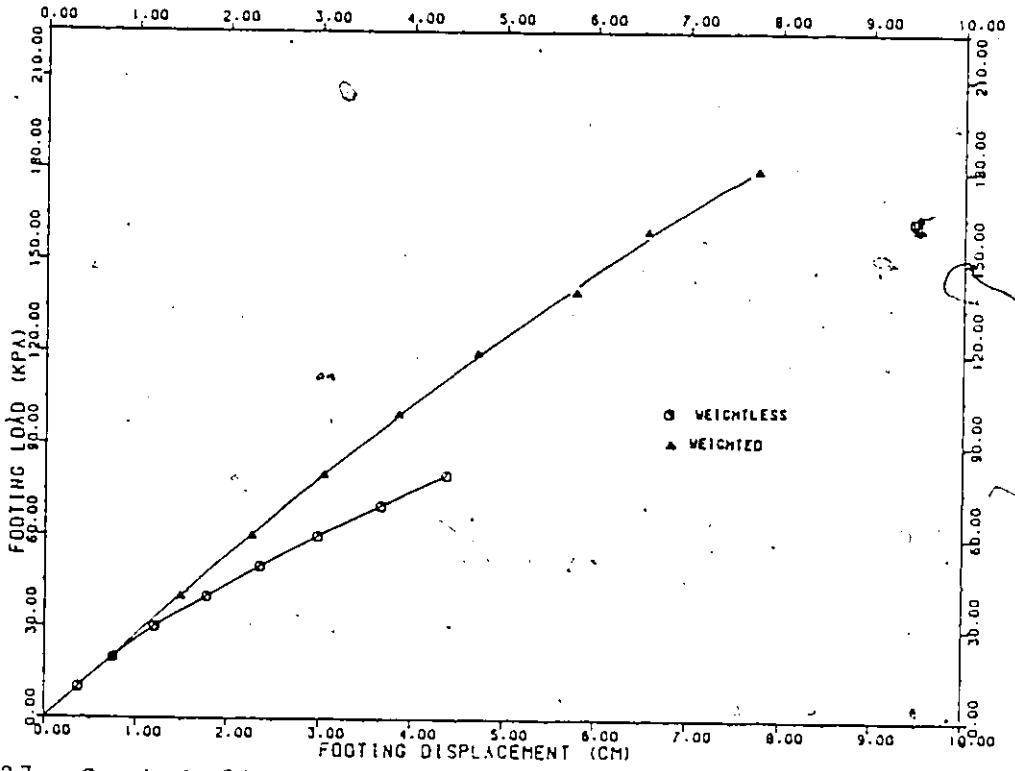


Fig. 5.27 Semi-infinite plastic medium: load displacement curves for weighted and weightless Mohr-Coulomb materials

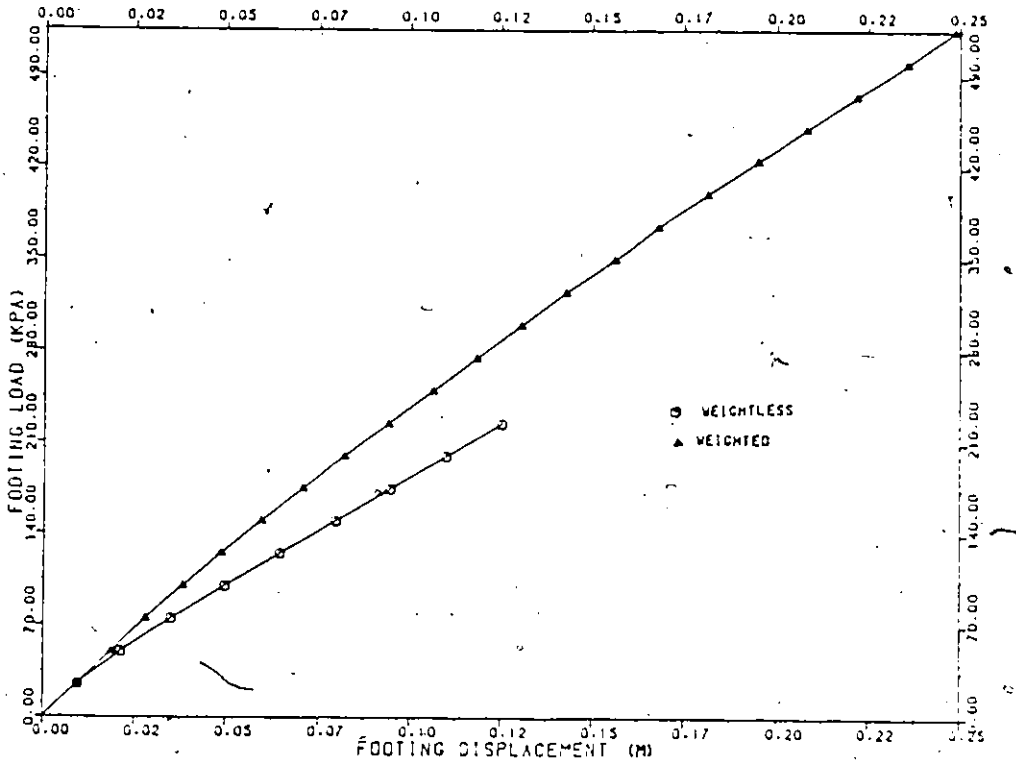


Fig. 5.28 Semi-infinite plastic medium: load displacement curves for weighted and weightless Drucker-Prager materials

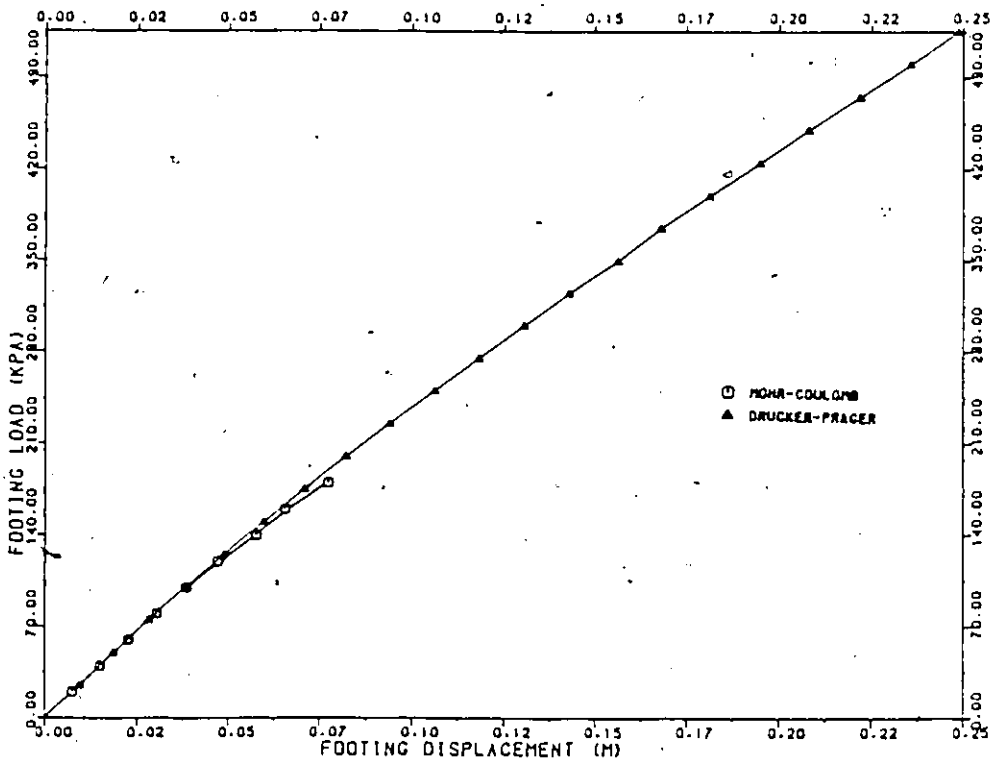
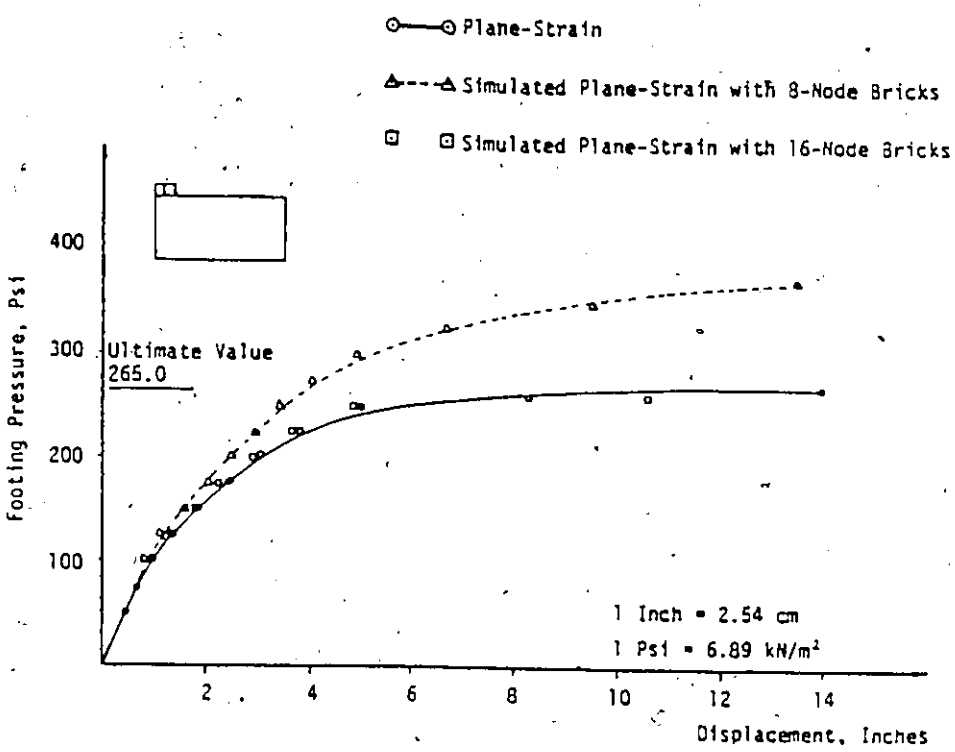
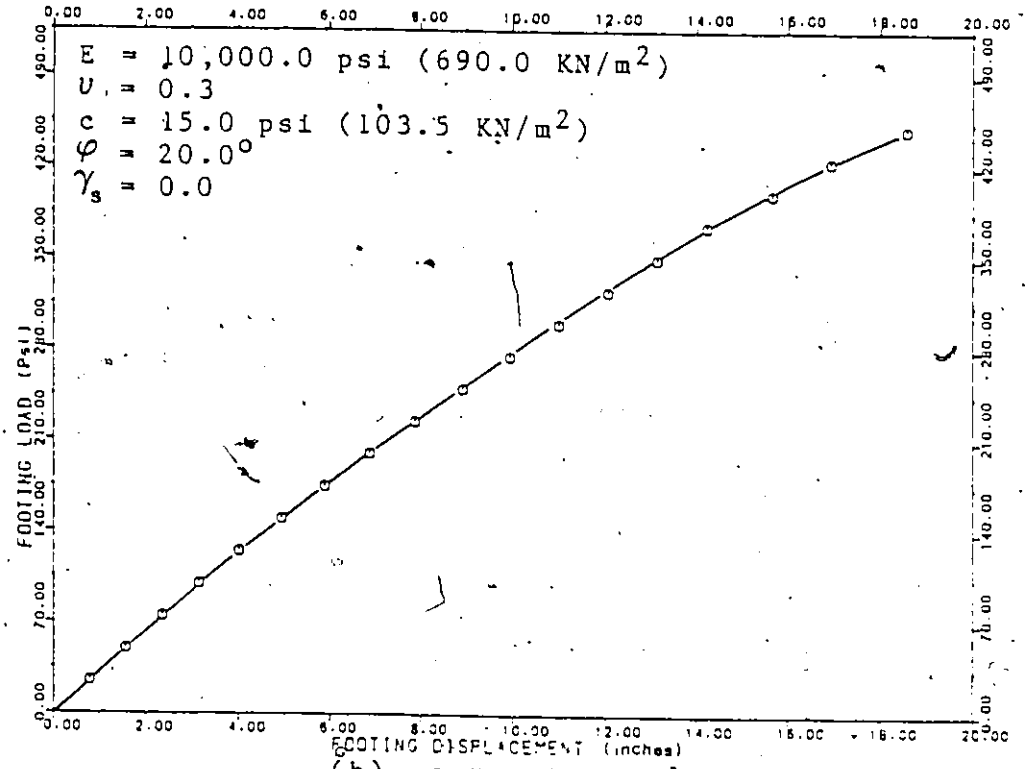


Fig. 5.29 Semi-infinite plastic medium: load displacement curves for weighted Mohr-Coulomb and Drucker-Prager materials



(a) results by Siriwardane and Desai, 1983



(b) current results

Fig. 5.30 Semi-infinite plastic medium: load displacement curves for Drucker-Prager material (comparison with previous results)

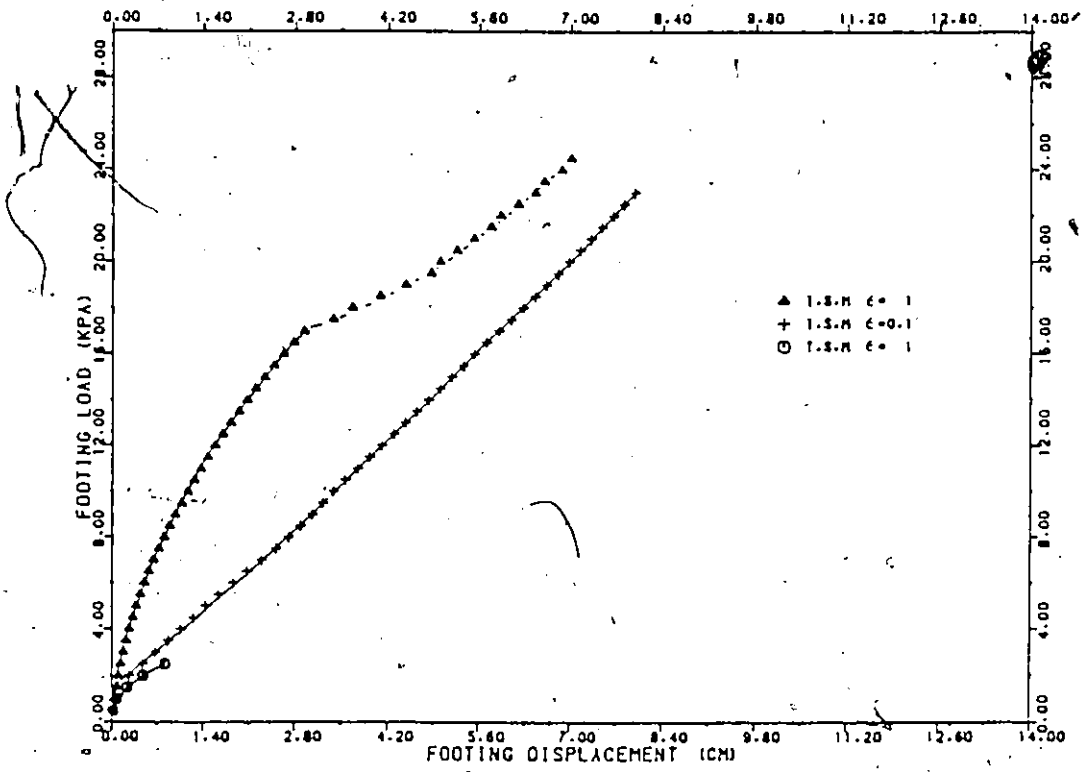


Fig. 5.31 Semi-infinite plastic medium: load displacement curves for Critical Mohr-Coulomb material

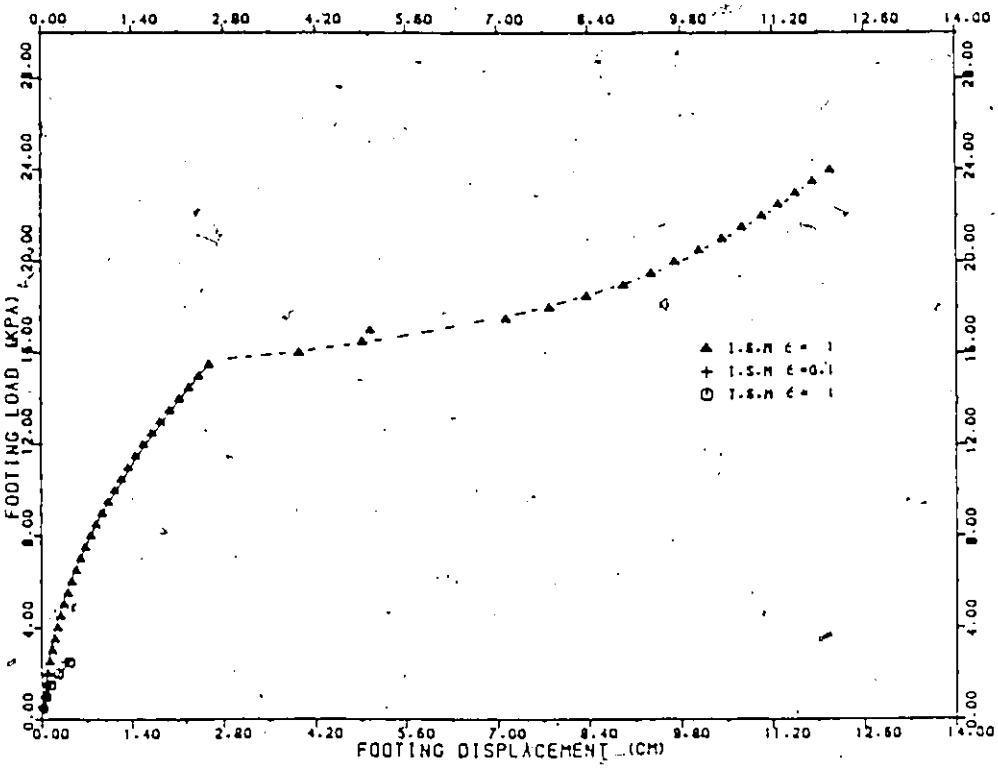


Fig. 5.32 Semi-infinite plastic medium: load displacement curves for Critical Drucker-Prager material

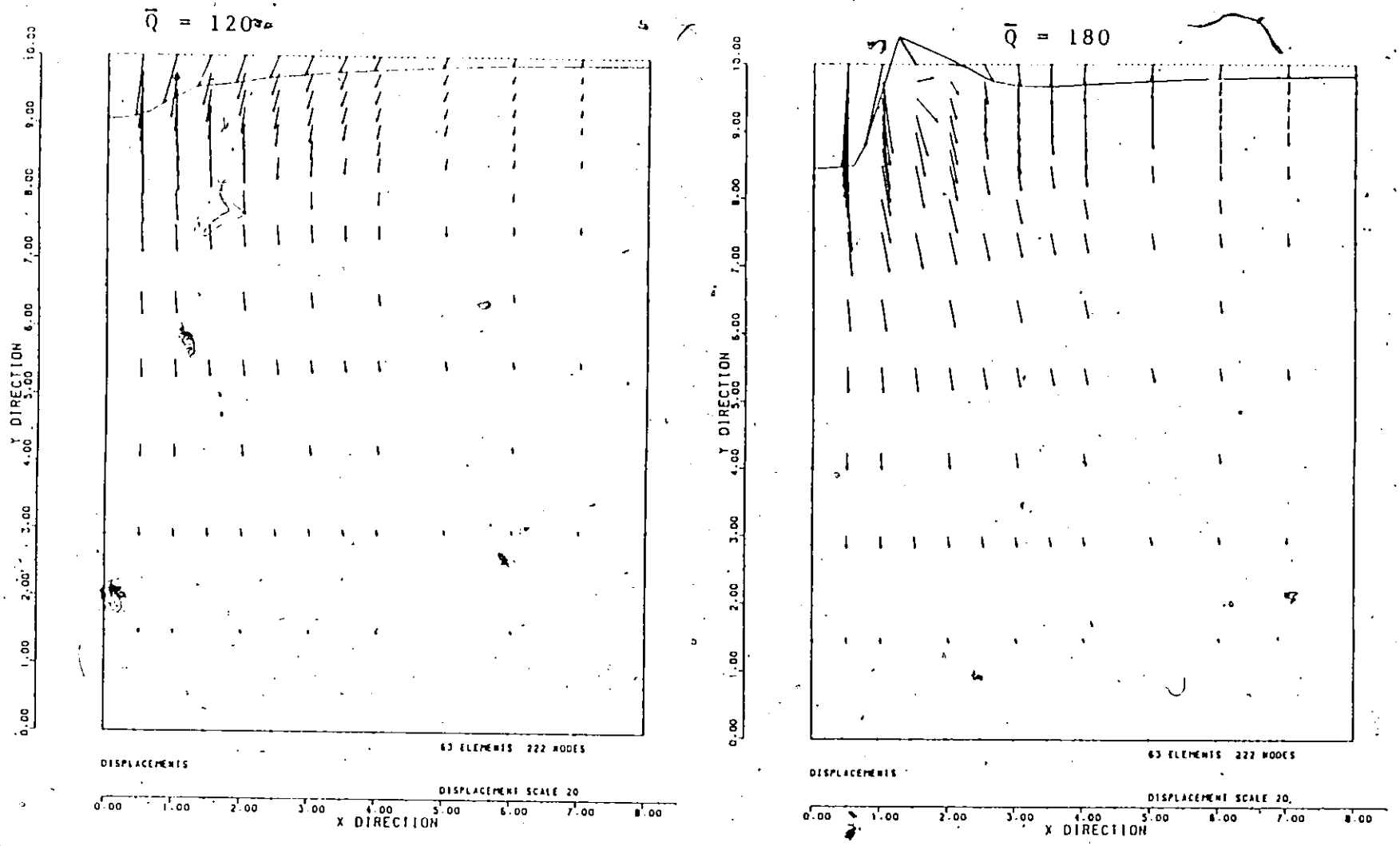


Fig. 5.33 Semi-infinite plastic medium: displacement fields for weighted Mohr-Coulomb material

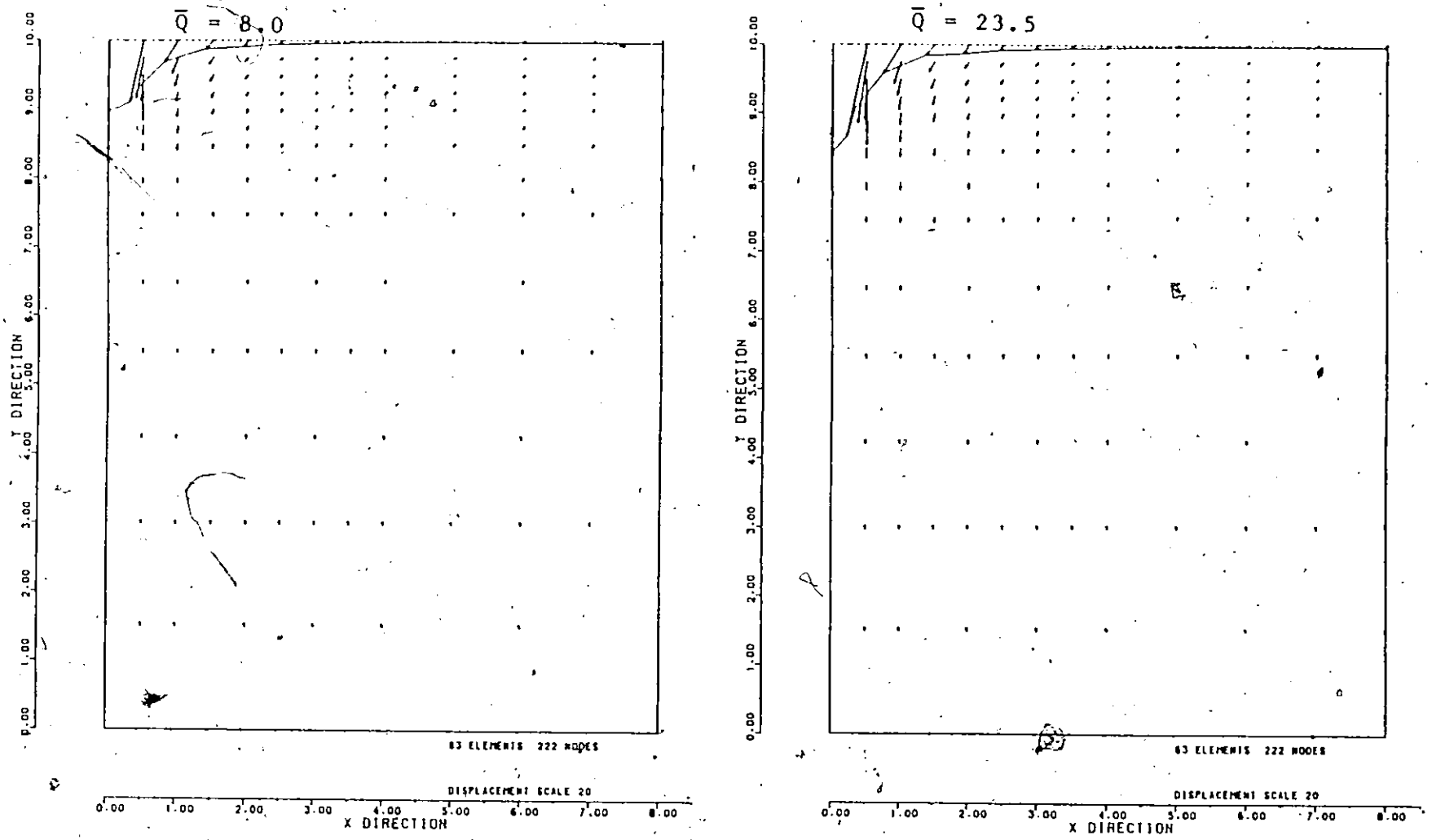


Fig. 5.34 Semi-infinite plastic medium: displacement fields for weighted Critical Mohr-Coulomb material

5.10 CONCLUSION

The four failure criteria used in the elastic perfectly plastic model produced excellent results in triaxial conditions. Thorough agreement with the theoretical expectation was obtained. On the other hand, under plane strain conditions equally good results were achieved with regards to the Limit Analysis values. It was found that the Limit Analysis rather corresponds to a Tresca or Mohr-Coulomb type of material rather than a von Mises or Drucker-Prager type which give unrealistically high ultimate loads. The independance of the limit deviatoric stress q on the angle θ in von Mises and Drucker-Prager failure criteria may account for this fact. Also it is at any rate a finding of importance that the post-yield responses of von Mises and Drucker-Prager materials are fundamentally different in plane strain conditions.

On the other hand, the plasticity model based on the Critical State concept were in close agreement with previous results in the triaxial condition. It was also possible to point out that the numerical results given by this model highly depended on the chosen algorithm (TSM or ISM). The TSM generally failed to adequately follow the load-displacement history in a strain softening material. This shortcoming can be remedied by using the ISM with a restrictive convergence tolerance of 0.1. It is obvious that much more computational effort has to be put in but in

return the complete load-displacement history can be at least followed. In that sense it is thought that room may be available for the development of a much more efficient algorithm such as the sophisticated acceleration methods.

Turning to the boundary value problem, a fundamentally different failure mode was observed based on the results in this study. The generalized Burland model gave a concentrated mobilization of displacements in a region below the foundation. As a result a much lower ultimate load was obtained in comparison with the linear elastic perfectly plastic model which spread uniformly the displacements throughout the medium. The Mohr-Coulomb limiting surface was used in both cases.

However, it is found that the generalized Burland model works better in axisymmetric situations since it has been originally developed assuming these particular conditions. As concern the plane strain and the boundary value problem further research may be needed for final conclusion on the suitability of the model through experimental investigations.

Chapter VI

COUPLED ELASTO-PLASTIC CONSOLIDATION ANALYSIS

6.1 INTRODUCTION

The first F.E. formulation was published by Sandhu and Wilson in 1969. On the basis of the works of Gurtin (1964), and Nickell and Sackmann (1960), the authors postulated a variational formulation of the coupled problem in terms of displacement and pore pressure.

Yokoo et al. (1971), Hwang et al. (1971), and Valliappan and Lee (1974) followed the works of Sandhu but used different space and temporal discretizations, while Ghaboussi and Wilson (1973) extended the study to compressible fluids.

Hwang et al. (1972), instead, were able to derive the F.E. formulation by applying directly Galerkin's method of weighted residuals.

During recent years, several attempts were done to extend the analysis to the non-linear behaviour of the soil skeleton and fluid flow in view of obtaining realistic answers.

Some investigators have considered non-linear elasticity (Lewis et al. 1976) or plasticity (Siriwardane and Desai 1981, Belkeziz and Magnan 1982, Chang and Duncan 1983), but yet some flaws remain to be solved.

Reed in 1984, pointed out the numerical errors that inevitably arise in the coupled consolidation analysis, even under the simple assumption of linear elasticity, if special precautions are not taken. Consequently, he proposed a "smoothing" technique which leads to more accurate calculation of the field variables. In addition, as discussed in the previous chapter, the analysis of elasto-plastic models is not well established.

This chapter is therefore devoted to the development of a coupled solid-fluid analysis F.E. computer code, TEPSA (Total Elasto-Plastic Stress Analysis). The program, on one hand, takes full advantage of ESEPA in which the elasto-plastic models have been proven-tested and, on the other hand, employs a "smoothing" technique to cope with the incoming numerical difficulties. The underlying numerical formulation is based on the works of Fusco (1985) and is herein reported in the next sections.

6.2 BASIC FINITE ELEMENT EQUATIONS

Thus far in previous chapters, analyses of solids and fluids have been considered separately. As regards the solid phase, the numerical formulation of the principle of virtual work, Eq. 5.18, was found to take the form of:

$$\int_{\Omega} B^T \, \Delta\underset{\sim}{\sigma}'(t) \, d\Omega - C \, \Delta\underset{\sim}{\pi}^e(t) = \Delta\underset{\sim}{f}'(t) \qquad (6.1)$$

The use of a constitutive law for the soil skeleton, written in incremental form as in Eq. 5.4a, therefore leads to the F.E. equation:

$$K^{ep}\Delta\underset{\sim}{d}(t) + C\,\Delta\underset{\sim}{\pi}^{e}(t) = \Delta\underset{\sim}{f}'(t) \qquad (6.2)$$

A quick glance at Eq. 6.2 reveals that there are two principal unknowns to be determined; namely the incremental displacement $\Delta\underset{\sim}{d}(t)$ and excess pore pressure $\underset{\sim}{\pi}^{e}(t)$. The additional set of equations can be found, following Biot's suggestion, from the continuity equation for saturated media:

$$\text{div } K \text{ grad } \frac{\pi^{e}}{\gamma_f} = \frac{d\varepsilon_v}{dt} \qquad (6.3)$$

where $d\varepsilon_v/dt$ is the rate of change of volume.

Consistently with the previous chapters, the excess pore pressure π^{e} and displacement d are then respectively discretized by two different shape functions S and $\bar{S}$ according to Eqs. 5.15a and 5.16c.

The use of Galérkin's method of weighted residuals yields the following F.E. equation:

$$-C^T \dot{\underset{\sim}{d}}(t) - \frac{P}{\gamma_f} \underset{\sim}{\pi}^e(t) = g(t) \qquad (6.4)$$

where P and g are respectively the hydraulic 'stiffness' matrix and the flow vector defined in Eqs. 3.2b and 3.2c.

Similar to TERCON, the time domain is discretized according to the θ-Wilson scheme resulting in:

$$\underset{\sim}{\pi}^e = \theta\, \underset{\sim}{\pi}^e_n + (1-\theta)\underset{\sim}{\pi}^e_{n-1} = \theta\, \Delta\underset{\sim}{\pi}^e_n + \underset{\sim}{\pi}^e_{n-1} \qquad (6.5)$$

$$\dot{\underset{\sim}{d}} = \frac{\Delta \underset{\sim}{d}_n}{\Delta t} \qquad (6.6)$$

$$-C^T\, \Delta \underset{\sim}{d}_n - \alpha\, P\, \Delta \underset{\sim}{\pi}^e_n = \Delta g'_n \qquad (6.7)$$

where the subscripts n-1 and n correspond to the previous and current times respectively.

Subsequent replacement of Eqs. 6.5, 6.6 and 6.7 into 6.4 leads to the following F.E. equation.

(6.8a)

where

$$\alpha = \frac{\theta \Delta t}{\gamma_f} \tag{6.8b}$$

$$\underset{\sim}{\Delta g}'_n = (\theta \, \Delta \underset{\sim}{g}_n + \underset{\sim}{g}_{n-1}) + \frac{\Delta t}{\gamma_f} P \, \underset{\sim}{\pi}^e_{n-1} \tag{6.8c}$$

Finally, the coupling of the solid phase, Eq. 6.2, with that of the fluid phase, Eq. 6.8, results in a composite matrix equation which has to be solved at each time step.

$$\begin{bmatrix} K^{ep} & -C \\ -C^T & -\alpha P \end{bmatrix} \begin{Bmatrix} \Delta \underset{\sim}{d}_n \\ \Delta \underset{\sim}{\pi}^e_n \end{Bmatrix} = \begin{Bmatrix} \Delta \underset{\sim}{f}_n \\ \Delta \underset{\sim}{g}'_n \end{Bmatrix} \tag{6.9}$$

6.3 NUMERICAL ERRORS

Past experience has revealed considerable numerical difficulties when implementing the coupled consolidation theory. This arises due to the discretization in space and time of two field variables described by functions of different orders. Consequently the choice of the type of element and time integration scheme are of prime importance for the correct solution of the problem.

6.3.1 Type of Element, Spatial discretization

According to the theory of small deformation, stress is proportional to the derivative of displacement. This requires the stress function to be one order lower than the displacement function.

This theoretical consideration probably led Sandhu (1968) to approximate the displacement d by a parabolic shape function and excess pore pressure π^e by a linear one. As a result, he introduced a "composite" triangular element denoted by 63, having displacement degrees of freedom at all 6 nodes but pore pressure degree of freedom only at the 3 corner nodes.

Another method consists of representing displacement in terms of an 8-node quadrilateral and pore pressure at the 4 corner nodes thereby the notation 84.

Sandhu et al. (1977) undertook an interesting analysis on the performance of these types of elements by comparing "composite" elements (types 63 and 84, Fig. 6.1b) with "standard" ones (types 66 and 88, Fig. 6.1a) on the basis of analytical results for 1-D elastic consolidation.

It was found that "standard" elements give better estimates of settlements but are prone to oscillatory errors of pore pressures in space. In return composite elements produce errors in pore pressures which tend to vanish in time.

Those types of errors led Cividini et al. (1982, 1983) to develop the so-called compatible element. Though they were able to obtain interesting results, this new type of element has the drawback of respecting compatibility of the displacement field only in an average standpoint.

Moreover, an alternative and very appealing approach was proposed by Reed (1982, 1984) who applied the so-called "smoothing" technique. The basic premise of Reed's analysis is that the pore pressure distribution produced by an 8-node isoparametric element gives the exact solution at the 4 Gauss points. Consequently the correct pore pressures at the nodes may be extrapolated from a plane defined by a bi-quadratic interpolating function satisfying all four exact Gauss point pore pressure values.

The smoothed (correct) pore pressures at the nodes are expressed in terms of the initial (inexact) ones as shown in Eqs. 6.10a and 6.10b. In this way accuracy may be simultaneously achieved for pore pressure and displacement so that the adaptation of the "smoothing" technique forms the basis of the F.E. computer code developed in this research.

$$\underset{\sim}{\pi}^{e*} = M \, \underset{\sim}{\pi}^{e} \qquad (6.10a)$$

where

$$M = \frac{1}{6}\begin{bmatrix} 2 & 4 & -2 & 0 & 0 & 0 & -2 & 4 \\ 0 & 4 & 0 & 2 & -1 & 0 & -1 & 2 \\ -2 & 4 & 2 & 4 & -2 & 0 & 0 & 0 \\ -1 & 2 & 0 & 4 & 0 & 2 & -1 & 0 \\ 0 & 0 & -2 & 4 & 2 & 4 & -2 & 0 \\ -1 & 0 & -1 & 2 & 0 & 4 & 0 & 2 \\ -2 & 0 & 0 & 0 & -2 & 4 & 2 & 4 \\ 0 & 2 & -1 & 0 & -1 & 2 & 0 & 4 \end{bmatrix} \qquad (6.10b)$$

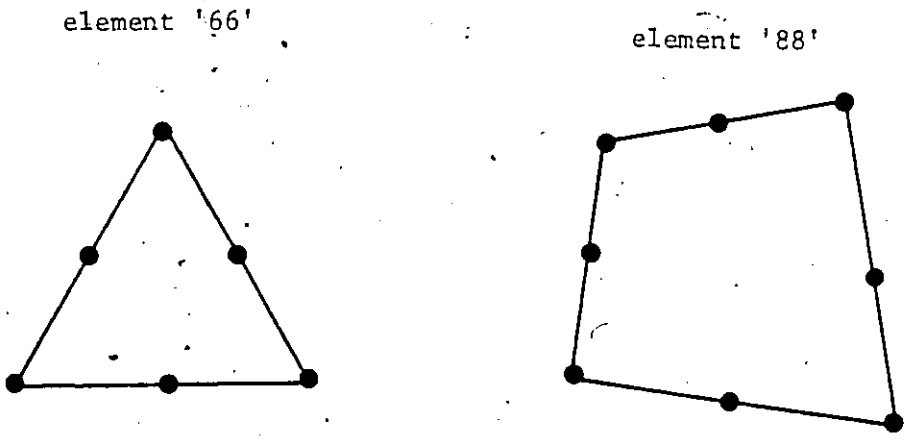


Fig. 6.1a Representation of "standard" elements

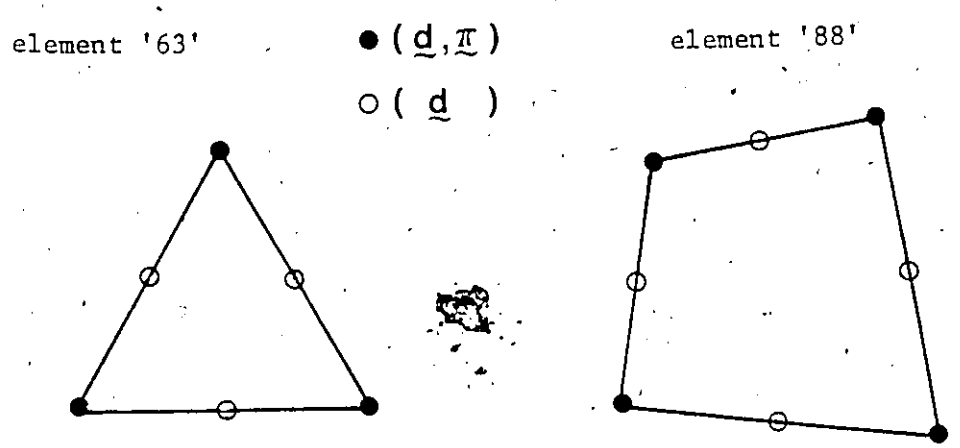


Fig. 6.1b Representation of "composite" elements

6.3.2 Time integration

Another determinant factor in the achievement of accurate results is the proper choice of time domain discretization.

Alternative ways to define the θ-values which govern the time discretization scheme were previously proposed by numerous authors, but their performances are still subject to investigation.

Sandhu et al. (1977) proposed to use a logarithmic variation of displacement with time, which apparently gave good results. A comprehensive table of values of θ according to different schemes was given in Table 2.2.

Moreover, Sandhu observed that the logarithmic interpolation scheme proposed by Hwang et al. 1971, had the drawback of being costly since the global matrix in Eq. 6.9 was evaluated at every time step. Instead, he proposed to divide the time domain into a number of constant intervals, so that the global matrix does not need to be recalculated within a sub-interval.

The present developed F.E. computer code contains, as in TERCON, the options of imposing the θ-value, or rather calculating it according to Sandhu's suggestion.

6.3.3 Ill-conditioning of the global stiffness matrix

The relative orders of magnitude of the elements of the elasto-plastic submatrix K^{ep} (mainly dependent on the Young's modulus E of the soil) and those of the hydraulic

"stiffness" matrix P (which has the soil permeability as a factor) may ill-condition the matrix preventing the solution of Eq. 6.9.

This discrepancy is eradicated by lumping the submatrices with a weighting factor w as proposed by Reed (1982), resulting in:

$$\begin{bmatrix} K^{ep} & -wC \\ -wC^T & -\alpha w^2 P \end{bmatrix} \begin{Bmatrix} \Delta u_{\sim n} \\ \dfrac{\Delta \pi_{\sim n}}{w} \end{Bmatrix} = \begin{Bmatrix} \Delta f'_{\sim n} \\ w \Delta g'_n \end{Bmatrix} \qquad (6.11)$$

and

$$\Delta f'_{\sim n} = \Delta f_{\sim n} + C \, \Delta \pi^e_{\sim n} \qquad (6.12)$$

6.3.4 Load history

It is a common practice, in engineering, to apply load to the soil gradually over a relatively short time t_o and thereafter maintain it constant. Fig. 6.2a shows the simplest form of such a loading which is initially applied steadily with time. More complex loading histories are met during the construction of embankments which proceeds in lifts (Fig. 6.2b). Clearly, dependent on the magnitudes of t_o and the permeability of the soil, a certain dissipation of excess pore pressure will have occurred by time t_o and

dissipation is of interest as well as the subsequent behaviour with time (Smith, 1976).

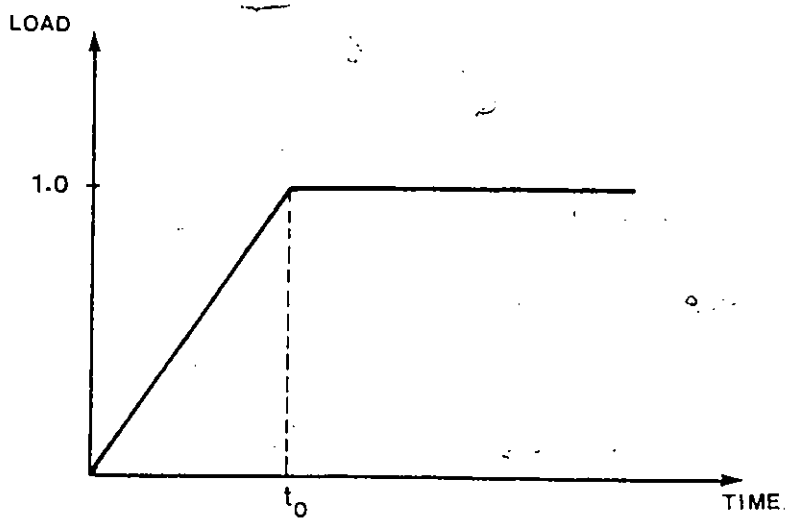


Fig. 6.2a Simple load history

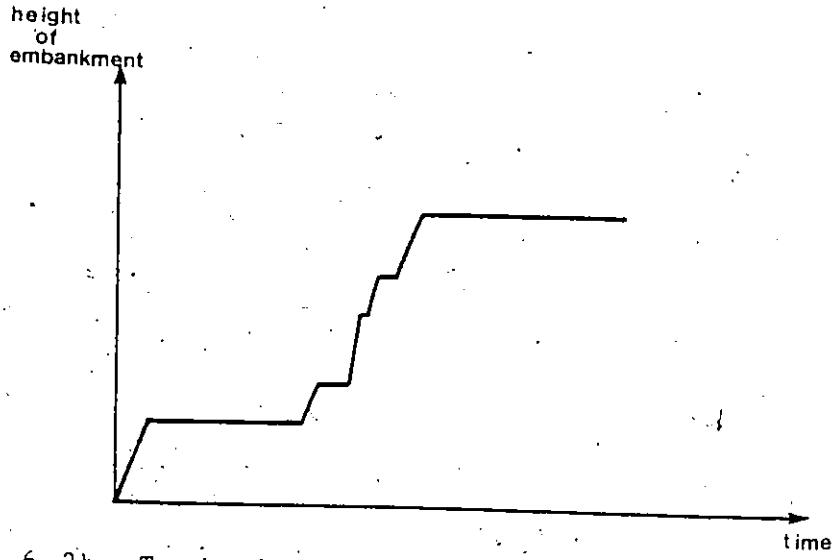


Fig. 6.2b - Typical load history during the construction of an embankment

6.4 TEPSA (TOTAL ELASTO-PLASTIC STRESS ANALYSIS)

The numerical implementation of the coupled consolidation theory is carried out in TEPSA by means of the basic formulation reported in Eq. 6.11, so as to eliminate the discrepancies caused by the difference in orders of magnitude between the constituent submatrices K^{ep} and P.

In order to achieve accuracy both in displacements and pore pressures Reed's "smoothing" technique, as previously presented, is herein employed. As regard constitutive relationships, TEPSA fully incorporates all the features of ESEPA for a correct elasto-plastic analysis. In addition, TEPSA permits the simulation of the loading history during construction.

Finally, computing difficulties are overcome by grouping all unknowns i.e. displacements d and pore pressures π^e for each node next to each other and subsequently rearranging the rows and columns of the coefficients of the matrix accordingly, Eqs. 6.13. This procedure advantageously suits the frontal solution subroutine used in the program, in view of achieving high efficiency.

6.4.1 Structure of TEPSA

The incorporation of all the previously mentioned features in TEPSA involves some adaptations from ESEPA and the computation sequence is illustrated in the flowchart in Fig. 6.3. The main program is made up of three loops nested on

loading increments, time marching steps and finally residual load balancing increments.

Since the permeability is considered constant, the evaluation of the matrices P and C, together with that of the load vectors $\underset{\sim}{f}$ and $\underset{\sim}{g}$ as established in Eqs. 5.17d and 3.2c are performed only once in Subroutine LOADPS.

Turning to the time marching scheme, Δt can be differently calculated according to the options in Subroutine TMARCH, as has been described in TERCON.

In Subroutine CLOAD the equivalent load $\Delta\underset{\sim}{g}'_n$ is calculated according to Eq. 6.8c and then assembled with $\Delta\underset{\sim}{f}_n$ to build up the right hand side of Eq. 6.11.

Subroutine ALGOR monitors the eventual change of time increment and the update of the stiffness matrix K^{ep}, thus deciding the reassembly of the composite stiffness matrix.

As in ESEPA, Subroutine STIFFP calculates the elasto-plastic stiffness matrix K^{ep}. Subsequent assembly of the submatrices C and P is done in Subroutine ASSEM, resulting in the composite stiffness matrix in Eq. 6.13a.

The determination of displacement $\Delta\underset{\sim}{d}_n$ and excess pore pressure $\Delta\underset{\sim}{\pi}^e_n$ during a time increment is done in two stages:

Firstly the system of equations 6.13a is solved subjected to the assigned fixity of the problem under consideration. The resulting excess pore pressures $\Delta\underset{\sim}{\pi}^e_n$ are then subsequently corrected in Subroutine SMOOTH according to Eqs. 6.10.

After having "smoothed" $\Delta\underset{\sim}{\pi}^e_n$, the displacements Δd_n compatible with the corrected excess pore pressures are then found. This is done by solving the system of equations 6.13a with the "smoothed" excess pore pressures as specified fixities.

Subroutine SPLIT separates displacements from excess pore pressures and equivalent flows from forces according to Eqs. 6.11 and 6.12. It also calculates the effective "equivalent" load which should be carried by the soil skeleton for the current value of excess pore pressure, namely:

$$\Delta\underset{\sim}{f}'_n = \Delta\underset{\sim}{f}_n + C\ \Delta\underset{\sim}{\pi}^e_n$$

The equivalent load carried by the soil skeleton for the current displacements is calculated similarly as in ESEPA by means of Subroutine RESIDU.

Subroutine CONVER1 calculates the residual load $\Delta\underset{\sim}{\psi}_i$ and eventually checks if an additional residual load balancing iteration is needed. CONVER2 monitors the degree of consolidation before passing on to the next time increment and ultimately to the next load increment.

Finally, the displacements, pore pressures, flows, reactions and effective stresses are printed in Subroutine OUTPUT.

The majority of the subroutines utilized are in common with the previous programs with slight modifications brought in, and only the additional Subroutines ALGOR, ASSEM, CLOAD, SMOOTH, SPLIT are described in details in Appendix D. Moreover, Subroutine FRONT was slightly adapted and the resulting changes are discussed in section D.1

$$\begin{bmatrix} k_{11} & k_{12} & \bar{C}_{11} & \cdots & \cdots & \cdots \\ k_{21} & k_{22} & \bar{C}_{21} & & & \\ \bar{C}_{11} & \bar{C}_{12} & \bar{P}_{11} & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \bar{P}_{kk} \end{bmatrix} \begin{Bmatrix} \Delta u_1 \\ \Delta v_1 \\ \Delta\bar{\pi}_1 \\ \cdot \\ \cdot \\ \cdot \end{Bmatrix} = \begin{Bmatrix} \Delta f_{x1} \\ \Delta f_{y1} \\ \Delta\bar{g}' \\ \cdot \\ \cdot \\ \cdot \end{Bmatrix} \qquad (6.13a)$$

where

$$\bar{C}_{ij} = -w\, C_{ij} \qquad (6.13b)$$

$$\bar{P}_{ij} = -\alpha w^2 P_{ij} \quad \text{and} \quad \overline{\Delta\pi}^e_j = \Delta\pi^e_j / w \qquad (6.13c)$$

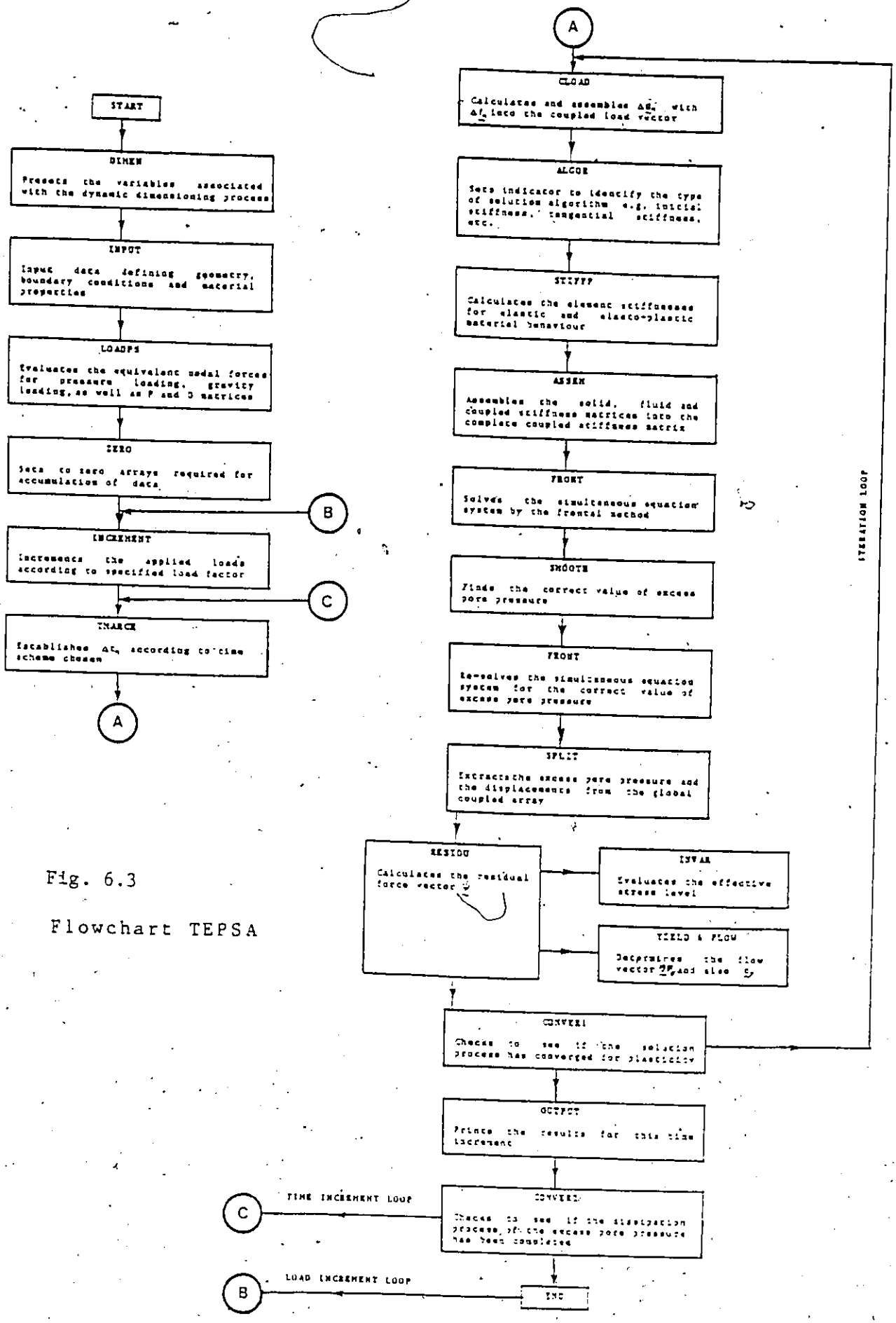


Fig. 6.3

Flowchart TEPSA

6.5 NUMERICAL EXAMPLES

In order to have a thorough grasp of both the theoretical and physical significances of the coupled consolidation theory, the numerical analysis covered a large range of examples; starting from the simple 1-D condition in elasticity and extending to the more complicated 2-D situation in elasto-plasticity. Once more the meshes relevant to this current analysis, the time marching scheme and the material properties were kept the same, with respect to the ones reported in previous chapters (see Fig. 4.2a,b,c; 5.23 and Table 5.9)

6.5.1 Coupled elastic consolidation

1-Dimensional conditions

TEPSA was tested in so far as analytical solutions were available from theory. In fact under 1-D conditions, the F.E.M analysis could be confronted with the closed form solution proposed by Taylor (1948), since coupled and uncoupled theories coincide. Figs. 6.4a,b show comparisons for the surface settlement evolution and excess pore pressure dissipation with time between the present analysis and Taylor's solution. It was found that both pore pressures and displacements were simultaneously accurate in contrast to the numerical oscillations cited by Sandhu et al. (1977). When the mesh was used under unchanged boundary conditions,

but with axisymmetric considerations in the calculations, the same numerical results were recovered since this case still exhibits 1-D features according to both the geometrical and hydraulic standpoints.

Axisymmetric conditions

For the conventional triaxial sample test which is allowed to deform laterally as shown in Fig. 6.5, two cases were studied always assuming an initial isotropic preconsolidation pressure of 100. In one case a uniform load $\bar{Q}$ = 135 was instantaneously applied and thereafter left constant. The other case comprised the imposition of a uniform surface displacement constant throughout the consolidation process. The imposed displacement was of the order of 0.018 equal to initial displacement obtained in the first case when a load was applied.

In the current situation, no analytical solution is available to check the numerical responses but it can be proved that the initial stresses and excess pore pressures can be determined by the following expressions:

$$\Delta p' = 0 \quad \therefore \quad \Delta p = -\pi^e \tag{6.14a}$$

$$\Delta \sigma_r = 0 \quad \therefore \quad \Delta \sigma'_r = \pi^e \tag{6.14b}$$

$$\pi^e = -\frac{\Delta\sigma'_z}{2} = -\frac{\Delta\sigma_z}{3} \qquad (6.14c)$$

where σ_z and σ_r are respectively the vertical and radial stresses.

The above simple relationships were straightforward derived from equilibrium considerations and from the fact that $\Delta p' = 0$ at t=0 (incompressible fluid). At infinite time, all the excess pore pressures are considered totally dissipated such that stresses can be very easily calculated.

Fig. 6.6a illustrates the displacement field evolution during consolidation in the case of applied load. It is interesting to notice that the sample initially (t=1 E-06) deforms laterally and with time the vertical surface retreats to its original position. In fact when ν =0.0 and under dry conditions no lateral deformations take place in elasticity. Fig. 6.6b shows the surface settlement evolution with time. The computed results shown in Figs. 6.6c to 6.6h, in so far as initial and ultimate values are concerned, completely agree with the theoretical values found in Table 6.1.

A well known feature of the coupled theory is that the excess pore pressure π^e continues to increase well before the dissipation process starts. Figs. 6.6c,d illustrate this phenomenon commonly known as the Mandel-Cryer effect. On the other hand, the uncoupled theory, as previously seen, predicts a monotonic decrease. In comparison with this

latter theory, the dissipation is faster in the coupled analysis since the completion of surface displacement is reached at t=60 against 300 in the former case.

Figs. 6.7 ,instead, show the sample response under an imposed surface displacement. It was again possible to theoretically determine and recover the stress values found in Table 6.1. Notice that the normalization of the parameters was done with respect to an initial total vertical stress of 135. As seen from Fig. 6.7c, it is not surprising to have a relatively constant effective vertical stress throughout the consolidation process since the surface displacements are kept constant. The resulting stress paths during consolidation are thereby shown in Figs. 6.6i and 6.7g.

Both analyses required only 156 CPU for 40-element mesh made up of 149 nodes.

2-Dimensional analysis. (smooth flexible footing resting on a semi-infinite layer)

The finite element mesh and drainage boundary conditions adopted in this example are shown in Fig. 4.2c; the strip load was assumed to be equal to 10.

In contrast to the uncoupled analysis, this present analysis allowed to follow the stress field evolution during consolidation. Figs. 6.8a,b,c,d show respectively the displacement field, the surface settlement evolution with

time, and the fluid velocity and stress fields. The dissipation pattern of the excess pore pressures are shown in Figs. 6.8e to 6.8g, from which it is possible to determine the initial excess pore pressure (undrained) distribution without any a priori assumption. The delayed development of the maximum excess pore pressure in Fig. 6.8e highlights once more the well-known Mandel-Cryer effect. It is noted that this was not detected in the uncoupled theory.

The total required CPU was 149 for the above analysis.

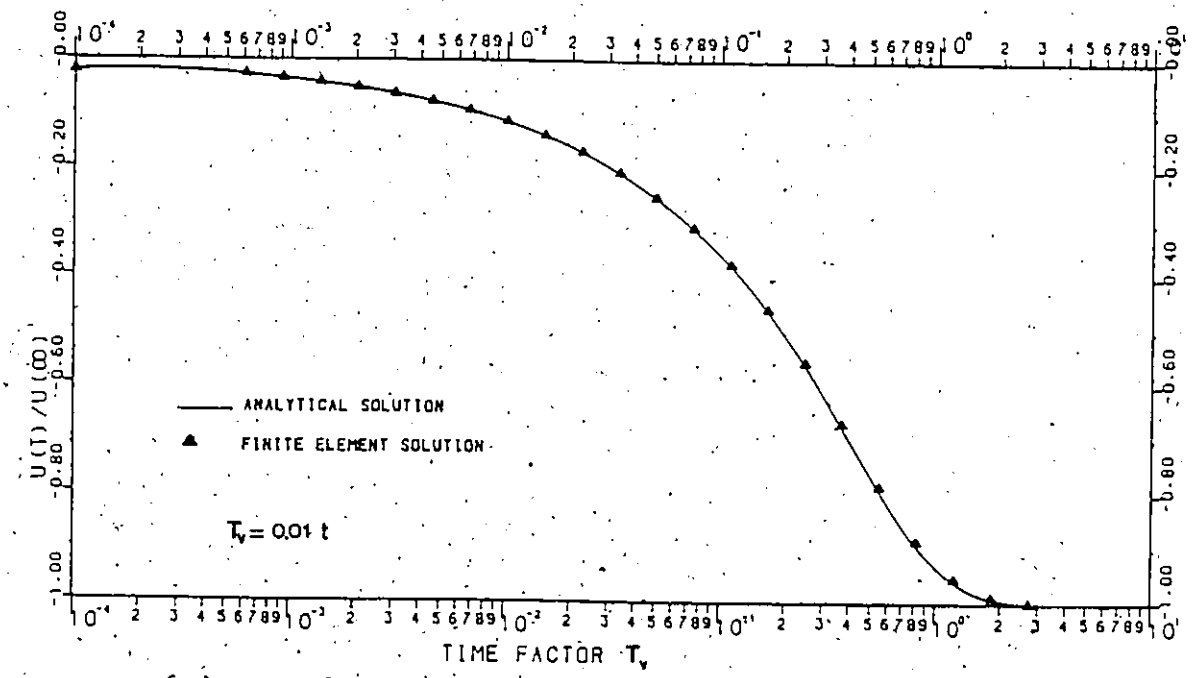
Smooth flexible footing resting on a semi-infinite medium

Fig. 5.23 shows the finite element mesh used and the load applied was assumed to be 10 once more.

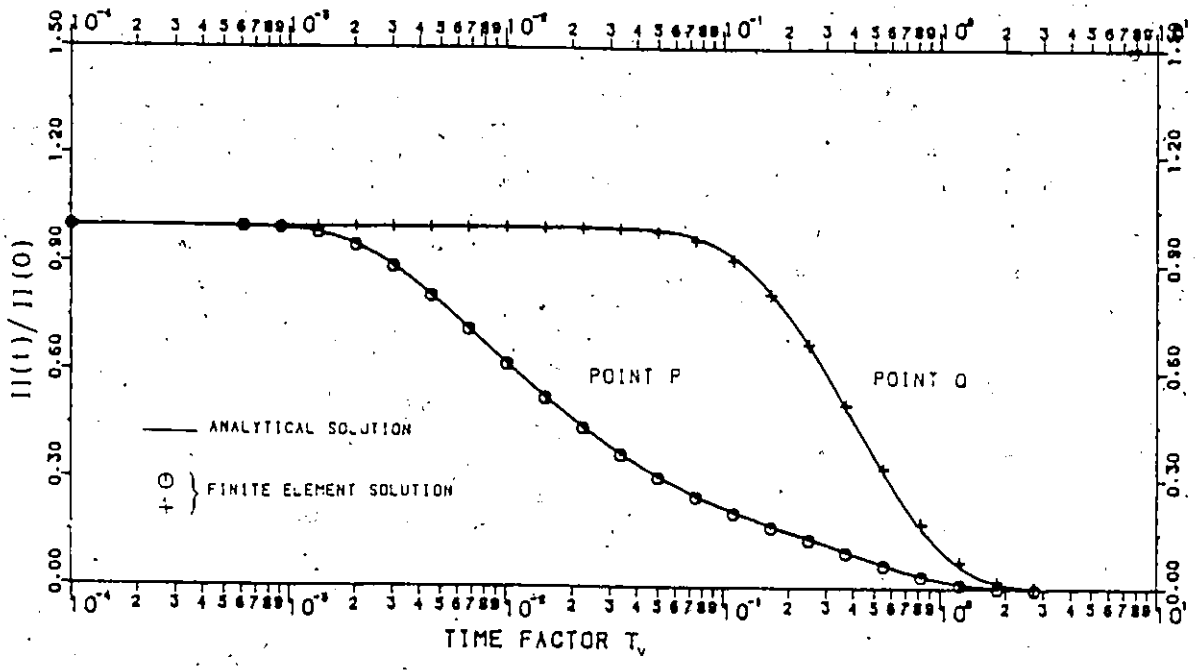
The present analysis was compared with the Biot's elastic solution (1941b). The predicted surface settlement deviates from the analytical solution by an error of ±2% up to a time factor of 100, Fig 6.9a. After this time the current analysis shows a trend to converge toward the theoretical expected value (0.0037025). Biot's solution, on the other hand, gives a steep plunge without any asymptotic behaviour. It has to be emphasized that Biot previously determined his solution by means of lengthy expansions of series within some degree of error. The general trend of an increase in excess pore pressure (Fig. 6.9b) was observed at a point distant a/2 below the centre line of the footing. Figs. 6.9d,e show alternating signs of the effective stress

variation which may explain the change in displacement field patterns illustrated in Fig. 6.9c at different times of consolidation.

The time span was extended to 66 steps adopting option 3 for the marching scheme (see Table B1), and the complete analysis required 487 CPU.



(a) surface settlement vs. time (comparison with analytical solution)



(b) excess pore pressure vs. time (comparison with analytical solution)

Fig. 6.4 1-D Coupled elastic consolidation

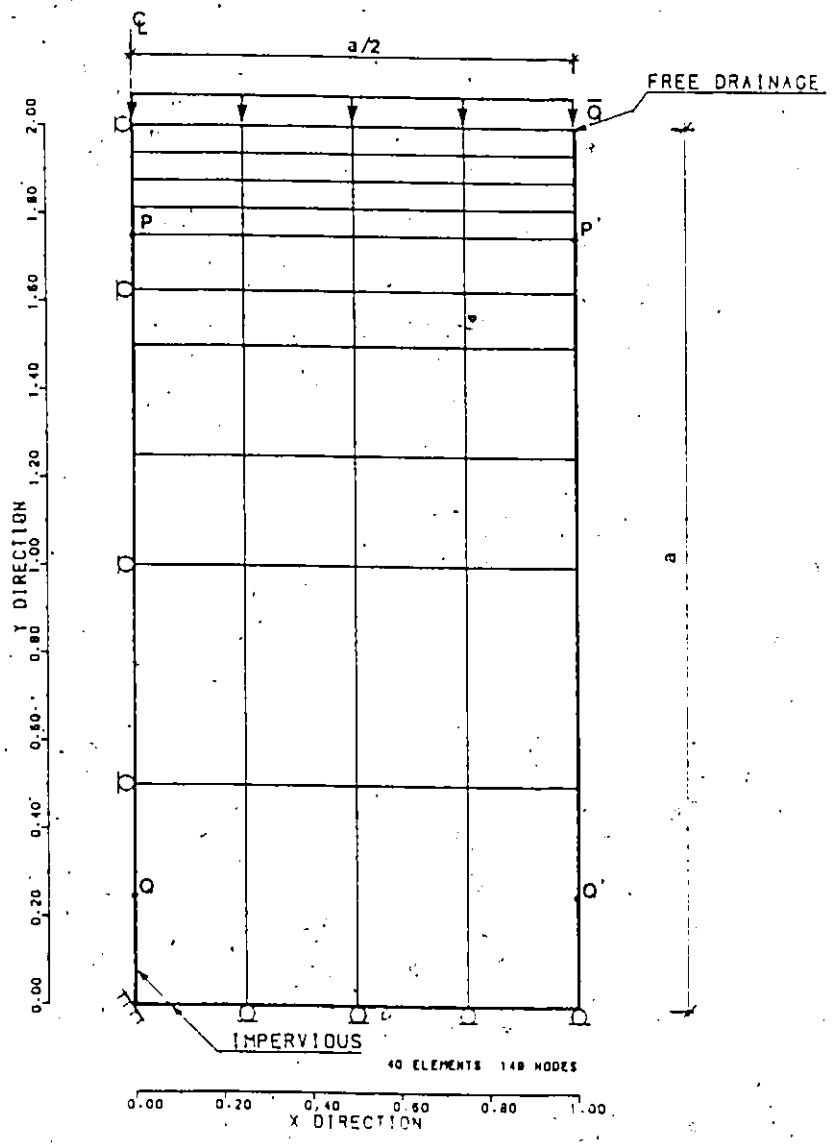
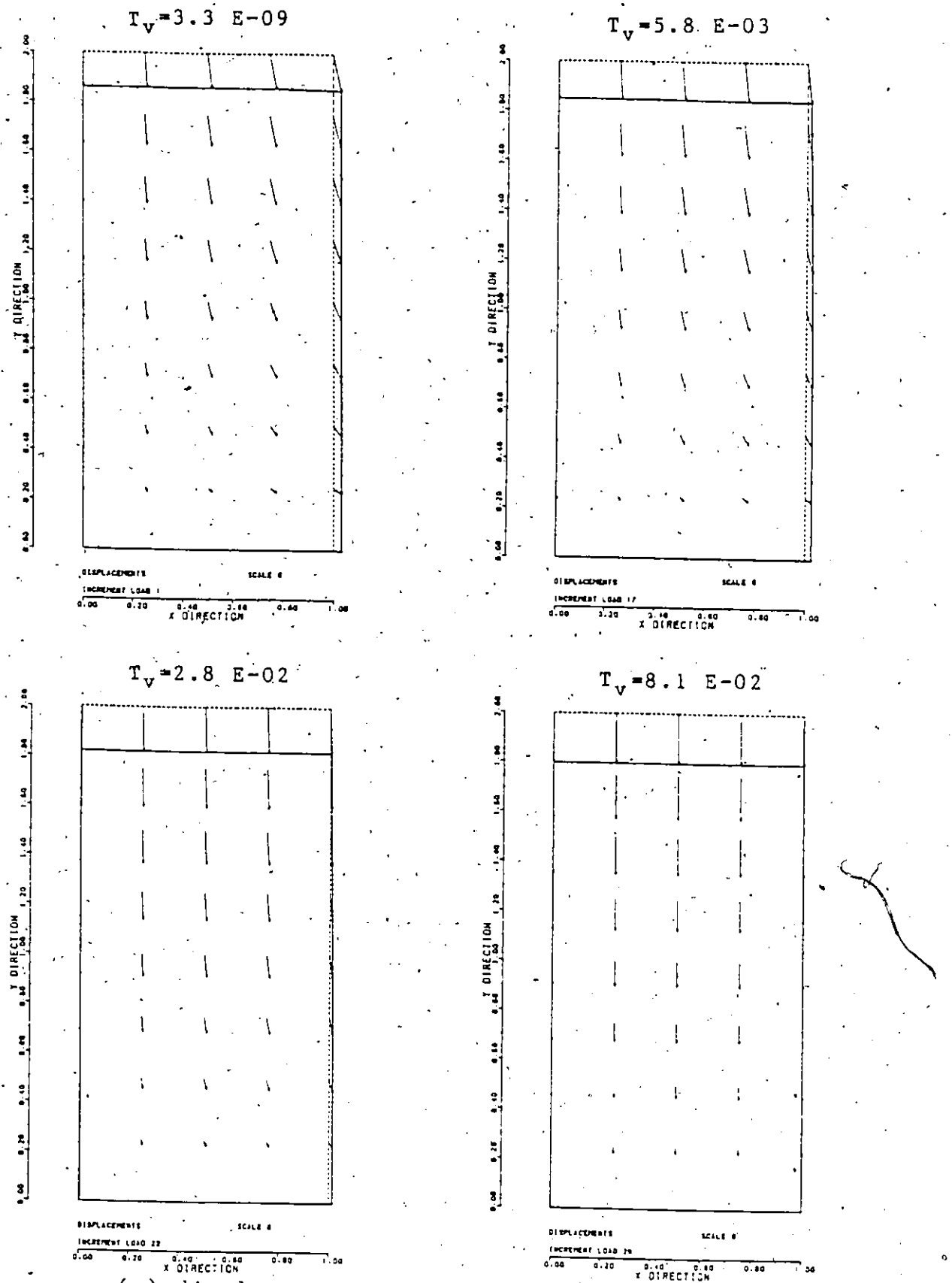


Fig. 6.5 Mesh layout for triaxial conditions

TABLE 6.1

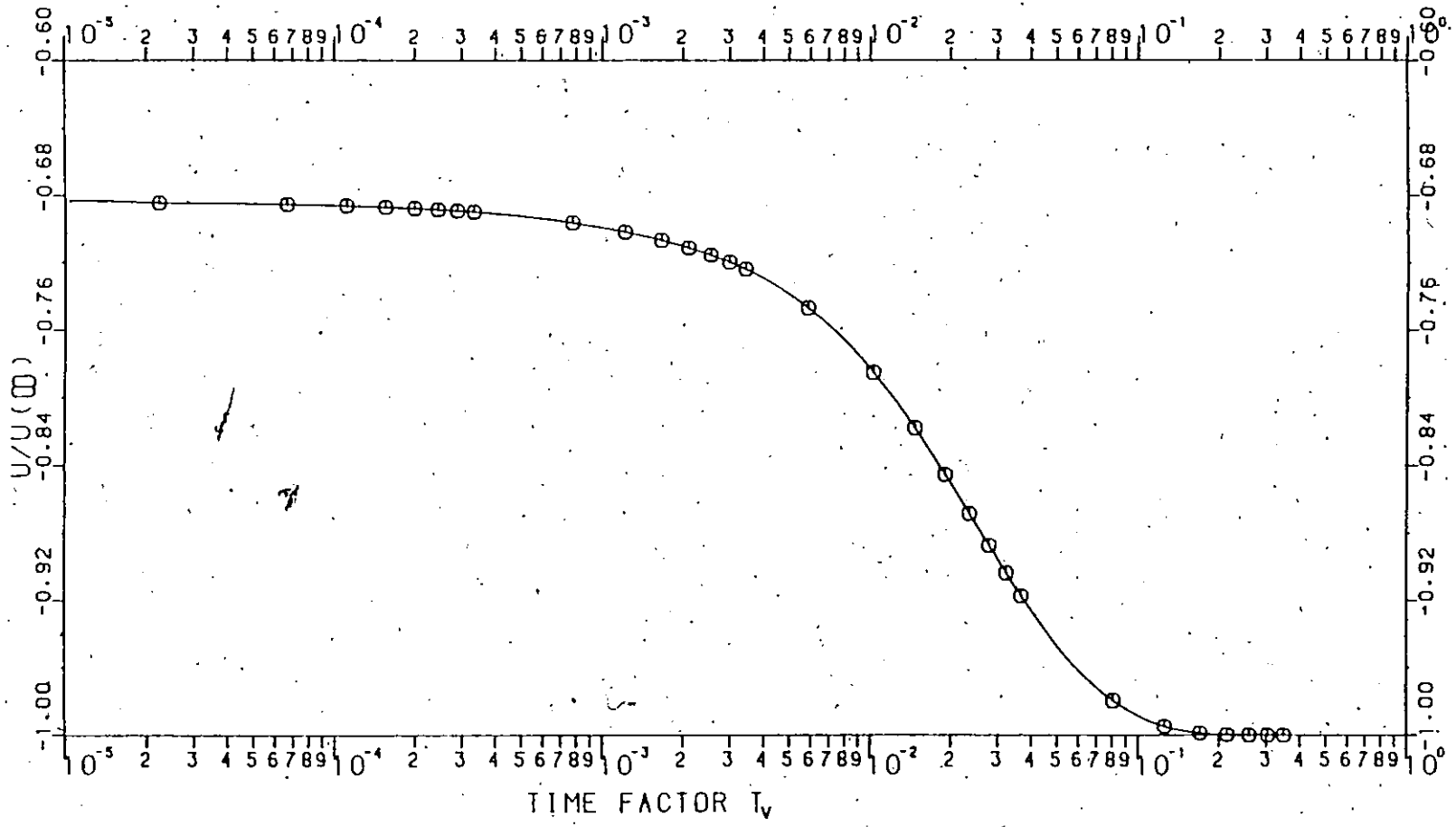
Theoretical values of stresses and pore pressures at initial and infinite times

Stress Ratio	Load imposed		Displacement imposed	
	t = 0	t = ∞	t = 0	t = ∞
$\frac{\pi^e}{Q}$	$\frac{1}{3}$	0	$\frac{1}{3}$	0
$\frac{\Delta\sigma'_z}{Q}$	$-\frac{2}{3}$	-1	$-\frac{2}{3}$	$-\frac{2}{3}$
$\frac{\Delta\sigma'_r}{Q}$	$\frac{1}{3}$	0	$\frac{1}{3}$	0
$\frac{\Delta p'}{p^e_o}$	0	-1	0	$-\frac{2}{3}$
$\frac{\Delta q}{\pi^e_o}$	3	3	3	2



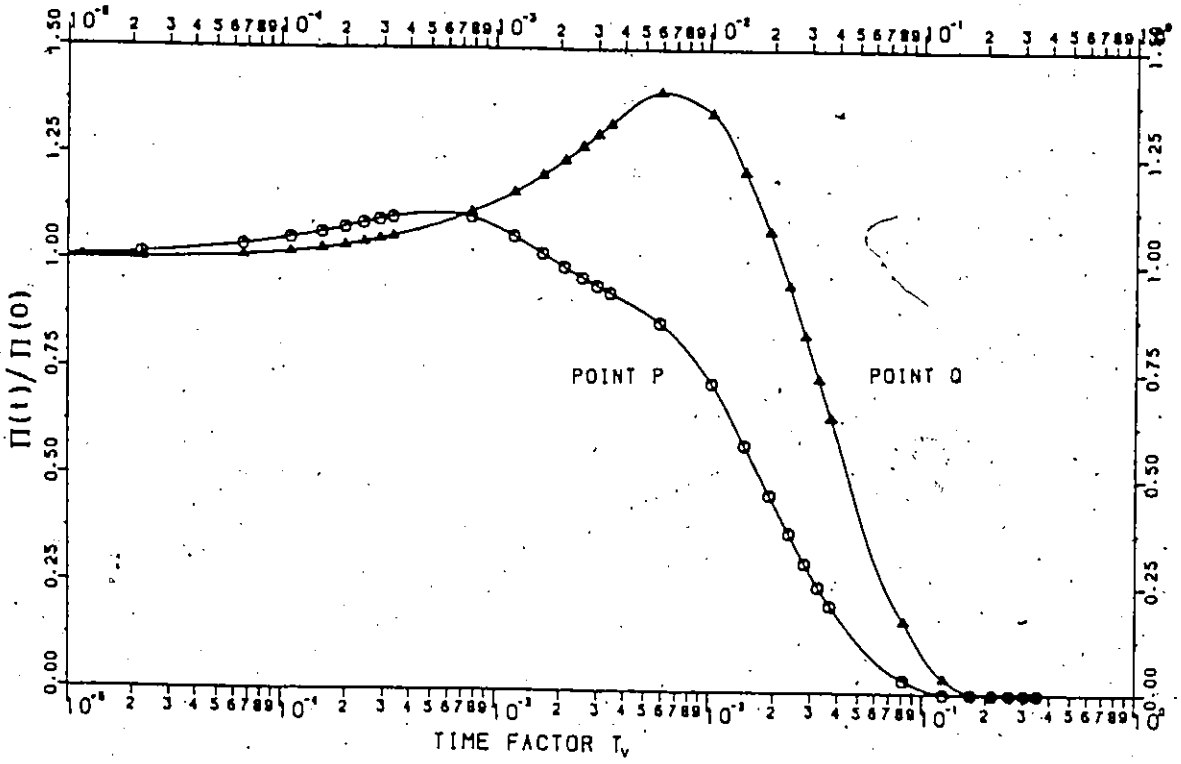
(a) displacement fields at selected times

Fig. 6.6 Triaxial coupled elastic consolidation (load imposed)

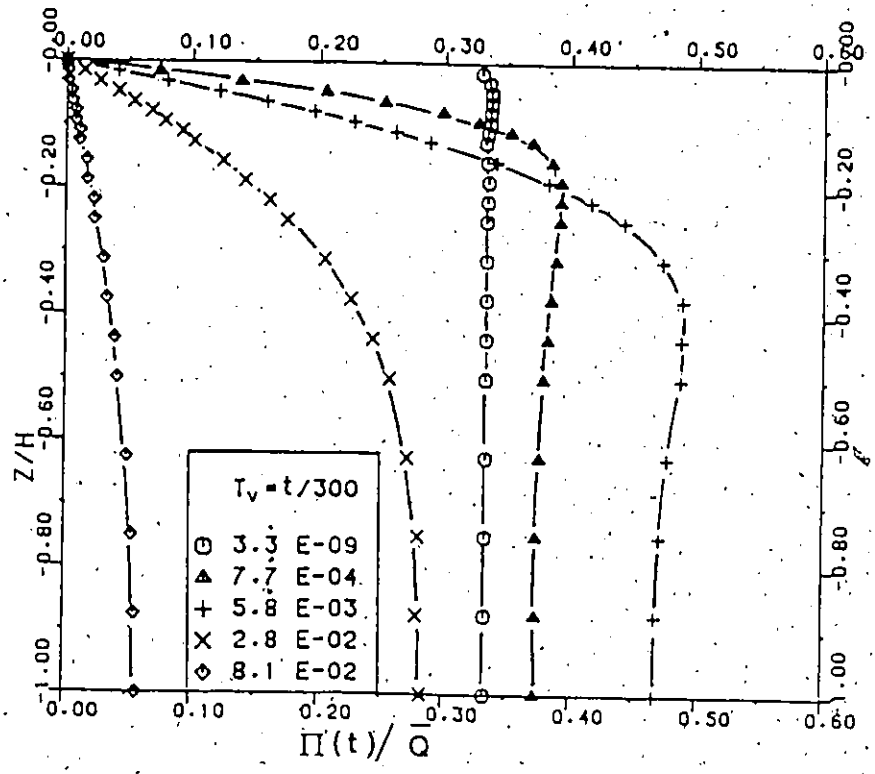


(b) surface settlement vs. time

Fig. 6.6 Triaxial coupled elastic consolidation (load imposed)

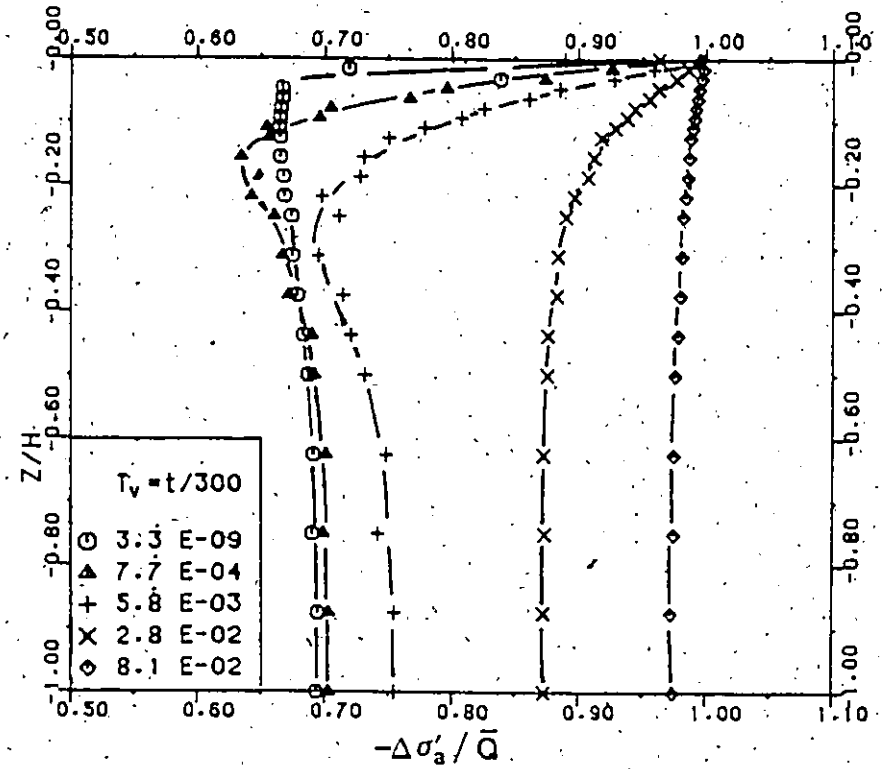


(c) excess pore pressure vs. time

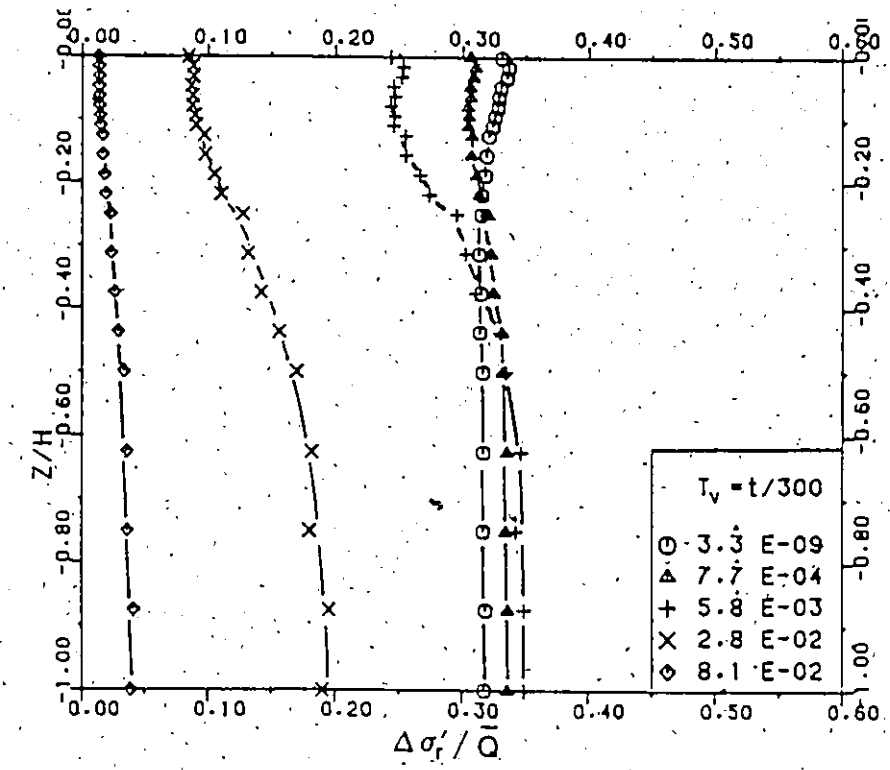


(d) excess pore pressure isochrones

Fig. 6.6 Triaxial coupled elastic consolidation (load imposed)

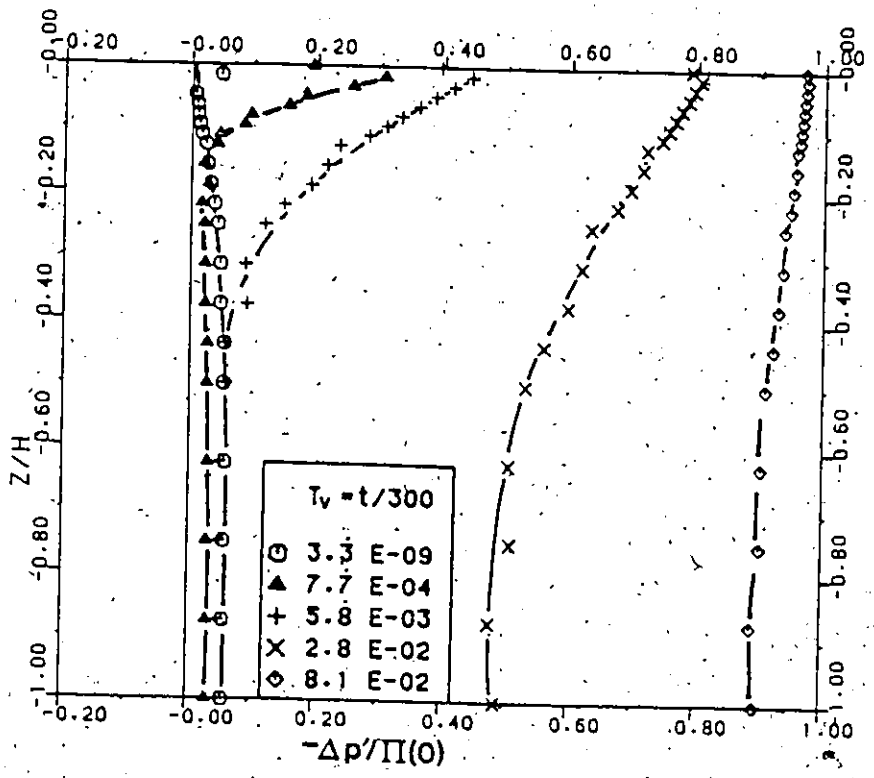


(e) axial effective stress isochrones

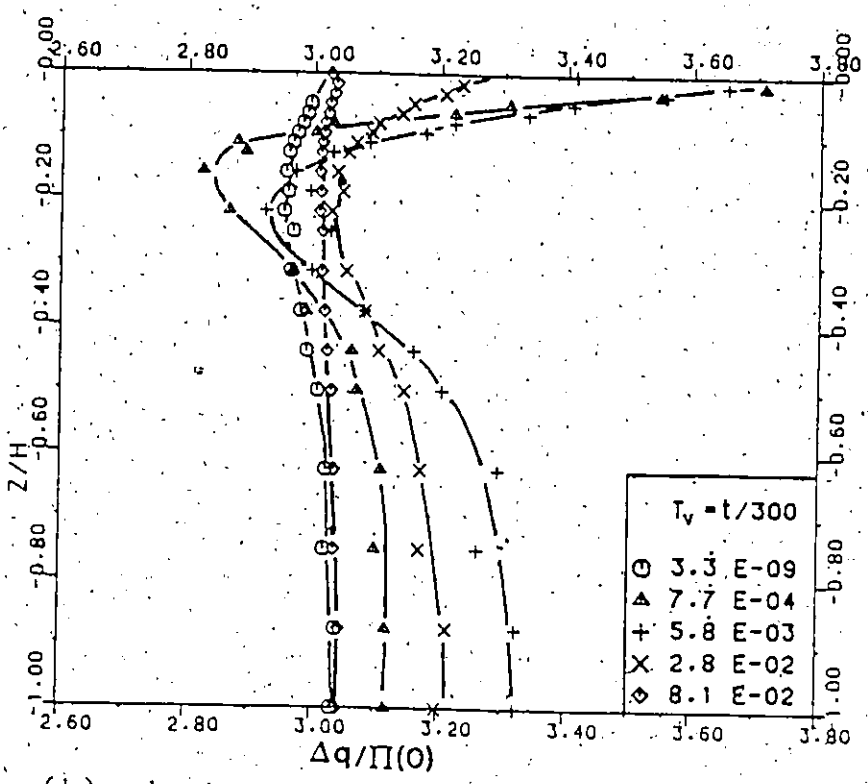


(f) radial effective stress isochrones

Fig. 6.6 Triaxial Coupled elastic consolidation (load imposed)

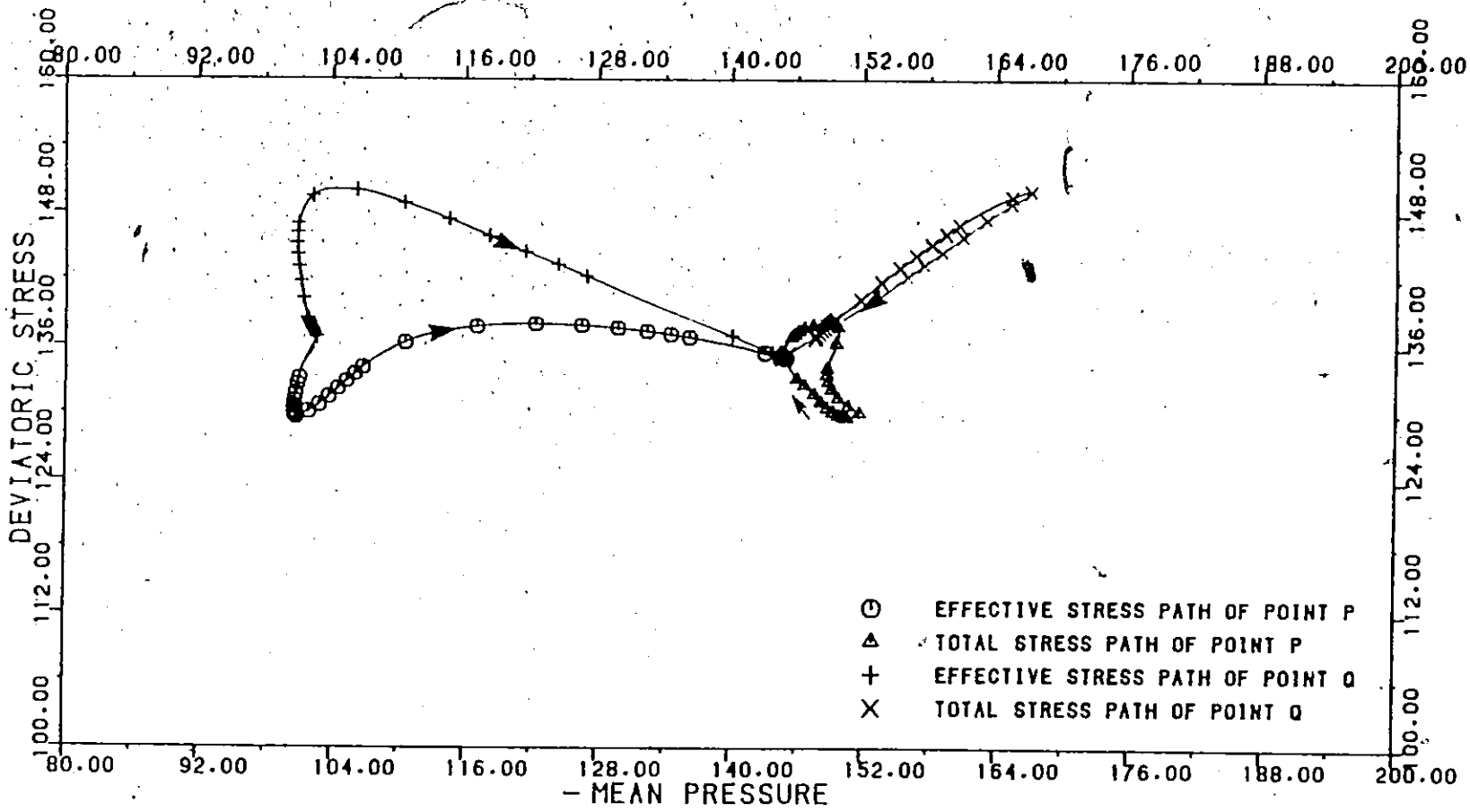


(g) effective mean preassure isochrones



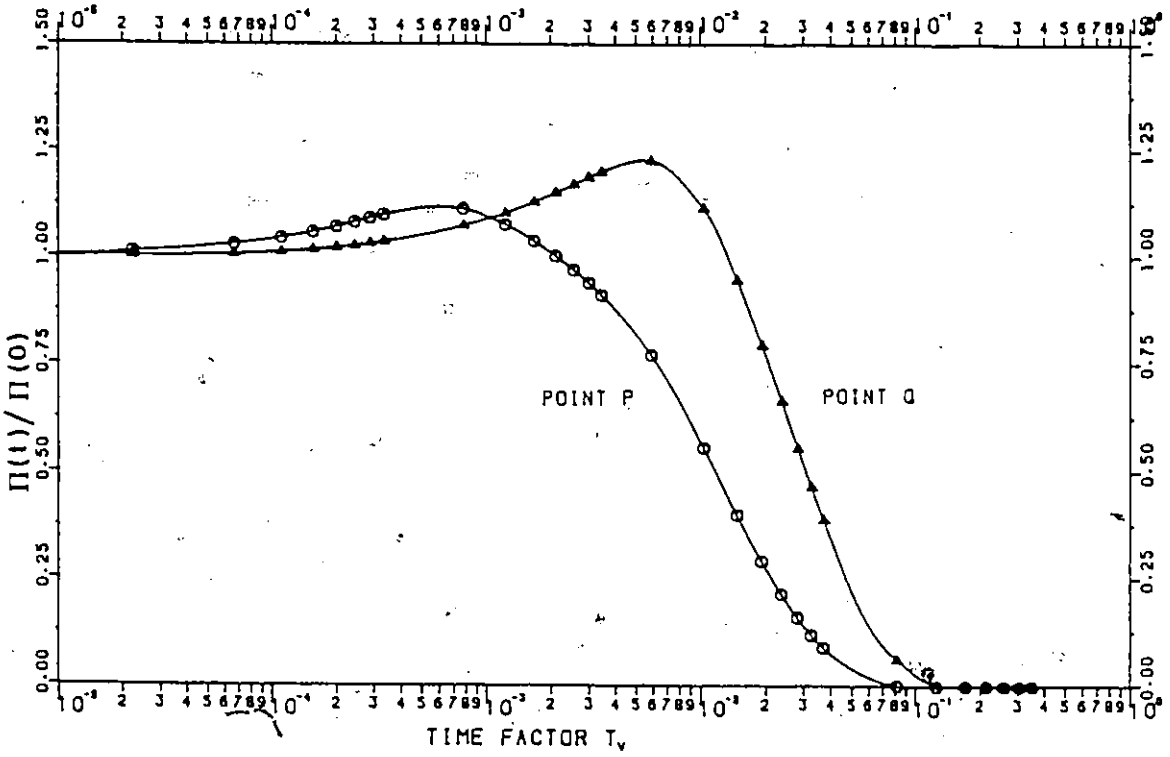
(h) deviatoric stress isochrones

Fig. 6.6 Triaxial Coupled elastic consolidation (load imposed)

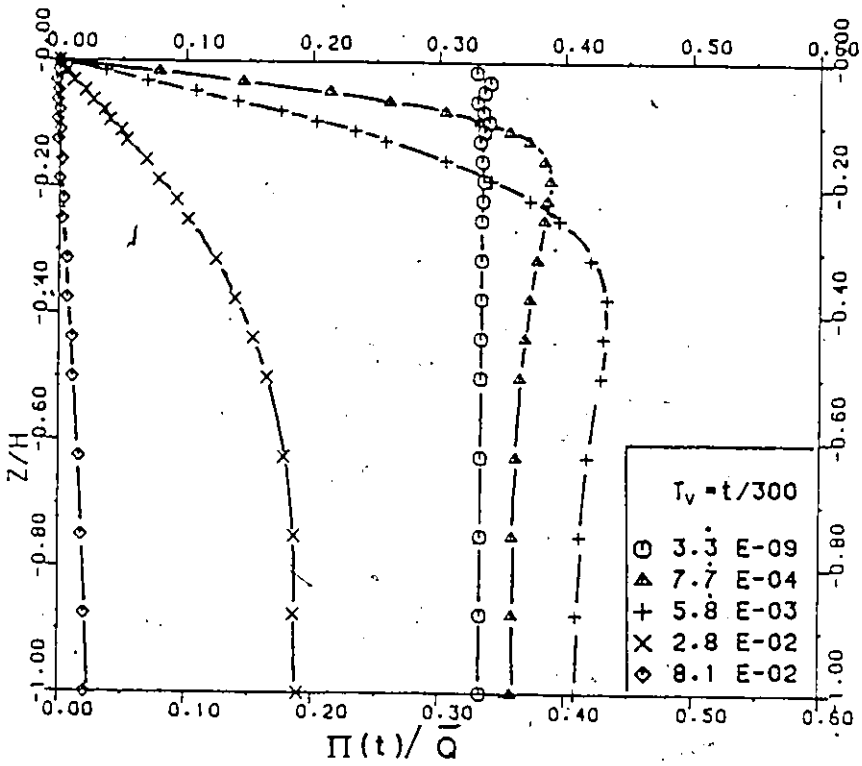


(i) stress path

Fig. 6.6 Triaxial Coupled elastic consolidation (load imposed)

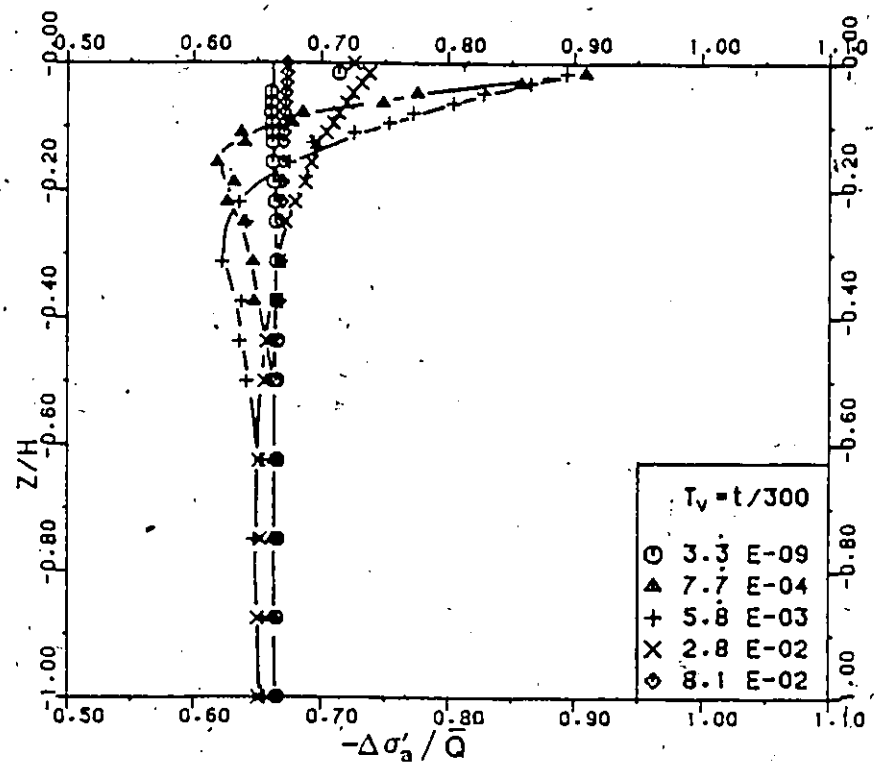


(a) excess pore pressure vs. time

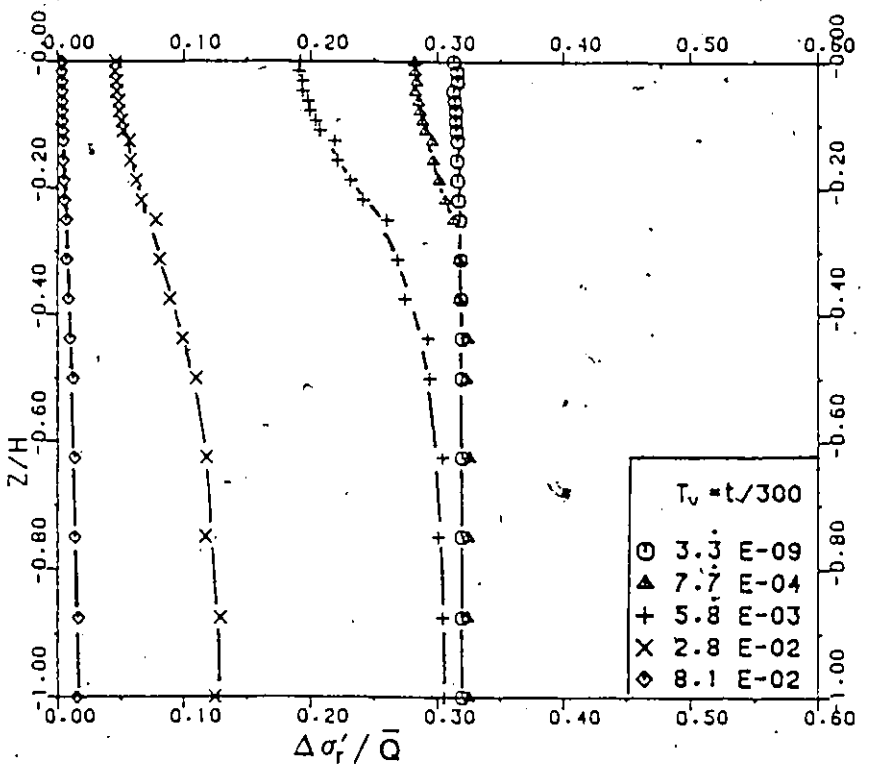


(b) excess pore pressure isochrones

Fig. 6.7 Triaxial Coupled elastic consolidation (displacement imposed)

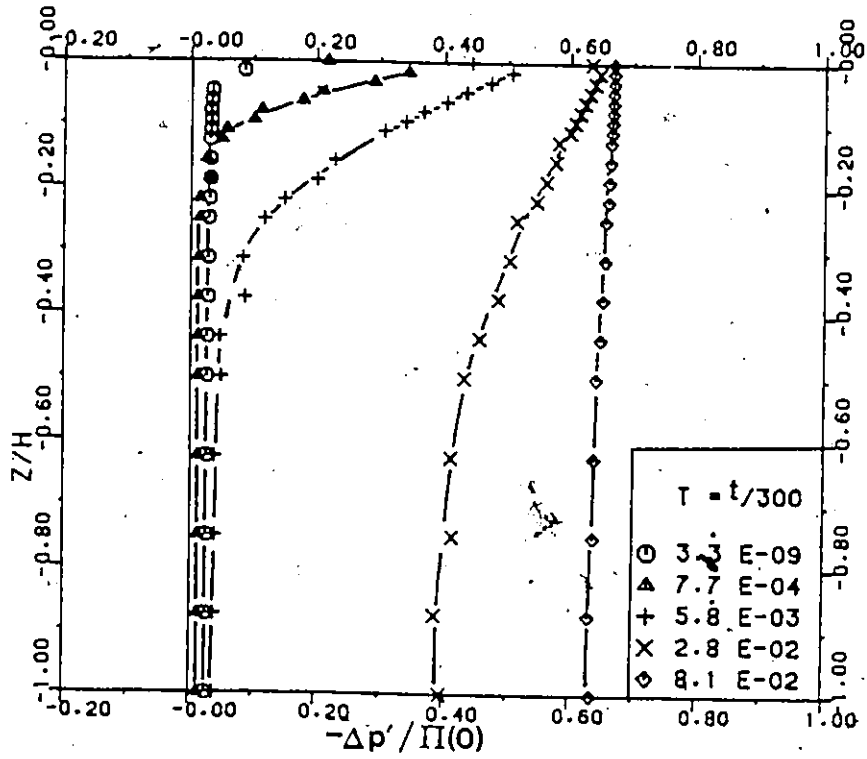


(c) axial effective stress isochrones

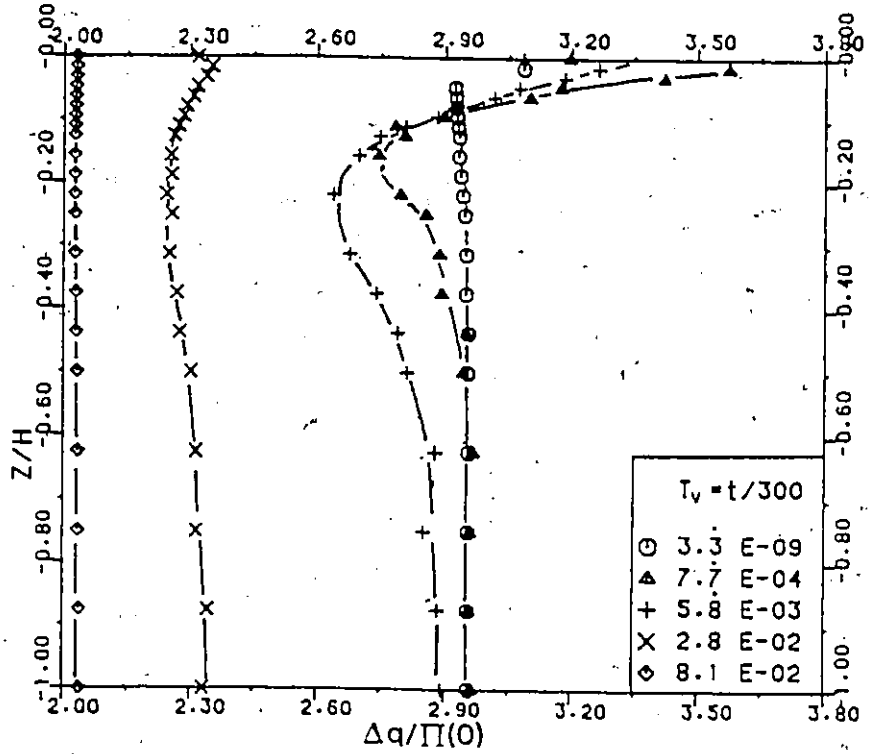


(d) radial effective stress isochrones

Fig. 6.7 Triaxial Coupled elastic consolidation (displacement imposed)

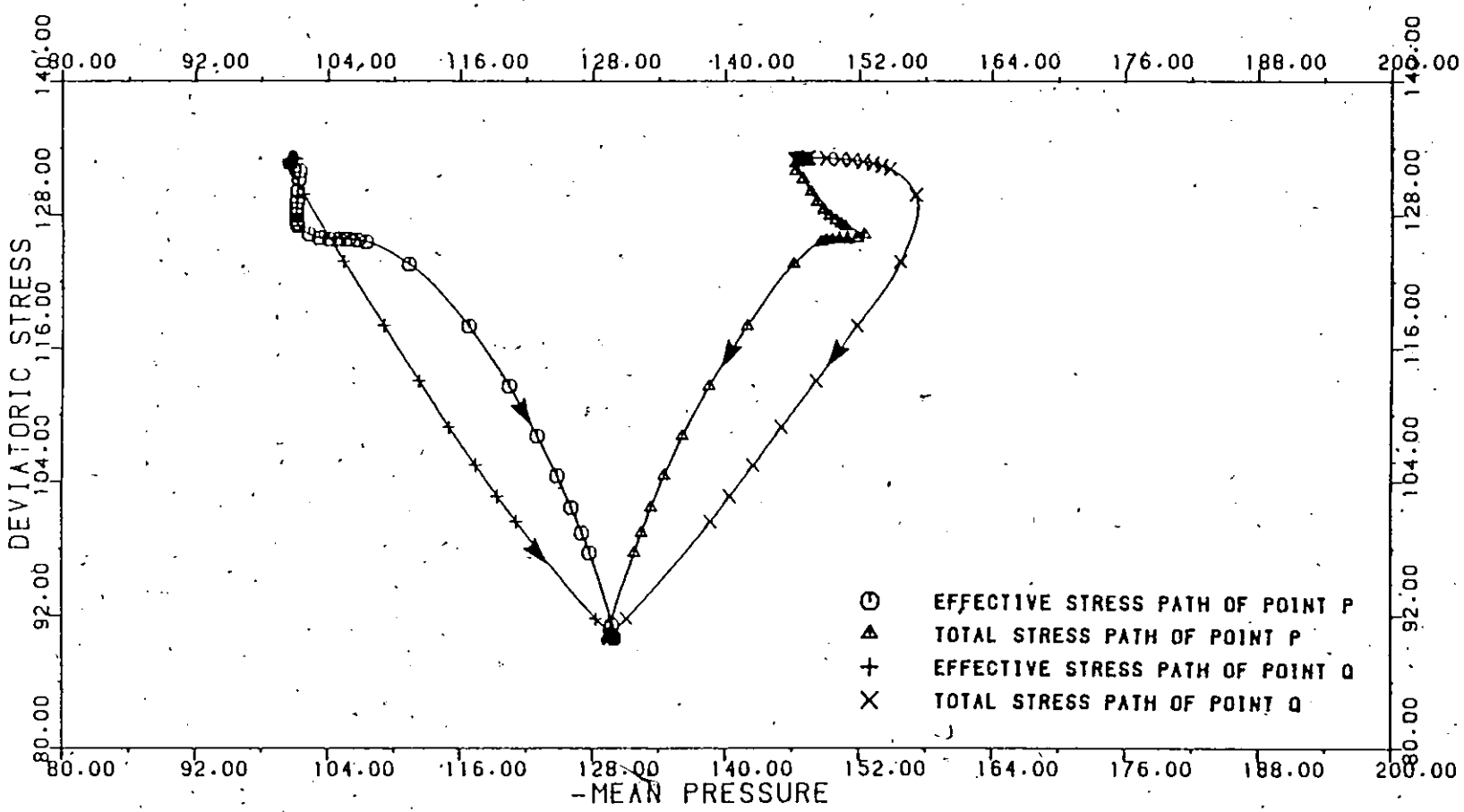


(e) effective mean pressure isochrones



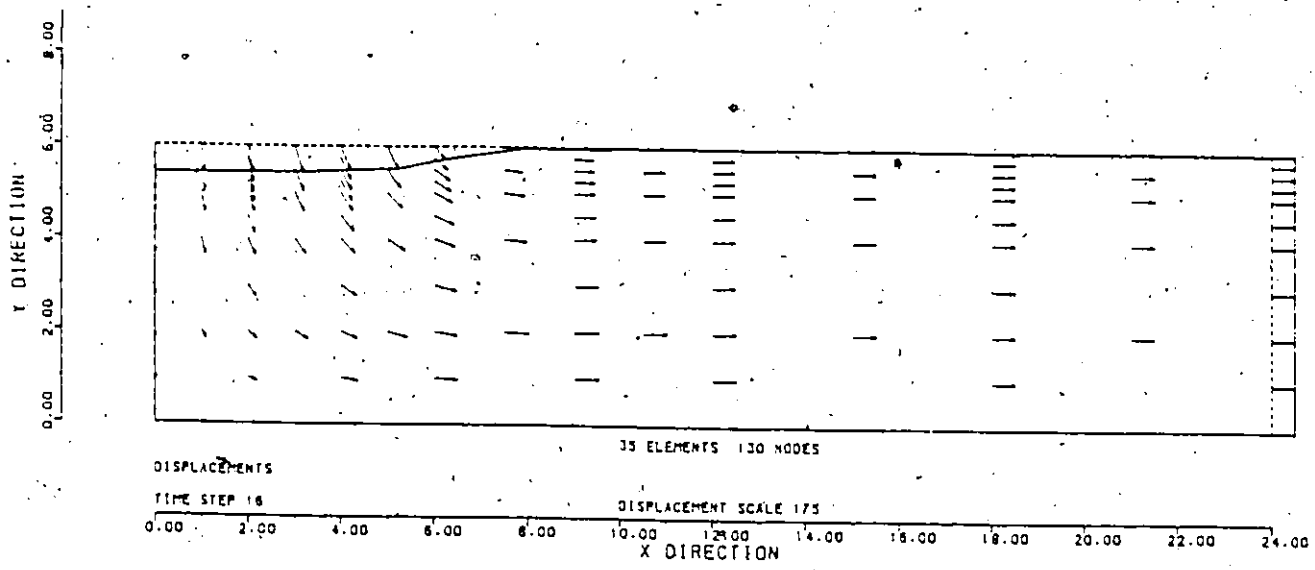
(f) deviatoric stress isochrones

Fig. 6.7 Triaxial Coupled elastic consolidation (displacement imposed)

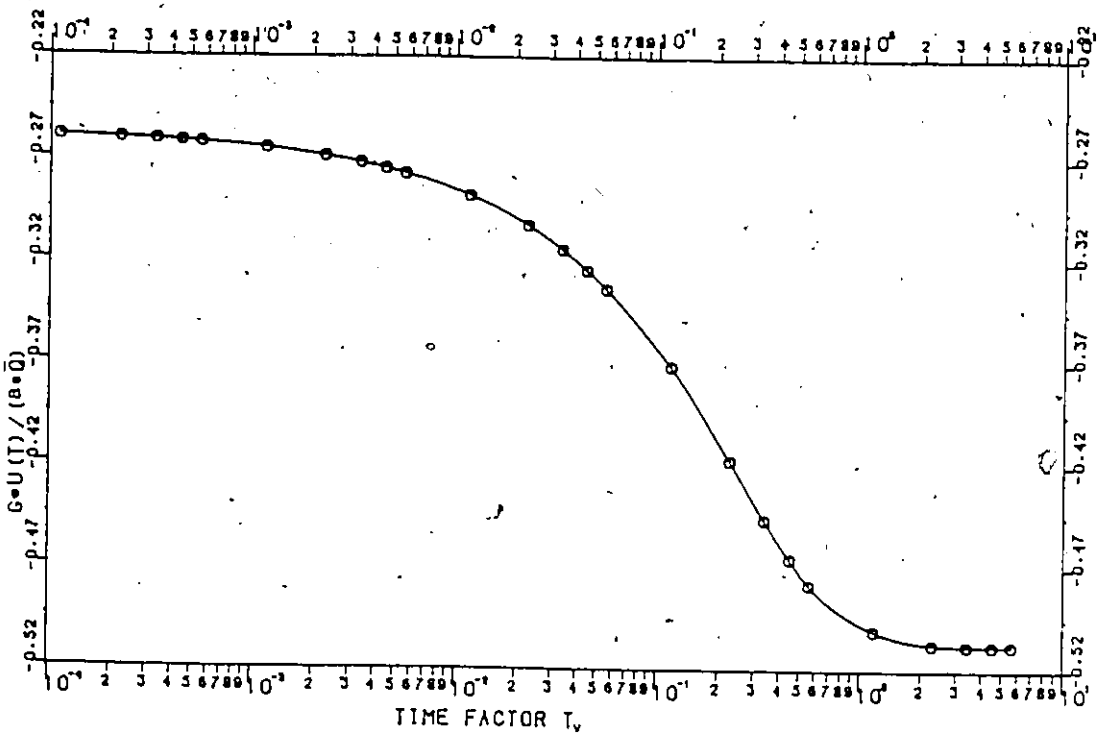


(g) stress path

Fig. 6.7 Triaxial Coupled elastic consolidation (displacement imposed)

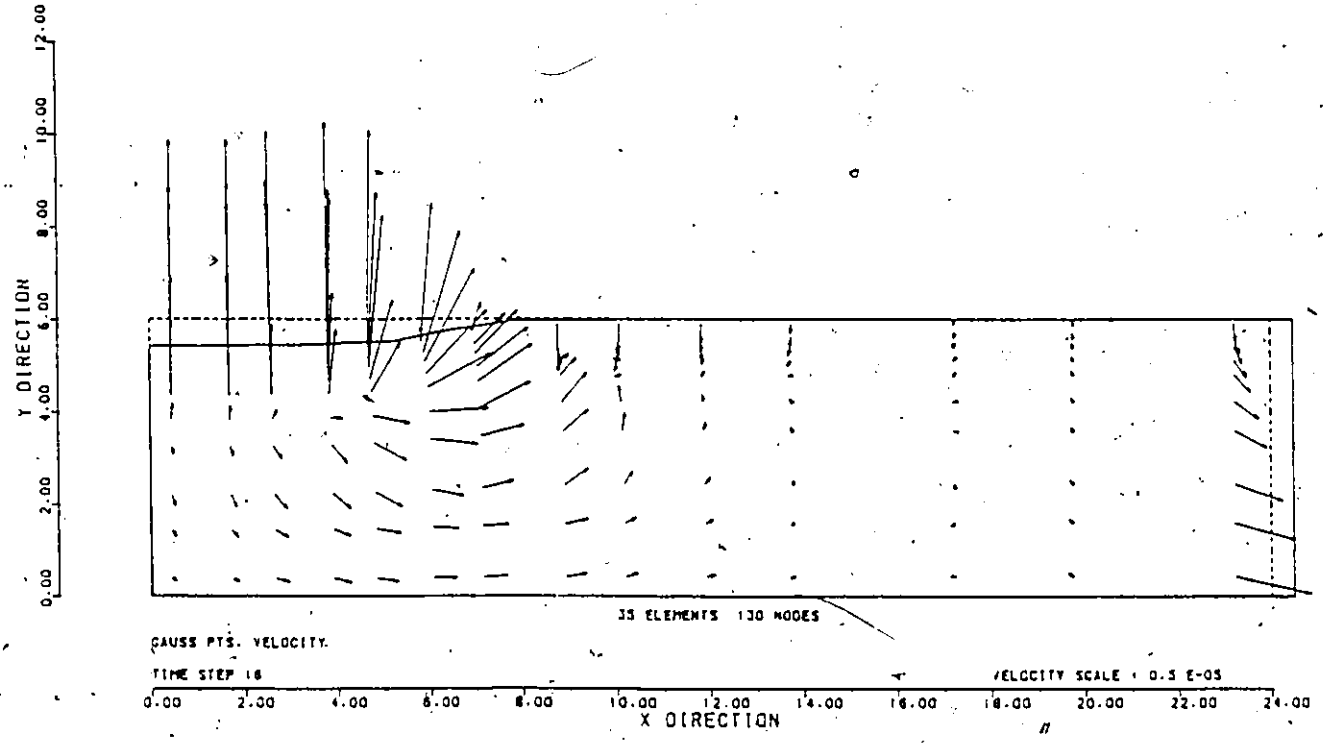


(a) displacement field T_v=3.3 E-04

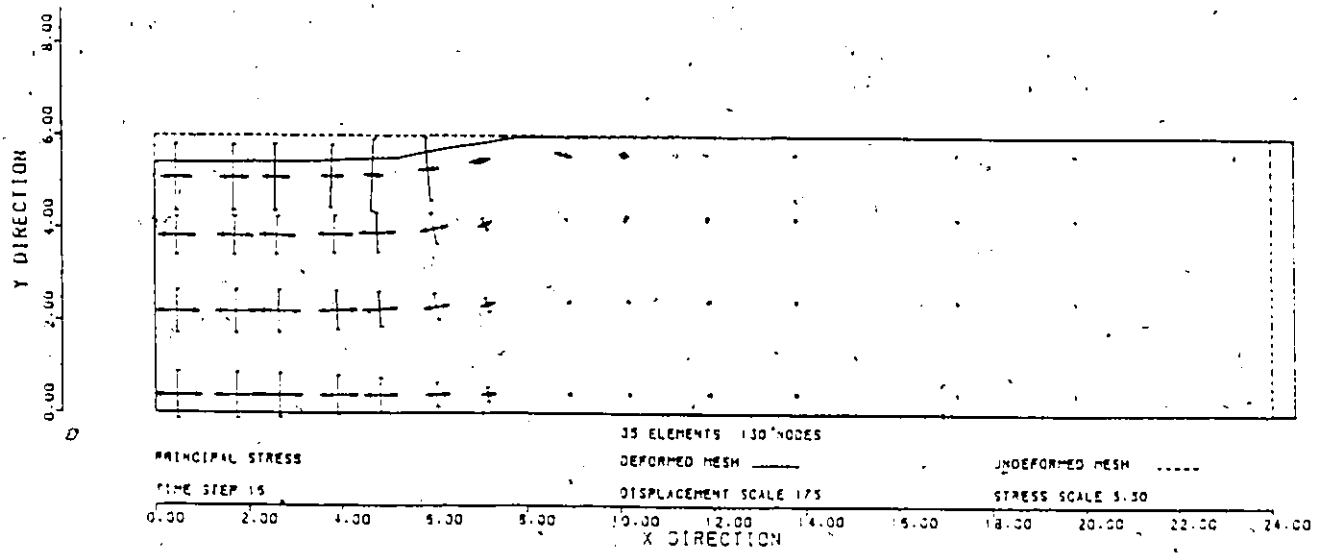


(b) surface settlement vs. time

Fig. 6.8 Coupled elastic consolidation in a semi-infinite layer

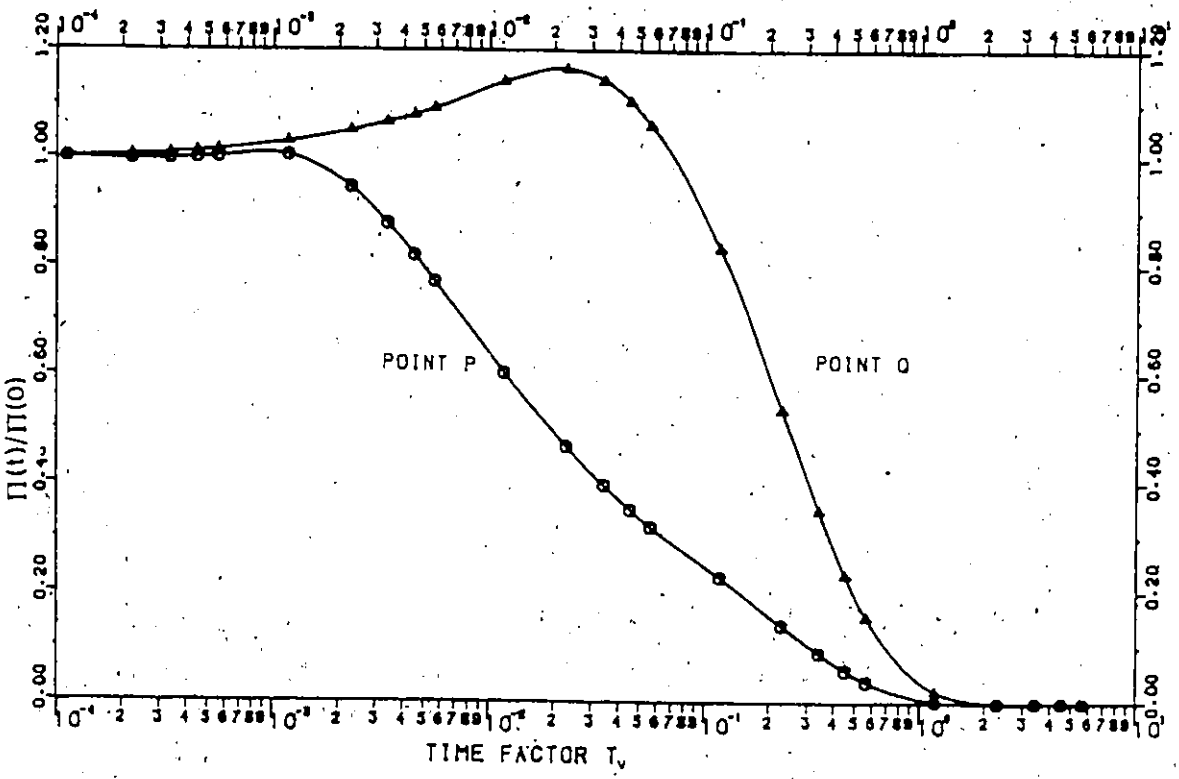


(c) fluid velocity field at T_v=3.3 E-04

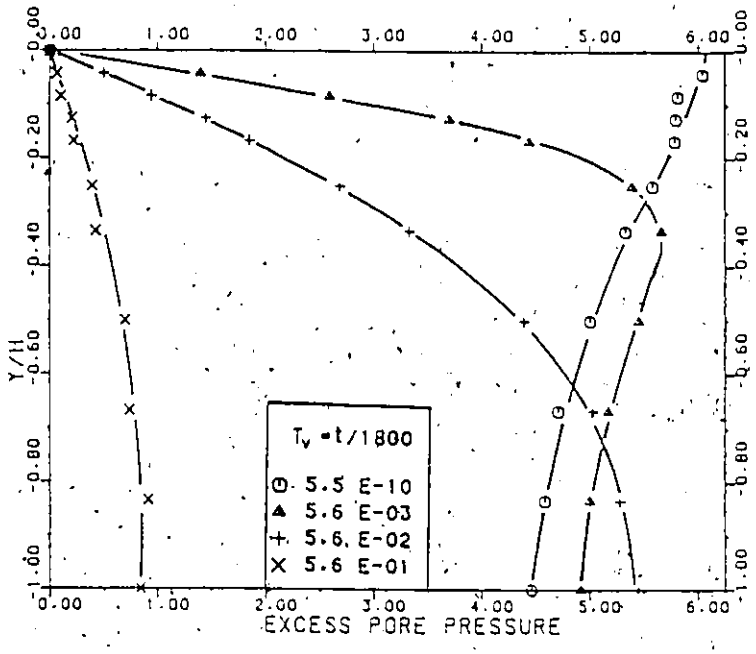


(d) principal effective stress field at T_v =3.3 E-04

Fig. 6.8 Coupled elastic consolidation in a semi-infinite layer

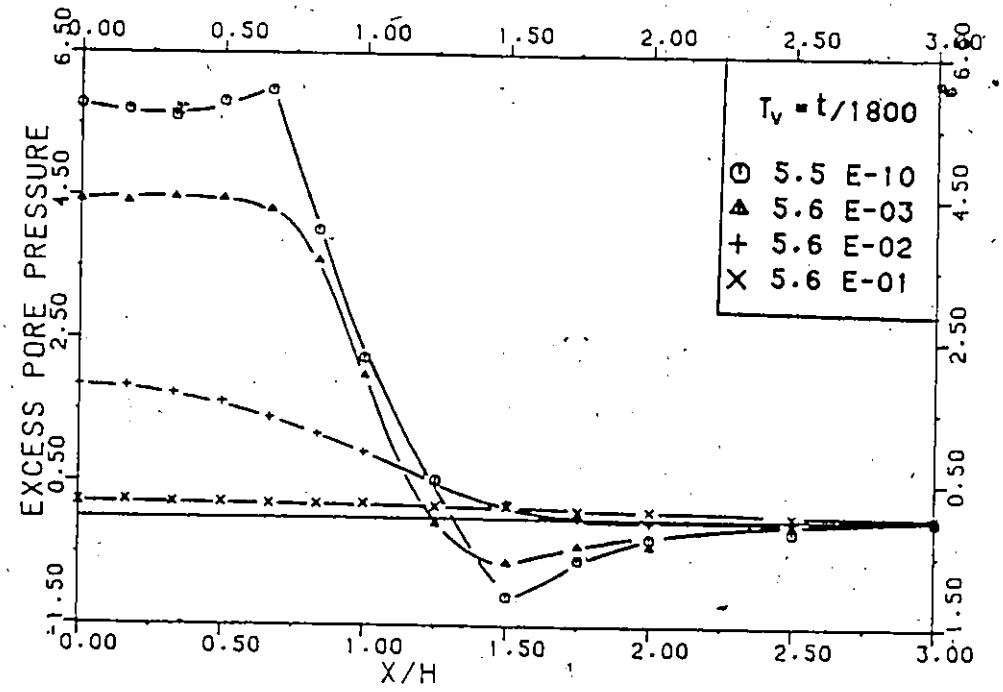


(e) excess pore pressure vs. time

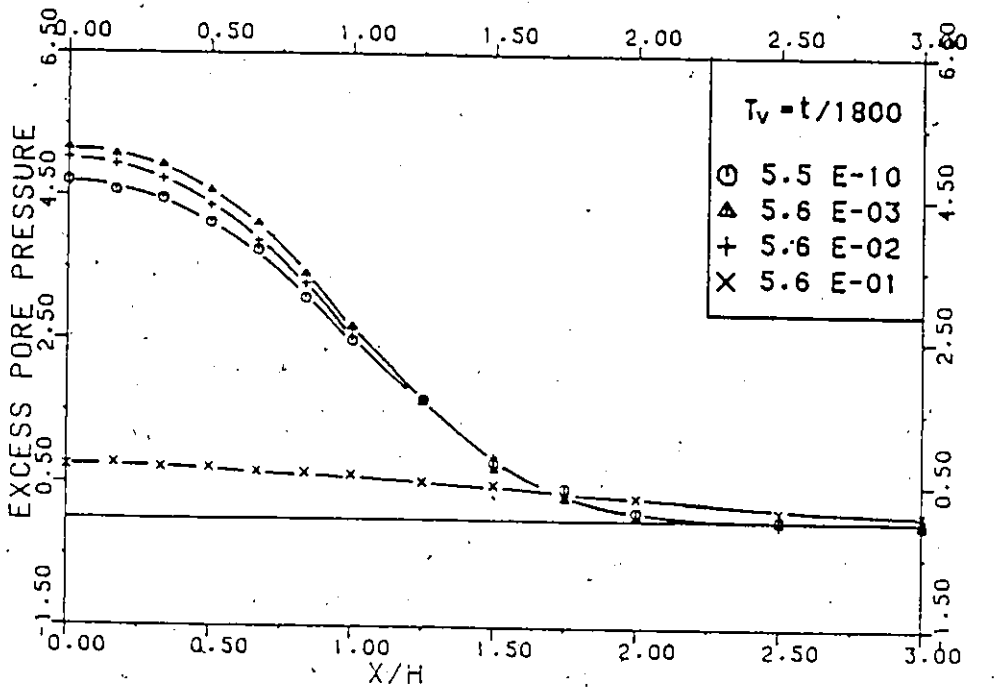


(f) excess pore pressure isochrones at section A-A

Fig. 6.8 Coupled elastic consolidation in a semi-infinite layer

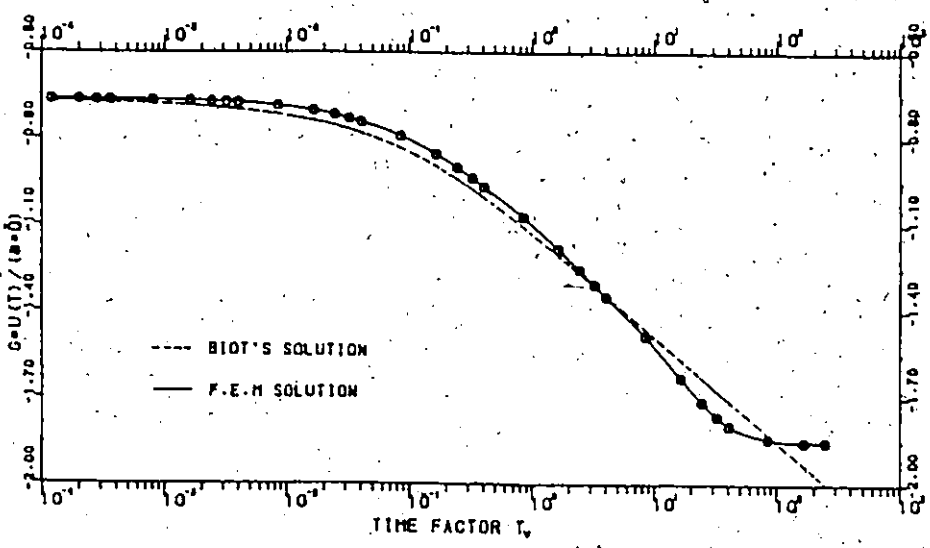


(g) excess pore pressure isochrones at section B-B

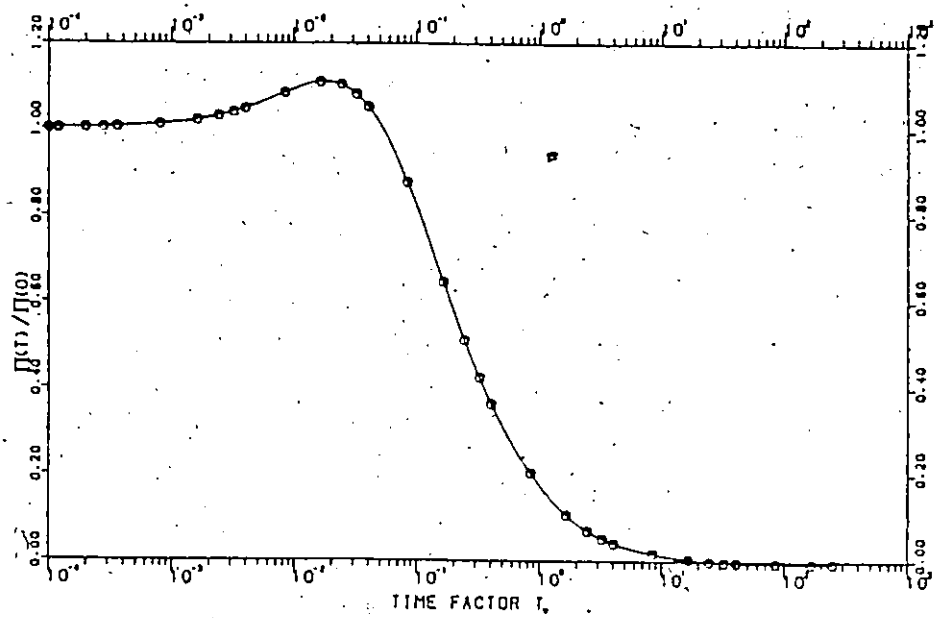


(h) excess pore pressure isochrones at section C-C

Fig. 6.8 Coupled elastic consolidation in a semi-infinite layer

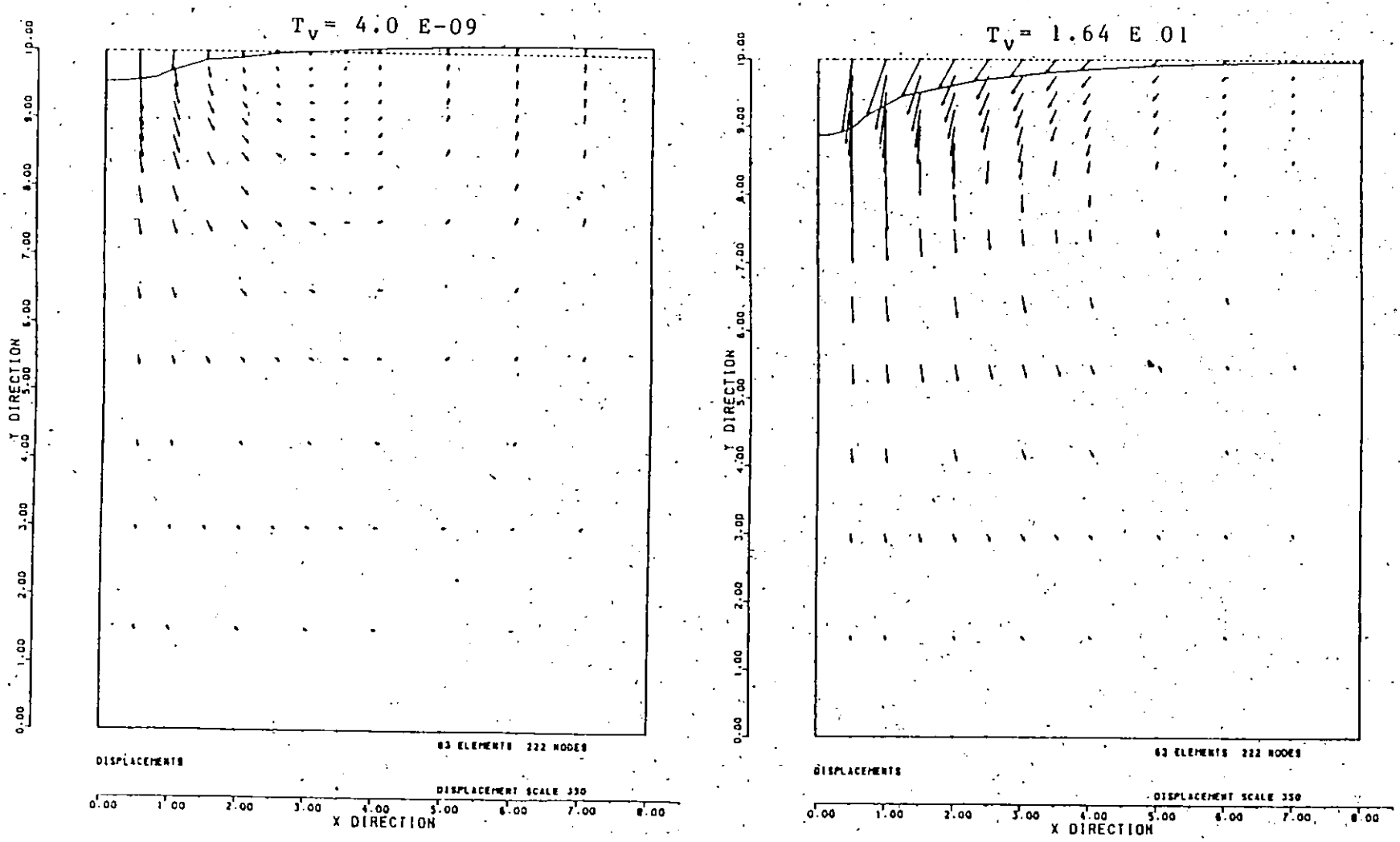


(a) surface settlement vs. time



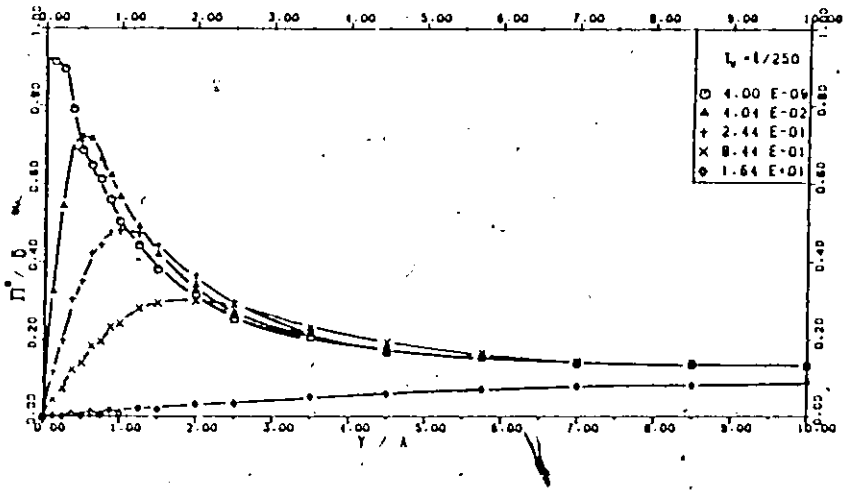
(b) excess pore pressure vs. time

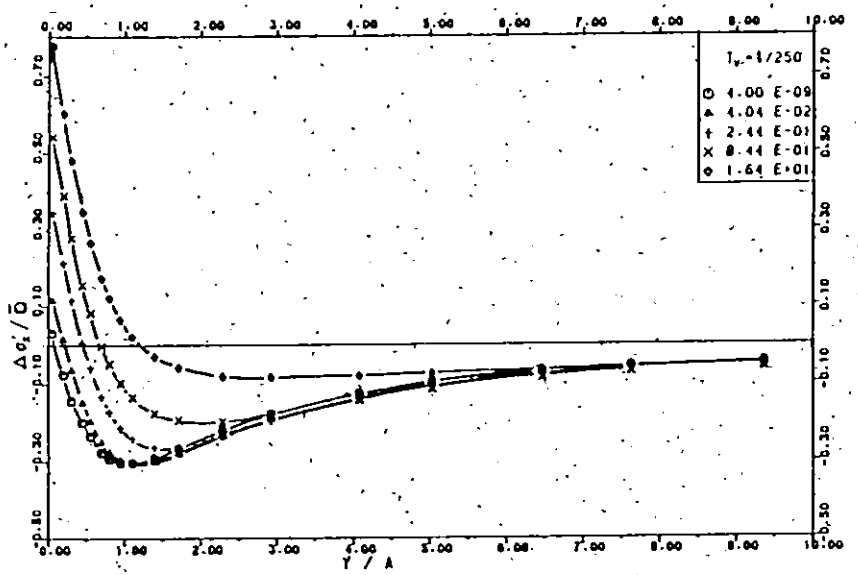
Fig. 6.9 Coupled elastic consolidation in a semi-infinite medium

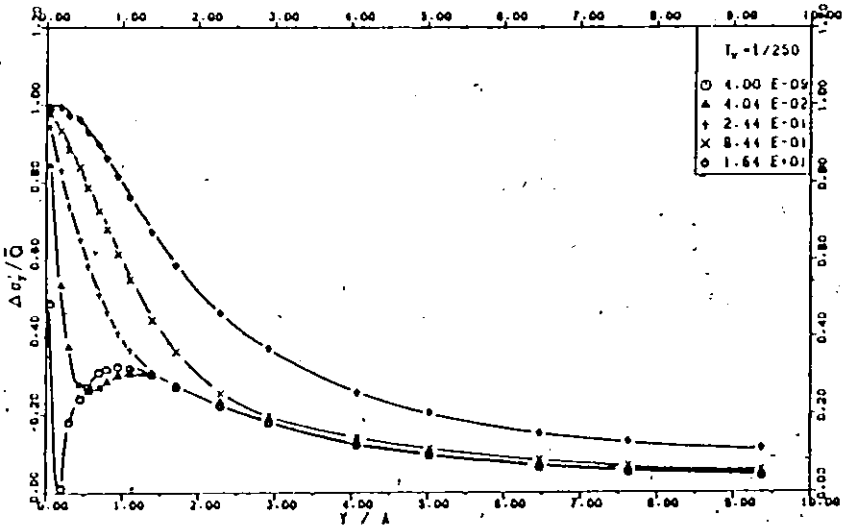


(c) displacement fields at different times

Fig. 6.9 Coupled elastic consolidation in a semi-infinite medium



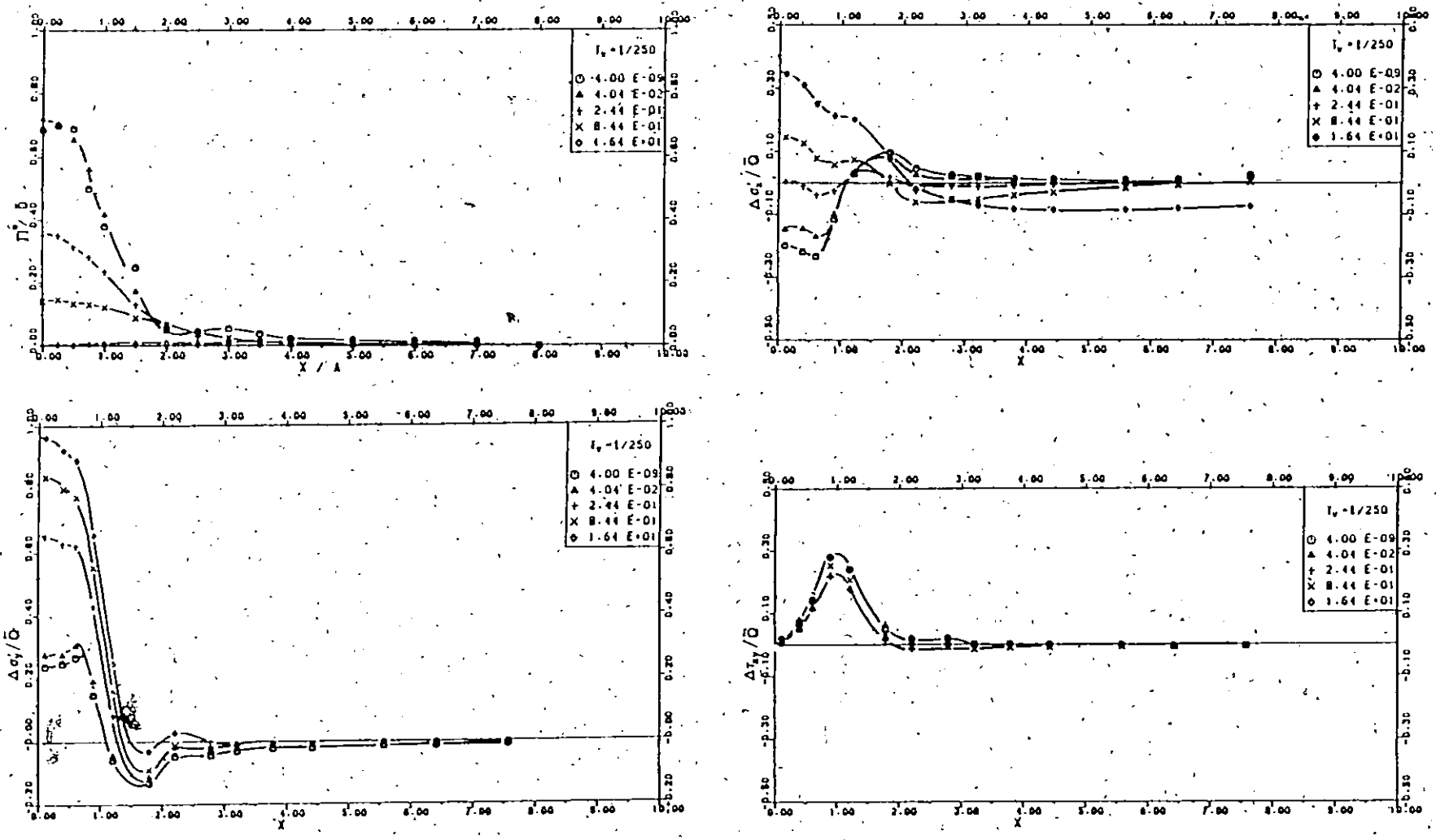




(d) excess pore pressure and effective stress isochrones along the centre line

Fig. 6.9

Coupled elastic consolidation in a semi-infinite medium



(e) excess pore pressure and effective stress isochrones along the horizontal section at y/a = 0.5

Fig. 6.9 Coupled elastic consolidation in a semi-infinite medium

6.5.2 Elasto-Plastic Consolidation

The performance of TEPSA in elasto-plasticity was demonstrated in the analysis of a triaxial test sample isotropically preconsolidated at 100. Under these conditions it was possible to theoretically evaluate the initial stress values within the concept of elasto-plasticity.

Fig. 6.10 illustrates the graphical representation of the initial conditions for both the linear elastic perfectly plastic and Critical State numerical model responses. Starting from an initial isotropic condition, stress point P, a stress increment provokes the point to move to point A along a line at slope of 3 vertical to 1 horizontal (triaxial line). According to the structure of the program the "undrained" solution (t=0.0) is first obtained assuming elastic skeleton behaviour. In this instance the variation of mean pressure due to the load applied is exclusively carried by the fluid phase, i.e $\Delta p = \Delta \pi^e$ and $\Delta p' = 0.0$. Thus the resultant effective stress point is located at A'.

In the current example under study, A' lies above the yield surface so that elasto-plastic deformations are deemed to occur. Assuming the material to behave linearly elastic perfectly plastic, the stress point is consequently reduced on the surface at point B' where elasto-plastic balancing takes place. This process requires the effective stress point to stay on the failure surface. When load is imposed to the sample, the point B' subsequently moves to D' to

satisfy stress equilibrium. On the other hand when displacements are prescribed, B' moves to a different point C' such that the calculated elasto-plastic displacements balance with the imposed ones.

A similar procedure is followed for the generalized Burland model. The stress point A' is first brought down on the failure surface at point P where elasto-plastic deformations start to occur. After the balance of the calculated elasto-plastic displacements with the imposed ones, point P finally moves to point E'.

It is interesting to mention that the linear elastic perfectly plastic model predicts a smaller excess pore pressure than the one calculated in elasticity. On the other hand, the situation is reversed for the generalized Burland model, which leads to a higher value. Another important distinction between the two models is the way plasticity is interpreted. In the linear elastic perfectly plastic model, when stresses are imposed, the major redistribution of plastic deformations is done at the start of the consolidation ("undrained" state ; t=0). The safety factor is at its minimum. With time the subsequent effective stress path stays inside the surface, in other words, in the elastic region and both settlement and safety factor increase. In the generalized Burland model, elasto-plastic deformations occur all the time during consolidation since the yield surface evolves with the effective stress point.

In the following two examples, a displacement (d=0.018) equal to the one applied in the elastic case was imposed to the same cylindrical sample, but in the assumption of the two models. The mesh used and boundary conditions are given in Fig. 6.5 whereas material properties are found in Table 5.9. The material was considered to be weightless and isotropically preconsolidated at a pressure of 100. A normally consolidated state was assumed in the generalized Burland model.

Figs. 6.11 and 6.12 show the computed stress and pore pressure histories in the linear elastic perfectly plastic and generalized Burland models respectively. The essential parameters were normalized with respect to $\bar{Q}$ taken as the initial total axial stress and equal to 114. The initial excess pore pressure was 33 compared to a value 45 in the analogous elastic case. For the generalized Burland model the initial axial total stress and pore pressure were 52.2 and 36.6 respectively. It was found that the latter model consumed (551 CPU) more than twice the computational time taken by the linear elastic perfectly plastic counterpart (215 CPU).

Two-dimensional analysis - Smooth, flexible footing resting on a semi-infinite medium

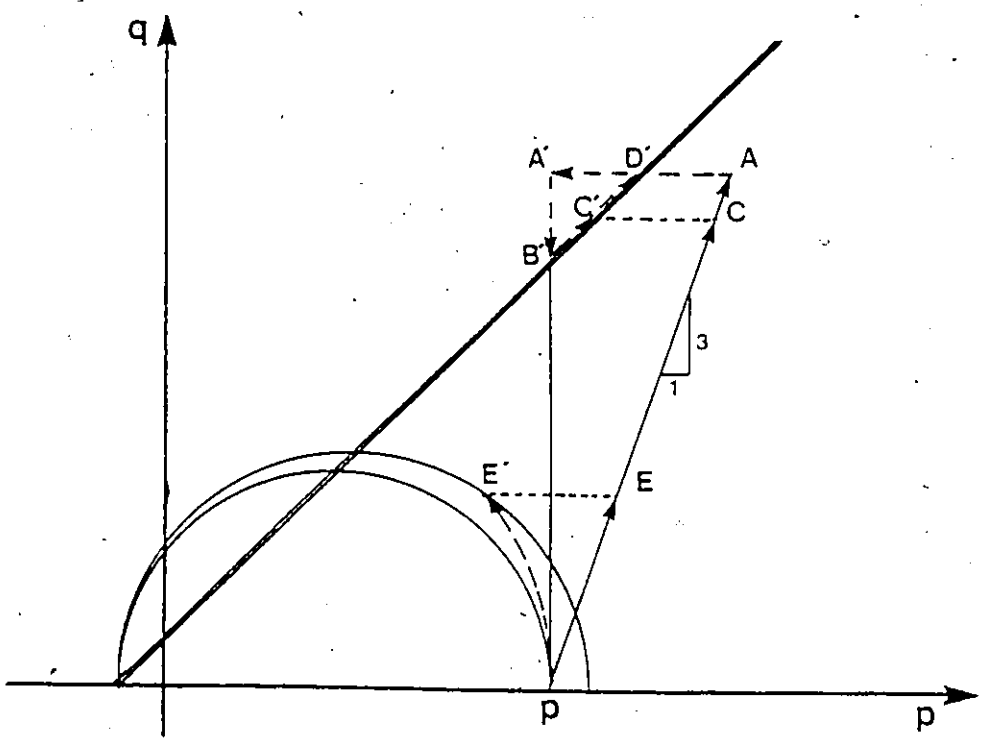
Fig. 5.23 shows the mesh layout. The unit weight of the material was assumed to be 20.0 whereas a hydrostatic

distribution of initial pore pressures was considered ($\gamma_f = 10$).

As it would be expected from the linear elastic perfectly plastic model the most dangerous conditions are met at the time of load application since much bigger plastic deformations occur at that particular instant. In the current investigation, failure occurred at a load application of 105.

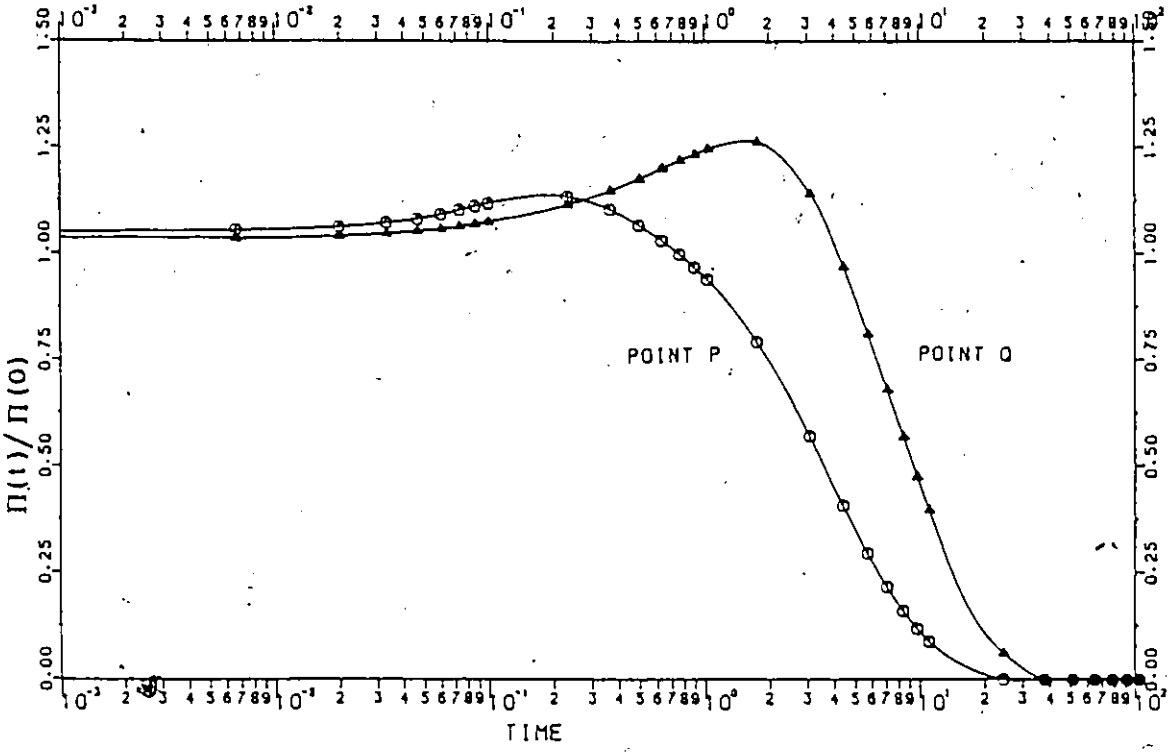
On the other hand, more interesting results were obtained for the generalized Burland model, as Figs. 6.13 illustrate. The mechanism of of plasticity can be readily explained in this case. The action of the pore pressure initially causes the effective stress point to move close to the surface, thus limiting the amount of plastic deformation. With time as consolidation proceeds, higher effective stresses are mobilized between the constituent skeleton grains, inducing larger displacements to occur. At the same time, the maximum excess pore pressure appear some instant after the application of the load. The above considerations account for the delay failure obtained in the present case shown in Figs. 6.13a,b. Also, turning to the resulting displacement field (Fig. 6.13c), most of the movements are mobilized in a confined zone below the footing, as confirmed by the previous effective stress analysis in Chapter 5.

Finally, computer-wise the generalized Burland model required 1539 CPU for the analysis of a 63-element mesh made up of 222 nodes.

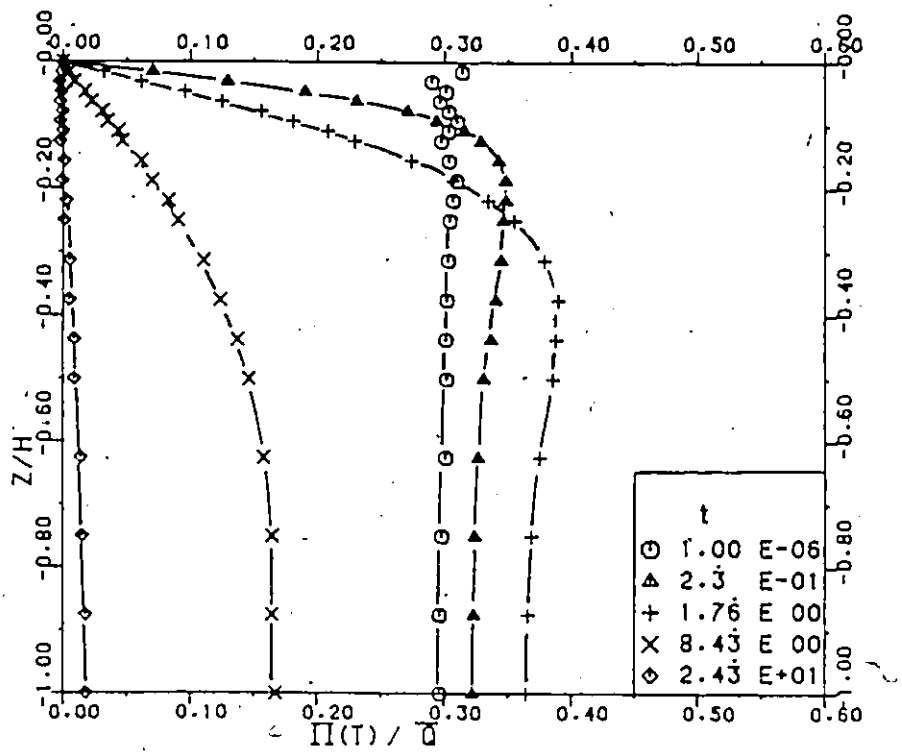


Points	p'	p	q
A		-145	135
A'	-100		135
B'	-100		110
C		-138.3	115
C'	-105		115
D'	-125		135
E'		-116.5	50
E'	-80.6		50

Fig. 6.10 Stress equilibrium in coupled elasto-plastic consolidation

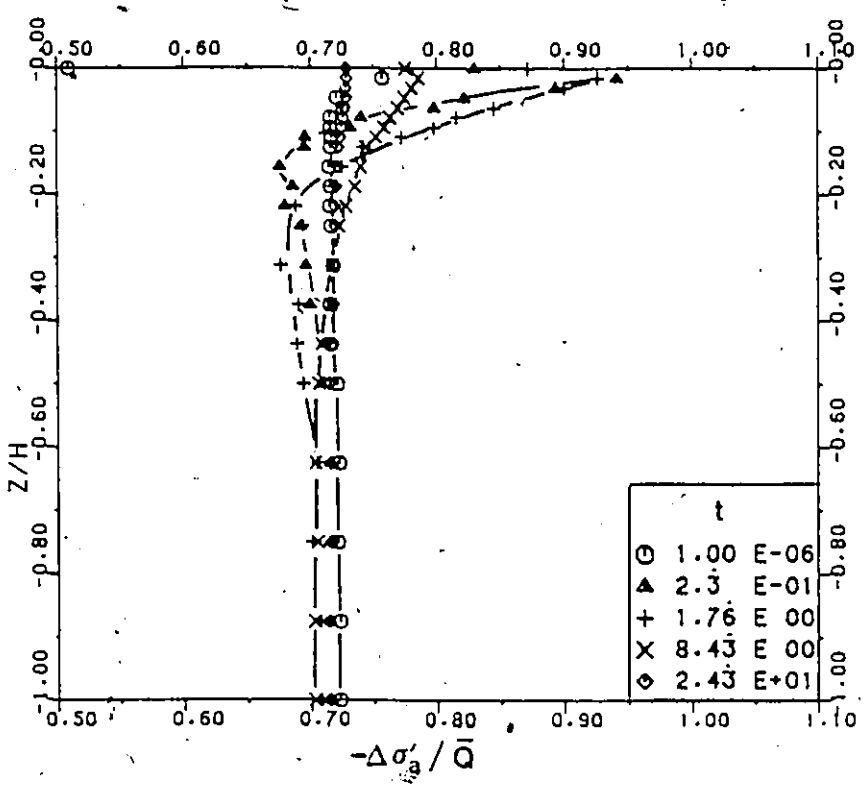


(a) excess pore pressure vs. time

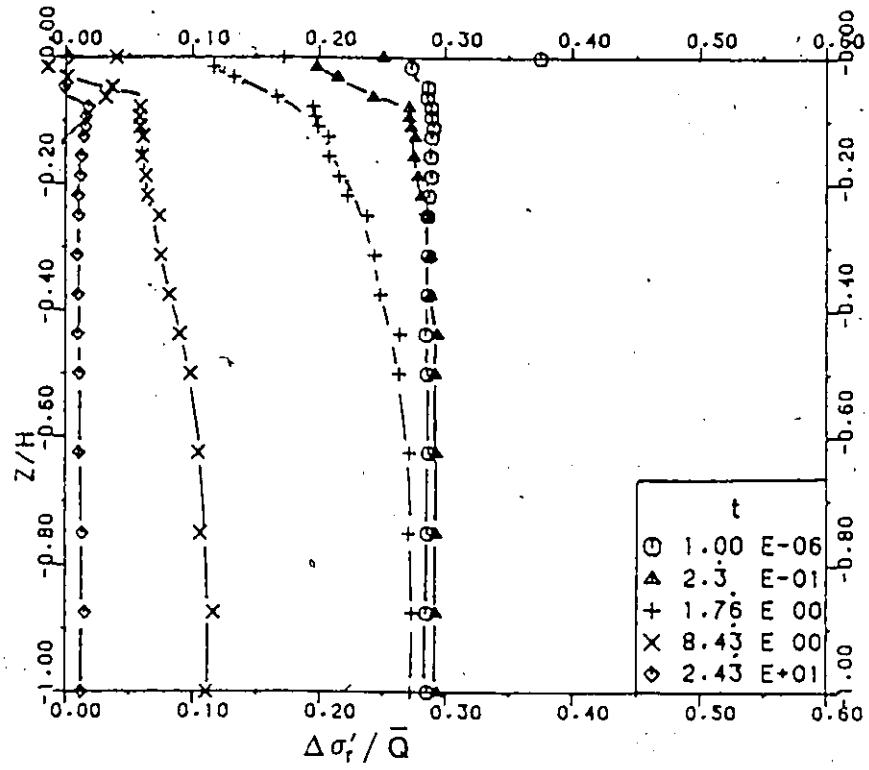


(b) excess pore pressure isochrones

Fig. 6.11 Triaxial Coupled plastic consolidation (Mohr-Coulomb material)

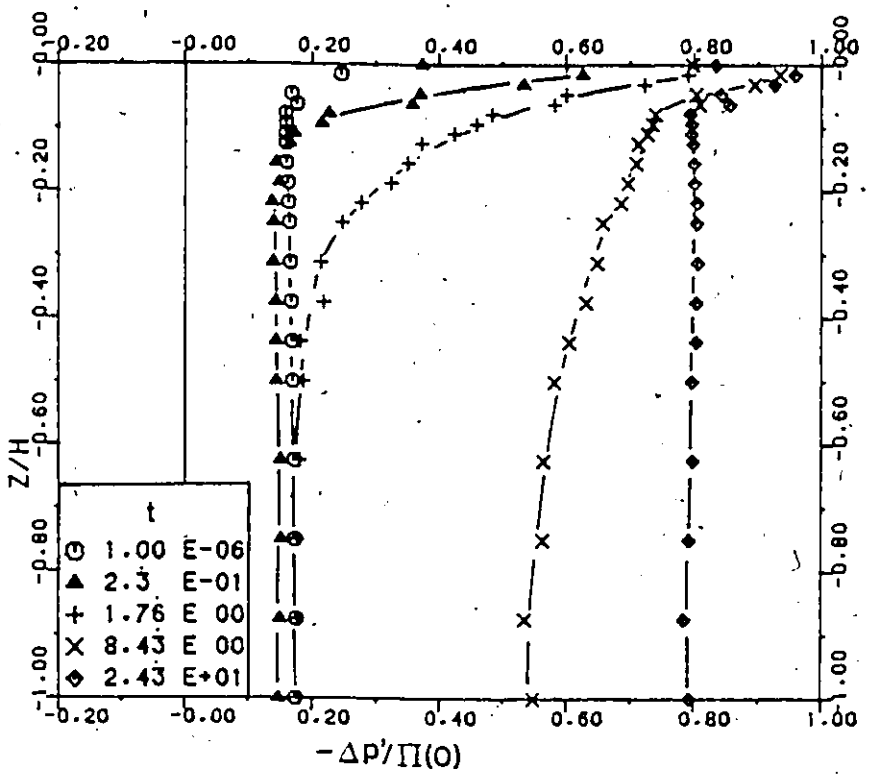


(c) axial effective stress isochrones

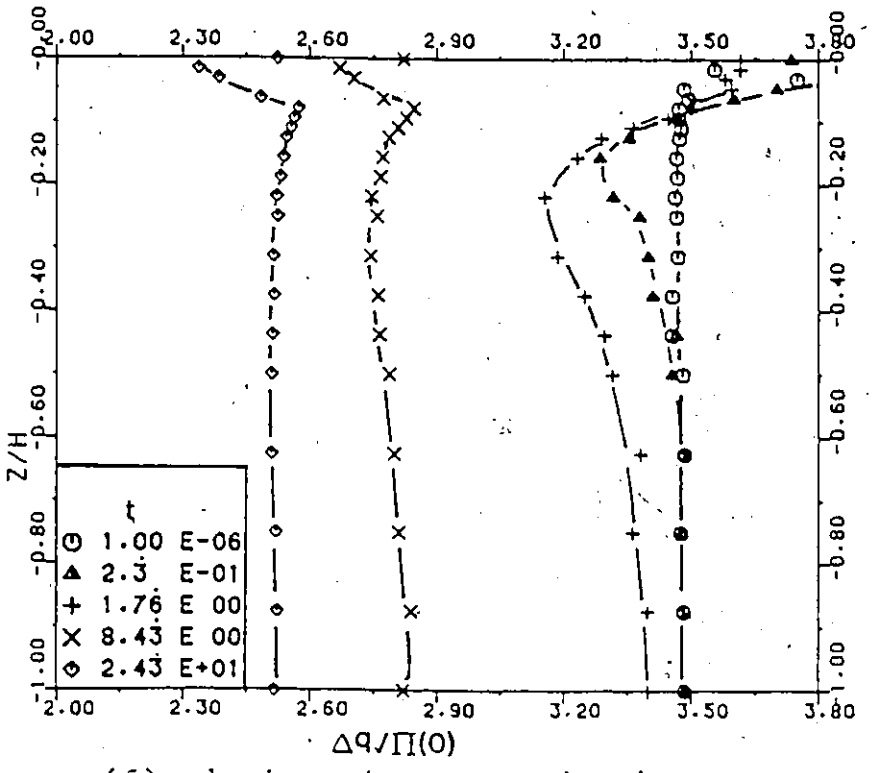


(d) radial effective stress isochrones

Fig. 6.11 Triaxial Coupled plastic consolidation (Mohr-Coulomb material)

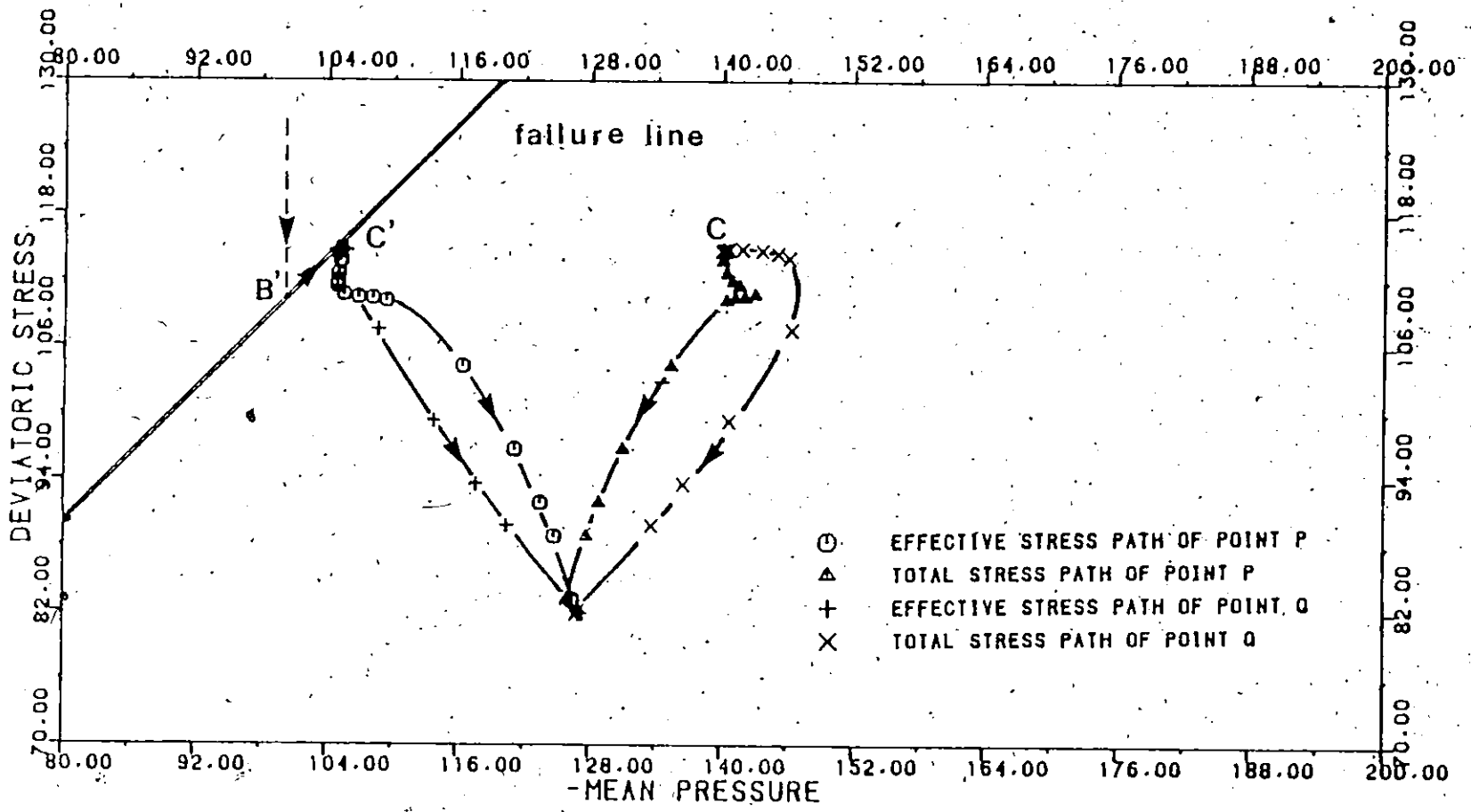


(e) effective mean pressure isochrones



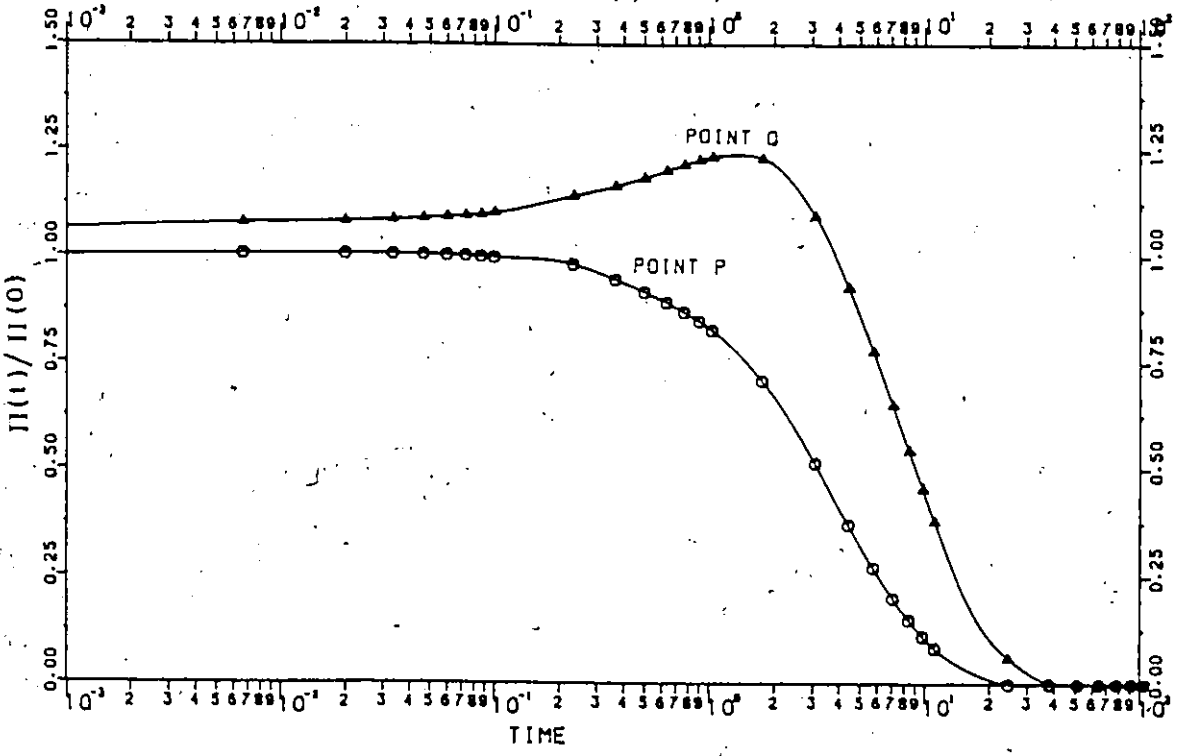
(f) deviatoric stress isochrones

Fig. 6.11 Triaxial Coupled plastic consolidation (Mohr-Coulomb material)

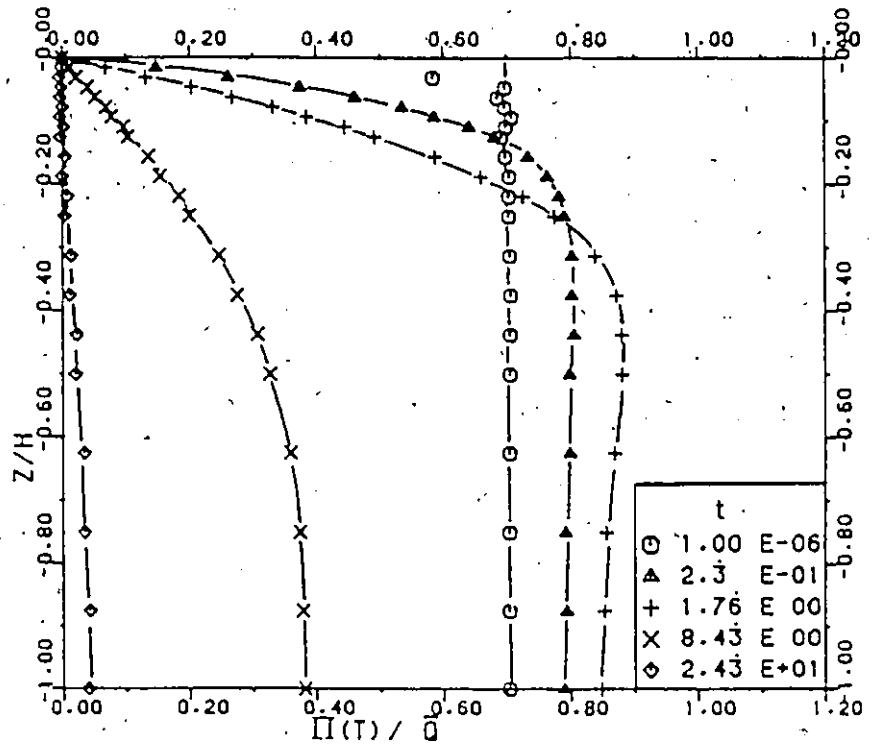


(g) stress path

Fig. 6.11 Triaxial coupled plastic consolidation (Mohr-Coulomb material)

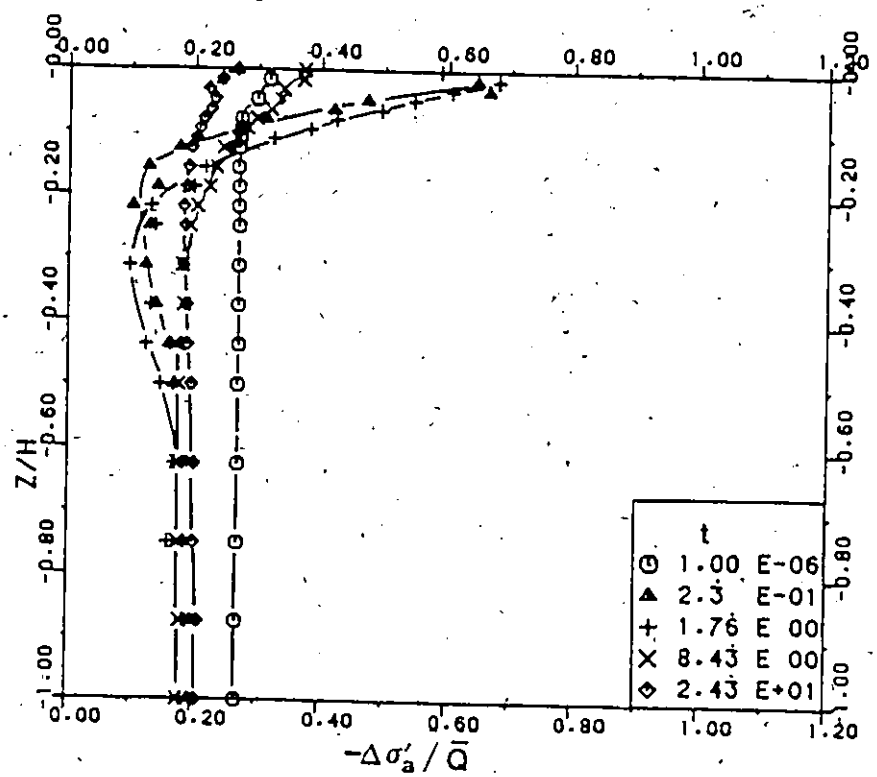


(a) excess pore pressure vs. time

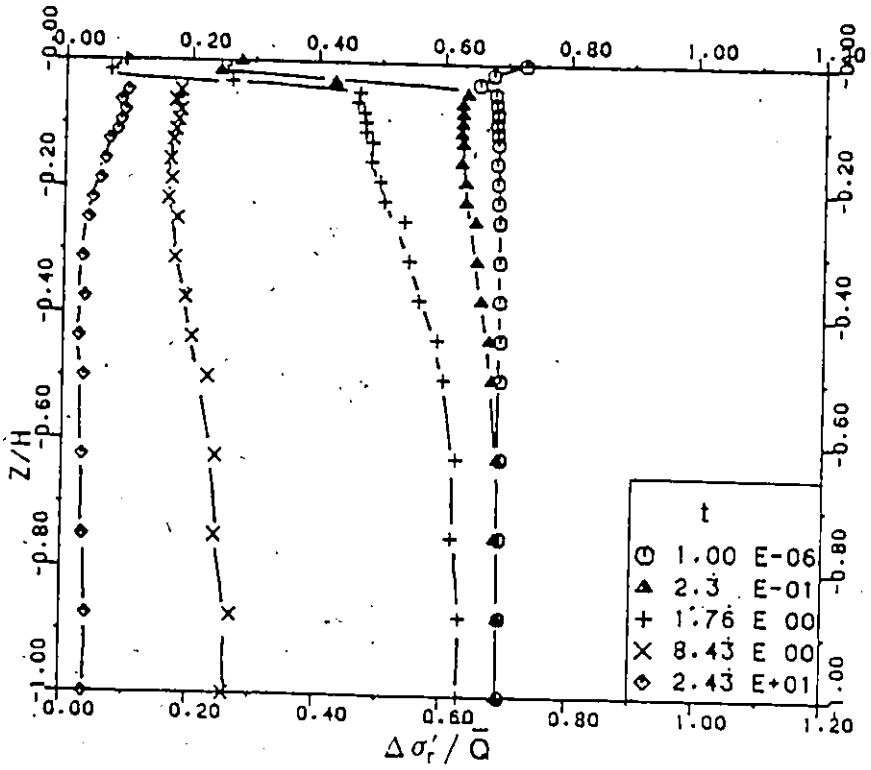


(b) excess pore pressure isochrones

Fig. 6.12 Triaxial Coupled plastic consolidation (Critical Mohr-Coulomb material)

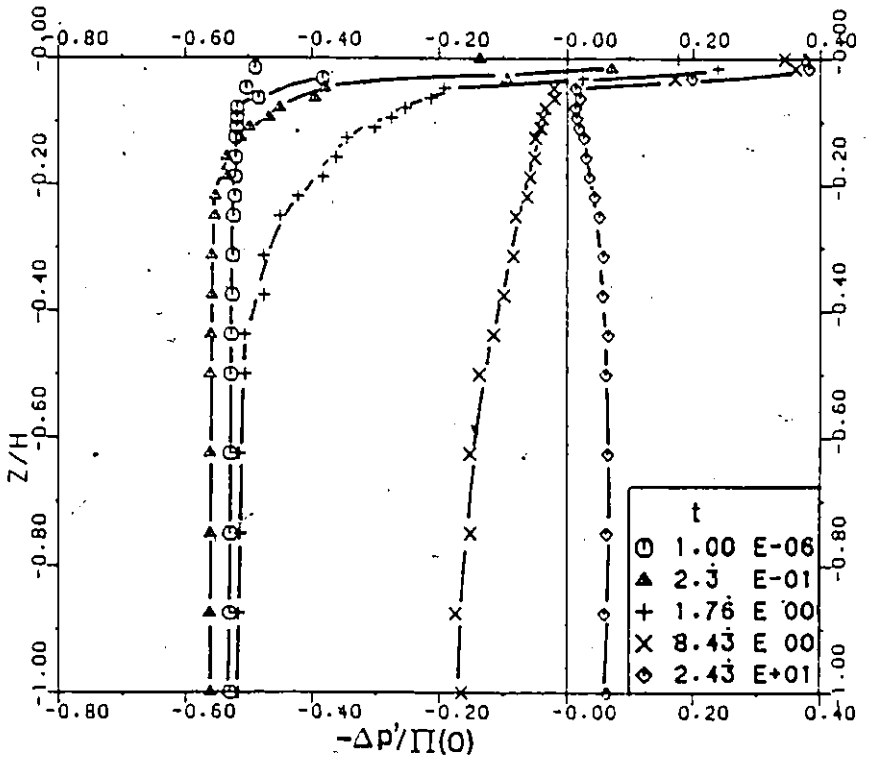


(c) axial effective stress isochrones

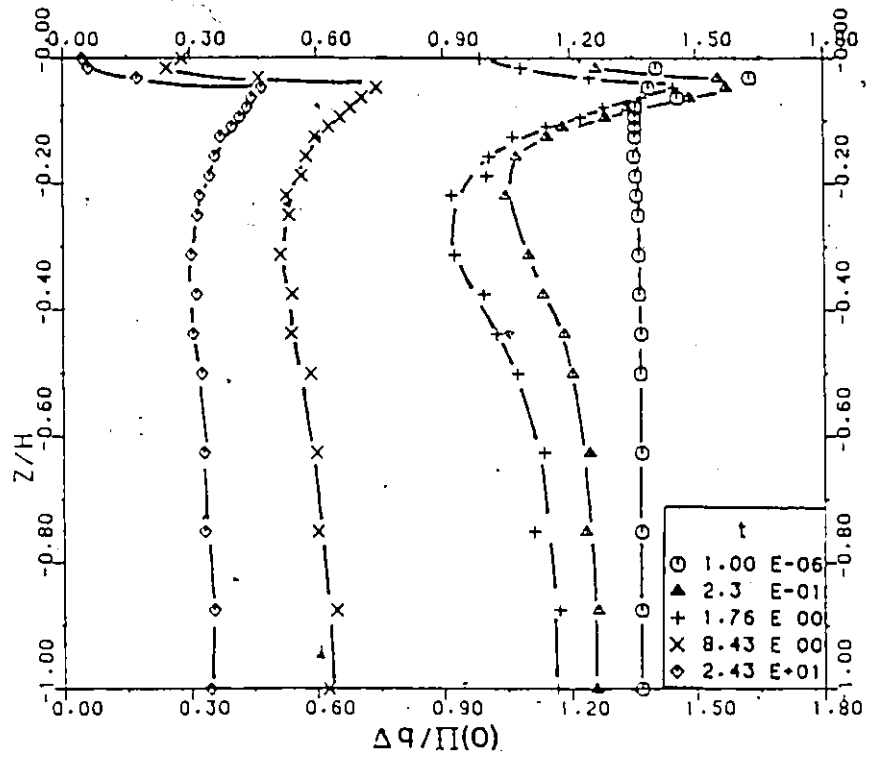


(d) radial effective stress isochrones

Fig. 6.12 Triaxial Coupled plastic consolidation (Critical Mohr-Coulomb material)

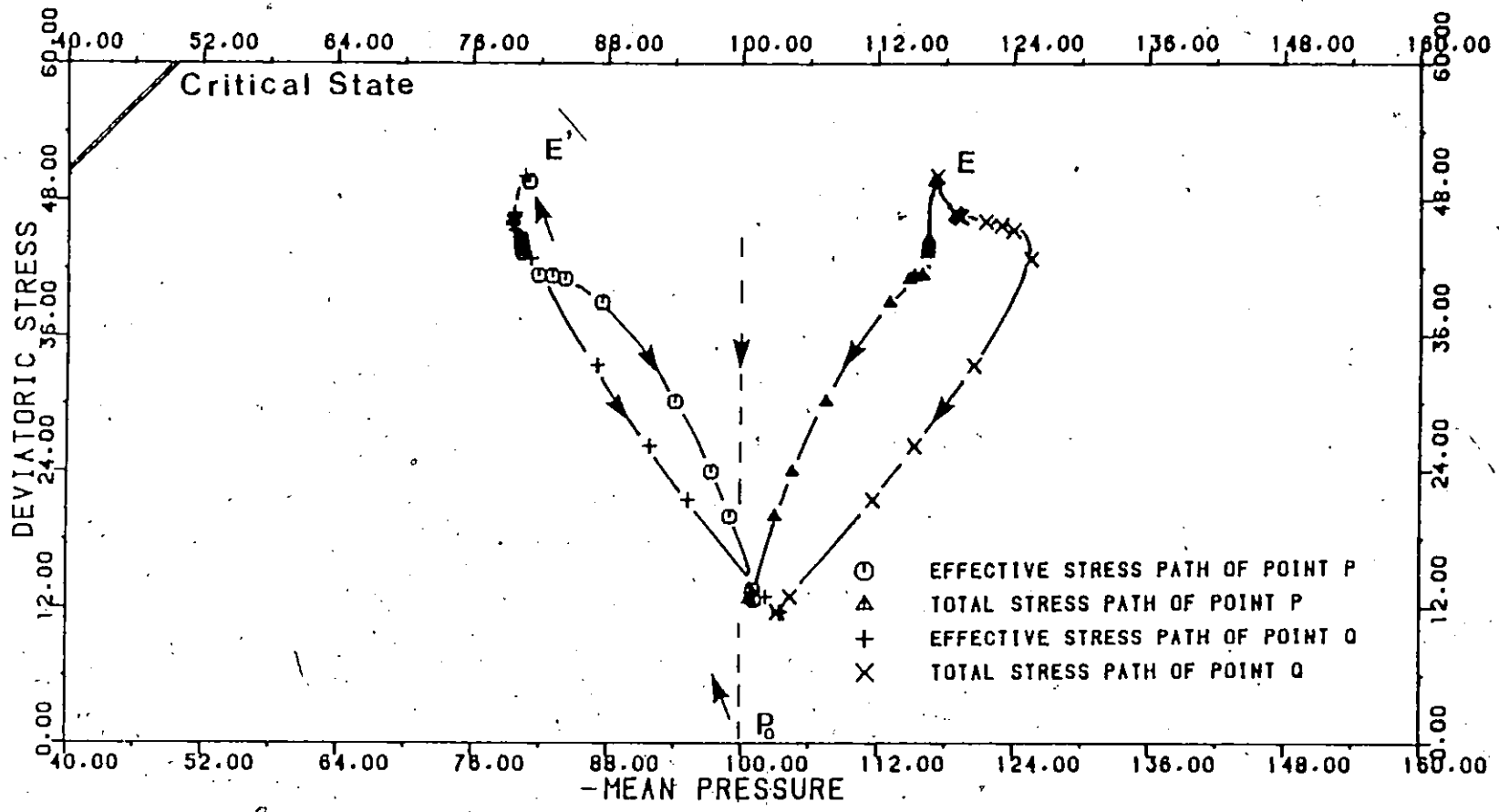


(e) effective mean pressure isochrones



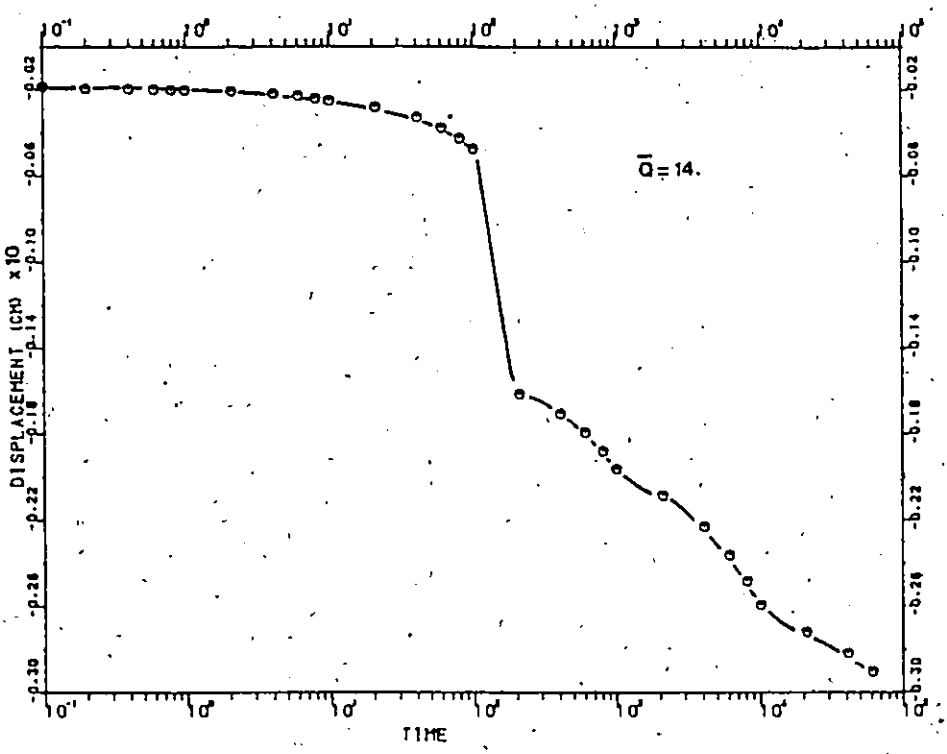
(f) deviatoric stress isochrones

Fig. 6.12 Triaxial Coupled plastic consolidation (Critical Mohr-Coulomb material)

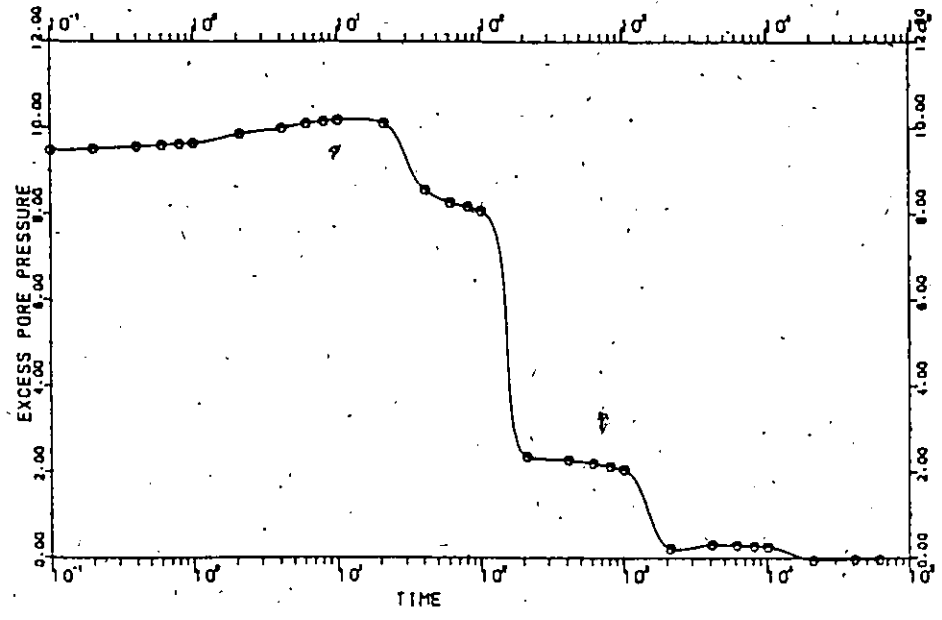


(g) stress path

Fig. 6.12 Triaxial coupled plastic consolidation
(Critical Mohr-Coulomb material)

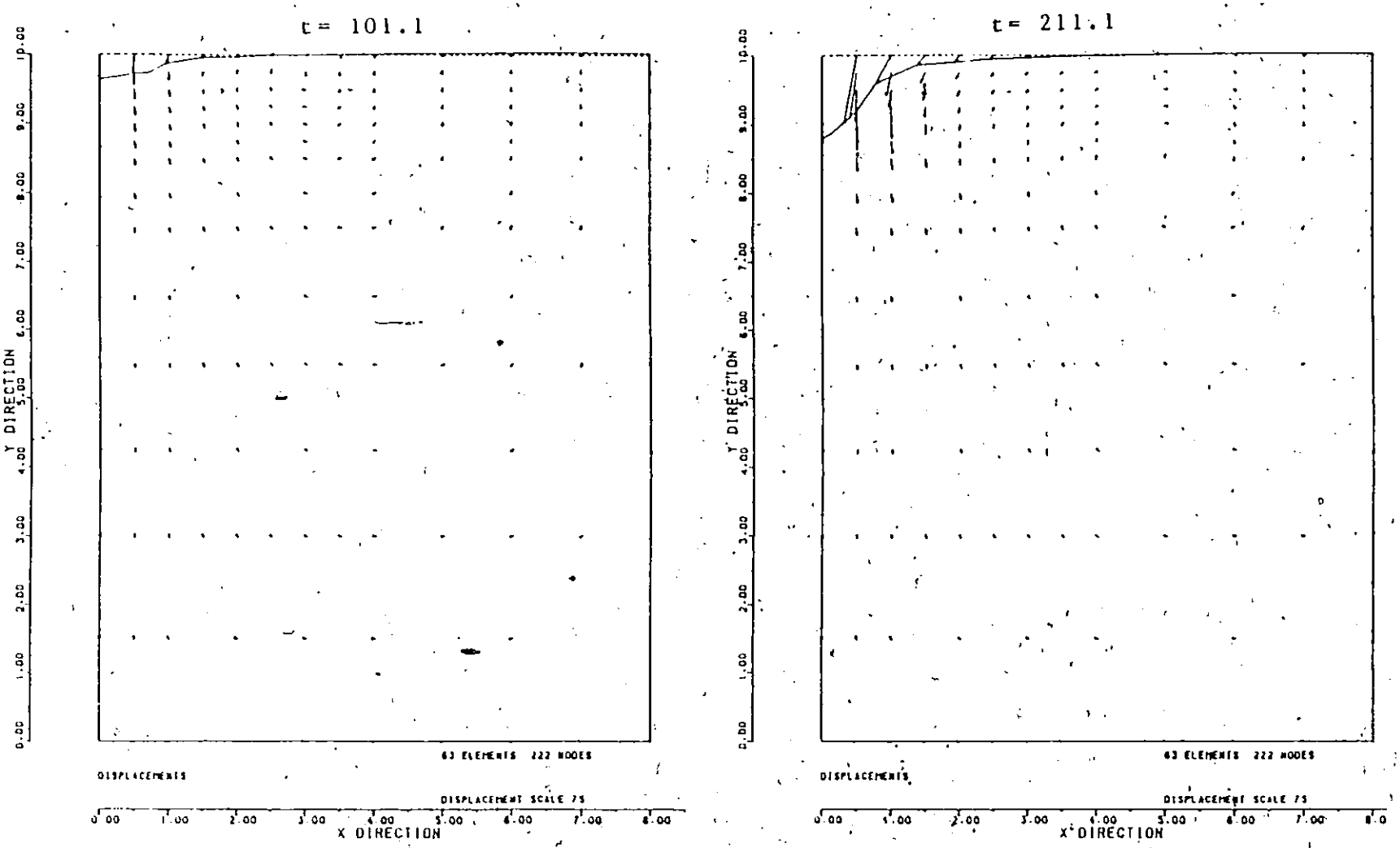


(a) surface settlement vs. time



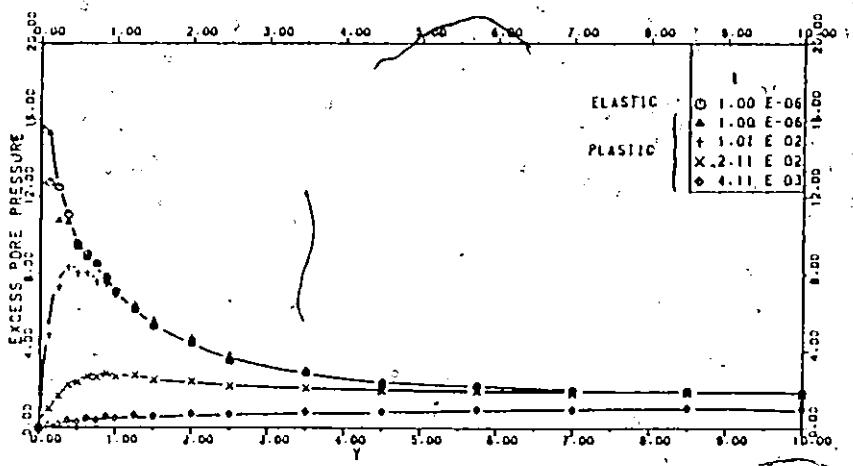
(b) excess pore pressure vs. time

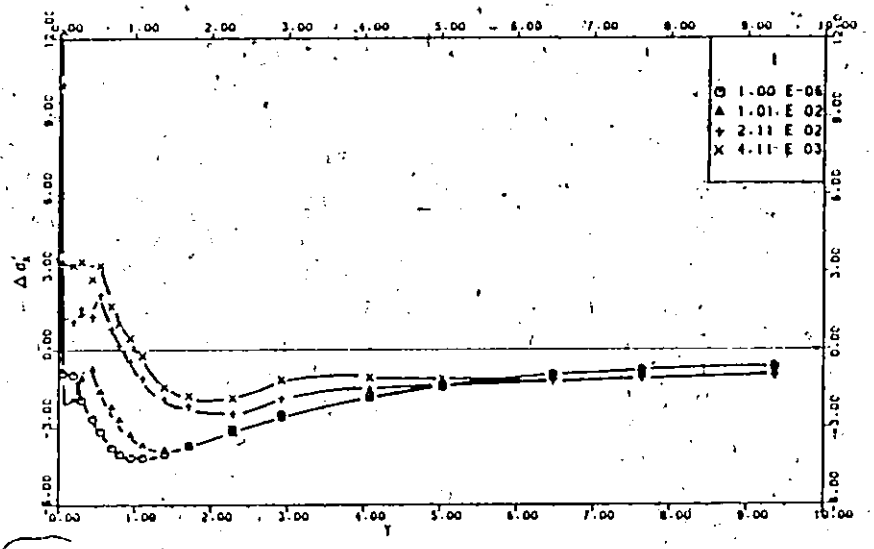
Fig. 6.13 Consolidation in a semi-infinite medium (Critical Mohr-Coulomb material)

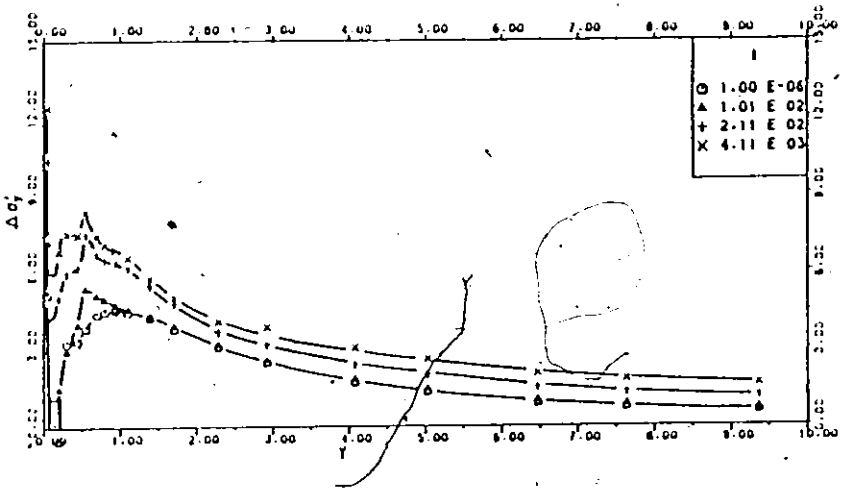


(c) displacement field at different time steps

Fig. 6.13 Consolidation in a semi-infinite medium
(Critical Mohr-Coulomb material)



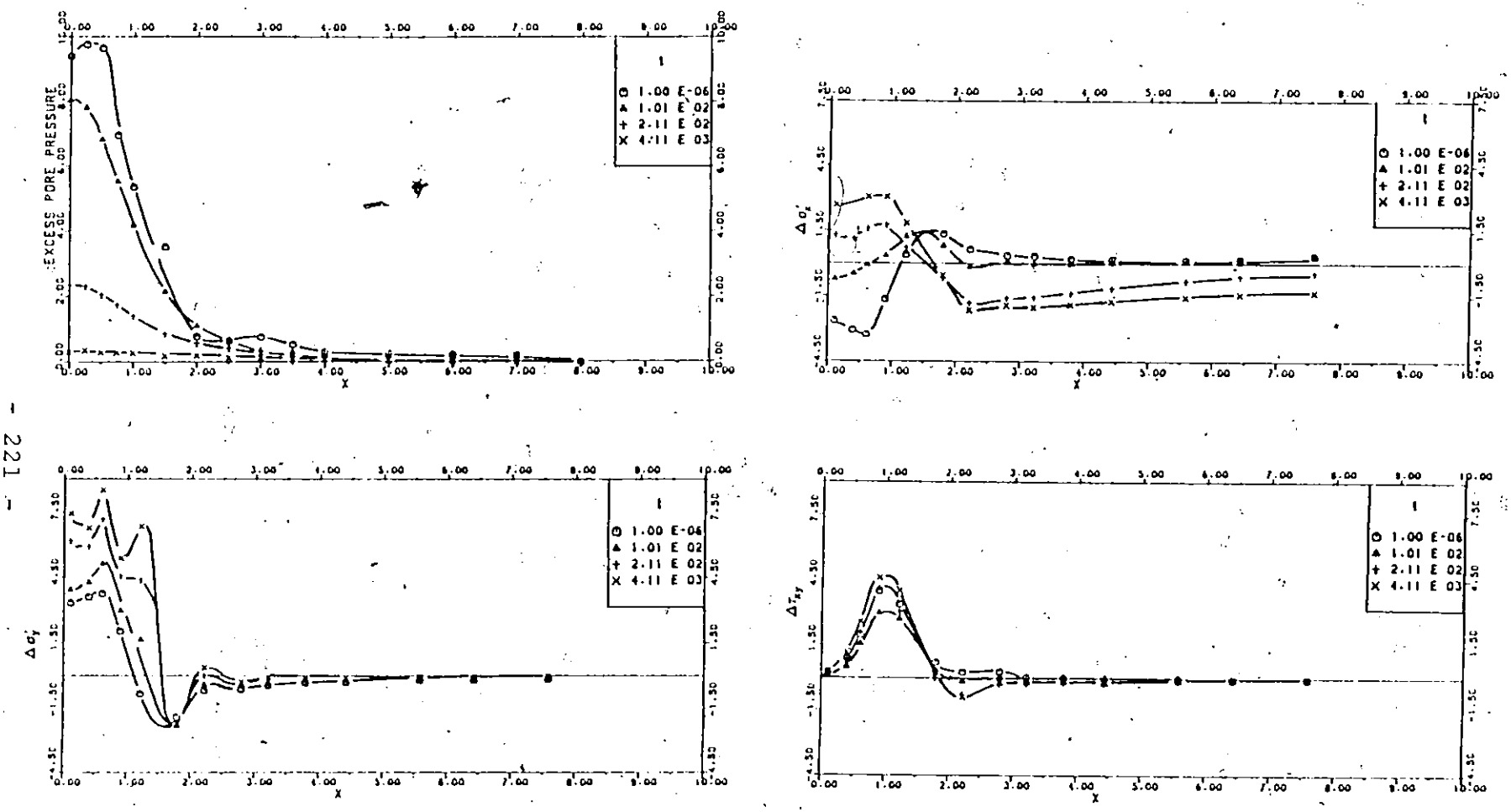




(d) excess pore pressure and effective stress isochrones along the centre line

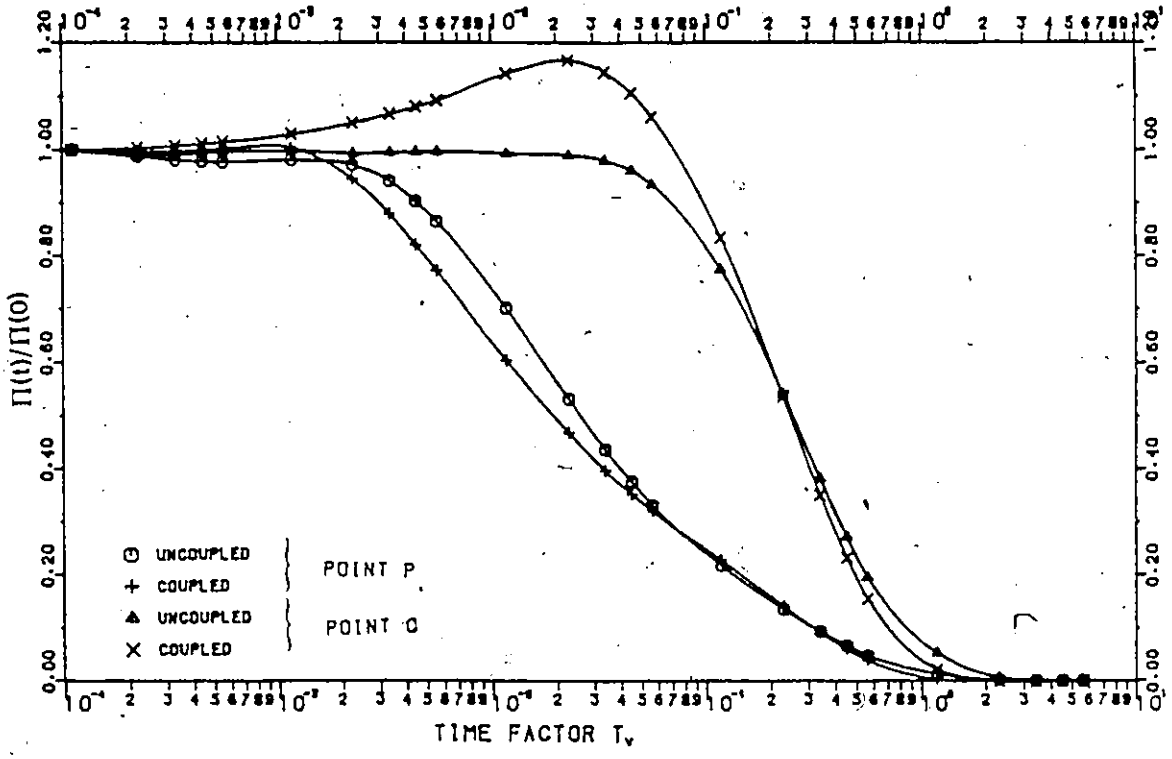
Fig. 6.13

Consolidation in a plastic semi-infinite medium (Critical Mohr-Coulomb material)

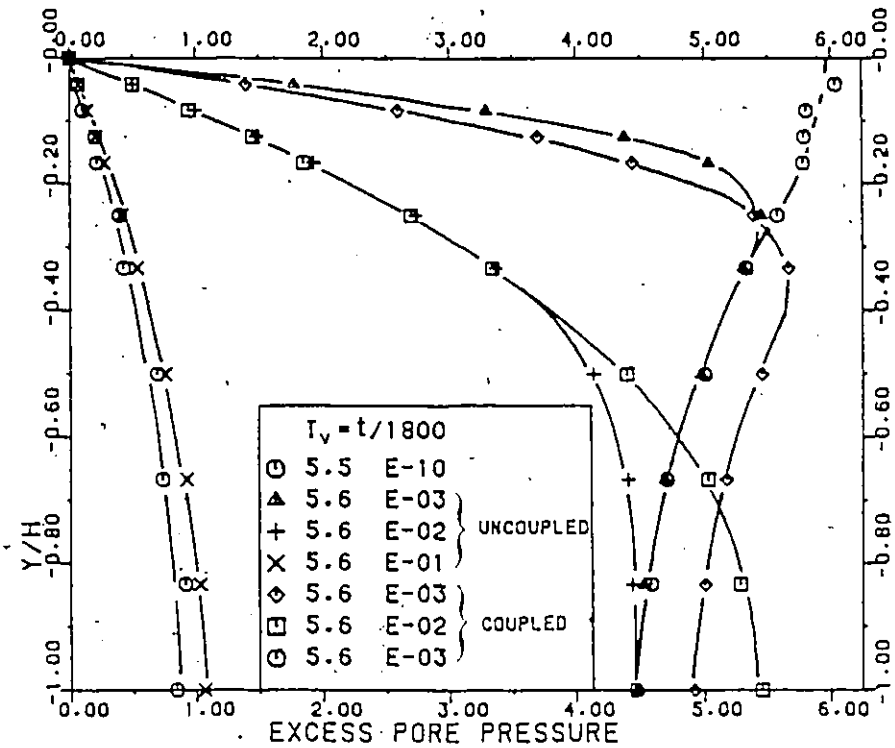


(e) excess pore pressure and effective stress isochrones (horiz. sect., Y=0.5)

Fig. 6.13 Consolidation in a semi-infinite medium (Critical Mohr-Coulomb material)



(a) excess pore pressure vs. time



(b) excess pore pressure isochrones at section A-A

Fig. 6.14 Comparison between Coupled and Uncoupled consolidation theories

6.6 CONCLUSION

The "standard" 8-node isoparametric finite elements together with a "smoothing" technique as introduced by Reed (1983) and a semi-variable time stepping such as Sandhu's discretization have been successfully used in the implementation of the coupled solid-fluid interaction problem. The "smoothing" of excess pore pressures definitely eradicates the oscillations and inaccuracies reported by Sandhu (1977). While the present analysis produces far more accurate results in the determination of both pore pressure and displacement, it is at any rate worth to emphasize that the adopted algorithm has proved its efficiency and reliability in the fundamental elastic case.

The available closed form solutions in elasticity, obtained by either transform techniques or series expansions, were closely reproduced by the F.E computations in both 1-D and 2-D situations. In axial symmetry, the numerical responses coincided with the theoretical expections. However, it was found that the computed horizontal displacements follow a peculiar trend in the elastic case study. The maximum lateral displacements are obtained at t=0 ("undrained" state); i.e when no dissipation of fluid has yet started. With time, as consolidation proceeds these displacements tend to retreat towards the original vertical boundary position. This phenomenon can be readily explained by the fact that the variation of elastic

lateral strain $d\varepsilon_x$ remains negative during consolidation. Moreover, the coupled elastic consolidation analysis gives a more rapid dissipation of excess pore pressure when confronted with the uncoupled approach (see Fig. 6.14)

When considering non-linear or elasto-plastic behaviour of the skeleton interesting findings were revealed. It was found that the type of plasticity model used had an important implication in the settlement and pore pressure behaviour. Hence, if a medium follows the linear elastic perfectly plastic model, the excess pore pressure predicted are smaller than the elastic one. The generalized Burland model contrasted with the former model by giving a higher excess pore pressure and thus lower effective stresses. It is noted that despite lower effective stresses, much larger displacements are obtained in comparison with the elastic case due to the non-linear nature of the analysis.

The two plasticity models, perfectly plastic and generalized Burland models, provided different behaviours in the consolidation process. In fact the elastic perfectly plastic model predicts an unsafe situation at the beginning of consolidation and an increase in safety factor with time. On the other hand, it appears that the generalized Burland model can adequately characterize the behaviour of a soft clay. The state of the material may well be safe at the beginning but as the excess pore pressure dissipates, failure may suddenly occur. This delay failure phenomenon was observed in the present numerical study.

It is felt that the computation described in this chapter has a potential value in the analysis of practical geotechnical problems, although further studies may be needed to improve the efficiency of the coupled elasto-plastic analysis with the generalized Burland model.

Chapter VII

PRE- AND POST- SUPPORT PROGRAMS

7.1 INTRODUCTION

Although the finite element method is both powerful and flexible, it requires extensive input data and produces as well a lot of numerical output. As such, the manual entering of the nodal coordinates and element topology is cumbersome, tedious and error-prone. In addition, the final results need to be processed for a comprehensive analysis of the numerical output.

After the geometrical input data, the mesh has to be plotted for preview, thus saving time with respect to abortive runs. It is therefore worthwhile to process and represent graphically the final results. For these purposes, three peripheral programs namely MESHGEN (MESHGENeration), ELABO (ELABOration) and FEMPLOT (FEMPLOTting) were developed and implemented.

The use of MESHGEN is two-fold; firstly it eliminates the drudgery of entering the input data manually and secondly it generates automatically the mesh in the analysis of the unconfined flow in SEEP. ELABO processes the final results from the different main programs and sets up the plotting arrays for FEMPLOT.

In addition, ELABO establishes the liaison between ESEPA and TERCON since it may also compute the excess pore pressure according to Henkel's formula from the stress calculated by ESEPA.

The interconnection between the F.E. codes and their linkage with the peripheral programs are shown in Fig. 7.1.

As illustrated, with appropriate definition of storing disk files, the main F.E. programs may intercommunicate, so that they may also be run sequentially. As a result, the pore pressure calculated in SEEP can be used as initial values for a latter stress analysis performed by either ESEPA or TEPSA. Furthermore, ESEPA calculates the initial stress distribution to be used in TEPSA.

In the remainder of this chapter the three peripheral programs are described. Herein, emphasis is laid only upon the programming structure since the resulting implementation is either highly machine facility dependent or trivial.

7.2 MESHGENERATION

During the past few years, a significant number of automatic mesh generation codes have been devised. Unfortunately, most of the available programs are quite specialized and are usually of considerable complexity.

For this purpose, it was considered most practical to develop an efficient and flexible automatic mesh generation program, MESHGEN, for parabolic 8-node isoparametric elements.

7.2.1 Procedure

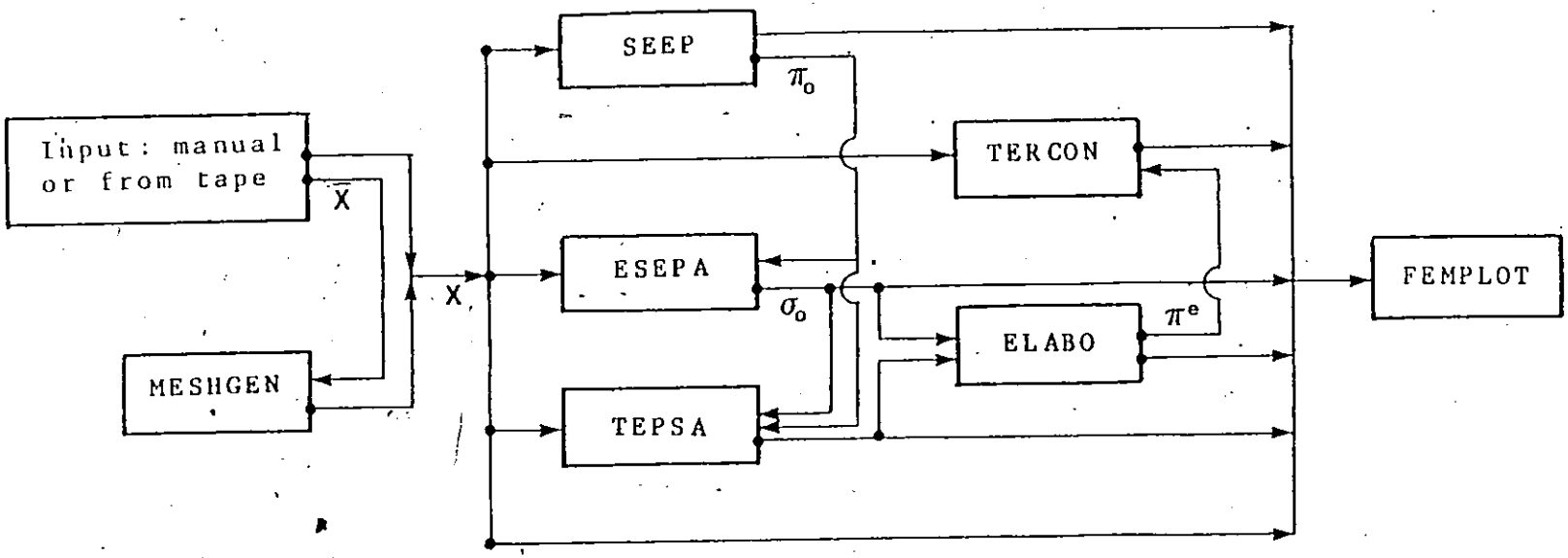
Herein, a semi-automatic approach is adopted, in which a given coarse mesh made up of so called Big Elements, BE, is further subdivided into finer elements by MESHGEN, Fig. 7.2.

Each BE is first numbered in an anticlockwise fashion starting from any corner node and subsequently refined according to 3 different types of subdivisions shown in Fig. 7.3. Associated with the first type is the subdivision of the BE into a number of 8-node isoparametric elements following the specified degrees of fineness in x and y directions. However, account is taken at this stage to accommodate the transition from finely to coarsely meshed regions by further development of the first type. Finally, the third type has been implemented to cope with situations where boundaries contain sharp corners, In this case, the BE is defined by a 6-node isoparametric triangular element.

The basic concept of mesh generating is the establishment of a certain regular subdivision in the local reference system (ξ, η) of the element, to be subsequently mapped onto the real configuration.

A comprehensive description of this technique is covered by CHEUNG, 1979. As a matter of fact, the mesh generation routine for the first type of subdivision is basically that published by the same author, whilst the two other types, instead, required a careful and tedious adaptation.

In addition, the individual BE had to be assembled "node-wise" and "coordinate-wise" in order to generate the entire mesh. The structure of the program, as such, is essentially made up of 3 important phases, Fig. 7.4. Firstly, the elements are generated in Subroutine GENTRE and then transferred to the global array in Subroutine COPY. Since the nodes are defined more than once along common boundaries of adjacent BE, it is necessary to reorder the generated nodes by Subroutine REDEF. Finally, the coordinates and element node definitions are printed in Subroutine OUTPUT. When MESHGEN was coupled with the analysis of unconfined flow, it was found necessary to introduce an additional Subroutine DUMMY. In fact, the coordinates of the BE next to the last dynamic domain (Chapter 2) are stored in temporary arrays, for later recovery. This is done to eliminate inherent disorders caused during mesh generation. The next BE is then generated till the whole domain is built up.



MESHGEN : MESh GENeration
SEEP : SEEPage analysis
ESEPA : Effective Stress Elasto-Plastic Analysis
TEPSA : Total Elasto-Plastic Stress Analysis
ELABO : ELABOration numerical results
FEMPLOT : FEMPLOTting

Fig. 7.1 Layout of program inter-connections

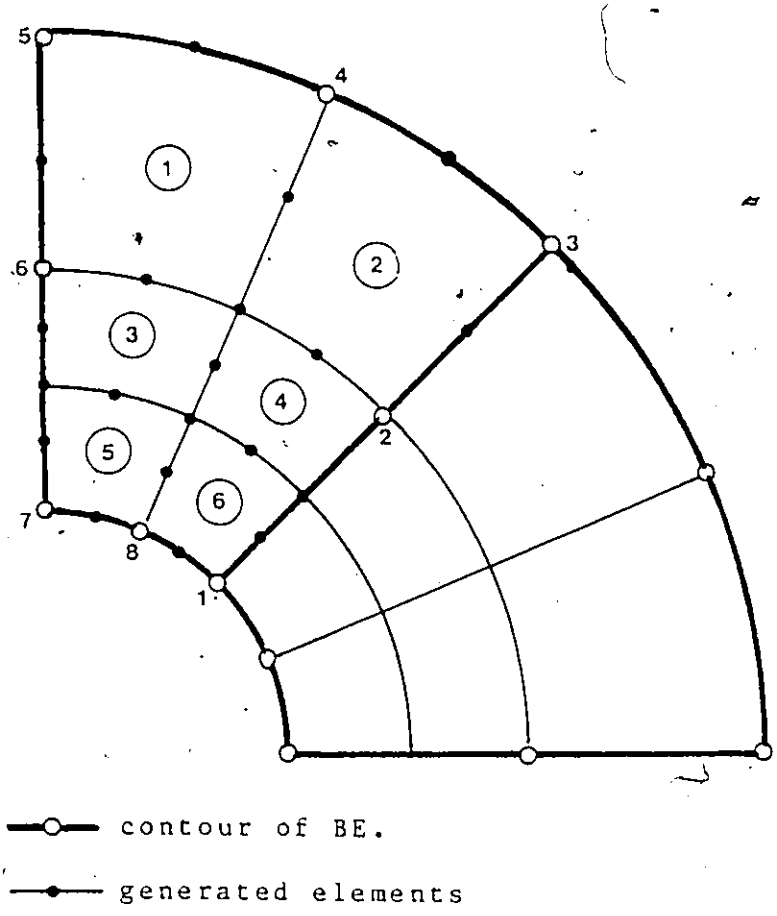
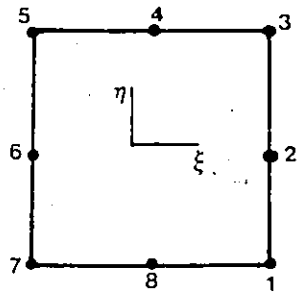
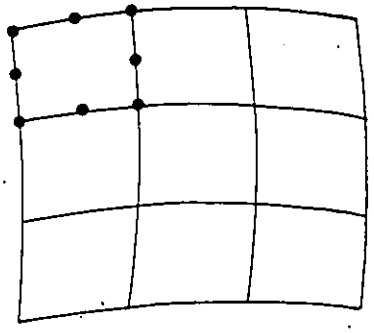


Fig. 7.2 Representation of a structure by BE.

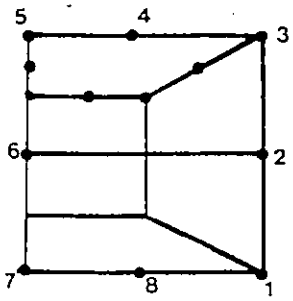
local configuration

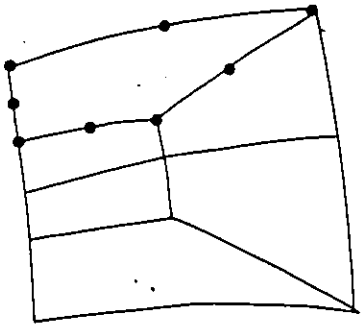
global configuration



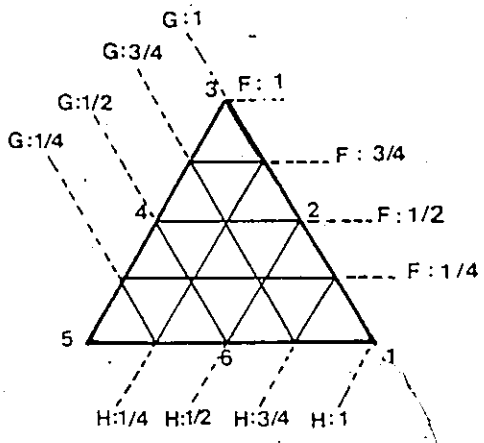


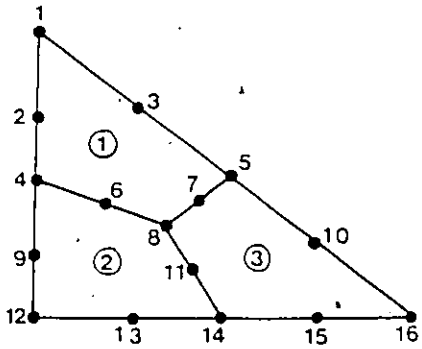
Type 1 "basic" B.E





Type 2 "transition" B.E





Type 3 "triangular" B.E

Fig. 7.3 Representation of different types of meshes generated

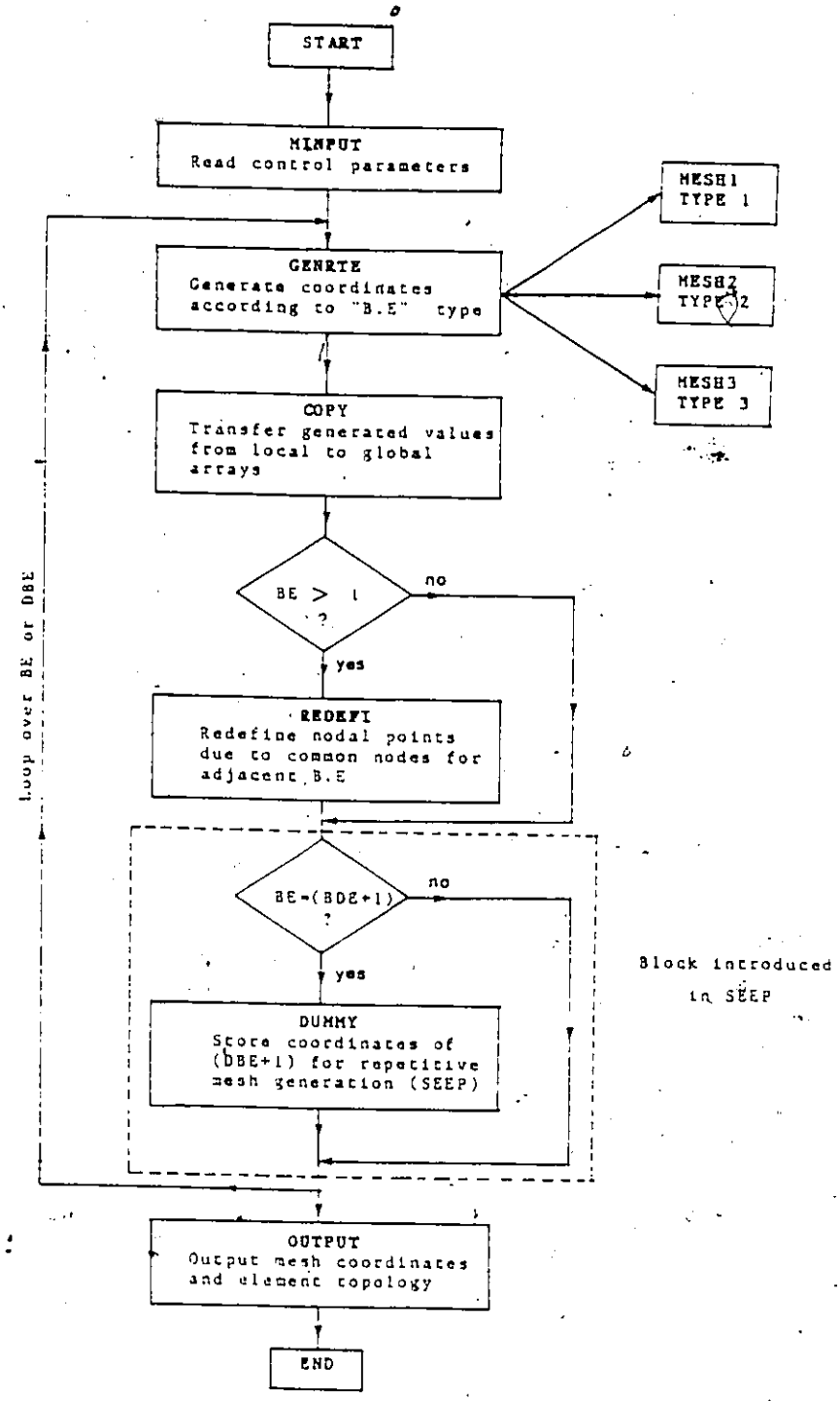


Fig. 7.4 Flowchart MESHGEN

7.3 ELABORATION

The F.E. technique, as previously described, leads to the calculation of displacements at the nodes and stresses at the Gauss integration points. A comprehensive analysis, instead, requires the field variables to be computed at the very same geometrical point. Also, in the analysis of a saturated medium, it is usually desirable to determine the regime of flow.

Following these aspects, ELABO permits the calculation of displacements, stresses, pore pressures and fluid velocities at both nodal and Gauss integration points. In addition, the program computes the deformed configuration, the principal stress directions as well as the safety factor at the considered point.

In particular, the nodal stresses are obtained from the values at Gauss integration points by means of the same smoothing technique used in SEEP for the calculation of the nodal fluid velocities.

On the other hand, adequate use of shape functions leads to the determination of displacements and pore pressures at Gauss integration points as follows:

$$f(\xi,\eta) = s_i(\xi,\eta) \cdot f_i \qquad (7.1)$$

where ξ, η are the local coordinates of the Gauss point, while f_i is the discrete value of any field variable at the nodal point i.

As regard hydraulic variables, the velocities at Gauss integration points are computed in a similar fashion as in SEEP, namely:

$$\underset{\sim}{v} = - K \cdot \text{grad } h \tag{7.2}$$

$$\text{where} \quad h = h_o + \frac{\pi^e}{\gamma_f} \tag{7.3}$$

The conversion to nodal velocities is subsequently performed according to the now familiar "smoothing" technique.

The safety factor against plastic yield is herein defined as the ratio of the current octahedral stress to the limiting value calculated on the failure surface for the same angle θ and mean pressure p, Fig. 7.5.

$$S.F = \frac{\text{Limiting } \tau_{oct}}{\text{Current } \tau_{oct}} \tag{7.4}$$

Finally the principal stress directions are determined according to the following equations:

$$\sigma_{max} = \frac{\sigma_x + \sigma_y}{2} + \sqrt{\left[\frac{(\sigma_x - \sigma_y)^2}{4} + \tau_{xy}^2\right]} \qquad (7.5)$$

$$\sigma_{min} = \frac{\sigma_x + \sigma_y}{2} - \sqrt{\left[\frac{(\sigma_x - \sigma_y)^2}{4} + \tau_{xy}^2\right]} \qquad (7.6)$$

$$\alpha = \frac{1}{2} \tan^{-1} \left(\frac{2\,\tau_{xy}}{\sigma_x - \sigma_y}\right) \qquad \sigma_x < \sigma_y \qquad (7.7)$$

$$\alpha = \alpha + 90° \qquad \sigma_x > \sigma_y \qquad (7.8)$$

where α is the angle which the maximum principal stress makes with the vertical.

With reference to the flowchart of ELABO in Fig. 7.6, the main control parameters, program options and data are first read in Subroutine INPUT and then echoed in OUTPUT1. In order to analyse the deformation field, the original nodal coordinates are incremented by the corresponding factored

displacements and subsequently stored in disk files for plotting purposes. This is performed in Subroutine DEFORM.

The safety factor against plastic yield and the principal stress directions at each discrete Gauss integration point are determined in Subroutines FACTOR and STANGL respectively. Subroutine SMOOTH, calculates the nodal stresses on the basis of the Gaussian point values.

For a comprehensive graphical representation of F.E. results, it is usually desirable to plot field variables along given profiles. For that purpose, Subroutine PROF selects the nodal points and corresponding field variables to be plotted.

When elaborating results coming from a coupled solid-fluid analysis, the fluid velocities are calculated in Subroutine HYDRO and stored in files. Subroutine OUTPUT prints out the elaborated data.

It is to be noticed that the elaboration process is repeated for each time step and /or load increment.

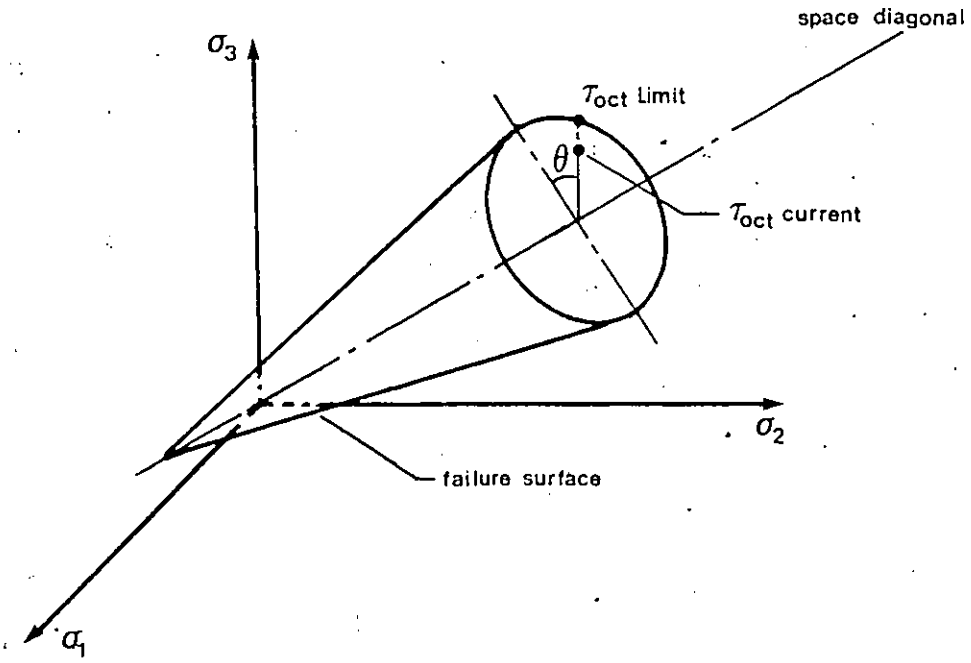


Fig. 7.5 · Geometrical representation of safety factor in 3-D stress space

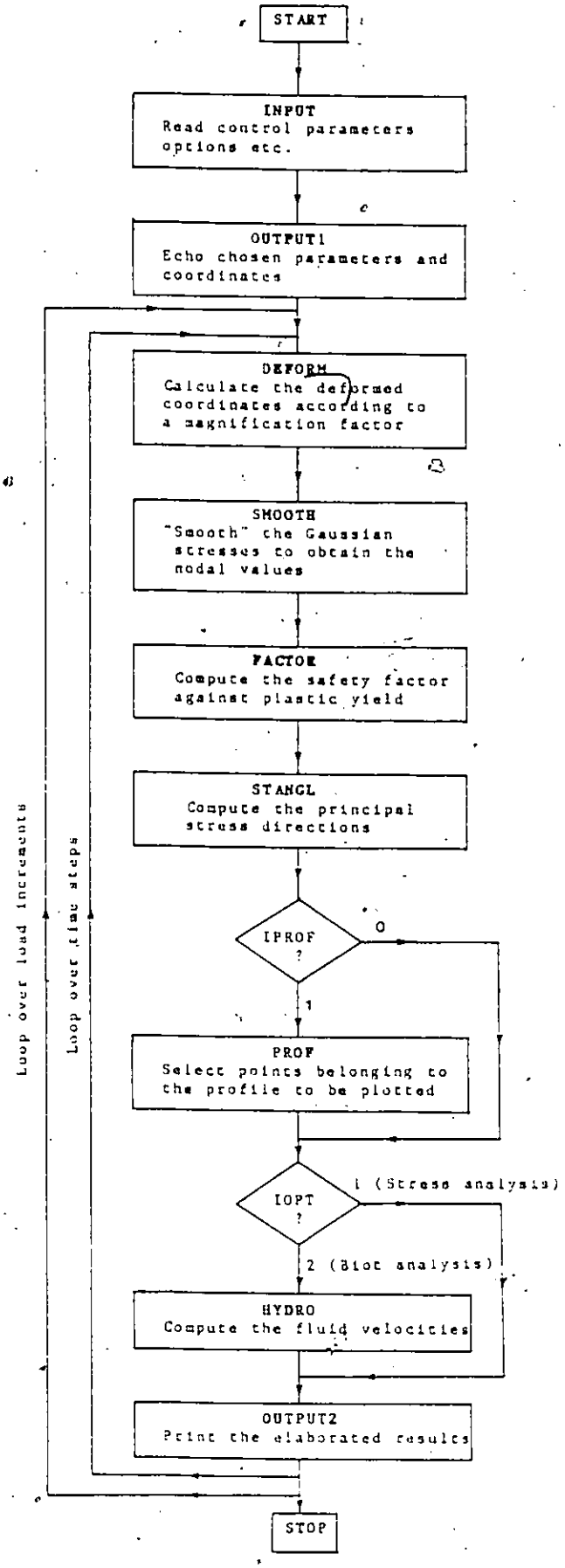


Fig. 7.6 Flowchart ELABO

7.4 FEMPLOT

FEMPLOT produces the graphical representation of the numerical results calculated by any of the four main F.E. programs which have been eventually processed by ELABO.

The items to be plotted depend on the class of problem being analysed and may be broken down under the following headings:

1. Stress analysis of highly permeable soils (ESEPA).

   This requires the representation of the initial and deformed meshes, the principal stresses, the safety factors and the displacement field.

2. Seepage analysis and uncoupled consolidation (SEEP and TERCON).

   In this case, besides the initial and deformed meshes, the hydraulic components such as head pressures, pore pressures and velocities at either Gauss integration or nodal points have to be represented.

3. Coupled consolidation (TEPSA).

   Herein, all the field variables pertaining to the two previous headings need to be graphically represented.

As previously mentioned, the FEM analysis involves enormous amount of output which, due to economical reasons and for a comprehensive representation, necessitates the

selection of particular time or load steps for plotting. In this regard, it was necessary to suit these specific needs by writing a plotting program since no compatible software was available. In fact, these types of programs involve graphical functions highly dependent on the machine facilities, and only the structural organization is important to be reported.

7.4.1 Program structure

The class of problem being analysed is monitored by the indicator ITYPE, namely:

```
ITYPE = 1 ESEPA
        2 SEEP or TERCON
        3 TEPSA
```

The selected plots within each chosen increment (time or load) are indicated by the following arrays:

ICHOSE(IINCS) = IPICK	containing the chosen increment "IPICK".
IPLOT(IINCS,IGRAF) = IOPT	containing the selected plots per increment "IOPT".

where:

IINCS = 1, NINCS

NINCS is the total number of time or load increments to be plotted.

```
IGRAF = 1 mesh plotting
        2 principal stresses
        3 safety factor
        4 displacement vectors
        5 velocity at Gauss points
        6 velocity at nodal points
        7 head pressures
        8 pore pressures
```

Finally the indicator IFLAG permits the choice of plotting the initial and deformed meshes at each increment in the following way:

IFLAG = 1 only initial mesh
2 current deformed mesh.
3 initial and current deformed mesh

The flowchart in Fig. 7.7 illustrates the progress of calculation in FEMPLOT. The main data and options are read in MAINPT, after which a loop on the desired total number of plots starts. The number of loops is controlled by the variable NPLOTS which can take different values depending on IFLAG, as follow:

NPLOTS =	1	if IFLAG = 1
	NINCS	if IFLAG = 2
	NINCS +1	if IFLAG = 3

At each time or load increment, all the necessary information is selected according to control variables IFLAG and IPLOT in Subroutine PICK. This information is then printed out in Subroutine OUTPUT for control. Subroutine MESHPL then draws the deformed or undeformed mesh according to IOPT and IPLOT.

In the next section of the program the indicator, ITYPE, determines the kind of variables to be plotted in the context of the class of problem being analysed. IOPT controls the items to be plotted.

The program, in short, consists of a series of routines whose functions are to plot and select data from disk files. Detailed description of these features is given in the next sections and probably deserves careful attention.

7.4.2 Subroutine PICK

With reference to Fig. 7.8, the main function of the subroutine is to transfer all information, pertaining to a selected time or load increment, from disk files to the appropriate plotting arrays. Since the information is stored sequentially on these disk files, it is necessary to scan them until the desired set of record length is found.

The crucial point in this subroutine is, therefore, the determination of the starting position of the pointer, IBEGIN, from which the scanning process must start. To plot only the initial mesh, IFLAG=1, this merely necessitates the transfer of the undeformed coordinates to the corresponding plotting arrays. In any other cases (IFLAG > 1), IBEGIN is set according to the position of the last picked step and the procedure is self-explained in Fig. 7.8.

7.4.3 Plotting subroutine

The actual graphical representation of the F.E. results comprises two basic functions namely the drawing of arrows for vectorial quantities, and the printing of numerical values for scalar quantities, at selected geometrical points. In this context, vectorial quantities such as velocities, displacements and principal stresses are commonly represented by arrows in both magnitude and direction, while scalar values head pressure, pore pressure and safety factor are simply illustrated by their numerical values.

The structure of FEMPLOT is built in such a way that the principal stresses, displacements and velocities at either Gauss points or nodes are plotted by the same Subroutine STVEDI (STress VElocity DIsplacement). On the other hand, the plotting of safety factors, head and pore pressures is performed by the common Subroutine SFPLOT.

However, both subroutines are structured in similar way, Fig. 7.9. In fact, the section common to both of them deals with the framing of the plotting surface, the writing of the title, the drawing of the boundaries and the selection of the points to be plotted. These are done sequentially by Subroutines FRAME, TITLE, BPLOT and CHOOSE.

The difference between the two subroutines only lies in the final part where either an arrow or a number has to be plotted.

As previously mentioned, the number of points to be plotted in the analysis may also inhibit the clarity of the graphical representation. It is therefore necessary to select only a few of them for convenience. This is performed in Subroutine CHOOSE.

This subroutine has the option to choose either from the total number of nodal or Gauss integration points. The structure in both cases is the same, the aim being to construct an array containing all the selected points, namely:

KCHGAS (ICHGAS) = chosen Gauss point

or

KCHNOD (ICHNOD) = chosen nodal point

where:

ICHGAS varies from 1 to NCHGAS
NCHGAS is the total number of selected Gauss points

or NCHNOD runs from 1 to NCHNOD
NCHNOD is the total number of selected nodes.

The selection of the Gauss point (or node) is accomplished by specifying for each element (or node), the number of points to be plotted by means of the following array:

LPLOC (IELEM) = NCOPT

where

NCOPT = 0 no Gauss point to be plotted
1 only 1 Gauss point --> 4 integration
4 all four Gauss points --> points
5 five Gauss points --> 9 integration
6 only mid-Gauss points --> points

Similarly, the nodal point selection is specified as follows:

LPLO(INODES) = NCOPT

where
NCOPT = 0 discard the current 'INODE'
> 1 keep the current 'INODE'

The actual construction of array LPLO(I) is controlled by the combination of two indicators IOCH and NCHA, Fig. 7.10a. If both IOCH and NCHA are equal to zero NCOPT is set by default to 4 or 5 according to the Gaussian integration scheme. Instead, if IOCH =,0 and NCHA equal to the number of changes per element to be made, the program requires the specification of the actual element, IELEM, and the option code, NCOPT, to be used.

In the case IOCH = 1, the array LPLO(IELEM) is first initialized to zero. Then on the basis of the number of desired changes, NCHA, the element and option code are introduced so that the array LPLO (IELEM) may be rectified. Notice that the combination, IOCH= 0 and NCHA= 0, is most suitable when few modifications are needed since the array LPLO(I) is mostly constructed by default.

As regards the selection of nodal points, the procedure is similar to that just described and merely involves the substitution of variable IELEM by INODE. The structure is illustrated in Fig. 7.10b.

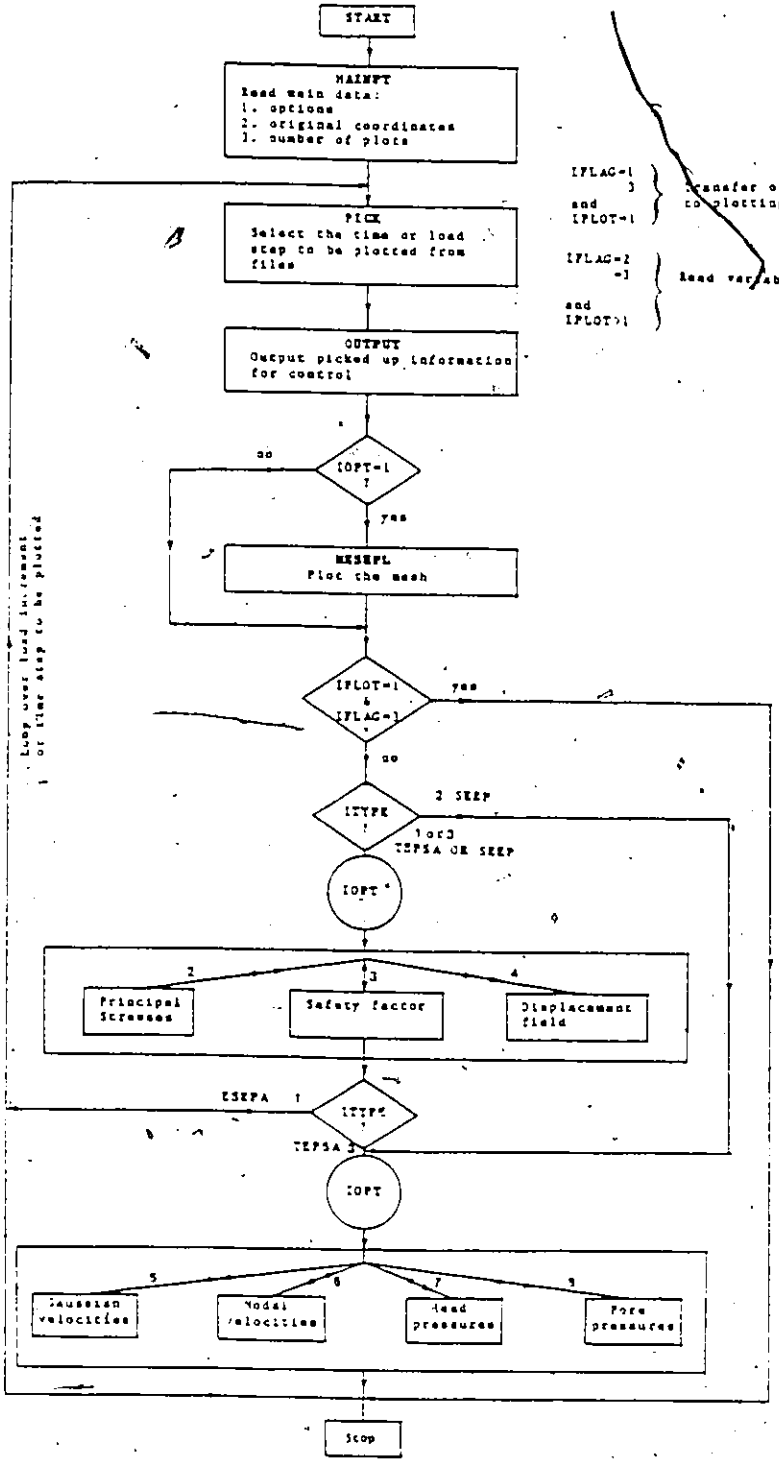


Fig. 7.7. Flowchart FEMPLOT

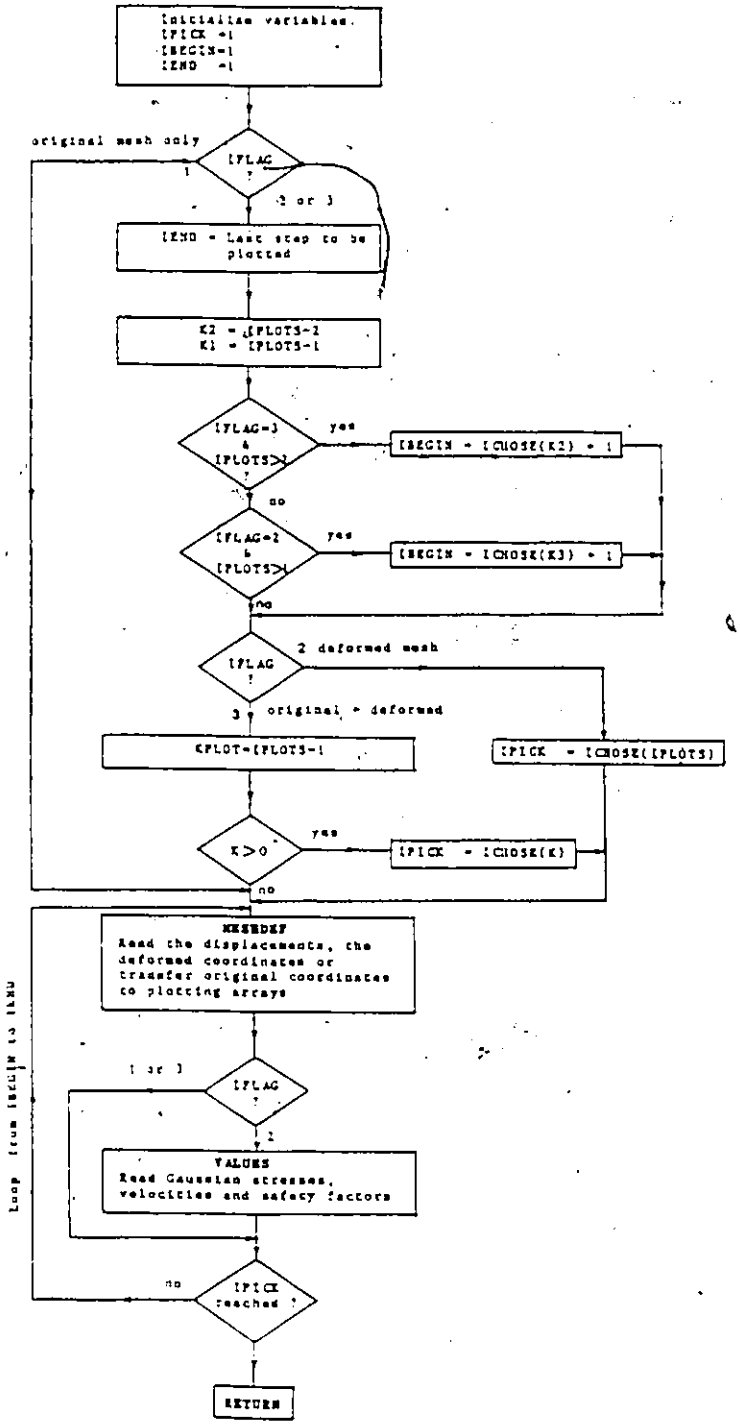


Fig. 7.8 Flowchart PICK

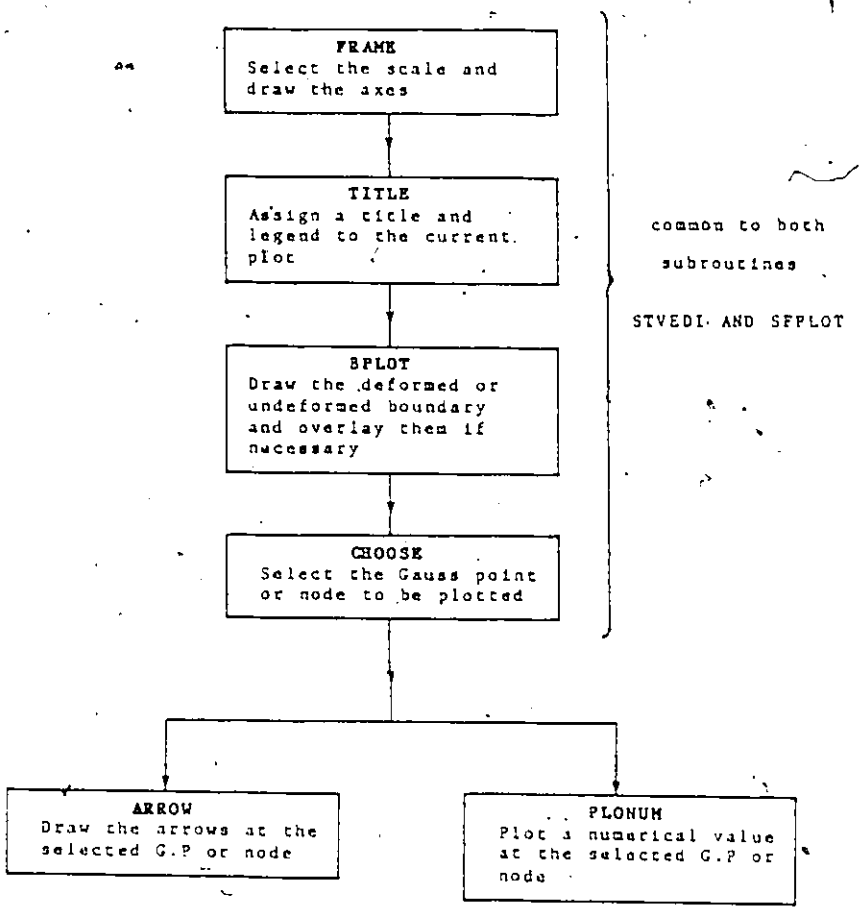


Fig. 7.9 Structure chart for Subroutines STVEDI and SFPLOT

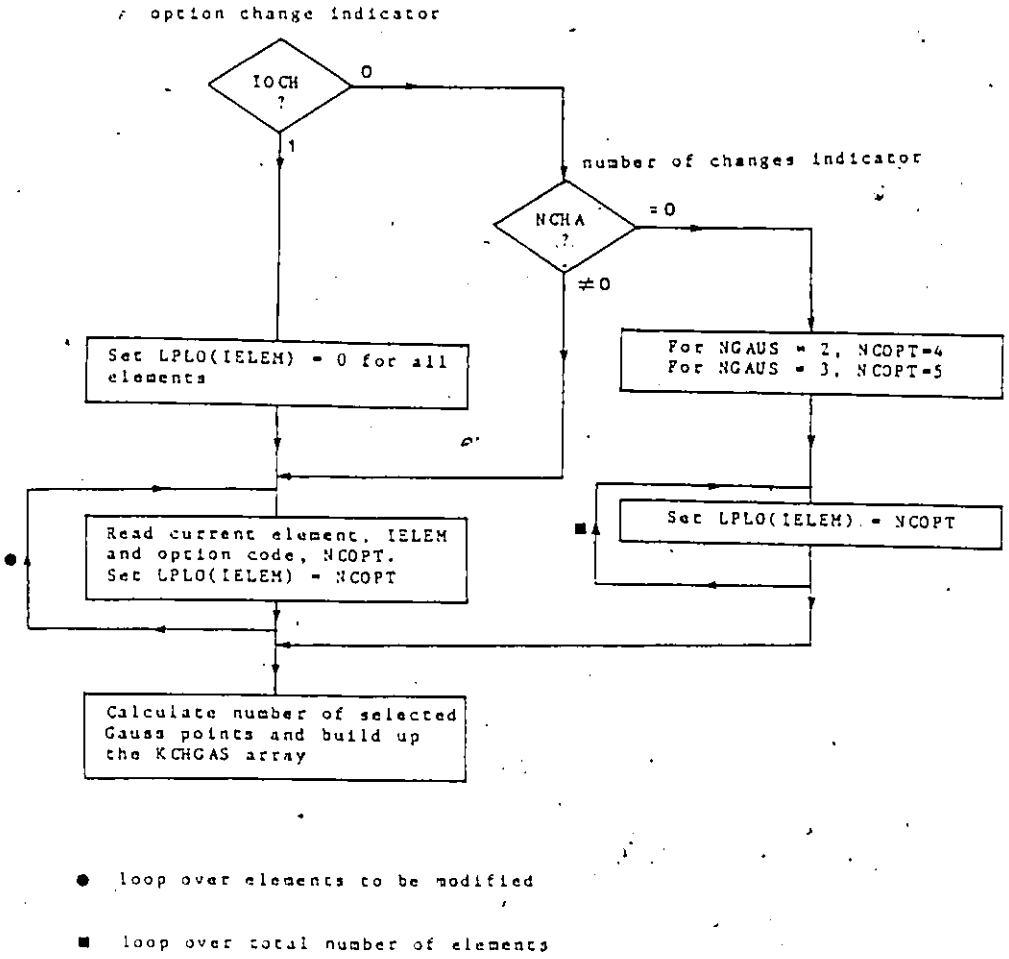


Fig. 7.10a Gauss point selection procedure in Subroutine CHOOSE

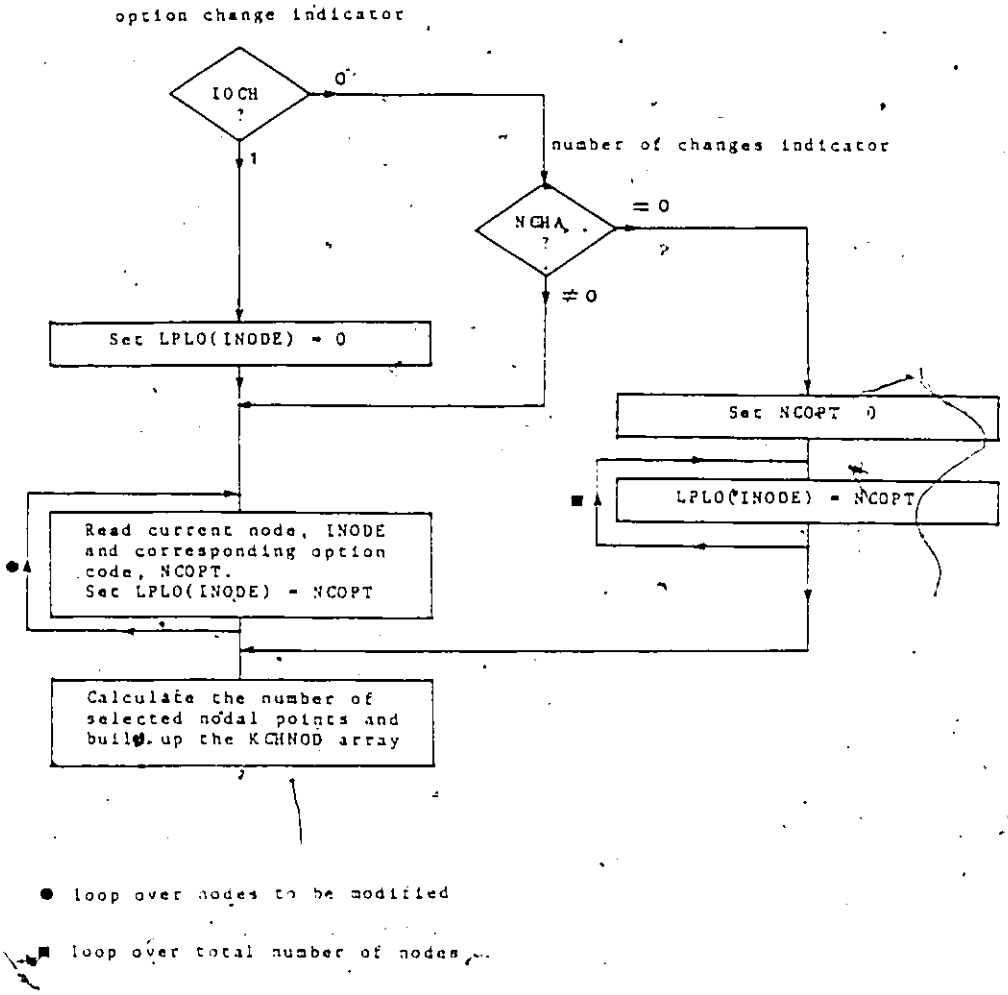


Fig. 7.10b Node selection procedure in Subroutine CHOOSE

Chapter VIII

GENERAL CONCLUSIONS

From the numerical investigations performed, one can conclude that the four finite element programs have proved their efficiency and reliability since the constituent algorithms provided very good results.

Concerning the analysis of a saturated medium, it was seen the true coupled performance in linear elasticity of a composite continuum in which two phases interact with no a priori assumption necessary on the determination of initial pore pressures. On the other hand, the uncoupled approach led to good results as well, to the extent to which it was found applicable. It was in fact observed that the dissipation processes were distinct in the two approaches. Despite the coupled analysis leads to a faster rate of consolidation, it is thought that the uncoupled theory may economically lead to a tentative first answer to a practical problem.

However, the assumption of linear elastic behaviour of the skeleton is known to be too oversimplified and consequently elasto-plasticity was replaced. The constitutive relationships pertaining to the linear elastic perfectly plastic and the generalized Burland models were

tested and critically compared with the theory whenever possible. For example in plane strain conditions the linear elastic perfectly plastic model predicted an ultimate load in close agreement with the one given by the Limit analysis. Moreover, the generalized Burland model behaved well in axisymmetric conditions but apparently predicted low ultimate values in the plane strain boundary value problem. It can be nonetheless observed that this model may suit well the behaviour of soft clay. Concerning convergence of the solution scheme, it is however felt that further work is needed towards the development of a more refined algorithm.

Finally the consideration of elasto-plastic behaviour of the skeleton in the coupled problem led to interesting results. The two models were confronted and results revealed different pore pressure responses to total stress changes. From observations of computed results, it was found that the generalized Burland model, by suiting very well soft clays, could detect delay failures, common in these materials.

The incorporation of non-linear behaviour of the skeleton in the coupled theory surely leads to a more comprehensive study of the consolidation process but still a lot has to be done to understand the mechanisms involved in the generation of pore pressures and as well as their dissipation.

Appendix A

SEEP SUBROUTINES

A.1 SUBROUTINE MOVE

A local numbering system of each DBE has been adopted.

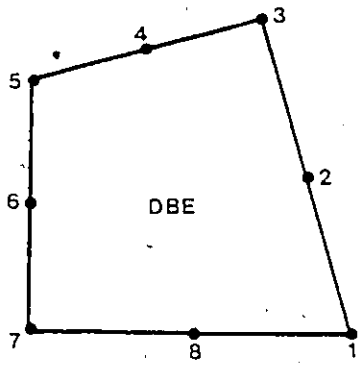


Fig. A1 Local numbering of DBE

The free surface is updated by revising only top nodes 5 and 4 because node 3 is in common with the adjacent DBE. In order to keep a good asperity ratio and uniform distortion of elements inside the DBE, nodes 2 and 6 are always kept as mid-points during the updating process; on the other hand

nodes 7, 8 and 1 are kept fixed except in the case of special DBE, when a regressive update is done (see Fig. 3.7).

Step	
4 - 9	Rewind disk file 3 prior to reading preliminary parameters: NDMN number of domains describing the entire mesh NDYN number of Dynamic Big Elements DBE NSPEC number of Special DBE
10 - 18	Read identification parameters IOPT, NN, NUMAT, NXL, NYL, NTIPE relative to each DBE (for further details refer to mesh generation chapter). Transfer DBE coordinates stored in 2-D array COOBI, into 1-D arrays XM, YM.
19	Consider the special case of the last DBE.
20 - 21	Define the DBE boundaries by means of the slopes of the segments.
22 - 29	Update the first two top nodes (5 and 4) of DBE. LOCFS is an array containing the node numbers of all nodes found on free surface.
30 - 34	Compute the difference, DELTA, between the computed total head and the elevation. Subsequently determine the new position by moving on the DBE segment lines. In the case of the last DBE, the nodes move along the slope of the physical boundary (example downstream slope).
35 - 38	In the case of special DBE, the top nodes are moved by a distance defined by a geometric regression law of ratio 1/2 as shown in Fig. 3.7.
39 - 40	The mid-side nodes (2 and 6) are accordingly adjusted.
45 - 52	Transfer updated coordinates back to 2-D array COOBI.

```
C
C
0001        SUBROUTINE MOVE(COOBI,HEAD,LOCFS,LNOBI,MDUMM,MELEM,MPOIN,
                           MTOTV,NDMN,NDYN,SLOPE)
C*************************************************************************
C
C     THIS SUBROUTINE UPDATES THE NODES BELONGING TO THE FREE SURFACE
C     AT EACH ITERATION.
C
C*************************************************************************
0002        IMPLICIT REAL*8 (A-H,O-Z)
0003        DIMENSION COOBI(MPOIN,2),HEAD(MTOTV),LNOBI(MELEM,9),
                    LOCFS(MDUMM),TANGT(2),YM(8),XM(8)
0004        REWIND 3
C
C***READ INFORMATION ON TAPE DISK
C
0005        READ(3) NDMN,NDYN,NSPEC
0006        NDOM=NDYN-NSPEC
0007        NDO1=NDOM-1
0008        NDO2=NDOM+1
0009        NSP1=NSPEC+1
0010        DO 1000 IDMN=1,NDYN
C
0011        READ(3) IOPT,NN,NUMAT,NXL,NYL,NTIPE
0012        NNOBI=8
0013        IF(IOPT.EQ.3) NNOBI=6
C
C***SET UP SUBDOMAIN NODAL COORDINATES
C
0014        DO 10 INOBI=1,NNOBI
0015        NLOCA=LNOBI(IDMN,INOBI)
0016        XM(INOBI)=COOBI(NLOCA,1)
0017        YM(INOBI)=COOBI(NLOCA,2)
0018     10 CONTINUE
0019        IF(SLOPE.NE.0.0.AND.IDMN.EQ.NDOM) GO TO 20
C
C***DEFINE THE BOUNDARIES OF THE SUBDOMAIN
C
0020        TANGT(1)=(XM(3)-XM(1))/(YM(3)-YM(1))
0021        TANGT(2)=(XM(4)-XM(8))/(YM(4)-YM(8))
C
C***ENTER LOOP TO CHANGE NODE COORDINATES ON FREE SURFACE
C
0022        DO 30 ILOOP= 1,2
0023        INODE=ILOOP+2
0024        IF(IDMN.GT.NDO2) GO TO 35
0025        IGASH=2*(IDMN+1)-ILOOP
0026        IF(IDMN.GE.NSP1.AND.NSPEC.NE.0) IGASH=2*(IDMN+1)-ILOOP+1
0027        IF(IDMN.EQ.NDO2.AND.ILOOP.EQ.1) IGASH=2*(NSPEC+1)
0028        IF(IDMN.EQ.NDO2.AND.ILOOP.EQ.2) GO TO 35
0029        NLOCA=LOCFS(IGASH)
```

```
0030                DELTA=HEAD(NLOCA)-YM(INODE)
0031                IF(SLOPE.NE.0.0.AND.IDMN.EQ.NDO1) TANGT(ILOOP)=SLOPE
0032                XM(INODE)=XM(INODE)+DELTA*TANGT(ILOOP)
0033                YM(INODE)=HEAD(NLOCA)
0034                GO TO 30
                C
                C***UPDATE NODES FOR SPECIAL SUBDOMAINS
                C
0035             35 DELTA=DELTA/2.
0036                XM(INODE)=XM(INODE)+DELTA*TANGT(ILOOP)
0037                YM(INODE)=YM(INODE)+DELTA
                C
0038             30 CONTINUE
                C
0039                IF(SLOPE.NE.0.0.AND.IDMN.EQ.NDO1) XM(4)=(XM(5)+XM(3))/2.
                C
                C***UPDATE MID SIDE NODES
                C
0040                XM(2)=(XM(3)+XM(1))/2.
0041                YM(2)=(YM(3)+YM(1))/2.
0042                GO TO 40
0043             20 XM(4)=(XM(5)+XM(3))/2.
0044                YM(4)=(YM(5)+YM(3))/2.
                C
                C***SET BACK SUBDOMAIN NODAL COORDINATES
                C
0045             40 DO 50 INOBI=1,NNOBI
0046                NLOCA=LNOBI(IDMN,INOBI)
0047                COOBI(NLOCA,1)=XM(INOBI)
0048                COOBI(NLOCA,2)=YM(INOBI)
0049             50 CONTINUE
                C
0050           1000 CONTINUE
0051                RETURN
0052                END
```

A.2 SUBROUTINE HEDFIX

After the update of the free surface, some of the boundary nodes have been inevitably upset so that the fixity array needs to be corrected.

In this subroutine, the nearest node to the downstream water level is located and made to coincide as shown below. The following computer statements are self-explanatory.

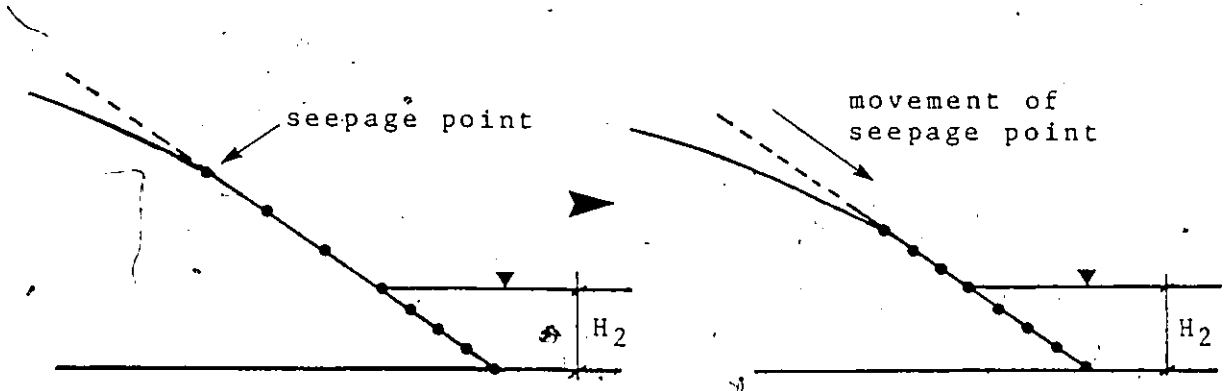


Fig. A2 Movement of seepage point

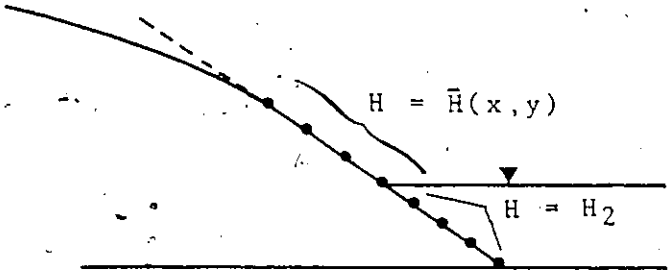


Fig. A3 Boundary conditions

```
                C
                C
0001                 SUBROUTINE HEDFIX(COORD,H2,MPOIN,MVFIX,NDOFN,NDOWN,NOFIX,
                                       NVFIX,PRESC)
                C*****************************************************************
                C
                C     ADJUST THE DOWNSTREAM NODE AND HEAD FIXITY BEFORE EACH ITERATION
                C
                C*****************************************************************
0002                 IMPLICIT REAL*8(A-H,O-Z)
0003                 DIMENSION COORD(MPOIN,2),NOFIX(MVFIX),PRESC(MVFIX,NDOFN)
                C
                C***PRESCRIBE GEOMETRICAL HEIGHT AS HEAD
                C
0004                 JDOWN=NVFIX-NDOWN+1
0005                 DO 10 IVFIX=JDOWN,NVFIX
0006                 NLOCA=NOFIX(IVFIX)
0007                 IF(H2.GT.COORD(NLOCA,2))GO TO 20
0008                 PRESC(IVFIX,1)=COORD(NLOCA,2)
0009              10 CONTINUE
0010                 RETURN
                C
                C***NODES BELOW DOWNSTREAM WATER LEVEL
                C
0011              20 CONTINUE
                C
                C***LOCATE NEAREST NODE TO W.LEVEL AND MAKE IT COINCIDE
                C
0012                 IVFI1=IVFIX-1
0013                 NLOC1=NOFIX(IVFI1)
0014                 TANGT=(COORD(NLOC1,1)-COORD(NLOCA,1))/
                           (COORD(NLOC1,2)-COORD(NLOCA,2))
0015                 TEST=(COORD(NLOC1,2)+COORD(NLOCA,2))/2.
0016                 IMOD1=NLOCA
0017                 JDOWN=IVFIX
0018                 IF(H2.LE.TEST) GO TO 30
0019                 IMOD1=NLOC1
0020                 JDOWN=IVFI1
0021              30 CONTINUE
0022                 COORD(IMOD1,1)=COORD(NLOCA,1)+(H2-COORD(NLOCA,2))*TANGT
0023                 COORD(IMOD1,2)=H2
                C
                C***PRESCRIBE DOWNSTREAM HEAD TO ALL NODES BELOW W.LEVEL
                C
0024                 DO 40 JVFIX=JDOWN,NVFIX
0025                 PRESC(JVFIX,1)=H2
0026              40 CONTINUE
                C
0027                 RETURN
0028                 END
```

A.3 SUBROUTINE GAUSVEL (GAUSSIAN VELOCITIES)

The role of this subroutine is to calculate the fluid velocity components at the Gauss integration points.

Step	
4	Rewind disk file 12 prior to reading stored data.
3 - 11	Loop over the elements to read the cartesian derivatives CARTD of shape functions at Gauss integration points. $$\frac{\partial N_1}{\partial x}(\xi,\eta),\ \frac{\partial N_2}{\partial x}(\xi,\eta) \ldots \frac{\partial N_8}{\partial x}(\xi,\eta)$$ $$\frac{\partial N_1}{\partial y}(\xi,\eta),\ \frac{\partial N_2}{\partial y}(\xi,\eta) \ldots \frac{\partial N_8}{\partial y}(\xi,\eta)$$
12 - 26	Compute the components of velocity at Gauss integration points by means of Darcy's law. hydraulic gradient $i_j(\xi,\eta) = \frac{\partial N_i}{\partial x_j}(\xi,\eta) \cdot H_i$ $v_i(\xi,\eta) = K_{ij}\, i_j$ (ξ,η) Local coordinates of Gauss integration points.

The cartesian derivatives of shape functions are not rigorously defined at the element nodes i.e. on its boundary because of discontinuity in derivatives. In return they are completely defined inside the element.

```
      C
      C
0001        SUBROUTINE GAUVEL(COORD,HEAD,LNODS,MATNO,MELEM,MMATS,MPOIN,
                              MTOTG,MTOTV,NDIME,NELEM,NGAUS,NNODE,NPOIN,
                              NPROP,PROPS,VELCG)
      C*****************************************************************
      C
      C           CALCULATE THE VELOCITY COMPONENTS AT EACH GAUSSIAN POINT
      C
      C*****************************************************************
0002        IMPLICIT REAL*8 (A-H,O-Z)
0003        DIMENSION CARTD(2,9),COORD(MPOIN,2),HEAD(MTOTV),
                      LNODS(MELEM,9),MATNO(NELEM),PROPS(MMATS,NPROP),SHAPE(9),
                      VELCG(MTOTG,4)
0004        REWIND 12
      C
      C***LOOP OVER EACH ELEMENT
      C
0005        KGAUS=0
0006        DO 40 IELEM=1,NELEM
0007        LPROP=MATNO(IELEM)
      C
      C***START LOOP OVER EVERY GAUSSIAN POINT
      C
0008        DO 50 IGAUS=1,NGAUS
0009        DO 50 JGAUS=1,NGAUS
0010        KGAUS=KGAUS+1
      C
      C***CALL BACK CARTESIAN DERIVATIVES OF SHAPE FUNCTIONS AT G.P
      C
0011        READ(12) CARTD
      C
      C***CALCULATE THE HYDRAULIC GRADIENTS AND VELOCITIES.
      C
0012        DO 60 IDIME=1,NDIME
0013        HYDGD=0.0
0014        DO 70 I=1,NNODE
0015        MNODE=IABS(LNODS(IELEM,I))
0016     70 HYDGD=HYDGD+CARTD(IDIME,I)*HEAD(MNODE)
0017     60 VELCG(KGAUS,IDIME)=-PROPS(LPROP,IDIME)*HYDGD
0018        VELCG(KGAUS,3)=DSQRT(VELCG(KGAUS,1)**2+VELCG(KGAUS,2)**2)
0019        IF (DABS(VELCG(KGAUS,1)).LE.0.1E-3) GO TO 80
0020        VELCG(KGAUS,4)=DATAN(VELCG(KGAUS,2)/VELCG(KGAUS,1))*57.29577951
0021        GO TO 50
0022     80 VELCG(KGAUS,4)=0.0
0023     50 CONTINUE
0024     40 CONTINUE
0025        RETURN
0026        END
```

A.4 SUBROUTINE NODVEL (NODAL VELOCITY)

This subroutine calculates the fluid velocity components at the nodes on the basis of Gauss point values using the "smoothing" technique.

Step	
4	Evaluate square root 3
5 - 40	Set up "smoothing" matrix SMATX according to matrix M in Table 3.1.
41 - 42	Compute the number of Gauss integration points, NGAU2, per element and initialize counter KGAUS.
43 - 50	Loop over all elements in order to initialize the counter, ICOUN, and nodal velocity, VELCN.
51	Loop over total number of elements NELEM.
52	Loop over each "INODE" of current element
53	Establish the "local" definition of the current node by means of array LNODS.
54	Increment counter ICOUN to determine the number of common nodes.
55	Determine the global Gauss point number KGAUS.
56	Loop over each dimension "IDIME", for each Gauss integration point KGAUS.
57 - 60	Compute the "smoothed" nodal velocity according to Eq. 3.8.
61	End the loop on elements.
62 - 70	Average the "smoothed" nodal velocities to take into account common nodes. This gives a globally "smoothed" configuration. Also, calculate the modulus and the angle which the velocity vector makes with the x-axis.

```
C
C
0001        SUBROUTINE NODVEL(ICOUN,LNODS,MELEM,MPOIN,MTOTG,NDIME,NELEM,
                              NGAUS,NNODE,NPOIN,VELCG,VELCN)
C*****************************************************************************
C
C       SMOOTHS THE NODAL VELOCITIES FROM THE GAUSSIAN VALUES
C
C*****************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION ICOUN(MPOIN),LNODS(MELEM,NNODE),SMATX(8,4),
                      VELCN(MPOIN,4),VELCG(MTOTG,4)
0004        ROOT3=1.73205080757
C
C*** SET UP SMATX
C
0005        CON1P=(1.+ROOT3)/2.
0006        CON1M=(1.-ROOT3)/2.
0007        CON2P=(2.+ROOT3)/2.
0008        CON2M=(2.-ROOT3)/2.
C
0009        SMATX(1,1)= -0.5
0010        SMATX(1,2)=CON2M
0011        SMATX(1,3)= -0.5
0012        SMATX(1,4)=CON2P
C
0013        SMATX(2,1)=CON1P/2.
0014        SMATX(2,2)=CON1M/2.
0015        SMATX(2,3)=CON1M/2.
0016        SMATX(2,4)=CON1P/2.
C
0017        SMATX(3,1)=CON2P
0018        SMATX(3,2)=-0.5
0019        SMATX(3,3)=CON2M
0020        SMATX(3,4)=-0.5
C
0021        SMATX(4,1)=CON1P/2.
0022        SMATX(4,2)=CON1P/2.
0023        SMATX(4,3)=CON1M/2.
0024        SMATX(4,4)=CON1M/2.
C
0025        SMATX(5,1)= -0.5
0026        SMATX(5,2)=CON2P
0027        SMATX(5,3)= -0.5
0028        SMATX(5,4)=CON2M
C
0029        SMATX(6,1)=CON1M/2.
0030        SMATX(6,2)=CON1P/2.
0031        SMATX(6,3)=CON1P/2.
0032        SMATX(6,4)=CON1M/2.
C
0033        SMATX(7,1)=CON2M
```

```
0034                SMATX(7,2)=-0.5
0035                SMATX(7,3)=CON2P
0036                SMATX(7,4)=-0.5
                C
0037                SMATX(8,1)=CON1M/2.
0038                SMATX(8,2)=CON1M/2.
0039                SMATX(8,3)=CON1P/2.
0040                SMATX(8,4)=CON1P/2.
                C
0041                NGAU2=NGAUS*NGAUS
0042                KGAUS=0
                C
                C***INITIALISE SOME VARIABLES
                C
0043                DO 10 IELEM=1,NELEM
0044                DO 10 INODE=1,NNODE
0045                NLOCA= IABS(LNODS(IELEM,INODE))
0046                ICOUN(NLOCA)=0
0047                DO 20 IDIME=1,NDIME
0048                VELCN(NLOCA,IDIME)=0.0
0049             20 CONTINUE
0050             10 CONTINUE
                C
                C***LOOP OVER ELEMENTS
                C
0051                DO 100 IELEM=1,NELEM
0052                DO 100 INODE=1,NNODE
0053                NLOCA=IABS(LNODS(IELEM,INODE))
0054                ICOUN(NLOCA)=ICOUN(NLOCA)+1
0055                KGAUS=(IELEM-1)*NGAU2
0056                DO 110 IDIME=1,NDIME
0057                DO 110 IGAUS=1,NGAU2
0058                IGASH=KGAUS+IGAUS
0059                VELCN(NLOCA,IDIME)=VELCN(NLOCA,IDIME)+SMATX(INODE,IGAUS)*
                                       VELCG(IGASH,IDIME)
0060             110 CONTINUE
0061             100 CONTINUE
                C
                C***AVERAGE THE COMMON NODAL SMOOTHED VALUES
                C
0062                DO 125 IPOIN=1,NPOIN
0063                DO 120 IDIME=1,NDIME
0064             120 VELCN(IPOIN,IDIME)=VELCN(IPOIN,IDIME)/ICOUN(IPOIN)
                C
0065                VELCN(IPOIN,3)=DSQRT(VELCN(IPOIN,1)**2+VELCN(IPOIN,2)**2)
0066                VELCN(IPOIN,4)=0.0
0067                IF (DABS(VELCN(IPOIN,1)).GT.0.1E-3)
                   .VELCN(IPOIN,4)=DATAN(VELCN(IPOIN,2)/VELCN(IPOIN,1))*57.29577951
0068             125 CONTINUE
0069                RETURN
0070                END
```

Appendix B

TERCON SUBROUTINES

B.1 SUBROUTINE INCREM (INCREMENT)

The role of this subroutine is to update the stiffness matrix and the load array at the beginning of each time step before entering the frontal solution process. The new matrix P and array Δg are updated as prescribed by Eqs. 4.9b and 4.9c.

$$P = \theta P + \frac{M}{\Delta t}$$

$$\Delta g' = \theta \Delta g_n + g_{n-1} - P \pi^e_{n-1}$$

It is to be noted that Δg is completely defined at time t_n since both the law of variation of flow represented by Δg_n and g_{n-1}, and the previous excess pore pressure π^e_{n-1} are known.

Step

4 - 6	Rewind disk files 1, 3 and 10 prior to reading of stored data.
7 - 9	Set load array ELOAD to zero before defining the updated one.

10	Enter the loop over elements.
11 - 12	Read from disk file 3 and 10 EPSTF and CMATX matrices relative to the current element.
13 - 16	Compute the updated matrix P represented by variable PSTIF according to Eq. 4.9b.
17	Store matrix PSTIF for the current element on disk file 1 immediately.
18 - 25	Evaluate the load array according to Eq. 4.9c and store in array ELOAD.

```
C
C
0001        SUBROUTINE INCREM(CMATX,DTIME,ELOAD,EPSTF,EXSPR,FIXED,
           .                  LNODS,MELEM,MEVAB,MPOIN,MTOTV,MVFIX,NDOFN,
           .                  NELEM,NEVAB,NNODE,NOFIX,NTOTV,NVFIX,
           .                  PRESC,PSTIF,RLOAD,TIMEX,TLOAD)
C*****************************************************************
C
C*** UPDATES THE STIFFNESS MATRIX AND THE APPLIED LOAD
C
C*****************************************************************
0002        IMPLICIT REAL*8 (A-H,O-Z)
0003        DIMENSION CMATX(MEVAB,MEVAB),ELOAD(MELEM,MEVAB),
           .          EPSTF(MEVAB,MEVAB),EXSPR(MPOIN),FIXED(MTOTV),
           .          LNODS(MELEM,9),NOFIX(MVFIX),PRESC(MVFIX,NDOFN),
           .          PSTIF(MEVAB,MEVAB),RLOAD(MELEM,MEVAB),
           .          TLOAD(MELEM,MEVAB)
C
0004        REWIND  1
0005        REWIND  3
0006        REWIND 10
C
C*** SET ELOAD ARRAY TO ZERO
C
0007        DO 10 IELEM=1,NELEM
0008        DO 10 IEVAB=1,NEVAB
0009     10 ELOAD(IELEM,IEVAB)=0.0
C
C*** EVALUATE PSTIF FOR EACH ELEMENT
C
0010        DO 40 IELEM=1,NELEM
0011        READ(3) EPSTF
0012        READ(10) CMATX
0013        DO 20 INODE=1,NNODE
0014        DO 20 JNODE=1,NNODE
0015        PSTIF(INODE,JNODE)=TIMEX*EPSTF(INODE,JNODE)+
           .                   CMATX(INODE,JNODE)/DTIME
0016     20 CONTINUE
C
C*** COPY PSTIF IN TAPE 1
C
0017        WRITE(1) PSTIF
0018        DO 30 IEVAB=1,NNODE
0019        DO 35 INODE=1,NNODE
0020        NLOCA=IABS(LNODS(IELEM,INODE))
0021     35 ELOAD(IELEM,IEVAB)=ELOAD(IELEM,IEVAB)+EPSTF(IEVAB,INODE)
           .                   *EXSPR(NLOCA)
0022        ELOAD(IELEM,IEVAB)=RLOAD(IELEM,IEVAB)-ELOAD(IELEM,IEVAB)
0023        TLOAD(IELEM,IEVAB)=ELOAD(IELEM,IEVAB)
0024     30 CONTINUE
0025     40 CONTINUE
C
C*** INTERPRET FIXITY DATA IN VECTOR FORM
C
0026        DO 50 ITOTV=1,NTOTV
0027     50 FIXED(ITOTV)=0.0
C
0028        DO 60 IVFIX=1,NVFIX
0029        NLOCA=(NOFIX(IVFIX)-1)*NDOFN
0030        DO 60 IDOFN=1,NDOFN
0031        NGASH=NLOCA+IDOFN
0032        FIXED(NGASH)=PRESC(IVFIX,IDOFN)
0033     60 CONTINUE
0034        RETURN
0035        END
```

B.2 SUBROUTINE TMARCH (TIME MARCH)

This subroutine calculates the time increment according to four different schemes. The available options are summarized in Table B.1.

Step	
4	Do not calculate the time increment Δt for the first time step (ISTEP=1) or for option 4 (constant time stepping).
5	"Branch" to the three other different time marching schemes according to Table B1
6 - 7	For IOPT= 1, calculate the time interval by $\Delta t_n = (e^r - 1)t_{n-1}$ in terms of the previous time. However, do not update the θ-parameter (TIMEX).
8 - 10	For IOPT= 2, calculate the time interval t as above and update the θ-parameter by means of Sandhu's logarithmic interpolation (Table B.1)
11	Set KSF1 = 2, reference value for the computation of time steps.
12	Determine the size of the sub-interval by variable INTER (number of time steps in the sub-interval).
13	Define parameter identifying sub-interval so far reached.
14 - 18	Test if time interval Δt needs to be changed when shifting to next sub-interval.
19 - 20	Update Δt and θ-parameter according to Table B.1
21 - 22	Print current value of time increment and θ-value.

```
C
C
0001        SUBROUTINE TMARCH(DTIME,GAMAW,HEIGH,ICOUN,IFLAG,IOPTS,ISTEP,
                              MMATS,NPROP,PROPS,TIMEX,TTIME,XTIME)
C*****************************************************************************
C
C***THIS SUBROUTINE CHOOSES THE TIME MARCHING SCHEME
C
C*****************************************************************************
0002        IMPLICIT REAL*8 (A-H,O-Z)
0003        DIMENSION PROPS(MMATS,NPROP)
C
0004        IF(ISTEP.EQ.1.OR.IOPTS.EQ.4) RETURN
0005        GO TO (1,2,3),IOPTS
0006      1 DTIME=TTIME*(2.718282**XTIME-1.0)
0007        RETURN
0008      2 DTIME=TTIME*(2.718282**XTIME-1.0)
0009        TIMEX=1.0+1.0/DTIME-1.0/DLOG(1+DTIME)
0010        RETURN
0011      3 KSF1=2
0012        INTER=XTIME
0013        ITEST=((ISTEP-KSF1)/INTER)*INTER
0014        IF(ISTEP.EQ.KSF1) GO TO 10
0015        ISTEP1=ISTEP-KSF1
0016        IF(ITEST.EQ.ISTEP1) GO TO 10
0017        RETURN
0018       10 ITEST=ITEST/INTER
0019        DTIME=0.1*10.0**ITEST/XTIME
0020        TIMEX=1.0+1.0/DTIME-1.0/DLOG(1+DTIME)
0021        WRITE(6,905) DTIME,TIMEX
0022      905 FORMAT(' DTIME,TIMEX',2E15.6)
0023        RETURN
0024        END
```

TABLE B1

Options for time domain discretization

IOPT	time interval Δt	θ-parameter
1 logarithmic variation of Δt and linear temporal interpolation	$\Delta t_n = (e^r - 1)\, t_{n-1}$ $r = \log_e (t_n) - \log_e (t_{n-1})$ $t_n = t_{n-1} + \Delta t_n$	θ constant with time $\theta = 0.5$ trapezium rule $\theta = 2/3$ optimum value
2 logarithmic variation of Δt and θ-value	same as above	$\theta = 1 + \frac{1}{\Delta t_n} - \frac{1}{\log_e (1+\Delta t_n)}$
3 decomposition into time sub-interval according to Sandhu (1977)	see Table 4.1	same as above
4 constant time step Δt and θ-parameter	as imposed by user	as imposed by user

Appendix C

ESEPA SUBROUTINES

C.1 SUBROUTINE ALGOR (ALGORITHM)

In ESEPA, as previously mentioned, the stiffness matrix can either be kept constant or be updated depending on the user's choice. The type of scheme is introduced in the program through the solution parameter NALGO. The main function of this subroutine therefore is to control the solution process according to the value NALGO.

The subroutine sets the value of the indicator KRESL to either 1 or 2 according to NALGO, and the current value of the iteration and increment. A value of KRESL = 1 indicates that the stiffnesses are to be reformulated and consequently a full system of simultaneous equations must be subsequently solved. If KRESL = 2 the stiffnesses are not to be redefined and therefore only solution of the equation needs to be undertaken. This results in a considerable saving in computer time with equation resolution requiring only 20 % of time needed for complete analysis.

```
C
C
0001        SUBROUTINE ALGOR(FIXED,IINCS,IITER,KRESL,
                             MTOTV,NALGO,NTOTV,KUNLD)
C*************************************************************************
C
C**** THIS SUBROUTINE SETS EQUATION RESOLUTION INDEX,KRESL
C
C*************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION FIXED(MTOTV)
0004        KRESL=2
0005        IF(NALGO.EQ.1.AND.IINCS.EQ.1.AND.IITER.EQ.1) KRESL=1
0006        IF(NALGO.EQ.2) KRESL=1
0007        IF(NALGO.EQ.3.AND.(IITER.EQ.1.OR.KUNLD.EQ.1)) KRESL=1
0008        IF(NALGO.EQ.4.AND.IINCS.EQ.1.AND.IITER.EQ.1) KRESL=1
0009        IF(NALGO.EQ.4.AND.(IITER.EQ.2.OR.KUNLD.EQ.1)) KRESL=1
0010        IF(IITER.EQ.1) RETURN
0011        DO 100 ITOTV = 1,NTOTV
0012        FIXED(ITOTV)=0.0
0013      100 CONTINUE
0014        RETURN
0015        END
```

C.2 SUBROUTINE RESIDU (RESIDUAL STRESS)

The role of this subroutine is to evaluate the nodal forces statically equivalent to the stress field in elasto-plastic conditions by means of the "Initial Stress" method. Detailed explanations were reported in Sect. 5.8.2.

```
                C
                C
0001                  SUBROUTINE RESIDU(ASDIS,COORD,EHIST,ELOAD,EPSIV,ITER,LNODS,
                     .                  MATNO,MELEM,MMATS,MPOIN,MTOTG,MTOTV,NCRIT,NDOFN,
                     .                  NELEM,NEVAB,NGAUS,NMATS,NNODE,NPROP,NSTR1,NSTRE,
                     .                  NTYPE,POSGP,PROPS,STRSG,WEIGP,VSTOT,KUNLD)
                C*****************************************************************
                C
                C**** THIS SUBROUTINE REDUCES THE STRESSES TO THE YIELD SURFACE AND
                C     EVALUATES THE EQUIVALENT NODAL FORCES
                C
                C*****************************************************************
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  DIMENSION ASDIS(MTOTV),AVECT(4),BMATX(4,18),CARTD(2,9),
                     .          COORD(MPOIN,2),DEVIA(4),DERIV(2,9),DESIG(4),
                     .          DMATX(4,4),DVECT(4),EHIST(MTOTG,10),ELCOD(2,9),
                     .          ELDIS(2,9),ELOAD(MELEM,18),EPSIV(MTOTG),GPCOD(2,9),
                     .          LNODS(MELEM,9),POSGP(4),PRESG(4),PROPS(MMATS,NPROP)
0004                  DIMENSION SGTOT(4),SHAPE(9),SIGMA(4),STRAN(4),STRSG(4,MTOTG),
                     .          MATNO(MELEM),WEIGP(4)
0005                  ROOT3=1.73205080757
0006                  TWOPI=6.283185308
0007                  KGAUS=0
0008                  KUNLD=0
                C
                C*** LOOP OVER ELEMENTS
                C
0009                  DO 1000 IELEM=1,NELEM
0010                  DO 10 IEVAB=1,NEVAB
0011               10 ELOAD(IELEM,IEVAB)=0.0
                C
                C*** READ ELEMENT PARAMETERS AND BUILD UP THE ELASTIC MATRIX
                C
0012                  LPROP=MATNO(IELEM)
0013                  THICK=PROPS(LPROP,3)
0014                  CALL      MODPS(DMATX,LPROP,MMATS,NPROP,NTYPE,PROPS)
                C
                C*** COMPUTE COORDINATE AND INCREMENTAL DISPLACEMENTS OF THE
                C     ELEMENT NODAL POINTS
                C
0015                  DO 20 INODE =1,NNODE
0016                  LNODE=IABS(LNODS(IELEM,INODE))
0017                  NPOSN=(LNODE-1)*NDOFN
0018                  DO 20 IDOFN=1,NDOFN
0019                  NPOSN=NPOSN+1
0020                  ELCOD(IDOFN,INODE)=COORD(LNODE,IDOFN)
0021               20 ELDIS(IDOFN,INODE)=ASDIS(NPOSN)
                C
                C*** EVALUATE STRESSES AT GAUSSIAN PTS AND CALCULATE THE EQUIVALENT
                C     NODAL FORCES
                C
0022                  KGASP=0
```

```
0023              DO 100 IGAUS=1,NGAUS
0024              DO 100 JGAUS=1,NGAUS
0025              EXISP=POSGP(IGAUS)
0026              ETASP=POSGP(JGAUS)
0027              KGAUS=KGAUS+1
0028              KGASP=KGASP+1
                  C
                  C*** EVALUATE FROM THE CALCULATED STRAIN THE STRESS INCREMENT,
                  C     THE INVARIANTS ASSUMING LINEAR ELASTIC RESPONSE
                  C
0029              CALL        SFR2(DERIV,ETASP,EXISP,NNODE,SHAPE)
0030              CALL        JACOB2(CARTD,DERIV,DJACB,ELCOD,GPCOD,IELEM,KGASP,
                                 NNODE,SHAPE)
0031              DVOLU=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
0032              IF(NTYPE.EQ.3) DVOLU=DVOLU*TWOPI*GPCOD(1,KGASP)
0033              IF(THICK.NE.0.0) DVOLU=DVOLU*THICK
0034              CALL BMATPS(BMATX,CARTD,NNODE,SHAPE,GPCOD,NTYPE,KGASP)
0035              CALL LINEAR(CARTD,DEPSV,DMATX,ELDIS,LPROP,MMATS,NDOFN,NNODE,NPROP,
                                 NSTRE,NTYPE,PROPS,STRAN,DESIG,KGASP,GPCOD,SHAPE)
0036              EPSIV(KGAUS)=EPSIV(KGAUS)+DEPSV
                  C
                  C*** EVALUATE THE TOTAL STRESS SIGMA
                  C
0037              DO 30 ISTR1=1,NSTR1
0038              PRESG(ISTR1)=STRSG(ISTR1,KGAUS)
0039           30 SIGMA(ISTR1)=PRESG(ISTR1)+DESIG(ISTR1)
0040              VSTOT=EHIST(KGAUS,5)
                  C
                  C*** CHECK IF PLASTICITY HAS OCCURED AND EVENTUALLY CALCULATE
                  C     THE RESIDUAL STRESSES
                  C
0041              CALL PLASTC(DEPSV,DESIG,DVSPE,EHIST,ESCUR,ITER,KGAUS,
                                 KUNLD,LPROP,MMATS,MSTEP,MTOTG,NCRIT,NMATS,
                                 NPROP,NSTR1,PRESG,PROPS,SGTOT,SIGMA,
                                 STRES,VSTOT,YIELD)
                  C
0042              EHIST(KGAUS,7)=ESCUR
0043              IF(ESCUR.LE.0.OR.KUNLD.EQ.1) GO TO 50
                  C
                  C*** CONSIDER THE ELASTO-PLASTIC EQUILIBRIUM
                  C
0044              CALL ELPLAS(DESIG,DVSPE,EHIST,KGAUS,LPROP,MMATS,
                                 MSTEP,MTOTG,NCRIT,NPROP,NSTRE,NSTR1,NTYPE,
                                 PRESG,PROPS,SGTOT,VSTOT,YIELD)
0045              DO 40 ISTR1=1,NSTR1
0046              STRSG(ISTR1,KGAUS)=SGTOT(ISTR1)
0047           40 CONTINUE
0048              GO TO 70
                  C
                  C*** CONSIDER STRESS PATH FOUND INSIDE YIELD SURFACE
                  C
0049           50 DO 60 ISTR1=1,NSTR1

0050              STRSG(ISTR1,KGAUS)=STRSG(ISTR1,KGAUS)+DESIG(ISTR1)
0051           60 CONTINUE
0052              EHIST(KGAUS,5)=(1.+DEPSV)*EHIST(KGAUS,5)
                  C
                  C*** CALCULATE THE EQUIVALENT NODAL FORCES AND ASSOCIATE WITH THE
                  C     ELEMENT NODES
                  C
0053           70 MGASH=0
0054              DO 90 INODE=1,NNODE
0055              DO 80 IDOFN=1,NDOFN
0056              MGASH=MGASH+1
0057              DO 80 ISTRE=1,NSTRE
0058              ELOAD(IELEM,MGASH)=ELOAD(IELEM,MGASH)+BMATX(ISTRE,MGASH)*
                  .STRSG(ISTRE,KGAUS)*DVOLU
0059           80 CONTINUE
0060           90 CONTINUE
0061          100 CONTINUE
0062         1000 CONTINUE
0063              RETURN
0064              END
```

C.3 SUBROUTINE SETCO (SET COEFFICIENT)

It is convenient to convert into coefficients repetitive groups of calculations, once and for all at the beginning of the program, before any major computation starts.

Following this line of thought, Subroutine SETCO has been written in order to transfer all the coefficients to the array of material property PROPS. The balance of excess stresses involves many calculation loops and by adopting the proposed strategy, computer time may be in some sense reduced.

The contents of PROPS are presented in Table C1 for clarity.

Step	
4	Evaluate squate root of 3
5	Loop over total number of materials, NMATS.
6 - 8	Set up cohesion c, internal friction angles ϕ_a and ϕ_b in radians.
9 - 15	Consequently compute Cos ϕ_a, Sin ϕ_a, Sin ϕ_b, c Cosϕ_a and transfer values to PROPS(LPROP,14), PROPS(LPROP,15), PROPS(LPROP,16), PROPS(LPROP,17) respectively for later use.
16 - 22	Define parameters N, M_1, M_2 and n for Drucker-Prager limiting surface according to Tables 5.4. For Mohr-Coulomb limiting surface calculate $n/(M_1 + M_2)$ and transfer to PROPS(LPROP,8) and PROPS(LPROP,18) respectively for later use.
26 - 39	For Drucker-Prager limiting surface calculate a,b,d, g, ā as given in Table 5.4 and transfer to PROPS (LPROP,18) to PROPS(LPROP,22) for latter use.

TABLE C1

Property and stress history allocations

PROPERTY ALLOCATION.

PROPS (NUMAT, 1)	Young's modulus	E	
..... 2)	Poisson ratio	ν	
..... 3)	thickness	t	
..... 4)	density	ρ	
..... 5)	yield stress	σ_y	
..... 6)	Friction angle in compression	ϕ_a	
..... 7)	Friction angle in extension	ϕ_b	
..... 8)	Semi-axes ratio	n	
..... 9)	Slope of normal consolidation line	λ	
..... 10)	Slope of overconsolidation line	χ	
..... 11)	permeability in x direction	K_x	TEPSA
..... 12)	..... y	K_y	TEPSA
..... 13)	..... z	K_z	TEPSA
..... 14)	$\sin \phi_a$		
..... 15)	$\cos \phi_a$		
..... 16)	$\sin \phi_b$		
..... 17)	2c or 2c cos ϕ		
..... 18)	ACOEF	a	
..... 19)	BCOEF	b	
..... 20)	DCOEF	d	
..... 21)	GCOEF	g	
..... 22)	ABAR	$\bar{a}$	

STRESS HISTORY ALLOCATION.

EHIST (KGAUS, 1)	p-coordinate of ellipse centre	α
..... 2)	q	β
..... 3)	major semi-axis of ellipse	h
..... 4)	effective pressure or major axis of ellipse	p_e
..... 5)	specific volume	$\bar{v}$
..... 6)	hardening parameter	k
..... 7)	Escursion (measure of excess stress beyond yield)	ESCUR
..... 8)	Indicator for determining region where stress point is found (over consolidated/normally consolidated)	ICRIT
..... 9)	Indicator for change of region. IOVCO = 1 critical state reached from N.C. region. 2 critical state reached from super-consolidated region.	IOVCO

```
C
C
0001        SUBROUTINE SETCO(MMATS,NCRIT,NMATS,NPROP,PROPS)
C*****************************************************************
C
C  SET COEFFICIENT OF THE BURLAND ELLIPSE FOR LATER USE
C
C*****************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION PROPS(MMATS,NPROP)
C
C***LOOP OVER ALL DIFFERENT MATERIAL
C
0004        ROOT3=1.73205080757
C
0005        DO 10 LPROP=1,NMATS
0006        COHES=PROPS(LPROP,5)
0007        PHIRA=PROPS(LPROP,6)/57.29577951
0008        PHIRB=PROPS(LPROP,7)/57.29577951
C
0009        SINPH=DSIN(PHIRA)
0010        COSPH=DCOS(PHIRA)
0011        SINPB=DSIN(PHIRB)
0012        PROPS(LPROP,14)=SINPH
0013        PROPS(LPROP,15)=COSPH
0014        PROPS(LPROP,16)=SINPB
0015        IF(NCRIT.EQ.1.OR.NCRIT.EQ.2) PROPS(LPROP,17)=2.*COHES
0016        IF(NCRIT.EQ.3.OR.NCRIT.EQ.4) PROPS(LPROP,17)=COHES*COSPH
C
0017        DENOM=(3.0-SINPH)/2.0
0018        CINTE=3.0*COSPH*COHES/DENOM
0019        FINC1=3.0*SINPH/DENOM
0020        DENOM=(3.0-SINPB)/2.0
0021        FINC2=3.0*SINPB/DENOM
0022        RATIO=PROPS(LPROP,8)
C
0023        IF(NCRIT.EQ.2.OR.NCRIT.EQ.4) GO TO 20
0024        PROPS(LPROP,8)=RATIO/(FINC1+FINC2)
0025        PROPS(LPROP,18)=FINC1-FINC2
0026        GO TO 10
C
C***RESERVED FOR BURLAND WITH DRUKER-PRAGER
C
0027     20 RAT12=RATIO*RATIO
0028        ACOEF=RATIO*(FINC1-FINC2)/(FINC1+FINC2)
0029        BCOEF=2.0*RATIO/(FINC1+FINC2)
0030        DCOEF=CINTE/FINC1
0031        GCOEF=DCOEF/(1.0+BCOEF)
0032        ABARR=BCOEF*BCOEF+ACOEF*ACOEF/RAT12-1.0
C
0033        PROPS(LPROP,8) =RATIO
0034        PROPS(LPROP,18) =ACOEF
0035        PROPS(LPROP,19) =BCOEF
0036        PROPS(LPROP,20)=DCOEF
0037        PROPS(LPROP,21)=GCOEF
0038        PROPS(LPROP,22)=ABARR
C
0039     10 CONTINUE
0040        RETURN
0041        END
```

C.4 SUBROUTINE PREVOS (PREVIOUS STRESS)

This subroutine reads the initial stresses $\underset{\sim}{\sigma}_o$ and the initial position of the yield surface.

Step	
4	Rewind disk file 9
5	Evaluate 2
6 - 7	Read the type of initial stresses to be read. ISOPT = 1 for nodal stresses, ISOPT = 1 for Gaussian stresses. Also read the disk file ITAPE, where initial stresses are to be read.
8 - 9	Set ITAPE equal to 5 as card reader by default.
10 - 14	Read the nodal stresses σ_x^o, σ_y^o, σ_z^o and τ_{xy}^o for all points of the structure if ISOPT = 1.
15 - 25	Else, read the initial stresses σ_x^o, σ_y^o, σ_z^o, τ_{xy}^o at all the Gauss integration points.
26 - 34	Read EHIST at each Gauss integration point. EHIST describes the previous stress history of the material (i.e its yield function) of each Gauss integration point. EHIST (KGAUS, 1). . . . . α " (. . , 2). . . . . β " (KGAUS, 3). . . . . h " (KGAUS, 4). . . . . p_e " (KGAUS, 5). . . . . $\bar{v}$ KGAUS: current integration point.
35	Initialize the counter KGAUS which ranges from 1 to NGAUS * NGAUS * NELEM, the total number of Gauss points.
36 - 37	Enter the loop over each element in the structure and identify the material property of the current element.
38 - 42	Identify the thickness of the element and store the current element coordinates in array ELCOD

for convenient use later on.

43 - 49 Set KGASP to ZERO. KGASP ranges from 1 to NGAUS. Then enter the loop on Gauss points and identify the coordinates ξ, η of the current point.

50 - 51 Evaluate the shape functions S and their derivatives $\frac{\partial s_i}{\partial \xi}$, $\frac{\partial s_i}{\partial \eta}$ at current Gauss point as well as the Jacobian matrix and cartesian derivatives $\frac{\partial s_i}{\partial x}$, $\frac{\partial s_i}{\partial y}$ or $\frac{\partial s_i}{\partial r}$, $\frac{\partial s_i}{\partial z}$.

52 - 56 Transform the initial stress at nodes into values at Gauss points.

57 - 59 Compute DVOLU = $d\Omega$. According to the type of problem (axisymmetric plane strain or plane stress.)

60 Call Subroutine BMATPS for the evaluation of the strain-displacement matrix B.

61 -67 Integrate the initial stresses to give the equivalent nodal load TLOAD.

$$f = \int_{\Omega} B^T \underset{\sim}{\sigma}_o \, d\Omega$$

68 - 69 End the loops on Gauss point and elements.

```
C
C
0001        SUBROUTINE PREVOS(COORD,EHIST,LNODS,MATNO,MELEM,MEVAB,MMATS,
                     .                MPOIN,MTOTG,NDIME,NDOFN,NELEM,NEVAB,NGAUS,
                     .                NNODE,NPROP,NSTR1,NSTRE,NTYPE,POSGP,PROPS,
                     .                STRND,STRIN,TLOAD,WEIGP)
            C*****************************************************************
            C
            C*** READS THE INITIAL STRESSES
            C
            C*****************************************************************
0002              IMPLICIT REAL*8(A-H,O-Z)
0003              DIMENSION BMATX(4,18),CARTD(2,9),COORD(MPOIN,2),DERIV(2,9),
                  .         EHIST(MTOTG,10),ELCOD(2,9),GPCOD(2,9),LNODS(MELEM,9),
                  .         MATNO(MELEM),PROPS(MMATS,NPROP),POSGP(4),
                  .         SHAPE(9),STRIN(4,MTOTG),STRND(MPOIN,4),
                  .         TLOAD(MELEM,MEVAB),WEIGP(4)
0004              REWIND  9
0005              TWOPI=6.283185308
            C
            C*** ISOPT=1 ,NODAL STRESSES, ISOPT=0 GAUSSIAN STRESSES
            C
0006              READ(5,900) ISOPT,ITAPE
0007          900 FORMAT(2I5)
0008              IF(ITAPE.EQ.0) ITAPE=5
0009              IF(ISOPT.EQ.0) GO TO 20
            C
            C*** READ THE INITIAL NODAL STRESSES
            C
0010              DO 10 IPOIN=1,NPOIN
0011              READ(ITAPE,901) (STRND(IPOIN,ISTR1),ISTR1=1,NSTR1)
0012           10 WRITE(6,901) (STRND(IPOIN,ISTR1),ISTR1=1,NSTR1)
0013          901 FORMAT(4E15.6)
            C
0014              GO TO 25
            C
            C*** READ THE INITIAL STRESSES
            C
0015           20 WRITE(6,905)
0016          905 FORMAT(//2X,'GAUSS PT.',' X-STRESS',' Y-STRESS',' XY-STRESS',
                  .       ' Z-STRESS',/)
0017              KGAUS=0
0018              DO 100 IELEM=1,NELEM
0019              DO 100 IGAUS=1,NGAUS
0020              DO 100 JGAUS=1,NGAUS
0021              KGAUS=KGAUS+1
0022              READ(ITAPE,910) KELGS,(STRIN(ISTR1,KGAUS),ISTR1=1,NSTR1)
0023              WRITE(6,910) KELGS,(STRIN(ISTR1,KGAUS),ISTR1=1,NSTR1)
0024          100 CONTINUE
0025          910 FORMAT(I5,2X,8E14.6)
            C
            C*** READ THE STRESS HISTORY  : 1= ALPHA, 2= BETAS, 3= HAPAR, 4= PE.
```

```
C
0026       25 KGAUS=0
0027          DO 105 IELEM=1,NELEM
0028          DO 105 IGAUS=1,NGAUS
0029          DO 105 JGAUS=1,NGAUS
0030          KGAUS=KGAUS+1
0031          READ(ITAPE,910) KELGS,(EHIST(KGAUS,IPROP),IPROP=1,3),
                             EHIST(KGAUS,5)
0032          EHIST(KGAUS,4)=EHIST(KGAUS,2)+EHIST(KGAUS,3)
0033          WRITE(6,910) KELGS,(EHIST(KGAUS,IPROP),IPROP=1,5)
0034        105 CONTINUE
C
0035          KGAUS=0
0036          DO 1000 IELEM=1,NELEM
0037          LPROP=MATNO(IELEM)
0038          THICK=PROPS(LPROP,3)
0039          DO 200 INODE =1,NNODE
0040          LNODE=LNODS(IELEM,INODE)
0041          DO 200 IDIME=1,NDIME
0042        200 ELCOD(IDIME,INODE)=COORD(LNODE,IDIME)
C
C*** ENTER LOOP FOR AREA NUMERICAL INTEGRATION
C
0043          KGASP=0
0044          DO 270 IGAUS=1,NGAUS
0045          DO 270 JGAUS=1,NGAUS
0046          KGASP=KGASP+1
0047          KGAUS=KGAUS+1
0048          EXISP=POSGP(IGAUS)
0049          ETASP=POSGP(JGAUS)
C
C*** EVALUATE THE SHAPE FUNCTIONS AT THE SAMPLING POINTS,
C     ELEMENTAL VOLUME AND CARTESIAN DERIVATIVES
C
0050          CALL SFR2(DERIV,ETASP,EXISP,NNODE,SHAPE)
0051          CALL JACOB2(CARTD,DERIV,DJACB,ELCOD,GPCOD,IELEM,KGASP,
                          NNODE,SHAPE)
C
0052          IF(ISOPT.EQ.0) GO TO 35
C
C*** CALCULATE GAUSSIAN STRESSES
C
0053          DO 120 ISTR1=1,NSTR1
0054          DO 120 INODE=1,NNODE
0055          LNODE=LNODS(IELEM,INODE)
0056        120 STRIN(KGAUS,ISTR1)=STRIN(KGAUS,ISTR1)+SHAPE(INODE)*
                                   STRND(LNODE,INODE)
C
0057         35 DVOLU=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
0058          IF(NTYPE.EQ.3) DVOLU=DVOLU*TWOPI*GPCOD(1,KGASP)
0059          IF(THICK.NE.0.0) DVOLU=DVOLU*THICK
0060          CALL BMATPS(BMATX,CARTD,NNODE,SHAPE,GPCOD,NTYPE,KGASP)
C
C*** CALCULATE THE EQUIVALENT NODAL FORCES DUE TO INITIAL STRESSES
C
0061          MGASH=0
0062          DO 150 INODE=1,NNODE
0063          DO 160 IDOFN=1,NDOFN
0064          MGASH=MGASH+1
0065          DO 160 ISTRE=1,NSTRE
0066        160 TLOAD(IELEM,MGASH)=TLOAD(IELEM,MGASH)+BMATX(ISTRE,MGASH)*
                 STRIN(ISTRE,KGAUS)*DVOLU
0067        150 CONTINUE
0068        270 CONTINUE
0069       1000 CONTINUE
0070          RETURN
0071          END
```

C.5 SUBROUTINE PORMTX (PORE MATRIX).

This subroutine evaluates the C-matrix as given in Eq. 5.17c. This latter matrix when post-multiplied by the array of pore pressures, represents the load contribution of the fluid phases, in the effective stress analysis.

Step	
4 - 7	Set up the Kronecker array $\hat{\underset{\sim}{m}}$ named as DELTA.
8	Calculate the strain-displacement matrix B at the current Gauss integration point by means of the standard Subroutine BMATPS.
9 - 12	First calculate the product B * $\hat{\underset{\sim}{m}}$, then multiply by the shape function before integration over DVOLU (Eq. 5.17c). DVOLU takes different forms according to the conditions of plane stress, plane strain and axial symmetry.

```
C
C
0001        SUBROUTINE PORMTX(CARTD,CMATX,DVOLU,GPCOD,KGASP,MMATS,
                              NEVAB,NNODE,NSTRE,NTYPE,SHAPE)
C*******************************************************************************
C
C**** EVALUATES THE CMATRIX
C
C*******************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION BTM(18),BMATX(4,18),CARTD(2,9),CMATX(18,9),DELTA(4),
                      GPCOD(2,9),SHAPE(9)
C
C***  SET UP MDELTA MATRIX
C
0004        DELTA(1)= 1.0
0005        DELTA(2)= 1.0
0006        DELTA(3)= 0.0
0007        DELTA(4)= 1.0
C
C***  EVALUATE BTM MATRIX AT GAUSSIAN POINT
C
0008        CALL BMATPS(BMATX,CARTD,NNODE,SHAPE,GPCOD,NTYPE,KGASP)
C
C*** EVALUATE CMATX
C
0009        DO 10 IEVAB=1,NEVAB
0010        BTM(IEVAB)=0.0
C
0011        DO 20 ISTRE=1,NSTRE
0012     20 BTM(IEVAB)=BTM(IEVAB)+BMATX(ISTRE,IEVAB)*DELTA(ISTRE)
C
0013        DO 30 INODE=1,NNODE
0014     30 CMATX(IEVAB,INODE)=CMATX(IEVAB,INODE)+BTM(IEVAB)*SHAPE(INODE)*
                               DVOLU
C
0015     10 CONTINUE
C
0016        RETURN
0017        END
```

C.6 SUBROUTINE PLASTC (PLASTIC)

The role of this subroutine is to determine the intersection of the stress vector with the yield surface.

Step	
4	Establish the function $$f = [(p+\beta)^2 + \frac{(q-\alpha)^2}{n^2}]^{1/2}$$ for later evaluation at different stress points.
5 - 8	Set working yield function coefficients α, β, h, n as defined earlier.
9 - 10	Establish square root of 3 and set REDUC to one (r = 1).
11	Calculate the invariants p, q, θ associated with previous stress point $\underset{\sim}{\sigma}_{n-1}$.
12	Calculate the semi-axes ratio n, function of θ, for Mohr-Coulomb limiting surface.
13	Compute function f defined in step 4 at stress point $\underset{\sim}{\sigma}_{n-1}$.
14 - 16	Repeat the calculations in steps 11-13 at current stress point $\underset{\sim}{\sigma}_{n}$.
17 - 20	Check if the current Gauss point was previously elastic. If yield had already occured go to 10.
21 - 23	If the Gauss point was previously elastic, check to see if it has yielded during this iteration.
24	For the Gauss point which is still elastic during the current iteration, go back to calling subroutine.
25	If the Gauss point is yielding, call Subroutine FACTOR to calculate reduction factor r.
26 - 32	Check to see if a Gauss point which had previously yielded is undergoing unloading. If yes, return to calling subroutine.
33	Otherwise set r= 0.

34 - 36 Calculate the number of balancing steps MSTEP.

37 - 40 Truncate MSTEP to 500 in case it is greater.

41 Define RFACT as (1-REDUC)

42 - 45 Evaluate $\underset{\sim}{\sigma}_n = \underset{\sim}{\sigma}_{n-1} + r\,\Delta\underset{\sim}{\sigma}_e$ and store in array SGTOT and calculate $s/m\,\Delta g^e$ and store in array DESIG.

46 - 47 Compute the current specific volume and according to

$$\bar{v}_i = (1 + r d\varepsilon_v) \cdot v_{i-1}$$

$$dv = (1-r)\, d\varepsilon_v \cdot \bar{v}_{i-1} = s\, d\varepsilon_v \cdot \bar{v}_{i-1}$$

48 Go back to calling subroutine for the first iteration of balancing process.

```
C
C
0001        SUBROUTINE PLASTC(DEPSV,DESIG,DVSPE,EHIST,ESCUR,ITER,KGAUS,
                              KUNLD,LPROP,MMATS,MSTEP,MTOTG,NCRIT,NMATS,
                              NPROP,NSTRI,PRESG,PROPS,SGTOT,SIGMA,
                              STRES,VSTOT,YIELD)
            C*********************************************************************
            C
            C              CALCULATES THE REDUCTION FACTOR
            C
            C*********************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION DESIG(4),EHIST(MTOTG,10),PREDV(4),
                      PRESG(4),PROPS(MMATS,NPROP),SIGDV(4),SIGMA(4),
                      SGTOT(4)
0004        FUNCT(PMEAN,STEFF,ALPHA,BETAS,RATIO)=DSQRT((PMEAN+BETAS)**2+
                  ((1.73205080757*STEFF-ALPHA)/RATIO)**2)
0005        ALPHA=EHIST(KGAUS,1)
0006        BETAS=EHIST(KGAUS,2)
0007        PREYS=EHIST(KGAUS,3)
0008        RATIO=PROPS(LPROP,8)
0009        ROOT3=1.73205080757
0010        REDUC=1.0
            C
            C*** CALCULATE THE EQUIVALENT PRESSURE FOR PREVIOUS & CURRENT STRESS.
            C
0011        CALL INVAR(PREDV,LPROP,MMATS,NCRIT,NPROP,PROPS,
                       PRESF,PRESG,THETA,PREMN,PREJ2,PREJ3,YIELD)
0012        IF(NCRIT.EQ.3)
           .CALL POTEN(ALPHA,PREBE,DEFEH,EFFST,HARDS,HPARA,3,KGAUS,
                       LPROP,MMATS,NCRIT,NPROP,PROPS,PREQU,PREMN,
                       PRESF,THETA,VSTOT)
0013        EFFST=FUNCT(PREMN,PRESF,ALPHA,BETAS,RATIO)
            C
0014        CALL INVAR(SIGDV,LPROP,MMATS,NCRIT,NPROP,PROPS,
                       SIGSF,SIGMA,THETA,SIGMN,SIGJ2,SIGJ3,YIELD)
0015        IF(NCRIT.EQ.3)
           .CALL POTEN(ALPHA,BETAS,DEFEH,SIGYS,HARDS,HPARA,3,KGAUS,
                       LPROP,MMATS,NCRIT,NPROP,PROPS,SIGQU,SIGMN,
                       SIGSF,THETA,VSTOT)
0016        SIGYS=FUNCT(SIGMN,SIGSF,ALPHA,BETAS,RATIO)
            C
0017        ESPRE=EFFST-PREYS
0018        DESPR=DABS(ESPRE)
0019        IF(DESPR.LT.1.E-5) ESPRE=0.0
0020        IF(ESPRE.GE.0.) GO TO 10
0021        ESCUR=SIGYS-PREYS
0022        DESCU=DABS(ESCUR)
0023        IF(DESCU.LT.1.E-5) ESCUR=0.0
0024        IF(ESCUR.LE.0.) RETURN
            C
0025        CALL FACTOR(DESIG,EHIST,KGAUS,LPROP,MMATS,MTOTG,NCRIT,NPROP,
```

```
                                PREDV,PREMN,PREJ2,PREJ3,PROPS,REDUC)
0026                  GO TO 30
                C
0027               10 ESCUR=SIGYS-EFFST
0028                  DESCU=DABS(ESCUR)
0029                  IF(DESCU.LT.1.E-5) ESCUR=0.0
0030                  IF(ESCUR.GT.0.) GO TO 20
0031                  KUNLD=1
0032                  RETURN
                C
0033               20 REDUC=0.0
                C
                C*** REDUCE THE STRESS ON THE YIELD SURFACE
                C
                C
0034               30 DENOM=PREYS
0035                  IF(DENOM.EQ.0.) DENOM=SIGYS
0036                  MSTEP=ESCUR*6.0/DENOM+6
0037                  IF(MSTEP.LT.500) GO TO 35
0038                  WRITE(6,905) KGAUS
0039              905 FORMAT(3X,' MSTEP AT GAUSS PT ',I5,' TRUNCATED TO 500';/)
0040                  MSTEP=500
0041               35 RFACT=1.-REDUC
                C
0042                  DO 40 ISTR1=1,NSTR1
0043                  SGTOT(ISTR1)=PRESG(ISTR1)+REDUC*DESIG(ISTR1)
0044                 ?DESIG(ISTR1)=RFACT*DESIG(ISTR1)
0045               40 CONTINUE
                C
0046                  VSTOT=(1.+REDUC*DEPSV)*EHIST(KGAUS,5)
0047                  DVSPE=RFACT*DEPSV*EHIST(KGAUS,5)
0048                  IF(ITER.GT.1) RETURN
                C
0049                  EHIST(KGAUS,9)=1
                CCCCCCCCCCCC
                C
                C*** EHIST(..,9) CONTROLS UPDATING ELLIPSE'S PARAMETER IN SUB. POTENTIAL
                C     TO BE STUDIED... FEB.6  85
                C       TMEAN=-(SGTOT(1)+SGTOT(2)+SGTOT(4))/3.
                C       TESTC=TMEAN-EHIST(KGAUS,2)
                C       IF(TESTC.LE.0.0) EHIST(KGAUS,9)=2
                CCCCCCCCCCCC
0050                  RETURN
0051                  END
```

C.7 SUBROUTINE POTEN (POTENTIAL FUNCTION)

The yield surface composed of a Mohr-Coulomb or Drucker-Prager surface combined with a Cam-clay cap has to be updated during the process of plastic deformation.

This subroutine, therefore, calculates the ellipse that satisfies, for a given limiting surface (Mohr-Coulomb or Drucker-Prager), the current stress point (p,q) in a p-q plot, Fig. 5.4. Also, the plastic modulus A is calculated as in Eqs. 5.41. It is noted that all pertaining equations are reported in Table 5.4.

Step	
4	Compute square root of 3.
5 - 12	Recover from array PROPS constant parameters n, a, b, d, g, $\bar{a}$ prior to calculation of the ellipse for Drucker-Prager limiting surface.
14 - 17	Recover from array PROPS cohesion c, $\sin\phi_a$, $\cos\phi_a$ and $\sin\phi_b$. ϕ_a and ϕ_b are the internal friction angle in compression and extension respectively for Mohr-Coulomb limiting surface.
18 - 22	Calculate slopes M_1 and M_2 of the Mohr-Coulomb limiting lines, Eq. 5.40 f in Table 5.4.
23	Calculate the ratio n of the ellipse semi-axes as in Eq. 5.40c.
24	Return to the calling program if ITEST = 3, in which case only the ratio n is being calculated.
25 - 30	Define parameters a, b, d, g, $\bar{a}$ as in Table 5.4 for Mohr-Coulomb limiting surface.
31 - 32	Calculate invariant q from the second deviatoric invariant J_2 as $q = \sqrt{3 J_2}$, and $\bar{b}$.
33	Calculate the plastic modulus A in case ITEST = 2.

34 Compute $\bar{c}$ according to Eq. 5.40k.

35 - 43 Calculate the major semi-axis h according to Eq. 5.40g and 5.40h following the value of a.

44 - 47 In case h is negative, stop the calculation and print out the Gauss point at which this occurs.

48 - 49 Calculate accordingly the coordinates α and β of the ellipse, Eq. 5.40 b

50 Return to the call program if ITEST= 0, i.e. only the ellipse parameters need to be calculated.

52 - 58 Calculate the plastic modulus A according to Eq. 5.41.

```
C
C
0001        SUBROUTINE POTEN(ALPHA,BETAS,DEFEH,EMEAN,HARDS,HPARA,ITEST,KGAUS,
                            LPROP,MMATS,NCRIT,NPROP,PROPS,QINVA,SMEAN,
                            STEFF,THETA,VSTOT)
C*****************************************************************************
C
C          CALCULATES THE ASSOCIATED ELLIPSE PARAMETERS
C
C    ITEST=0  CALCULATE ELLIPSE PARAMETERS ONLY
C    ITEST=1  CALCULATE ELLIPSE PARAMETERS AND HARDENING PARAMETER
C    ITEST=2  CALCULATE HARDENING PARAMETER ONLY
C    ITEST=3  CALCULATE RATIO ONLY
C*****************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION PROPS(MMATS,NPROP)
0004        ROOT3=1.73205080757
0005        IF(NCRIT.EQ.3)GO TO 10
C
C*** READ DIRECTLY THE ELLIPSE COEFICIENT WICH ARE CONSTNT R DRUKER-PRAG
C
0006        RATIO=PROPS(LPROP,8)
0007        RATI2=RATIO*RATIO
0008        ACOEF=PROPS(LPROP,18)
0009        BCOEF=PROPS(LPROP,19)
0010        DCOEF=PROPS(LPROP,20)
0011        GCOEF=PROPS(LPROP,21)
0012        ABARR=PROPS(LPROP,22)
0013        GO TO 20
C
C*** RESERVED ONLY FOR MOHR COULOMB
C
0014     10 COHES=PROPS(LPROP,5)
0015        SINPH=PROPS(LPROP,14)
0016        COSPH=PROPS(LPROP,15)
0017        SINPB=PROPS(LPROP,16)
C
0018        DENOM=ROOT3*DCOS(THETA)-SINPH*DSIN(THETA)
0019        CINTE=3.0*COSPH*COHES/DENOM
0020        FINC1=3.0*SINPH/DENOM
0021        DENOM=ROOT3*DCOS(THETA)-SINPB*DSIN(THETA)
0022        FINC2=3.0*SINPB/DENOM
C
0023        RATIO=PROPS(LPROP,8)*(2.*FINC1-PROPS(LPROP,18))
0024        IF(ITEST.EQ.3) RETURN
0025        RATI2=RATIO*RATIO
0026        ACOEF=RATIO*(FINC1-FINC2)/(FINC1+FINC2)
0027        BCOEF=2.0*RATIO/(FINC1+FINC2)
0028        DCOEF=CINTE/FINC1
0029        GCOEF=DCOEF/(1.0+BCOEF)
0030        ABARR=BCOEF*BCOEF+ACOEF*ACOEF/RATI2-1.0
C
```

```
C*** IN COMMON FOR BOTH LIMITING SURFACE
C
0031       20 QINVA=ROOT3*STEFF
0032          BBARR=QINVA*ACOEF/RATI2-(SMEAN-DCOEF)*BCOEF
C
0033          IF(ITEST.EQ.2) GO TO 40
C
0034          CBARR=QINVA*QINVA/RATI2+(SMEAN-DCOEF)*(SMEAN-DCOEF)
0035          ABSOL=DABS(ABARR)
0036          BBSOL=DABS(BBARR)
0037          IF(ABSOL.LE.1.0E-5.AND.BBSOL.GT.1.0E-5)HPARA=CBARR/(2*BBARR)
0038          IF(ABSOL.LE.1.0E-5.AND.BBSOL.LE.1.0E-5)HPARA=0.0
0039          IF(ABSOL.LE.1.0E-5) GO TO 30
0040          HPAR1=(BBARR-DSQRT(BBARR*BBARR-ABARR*CBARR))/ABARR
0041          HPAR2=(BBARR+DSQRT(BBARR*BBARR-ABARR*CBARR))/ABARR
0042          HPARA=HPAR1
0043          IF(HPARA.LT.0.)HPARA=HPAR2
0044          IF(HPARA.GT.0.)GO TO 30
0045          WRITE(6,900) HPAR1,HPAR2,KGAUS
0046        900 FORMAT(3X,'IMPOSSIBLE ELLIPSE :HPARA < 0.0',/,
                      'HPAR1=',E15.6,3X,'HPAR2=',E15.6,3X,'GAUSSIAN POINT',I5,/)
0047          STOP
C
0048       30 ALPHA=ACOEF*HPARA
0049          BETAS=BCOEF*HPARA-DCOEF
0050          IF(ITEST.EQ.0) GO TO 50
C
C*** SET UP THE HARDENING PARAMETER AT FIRST
C
0051       40 CONTINUE
0052          HLAMD=PROPS(LPROP,9)
0053          CHAIS=PROPS(LPROP,10)
0054          DEFEH=2.*(ABARR*HPARA-BBARR)
0055          DEFPI=2.*(SMEAN+BETAS)
0056          HARDS=(HPARA-GCOEF)*DEFEH*DEFPI*VSTOT/(HLAMD-CHAIS)
C
0057       50 RETURN
0058          END
```

C.8 SUBROUTINE FACTOR.

This subroutine calculates the reduction factor r.

Step	
4	Evaluate the increments ds_{ij}, dI_1, dJ_2, dJ_3 due to the stress increment $d\sigma_{ij}$ by calling Subroutine INVAR.
5	Calculate coefficient Z according to Table 5.7.
6 - 8	Set up ALPHA, BETA and HPARA which represent the ellipse parameters.
9 - 10	Distinguish between the two failure criteria namely Mohr-Coulomb and Drucker-Prager, as well as the case $\alpha = 0$. (ellipse is symmetric about the p-axis).
11 - 16	Compute coefficients Y, Z, W as given in Table 5.7.
18	In the event of "quasi-isotropic" function like the Mohr-Coulomb limiting surface, or when α is different from zero solve for r-factor iteratively by using the bisection method (Subroutine BISECT).
19 - 20	For Drucker-Prager limiting surface or in the simple case $\alpha = 0$, solve trivially for r using Eq. 3 in Table 5.8.

```
C
C
0001        SUBROUTINE FACTOR(DESIG,EHIST,KGAUS,LPROP,MMATS,MTOTG,
     .                        NCRIT,NPROP,PREDV,PREMN,PREJ2,PREJ3,PROPS,REDUC)
C*********************************************************************
C
C   CALCULATES THE REDUCTION FACTOR TO BE APPLIED TO DSIGMA IN ORDER TO
C   HAVE THE TOTAL STRESS ON THE SURFACE
C
C*********************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION DESDV(4),DESIG(4),EHIST(MTOTG,10),PREDV(4),
     .            PROPS(MMATS,NPROP)
C
C*** SET UP CONSTANT
C
0004        CALL INVAR(DESDV,LPROP,MMATS,NCRIT,NPROP,PROPS,
     .             DESSF,DESIG,THETA,DESMN,DESJ2,DESJ3,YIELD)
C
0005        ZCOJ2=PREDV(3)*DESDV(3)+0.5*(PREDV(1)*DESDV(1)+PREDV(2)*DESDV(2)
     .      +PREDV(4)*DESDV(4))
C
0006        ALPHA=EHIST(KGAUS,1)
0007        BETAS=EHIST(KGAUS,2)
0008        HPARA=EHIST(KGAUS,3)
0009        IF(ALPHA.EQ.0.AND.NCRIT.EQ.4) GO TO 3
0010        IF(NCRIT.EQ.4) GO TO 2
C
0011        ACOJ3=PREDV(1)*PREDV(2)*DESDV(4)+PREDV(1)*PREDV(4)*DESDV(2)
     .        +PREDV(2)*PREDV(4)*DESDV(1)
0012        CCOJ3=2.0*PREDV(4)*PREDV(3)*DESDV(3)+PREDV(3)*PREDV(3)*DESDV(4)
0013        DCOJ3=PREDV(1)*DESDV(2)*DESDV(4)+PREDV(2)*DESDV(1)*DESDV(4)+
     .         PREDV(4)*DESDV(1)*DESDV(2)
0014        FCOJ3=2.0*(PREDV(3)*DESDV(4)*DESDV(3))+PREDV(4)*DESDV(3)*DESDV(3)
0015        YCOJ3=ACOJ3+CCOJ3
0016        WCOJ3=DCOJ3+FCOJ3
C
0017      2 CALL BISECT(ALPHA,BETAS,DESMN,DESJ2,DESJ3,DIFLF,HPARA,
     .              LPROP,MMATS,NCRIT,NPROP,PREMN,PREJ2,PREJ3,PROPS,
     .              REDUC,WCOJ3,YCOJ3,ZCOJ2)
0018        RETURN
C
C*** BURLAND  FOR ALPHA EQ 0. AND DRUCKER-PRAGER
C
0019      3 CALL SOLVE(BETAS,DESJ2,DESMN,HPARA,LPROP,MMATS,NPROP,
     .             PREMN,PREJ2,PROPS,REDUC,ZCOJ2,KGAUS)
0020        RETURN
0021        END
```

C.9 SUBROUTINE SOLVE

This subroutine calculates the reduction factor r in the case the equations reduce to the solution of a quadratic equation whose coefficients A, B, C are reported in Table 5.8. The following computer statements are self-explanatory.

```
                C
                C
0001                  SUBROUTINE SOLVE(BETAS,DESJ2,DESMN,HPARA,LPROP,MMATS,
                                       NPROP,PREMN,PREJ2,PROPS,REDUC,ZCOJ2,KGAUS)
                C*************************************************************************
                C
                C       FINDS REDUC WHENEVER IS AVAILABLE A CLOSED FORM SOLUTION
                C
                C*************************************************************************
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  DIMENSION PROPS(MMATS,NPROP)
                C
                C***BURLAND IN THE HPS: DRUCKER-PRAGER & ALPHA=0.0
                C
0004                  RATIO=PROPS(LPROP,8)
0005                  RAT12=RATIO*RATIO
0006                  PREB2=(PREMN+BETAS)*(PREMN+BETAS)
0007                  ACOEF=DESMN*DESMN+3.*DESJ2/RAT12
0008                  BCOEF=DESMN*(PREMN+BETAS)+3.0*ZCOJ2/RAT12
0009                  CCOEF=PREB2-HPARA*HPARA+3.0*PREJ2/RAT12
                C
0010                  DISCR=BCOEF*BCOEF-ACOEF*CCOEF
0011                  IF(DISCR.GT.0.0) GO TO 5
0012                  WRITE(6,900) KGAUS
0013            900   FORMAT( 3X,'NEGATIVE DISCRIMINANT AT GAUSS PT',I5)
0014                  STOP
0015              5   REDU1=(-BCOEF+DSQRT(BCOEF*BCOEF-ACOEF*CCOEF))/ACOEF
0016                  DRED1=DABS(REDU1)
0017                  IF(DRED1.LT.0.00001) REDU1=0.0
0018                  IF(REDU1.GE.0.0.OR.REDU1.LE.1.) GO TO 10
0019                  WRITE(6,910) REDU1
0020            910   FORMAT(3X,E15.6,'***NEGATIVE REDUCTION FACTOR, CHECK PROG')
0021                  STOP
0022             10   REDUC=REDU1
0023                  RETURN
0024                  END
```

C.10 SUBROUTINE BISECT

This subroutine finds the solution of the equation $F(x) = 0$ by means of the bisection method. The function $F(x)$ has to be continuous over a given interval for the iterative process to converge.

The function to be solved is the yield function F given by Eqs. 1 and 2 in Table 5.8 where as the interval of study is [0, 1].

At each step the length of the interval is reduced by a factor 2. Generally each step produces one more accurate decimal point of the root r of $F(p,q,\theta) = 0$. Therefore, the solution can be performed to any desired accuracy provided the function is continuous, this at the expense of the amount of calculations.

For steeply changing functions, the bisection process may lead to oscillatory results, thus requiring a large number of iterations for convergence. To curtail this problem, the number of iterations is restricted to 30 whereas the size of the current bisected interval is limited to 0.0001.

The accuracy of the root is controlled by RCONV which is equal to 0.1 in the subroutine. In fact the accuracy in finding the root i.e. the reduction factor r need not to be great because Subroutine BACK can refine the solution by bringing the "drifted" stress on the yield surface.

Step	
4 - 6	Set up the convergence criterion RCONV and define the solution interval.
7 - 8	Evaluate the yield function F at one extremity RLEFT as TOLLE by calling Subroutine VERIF.
9	Compare TOLLE with RCONV
10 - 11	If TOLLE is less than RCONV, keep RLEFT as solution and return to calling subprogram.
12 - 13	Instead, evaluate the yield function F at RIGHT, the other extremity.
14 - 16	Keep RIGHT as solution and return to calling subprogram if TOLLE is less than RCONV.
17 - 20	Else, check for the sign of the product F(RIGHT)* F(LEFT) to locate the interval for the next bisection.
21 - 26	Establish a counter to determine the number of iterations and restrict it to 30 at most.
27	Compute the difference DIFRL between the two extremities of the interval.
28	Bisect the interval.
29	Stop the bisection process if DIFRL is less than 0.0001.
30 - 32	Evaluate the yield function F at the point of bisection and compare with RCONV.
33 - 40	Maintain the bisection process until convergence is achieved i.e. F(HALPL) < RCONV.
41 - 43	Call solution as REDUC and return to calling subprogram.

```
                C
                C
0001                SUBROUTINE BISECT(ALPHA,BETAS,DESMN,DESJ2,DESJ3,DIFLF,HPARA,
                   .                  LPROP,MMATS,NCRIT,NPROP,PREMN,PREJ2,PREJ3,PROPS,
                   .                  REDUC,WCOJ3,YCOJ3,ZCOJ2)
                C*********************************************************************
                C
                C     SOLVES THE EQUATION USING THE BISECTION METHOD
                C
                C*********************************************************************
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                DIMENSION PROPS(MMATS,NPROP)
0004                RCONV=0.1
0005                RLEFT=0.0
0006                RIGHT=1.0
0007                CALL VERIF(ALPHA,BETAS,DESMN,DESJ2,DESJ3,DIFLF,HPARA,
                   .           LPROP,MMATS,NCRIT,NPROP,PREMN,PREJ2,PREJ3,PROPS,
                   .           RLEFT,WCOJ3,YCOJ3,ZCOJ2)
0008                TOLLE=DABS(DIFLF)
0009                IF(TOLLE.GT.RCONV)GO TO 10
0010                REDUC=RLEFT
0011                RETURN
                C
0012             10 CALL VERIF(ALPHA,BETAS,DESMN,DESJ2,DESJ3,DIFRG,HPARA,
                   .           LPROP,MMATS,NCRIT,NPROP,PREMN,PREJ2,PREJ3,PROPS,
                   .           RIGHT,WCOJ3,YCOJ3,ZCOJ2)
0013                TOLLE=DABS(DIFRG)
0014                IF(TOLLE.GT.RCONV)GO TO 20
0015                REDUC=RIGHT
0016                RETURN
0017             20 SIGRT=DIFLF*DIFRG
0018                IF(SIGRT.LT.0.) GO TO 30
0019                REDUC=0.0
0020                RETURN
                C
                C***LOOP OVER BY TRIAL AND ERROR
                C
0021             30 ICOUN=0
0022             40 ICOUN=ICOUN+1
0023                IF(ICOUN.LE.30) GO TO 45
0024                WRITE(6,950)
0025            950 FORMAT(3X,'NUMBER OF ITERATION EXCEEDED IN BISECT',/,
                   .       3X,'TRY TO CHANGE THE LOAD INCREMENT',/)
0026                STOP
0027             45 DIFRL=DABS(RIGHT-RLEFT)
0028                HALPH=(RIGHT+RLEFT)/2.
0029                IF(DIFRL.LT.0.0001) GO TO 100
0030                CALL VERIF(ALPHA,BETAS,DESMN,DESJ2,DESJ3,DIFHA,HPARA,
                   .           LPROP,MMATS,NCRIT,NPROP,PREMN,PREJ2,PREJ3,PROPS,
                   .           HALPH,WCOJ3,YCOJ3,ZCOJ2)
0031                TOLLE=DABS(DIFHA)
0032                IF(TOLLE.LE.RCONV) GO TO 100
                C
0033                SIGRT=DIFRG*DIFHA
0034                IF(SIGRT.GE.0.) GO TO 50
0035                RLEFT=HALPH
0036                DIFLF=DIFHA
0037                GO TO 40
0038             50 RIGHT=HALPH
0039                DIFRG=DIFHA
0040                GO TO 40
                C
0041            100 REDUC=HALPH
0042                RETURN
0043                END
```

C.11 SUBROUTINE VERIF (VERIFY)

This subroutine evaluates the yield function F, given $\bar{p}$, $\bar{q}$, $\bar{\theta}$, in other words, checks if the trial reduction factor gives a stress state satisfying the yield surface.

Step	
4	Evaluate square root 3
5 - 9	Establish the working parameters c, n, sin ϕ_a, cos ϕ_a, sin ϕ_b.
10 - 13	Compute the reduced invariants $\bar{p}$, $\bar{q}$.
14	Distinguish between Mohr-Coulomb and Drucker-Prager limiting surface.
15 - 20	Check for the value of $\bar{J}_2$. If $\bar{J}_2$ is zero, set $\bar{\theta}$=0, or else calculate $\bar{\theta}$ as given in Table 5.8.
21 - 22	Truncate the value of sin 30° to 1 to avoid numerical instabilities.
23 - 28	Calculate M_1 and n according to Table 5.5.
29 - 32	Compute the value of the yield function F as in Eq. 2 in Table 5.8

```
C
C
0001        SUBROUTINE VERIF(ALPHA,BETAS,DESMN,DESJ2,DESJ3,DIFFE,HPARA,
     .                       LPROP,MMATS,NCRIT,NPROP,PREMN,PREJ2,PREJ3,PROPS,
     .                       REDUC,WCOJ3,YCOJ3,ZCOJ2)
C*********************************************************************
C
C     THIS SUBROUTINE CHECKS IF THE TRIAL REDUCTION SATISFIES THE
C     BOUNDARY  SURFACE.
C
C*********************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION PROPS(MMATS,NPROP)
0004        ROOT3=1.7320580757
0005        COHES=PROPS(LPROP,5)
0006        RATIO=PROPS(LPROP,8)
0007        SINPH=PROPS(LPROP,14)
0008        COSPH=PROPS(LPROP,15)
0009        SINPB=PROPS(LPROP,16)
0010        REDC2=REDUC*REDUC
0011        BARMN=PREMN+REDUC*DESMN
0012        BARJ2=(PREJ2+2.0*REDUC*ZCOJ2+REDC2*DESJ2)
0013        BARSF=DSQRT(BARJ2)
0014        IF(NCRIT.EQ.4) GO TO 10
C
C***PART RESERVED FOR MOHR-COULOMB ,CALCULATES THETA
C
0015        IF(BARJ2.EQ.0.0) GO TO 20
0016        BARJ3=PREJ3+REDUC*YCOJ3+REDC2*WCOJ3+REDC2*REDUC*DESJ3
0017        SINT3=-3.0*ROOT3*BARJ3/(2.0*BARJ2*BARSF)
0018        GO TO 30
0019     20 SINT3=0.0
0020     30 CONTINUE
0021        IF(SINT3.LT.-1.0)SINT3=-1.0
0022        IF(SINT3.GT.1.0) SINT3=1.0
0023        THETA=DARSIN(SINT3)/3.0
0024        SINTE=DSIN(THETA)
0025        COSTE=DCOS(THETA)
0026        DENOM=ROOT3*COSTE-SINPH*SINTE
0027        FINC1=3.0*SINPH/DENOM
0028        RATIO=PROPS(LPROP,8)*(2.*FINC1-PROPS(LPROP,18))
C
C***CHECK IF THE REDUCED STRESS LIES ON THE SURFACE
C
0029     10 FIRTE=BARMN+BETAS
0030        SECTE=(ROOT3*BARSF-ALPHA)/RATIO
C
0031        DIFFE=FIRTE*FIRTE+SECTE*SECTE-HPARA*HPARA
0032        RETURN
0033        END
```

C.12 SUBROUTINE ELPLAS (ELASTO-PLASTIC)

The role of this subroutine is to reduce the stress stepwise on the yield surface.

Step	
4	Establish square root 3
5 - 8	Set up working parameters α, β, h, ITEST. ITEST checks if the these parameters need to be updated when calling Subroutine POTEN.
9 - 10	Determine the number of stress components NSTRS according to the type of problem.
11 - 12	Loop over stress reduction steps.
13	Calculate the stress invariants I_1, J_2, J_3 at stress point to be reduced.
14	Calculate the specific volume v at current balancing step according to Eq. 5.49a.
16 - 17	Compute flow vector $\underset{\sim}{a} = \underset{\sim}{\nabla}_{\underset{\sim}{\sigma}} F$.
18 - 22	Compute $d\bar{l}$ according to equation 5.42c. and store in DLAMB.
23 - 29	Compute $\underset{\sim}{\sigma}_{n-1} + \frac{s}{m} \Delta\underset{\sim}{\sigma}^e - \overline{d\ell}\underset{\sim}{c}_F$ (refer to Eqs. 5.42) and store in SGTOT.
30	End the loop over stress reduction steps to finally give the stress point D.
OPTION	Bring back stress point D on yield surface for linear elastic perfectly plastic model.
31 - 38	Calculate the updated ellipse parameters associated with balanced stress point to follow the hardening or softening process.

```
C
C
0001        SUBROUTINE ELPLAS(DESIG,DVSPE,EHIST,KGAUS,LPROP,MMATS,
                              MSTEP,MTOTG,NCRIT,NPROP,NSTRE,NSTR1,NTYPE,
                              PRESG,PROPS,SGTOT,VSTOT,YIELD)
C*********************************************************************
C
C           REDUCES THE STRESS STEPWISE ON THE SURFACE
C                          **
C     C.STATE REACHED FROM SUPER C.REGION (EHIST(KGAUS,9)=2)
C     C.STATE REACHED FROM N.CONSOL.REGION (EHIST(KGAUS,9)=1)
C*********************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION AVECT(4),DESIG(4),DEVIA(4),DVECT(4),
                      EHIST(MTOTG,10),PREDV(4),PRESG(4),PROPS(MMATS,NPROP),
                      SGTOT(4),SIGMA(4),STRES(4)
0004        ROOT3=1.73205080757
0005        ALPHA=EHIST(KGAUS,1)
0006        BETAS=EHIST(KGAUS,2)
0007        HPARA=EHIST(KGAUS,3)
0008        ITEST=EHIST(KGAUS,9)
0009        NSTRS=NSTRE
0010        IF(NTYPE.EQ.2) NSTRS=NSTR1
C
C*** INSIDE CRITICAL SURFACE
C
0011        ASTEP=MSTEP
0012        DO 1000 ISTEP=1,MSTEP
0013        CALL INVAR(DEVIA,LPROP,MMATS,NCRIT,NPROP,PROPS,STEFF,SGTOT,
                       THETA,SMEAN,VARJ2,VARJ3,YIELD)
C
0014        VSTOT=VSTOT+DVSPE/ASTEP
C
0015        CALL POTEN(ALPHA,BETAS,DEFEH,EMEAN,HARDS,HPARA,ITEST,KGAUS,
                       LPROP,MMATS,NCRIT,NPROP,PROPS,QINVA,SMEAN,
                       STEFF,THETA,VSTOT)
0016        CALL YIELDF(ALPHA,AVECT,BETAS,DEVIA,LPROP,MMATS,NCRIT,NPROP,
                        NSTR1,PROPS,SMEAN,STEFF,THETA,VARJ2,KGAUS)
0017        CALL FLOWPL(AVECT,ABETA,DVECT,HARDS,LPROP,NPROP,NTYPE,
                        PROPS,MMATS,NSTR1)
0018        AGASH=0.0
0019        DO 30 ISTR1=1,NSTR1
0020     30 AGASH=AGASH+AVECT(ISTR1)*DESIG(ISTR1)/ASTEP
0021        DLAMD=AGASH*ABETA
0022        IF(DLAMD.LT.0.0) DLAMD=0.0
0023        BGASH=0.0
0024        DO 40 ISTRS=1,NSTRS
0025        PRESG(ISTRS)=SGTOT(ISTRS)
0026        BGASH=BGASH+AVECT(ISTRS)*SGTOT(ISTRS)
0027        STRES(ISTRS)=DESIG(ISTRS)/ASTEP-DLAMD*DVECT(ISTRS)
0028        SGTOT(ISTRS)=SGTOT(ISTRS)+STRES(ISTRS)
0029     40 CONTINUE
0030   1000 CONTINUE
C*********************************************************************
C
C  IN CASE OF FIXING THE BOUNDARY SURFACE, AFTER BALACING FORCES IT IS
C  NECESSARY TO BRING BACK THE STRESS ON THE BOUNDARY SURFACE
C*** BRING BACK TO THE SURFACE THE ALMOST BALANCED STRESS
C
C       IF(EHIST(KGAUS,9).EQ.2.)
C      .CALL BACK(ALPHA,BETAS,HPARA,KGAUS,LPROP,MMATS,
C      .          NCRIT,NPROP,NSTR1,PROPS,SGTOT)
C
C1000 CONTINUE
C       RETURN
C*********************************************************************
C
C*** UPDATE ELLIPSE PARAMETER TO THE LAST BALANCED STRESS POINT
C
0031        CALL INVAR(DEVIA,LPROP,MMATS,NCRIT,NPROP,PROPS,STEFF,SGTOT,
                       THETA,SMEAN,VARJ2,VARJ3,YIELD)
0032        CALL POTEN(ALPHA,BETAS,DEFEH,EFFST,HARDS,HPARA,ITEST,KGAUS,
                       LPROP,MMATS,NCRIT,NPROP,PROPS,QINVA,SMEAN,
                       STEFF,THETA,VSTOT)
0033        EHIST(KGAUS,1)=ALPHA
0034        EHIST(KGAUS,2)=BETAS
0035        EHIST(KGAUS,3)=HPARA
0036        EHIST(KGAUS,4)=EFFST
0037        EHIST(KGAUS,5)=VSTOT
0038        EHIST(KGAUS,6)=HARDS
C
0039        RETURN
0040        END
```

C.13 SUBROUTINE BACK

This subroutine has 2 functions. It can either bring back the drifted stress on the yield surface at the end of the stress balancing process or refine the solution calculated by Subroutine BISECT.

Step

4	Establish square root 3
5 - 9	Set up working parameters c, n, $\sin \phi_a$, $\cos \phi_a$, $\sin \phi_b$.
10	Calculate the mean stress I_1 and deviatoric stress J_2 at considered stress point by calling Subroutine INVAR.
11 - 15	Calculate slope M_1 and ratio n as defined previously for Mohr-Coulomb limiting surface.
19 - 22	Calculate value of $\bar{q}$ which satisfies given yield surface, given the value of the mean pressure $\bar{p}$ and associated parameters.
23 -24	Calculate the reduction factor r which would bring back the stress point on the surface.
25 - 28	Compute the stress components σ_x, σ_y, σ_z associated to final reduced stress point.

```
C
C
0001        SUBROUTINE BACK(ALPHA,BETAS,HPARA,KGAUS,LPROP,MMATS,
                            NCRIT,NPROP,NSTR1,PROPS,SGTOT)
C*****************************************************************
C
C           BRING BACK THE STRESS POINT ON THE SURFACE
C
C*****************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION AVECT(4),DEVIA(4),PROPS(MMATS,NPROP),SGTOT(4)
0004        ROOT3=1.73205080757
0005        COHES=PROPS(LPROP,5)
0006        RATIO=PROPS(LPROP,8)
0007        SINPH=PROPS(LPROP,14)
0008        COSPH=PROPS(LPROP,15)
0009        SINPB=PROPS(LPROP,16)
C
0010        CALL INVAR(DEVIA,LPROP,MMATS,NCRIT,NPROP,PROPS,
                       STEFF,SGTOT,THETA,SMEAN,VARJ2,VARJ3,YIELD)
C
0011        IF (NCRIT.EQ.4) GO TO 10
C
C***RESERVED ONLY FOR MOHR COULOMB
C
0012        DENOM=ROOT3*DCOS(THETA)-SINPH*DSIN(THETA)
0013        FINC1=3.0*SINPH/DENOM
C
0014        RATIO=PROPS(LPROP,8)*(2.0*FINC1-PROPS(LPROP,18))
C
0015        GO TO 20
0016     10 CONTINUE
0017        FINC1=6.*SINPH/(3.-SINPH)
0018     20 CONTINUE
0019        RAHPA=RATIO*HPARA
0020        SMBET=SMEAN+BETAS
C
C*** CALCULATE Q(J2) FOR GIVEN SURFACE AND P(I1)
C
0021        QUTOT=ROOT3*STEFF
0022        QUCRI=ALPHA+DSQRT(RAHPA*RAHPA-SMBET*SMBET)
C
0023        REDUC=QUCRI/QUTOT
0024     30 CONTINUE
C
C*** BRING BACK THE STRESS POINT ON THE SURFACE
C
0025        SGTOT(1)=REDUC*DEVIA(1)+SMEAN
0026        SGTOT(2)=REDUC*DEVIA(2)+SMEAN
0027        SGTOT(4)=REDUC*DEVIA(4)+SMEAN
C
0028        RETURN
```

Appendix D

TEPSA SUBROUTINES

D.1 SUBROUTINE FRONT

The solution of all previously described F.E. programs were performed by means of the subroutine FRONT published by OWEN and HINTON in 1980. In the current F.E. implementation of the coupled problem, this subroutine had to be slightly amended.

In fact the frontal technique is performed in two stages. The first stage comprises a peculiar procedure to assemble and reduce the stiffness ma[illegible]t this stage, the element stiffness components a[illegible]ated and stored in a way depending on the [illegible] assigned to the problem. The second stage in[illegible]dertakes the actual solution of the problem.

For iterative solutions involving constant stiffness matrices and nodal fixities, the first stage can be performed only once since the assembled and reduced global stiffness matrices remain unchanged.

In the present coupled analysis, the algorithm adopted in TEPSA requires, at each iteration, twice the solution of Eq. 6.13a with different nodal fixities. This implies that at each solution the assembled and reduced global stiffness

matrices are different and have therefore to be stored in distinct disk files for later recovery. Subroutine FRONT was therefore modified to suit the algorithm, thus leading to considerate saving of computer time.

D.2 SUBROUTINE ALGOR (ALGORITHM)

As in ESEPA, this subroutine controls both the reformulation of the composite stiffness matrix and the solution process. Herein, the reformulation depends on the update of the constituent submatrix K^{ep} and time parameter θ. Therefore, an indicator, KSTIF, controls the recalculation of the element stiffness K^{ep}, while KDTIM indicates the change of parameter θ. Finally, according to KSTIF and KDTIM, a third indicator, KRESL, monitors the reconstruction of the composite stiffness matrix, as well as the reformulation of the solution process.

The value of KSTIF is determined according to the solution parameter, NALGO, in the same fashion as KRESL in ESEPA.

```
C
C
0001        SUBROUTINE ALGOR(CFIXED,FIXED,IINCS,IITER,ISTEP,KDTIM,KRESL,
                             KSTIF,MCTOTV,NALGO,NCHEK,NCTOTV,KUNLD)
C*************************************************************************
C
C**** THIS SUBROUTINE SETS EQUATION RESOLUTION INDEX,KRESL
C
C*************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION CFIXED(MCTOTV),FIXED(MCTOTV)
0004        KRESL=2
0005        KSTIF=2
C DOUBT ON KUNLD...
0006        IF(ISTEP.GT.1.AND.NCHEK.EQ.0.AND.KUNLD.EQ.0)GO TO 10
C
C*** DECIDE IF REDEFINE STIFFNESS
C
0007        IF(NALGO.EQ.1.AND.IINCS.EQ.1.AND.IITER.EQ.1) KSTIF=1
0008        IF(NALGO.EQ.2) KSTIF=1
0009        IF(NALGO.EQ.3.AND.(IITER.EQ.1.OR.KUNLD.EQ.1)) KSTIF=1
C
C.DOUBTS ON ISTEP  ..IF(NALGO=4.AND.(IITER.LE.2.OR.KUNLD.EQ.1))KSTIF=1
0010        IF(NALGO.EQ.4.AND.IINCS.EQ.1.AND.IITER.EQ.1) KSTIF=1
0011        IF(NALGO.EQ.4.AND.(IITER.EQ.2.OR.KUNLD.EQ.1)) KSTIF=1
C
C*** DECIDE IF ASSEMBLE A NEW TOTAL STIFFNESS AND SOLVE
C
0012     10 IF(KDTIM.EQ.1.OR.KSTIF.EQ.1)KRESL=1
0013        IF(IITER.EQ.1.AND.ISTEP.EQ.1) RETURN
0014        DO 100 ICTOTV = 1,NCTOTV
0015        FIXED(ICTOTV)=0.0
0016        CFIXED(ICTOTV)=0.0
0017    100 CONTINUE
0018        RETURN
0019        END
```

D.3 SUBROUTINE CLOAD

This subroutine builds up the R.H.S of Eq. 6.11. Two load arrays CELOAD and CTLOAD are built in parallel, one containing the load yet to be balanced and the other one the total load applied.

Step	
4	Rewind disk file 12 prior to reading the coupling submatrix CMATX and hydraulic stiffness EPSTF.
5	Loop over elements:
6	Read from disk file 12 coupling submatrix CMATX and hydraulic "stiffness" submatrix EPSTF.
7	Loop over each "INODE" of current element.
8	Loop over each degree of freedom of the current "INODE".
9 - 10	Establish the "local" and "global" positions, IEVAB and ICEVB, of the "displacement" degree of freedom.
11 - 14	Increment the load arrays ELOAD and TCLOAD by the external load WLOAD in steps following the imposed load history.
15	Transfer the "local" load array ELOAD to the "global" array CELOAD.
16	Compute the effective "equivalent" load which is carried out by the soil skeleton for the current value of excess pore pressure according to Eq. 6.12 and store into the global array CTLOAD.
17	End loop over degrees of freedom.
18	Establish the location of the "pore pressure" degree of freedom.
19 - 23	Compute the hydraulic load contribution according to Eq. 3.2c.
24 - 25	Store hydraulic load contribution lumped with the

weighting factor w in global load arrays CELOAD and CTLOAD.

26 End the loop over the number of nodes.

27 End the loop over elements.

```
C
C
0001        SUBROUTINE CLOAD(CELOAD,CTLOAD,DTIME,ELOAD,EXPWP,FCLOAD,ISTEP,
     .                       KSF,LNODS,MCEVAB,MELEM,MEVAB,MPOIN,NCDOFN,
     .                       NCHEK,NDOFN,NELEM,NNODE,TLOAD,WLOAD,WLUMP)
C*****************************************************************************
C
C*** INITIALISES AND REDUCES CELOAD & CTLOAD
C
C*****************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION CELOAD(MELEM,MCEVAB),CMATX(18,9),CTLOAD(MELEM,MCEVAB),
     .            ELOAD(MELEM,MEVAB),EPSTF(9,9),EXPWP(MPOIN),
     .            FCLOAD(MELEM,MEVAB),LNODS(MELEM,9),
     .            TLOAD(MELEM,MEVAB),WLOAD(MELEM,MEVAB)
0004        REWIND 12
0005        DO 1000 IELEM=1,NELEM
0006        READ(12) CMATX,EPSTF
0007        DO 10 INODE=1,NNODE
0008        DO 20 IDOFN=1,NDOFN
0009        IEVAB=(INODE-1)*NDOFN+IDOFN
0010        ICEVB=(INODE-1)*NCDOFN+IDOFN
0011        IF(NCHEK.GT.0.OR.ISTEP.GT.KSF) GO TO 30
C
C***CALCULATE & SMOOTH THE LOAD DUE TO EXTERNAL MATERIAL FORCES
C
0012        ELOAD(IELEM,IEVAB)=ELOAD(IELEM,IEVAB)+WLOAD(IELEM,IEVAB)/KSF
0013        TLOAD(IELEM,IEVAB)=TLOAD(IELEM,IEVAB)+WLOAD(IELEM,IEVAB)/KSF
0014     30 CONTINUE
0015        CELOAD(IELEM,ICEVB)=ELOAD(IELEM,IEVAB)
C
C***ADD THE CONTRIBUTION OF THE FLUID PART
C
0016        CTLOAD(IELEM,ICEVB)=TLOAD(IELEM,IEVAB)-FCLOAD(IELEM,IEVAB)
0017     20 CONTINUE
C
C***CONSTRUCT THE SECOND PART OF THE LOAD ARRAY
C
0018        ISPARE=ICEVB+1
0019        GCLOAD=0.0
0020        DO 40 JNODE=1,NNODE
0021        LOCA=IABS(LNODS(IELEM,JNODE))
0022        GCLOAD=GCLOAD+EPSTF(INODE,JNODE)*EXPWP(LOCA)
0023     40 CONTINUE
0024        CELOAD(IELEM,ISPARE)=GCLOAD*DTIME*WLUMP
0025        CTLOAD(IELEM,ISPARE)=GCLOAD*DTIME*WLUMP
0026     10 CONTINUE
0027   1000 CONTINUE
0028        RETURN
0029        END
```

D.4 SUBROUTINE ASSEM (ASSEMBLE)

The role of this subroutine is to build up the composite stiffness matrix from its constituent submatrices K^{ep}, P and C the way illustrated in Eq. 6.13a.

Step	
4 - 6	Rewind disk files 1, 11, 12 on which the element "composite" matrices K^{ep}, C and P are stored.
7	Loop over elements.
8 - 9	Read from disk file 11 and 12 ESTIF, CMATX and EPSTF representing respectively K^{ep}, C and P submatrices.
10	Loop over each 'INODE' of current element.
11	Loop over each degree of freedom of the current node "INODE"
12 - 13	Establish the "local" and "global" row positions, IEVAB and ICEVB, for the "displacement" degree of freedom.
14	Loop over each "INODE" of current element.
15	Loop over each degree of freedom of the current node "INODE".
16 - 17	Establish the "local" and "global" column positions, JEVAB and ICEVB for the "displacement" degree of freedom.
18 - 19	Transfer corresponding element stiffness component ESTIF(IEVAB,JEVAB) to its "global" location.
20 - 21	Establish the "global" column position of the "pore pressure" degree of freedom and transfer the corresponding coupling submatrix component CMATX(IEVAB,JNODE).
22 - 23	End the loop on "JNODE" and degree of freedom.
24	Establish the "global" row position of the "pore

pressure" degree of freedom.

25 Loop over "INODE".

26 Loop over each degree of freedom of the current node "JNODE".

27 Establish the column position of JEVAB of the in the element submatrix.

28 Establish the "global" column position JCEVB of the "displacement" degree of freedom.

29 Transfer corresponding coupling matrix component CMATX(JEVAB,INODE) to its global location.

30 End the loop on degrees of freedom.

31 - 32 Establish the "global" column position of the "pore pressure" degree of freedom and transfer the corresponding hydraulic "stiffness" component to its "global" location.

33 - 34 End the loops on columns and rows i.e. "JNODE" and "INODE".

35 Store the element "global" stiffness matrix on disk file 1

36 End the loop on elements.

```
                C
                C
0001                  SUBROUTINE ASSEM(CESTIF,DTIME,ESTIF,NNODE,NDOFN,NCDOFN,NELEM,
                                       TIMEX,WLUMP)
                C*****************************************************************
                C                                                        EECEE
                C*** ASSEMBLES ESTIF,CMATX AND EPSTF MATRICES AS         EECEE
                C                                                        CCPCC
                C*****************************************************************
0002                  IMPLICIT REAL*8 (A-H,O-Z)
0003                  DIMENSION CESTIF(27,27),CMATX(18,9),ESTIF(18,18),EPSTF(9,9)
0004                  REWIND 1
0005                  REWIND 11
0006                  REWIND 12
                C
                C*** LOOP OVER ELEMNTS
                C
0007                  DO 1000 IELEM=1,NELEM
0008                  READ(11) ESTIF
0009                  READ(12) CMATX,EPSTF
                C
0010                  DO 10 INODE=1,NNODE
0011                  DO 20 IDOFN=1,NDOFN
0012                  IEVAB=(INODE-1)*NDOFN+IDOFN
0013                  ICEVB=(INODE-1)*NCDOFN+IDOFN
                C
0014                  DO 30 JNODE=1,NNODE
0015                  DO 40 JDOFN=1,NDOFN
0016                  JEVAB=(JNODE-1)*NDOFN+JDOFN
0017                  JCEVB=(JNODE-1)*NCDOFN+JDOFN
0018                  CESTIF(ICEVB,JCEVB)=ESTIF(IEVAB,JEVAB)
0019               40 CONTINUE
0020                  KSPARE=JCEVB+1
0021                  CESTIF(ICEVB,KSPARE)=-CMATX(IEVAB,JNODE)*WLUMP
0022               30 CONTINUE
0023               20 CONTINUE
                C
0024                  ISPARE=ICEVB+1
0025                  DO 50 JNODE=1,NNODE
0026                  DO 60 JDOFN=1,NDOFN
0027                  JEVAB=(JNODE-1)*NDOFN+JDOFN
0028                  JCEVB=(JNODE-1)*NCDOFN+JDOFN
0029                  CESTIF(ISPARE,JCEVB)=-CMATX(JEVAB,INODE)*WLUMP
0030               60 CONTINUE
                C
0031                  JSPARE=JCEVB+1
0032                  CESTIF(ISPARE,JSPARE)=-EPSTF(INODE,JNODE)*DTIME*TIMEX
                                            *WLUMP*WLUMP
0033               50 CONTINUE
0034               10 CONTINUE
0035                  WRITE(1) CESTIF
0036             1000 CONTINUE
                C
0037                  RETURN
0038                  END
```

D.5 SUBROUTINE SMOOTH

After the first call of Subroutine FRONT, the accumulated displacements and excess pore pressures, stored in the "global" array CTDISP, are incorrect. In order to suit with the cumulative structure of FRONT, these values are set back to their original ones, by subtracting the corresponding displacement and excess pore pressure increments found current time interval, before calling Subroutine FRONT a second time. The "smoothing" process is then carried out according to Eqs. 6.10.

Step	
5	Store components of "smoothing" matrix C in form of data declaration.
6	Loop over total number of nodal points.
7 - 8	Initialize array SMPWP containing the "smoothed" pore pressures.
9	Loop over each degree of freedom.
10	Establish the position of the "displacement" degree of freedom in the "global" array.
11	Subtract the increment of displacement found during the current time interval from the accumulated displacements CTDISP.
12	Establish the position of the "pore pressure" degree of freedom in the "global" array.
13	Store the actual excess pore pressure to be "smoothed" in array EXPWP.
14	Subtract the increment of excess pore pressure found during the current time interval from the accumulated excess pore pressure CTDISP.
15	End the loop on total number of nodal points.

16	Loop over elements.
17	Loop over each "INODE" of each element.
18 - 19	Locate the nodes in common and increment the array SMPWP(NLOCA,2) by 1 each time.
20 - 30	Loop over nodes and perform the matrix multiplication $M*\pi$ for each element.
24	End the loop on elements.
25 - 30	Calculate an average "smoothed" value for nodes in common, then store all "smoothed" excess pore pressures into array CFIXED as assigned fixities for the second call of FRONT.

```
C
C
0001        SUBROUTINE SMOOTH(CASDIS,CFIXED,CTDISP,EXPWP,IFFIX,LNODS,
     .                        MELEM,MCTOTV,MPOIN,NELEM,NCDOFN,NDOFN,
     .                        NNODE,NPOIN,SMPWP,ICFFIX)
C*****************************************************************************
C
C*** SMOOTHS THE PORE PRESSURES ASSUMING LINEAR VARIATION
C
C*****************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        REAL*4 C
0004        DIMENSION C(8,8),CASDIS(MCTOTV),CFIXED(MCTOTV),CTDISP(MCTOTV),
     .            EXPWP(MPOIN),IFFIX(MCTOTV),LNODS(MELEM,9),
     .            SMPWP(MPOIN,2),ICFFIX(MCTOTV)
0005        DATA C/
     .          2., 0.,-2.,-1., 0.,-1.,-2., 0.,
     .          4., 4., 4., 2., 0., 0., 0., 2.,
     .         -2., 0., 2., 0.,-2.,-1., 0.,-1.,
     .          0., 2., 4., 4., 4.,-2., 0., 0.,
     .          0.,-1.,-2., 0., 2., 0.,-2.,-1.,
     .          0., 0., 0., 2., 4., 4., 4., 2.,
     .         -2.,-1., 0.,-1.,-2., 0., 2., 0.,
     .          4., 2., 0., 0., 0., 2., 4., 4./
C
C*** SET BACK DISPL.TO THE PREVIOUS TIME STEP & EXTRACT THE NEW XSPWP
C
0006        DO 10 IPOIN=1,NPOIN
0007        SMPWP(IPOIN,1)=0.0
0008        SMPWP(IPOIN,2)=0.0
0009        DO 20 IDOFN=1,NDOFN
0010        ICTOTV=(IPOIN-1)*NCDOFN+IDOFN
0011     20 CTDISP(ICTOTV)=CTDISP(ICTOTV)-CASDIS(ICTOTV)
0012        ISPARE=ICTOTV+1
0013        EXPWP(IPOIN)=CASDIS(ISPARE)
0014        CTDISP(ISPARE)=CTDISP(ISPARE)-CASDIS(ISPARE)
0015     10 CONTINUE
C
C*** LOOP OVER THE ELEMENTS
C
0016        DO 1000 IELEM=1,NELEM
0017        DO 30 INODE=1,NNODE
0018        NLOCA=IABS(LNODS(IELEM,INODE))
0019        SMPWP(NLOCA,2)=SMPWP(NLOCA,2)+1
0020        DO 30 JNODE=1,NNODE
0021        NPOSI=IABS(LNODS(IELEM,JNODE))
0022        SMPWP(NLOCA,1)=SMPWP(NLOCA,1)+C(INODE,JNODE)*EXPWP(NPOSI)
0023     30 CONTINUE
0024   1000 CONTINUE
C
C*** AVERAGE THE COMMON VALUES AND LOAD THE NEW FIXED XSPWP
C
0025        DO 40 IPOIN=1,NPOIN
0026        IF(IPOIN.LE.3) GO TO 40
0027        NLOCA=(IPOIN-1)*NCDOFN+NCDOFN
0028        IF(IFFIX(NLOCA).EQ.1)GO TO 40
0029        CFIXED(NLOCA)=SMPWP(IPOIN,1)/(6*SMPWP(IPOIN,2))
0030     40 CONTINUE
0031        RETURN
0032        END
```

D.6 SUBROUTINE SPLIT

The role of this subroutine is to extract fluid variables from skeleton counterparts for compatibility with the structure of Subroutine RESIDU when calculating the unbalanced load.

Step	
4	Rewind disk file 12 prior to reading coupling matrix CMATX.
5	Initialize counter ISTOP which determines the number of negative excess pore pressure.
6	Loop over the total number of nodal points.
7	Loop over each degree of freedom.
8 - 9	Establish the "local" and "global" position of the "displacement" degree of freedom.
10 - 11	Store increments and accumulated displacements in arrays ASDIS and TDISP respectively.
12	End the loop on the number of degrees of freedom.
13	Establish the "global" position of the "pore pressure" degree of freedom.
14	Convert the excess pore pressure to the real value by multiplying by weighting factor W.
15	Increment counter ISTOP by one whenever the excess pore pressure is negative.
16	End the loop over total number of nodal points.
17	Loop over elements.
18	Read from disk file 12 the coupling matrix CMATX and the hydraulic "stiffness" EPSTF.
19	Loop over each "INODE" of current element.
20	Loop over each degree of freedom of the current "INODE".

21 - 22	Establish the "local" and "global" position of the "displacement" degree of freedom.
23 - 27	Calculate the hydraulic load contribution according to Eq. 3.2c.
28	Calculate the effective "equivalent" load which should be carried by the skeleton for the current excess pore pressure.
29	End the loop on degrees of freedom.
30	End the loop on elements.

```
C
C
0001        SUBROUTINE SPLIT(ASDIS,CASDIS,CELOAD,CTLOAD,CTDISP,EXPWP,
     .                       FCLOAD,ISTOP,LNODS,MCEVAB,MCTOTV,MELEM,
     .                       MEVAB,MPOIN,MTOTV,NCDOFN,NCEVAB,NCTOTV,NDOFN,
     .                       NELEM,NNODE,NPOIN,NTOTV,TDISP,TLOAD,WLUMP)
C*******************************************************************************
C
C*** SPLITS THE RESULTS OF FRONT INTO HYDRAULIC & MATERIAL COMPONENTS
C
C*******************************************************************************
0002        IMPLICIT REAL*8(A-H,O-Z)
0003        DIMENSION ASDIS(MTOTV),CASDIS(MCTOTV),CELOAD(MELEM,MCEVAB),
     .            CMATX(18,9),CTLOAD(MELEM,MCEVAB),CTDISP(MCTOTV),
     .            EPSTF(9,9),EXPWP(MPOIN),FCLOAD(MELEM,MEVAB),
     .            LNODS(MELEM,9),TDISP(MTOTV),
     .            TLOAD(MELEM,MEVAB)
0004        REWIND 12
0005        ISTOP=0
C
C*** SPLIT DISPLACEMENT FROM EXCESS PORE WATER PRESSURE
C
0006        DO 10 IPOIN=1,NPOIN
0007        DO 20 IDOFN=1,NDOFN
C
0008        ITOTV=(IPOIN-1)*NDOFN+IDOFN
0009        ICTOTV=(IPOIN-1)*NCDOFN+IDOFN
0010        ASDIS(ITOTV)=CASDIS(ICTOTV)
0011        TDISP(ITOTV)=CTDISP(ICTOTV)
0012     20 CONTINUE
C
0013        ISPARE=ICTOTV+1
0014        EXPWP(IPOIN)=CTDISP(ISPARE)*WLUMP
0015        IF(EXPWP(IPOIN).LT.0) ISTOP=ISTOP+1
0016     10 CONTINUE
C
C*** SPLIT FORCE FROM HEAD
C
0017        DO 30 IELEM=1,NELEM
0018        READ(12) CMATX,EPSTF
0019        DO 30 INODE=1,NNODE
0020        DO 40 IDOFN=1,NDOFN
0021        IEVAB=(INODE-1)*NDOFN+IDOFN
0022        ICEVAB=(INODE-1)*NCDOFN+IDOFN
C
C***TAKE OFF FROM TOTAL LOAD THE FLUID PART CONTRIBUTION
C
0023        FCLOAD(IELEM,IEVAB)=0.0
0024        DO 50 JNODE=1,NNODE
0025        LOCA=IABS(LNODS(IELEM,JNODE))
0026        FCLOAD(IELEM,IEVAB)=FCLOAD(IELEM,IEVAB)+CMATX(IEVAB,JNODE)*
     .                          EXPWP(LOCA)
0027     50 CONTINUE
0028        TLOAD(IELEM,IEVAB)=CTLOAD(IELEM,ICEVAB)+FCLOAD(IELEM,IEVAB)
0029     40 CONTINUE
0030     30 CONTINUE
0031        RETURN
0032        END
```

REFERENCES

1. ARAVIN, V.I., and NUMEROV, S.N., 1965. Theory of fluid flow in undeformable porous media. Israel Programme for Scientific Translation, Jerusalem, 511 p.

2. ARGYRIS, J.H., and SCHARPF, D.W., 1972. Methods of elasto-plastic analysis. Journal of Applied Mathematics and Physics, Vol. 23, pp. 517-552.

3. ARMEN, H., LEVINE, H.S., and PIFCO, A.B., 1972. Plastic theory and finite element applications. In Advances in Computational Methods in Structural Mechanics and Design edited by Oden, J.T., Clough, R.W., and Yamamoto.

4. BARLOW, J., 1976. Optimal stress locations in Finite Element Models. International Journal for Numerical Methods in Engineering, Vol. 10, pp. 243-251.

5. BELKEZIZ, A., MAGNAN, J.P., 1982. Analyse numérique de la consolidation bidimensionnelle des sols elastoplastiques. Bulletin de liaison du Laboratoire Central des Ponts et Chaussées, No. 115, Juillet 1982.

6. BIOT, M.A., 1941a. General theory of three dimensional consolidation. Journal of Applied Physics, Vol. 12, pp. 155-164.

7. BIOT, M.A., 1941b. Consolidation settlement under a rectangular load distribution. Journal of Applied Physics, Vol. 12, pp. 426-430.

8. BIOT, M.A., and CLINGAN, F.M., 1941c. Consolidation settlement of a soil with an impervious top surface. Journal of Applied Physics, Vol. 12, pp. 578-581.

9. BISHOP, A.W., 1966. The strength of soils as engineering materials. Geotechnique, Vol. 21, No. 2, pp. 168-172.

10. BURLAND, J.B., 1965. The yielding and dilatation of clay; correspondance. Geotechnique, Vol. 15, No. 2, pp. 211-219.

11. CARSLAW, H.S., and JAEGER, J.C., 1959. Conduction of heat in solids. Second edition, Clarendon Press Oxford, 510 p.

12. CASAGRANDE, A., 1940. Seepage through dams. In contribution to Soil Mechanics 1925-1940. Boston Society of Civil Engineers.

13. CASTIGLIANO, A., 1879. Teoria dell' equilibrio dei sistemi elastici. Tesi di laurea, Politecnico di Torino.

14. CAUCHY, 1827. Sur les relations qui existent dans l'etat d'equilibre d'un corps solide ou fluide, entre les pressions ou tensions et les forces acceleratrices. Extraits de Mathematiques, Vol. 2, No. 7, pp. 141-145.

15. CEDERGEN, H.R., 1967. Seepage, drainage and flow nets, John Wiley, New York, 489 p.

16. CHANG, C.S., DUNCAN, J.M., 1983. Consolidation analysis for partly saturated clay by using an elastic-plastic effective stress-strain model. International Journal for Numerical and Analytical Methods in Geomechanics, Vol. 7, pp. 39-35.

17. CHEUNG, Y.K., YEO M.F., 1979. A practical Introduction to Finite Element Analysis. Pitman Publishing limited, London, 180 p.

18. CIVIDINI, A. and GIODA, G., 1982. Compatible and equilibrium Finite Elements for soil consolidation: a preliminary comparison. Numerical Methods in Geomechanics. Edited by Eisenstein, Edmonton.

19. COULOMB, C.A., 1776. Essai sur l'application des règles de maximis et minimis a quelques problemes de statique relatifs a l'architecture. Memoires de l'Academie, Recits des Presents Savants Etrangers, Vol 7, pp. 342-382.

20. CRYER, C.W., 1963. A comparison of the three-dimensional theories of Biot and Terzaghi. Quaterly Journal of Mechanics and Applied Mathematics, Vol. 16, pp. 401-412.

21. DARCY, H., 1856. Les fontaines publiques de la ville de Dijon. Dunod, Paris.

22. DAVIDSON, H.L., and CHEN, W.F., 1974. Elasto-plastic large deformation response of clay to footing loads. Report No. 355-18, Department of Civil Engineering, Lehigh University, Bethlehem, Pensylvannia.

23. DESAI, C.S., and ABEL, J.F., 1972. Introduction to the Finite Element Method. Van Nostrand Reinhold Company. New York.

24. DESAI, C.S., and JOHNSON, L.D., 1972. Evaluation of two finite element formulations for one dimensional consolidation. Computers and Structures; Vol. 2, pp. 469-486.

25. DESAI, C.S., and CONTRACTOR, D.N., 1976. Finite element analysis of flow, diffusion, and salt water intrusion in porous media. Formulations and computational algorithm in finite element analysis. Edited by K.G. Bathe, J.T. Oden, W. Wunderlich.

26. DRUCKER, D.C., 1951. A more fundamental approach to plastic stress-strain relations. American Society of Mechanical Engineers, ASME Proceedings of the First U.S National Congress for Applied Mechanics, New York, pp. 487-491.

27. DRUCKER, D.C., 1961. On stress-strain relations for soil and load carrying capacity. Proceedings of the First International Conference on Mechanics of soil-vehicle systems, Torino. Edizioni Minerva Tecnica, pp. 15-23.

28. DRUCKER, D.C., 1966. Concept of path independence and material stability for soils. In Kravtchenko, J., and Sirieys, P.M. (Editors), Rheologie et mecanique des sols, Proceedings of International Union for Theoretical and Analytical Mechanics Symposium, Grenoble, Springer, Berlin, pp. 23-43.

29. DRUCKER, D.C., GIBSON, R.E., and HENKEL, D.J., 1957. Soil Mechanics and work-hardening theories of plasticity. American Society of Civil Engineers, ASCE Journal for Soil Mechanics and Foundation Division, Vol. 122, pp. 338-346.

30. DRUCKER, D.C and PRAGER, W., 1952. Soil Mechanics and Plastic analysis or Limit design. Quarterly Journal of Applied Maths, Vol. 10, pp. 157-165.

31. ERGATOUDIS, I., IRONS, B.M., and ZIENKIEWICZ, O.C., 1968. Curved isoparametric Quadrilater elements for finite element analysis. International Journal of Solids and Structures, Vol. 4, pp. 31-42.

32. FINLAYSON, B.A., 1972. The Method of Weighted Residual and Variational Principles. Academic Press, New-York, 412 p.

33. FINN, W.D. L., 1967. Finite element analysis of seepage through dams. American Society of Civil Engineers, ASCE Journal for Soil Mechanics and Foundation Division, Vol. 93 (SM6), pp. 41.

34. FINN, W.D.L., VAROGLU, E., 1976. Numerical solutions of unconfined seepage problem. Numerical Methods in Geomechanics edited by C.S. Desai, Vol. III, pp. 1465-1474.

35. FUSCO, A., 1985. Continuum Mechanics and Finite Element Numerical Solutions in Geotechnique. Ph.D Thesis in partial fulfillment, Deptartment of Civil Engineering, University of Ottawa, 316 p.

36. GHABOUSSI, J., and WILSON, E.L., 1973. Flow of compressible fluid in porous elastic media. International Journal for Numerical Methods in Engineering, Vol. 5, pp. 419-442.

37. GIBSON, R.E., and McNAMEE, J., 1963. A three dimensional problem of consolidation of a semi-infinite clay stratum. Quarterly Journal of Mechanics and Applied Mathematics, Vol. 16, pp. 34-50.

38. GIODA, G., and NOVA, R., 1982. Un' analisi pseudobidimensionale della consolidazione in argilla con permeabilita anisotropa. RIG Riv. Ital. di Geot. Edizioni Scientifiche Italiane, Napoli, anno XVI, No. 1, Genn.-Marzo 1982.

39. GURTIN, M., 1964. Variational principles for linear elastodynamics. Archives for Rational Mechanics and Analysis, Vol. 16, pp. 34-50.

40. HARR, M.E., 1962. Ground water and seepage. Mc. Graw-Hill, New York, 315 p.

41. HENKEL, D.J., 1960. The shear strength of saturated remolded clays. American Society of Civil Engineers, ASCE Conference on Shear Strength of Cohesive Soils, University of Colorado, June 1960, pp. 533.

42. HERRMANN, L.R., 1976. Improved stress calculations for simple quadrilateral elements. Computers and Structures, Pergamon Press, Vol. 6, pp. 141-148.

43. HILL, R., 1956. On the problem of uniqueness in the theory of a rigid-plastic solid. Journal of Mechanics and Physical Solids, Vol. 4, No. 247, pp. 153 and Vol. 5, No. 1, pp. 302.

44. HINTON, E., and CAMPBELL, J.S., 1974. Local and global smoothing of discontinuous finite element functions using a least square method. International Journal for Numerical Methods in Engineering, Vol. 8, pp. 461-480.

45. HINTON, E., SCOTT, F.C., and RICKETTS, R.E., 1975. Short communications. Local Stress smoothing for parabolic isoparametric elements. International Journal for Numerical Methods in Engineering, Vol. 9, pp. 235-256.

46. HVORSLEV, M.J., 1937. Ueber die Festigkeitseigenschaften gestoerter bindiger Boeden (On the physical proprieties of disturbed cohesive soil). Ingeniervidenskabelige Skrifter, A. No. 45, 159 p.

47. HWANG, C.T., MORGENSTERN, N.R., and MURRAY, D.W., 1971. On solution of plain strain consolidation problems by Finite Elements Methods. Canadian Geotechnical Journal, pp. 109-118.

48. HWANG, C.T., MORGENSTERN, N.R., and MURRAY, D.W., 1972. Application of the Finite Element Method to consolidation problems. Proceedings and Symposium on the Application of Finite Element Method in Geomechanical Engineering, Vicksburg, USA, pp. 739-765.

49. IRONS, B.M., 1970. A frontal solution for Finite Element analysis. International Journal for Numerical Methods in Engineering, Vol. 2, pp. 5-32.

50. JOSSELIN DE JONG, G. DE., 1957. Application of the stress function to Proceedings of the Fourth International Conference in Soil Mechanics London, Vol. 1, pp. 320-323.

51. KRIEG, R.D., and KRIEG, D.B., 1977. Accuracies of numerical solution methods for the elastic-perfectly plastic mode. American Society of Mechanical Engineers, ASME Journal of Pressure Vessel Technology, Vol. 99, No. 4, pp. 510-515.

52. LEWIS, R.W., ROBERTS, G.K., and ZIENKIEWICZ, O.C., 1976. A non-linear flow and deformation analysis of consolidated problems. Numerical Methods in Geomechanics edited by C.S. Desai, Vol. II, pp. 1106-1118.

53. MANDEL, J., 1957. Consolidation des couches d'argile. Proceedings of the Fourth International Conference in Soil Mechanics and Foundation Engineering, Vol. 1, pp. 360-367.

54. McNAMEE, J., and GIBSON, R.E., 1960a. Plane strain and axial symmetric problems of consolidation of a semi-infinite clay stratum. Quarterly Journal of Mechanics and Applied Mathematics, Vol. 13, pp. 210-227.

55. McNAMEE, J., and GIBSON, R.E., 1960b. Displacement function and linear transforms applied to diffusion through porous elastic media. Quaterly Journal of Mechanics and Applied Mathematics, Vol. 13, pp. 98-111.

56. MELAN, E., 1938. Ingenieur-Archiv, Vol. 9, 116 p.

57. MOHR, O., 1882. Uber die Darstellung des Spannungs zustandes und des Deformationzustandes eines Korperelements. Zivilingenieur, 28: 113-156

58. MROZ, Z., NORRIS, V.A., and ZIENKIEWICZ, O.C., 1978. An anisotropic hardening model for soils and its application to cyclic loading. International Journal for Numerical and Analytical Methods in Geomechanics, Vol. 2, pp. 203-221.

59. NAYLOR, D.J., and PANDE, G.N., 1981. Finite elements in geotechnical engineering. Pineridge Press, Swansea, U.K, 245 p.

60. NELSON-SKORNYAKOV, F.B., 1949. Seepage in homogeneous media. Gosudaretvennoe Izd. Sovetskaya Nauka, Moscow.

61. NEUMAN, S.P., and WITHERSPOON, P.A., 1970. Finite element method of analysing steady seepage with a free surface. Water Resources Research, Vol. 6, No. 3, June 1970, pp. 889-896.

62. ODEN, J. T., 1972. Finite elements of non linear continua. Mc. Graw-Hill, New York, 430 p.

63. OWEN, D.R.J., and HINTON, F., 1980. Finite elements in plasticity: theory and practice. Pineridge Press Limited, Swansea, U.K, 594 p.

64. PARRY, R.H.G., 1956. Strength and deformational clay. Ph.D. Thesis, University of London.

65. PARRY, R.H.G., 1960. Triaxial compression and extension test on remoulded saturated clay. Geotechnique, Vol. 10, pp. 166-180.

66. POLUBARINOVA-KOCHINA, P.Ya., 1962. The theory of ground water movement. Princeton University Press, Princeton, New Jersey, 613 p.

67. POULOS, H.G. and DAVIS, E.H., 1974. Elastic solutions for Soil and Rock Mechanics. John Wiley and Sons, New York, 411 p.

68. REED, M.B., 1982. Techniques for the analysis of seepage in porous elastic media using parabolic isoparametric elements. Proceedings of the International Conference in Finite Elements Methods, Gordon & Breach, New York, pp. 534-539.

69. REED, M.B., 1984. An investigation of numerical errors in the analysis of consolidation by finite elements. International Journal for Numerical and Analytical Methods in Geomechanics, Vol. 8, pp. 243-257.

70. RENDULIC, L., 1936, Porenziffer und Porenwasser Drueck in Tonen. Der Bauingenieur, 17(51/53), pp. 559-564.

71. ROSCOE, K.H., SCHOFIELD, A.N., and WROTH, C.P., 1958. On the yielding of soils. Geotechnique, Vol. 8, pp. 22-53.

72. ROSCOE, K.H., and POOROOSHAB, H.B., 1963. A theoretical and experimental study of strains in triaxial compression test on normally consolidated clays. Geotechnique, Vol. 13, No. 1, pp. 12-38.

73. ROSCOE, K.H., SCHOFIELD, A.N., and THURAIRAJAH, A., 1963. Yielding of soils in states wetter than critical. Geotechnique, Vol. 13, No. 3, pp. 211-240.

74. ROSCOE, K.H., and BURLAND, G.B., 1968. On the generalized stress-strain behaviour of wet clay. In Engineering Plasticity edited by J. Heyman and F.A. Lekie, Cambridge University Press, London, pp. 535-609.

75. SANDHU, R.S., 1968. Fluid flow in saturated soil. Ph.D., University of California, Berkeley.

76. SANDHU, R.S., and WILSON, E.L., 1969. Finite element analysis of seepage in elastic media. American Society of Civil Engineers, Proceedings of the Mechanical Division, Vol. 95, EM3.

77. SANDHU, R.S., LUI, H., and SINGH, K.J., 1977. Numerical performance of some finite element schemes for analysis seepage in porous elastic media. International Journal for Numerical and Analytical Methods in Geomechanics, Vol. 1, No. 3, pp. 177-194.

78. SCHIFFMAN, R.L., CHEN, A.T.F., and JORDAN, J.C., 1969. An analysis of consolidation theories. American Society of Civil Engineers, ASCE Journal for Soil Mechanics and Foundation Division, Vol. 95, No. SM1, pp. 285-311.

79. SCHOFIELD, A.N, and WROTH, C.P., 1968. Critical state soil mechanics. Mc. Graw-Hill, New York.

80. SCHREYER, H.L., KULAK, R.F., and KRAMER, J.M., 1979. Accurate numerical solutions for elastic-plastic models. American Society of Mechanical Engineers, Transactions, ASME Journal of Pressure Vessel Technology, Vol. 101, No. 3, pp. 226-234.

81. SIRIWARDANE, H.J., and DESAI, C.S., 1981. Two numerical schemes for non linear consolidation. International Journal for Numerical and Analytical Methods in Geomechanics, Vol. 7, pp. 405-426.

82. SIRIWARDANE, H.J., and DESAI, C.S., 1983. Computational procedures for non linear three dimensional analysis with some advanced constitutive laws. International Journal for Numerical and Analytical Methods in Geomechanics, Vol. 7, pp. 143-171.

83. SMITH, I.M., and HOBBS, R., 1976. Biot analysis of consolidation beneath embankments. Geotechnique, Vol. 26, No. 1, pp. 149-171.

84. SMITH, I.M., 1982. Programming the Finite Element Method with application to Geomechanics. John Wiley and sons, London, 351 p.

85. TAYLOR, D.W., 1948. Fundamentals of soil mechanics. Wiley & Sons, New York, pp. 345-347.

86. TAYLOR, R.L., and BROWN, C.B., 1967. Darcy flow solution with a free surface. American Society of Civil Engineers, ASCE Journal of the Hydraulic Division, Vol. 93, (HY2), pp. 25.

87. TERZAGHI, K., 1943. Theoretical soil mechanics. Wiley, New York, 729 p.

88. TILLERSON, J.R., STRICKLIN, J.A., and HAISLER, W.E., 1973. Numerical methods for the solution of non-linear problems in structural analysis. In Numerical solution of non-linear problems edited by Hartnung, R.F. American Society of Civil Engineers, ASME Special Publication, Vol. 6, November 1973, pp. 67-101.

89. TRESCA, H., 1868. Memoire sur l'ecoulement des corps solides. Mémoires présentés par divers Savants à l'académie des Sciences, Vol. 18, pp. 733-799.

90. VALLIAPPAN, S., LEE, I.K., and BOONLUALOHR, P., 1974. Finite element analysis of consolidation problems. Finite element methods in flow problems, edited by J.T. Oden, O.C. Zienkiewicz, R.H. Gallagher and C. Taylor, University of Alabama at Huntsville Press, Alabama.

91. VON MISES, R., 1928. Mechanik der plastischen Formaenderung von Kristallen. Z. Angew. Math. Mech., Vol. 8, pp. 161-185.

92. WILSON, C.P., 1965. Structural analysis of axisymmetric solids. American Institute of Aeronautics and Astronautics Journal, Vol. 3, pp. 2269-2274.

93. YAMADA, Y., YISHIMURA, N., and SAKURAI, T., 1968. Plastic stress-strain matrix and its application for the solution of elasto-plastic problems by finite element method. International Journal of Mechanical Sciences, Vol. 10, pp. 343-354.

94. YOKOO, Y., YAMAGATA, K., and NAGAONKA, H., 1971a. Finite element method applied to Biot's consolidation theory. Soils and Foundations, Vol. II, No.1, pp. 26-29.

95. ZIENKIEWICZ, O.C., and NAYLOR, D.J, 1971. Adaptation of Critical State Soil Mechanics for use in Finite Elements. Contribution to Roscoe Memorial Symposium, Cambridge.

96. ZIENKIEWICZ, O.C., VALLIAPPAN, S., and KING, I.P., 1969. Elastoplastic solutions of engineering problems: initial stress finite element approach. International Journal for Numerical Methods in Engineering, Vol. 1, pp. 75-100.

97. ZIENKIEWICZ, O.C., 1977. The Finite Element Method. Third Ed., Mc. Graw-Hill Company Ltd., U.K, 787 p.