A Computational Framework

for

Dynamic Soil-Structure Interaction Analysis

by

Kandiah Sribalaskandarajah

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Washington

1996

Approved by Gregory R. Miller

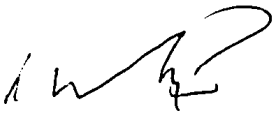
(Chairperson of Supervisory Committee)

Program Authorized

to Offer Degree Civil Engineering

Date March 15, 1996

In presenting this dissertation in partial fulfillment of the requirements for the Doctoral degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of this dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Request for copying or reproduction of this dissertation may be referred to University Microfilms, 1490 Eisenhower Place, P.O. Box 975, Ann Arbor, MI 48106, to whom the author has granted "the right to reproduce and sell (a) copies of the manuscript in microform and/or (b) printed copies of the manuscript made from microform."

Signature 

Date 15 March 1996.

University of Washington

Abstract

A Computational Framework for Dynamic Soil-Structure Interaction Analysis

by Kandiah Sribalaskandarajah

Chairperson of Supervisory Committee: *Professor Gregory R. Miller &*
Professor Sunirmal Banerjee
Civil Engineering

This dissertation presents: i) a simple but general soil model; and ii) a general computational framework, enabling the development of efficient interactive software for the analysis of dynamic soil-structure interaction problems.

With regard to the model, a general, non-associative, elasto-plastic soil model has been developed. An alternate volume change behavior of soils characterizing isotropic compression and extension with exponential and power laws respectively is proposed and included in the model. Distinct volumetric and distortional (shear) mechanisms are used based on double hardening concepts. An optimization scheme is used to estimate the model parameters using experimental results for a given soil. Essential features of sands such as path dependent dilatancy, hardening and softening, liquefaction, initial and induced anisotropy and that of clay are reproduced with unprecedented accuracy for a model of this simplicity.

With regard to the computational framework, a general computational framework is presented based on object-oriented programming for the analysis of soil-structure interaction problems. Based on this framework, a prototype finite element program (SAND) has been

developed implementing an iterative direct solution scheme, and a new kind of efficient interface element has been derived. Illustrative examples have been presented to verify the implementation of the prototype and to demonstrate the new capabilities of the interactive analysis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

The author is greatfully indebted to late Professor Sunirmal Banerjee for the valuable guidance, encouragement, continued support. Words fail to acknowledge the amicable treatment and kindness extended to the author by Professor Banerjee. The author is also appreciative of Professor Banerjee's ideas and assitance in the work throughout his study.

Heartfelt gratitude is extended to Professor Gregory R. Miller, not only for serving as the chairperson of the supervisory committee, but also for his continued supervision and interest and his devotion in the laborius task of editing the draft chapters. The author especially thanks him for being helpful and friendly.

The author also wishes to express his appreciation to Professors Robert D. Holtz, Dennis P. Lettenmaier, and Linda G. Shapiro for serving on his supervisory committee.

Finally, the author thanks his wife, parents, brothers and friends for their continued encouragement and patient endurance.

Chapter 1

INTRODUCTION

The scope of the geotechnical engineering design traditionally has been limited to pure geotechnical tasks, with limited analytical consideration of the interactions of geomaterials with structural components, known as soil-structure interaction. This situation has been changing with increased usage of computational mechanics, facilitating the potential inclusion of the effects of soil-structure interaction in increasing numbers of geotechnical engineering designs. Also, the abundance of material models to describe the behavior of soils can enable realistic representations of geomaterial behavior in such designs. However, the lack of simple constitutive models and convenient analysis tools has seriously limited the use and applicability in practical geotechnical problems involving complex static and dynamic loading conditions. To address these issues, a simple general constitutive model for soils and a general computational framework for dynamic analysis of soil-structure interaction problems are presented in this dissertation.

1.1 General

The analysis of soil-structure interaction problems can be classified broadly into two categories, static and dynamic, reflecting the analysis scheme employed. Although problems with dynamic loading conditions require a dynamic analysis, problems with static loading conditions can be solved by both static and dynamic solution schemes. Dynamic analysis is computationally more involved, and so static analysis is usually employed for problems with static loading conditions. However, most soil-structure interaction problems, such as

construction of soil-nailed walls, sheet piles, and foundations, are of an evolving nature. In addition, the behavior of the soil is time dependent due to rate dependent constitutive relations and the diffusion of pore fluids. Therefore, for most soil-structure interaction problems a simple static analysis may prove to be inadequate, requiring a dynamic analysis even for problems with essentially static loading conditions. This underscores the practical significance of dynamic analysis of soil-structure interaction problems. Analytic and numerical analysis of these interaction problems require knowledge of geometry, loading conditions, governing equations and constitutive models of the relevant components: soil; structure; and interface. Of these, the governing equations and constitutive models form the basis of the analysis used in the design, and the geometry and loading conditions serve as input parameters to the design. In the following subsections a general background of soil modeling, in view of the governing equations and constitutive models is presented.

1.1.1 Governing Equations

Soil, in general, is an assemblage of particles with different sizes and shapes forming a soil skeleton. The pores are filled with various fluids and gas/air, thus constituting a multi-phase material. For the analysis of most common geomechanics problems, the soil can be viewed as a one-phase, two-phase, or three-phase material depending on whether the soil in question is dry, fully saturated, or partially saturated, respectively. The mixture theories developed by Truesdell and Toupin (1960) form a general theoretical framework for the analysis of multi-phase media. The resulting equations capture the interactions between different phases, and are quite complex.

Even before the advent of such sophisticated theories, Terzaghi (1943) succeeded in explaining the interaction between the different phases with his "effective stress principle". In this principle, the stresses carried by the solid phase are termed as "effective stresses" while the stresses carried by the fluid phase are known as "pore-fluid pressures". This formed the basis for his one-dimensional consolidation theory, which was later generalized by Biot (1955) to model the true three dimensional nature of the diffusion process. In so doing,

Biot made the following assumptions: the soil is isotropic; the stress-strain relationship is linear elastic; the strains are small; the water contained in the pores is incompressible; the water may contain air bubbles; and the water flow through the porous skeleton is according to Darcy's law. However, the stress-strain behavior of real soil is strongly nonlinear, anisotropic, hysteretic, and path-dependent. In addition the flow of water through the porous skeleton may not always obey Darcy's law. The governing equations derived by Prevost (1987) using mixture theories overcome these deficiencies. These governing equations are more general in three important aspects: i) accommodating non-linear rate dependent constitutive models; ii) describing the dynamic behavior of multi-phase materials; and iii) including general flow conditions. The formulation of the numerical scheme and computational framework presented in this dissertation is based on these general constitutive laws; however, the implementation and testing of the results are done for models without pore-pressure as a variable.

1.1.2 Constitutive Models

Analysis of soil-structure interaction problems requires the understanding of the behaviors of structure, interface, and soil. The behavior of the structure is often relatively well defined compared to that of the soil, while the behavior of the interface could range from completely smooth to completely adhesive. The behavior of soil, influenced by various factors such as the shape and the size of the individual soil particles, and the configuration of the soil structure; the inter-granular stresses (effective stresses), the stress history, the degree of saturation, and the loading path, is quite complex. Numerous constitutive models have been proposed based on theories ranging from simple elasticity to sophisticated concepts of plasticity to capture the real behavior of soils.

Use of sophisticated constitutive relations enhances the predictive value of the analyses of soil-structure interaction problems. However, from the standpoint of practical usefulness, the application of such sophisticated models has a down side. This is especially true for general models, for which the high number of model parameters and internal variables

necessary for describing the complexities of material behavior tends to make it difficult to identify the parameters with physically meaningful material constants. Therefore, one can not find these model parameters by devising simple laboratory experiments or by using traditional testing methods. Practical use of a certain constitutive model, therefore, depends on the ease with which model parameters can be determined for a given soil. On the other hand, the existing simpler models with meaningful physical parameters often fail to reproduce some or most of the essential features of the behavior of soils, particularly that of sands, such as mean-stress and loading-path-dependent dilatancy, hardening and softening, liquefaction, and initial and induced anisotropy, which are quite complex. Due to this practical need for a simple but general constitutive model, newer constitutive models utilizing complex theories are being continuously investigated. This dissertation presents one such new constitutive model, which is simple but general enough to describe the complex behaviors of geomaterials.

1.2 Numerical Analysis and Computer Applications

Due to the complexity of general constitutive models of soils, geometry, geological strata, and loading conditions, it is difficult, if not impossible, to obtain analytical closed-form solutions for the general governing equations. Consequently, closed form solutions have been found only for simple problems and have very limited use in geotechnical analysis. Therefore, numerical approaches are typically necessary in dealing with practical geotechnical analysis and design. Critical designs of soil-structure interaction problems can be modeled and analyzed numerically with the aid of computers, using numerical techniques such as finite difference method (FDM), finite element method (FEM), and boundary element method (BEM).

Although various techniques and tools are presently available for performing design calculations to estimate stresses, deformations, and stability of geotechnical structures, computational perspectives in geotechnical engineering have changed little since the days when

programs were developed using computers managed by specialized staff, and engineers had minimal direct interaction with computers. This is mainly because the application of computers to geotechnical engineering was viewed with suspicion (Christian 1991), as geotechnical engineers traditionally have placed great emphasis on judgment. Nevertheless, the increased understanding of the behavior of geomaterials and advances in numerical methods has expanded the use of computer programs in geomechanics. Early applications of the finite element scheme to find the deformation pattern in an earth dam by Clough and Woodward (1966) and an iterative approach to the analysis of slope stability by Whitman and Bailey (1966) are noteworthy. Computer use in geotechnical engineering has increased ever since, but the developments in geotechnical software are not necessarily keeping pace with the rapid evolution of computer hardware or with changes in programming environments and styles.

Most of the present geotechnical software has been developed from a batch-mode perspective with minimal accommodation for interactive use. The capabilities of these software do not reflect the advances in areas such as artificial intelligence, expert systems, relational databases, solid modeling, computer graphics and graphical user interfaces. Also, the trend has been to develop special purpose programs, such as geographic information systems (Robinson & Frank 1987), interpretation of data from geographical logging (Bonnet & Dahan 1983, Smith & Baker 1983), correlation of lithologic data between bore-holes (Rehak et al. 1985) for each geotechnical task, without regard for the interactions among these tasks. Ideally, all geotechnical tasks including geophysical exploration, testing, modeling, different analyses, design, and presentation could be integrated into a single computing environment. Creating such an environment would be quite ambitious, requiring a fundamental reevaluation of the basic approach traditionally used in the development of geotechnical software and programming styles. This dissertation presents a restructured approach to the analysis of dynamic soil-structure interaction problems, with strong emphasis on increased user interaction using object-oriented programming concepts.

1.3 Scope of the Study

The development of a general soil model and the development of a computational framework for dynamic soil-structure interaction analysis are the main focuses of this study. In order to illustrate the outcome of the study, a prototype finite element program (SAND) based on the proposed framework is developed and the proposed soil model is implemented. The soil is assumed to be fully saturated in this study.

In regards to the soil model, as noted earlier, numerous constitutive models for soils with varying complexities are available in the geomechanics literature. A brief overview of different major classes of soil constitutive models is presented in Chapter 2. However, most of the general constitutive models are quite complex and their applications in real boundary value problems are very much limited. On the other hand, simpler models fail to capture the complex behavior of real soil. Therefore, a simple constitutive model to describe the general behavior of soils is developed in this study, based on the non-associative elasto-plastic double-hardening concepts. Initially, this model was developed for axisymmetric monotonic loading conditions and the corresponding formulations can be found in the paper by Banerjee et al. (1992). Then, to improve the volumetric behavior of this model, an alternate formulation for volumetric behavior of soils under isotropic loading is presented as in Section (3.1). A detailed treatment of this alternate formulation can be found in the paper by Banerjee et al. (1994). Based on the above formulations, a simple general soil model is developed in the reminder of Chapter (3). This chapter also describes the model in simplified forms for practical situations, to facilitate efficient implementation of this model in a numerical scheme. A numerical scheme based on optimization techniques is presented in Chapter 4 to identify the model parameters for a given soil. Prominent features of the soil behavior are also reproduced and discussed in this chapter. The contents of Chapter 3 and Chapter 4, excluding the contents of Section (3.1), appear in condensed form in the paper by Sribalaskandarajah and Banerjee (1996)

With respect to the computational framework, the scope is primarily restricted to the

design of an object-oriented framework for the dynamic analysis of soil-structure interaction boundary value problems with complex material models. Beyond the complexity of the constitutive models of soils, the question of implementation of such models for practical boundary value problems, such as the finite element formulation outlined in Chapter 5 is of major interest. The incremental iterative approach presented in this chapter is suitable to capture the inelastic and path-dependent behavior of the soil in a dynamic analysis. Complete dynamic response analysis using non-linear material models is computationally intensive; therefore, the analysis of large scale problems can be excessively time consuming on present day personal computers, often requiring supercomputers to be practicable. But, with the rapid evolution of the computer hardware, analysis of large soil-structure interaction problems with complex material models on personal computers will soon become practicable. This necessitates the design of an efficient and convenient to use computational framework. The design of the proposed computational framework and essential details of the implementation of the prototype finite element program are discussed in Chapter 6. The material in Chapters (5) and (6) was presented in an earlier form in the papers by Miller et al. (1992, 1995). Application of this prototype program in small real soil-structure interaction problems is demonstrated in Chapter 7 and Chapter 8 summarizes the findings of the study.

Chapter 2

REVIEW OF SOIL MODELS

In order to place the formulation of the proposed soil model in proper perspective, an overview of important classes of soil constitutive models is presented in this chapter.

2.1 General

Historically, the load-deformation problems that arise in Geotechnical analysis have been solved employing the most simple linear-elastic or rigid-plastic material models. However, soil is a multi-phase material consisting of solids, fluids and gases; hence its mechanical response is highly non-linear, inelastic, rate-dependent and anisotropic. In addition, the strength and stiffness are greatly dependent on the effective confining pressure and the strain levels. The existing geotechnical numerical analysis programs employ constitutive relations ranging from simple elasticity models to complex plasticity theories such as bounding surface plasticity model (Dafalias 1986), with the simple hyperbolic model (Duncan & Chang 1970) being most widely used. Even so, the applicability of these theories is often limited to a narrow range of loadings and the estimation of the model parameters is of prime concern. Many sophisticated models, such as functional-elastic models, hypo-elastic models, hyper-elastic models, visco-elastic models, endochronic models, and elasto-plastic models have been extensively described in the literature. Comprehensive surveys of such constitutive models can be found in the papers edited by Murayama (1985) and in the paper by Scott (1985). The following section briefly discusses the important classes of soil models and the pioneering work in each category.

2.2 An Overview of Soil Models

2.2.1 Functional Elastic Models

In linear-elastic theory, the strain and stress are related by Hooke's Law. Linear elastic theory is appropriate only when the soil is subjected to very small strain. Therefore, in order to describe the non-linear mechanical behavior of the soils, several functional elastic models have been proposed. Joyner and Chen (1975) and Pyke (1979) proposed a multilinear elastic model, where the non-linear stress-strain relation is approximated by a number of linear segments with different slopes. Such models are also called "piece-wise" linear elastic models.

A spline functional model proposed by Desai (1972) and a polynomial functional model proposed by Wong (1971) have also been applied to model non-linear soil behavior. However, the material parameters used in these types of models have no direct physical significance and depend strongly on loading conditions, such as stress level, stress path, and stress history. Konder (1963) observed that the stress-strain plot from a triaxial compression test can be approximated by a hyperbola. Subsequently, Duncan and Chang (1970) used this idea to propose a hyperbolic model to describe the non-linear stress-strain relationship of soil. The hyperbolic model and Ramberg-Osgood model (Ramberg & Osgood 1943) require an initial tangent modulus. This parameter has been studied by Hardin and Drnevich (1972), and Seed and Idriss (1970) among others, and their findings indicate that its magnitude is strongly dependent on confining pressure.

The applicability of the functional elastic models is limited to monotonic loading conditions, since the dependence of the stress-strain relationship on stress path and stress history is ignored. Without any modification, the unloading path predicted by a functional elastic model coincides with the initial loading path. Hence, a different modulus (unloading-reloading modulus) is required to model cyclic behavior. Further, soils usually exhibit hysteretic characteristics during cyclic loading. The Masing law (Masing 1926) is a common approach for taking the hysteretic effect into account. The Masing law reasonably simu-

lates the hysteresis effects of the soil response under cyclic loadings. However, it is virtually impossible to model the path dependence and the dilatant characteristics of soils by any functional elastic models.

2.2.2 *Hyperelastic and Hypoelastic Models*

Nonlinear and dilatant behavior of the soils can be modeled by higher order elastic theories, which are broadly classified as hyperelastic and hypoelastic theories. The hyperelasticty model of Saleeb and Chen (1980) and the hypoelasticity models of Truesdell (1955) and of Romano (1974) are a few successful examples of such applications in soil mechanics.

In a hyperelastic model, stress components are derived by differentiating a strain energy function with respect to strain components. For an isotropic and homogeneous material, the strain energy function can be expressed as a power series of strain invariants. Different orders of hyperelastic theories can be obtained by truncating the unnecessary higher order terms. Since the stresses in a hyperelastic model are simply expressed as functions of strains, material characteristics such as path dependence and stress history dependence cannot be modeled.

Unlike a hyperelastic theory, a hypoelastic model expresses the constitutive relationship in an incremental form. The stress increment can usually be expressed as a power series in terms of stresses and strain increments. By retaining partial terms, different orders of hypoelastic models can be obtained. Davis and Mullenger (1978) used different inelastic formulations on loading and unloading paths to describe the path-dependent characteristics of the soil. However, Mroz(1980) concluded that the influence of the stress history usually cannot be modeled properly in a hypoelastic model, since the stress history of a material is not traced.

2.2.3 *Endochronic Models*

The endochronic theory is an approach first developed by Valanis (1971) for modeling inelastic material behavior of metal without using a yield surface. A variable called "intrinsic time" is considered to be a function of selected cumulative strains and the elapsed time. The model is formulated by correlating the inelastic strain increment to the variation of the "intrinsic time" as well as the stress and strain fields. This approach has the advantage of modeling rate-dependent materials easily. Bazant and Krizek (1976), Bazant (1977), Valanis and Read (1982), and several other researchers have applied this theory to model the behavior of the soils.

In order to describe the inelastic response, the intrinsic time variables in this theory are related to the stress and strain variables through several empirical and semi-empirical equations. Therefore large number of model parameters are required to ensure the generality of the model. These model parameters usually do not have any physical significance. Therefore, the applicability of this model to general behavior of soil under general loading situations is seriously limited.

2.2.4 *Elasto-Plastic Models*

Material deformation characteristics may be idealized as behaving like a continuum with separable increments of recoverable, i.e. elastic, and irrecoverable, i.e. plastic deformations. The classical theory of plasticity(Hill 1950, Prager 1959) developed for metals allows for such differentiation between recoverable and irrecoverable deformations. However, because classical plasticity theory essentially was developed for metals (Hill 1950), modifications and extensions must be incorporated to model the behavior of soils, such that: i) yielding of soils is anisotropic and dependent upon the confining stress, ii) there exists a coupling between the volumetric changes and the changes in shear stress, and iii) during shear, dense soils expand whereas loose soils contract.

Based on work-hardening concepts, the early work of Drucker et al. (1957) marked the

beginning of the development of elasto-plastic models for the mechanical behavior of geo-materials. This first successful synthesis of soil mechanics and the mathematical theory of plasticity led to a steady progression of subsequent refinements in the application of plasticity theories towards a unified explanation of shear distortion and densification phenomena in soils. Many of the fundamental concepts of modeling were introduced by the "Cam-Clay" models, also known as critical state models, due to the work of Roscoe et al. (1958, 1968). These models are still considered significant milestones in the research on constitutive models, and the underlying concepts have been retained in many of the more recent models.

Since real geologic materials are much more complex than Cam-Clay in their inelastic response, a wide variety of macroscopic plasticity models have attempted to address the various complex aspects of the material behavior by incorporating more and more involved mathematical techniques. The capabilities of the models have been greatly enhanced by including concepts such as bounding and loading surfaces by Dafalias and Popov (1975), subloading surfaces by Hashiguchi (1986), kinematic and mixed hardening rules, transitional yielding by Banerjee and Pan (1986), nonassociated flow rules by Mroz and Pietruszczak (1983, 1983) and by Lade (1977), and multiple yield mechanisms proposed by Koiter (1953).

Based on the concept of multiple yield mechanisms (Koiter 1953), Prevost and Hoeg (1975) employed two separate yield mechanisms to describe the behavior of soils. This concept, referred to as "double hardening", was later been adopted by various researchers (Lade 1977, Vermeer 1978, Ohmaki 1979). In a recent constitutive model, Banerjee et al. (1992) have used this concept to model the behavior of soil in a triaxial space. This work utilized the traditional approach (Casagrande 1936, Roscoe et al. 1958, Roscoe & Burland 1968) of representing the volume change data in semi-logarithmic form.

2.3 Present Model

The semi-logarithmic representation of volume change behavior, originally proposed by Casagrande (1936), with minor modification, was adopted into the Cam-Clay models (Roscoe et al. 1958, Roscoe & Burland 1968) as the underlying law of isotropic compression. Subsequently, many constitutive models (Dafalias 1986, Mroz et al. 1981) have also incorporated the same law into the formulation of a general constitutive laws valid for all possible stress paths and stress ranges. However, this law was primarily conceived for one-dimensional settlement computations within a limited stress range thereby seriously limiting the applicability of the resulting constitutive laws.

An alternative formulation (Banerjee & Sribalaskandarajah 1994) to describe the volume behavior of soils under isotropic compression, has been developed in this study and is presented in the following chapter. Based on this formulation and the axisymmetric double-hardening model (Banerjee et al. 1992), a more general double-hardening model was developed and is presented in the subsequent chapters of this dissertation. As will be discussed later, the advantages of the proposed model are: the model is general; the number of parameters is small; parameters are physically meaningful; and the essential and prominent features of the soil behavior are reproducible.

Chapter 3

MODELING THE BEHAVIOR OF SOILS UNDER GENERAL LOADING CONDITIONS

A general constitutive model for soils and its implementation in simple numerical analysis schemes are presented in this chapter. As outlined in the previous chapter, a non-associative elasto-plastic "double hardening" concept is used in the development of this model. The model employs two distinct yield functions, namely shear and volumetric, each describing the volumetric and shear mechanisms respectively, leading to a simple but general model. The number of model parameters is small relative to most general constitutive models, and these parameters are physically meaningful. The model is tested against the published experimental data for different loading and drainage conditions. As these experimental data are essentially limited to triaxial loading conditions, the verification of the model is essentially confined to triaxial monotonic and cyclic loading conditions. Unlike existing simple models, this model reproduces the essential features of the behavior of sands, such as mean-stress and loading-path-dependent dilatancy, hardening and softening, liquefaction, and initial and induced anisotropy, as well as the essential features of clays. In other words, the model presented in this chapter reproduces the drained and undrained behavior of geomaterials under monotonic and cyclic loading conditions.

As mentioned in the previous chapter, the proposed model's volumetric behavior of soils during isotropic compression and extension is based on an alternative formulation. This alternative formulation, presented in the next section, describes the volumetric behavior of soils by an exponential law and a power law, characterizing isotropic compression and extension, respectively. As will be discussed later, this formulation addresses the deficiencies

observed in the traditional formulations. The proposed constitutive model and the simplified form of this model suitable for easy implementation in numerical analysis schemes are presented in the subsequent sections.

3.1 Modeling Volume-Change Behavior of Soils under Isotropic Compression

The volume change behavior of soils under isotropic and one-dimensional compression has been visualized in terms of semi-logarithmic plots of void ratio, e, and mean effective principal stress, p. The original proposal (Casagrande 1936) of $e - \log p$ relationships and the modified form of $e - \ln p$ relationships have been used tacitly in classical soil mechanics. However, over the years numerous researchers (Nishida 1956, Butterfield 1979, Janbu & Senneset 1979, Christie & Tonks 1985, Den Haan 1992) have attempted to overcome the deficiencies to develop a more reliable form of stress-strain relationship.

It has been noted by Butterfield (1979) that the traditional approach fails to ensure a physical limit for volumetric strain at large values of stress. In other words, it does not recognize the existence of a minimum void ratio at which the soil becomes incompressible. Typical void ratio against mean stress curves are concave upwards at higher values of stress indicating this incompressibility. The existence of an incompressible minimum void ratio is ensured in this study, and the proposed formulation and verification of the formulation are presented in the ensuing subsections.

3.1.1 Formulation

Preliminaries

Let us consider a soil sample with state parameters: void ratio, e, and mean or isotropic effective (compressive) stress, p. The volumetric strain increment, $d\epsilon_v$, can be written in terms of specific volume, $v(= 1 + e)$, as

$$d\epsilon_v = -\frac{dv}{v} = -\frac{de}{1+e} \tag{3.1}$$

The strain definition given above is the so called true strain or large strain definition, which is commonly used to incorporate the geometric nonlinearity of the material. However, for most practical purposes, it is sufficient to include material nonlinearity alone. Using small strain assumptions the above equation can be rewritten as

$$d\epsilon_v = -\frac{dv}{v_r} = -\frac{de}{1+e_r} \tag{3.2}$$

where e_r is the reference void ratio and v_r is the reference specific volume at which the volumetric strain is zero. The volumetric strain increment can be related to the mean effective stress increment by

$$d\epsilon_v = -\frac{dv}{v_r} = -\frac{dp}{B} \tag{3.3}$$

where B is the incremental bulk modulus of the soil.

Loading

It is well known that the bulk modulus of soils is not constant for the entire possible range of loading. In fact, for monotonic loading situations, the bulk modulus of geomaterials may lie within the range $B_0 \leq B < \infty$; when $v = v_0$ (i.e., $B = B_0$), the material is compressible, and when v approaches v_{min} (i.e, $B = \infty$), the material becomes incompressible. These bounds are satisfied by

$$B = B_0 \frac{v_0 - v_{min}}{v - v_{min}} \tag{3.4}$$

Then, from equations (3.3) and (3.4) the incremental stress-strain relationship can be written as

$$-\frac{dv}{v_r} = \frac{v - v_{min}}{B_0 (v_0 - v_{min})} dp \tag{3.5}$$

Integration of both sides of equation (3.5) between appropriate limits yields

$$v = v_{min} + (v_0 - v_{min}) e^{-\alpha\left(\frac{p-p_0}{p_a}\right)} \tag{3.6}$$

where p_0 is the initial mean effective stress (i.e., when v=v_0) and the parameter α is given by

$$\alpha = \frac{p_a v_r}{B_0 (v_0 - v_{min})} \tag{3.7}$$

In the foregoing expression the mean stress is normalized by a reference stress p_a which can be taken for convenience as the atmospheric pressure. The constant α can be viewed as a measure of the compressibility of the soil. From equation (3.2), the volumetric strain, ϵ_v, the initial value of volumetric strain, ϵ_{v0} at the beginning of loading or reloading, and the maximum volumetric strain, ϵ_{vm} that the soil sample can possibly undergo can be written, respectively, as

$$\begin{aligned} \epsilon_v &= \frac{v_r - v}{v_r} \\ \epsilon_{v0} &= \frac{v_r - v_0}{v_r} \\ \epsilon_{vm} &= \frac{v_r - v_{min}}{v_r} \end{aligned} \tag{3.8}$$

Using the above definitions, equation (3.6) can be rearranged to yield a nonlinear volumetric strain - mean stress relation of the form

$$\epsilon_v = \epsilon_{v0} + (\epsilon_{vm} - \epsilon_{v0}) \left\{ 1 - e^{-\alpha\left(\frac{p-p_0}{p_a}\right)} \right\} \tag{3.9}$$

It turns out that the above relationship is similar in form to the one proposed for the Cap models (DiMaggio & Sandler 1971). Strictly speaking, such a relationship applies only for monotonic isotropic compression of soils. Equation (3.6) also can be recast to obtain a volumetric strain-mean stress relation in the simple incremental form as

$$\frac{d\epsilon_v}{\epsilon_{vm} - \epsilon_v} = \alpha \frac{dp}{p_a} \tag{3.10}$$

It should be noted that α remains essentially constant for different soil samples with varying initial state v_0, while the changing difference between v_0 and v_{min} (equation (3.6)) accounts for the decrease in the rate of volumetric compression. In contrast, for the traditional relationship, three constants, compression and recompression indices, λ and κ, and preconsolidation pressure, p_c, are required for each sample of the same soil.

Unloading

It is customary to assume that the process of unloading is elastic. In fact, various researchers (Butterfield 1979, Den Haan 1992, Janbu & Senneset 1979) have shown that the stress-strain law during unloading can be described by a power law of the form:

$$\epsilon_v = \epsilon_{v0} \left(\frac{p}{p_0}\right)^{\zeta} \tag{3.11}$$

where ζ is a material constant. Differentiating both sides of equation (3.11) one obtains

$$d\epsilon_v = \frac{dp}{B_0 \left(\frac{p}{p_0}\right)^{1-\zeta}} \tag{3.12}$$

where $B_0 = p_0/(\zeta\epsilon_{v0})$ = initial value of bulk modulus. During the process of unloading, the bulk modulus ranges from B_0 (when $p = p_0$) to 0 (when $p = 0$) and varies as

$$B = B_0 \left(\frac{p}{p_0}\right)^{1-\zeta} \tag{3.13}$$

Equations (3.11) and (3.12) can also be combined to yield a volumetric strain - mean stress relation in the simple incremental form as

$$\frac{d\epsilon_v}{\epsilon_v} = \zeta \frac{dp}{p} \tag{3.14}$$

As was noted earlier, the foregoing incremental relation is of the type proposed by Butterfield and others.

3.1.2 Verification

The validity of the proposed relationships between volumetric strain and isotropic mean effective stress for situations of loading and unloading are examined in this section. Three sets of experimental data obtained from isotropic consolidation tests are used for the comparison: Negussey (1992) using Ottawa sand, Saada and Bianchini (1987) using Hostun sand, and Henkel (1959) using London clay .

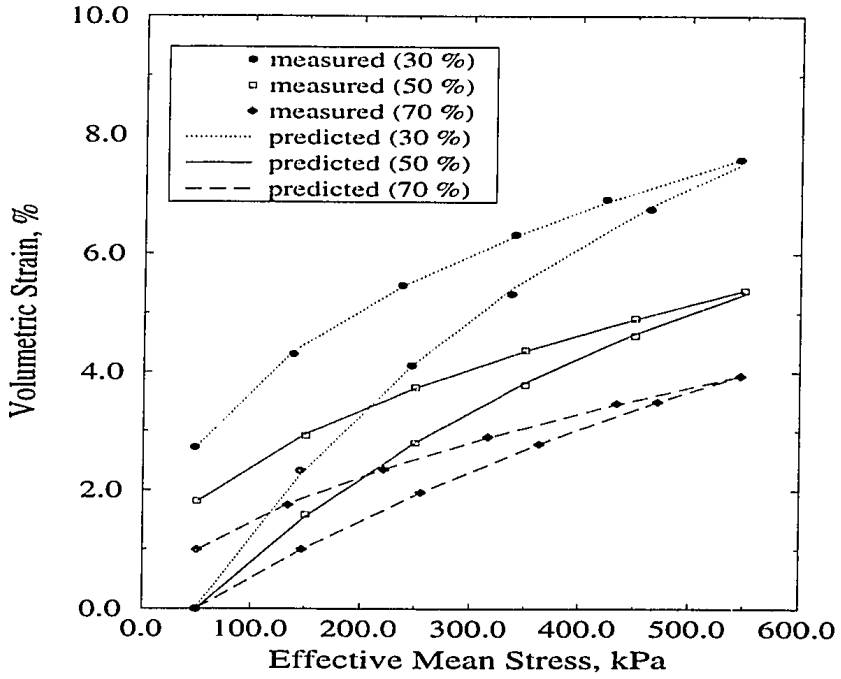


Figure 3.1: Comparison of Computed and Experimental Results (after Negussey 1992) for Ottawa Sand

Figures (3.1)-(3.3) show the comparison of computed and measured volume deformations. Figure (3.1) shows plots of volumetric strain against mean effective stress for samples of Ottawa sand with different initial void ratios (D_r = 30%, 50%, and 70%). The samples were subjected to isotropic compression in a standard triaxial setup (Negussey 1992). Figures (3.2) and (3.3) show plots of void ratio against mean effective stress for several cycles of isotropic compression and extension loading of Hostun sand and London clay samples respectively. While the data selected for Hostun sand were obtained in a cubical triaxial test apparatus, the data for London clay sample were obtained from a conventional triaxial apparatus.

The overall agreement between the predicted and the measured data is quite satisfactory. These comparisons tend to confirm that it is possible to describe the loading and reloading response as an exponentially decreasing function of mean effective stress and the unloading behavior by a power law, without violating the physics of the processes.

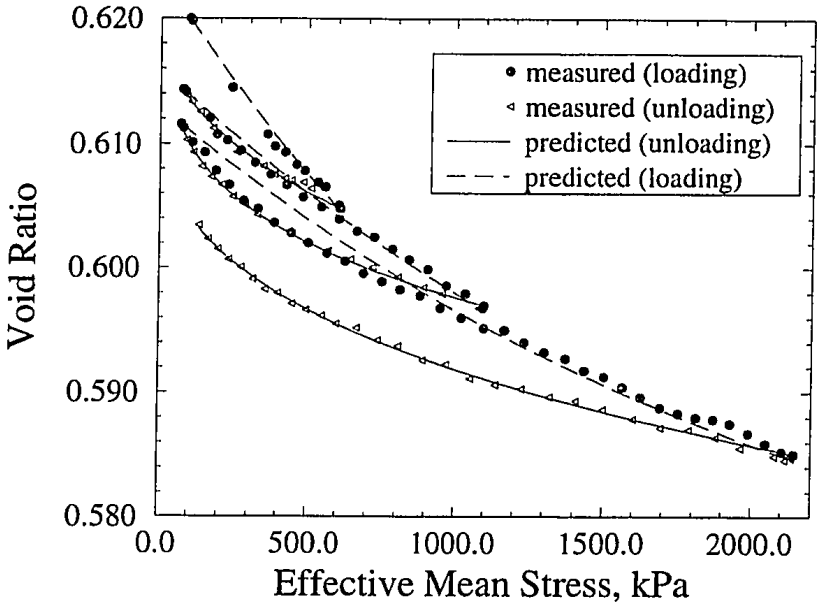


Figure 3.2: Comparison of Computed and Experimental Results (after Saada et al. 1987) for Hostun Sand

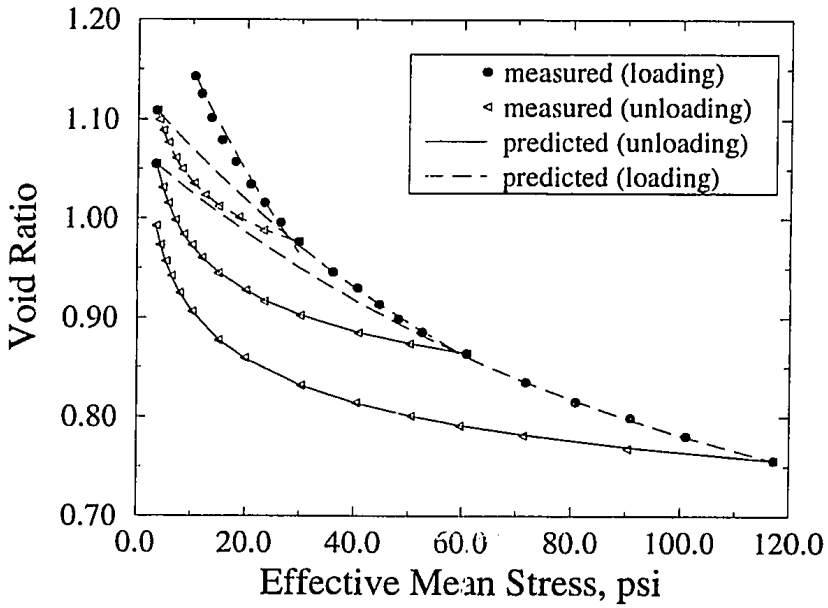


Figure 3.3: Comparison of Computed and Experimental Results (after Henkel 1959) for London Clay

3.2 A Double-Hardening Model for Geomaterials

3.2.1 Preliminaries

The effective stress tensor is denoted by σ_{ij} and the strain tensor by ϵ_{ij}. The invariants of the effective stress tensor are denoted by

$$\begin{aligned} J_1 &= \sigma_{ii} \\ J_2 &= \frac{1}{2}(J_1^2 - \sigma_{ik}\sigma_{kj}) \\ J_3 &= \frac{1}{6}(2\sigma_{ij}\sigma_{jk}\sigma_{ki} - 3J_1\sigma_{ij}\sigma_{ji} + J_1^3) \end{aligned} \tag{3.15}$$

The invariants of the strain tensor, defined in a similar manner, are denoted by I_1, I_2, and I_3. The deviatoric effective stress tensor is denoted as $s_{ij}(= \sigma_{ij} - \frac{J_1}{3}\delta_{ij})$, the deviatoric strain tensor is denoted as e_{ij} $(= \epsilon_{ij} - \frac{I_1}{3}\delta_{ij})$, and the invariants of the deviatoric stress tensor are denoted as

$$\begin{aligned} J_1' &= s_{ii} = 0 \\ J_2' &= \frac{1}{2}s_{ij}s_{ji} \\ J_3' &= \frac{1}{3}s_{ij}s_{jk}s_{ki} \end{aligned} \tag{3.16}$$

The yield and potential functions are functions of stress and are conveniently viewed in principal stress space (Figure (3.4)). As shown in Figure (3.4a), the space diagonal, also known as the hydrostatic axis, is oriented at equal angles to the three principal stress axes. Any plane that is perpendicular to the space diagonal is called a deviatoric plane (Figure (3.4b)) and the deviatoric plane that passes through the origin is known as the π-plane.

For an arbitrary state of stress given by the principal stress values, σ_1, σ_2, and σ_3, the projection of the stress vector on the hydrostatic axis is called the mean effective stress or octahedral normal stress, σ_m $(= \frac{J_1}{3})$. The projection of the stress vector on the deviatoric plane is given by the vector $\langle s_1, s_2, s_3 \rangle$; where, s_1, s_2, and s_3 are the components of the deviatoric principal stress tensor. The magnitude of this projection is called the octahedral shear

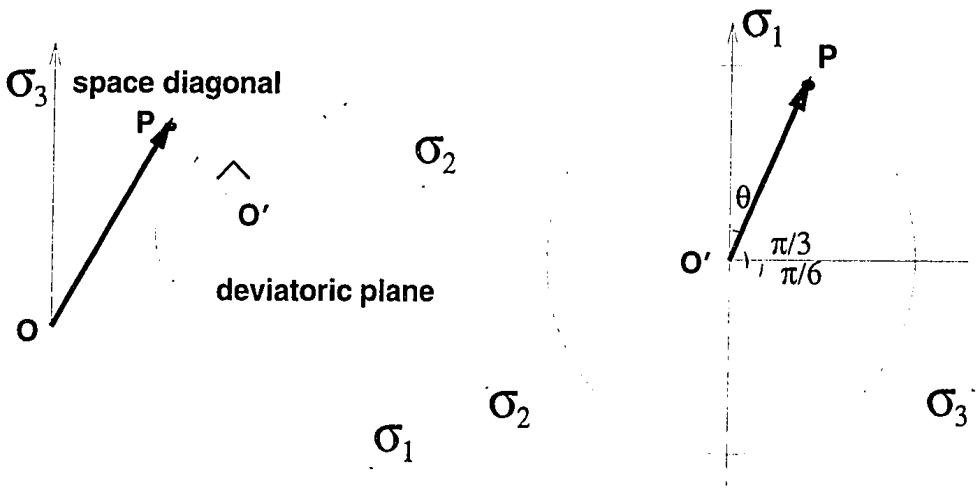


Figure 3.4: Stress Space and its Geometric Interpretation

stress and is given by $\sqrt{2J_2'}$. In Figure (3.4), the angle θ between the major principal axis and the stress vector on the deviatoric plane is known as the Lode angle and is related to the invariants by

$$\cos(3\theta) = \frac{3\sqrt{3}J_3'}{2(J_2')^{3/2}} \tag{3.17}$$

3.2.2 Formulation

In the incremental theory of elasto-plasticity, it is assumed tacitly that the total strain increment, $d\epsilon_{ij}$, due to any stress increment, $d\sigma_{ij}$, can be decomposed into a reversible (elastic) component, $d\epsilon_{ij}{}^e$, and an irreversible (plastic) component, $d\epsilon_{ij}{}^p$, i.e.

$$d\epsilon_{ij} = d\epsilon_{ij}{}^e + d\epsilon_{ij}{}^p \tag{3.18}$$

These elastic and plastic incremental strain components are discussed individually next in detail.

Elastic Strains

In general, soils exhibit predominantly plastic strains and thus it is sufficient to employ a simple linear isotropic elastic constitutive law, such as isotropic Hooke's law, to describe the elastic behavior of the material without losing much accuracy. Employing the isotropic Hooke's law implies that the elastic strains are not coupled, i.e., the changes in elastic distortional stress do not affect the elastic volumetric strain and the change in elastic volumetric stress does not affect the elastic distortional strains. Thus the elastic shear strains as well as the elastic volumetric strain could be expressed as

$$de_{ij}{}^{e} = \frac{ds_{ij}}{2G} \tag{3.19}$$

$$d\epsilon_{ii}{}^{e} = \frac{d\sigma_{ii}}{3K} \tag{3.20}$$

where G is the elastic shear modulus and K is the elastic bulk modulus different from the total bulk modulus, B, defined in the previous subsection.According to the findings of Hardin and Drenevich (1972), the elastic shear modulus in soils varies as a power of mean effective principal stress, $\sigma_m = \frac{\sigma_{ii}}{3}$. Although the shear modulus is traditionally defined by the secant shear modulus of the soil, the elastic shear modulus could be approximated by the initial tangent modulus, which is also the initial secant modulus. Therefore the elastic shear modulus should conform to the findings of Hardin and Drenevich:

$$G = G_o(\frac{\sigma_m}{\sigma_a})^{1-\beta} \tag{3.21}$$

in which σ_a is the atmospheric pressure; β is a material constant; and G_o is the reference value of bulk modulus when $\sigma_m = \sigma_a$.

For a linear elastic material, the shear modulus and the bulk modulus are related by Poisson's ratio. Therefore, it is reasonable to expect the value of the bulk modulus to have a similar variation with the mean effective principal stress as that of the shear modulus. Recently, in an attempt to describe the volume-change behavior of the isotropic swelling of soils, Banerjee and Sribalaskandarajah (1994) have shown that the total bulk modulus, B,

can be assumed to vary in a similar manner as in equation (3.13). This volume change behavior is usually assumed to be purely elastic, therefore the elastic bulk modulus could be expressed as a function of mean effective principal stress, σ_m, as

$$K = K_o(\frac{\sigma_m}{\sigma_a})^{1-\zeta} \tag{3.22}$$

where K_o is the reference value of bulk modulus when $\sigma_m = \sigma_a$, and ζ is a material constant.

Plastic Strains

The concept of "the theory of plastic potential", which assumes the existence of a single yield mechanism, was subsequently generalized by Koiter (1953) to include multiple yield mechanisms. If there are n such mechanisms and the associated yield functions as well as the potential functions are denoted by $F_k(s_{ij}) = 0, k = 1, n$ and $G_k(s_{ij}) = 0, k = 1, n$ respectively, then the plastic strain increments can be written as

$$d\epsilon_{ij}{}^p = \sum_{k=1}^{n} C_k H_k \frac{\partial G_k}{\partial \sigma_{ij}} \frac{\partial F_k}{\partial \sigma_{lm}} d\sigma_{lm} \tag{3.23}$$

where H_k is a nonnegative scalar function of stress, plastic strain, and strain history and C_k is a constant taking a value of "one" during the loading while taking a value of "zero" during the unloading.; i.e.

$$\left.\begin{array}{llll} C_k = 0 & F_k < 0 & \text{or} & \frac{\partial F_k}{\partial \sigma_{ij}} d\sigma_{ij} < 0 \\ C_k = 1 & F_k = 0 & \text{and} & \frac{\partial F_k}{\partial \sigma_{ij}} d\sigma_{ij} \geq 0 \end{array}\right\} \tag{3.24}$$

In soil plasticity studies, Prevost and Hoeg (1975) employed this concept with two separate yield mechanisms to describe the volumetric and shear behaviors of the soil. For such a model, which is also known as a "double hardening model", equation (3.23) can be written as

$$d\epsilon_{ij}{}^p = C_v H_v \frac{\partial G_v}{\partial \sigma_{ij}} dF_v + C_d H_d \frac{\partial G_d}{\partial \sigma_{ij}} dF_d \tag{3.25}$$

where the subscript v is used to denote the volumetric mechanism, and the subscript d is used to denote the shear mechanism. Figure (3.5) shows the yield functions and potential

functions for both mechanisms used in this study. The mathematical forms of these functions are outlined below.

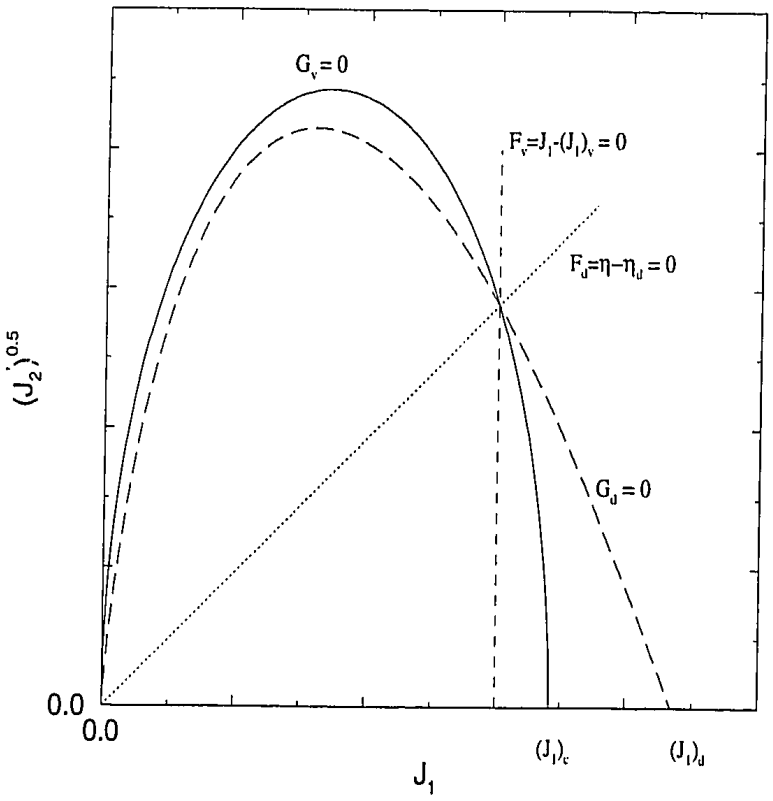


Figure 3.5: Yield Functions and Potential Functions

Shear Yielding

The yield function for the shear mechanism is assumed to be given by

$$F_d = s_\eta(\eta - \eta_d) \tag{3.26}$$

where η_d is a constant describing the nature of the hardening, s_η is a parameter denoting the sign of $d\eta$, and η is the stress ratio defined in terms of the stress invariants as

$$\eta = s_\pi \frac{3\sqrt{3J_2'}}{J_1} \tag{3.27}$$

where s_π is the sign of the third stress invariant J_3' and the value of $d\eta$ is

$$d\eta = \eta \left[\frac{dJ_2'}{2J_2'} - \frac{dJ_1}{J_1} \right] \tag{3.28}$$

In view of equation (3.26) and by the definition of s_η, it can be seen that the value of $\frac{\partial F_d}{\partial \sigma_{ij}} d\sigma_{ij}$ is always non-negative and therefore "unloading" is undefined in this mechanism. Therefore, the choice of yield function ensures that this mechanism always induces plastic strains whenever there is a change in η, i.e.

$$C_d = 1.0 \qquad \text{for any } d\eta \tag{3.29}$$

The flow potential function of the shear mechanism, G_d, is assumed to be of the following form

$$G_d = s_\eta \, \eta + M_c(\theta) \, \ln\left(\frac{J_1}{(J_1)_d} \right) = 0 \tag{3.30}$$

where $(J_1)_d$ is a constant and M_c is the absolute value of the ultimate stress ratio, i.e.

$$M_c = |\eta|_{ultimate} \tag{3.31}$$

For the special case of triaxial conditions, the potential function given by equation (3.30) becomes identical to the yield function used in the original Cam-Clay model (Roscoe et al. 1958). Also, the shape of the potential function depends on the direction of loading and the state of stress. This is illustrated in Figure (3.6a) for a cyclic triaxial compression test. The stress path is shown as the solid straight line and the evolution of the potential function is depicted for states of stress marked on the stress path. The potential function is concave downward when $d\eta > 0$ and concave upward when $d\eta < 0$.

The parameter M_c is, in general, a function of Lode angle, θ, and thus describes the anisotropic behavior of the soil. In order to study the variation of the parameter M_c with respect to Lode Angle, the failure criterion proposed by Matsuoka and Nakai (1974) is closely examined, because this failure function was derived by considering the mechanical aspects of the shearing mechanism. It is expressed as

$$\frac{J_1 J_2}{J_3} = c \tag{3.32}$$

where c is a material constant. In view of equation (3.31) and by the definition of η, the value of M_c can be expressed in terms of the stress ratio at failure, η_f as

$$M_c = s_\eta \, \eta_f \tag{3.33}$$

Therefore, in view of the definitions of stress ratio and Lode Angle, equation (3.32) could be expressed in terms of M_c as

$$c = \frac{27(9 - M_c^2)}{2s_\pi \, s_\eta \, \cos(3\theta) M_c^3 - 9M_c^2 + 27} \tag{3.34}$$

The above equation can be used to estimate the value of c from the value of the parameter M_c. By rearranging we arrive at a cubic equation for M_c in terms of θ as

$$2s_\pi \, s_\eta \, c \, \cos(3\theta) M_c^3 - 9(c-3)M_c^2 + 27(c-9) = 0 \tag{3.35}$$

By taking derivatives with respect to θ, equation (3.35) can be rearranged as

$$\frac{\partial M_c}{\partial \theta} = \frac{s_\pi \, s_\eta \, c M_c^2 \sin(3\theta)}{[s_\pi \, s_\eta \, c M_c \cos(3\theta) - 3(c-3)]} \tag{3.36}$$

Further, from the definitions of Lode angle, θ, the partial derivative of θ with respect to σ_{ij} can be expressed as

$$\frac{\partial \theta}{\partial \sigma_{ij}} = \frac{\cot(3\theta)}{3} \left[\frac{3}{2J_2'} s_{ij} - \frac{1}{J_3'} (s_{ik} s_{kj} - \frac{2}{3} J_2' \delta_{ij}) \right] \tag{3.37}$$

From equation (3.36) and equation (3.37), we get

$$\frac{\partial M_c}{\partial \sigma_{ij}} = \frac{s_\pi \, s_\eta \, c M_c^2 \cos(3\theta)}{3 \, [s_\pi \, s_\eta \, c M_c \cos(3\theta) - 3(c-3)]} \left[\frac{3}{2J_2'} s_{ij} - \frac{1}{J_3'} \left(s_{ik} s_{kj} - \frac{2}{3} J_2' \delta_{ij} \right) \right] \tag{3.38}$$

With simple algebra, the first invariant of the above tensor, $\frac{\partial M_c}{\partial \sigma_{ij}}$, can be shown to vanish, i.e.

$$\frac{\partial M_c}{\partial \sigma_{ii}} = 0 \tag{3.39}$$

The derivative of the potential function, G_d with respect to σ_{ij}, i.e., $\frac{\partial G_d}{\partial \sigma_{ij}}$, is a second order tensor and can be expressed as a sum of the deviatoric tensor component and an isotropic tensor component, as

$$\frac{\partial G_d}{\partial \sigma_{ij}} = D_{ij} + d \, \delta_{ij} \tag{3.40}$$

where D_{ij} and d are the deviatoric tensor and the first invariant of the tensor $\frac{\partial G_d}{\partial \sigma_{ij}}$, respectively and their values are given by

$$D_{ij} = \left[\frac{s_\eta\,\eta}{2J_2'}s_{ij} + \ln\left(\frac{J_1}{(J_1)_d}\right)\frac{\partial M_c}{\partial \sigma_{ij}}\right] \tag{3.41}$$

$$d = \frac{M_c - s_\eta\,\eta}{J_1} \tag{3.42}$$

Volumetric Yielding

The volumetric yield function, F_v is assumed to be given by

$$F_v = J_1 - (J_1)_v = 0 \tag{3.43}$$

where $(J_1)_v$ is a constant determining the nature of the hardening.

In view of equation (3.26) and the volumetric yield function given by equation (3.43), the volumetric loading is assumed to occur when dJ_1 is positive, while unloading is assumed to occur when dJ_1 is negative. Therefore, the value C_v as in equation (3.25) is given by

$$\left.\begin{array}{ll} C_v = 1.0 & dJ_1 > 0 \\ C_v = 0.0 & dJ_1 < 0 \end{array}\right\} \tag{3.44}$$

The plastic strains are assumed to occur only due to virgin loading and reloading and the volumetric unloading induces purely elastic strains, however, the yield surface expressed by equation (3.43) is assumed to remain valid even during unloading. In other words, the yield surface is assumed to collapse or degrade during unloading.

The potential function of this mechanism, G_v, is assumed to be of the form

$$G_v = J_1 + \frac{27J_2'}{J_1 M_p^2} - (J_1)_c = 0 \tag{3.45}$$

where $(J_1)_c$ is a constant and M_p is the ultimate value of stress ratio, η. For the special case of triaxial conditions the above potential function is identical to the one used in the modified cam-clay model (Roscoe & Burland 1968). The family of potential functions during a cyclic triaxial compression test is illustrated in Figure (3.6b).

The parameter M_p is assumed to change with the plastic distortional strain history in an incremental form such that the ultimate value of M_p is in fact equal to the value of M_c. This use of a different parameter for the ultimate value of stress ratio evolving with the loading history allows us to model the strain-softening behavior of the soil. The incremental variation is assumed to be given by

$$dM_p = \eta(M_c - \eta)(M_p^2 - M_c^2)dW^p \tag{3.46}$$

where dW^p is the change in the plastic distortional work given by

$$dW^p = e_{ij}{}^p \, d\sigma_{ij} \tag{3.47}$$

Since both the parameters M_c and M_p denote the ultimate stress ratio, it is reasonable to assume that the variation of M_p with respect to σ_{ij} is of the same form as that of the parameter M_c. Therefore, in view of equation (3.48), we get

$$\frac{\partial M_p}{\partial \sigma_{ij}} = \frac{s_\pi \, s_\eta \, c M_p^2 \cos(3\theta)}{3\left[s_\pi \, s_\eta \, c M_p \cos(3\theta) - 3(c-3)\right]} \left[\frac{3}{2J_2'} s_{ij} - \frac{1}{J_3'}\left(s_{ik}s_{kj} - \frac{2}{3}J_2'\delta_{ij}\right)\right] \tag{3.48}$$

With simple algebraic manipulations, it can be shown that the first invariant of the above tensor, $\frac{\partial M_p}{\partial \sigma_{ij}}$, vanishes, i.e.

$$\frac{\partial M_p}{\partial \sigma_{ij}} = 0 \tag{3.49}$$

The derivatives of the volumetric potential function, G_v with respect to σ_{ij}, i.e., $\frac{\partial G_v}{\partial \sigma_{ij}}$, is a second order tensor and can be expressed as the sum of isotropic and deviatoric components as

$$\frac{\partial G_v}{\partial \sigma_{ij}} = V_{ij} + v \, \delta_{ij} \tag{3.50}$$

where V_{ij} and v are the deviatoric tensor and the first invariant of the tensor $\frac{\partial G_d}{\partial \sigma_{ij}}$ respectively and are given by

$$V_{ij} = \frac{1}{M_p^2}\left[\frac{27}{J_1} s_{ij} - \frac{2\eta^2 \, J_1}{M_p} \frac{\partial M_p}{\partial \sigma_{ij}}\right] \tag{3.51}$$

$$v = \frac{M_p^2 - \eta^2}{M_p^2} \tag{3.52}$$

Hardening parameters

The hardening moduli for volumetric yielding, H_v, and for shear yielding, H_d, were chosen to be functions of the stress and strain state parameters as well as the model parameters as

$$H_v = \frac{M_p}{9(M_p + |\eta|)} \left[\frac{\alpha(I_{1m} - I_1)}{\sigma_a} - \frac{1}{K} \right] \tag{3.53}$$

$$H_d = \exp\left(\frac{-I_1}{I_{1m} - I_1}\right) \left[\frac{|\eta - \eta_r| J_1^2}{27G} \frac{M_p}{(M_p - \eta)} \right] \left(\frac{J_1}{\sigma_a + \sigma_p} \right)^{\beta} \tag{3.54}$$

where α is a material parameter that describes the total volume-change behavior due to the change in J_1 (Banerjee & Sribalaskandarajah 1994); η_r is the value of η at the previous shear reversal; I_{1m} is the maximum value of the first strain invariant, I_1; and σ_p is the excess pore pressure. The value of I_{1m} can be expressed in terms of the initial void ratio, e_0 and the minimum void ratio, e_{min} as

$$I_{1m} = \frac{e_0 - e_{min}}{1 + e_0} \tag{3.55}$$

3.3 Implementation of the Model in Numerical Schemes

In this section computational procedures are outlined for the most common situations encountered in geomechanics practice to facilitate the implementation of the above constitutive model into numerical schemes. In general, either stress or strain increments are known and drainage conditions are assumed to be extreme, i.e., drained or undrained.

3.3.1 Computation of Incremental Stress from Incremental Strain

The incremental strains are primary variables, therefore the stress increments are calculated from the current state parameters as well as the incremental strains. The choice of incremental strains as primary variables does in fact implicitly specify the drainage conditions, as incremental volumetric strain is also being assumed. However, during undrained conditions the first invariant of the incremental strain tensor must remain zero. In other words, when incremental strains are specified the computational details remain the same for both drained and undrained conditions as outlined below.

In view of the incremental elasto-plastic theory (equation (3.18)) and the model outlined in the previous section using equations (3.25), (3.26), (3.30), (3.43), and (3.45), the incremental elastic strain components can be expressed as

$$d\epsilon_{ij}{}^{e} = d\epsilon_{ij} - C_v H_v \left(V_{ij} + v\delta_{ij}\right)\, dJ_1 - C_d H_d\, s_\eta \left(D_{ij} + d\delta_{ij}\right)\, d\eta \tag{3.56}$$

Taking the first invariant of the incremental elastic strain tensor and considering the generalized Hooke's law (equation (3.20)), one obtains from equation (3.56)

$$dJ_1 = 3K\left(d\epsilon_{ii} - 3C_v H_v v\, dJ_1 - 3C_d H_d\, s_\eta d\, d\eta\right) \tag{3.57}$$

Now, considering the definition of the second deviatoric invariant, J_2', of the stress tensor given by equation (3.16), one gets

$$dJ_2' = s_{ij} ds_{ij} \tag{3.58}$$

Substituting for ds_{ij} from equation (3.19), we arrive at

$$dJ_2' = 2G\, s_{ij}\, de_{ij}{}^{e} \tag{3.59}$$

In view of equation (3.56), equation (3.59) can be rearranged as

$$dJ_2' = 2G\left[s_{ij}\, de_{ij} - C_v H_v s_{ij} V_{ij}\, dJ_1 - C_d H_d\, s_\eta s_{ij} D_{ij}\, d\eta\right] \tag{3.60}$$

In view of equations (3.39) and (3.49), the following simplifications can be derived.

$$s_\eta s_{ij} D_{ij} = \eta \tag{3.61}$$

$$s_{ij} V_{ij} = \frac{2\eta^2 J_1}{M_p^2} \tag{3.62}$$

Substituting these expressions, equation (3.60) can further be simplified as

$$dJ_2' = 2G\left[s_{ij}\, de_{ij} - \frac{2C_v H_v \eta^2 J_1}{M_p^2}\, dJ_1 - C_d H_d \eta\, d\eta\right] \tag{3.63}$$

In view of equation (3.28), substituting for the term dJ_2' in terms of $d\eta$ and dJ_1, the above equation can be rewritten as

$$\left[C_d H_d G\,\eta + \frac{J_2'}{\eta}\right]\, d\eta + \left[\frac{J_2'}{J_1} + \frac{2C_v H_v G \eta^2 J_1}{M_p^2}\right]\, dJ_1 = G s_{ij} de_{ij} \tag{3.64}$$

Rewriting equation (3.57), we get

$$[9C_d H_d\, s_\eta K d]\; d\eta + [1 + 9C_v H_v K\, v]\; dJ_1 = 3K\, d\epsilon_{ii} \tag{3.65}$$

Equations (3.64) and (3.65) form a set of simultaneous equations in terms of the two unknowns $d\eta$ and dJ_1. These unknowns can be solved explicitly to give

$$dJ_1 = \frac{v_s\, r_s - s_s\, r_v}{v_s\, s_v - s_s\, v_v} \tag{3.66}$$

$$d\eta = \frac{s_v\, r_v - v_v\, r_s}{v_s\, s_v - s_s\, v_v} \tag{3.67}$$

where

$$\begin{aligned} s_v &= \frac{J_2'}{J_1} + \frac{2C_v H_v G \eta^2 J_1}{M_p^2} \\ s_s &= C_d\, H_d\, s_\eta\, G\eta + \frac{J_2'}{\eta} \\ v_v &= 1 + 9C_v\, H_v\, K v \\ v_s &= 9C_d\, H_d\, s_\eta\, K\, d \\ r_s &= G\, s_{ij}\, de_{ij} \\ r_v &= 3K\, d\epsilon_{ii} \end{aligned} \tag{3.68}$$

The incremental effective stress tensor can be written in terms of dJ_1 and $d\eta$ as

$$\begin{aligned} d\sigma_{ij} = {} & \frac{dJ_1}{3}\left(1 - \frac{2G}{3K}\right)\delta_{ij} + \\ & 2G\left\{d\epsilon_{ij} - C_v H_v\left(V_{ij} + v\delta_{ij}\right)\, dJ_1 + C_d H_d\, s_\eta\, \left(D_{ij} + d\delta_{ij}\right)\, d\eta\right\} \end{aligned} \tag{3.69}$$

3.3.2 Computation of Incremental Strain from Incremental Stress

The choice of components of incremental stress as primary variables does not require any assumptions regarding the drainage conditions. However, the drainage conditions have an impact on how the incremental strains are computed. This is because during undrained conditions excess pore pressure is induced, and the effective stress tensor is therefore different from the total stress tensor. Due to the effective stress principle, the incremental effective

stress tensor, $d\sigma_{ij}$, can be expressed in terms of the incremental total stress tensor, $d\sigma_{ij}{}^{t}$, and the incremental pore pressure, $d\sigma_p$, as

$$d\sigma_{ij} = d\sigma_{ij}{}^{t} - d\sigma_p \delta_{ij} \tag{3.70}$$

Drained

Since excess pore pressure is not generated during fully drained conditions, the incremental effective stress tensor is identical to the incremental total stress tensor. Thus, one can readily calculate the values of dJ_1 and $d\eta$ from the the incremental stress and the incremental elastic and plastic strain tensors can be evaluated explicitly. The incremental strain tensor can be expressed as

$$d\epsilon_{ij} = \frac{ds_{ij}}{2G} + \frac{d\sigma_{ii}}{9K}\,\delta_{ij} + C_v H_v\,(V_{ij} + v\delta_{ij})\;dJ_1 + C_d H_d\,s_\eta\;(D_{ij} + d\delta_{ij})\;d\eta \tag{3.71}$$

Undrained

In practice, components of effective stress are difficult to control, therefore the components of incremental total stresses are considered as the primary variables. Although the components of incremental total stress are known, due to the development of pore pressure, the components of incremental effective stress remain unknown. However, the first invariant of the incremental strain tensor is zero due to the drainage conditions. Therefore, in view of the equations (3.64)-(3.69) one can arrive at

$$d\eta = \frac{-v_v}{v_s}\,dJ_1 \tag{3.72}$$

Substituting for $d\eta$ from equation (3.28) and eliminating dJ_2' in view of equation (3.58), the above equation simplifies to

$$dJ_1 = \frac{\eta\, J_1\, v_s}{2J_2'\,(v_s\eta - v_v J_1)}\;s_{ij}\;ds_{ij} \tag{3.73}$$

where

$$ds_{ij} = d\sigma_{ij}^{t} - \frac{d\sigma_{ii}^{t}}{3}\,\delta_{ij} \tag{3.74}$$

Substituting equation (3.73) into equation (3.72), the value of $d\eta$ can be expressed as

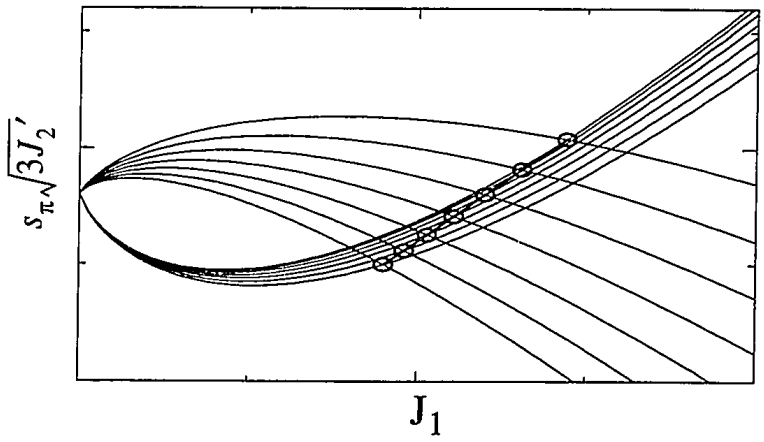
$$d\eta = \frac{-\eta\, J_1\, v_v}{2J_2'\,(v_s\eta - v_v J_1)}\; s_{ij}\; ds_{ij} \tag{3.75}$$

Using equations (3.73)-(3.75), the values of dJ_1 and $d\eta$ can be calculated. Thus, the values of incremental strain tensor and the incremental effective stress tensor are expressed in terms of dJ_1 and $d\eta$ as

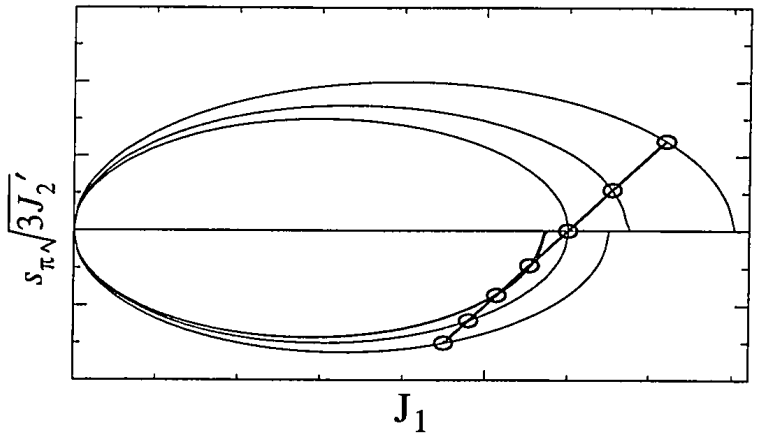
$$\begin{aligned} d\epsilon_{ij} &= \frac{ds_{ij}}{2G} + \frac{dJ_1}{9K}\,\delta_{ij} + \\ &\quad C_v H_v\,(V_{ij} + v\delta_{ij})\; dJ_1 + C_d H_d\, s_\eta\; (D_{ij} + d\delta_{ij})\; d\eta \qquad (3.76) \\ d\sigma_{ij} &= ds_{ij} + \frac{dJ_1}{3}\,\delta_{ij} \qquad (3.77) \end{aligned}$$

3.4 Summary

An alternative formulation to describe the volumetric behavior of soils under isotropic compression has been proposed and verified in the Section (3.1). In addition, a general constitutive law has been proposed in the Section (3.2) with only seven model parameters, denoted as, G_o, K_o, α, β, ζ, M_c, and M_p, and state parameters, e_0 and e_{min}. This model can be categorized as a non-associative plasticity model based on multiple yielding concepts as in Koiter (1953). Two separate yielding mechanisms are used in this model, each characterizing the volumetric and shear yielding behavior of the geomaterial. The complete set of the equations characterizing the model is included in this paper so that they can be incorporated in numerical techniques. For general situations encountered in the most common numerical schemes, simplified model equations have been presented in the Section (3.3). The performance of this model against published experimental data is discussed in the next chapter.



a). Shear Mechanism



b). Volumetric Mechanism

Figure 3.6: Evolution of Potential functions during cyclic triaxial compression test

Chapter 4

PERFORMANCE OF THE MODEL

The performance of the general soil model proposed in the previous chapter is examined in this chapter. The practical usefulness of this model depends on the actual modeling performance, as well as the ease with which the model parameters can be identified for a given soil. The model parameters could be identified either by devising simple experiments reflecting the influence of each individual parameters or by using numerical techniques to match experimental data from the traditional laboratory experiments. In this study, the latter approach is employed and the method used in this study is outlined in the next section.

The model proposed in the previous chapter handles general three dimensional loading conditions. But, most of the reliable experimental studies of the behavior of soils are limited to specific loading conditions such as triaxial, one-dimensional compression, direct shear, and pure shear. Among these experiments the triaxial tests are the most common, therefore, the model performance in this study is essentially limited to the performance of the proposed soil model under monotonic and cyclic triaxial loading conditions.

The response of the proposed model cannot be expressed analytically, as the formulation of the model is incremental and complex. Hence, the model response has to be computed and expressed numerically. Using the model estimates and the observed values, a numerical scheme using optimization techniques to estimate the model parameters by minimizing the error in the computed response, is described in the next section. In the following section, observed and computed results obtained from this numerical technique are presented and discussed for monotonic and cyclic loading conditions.

4.1 Parameter Estimation Using Optimization

As mentioned earlier, a numerical scheme based on optimization techniques to estimate the numerical values of the parameters of the model presented in the previous chapter is outlined in this section. The objective of the scheme is to find a set of model parameters reproducing the experimental results with minimum error. The total error can be expressed in terms of the observed and computed values of the experiment. This error function serves as the objective function in a minimization algorithm and a solution is found by minimizing the objective function over the acceptable range of the model parameters. In other words, the goal of the numerical scheme presented in this section is to locate the "global minimum" of the objective function and the corresponding model parameters.

The algorithms used in the optimization schemes for this type of non-linear objective function are iterative in nature. The algorithms start with an initial estimate, proceeding to find a sequence of newer estimates, each representing an improvement over the previous estimates. These algorithms are broadly classified as: direct search methods; gradient and second derivative methods; and non-linear least square methods. Of these algorithms the direct search methods are the simplest. These algorithms require only the function values to direct the search, while gradient based methods require the knowledge of the gradients as well. Although direct search methods are less efficient than gradient based methods, they are simple and easier to implement. These algorithms are well suited for applications in which the objective functions are non-smooth or non-differentiable, or the derivatives are discontinuous. On the other hand, nonlinear least squares methods are well suited for minimization problems with nonlinear least squares objective functions, which are expressed as a sum of squares of n ($> n_p$) nonlinear functions.

$$F(P) = \sum_{i=1}^{n} \left[F_i^2\right] \tag{4.1}$$

Where P is the vector of parameters the objective function depends on and n_p is the length of the vector P.

The model performance and observed performance in this study are numerical in nature. Therefore, the gradients of the observed and computed functions have to be computed using finite difference assumptions and any one of the above optimization algorithms can be used to minimize the objective function. But, since the objective function in this study is naturally expressed as a sum of squares of nonlinear functions, a minimization based on a non-linear least squares algorithm (Dennis et al. 1981a, Dennis et al. 1981b) is used in this study.

In general the objective function can be expressed as a function of the experimentally observed and the computed variables, mean total stress, shear stress, volumetric strain, shear strain, and excess pore pressure. Noting that the model presented is based on the effective stress principle and the value of effective stress can be estimated from the values of mean total stress and excess pore pressure, the variables of interest can be limited to these four: mean effective stress, p; shear stress, $q(= \sqrt{3J_2'})$; volumetric strain, $\epsilon_v(= \epsilon_{ii})$; and shear strain, ϵ_q. These variables in general are independent of each other. But, depending on stress path and drainage conditions, the values of some of these variables are known or are interrelated. The observed values of the mean effective stress, shear stress, volumetric strain, and shear strain are denoted with a superscript "o" as, p_i^o, q_i^o, $\epsilon_{v_i}^o$, and $\epsilon_{q_i}^o$, respectively, and are recorded during an experiment at discrete intervals. In addition, since the model is incremental in nature, it is not possible to express the model prediction explicitly as a function of the model parameters and the state parameters. Hence, the predicted values of the above variables, denoted with a superscript "c" as, p_i^c, q_i^c, $\epsilon_{v_i}^c$, and $\epsilon_{q_i}^c$, respectively, are computed numerically at discrete points at which the experimental values of these variables were recorded.

During drained triaxial loading, no excess pore pressure is developed and the ratio between incremental mean effective stress and incremental shear stress is a constant determined by the type of triaxial loading to which soil sample is subjected, such as axial compression/extension, lateral compression/extension, constant mean stress, constant shear stress, or constant shear stress ratio. Considering that the experiment is done on a prescribed stress

path, the natural choice of independent variable is either the mean effective principal stress or the shear stress. Hence, the volumetric strain and shear strain can be expressed as functions of model parameters and mean effective principal stress.

During undrained triaxial loading, on the other hand, no volumetric strain is developed. But due to the development of excess pore pressure, the ratio between incremental mean effective stress and incremental shear stress is not a constant. Therefore, the choice of shear strain as the primary variable seems to be the most logical choice in this case. The mean effective stress and shear stress can be expressed as functions of model parameters and shear strain. The following subsections describe the optimization schemes under these two different choices of primary variables.

4.1.1 Drained Triaxial Loading

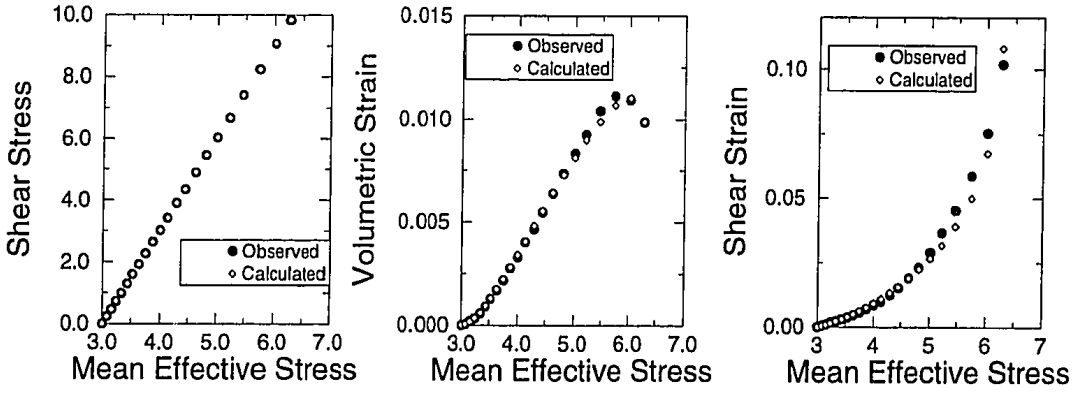


Figure 4.1: Results of a typical drained monotonic axial compression test

Consider a drained axial compression test in which the typical results from the laboratory test and the model prediction are expressed in terms of the mean effective stress, as shown in the Figure (4.1). Since the primary variable is mean effective stress and the stress path is axial compression, the observed parameters, p_i^o and q_i^o, are identical with the computed parameters, p_i^c and q_i^c, producing no errors. The objective function, F, which is a function of model variables G_o, K_o, α, β, ζ, M_c, and M_p, and the state variables, e_0 and e_{min}, therefore

can be expressed in terms of the variables, $\epsilon_{v_i}^o$, $\epsilon_{q_i}^o$, $\epsilon_{v_i}^c$, and $\epsilon_{q_i}^c$, i.e.,

$$F(P) = \sum_{i=1}^{n} \left[(\epsilon_{v_i}^o - \epsilon_{v_i}^c)^2 + (\epsilon_{q_i}^o - \epsilon_{q_i}^c)^2 \right] \tag{4.2}$$

where n is the total number of observed points, and the vector P is a collection of the model parameters and the state parameters:

$$P = < G_o, K_o, \alpha, \beta, \zeta, M_c, M_p, e_o, e_{min} >^T \tag{4.3}$$

The values of P can be estimated by minimizing the above objective function, using standard optimization techniques.

Using the above error function given in the equation (4.2), the optimization techniques will yield a reasonable set of parameters, if the magnitudes of volumetric strain and shear strain data are of the same order. On the other hand, if the volumetric strain and shear strain data follow distinctly different patterns and magnitudes, then the model parameters obtained will reflect an unreasonable weighting towards the variable with the larger magnitude and towards the range at which the large magnitudes occur. This problem can be overcome by estimating the error function with normalized values of the observed data as

$$F(P) = \sum_{i=1}^{n} \left[\left(\frac{\epsilon_{v_i}^o - \epsilon_{v_i}^c}{(\epsilon_{v_i}^o)_{max}} \right)^2 + \left(\frac{\epsilon_{q_i}^o - \epsilon_{q_i}^c}{(\epsilon_{q_i}^o)_{max}} \right)^2 \right] \tag{4.4}$$

where $(\epsilon_{v_i}^o)_{max}$ is the maximum absolute value of the observed volumetric strain, and $(\epsilon_{q_i}^o)_{max}$ is the maximum absolute value of the observed volumetric strain.

4.1.2 Undrained Triaxial Loading

In an undrained triaxial compression test, the choice of shear strain as primary variable is quite convenient as has been discussed earlier. Figure (4.2) shows the typical results from laboratory experiment (i.e., observed) and model predictions (i.e., computed) for an undrained triaxial compression test. As has been discussed earlier, the values of $\epsilon_{v_i}^o$ and $\epsilon_{v_i}^c$ are zero and $\epsilon_{q_i}^o$ is equal to $\epsilon_{q_i}^c$. Therefore, the objective function can be viewed as an error

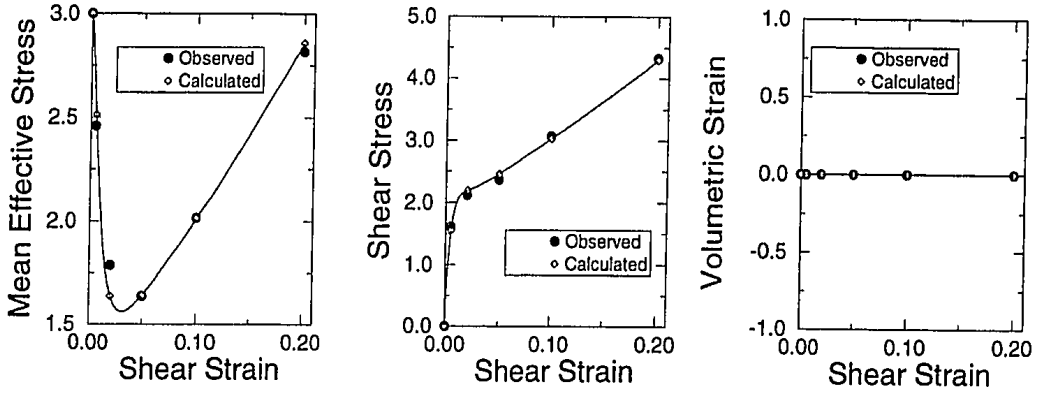


Figure 4.2: Results of a typical undrained monotonic axial compression test

function in terms of the variables, p_i^c, q_i^c, p_i^o, and q_i^o, as

$$F(P) = \sum_{i=1}^{n} \left[(p_i^o - p_i^c)^2 + (q_i^o - q_i^c)^2 \right] \tag{4.5}$$

4.2 *Performance of the Model*

4.2.1 Monotonic Behavior

The performance of the proposed model under monotonic loading conditions is discussed in this subsection, by comparing the observed experimental performance from published sources (Tatsuoka & Ishihara 1973, Tatsuoka & Ishihara 1974b) with the predicted performance. The model performance is predicted with the model parameters obtained using the numerical optimization outlined above, minimizing the error between the observed and predicted data.

Figure (4.3) presents the experimental results (Tatsuoka & Ishihara 1973) and the predicted results of drained triaxial compression tests on soil samples with three different initial confining pressures. The observed and predicted volumetric strain (Figure (4.3a)) and shear strain (Figure (4.3b)) compare well. Similar comparison of observed and experimental data for undrained triaxial compression test is presented in Figure (4.4). Close agreement of the observed and predicted stress paths in conjunction with the strain levels is demonstrated. It

should also be noted that the values of the model parameters are within a reasonable limit, as bounded optimization is used to limit the solution to within a physically reasonable range. The reasonable match of observed and predicted data noted above, favorably evaluates the performance of the model for drained and undrained monotonic triaxial compression tests.

Figures (4.5) and (4.6) show the experimental results (Tatsuoka & Ishihara 1974b) and predicted results for drained stress paths given by Figures (4.5a) and (4.6a). The observed and computed volumetric strains for the above stress paths are depicted in the Figures (4.5b) and (4.6b), respectively. The corresponding shear strains are depicted in the Figures (4.5c) and (4.6c). The above stress paths comprise a combination of axial compression, axial extension and constant shear triaxial loading paths. The model parameters used in the computed results are obtained by the optimization procedure outlined above. These figures show excellent agreement between observed and computed data. This demonstrates the proposed model's validity under a complex loading path with stress reversal.

4.2.2 Cyclic Behavior

The performance of the proposed model under cyclic loading is examined by simulating the qualitative behavior observed in the experimental data from published sources (Tatsuoka & Ishihara 1974a, Ishihara et al. 1975). The simulations included cyclic triaxial tests under stress controlled and strain controlled test conditions with drained and undrained drainage conditions. Although the experimental results and the calculated model results are shown side by side for easy comparison, no attempt was made to identify the values of the model parameters corresponding to the soil in question.

Figure (4.7a) and Figure (4.8a) show the experimental results (Tatsuoka & Ishihara 1974a) of a stress controlled drained test on a loose sand sample subjected to triaxial compression and extension between constant amplitudes of stress ratios, while Figure (4.7b) and (4.8b) show the behavior produced by the proposed model. The volumetric strain change is depicted in Figure (4.7) and it is observed that the volumetric deformation during the initial

cycles is predominantly plastic, and in the subsequent cycles it is essentially elastic. Similar observation could be made with respect to the shear strains (Figure (4.8)) as the hysteresis exhibited becomes smaller with each cycle. Figure (4.9) shows a sample of the same sand subjected to the same test between larger amplitudes of stress ratios producing larger volumetric deformation. More importantly, much of the volumetric deformation in this case is plastic.

Figures (4.10a) and (4.11a) show the experimental results (Tatsuoka & Ishihara 1974a) of a strain controlled drained test on a loose sand sample subjected to triaxial compression and extension between constant amplitudes of shear strains. With each additional cycle the sample sustains higher stress ratios and therefore higher shear stresses. This explains why the response is a combination of responses that were discussed in the previous paragraphs. Figures (4.10b) and (4.11b) show the behavior produced by the proposed model.

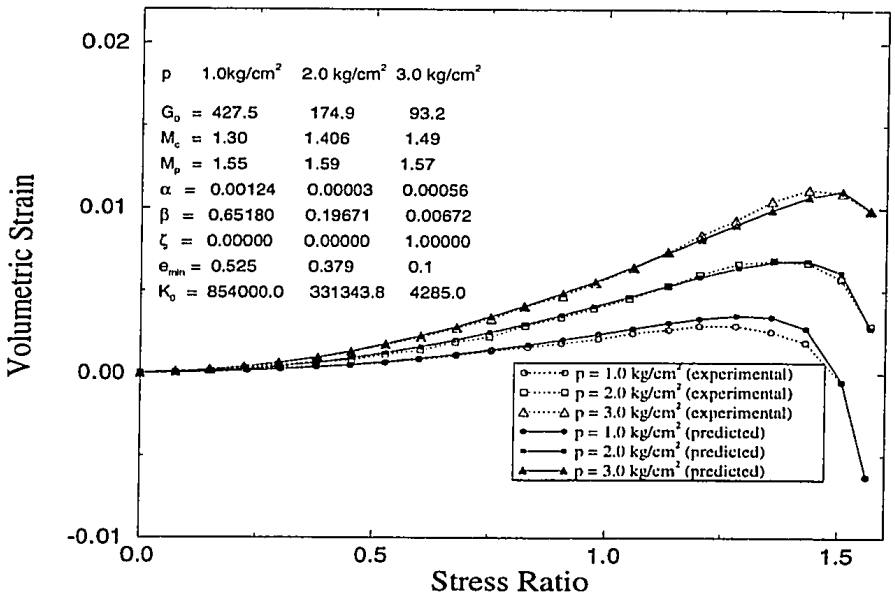
Figures (4.12a) and (4.13a) show the experimental results (Ishihara et al. 1975) of a stress controlled undrained test on a loose sand sample subjected to triaxial compression and extension between constant amplitudes of shear stresses. Due to pore-pressure buildup, the range of stress ratio increases with each additional cycle. The initial buildup of the excess pore pressure is small and the rate of pore-pressure increase is slowly increasing with each additional cycle. However, when the stress ratio approaches the value of the critical state parameter, the rate of increase becomes significantly large. In the subsequent cycles either liquefaction or cyclic mobilty is observed. This behavior is reproduced by the model in Figure (4.12b). As to the shear strain, the magnitude increases with each cycle; however, the increase is significantly large after the liquefaction is initiated. The area of the hysteretic loop exhibited also becomes significantly larger. This behavior is reproduced in Figure (4.13b) by the proposed model.

Figures (4.14a) and (4.15a) show the experimental results (Ishihara et al. 1975) of a strain controlled undrained test on a loose sand sample subjected to triaxial compression and extension between constant amplitudes of shear strains. Unlike in drained conditions the shear stress is decreasing with each cycle and therefore the cyclic shear modulus (se-

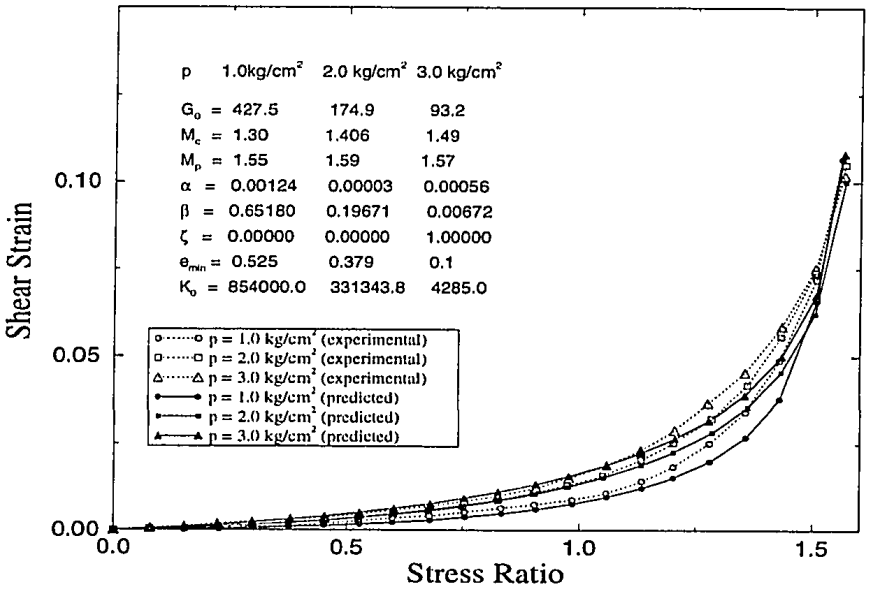
cant) decreases. This is depicted in Figure (4.15c) and Figures (4.14b) and (4.15b) show that the model behavior is in agreement with the experimental behavior.

Summary

The performance of the model presented in Chapter (3) has been verified in this chapter. From the comparisons of model predictions and experimental results, it has been shown that the experimental results are adequately represented by the present model under monotonic and cyclic loading conditions. The subsequent chapters deal with the formulation of the finite element scheme, in view of object-oriented programming techniques.



(a) Volumetric Strain



(b) Shear Strain

Figure 4.3: Comparison between computed and experimental results of drained axial compression tests (after Tatsuoka and Ishihara, 1973)

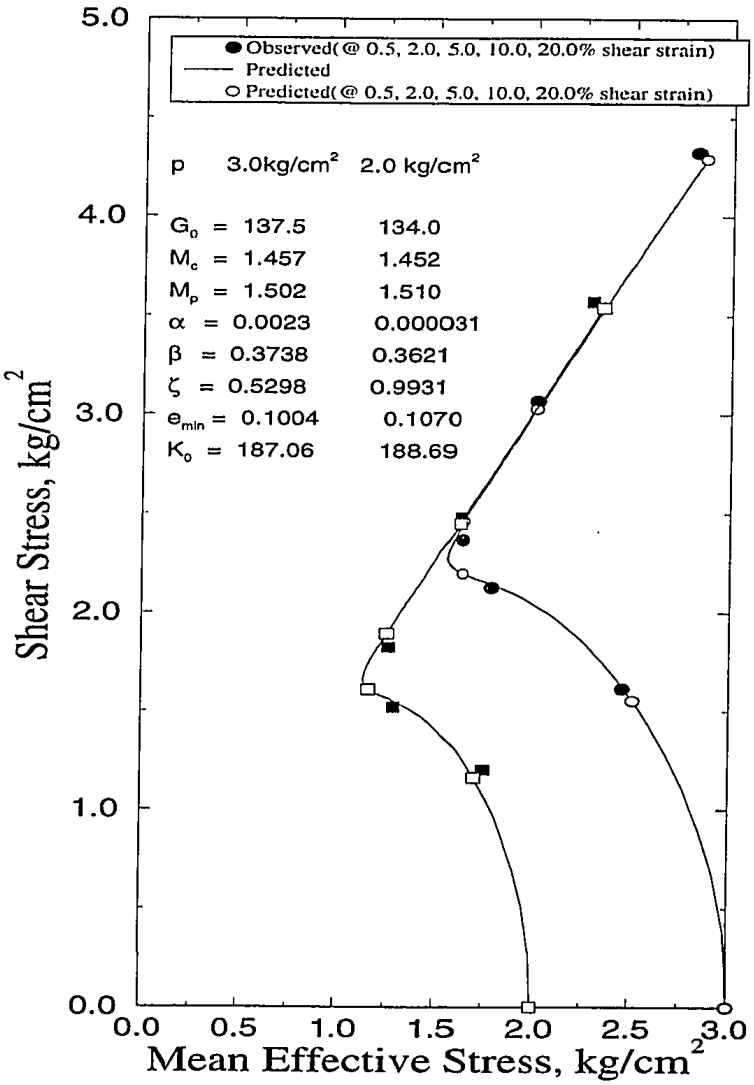


Figure 4.4: Comparison between computed and experimental results of undrained axial compression tests (after Tatsuoka and Ishihara, 1973)

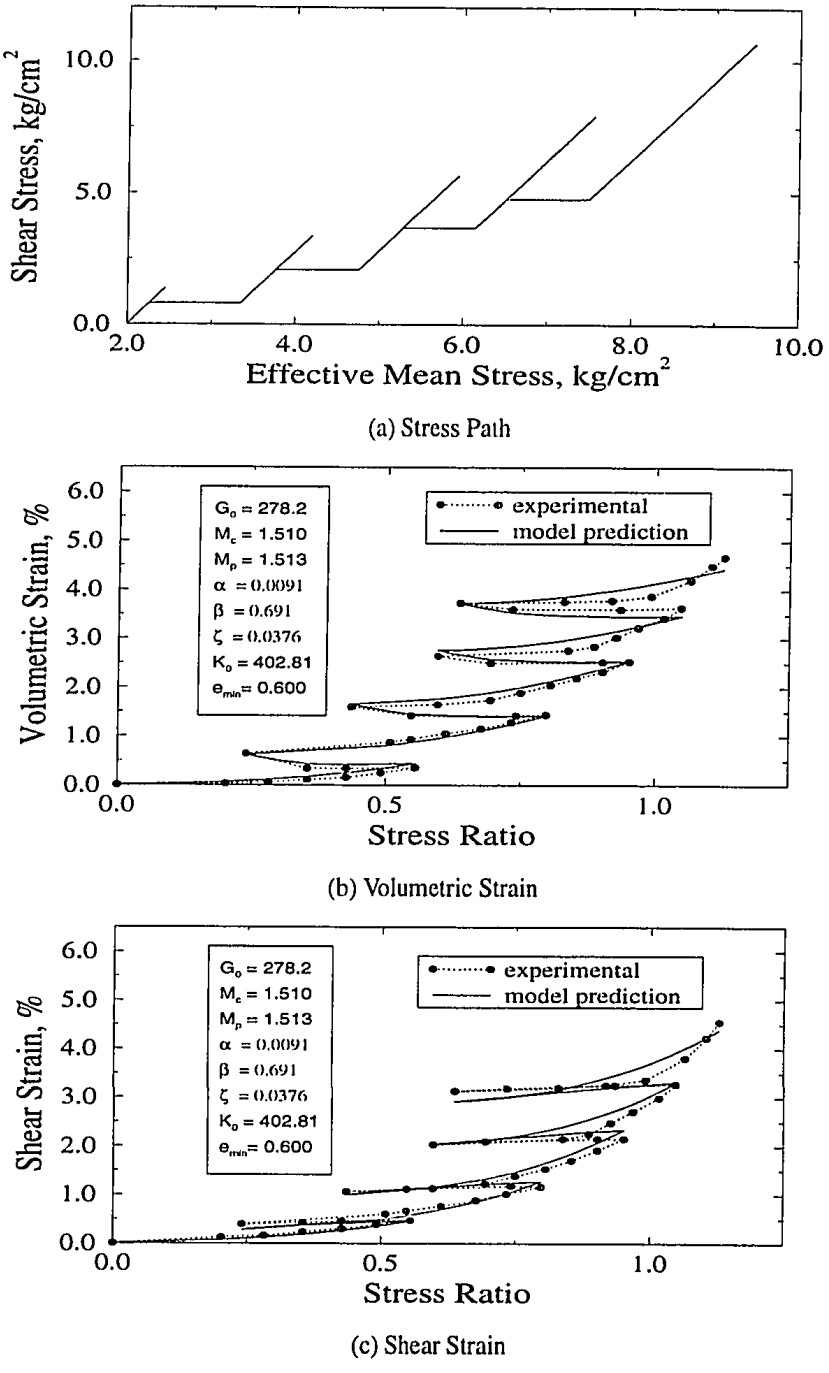


Figure 4.5: Comparison between computed and experimental results (after Tatsuoka and Ishihara, 1974b) for Fuji Sand

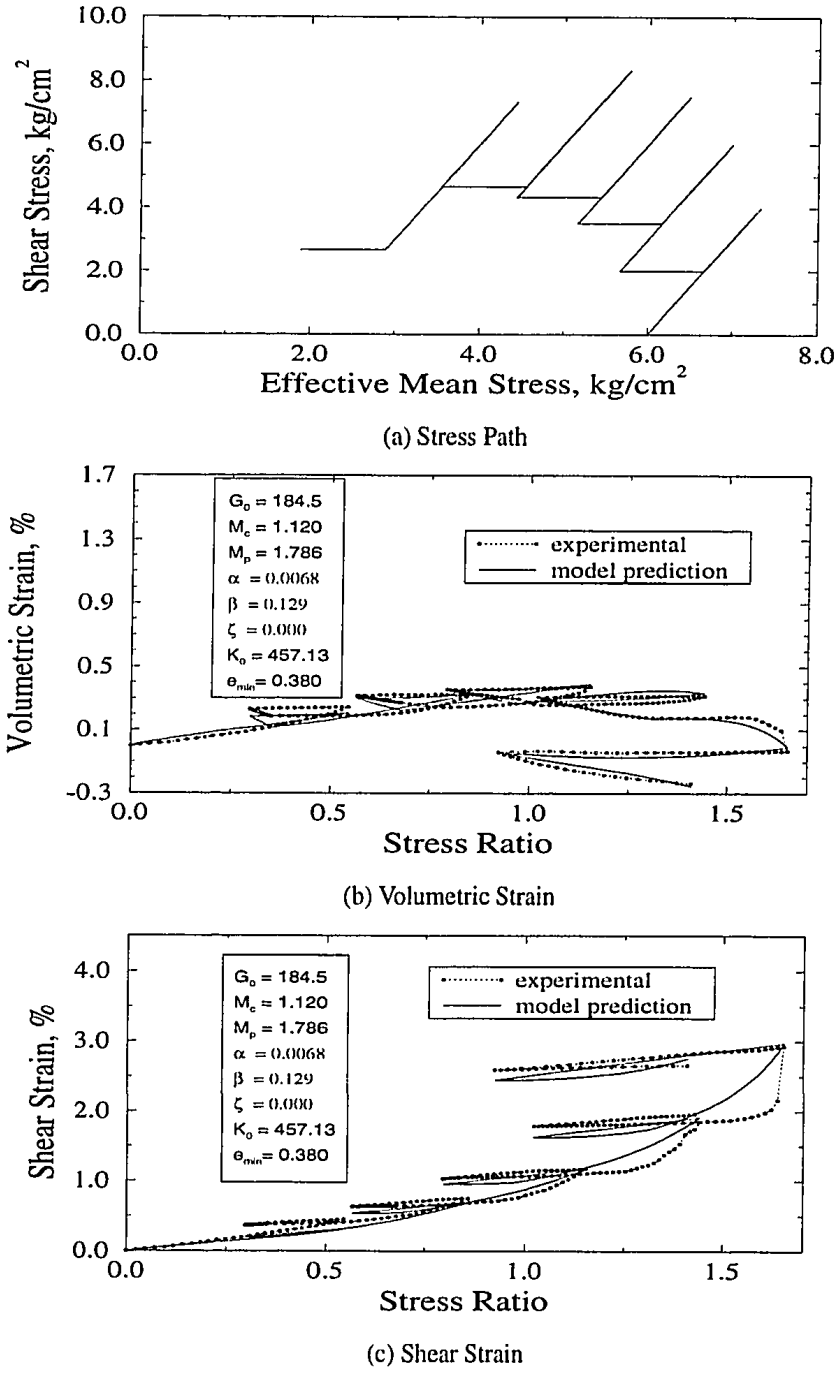
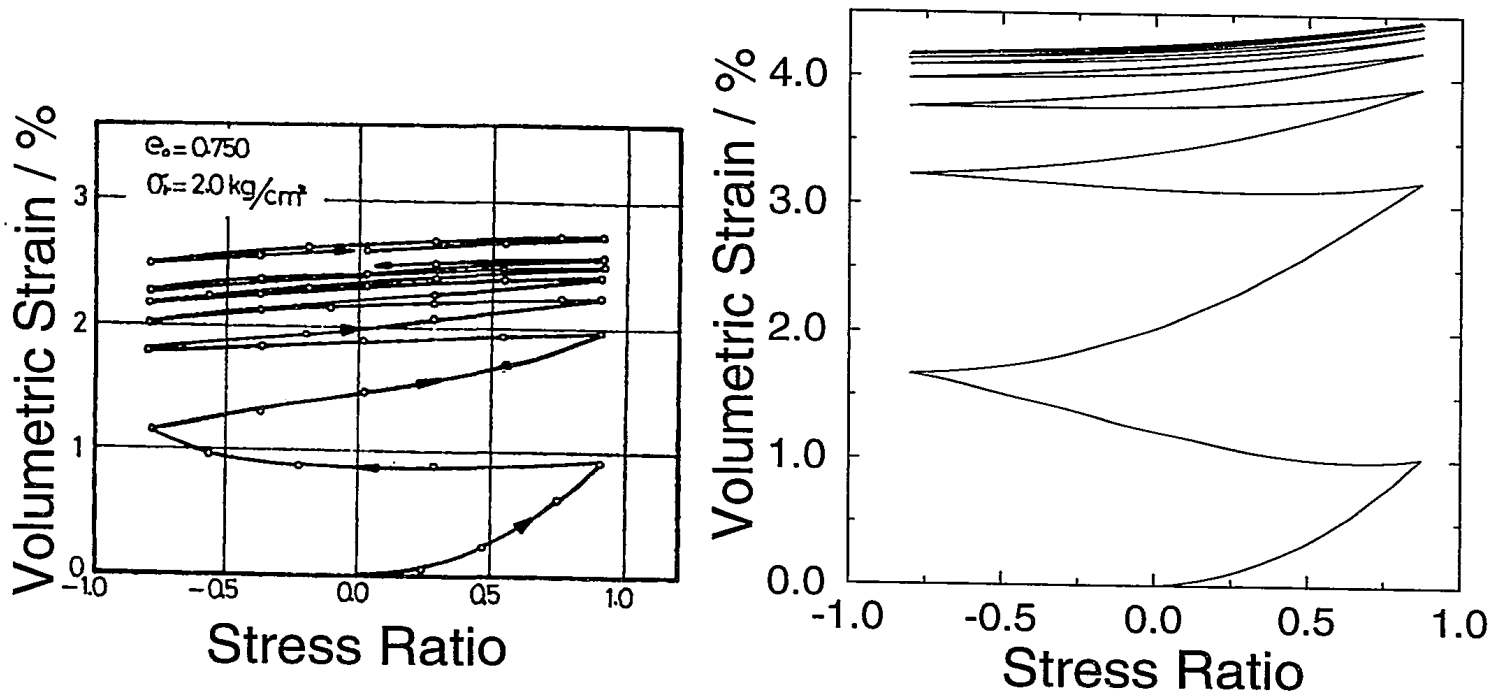


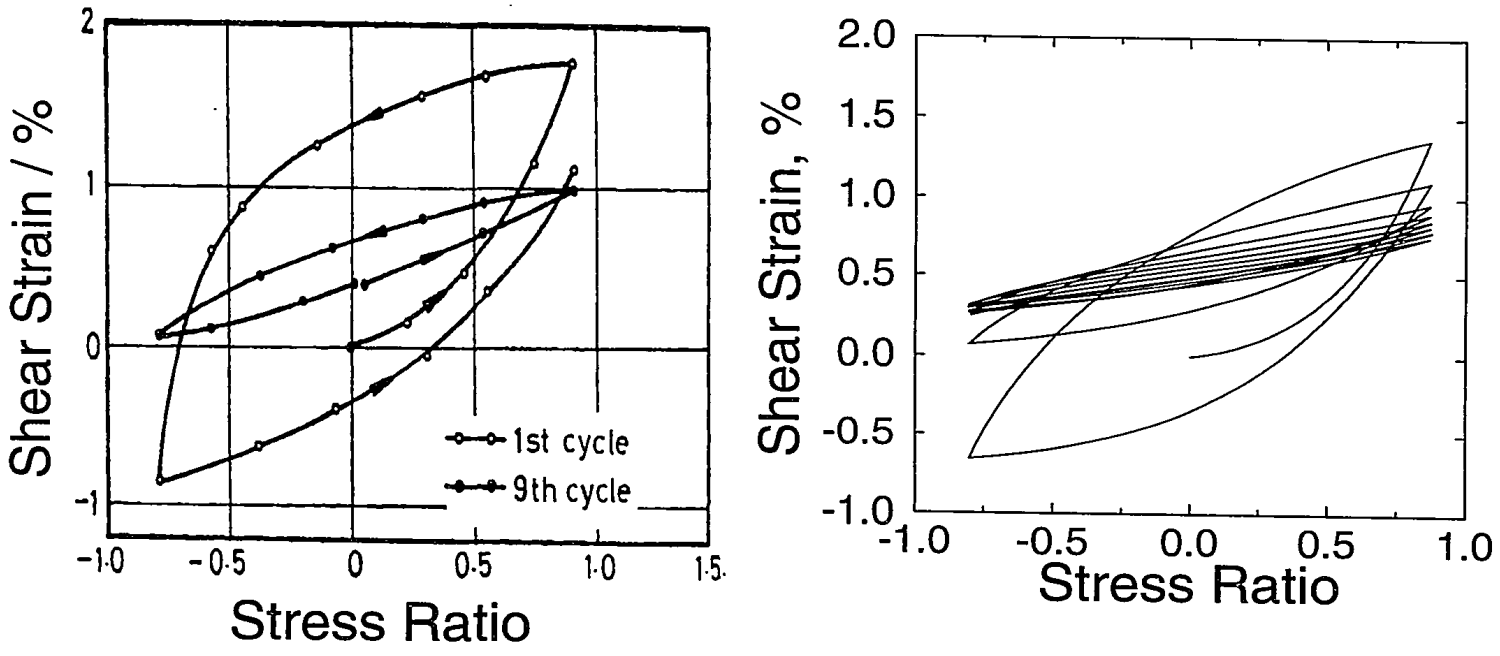
Figure 4.6: Comparison between computed and experimental results (after Tatsuoka and Ishihara, 1974b) for Fuji Sand



a). Experimental (after Tatsuoka and Ishihara, 1974a)

b). Model behavior

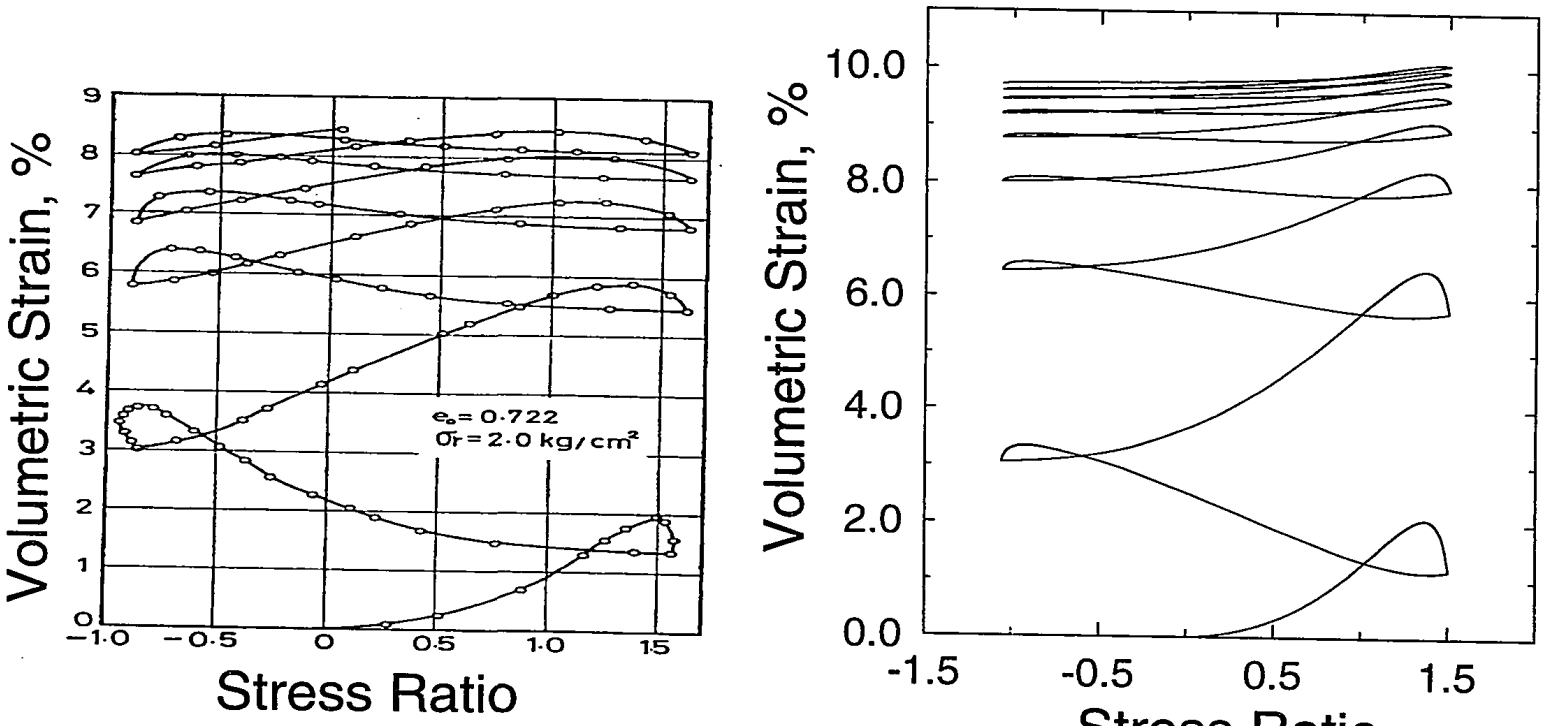
Figure 4.7: Volumetric strain against stress ratio under stress controlled drained cyclic triaxial test with medium amplitude of stress ratio



a). Experimental (after Tatsuoka and Ishihara, 1974a)

b). Model behavior

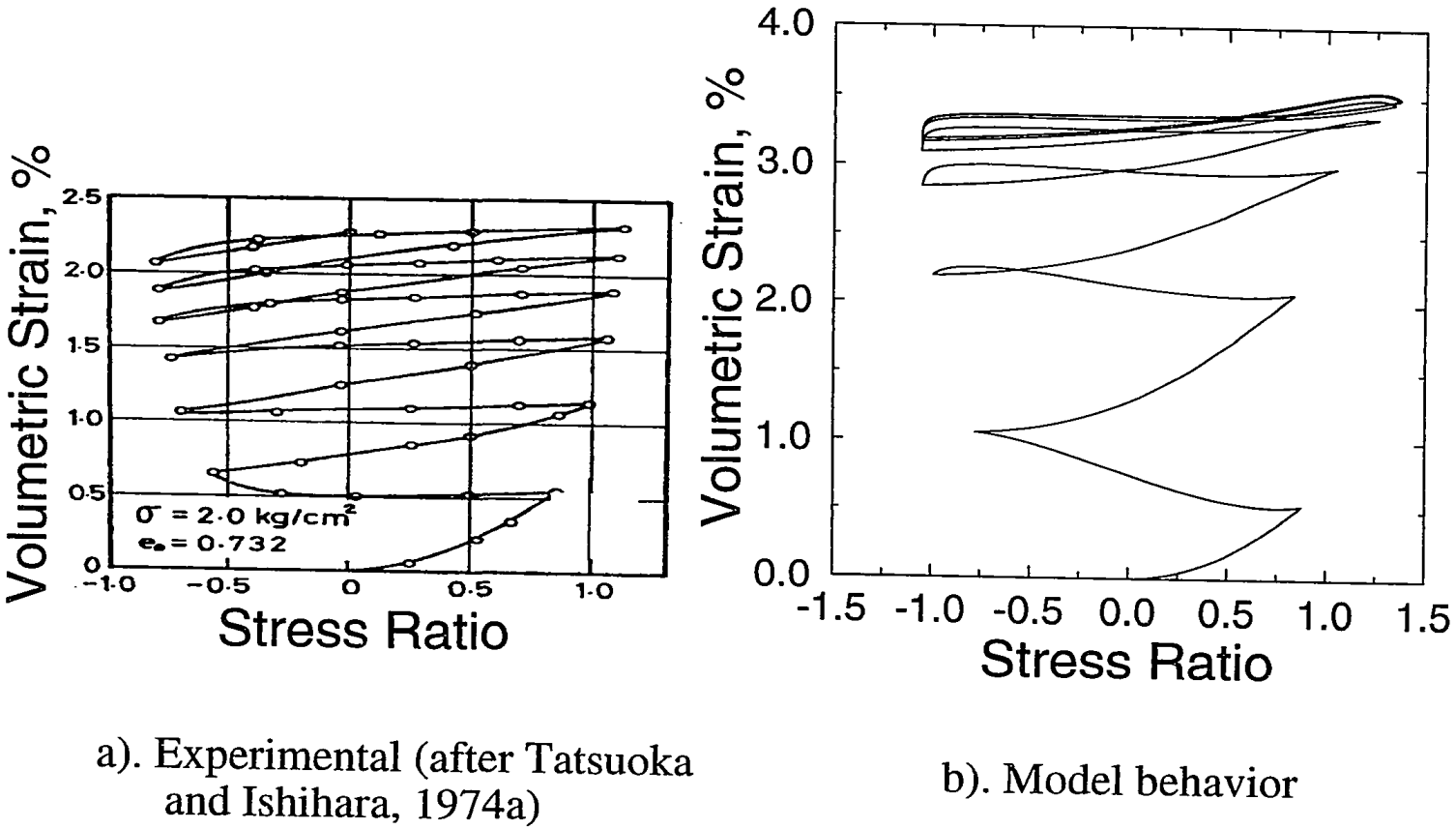
Figure 4.8: Shear strain against stress ratio under stress controlled drained cyclic triaxial test with medium amplitude of stress ratio



a). Experimental (after Tatsuoka and Ishihara, 1974a)

b). Model behavior

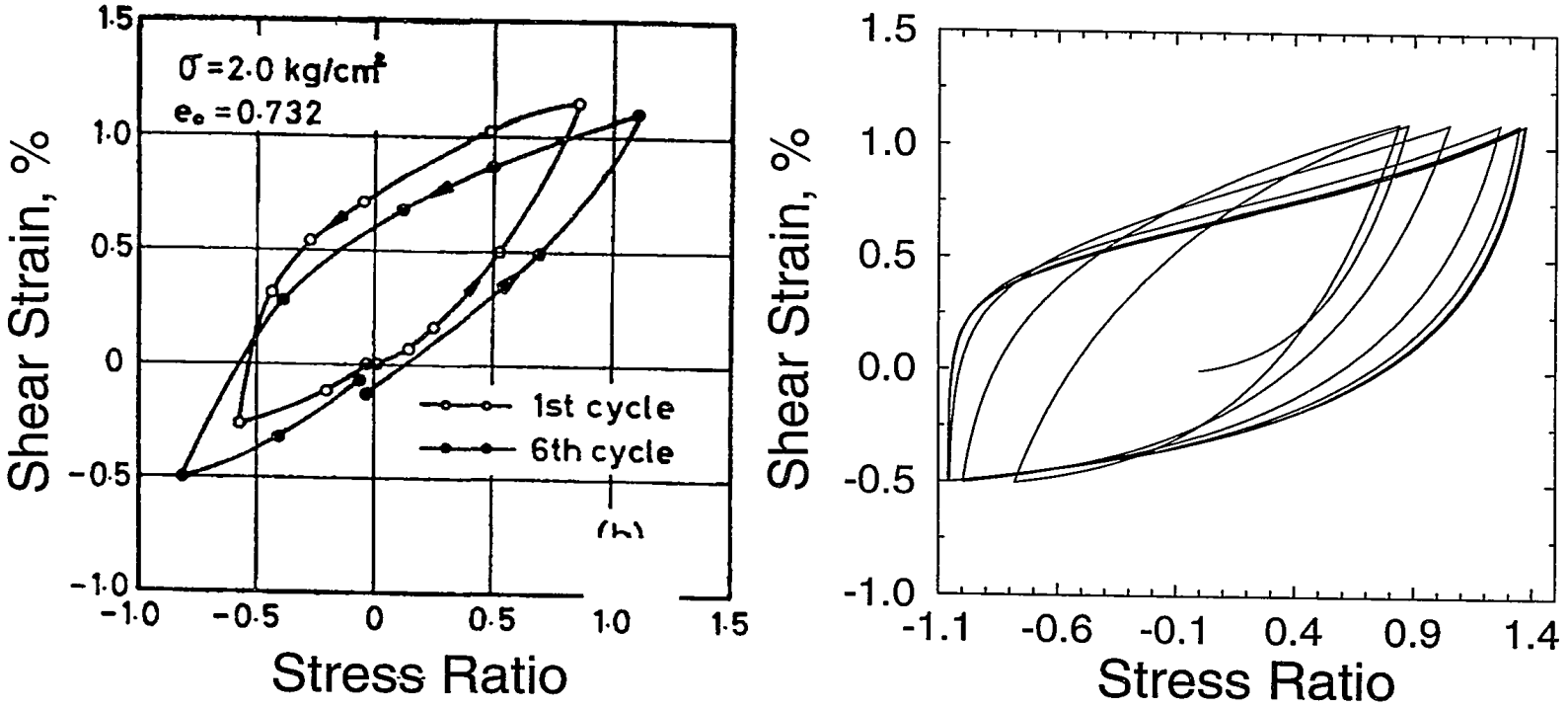
Figure 4.9: Volumetric strain against stress ratio under stress controlled drained cyclic triaxial test with large amplitude of stress ratio



a). Experimental (after Tatsuoka and Ishihara, 1974a)

b). Model behavior

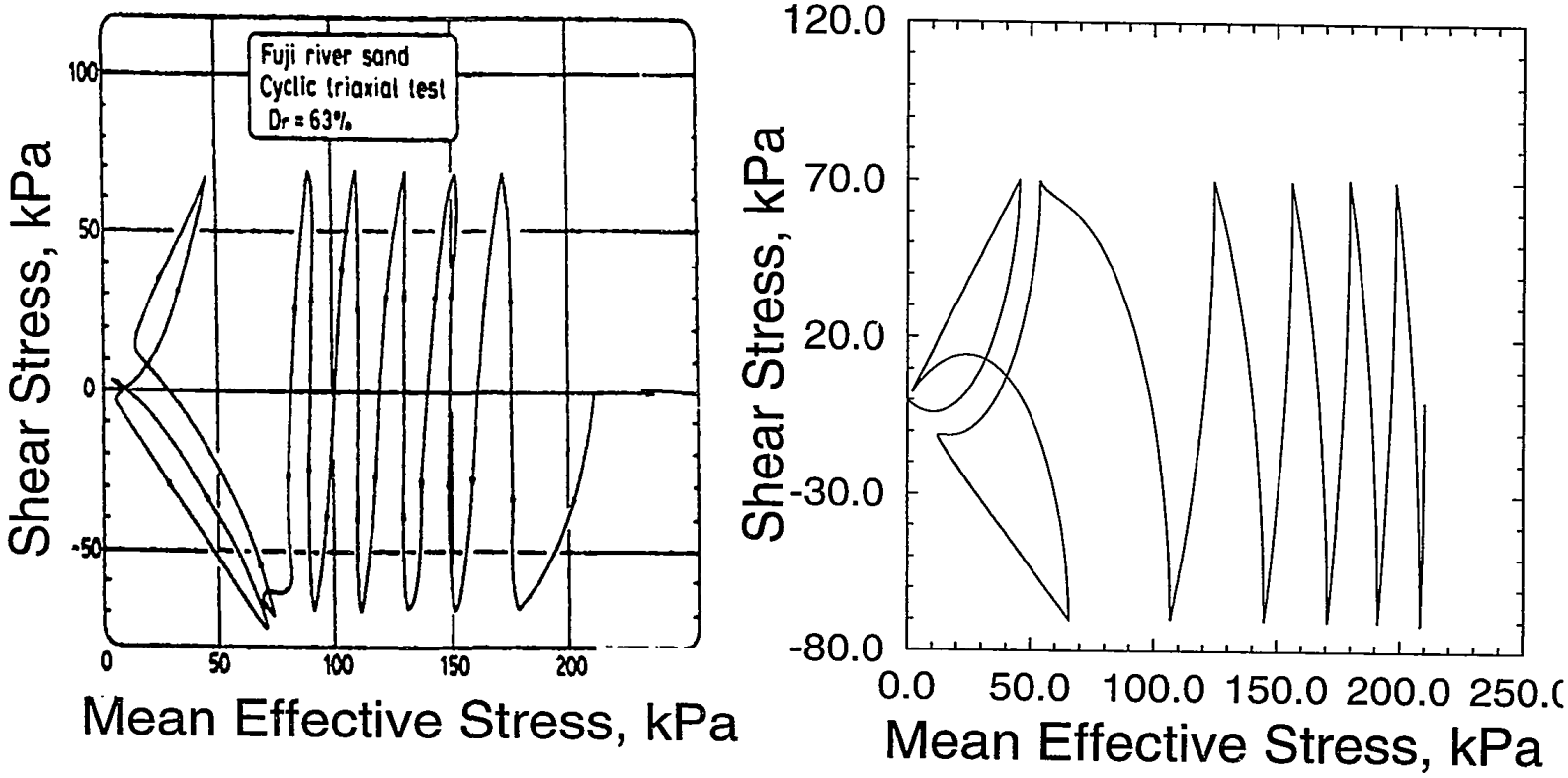
Figure 4.10: Volumetric strain against stress ratio under constant amplitude strain controlled drained cyclic triaxial test



a). Experimental (after Tatsuoka and Ishihara, 1974a)

b). Model behavior

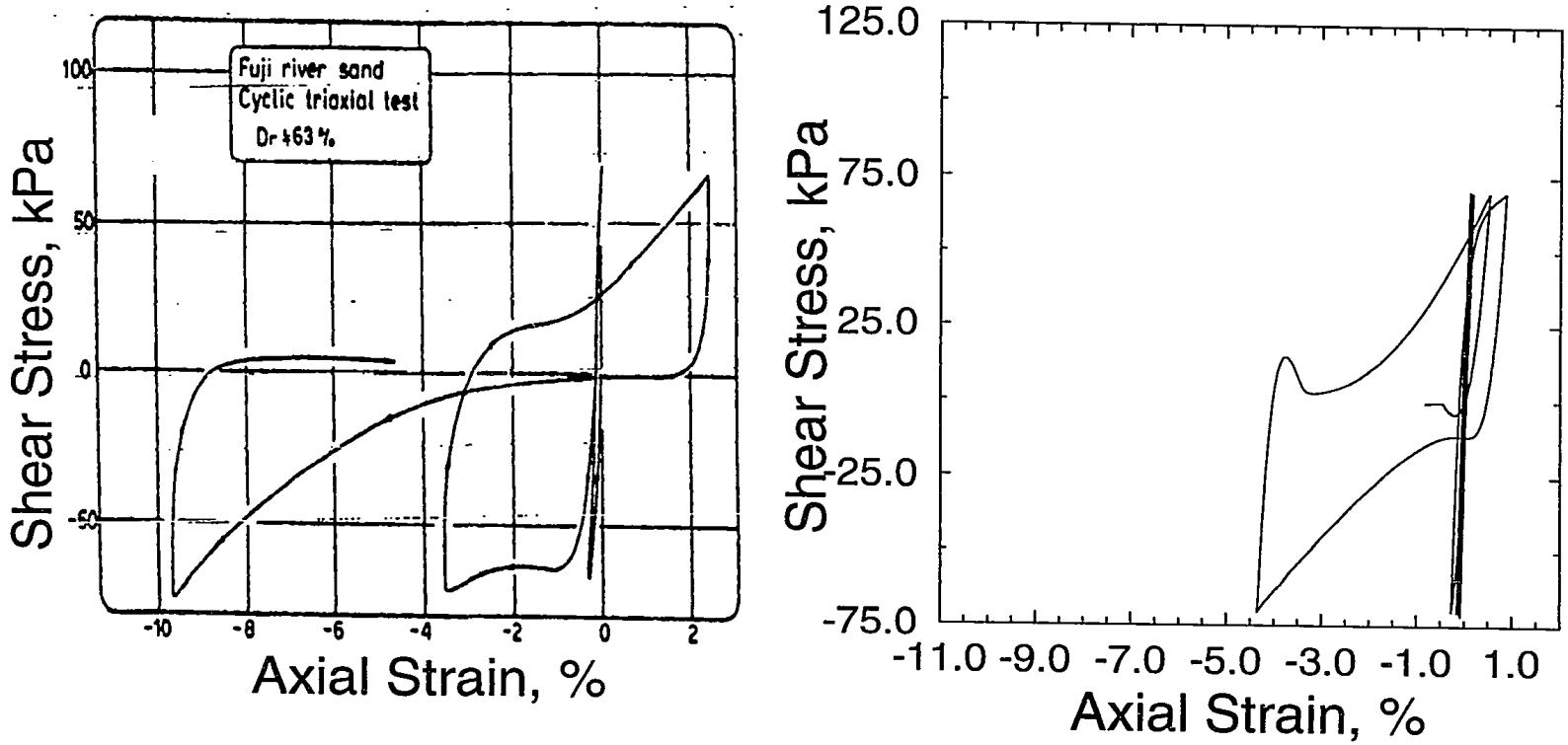
Figure 4.11: Shear strain against stress ratio under constant amplitude strain controlled drained cyclic triaxial test



a). Experimental (after Ishihara et al., 1975)

b). Model behavior

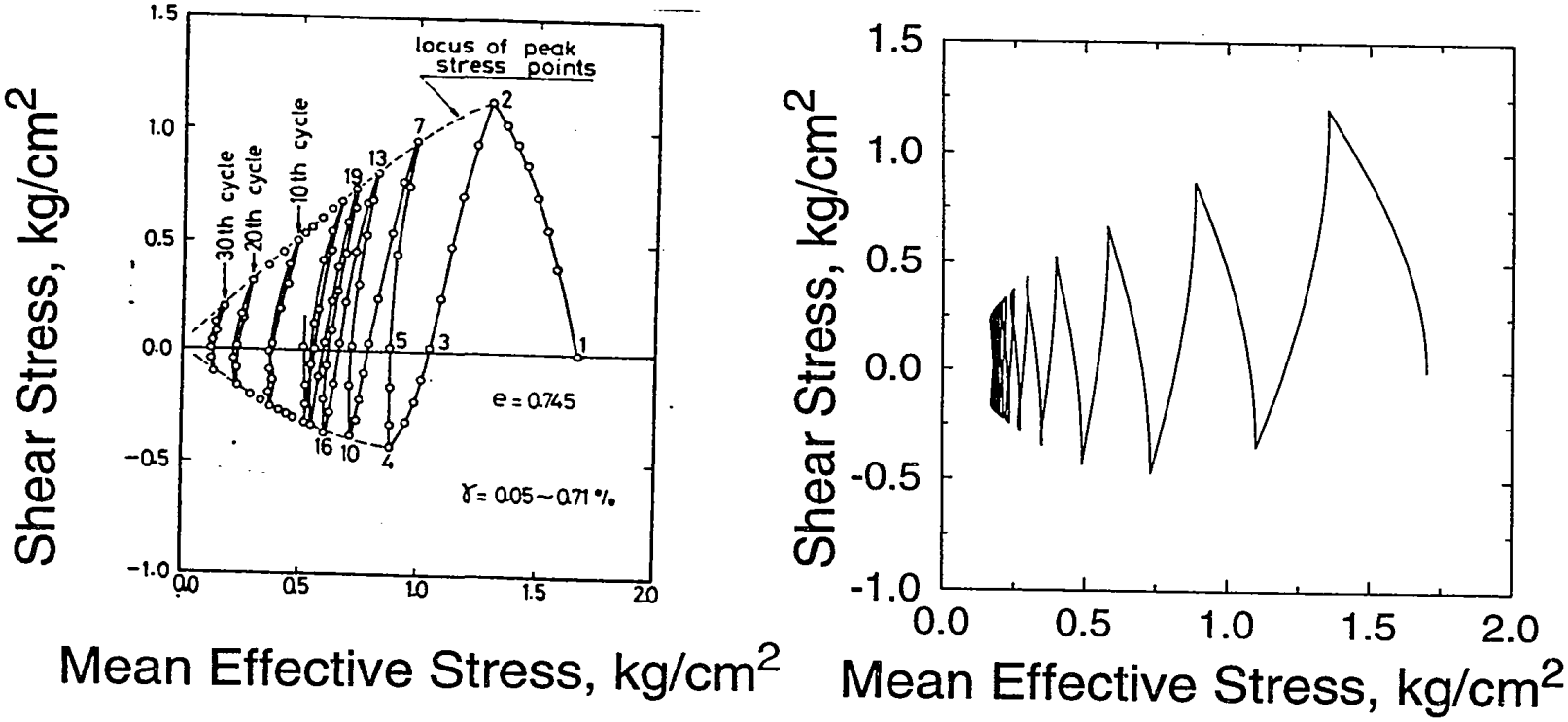
Figure 4.12: Shear stress against mean effective Stress under constant amplitude stress controlled undrained cyclic triaxial test



a). Experimental (after Ishihara et al., 1975)

b). Model behavior

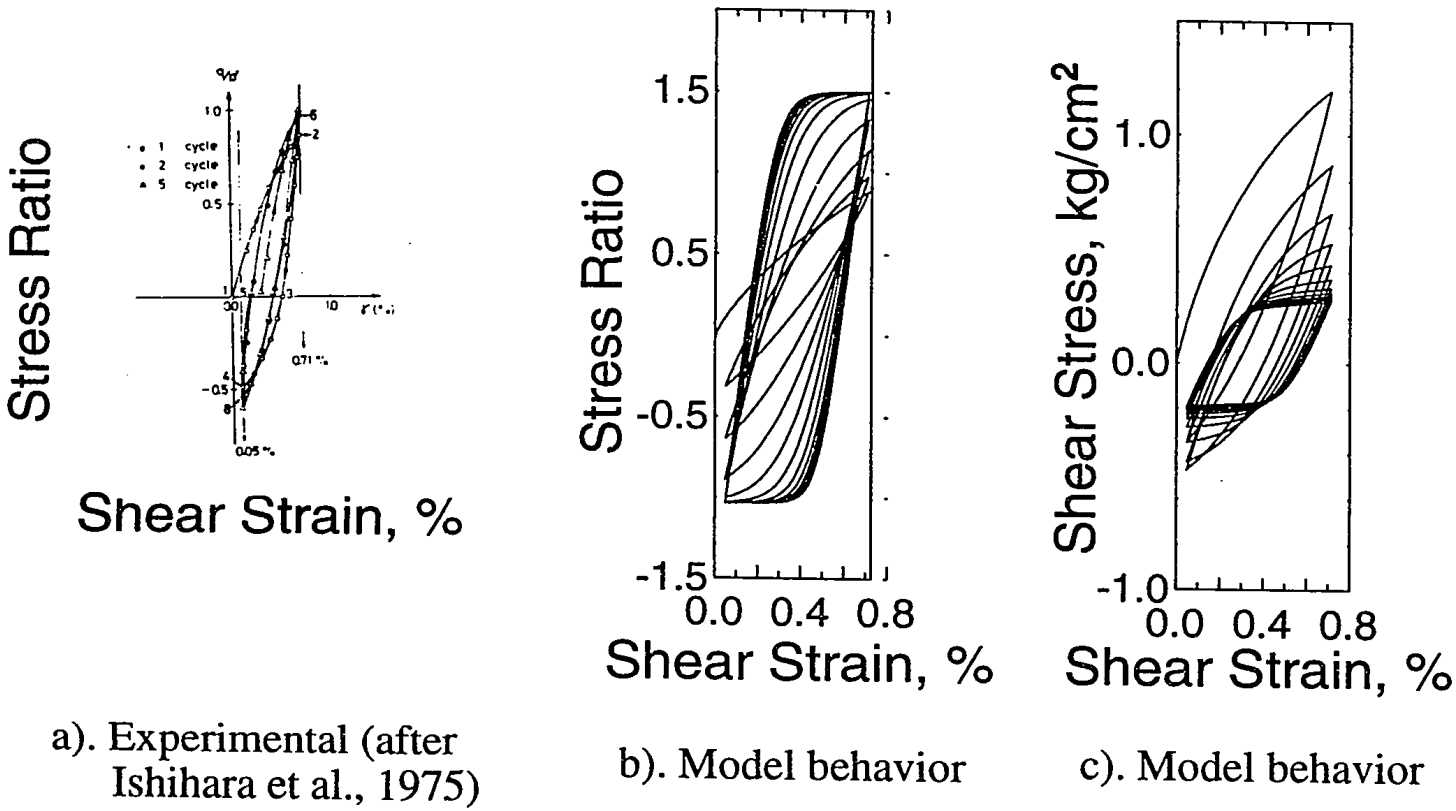
Figure 4.13: Shear stress against axial Strain under constant amplitude stress controlled undrained cyclic triaxial test



a). Experimental (after Ishihara et al., 1975)

b). Model behavior

Figure 4.14: Shear stress against mean effective Stress under constant amplitude strain controlled undrained cyclic triaxial test



a). Experimental (after Ishihara et al., 1975)

b). Model behavior

c). Model behavior

Figure 4.15: Shear stress against axial Strain under constant amplitude strain controlled undrained cyclic triaxial test

Chapter 5

FINITE ELEMENT FORMULATION

This chapter presents a numerical scheme to model dynamic geomechanics problems with continuum and porous, continuum type finite elements. Since a coordinate free geometric system is used, no implicit assumptions with regard to the dimensions of the embedding space are made in this presentation. As the study is focused on solving dynamic problems and non-linear material models, a time-marching direct integration solution scheme is utilized in the design and presentation of the study. The matrix equations resulting from the finite element analysis with lumped mass formulations can be simplified to yield a set of uncoupled simultaneous equations as will be demonstrated in this chapter. This solution scheme is used in the following chapter to illustrate the design and implementation of the finite element software developed as part of this study.

5.1 Finite Element for Porous Media

This section presents the finite element formulation of a general continuum porous finite element. The governing equations and boundary conditions pertaining to this formulation are also presented. The formulation of a continuum type finite element is derived as a special case of the porous continuum element.

5.1.1 Governing Equations

The dynamic equilibrium of the saturated porous media may be expressed as:

$$\nabla \cdot \sigma + \nabla p + \rho F = \rho \ddot{u} \tag{5.1}$$

where σ, p, ρ , F, and u are the effective stress tensor, the pore pressure, the total mass density of the soil, the body force vector, and the displacement vector, respectively. The superposed dot denotes differentiation with respect to time; hence, $\ddot{u}$ is the acceleration vector. The strain tensor, ϵ, is expressed in terms of the displacement vector, u, as:

$$\epsilon = \frac{1}{2}[\nabla u + (\nabla u)^T + \nabla u(\nabla u)^T] \tag{5.2}$$

The stress-strain relationship of the solid phase, in general, may be expressed as:

$$\sigma = \int_0^t [D : d\epsilon]\, dt + \sigma_o \tag{5.3}$$

where D is the fourth order stiffness tensor, σ_o is the initial stress tensor, and the operator : denotes double contraction between tensors. Neglecting the inertia effects of the water, we assume that the diffusion process is governed by Darcy's law as:

$$q = H \cdot (\nabla p + \rho_f F) \tag{5.4}$$

where q is the flux vector; H is the permeability tensor, and ρ_f is the density of the water. Note that the assumptions by Darcy and Biot regarding the flow of fluid in the porous media are tacitly assumed to hold. Further, the condition of saturation implies that the outflow from the soil must equal the reduction in volume.

$$\nabla \cdot q = -trace(\dot{\epsilon}) = -\nabla \cdot \dot{u} \tag{5.5}$$

On the assumption that there is no instantaneous dilatation at the time of load application, the convolution product associated with the equation (5.4), can be written as:

$$g * \nabla \cdot [H \cdot (\nabla p + \rho_f F)] + \nabla \cdot u = 0 \tag{5.6}$$

in which the notation $*$ denotes convolution product, such that:

$$v * w = \int_0^t v(x,t)w(x,t-\tau)d\tau \tag{5.7}$$

and,

$$g = 1 \tag{5.8}$$

5.1.2 *Boundary Conditions*

The traction and the displacement boundary conditions associated with the deformation of the porous medium are:

$$(\sigma + pI) \cdot n = \hat{T} \quad on \ S_t \tag{5.9}$$

$$u = \hat{u} \quad on \ S_u \tag{5.10}$$

where S_t and S_u are the respective portions of the boundary of the volume V in which the traction and deformation are prescribed, n is the unit outward normal vector to the boundary of the volume V, and I is the second order identity tensor.

For the diffusion process, the boundary conditions associated with the prescribed pore pressure and the prescribed flow of pore water are expressed as:

$$p = \hat{p} \quad on \ S_p \tag{5.11}$$

$$q \cdot n = \hat{Q} \quad on \ S_q \tag{5.12}$$

where S_p and S_q are the respective portions of the boundary of the volume V in which the pore pressure and flow are prescribed.

5.1.3 *Formulation*

A functional $\Omega(u, p)$ can be constructed taking u and p as primary field variables, subject to equations (5.3), (5.2), (5.10), and (5.11), such that the trial functions of u and p that produce a stationary value of Ω automatically satisfy the field equations (5.1) and (5.4), and the natural boundary conditions, equations (5.9) and (5.12), at any time $t \in [0, \inf)$. An expression of Ω can be written as:

$$\begin{aligned} \Omega(u,p) \ = \ & \int_V [\tfrac{1}{2}\sigma : \epsilon - \rho F \cdot u + p\nabla u - \tfrac{1}{2} g * q \cdot (\nabla p + \rho_f F)] dv - \\ & \int_{S_t} \hat{T} \cdot u ds + \int_{S_q} [g * \hat{Q} p] ds \end{aligned} \tag{5.13}$$

The region V is discretized into an assemblage of finite elements; as a result, equation (5.13) can be written as the accumulation of individual element contributions, i.e.:

$$\Omega(u,p) = \sum_e \Omega^e(u,p) \tag{5.14}$$

in which

$$\begin{aligned}\Omega^e(u,p) &= \int_{V^e} [\tfrac{1}{2}\sigma : \epsilon - \rho F \cdot u + p\nabla u - \tfrac{1}{2} g * q \cdot (\nabla p + \rho_f F)] dv - \\ &\int_{S_t^e} \hat{T} \cdot u ds + \int_{S_q^e} [g * \hat{Q} p] ds\end{aligned} \tag{5.15}$$

where V^e denotes the volume of the element, and S_t^e and S_q^e are the respective parts of S_t and S_q that are associated with the element.

Discretization

Now, consider the element submanifold, $\mathcal{M} \in E^3$, defined by the map $\phi : R^m \rightarrow E^3$ such that $P = \phi(\xi^1, \ldots, \xi^m), P \in \mathcal{M}, m \leq 3$. The map ϕ can be expressed in terms of shape functions as:

$$\phi(\xi^\alpha) = N^i(\xi^\alpha) P_i \tag{5.16}$$

in which the notation $\phi(\xi^\alpha)$ implies $\phi(\xi^1, \ldots, \xi^m)$, P_i represent the set of nodal points of the element, $N^i(\xi^\alpha)$ represent the corresponding set of scalar shape functions, and summation over i is implied.

The natural basis in the parametric tangent space, $\partial/\partial\xi^\alpha$ induces a basis in the physical space, which can be calculated as:

$$g_\alpha = \frac{\partial \phi}{\partial \xi^\alpha} \tag{5.17}$$

Similarly, the parametric dual basis has a physical counterpart, g^β, which can be determined from the requirement:

$$g_\alpha \cdot g^\beta = \delta_\alpha^\beta \tag{5.18}$$

in which δ_α^β is the Kronecker delta.

The element displacements can be represented by a vector field $u : R^m \rightarrow T_p(\mathcal{M})$. That is, u assigns to each parametric point a vector in the physical tangent space. This displacement field can be expressed in terms of the nodal values by using the interpolation functions, N_u, as:

$$u(\xi^\alpha, t) = \sum_{i=1}^{n_u} N_u^i(\xi^\alpha) u_i(t) \tag{5.19}$$

where N_u^i and u_i are the values of the displacement interpolation function and the values of the displacement corresponding to the i^{th} node, and n_u is the number of element nodes associated with the displacement degree of freedom. Similarly, the fluid pore pressure field, p, within each element can be expressed in terms of the nodal values by using the interpolation functions, N_p, as:

$$p(\xi^\alpha, t) = \sum_{i=1}^{n_p} N_p^i(\xi^\alpha) p_i(t) \tag{5.20}$$

where N_p^i and p_i are the value of the pore pressure interpolation function and the value of the pore pressure corresponding to the i^{th} node, and n_p is the number of element nodes associated with the pore pressure degree of freedom.

The strain-displacement relation, equation (5.2), is expressed in terms of tensors. For our purposes, it is preferable to reformulate the relation in terms of tensor products of vectors, allowing the coordinate-free representation of the strain to be expressed in geometric terms. To this end, the deformation gradient term, ∇u, can be expressed as:

$$\nabla u = g^\alpha \otimes \frac{\partial u}{\partial \xi^\alpha} \tag{5.21}$$

Substituting this relation, equation (5.21), equation (5.2) can be expressed as

$$\epsilon = \tfrac{1}{2}[g^\alpha \otimes \frac{\partial u}{\partial \xi^\alpha} + \frac{\partial u}{\partial \xi^\alpha} \otimes g^\alpha + \left(g^\alpha \otimes g^\beta\right) \frac{\partial u}{\partial \xi^\alpha} \cdot \frac{\partial u}{\partial \xi^\beta}] \tag{5.22}$$

This equation provides a general coordinate-free representation for the strain tensor at any parametric point. For computational purposes it is helpful to use the relation:

$$g^\alpha \otimes \frac{\partial u}{\partial \xi^\alpha} = (g^\alpha \otimes g^\beta) g_\alpha \cdot \frac{\partial u}{\partial \xi^\beta} \tag{5.23}$$

to express equation (5.22) as:

$$\epsilon = \tfrac{1}{2}[g_\alpha \cdot \frac{\partial u}{\partial \xi^\beta} + \frac{\partial u}{\partial \xi^\alpha} \cdot g_\beta + \frac{\partial u}{\partial \xi^\alpha} \cdot \frac{\partial u}{\partial \xi^\beta}](g^\alpha \otimes g^\beta) \tag{5.24}$$

Equation (5.24) can be rearranged as

$$\epsilon = \tfrac{1}{2}\left(g^\alpha \otimes g^\beta + g^\beta \otimes g^\alpha\right)[\left(g^\beta + \tfrac{1}{2}\frac{\partial u}{\partial \xi^\beta}\right) \cdot \frac{\partial u}{\partial \xi^\alpha}] \tag{5.25}$$

In view of equation (5.19), equation (5.25) can be rewritten as:

$$\epsilon = \sum_{i=1}^{n_u} N_\epsilon^i(\xi^\alpha) \cdot u_i \tag{5.26}$$

in which

$$N_\epsilon^i(\xi^\alpha) = \tfrac{1}{2}\frac{\partial N_u^i(\xi^\alpha)}{\partial \xi^\alpha}\left(g^\alpha \otimes g^\beta + g^\beta \otimes g^\alpha\right) \otimes \left(g^\beta + \tfrac{1}{2}\frac{\partial u}{\partial \xi^\beta}\right) \tag{5.27}$$

Where $N_\epsilon^i(\xi^\alpha)$ is a third order tensor. Using this result, equation (5.3) can be expressed as:

$$\sigma = \sum_{i=1}^{n_u} \int_0^t [D : (N_\epsilon^i(\xi^\alpha) \cdot u_i)]\, dt + \sigma_o \tag{5.28}$$

where σ_o is the initial stress tensor in the element before any deformation and D is the fourth order tensor describing the constitutive behavior of the material.

Similarly, equation (5.4) can also be expressed in terms of the nodal values of the displacements and pore pressure as:

$$q = \sum_{i=1}^{n_p} \{H \cdot [(\frac{\partial N_p^i(\xi^\alpha)}{\partial \xi^\alpha} p_i) g^\alpha + \rho_f N^i F_i]\} \tag{5.29}$$

In view of equations (5.26), (5.28), and (5.29), and by taking variations of the equation (5.15) with respect to u_i and p_i and by eliminating convolution, we get

$$M \cdot \ddot{U} + K \cdot U + Y \cdot P = \overline{B} - \overline{\overline{B}} + \overline{R} \tag{5.30}$$

$$Y^T \cdot \dot{U} - \overline{K} \cdot P = \overline{\overline{B}} - \overline{\overline{R}} \tag{5.31}$$

in which

$$M_{ij} = \sum_{e=1}^{n_e} \int_{V^e} \left[\rho N_u^i N_u^j\right] dv \tag{5.32}$$

$$K_{ij} = \sum_{e=1}^{n_e} \int_{V^e} [N_\epsilon^i : D : N_\epsilon^j] dv \tag{5.33}$$

$$Y_{ij} = \sum_{e=1}^{n_e} \int_{V^e} N_p^i \frac{\partial N_u^j}{\partial \xi^\alpha} g^\alpha dv \tag{5.34}$$

$$\overline{K}_{ij} = \sum_{e=1}^{n_e} \int_{V^e} \left[H \cdot \left(\frac{\partial N_p^i}{\partial \xi^\alpha} g^\alpha \right) \cdot \left(\frac{\partial N_p^j}{\partial \xi^\beta} g^\beta \right) \right] dv \tag{5.35}$$

$$\overline{B}_i = \sum_{e=1}^{n_e} \int_{V^e} \frac{1}{2} [\sigma_o : N_\epsilon^i] dv \tag{5.36}$$

$$\overline{\overline{B}} = \sum_{e=1}^{n_e} \int_{V^e} \left[\rho N_u^i F \right] ds \tag{5.37}$$

$$\overline{\overline{B}}_i = \sum_{e=1}^{n_e} \int_{V^e} \frac{1}{2} \left[\rho_f \frac{\partial N_p^i}{\partial \xi^\alpha} [(H \cdot g^\alpha) \cdot F] \right] dv \tag{5.38}$$

$$\overline{R}_i = \sum_{e=1}^{n_e} \int_{S_t^e} \left[N_u^i \hat{T} \right] ds \tag{5.39}$$

$$\overline{\overline{R}}_i = \sum_{e=1}^{n_e} \int_{S_q^e} \left[N_p^i \hat{Q} \right] ds \tag{5.40}$$

where the matrix equations (5.30) and (5.31) form the simultaneous equations in terms of the nodal variables for the entire structure. Physically, $\overline{B}$ is the force component due to the initial stress, σ_o; $\overline{\overline{B}}$ and $\overline{\overline{B}}$ are force components due to body forces; $\overline{R}$ and $\overline{\overline{R}}$ are force components due to boundary traction and flow conditions respectively; and K, Y and $\overline{K}$ constitute the stiffness components.

From equation (5.31), the variable P can be solved for and by substituting this into equation (5.30), we get

$$M \cdot \ddot{U} + C \cdot \dot{U} + K \cdot U = F_{ext} \tag{5.41}$$

in which

$$C = Y \overline{K}^{-1} Y^T \tag{5.42}$$

$$F_{ext} = Y \overline{K}^{-1} \left(\overline{\overline{B}} - \overline{\overline{R}} \right) + \overline{B} - \overline{\overline{B}} + \overline{R} \tag{5.43}$$

The matrix differential equation (5.41) above is similar to the general dynamic matrix equilibrium equations. The matrix C defined by the equation (5.42) represents the equivalent damping matrix of the structure.

5.1.4 *Finite Element for Continua*

A general formulation of a finite element for the dynamic analysis of porous media has been presented in the above subsection. It turns out, the final differential matrix equations obtained in equation (5.41) are identical in form for a finite element formulation of a continuum element. For this problem the dynamic equilibrium equations are given by

$$\nabla \cdot \sigma + \rho F = \mu \dot{u} + \rho \ddot{u} \tag{5.44}$$

in which μ is a constant reflecting the damping properties of the material. The matrix equilibrium equations can be derived as

$$M \cdot \ddot{U} + C \cdot \dot{U} + K \cdot U = F_{ext} \tag{5.45}$$

where

$$F_{ext} = \overline{B} - \overline{\overline{B}} + \overline{R} \tag{5.46}$$

$$M_{ij} = \sum_{e=1}^{n_e} \int_{V^e} \left[\rho N_u^i N_u^j \right] dv \tag{5.47}$$

$$C_{ij} = \sum_{e=1}^{n_e} \int_{V^e} \left[\mu N_u^i N_u^j \right] dv \tag{5.48}$$

$$K_{ij} = \sum_{e=1}^{n_e} \int_{V^e} [N_\epsilon^i : D : N_\epsilon^j] dv \tag{5.49}$$

$$\overline{B}_i = \sum_{e=1}^{n_e} \int_{V^e} \frac{1}{2} [\sigma_o : N_\epsilon^i] \, dv \tag{5.50}$$

$$\overline{\overline{B}} = \sum_{e=1}^{n_e} \int_{V^e} \left[\rho N_u^i F \right] ds \tag{5.51}$$

$$\overline{R}_i = \sum_{e=1}^{n_e} \int_{S_t^e} \left[N_u^i \hat{T} \right] ds \tag{5.52}$$

The above equations (5.46) - (5.52) summarize the components of the matrix differential equation (5.45). The "consistent" mass matrix defined above has off-diagonal terms which in-turn lead to "mass coupling" and hence making computations more complex. On the other hand, the inertia forces could be simply expressed in terms of the accelerations and concentrated masses at the nodes, producing a lumped mass matrix for the analysis. The

resulting matrix is diagonal and has non-zero diagonal terms associated with translational degrees of freedom. The following section briefly describes the direct integration algorithm implemented in the study, and describes an element based solution strategy presenting a set of uncoupled dynamic equilibrium equations. This solution strategy eliminates the need for explicitly assembling the global stiffness matrices.

5.1.5 Solution Scheme

The procedures for the solution of general systems of matrix differential equations can become computationally expensive if the order of the matrices is large. In the following, the special characteristics of the coefficient matrices M, C, and K are taken into account in the choice of a solution scheme to reduce the computational effort. As mentioned earlier, a lumped mass system is considered and a direct integration solution scheme is presented in this chapter, although the design of the computational framework itself is general and can accommodate various numerical solution schemes. In a direct integration solution scheme, the differential equations are integrated using a numerical step-by-step procedure satisfying the differential equation at discrete time intervals, Δt, apart. This means that the dynamic equilibrium equations given by equation (5.45) could be viewed as static equilibrium equations including the effects of inertia and damping, with the need for satisfying equilibrium at discrete time intervals. Different direct integration solution schemes are available, each being based on distinct assumptions concerning the form of the variation of displacements, velocities, and accelerations. The Newmark method (Newmark 1959) is used in the derivations presented in this chapter.

The Newmark method is essentially an implicit method requiring the solution of a non-diagonal system of equations. The assumptions for the variations of velocities and displacements are given by

$$\dot{U}_{t+\Delta t} = \dot{U}_t + \left[(1-\delta)\,\ddot{U}_t + \delta \ddot{U}_{t+\Delta t}\right]\Delta t \tag{5.53}$$

$$U_{t+\Delta t} = U_t + \Delta t \dot{U}_t + \left[\left(\tfrac{1}{2} - \gamma\right)\ddot{U}_t + \gamma \ddot{U}_{t+\Delta t}\right](\Delta t)^2 \tag{5.54}$$

where δ and γ are parameters that can be determined to obtain integration accuracy and stability, and the equilibrium equations are satisfied at $t+\Delta t$. For consistent mass matrices, the Newmark method remains implicit for all values of parameters, δ and γ. In case of a lumped mass model, an explicit scheme can also be formulated producing uncoupled system of equations, and each individual value can be computed directly from the corresponding value at the previous time step. This explicit scheme, with $\delta = \frac{1}{2}$, is given as

$$\dot{\hat{U}}_{t+\Delta t} = \dot{U}_t + \frac{\Delta t}{2}\ddot{U}_t \tag{5.55}$$

$$\hat{U}_{t+\Delta t} = U_t + \Delta t\dot{U}_t + (\Delta t)^2\left(\tfrac{1}{2} - \gamma\right)\ddot{U}_t \tag{5.56}$$

$$\dot{U}_{t+\Delta t} = \dot{\hat{U}}_{t+\Delta t} + \frac{\Delta t}{2}\ddot{U}_{t+\Delta t} \tag{5.57}$$

$$U_{t+\Delta t} = \hat{U}_{t+\Delta t} + (\Delta t)^2\gamma\ddot{U}_{t+\Delta t} \tag{5.58}$$

in which $\dot{\hat{U}}_{t+\Delta t}$ and $\hat{U}_{t+\Delta t}$, given by equations (5.55) and (5.56), are the approximate estimates of the velocities and displacements at $t + \Delta t$, and equations (5.57) and (5.57) give the corresponding updated values. Equations (5.57) and (5.58) are identical with that of the implicit formulation given by equations (5.53) and (5.54). The following equation is considered to satisfy equilibrium.

$$M \cdot \ddot{U}_{t+\Delta t} + C \cdot \dot{\hat{U}}_{t+\Delta t} + K \cdot \hat{U}_{t+\Delta t} = F_{ext(t+\Delta t)} \tag{5.59}$$

In view of equations (5.54), (5.55), and (5.56), the above equilibrium equation (5.59) can be rewritten as

$$\begin{aligned}\left\{\frac{1}{\gamma(\Delta t)^2}M\right\}U_{t+\Delta t} = {} & \left\{\frac{1}{\gamma\Delta t^2}M - K\right\}U_t + \left\{\frac{1}{\gamma\Delta t}M - C - \Delta tK\right\}\dot{U}_t + \\ & \left\{\left(\frac{1}{2\gamma} - 1\right)M - \frac{\Delta t}{2}C - \left(\tfrac{1}{2} - \gamma\right)\Delta t^2K\right\}\ddot{U}_t + \\ & F_{ext(t+\Delta t)}\end{aligned} \tag{5.60}$$

As noted earlier, the solution of the above recurrence scheme does not require inversion of M for lumped mass systems, i.e., it is explicit. Consistent mass systems result in an implicit recurrence scheme, which requires the inversion of M. For most practical problems

M remains constant and is independent of the values of U. However, in the case of non-linear problems one or more matrices M, C, K or the vector F_{ext} depend on the values of U. The application of time marching scheme to non-linear problems, in general, typically requires iteration within each time step. An alternative to iteration is to extrapolate the various matrices M, C, K from the previously established steps (Zienkiewicz 1977). Such extrapolation leads to more stable solutions, and instabilities that may arise solely due to iteration are thereby avoided. A very simple extrapolation uses the values of these matrices corresponding to those at the previous time step. Accumulation of errors can be overcome by correcting equilibrium equations for unbalanced forces at the end of each step.

5.2 Summary

For the analysis of dynamic soil-structure interaction problems, finite element formulations have been presented for porous and continuum conditions of the soil media. Based on these formulations, a direct integration solution scheme briefly has been outlined for use in the practical soil-structure interaction boundary value problems using a finite element analysis program. The design, development, and implementation details of this proposed finite element software are presented in the next chapter.

Chapter 6

COMPUTATIONAL FRAMEWORK

In the previous chapter the process of discretization is viewed as a mathematical exercise to solve the governing partial differential equations of an engineering problem. The discretization process provides an engineer with a mechanism to model complex geometry and to treat non-homogeneity of the material properties. These components of the discretization and additional conceptual objects used in modeling, analysis, and presentation form the basis of the design of the proposed computational framework for dynamic soil-structure interaction analysis. The framework presented in this chapter is based on the concepts of object-oriented programming. These concepts and the ensuing discussion do not rely on the use of any specific programming language. However, C++ was used in this study during the prototyping and implementation stages of the design. Therefore, C++ code fragments and terminology are used occasionally in the discussion for illustrative purposes.

The design of an object-oriented analysis requires the identification of classes of objects, their attributes, and the functionalities of these objects. This involves: determining the objectives of the software; identifying the candidate classes, properties and functionalities; identifying the user-interfaces of the software; and refining the designed classes. The end-result of the design of the computational framework using object-oriented programming technique is, therefore, the identification of abstract data types and the associated functionality pertinent to describe the physical and numerical models of interest, as well as the data handling and presentation techniques and tools. The classes necessary for the graphical user interface (GUI) are derived from an existing object-oriented graphics library *Unidraw* (Vlissides 1990) based on the *InterViews* (Linton et al. 1988) graphics library. Ideally, the GUI related classes and the numerical analysis related classes would be completely

independent to allow switching among different GUI libraries. As this is not possible, the objective of the design is to minimize the effort needed to make such a switch by limiting the interdependence between the analysis related classes and the GUI related classes. The GUI-independent classes as well as GUI-dependent classes used in this study are discussed in detail in the following sections.

6.1 Analysis Related Classes

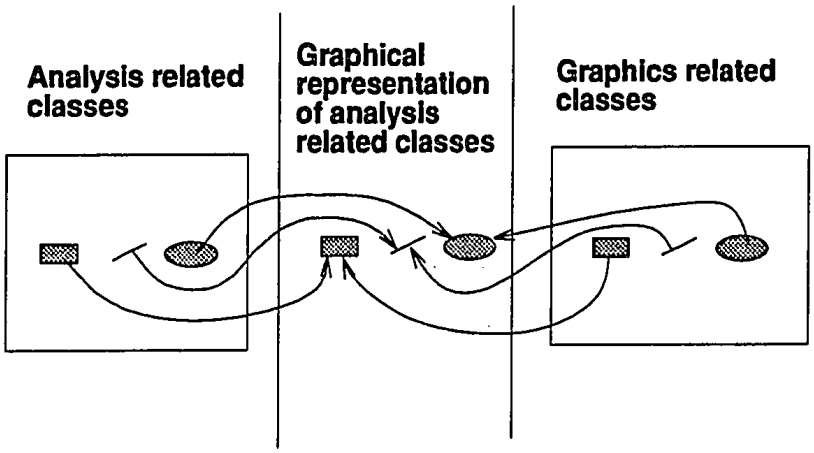


Figure 6.1: Separation of analysis and graphics classes

As mentioned earlier, the bulk of the analysis related classes are kept free of GUI dependence and are described in this section. Objects that require a graphical representation are identified and classes defining their graphical representations are discussed in the next section. The information flow between the analysis and graphics classes are handled through these graphical representation objects as shown in Figure (6.1). Numerical modeling and analysis procedures, especially finite element method, is built on concepts such as nodes, degrees of freedom, elements, materials, stresses, strains, loads, and constraints. Each of these components as well as the overall analysis and data management structure are dis-

cussed and appropriate abstract classes are defined in the following subsections .

6.1.1 Nodes

In any numerical method, nodes primarily represent the discretized topography of the problem as well as the locations at which the primary variables of the differential equations are either specified or to be solved. A node is also associated with internal degrees of freedom, constraints, applied loads, and optionally, a set of attached elements besides its location. The roles of these entities in the definition of the `Node` class is briefly discussed below.

The response of a node to generalized forces is governed by its associated degrees of freedom. Explicit awareness of individual degrees of freedom is equivalent to explicit awareness of individual vector components, which counters the spirit of the coordinate-free geometry concept. For this reason, a `Node`'s degrees of freedom are encapsulated within the generalized degree of freedom classes, described in the next subsection. This encapsulation makes it possible to shield the `Node` class from the details of the degree of freedom implementation, and to support the dynamic alteration of the type and number of a `Node`'s degrees of freedom. This latter capability can be very useful in interactive environments because the degrees of freedom of a particular node depends on the nature of the connecting elements.

The role of constraints is to model generalized kinematical restrictions on the generalized displacements of a node. Unlike in traditional finite elements, the constraints are handled by suppressing the internal and external force components along the constrained directions. A distinct `Constraint` class manages this operation, and is described in detail in one of the following subsections.

For static problems, it would be simple to store loading information directly with each node. For dynamic loads, however, it is preferable to give each load its own identity, and to make them the active agents, with the `Nodes` being passive. Thus the nodal loads are implemented in distinct classes, as will be described later in this chapter. `Nodes` have the capability to receive generalized load information dynamically, but the `Node` itself is shielded

from the details of how the load is implemented.

With respect to attached elements, the role of nodes is analogous to the role of vertices in graph theory. That is, the topology of the system is represented by the connectivity of the nodes. For traditional finite elements, ordering of this topological structure is quite important, since the degree of banding of the system stiffness matrix is related to the ordering of the nodes. As will be seen later, the approach presented here is essentially immune to nodal ordering, and so connectivity monitoring within the Node class turns out to be a relatively minor function that is not even strictly necessary. But to compute the element properties and to ensure the appropriate load transformation between elements, information on the topology of the system is needed. A detailed discussion on this topic is included in the subsection which discusses elements.

The geometry of the soil-structure interaction problem and the underlying geometric manipulations can be represented by an abstract class, Node. The underlying geometric manipulations, however, are not trivial and could be handled efficiently by defining separate classes for geometric manipulations. The definition of abstract geometric classes allows different representations of the geometry and such representation details are hidden from the definitions of the class Node. The finite element formulations presented in the previous chapter use a coordinate free geometric notation. A set of abstract classes defining the geometric manipulations presented in the following subsection has been inspired directly by the coordinate-free geometric system presented by DeRose (1989) based on abstract affine geometry.

Coordinate Free Geometry

The coordinate-free geometric system discussed by DeRose is based on abstract affine geometry. This system allows the analytical geometric manipulations and calculations be expressed in abstract, coordinate-free, unambiguous terms. This makes it possible to glean the expressive clarity and logical consistency built into rigorous formulations of geometric concepts, along with the generality inherent in the abstractions themselves. As discussed by

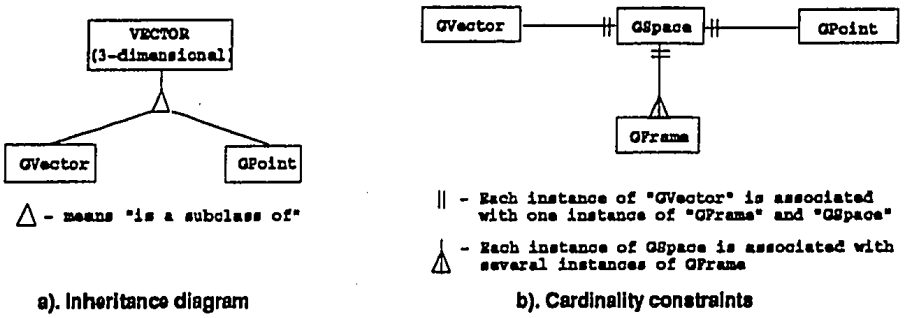


Figure 6.2: Classes for geometric manipulations

DeRose, one can ensure that only operations that make sense geometrically are performed, by requiring all geometric operations to take place through abstractions. For example, consistent with affine geometry, distinction can be made between vectors and points by defining distinct classes, say `GPoint` and `GVector`. These two classes could be derived from an abstract base class `Vector` inheriting the common three dimensional vector operations (Figure (6.2a)). With this class structure, geometrically meaningless operations, such as adding two points, can be avoided. Operations that do make sense, such as adding a vector to a point to obtain a new point, can be performed without regard to details such as the dimension of the space or the coordinate system used to implement the geometric entities; i.e., all geometric operations can be expressed in a coordinate-free, dyadic form. The graphical representation of a structure defined on a computational model requires different geometric spaces and transformation between such geometric spaces (`GSpace`). For example, the space in which the structure is defined is generally referred to as the 'world coordinate system', while the space in which the representation is defined is referred to as 'screen coordinate system'. Geometric operations between different spaces are disallowed to retain the clarity of the approach, while the transformations from one space to another is permitted. In each such space, there could be several different coordinate systems and such coordinate systems are defined by the class `GFrame` and this is in parallel to the idea of several local

Table 6.1: Protocol for geometric classes GVector and GPoint

CLASS	OPERATION	ARGUMENTS	EFFECT
GVector	Operator+	GVector	returns GVector
		GPoint	returns GPoint
	Operator-	GVector	returns GVector
		-	returns GVector
	Operator*	Scalar	returns GVector
	Operator/	Scalar	returns GVector
GPoint	Operator+	GVector	returns GPoint
	Operator-	GPoint	returns GVector
-	Dot	GVector, GVector	returns the *Dot product*
	Cross	GVector, GVector	returns the *Cross product*

coordinate systems. Each GFrame object has the knowledge to transfer the geometric objects to and from the underlying coordinate system, i.e., GFrame, of the GSpace, and this coordinate system could be viewed as the global coordinate system. This treatment allows the designer to hide the coordinate system from the representation of the geometric objects. Figure (6.2b) summarizes the association between the above classes as described above. The essential functions of GVector and GPoint classes are summarized in Table (6.1).

Abstraction of Node *Class*

As we have discussed earlier, the encapsulation of the behaviors of the degrees of freedom, constraints, loads, elements, elements' connectivity, and geometric manipulations is handled separately from the definition of the class Node. The Node class is implemented to

Table 6.2: Protocol for Node class

CLASS	OPERATION	ARGUMENTS	EFFECT
Node	Location	-	returns GPoint
	DeformedLocation	-	returns the deformed location

describe only the location of the node, which is represented by an associated geometric point GPoint. Each Node object contains a GPoint object and when queried for its location returns the associated GPoint. However, a Node object does not have any detailed knowledge as to how the location, GPoint, is represented and is manipulated internally. The main functions of Node class are summarized in Table (6.2).

6.1.2 Degrees of Freedom

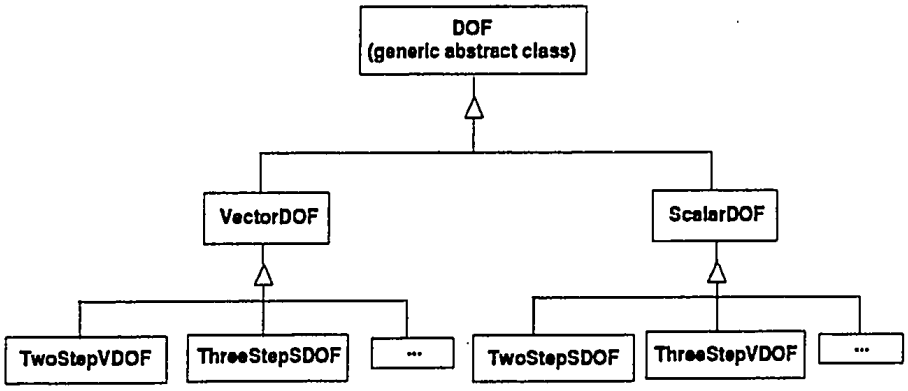


Figure 6.3: Inheritance diagram for degrees of freedom related classes

As mentioned in the previous subsection, degrees of freedom are encapsulated in their own class to maintain the advantages of coordinate-free representations. In particular, there are two basic classes, namely ScalarDOF and VectorDOF. The ScalarDOF encapsulates

the scalar degrees of freedom entities such as temperature and pore-pressure, while `Vector-DOF` encapsulates the vector degrees of freedom such as displacements and rotations, and a base class `DOF` encapsulates the common data and methods of these classes. On the other hand, the depth of state information required by various numerical schemes depends on the numerical scheme. A new level of degree of freedom classes such as, `TwoStepVDOF` and `TwoStepSDOF` are defined as depicted in Figure (6.3), encapsulating the representation of the state(s) pertaining to themselves.

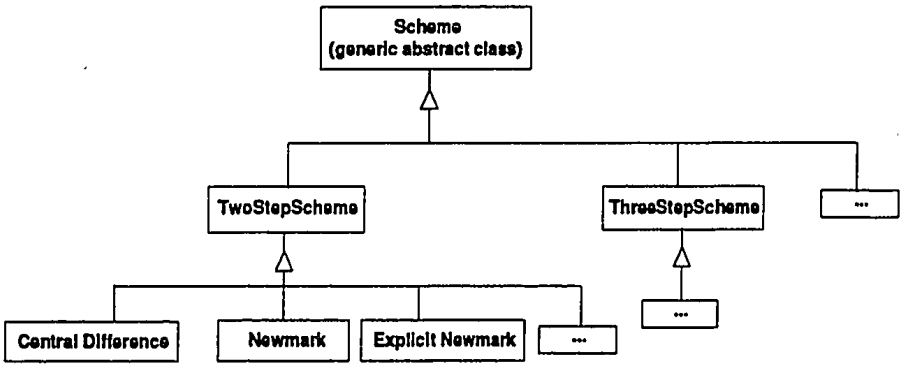


Figure 6.4: Inheritance diagram for `Scheme` class

The principal responsibility of the `DOF` class is to implement the details of the solution algorithms, which can be viewed as methods for determining the response of a set of degrees of freedom to a non-equilibrium set of generalized forces. In the case of dynamic non-linear problems there are a large number of available solution algorithms, but to a large degree these various algorithms share a common procedural framework. Broadly speaking, each algorithm involves computing information explicitly or implicitly at each time step in terms of the information known from the previous time step(s), then recursively moving forward in time. The strategy used here has been to capture the essential outline of this algorithmic procedure in a distinct high level class `Scheme`, while implementing the varying details in the derived classes. The interface to this functionality has been setup to allow for sub-

iteration within the overall time-stepping recursion, since this is usually necessary for non-linear problems. The framework of this interface is represented by the following primary member functions:`PreStep`; `SubStep`; `FinalSubStep`; and `PostStep`. `PreStep` and `PostStep` are called at the beginning and end of each time step, while `SubStep` and `FinalSubStep` form the internal iterative updates necessary for nonlinear problems.

Table 6.3: Protocol for abstract classes defining degrees of freedom

CLASS	OPERATION	ARGUMENTS	EFFECT
DOF	SetScheme	Scheme	select the numerical algorithm
	GetScheme	-	returns the pointer to the chosen numerical scheme
Scheme	PreStep	-	calculates the unbalanced generalized forces
	SubStep	-	calculates the next guess
	FinalSubStep	-	calculates the final guess
	PostStep	-	updates the appropriate variables in the DOF class

The solution algorithms can be broadly grouped together reflecting the number of steps used in the recurrence scheme. Within such a group, patterns of calculations are similar but the multipliers are different. Abstract classes derived from `Scheme` such as, `TwoStepScheme`, and `ThreeStepScheme`, encapsulate the recurrence equations, while another level of derived classes implement the actual numerical scheme. An inheritance diagram depicting these classes is shown in Figure (6.4).

By encapsulating each algorithm in its own subclass of the generic `Scheme` class it is possible to switch solution algorithms easily, and if desirable to switch algorithms at run-time by means of virtual functions. It is also possible to use heterogeneous algorithms, i.e.,

within a single analysis it would be fairly easy to have different algorithmic details at different nodes or for different types of degrees of freedom. The basic functions defining the protocol for abstract DOF and Scheme classes are summarized in Table (6.3).

6.1.3 Elements

Table 6.4: Protocol for abstract DOFConnector class

CLASS	OPERATION	ARGUMENTS	EFFECT
DOFConnector	Connect	DOFConnector*	connects it to the specified DOFConnector
	Disconnect	DOFConnector*	disconnects it from the specified DOFConnector
	ConnectMe	Node*	connects it to the specified Node
	SetElement	Element*	associates an Element
	GetElement	-	returns the associated Element*
	GetNode	-	returns the associated Node*
	SetDOFList	DOFList*	assigns a DOFList*
	GetDOFList	-	returns the associated DOFList*
	GetDOF	DOFType	returns the associated DOF*
	AddDOF	DOF*, DOFType	adds the DOF* with DOFType
	GetConnections	-	returns a list of DOFConnectors

The heart of the physics of any finite element analysis or boundary element analysis lies in the definition of elements. The primary role of an element is viewed as providing stiffness matrices of the element to compute the generalized forces from the generalized displacement components as will be discussed later in this subsection. The geometric details required to perform these computations can be accessed by maintaining the topological details of each element. As Nodes have knowledge of geometry, this could be achieved by keeping track of the nodes to which an element is connected. Instead, a list of DOFConnector

instances are stored to represent the element connectivity. The DOFConnector class is defined to allow discontinuities between two Elements connected to a particular Node enclosing the functionality of such connections.

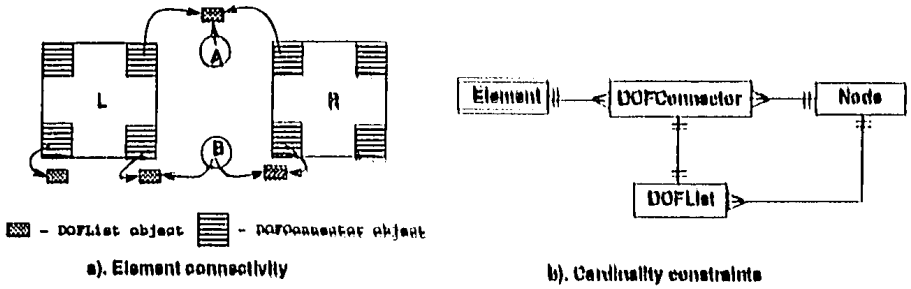


Figure 6.5: Element Connectivity and DOFConnector class

The design of the DOFConnector is such that each vertex of the Element is associated with a unique instance of DOFConnector class. The connections between two elements such as, L and R, as depicted in Figure (6.5a) are accommodated by connecting the relevant DOFConnectors to the appropriate nodes. The definition of DOFConnector includes the list of appropriate DOF instances for a specific element. This is implemented through a DOFList class which is essentially a list of elements comprising the pointers to DOF instances and their types such as, Translations, Rotations, Temperature, Pressure, defined by DOFType. In case of rigid connections the same DOFList instance can be shared between the connected connectors as well as the node. For flexible connections, separate DOFList instances need to be maintained in each connector and a list of DOFList instances are maintained in the Node as suggested in the Figure (6.5a). Figure (6.5b), showing the cardinality constraints among the instances of classes Element, DOFConnector, Node, and DOFList, also suggests the flow of information among these classes. Table (6.4) summarizes the essential functions of DOFConnector class.

The conceptual role of representing the mechanics of an element is largely influenced by

the assumptions about linearity with respect to deformation kinematics and material behavior, as well as the overall solution environment. For the case of evolving earth structures, it is preferable to make no restrictive assumptions about linearity with respect to deformation kinematics or material behavior. For traditional direct solution algorithms, the role of an element is to supply an element stiffness matrix. For iterative algorithms, however, there are other ways to view the elements' role. In particular, for the case of element-by element

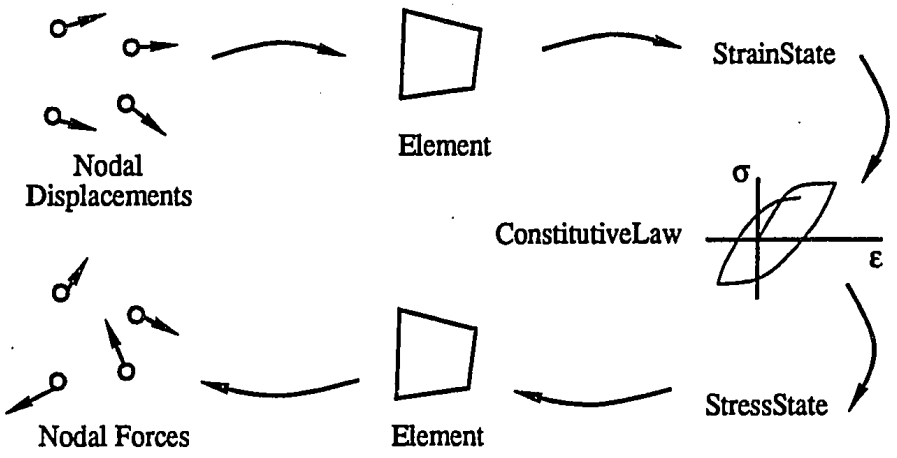


Figure 6.6: Nodal displacement to nodal force computation (Miller et al., 1995)

solution strategies (Couthino et al. 1991), the main computational responsibility of an element can be viewed as constructing a set of generalized internal nodal forces consistent with a given set of generalized nodal displacements, rather than constructing a stiffness matrix. Figure (6.6) shows the basic process by which nodal displacements are converted into a corresponding set of nodal forces. This process can be viewed in two main parts, i) geometric transformations of displacements to strains and stresses to nodal forces; and ii) a transformation from strains to stresses. The geometric transformations are deemed to be the work of the element itself, while the strain-stress transformation is considered to be the work of the material the element is made of. For this reason, the strain-stress transformation computa-

tions are separated from the definition of the `Element` class and are discussed in the subsequent subsection on materials. In this treatment an element uses its associated `material` to carry out the strain-stress conversion making it straightforward to alter an individual element's material interactively. It also provides a useful division of labor, where the class `Element` handles only the geometric transformations. An abstract `Element` class encapsulates this functionality as virtual functions and the actual mechanics for various element types are implemented in the subclasses.

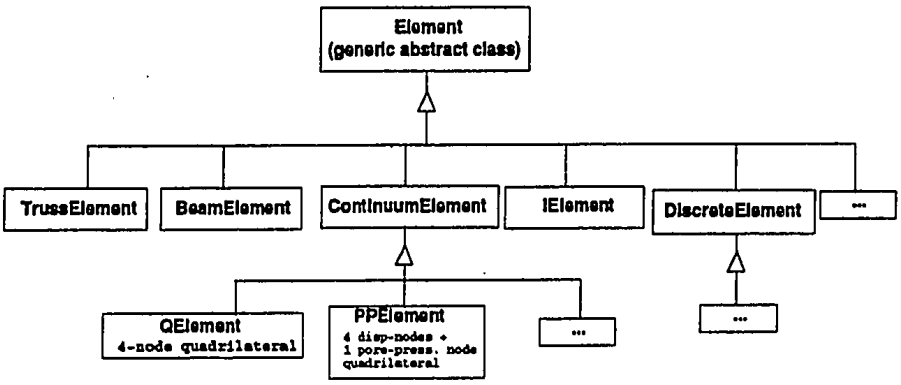


Figure 6.7: Inheritance diagram for `Element` classes

Table 6.5: Protocol for abstract `Element`

CLASS	OPERATION	ARGUMENTS	EFFECT
`Element`	DisplToStrain	-	returns `StrainState`
	StressToForce	`StressState`	calculates and sets the nodal forces
	UpdateInternalForces	`UpdateChoice`	stress and strain values are updated, when requested

A class structure encapsulating the different elements pertaining to this study is shown in

Figure (6.7). The basic functions defining the protocol for the abstract `Element` are summarized in the following Table (6.5). The 4–node quadratic continuum element (`QElement`) and 4–node, constant pore-pressure porous element (`PPElement`) are derived from the `ContinuumElement` class derived from the `Element` enclosing the properties that pertain only to the continuum type elements. Similarly, properties corresponding to discrete elements are defined in the `DiscreteElement` class. In order to handle soil-structure interaction problems, `TrussElement`, `BeamElement`, and interface element (`IElement`) are defined and implemented in this study.

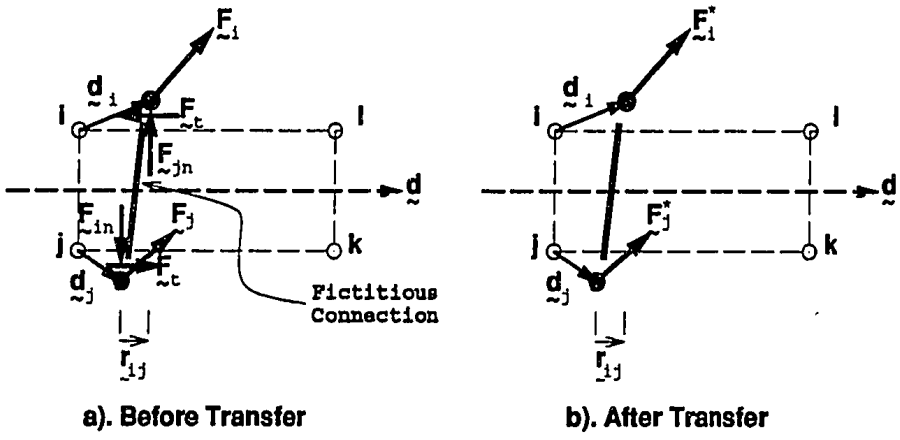


Figure 6.8: Data Transfer between two Elements through an interface element

The treatment of the interface elements in this study is different from traditional finite element technique in that an instance of `IElement` is fictitious. In traditional finite element analysis, interface elements are treated as elements with one negligible, but finite dimension consisting of material properties simulating the interface behavior. As is to be expected, the magnitude of this dimension has an impact on the accuracy and stability of the solution. In contrast, the interface element (`IElement`) encapsulates the interface behavior without actually introducing a finite thickness element. This is possible due to the solution scheme and the strategy of using `DOFConnectors` to model the connections; this enables keeping

track of discontinuities and transferring internal and external forces between the elements appropriately.

6.1.4 Materials

As indicated in the previous subsection, the primary role of the `Material` class is to convert a set of strains into a set of stresses according to its constitutive law. For most geomaterials, the material behavior computations require the knowledge of internal state(s) reflecting the history of the loading. The data related to behavior and state of the material are encapsulated separately by defining distinct internal classes called `MaterialState` and `ConstitutiveLaw`, as described in the subsequent paragraphs. In fact the state information defined by `MaterialState` are more suitably associated with `Element` class to maintain its state information allowing the `Material` instances be shared by several `Element` instances. Each element has an associated `MaterialState` and has knowledge of its associated `Material`, while each `Material` has information on the `ConstitutiveLaw` it is associated with. Receiving the current state information stored in `MaterialState` and the new strain information from `Material`, the `ConstitutiveLaw` class computes the new values of stress.

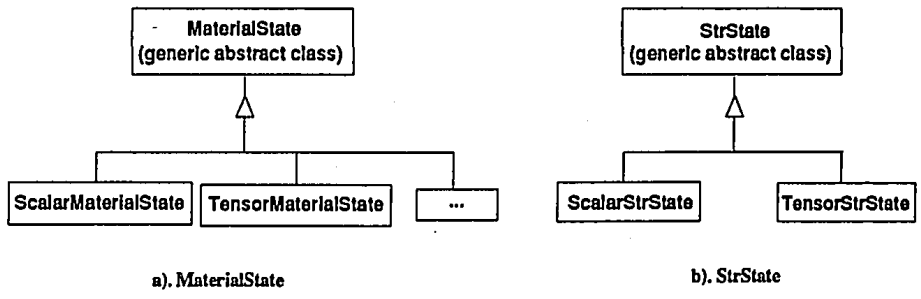


Figure 6.9: Inheritance diagrams for `StrState` and `MaterialState` classes

The representation of state information, such as values of stress, values of strain, and

values of pore-pressure, is the primary responsibility delegated to the `MaterialState` classes. The classes `ScalarStrState` and `TensorStrState`, derived from the abstract base class `StrState`, encapsulate the `Scalar` and `Tensor` state information respectively. Reflecting this argument, the classes `ScalarMaterialState` and `TensorMaterialState` are derived from the abstract base class `MaterialState` to contain the state information defined in `ScalarStrState` and `TensorStrState` respectively. Inheritance diagrams for the above two Sets of classes are shown in Figure (6.9).

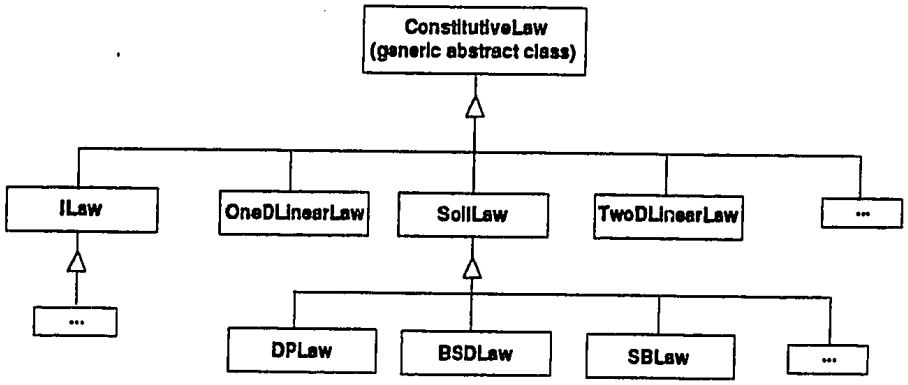


Figure 6.10: Inheritance diagram for `ConstitutiveLaw` class

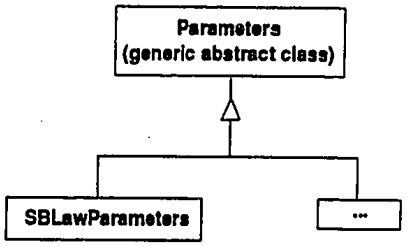


Figure 6.11: Inheritance diagram for `Parameters` class

The encapsulation of behavior of the material is the primary responsibility of the new

ConstitutionalLaw class hierarchy. This class structure organizes various constitutive laws of the material providing functionality to perform stress-strain transformations given its state. Figure (6.10) shows an inheritance diagram depicting the class structure used in this study. The abstract ConstitutionalLaw class defines the methods to perform stress-strain transformations as virtual functions and the subclasses, such as TwoDLinearLaw and OneDLinearLaw, implement these functionalities. In defining soil laws, an intermediate class SoilLaw is defined enclosing the constants and parameters that are particular to soils such as void-ratio, drainage conditions, and stress path. The constitutive models for soils such as Drucker-Prager law (DPLaw), BSDLaw (Banerjee et al. 1992), and SBLaw (Sribalaskandarajah & Banerjee 1996) are defined as a derived classes of SoilLaw implementing the material behavior.

Table 6.6: Protocol for abstract classes defining the material behavior

CLASS	OPERATION	ARGUMENTS	EFFECT
ConstitutiveLaw	StrainToStress	MaterialState, StressState	returns StrainState
	StressToStrain	MaterialState, StressState	returns StressState
MaterialState	CurrentStrain	-	returns StrainState
	CurrentStress	-	returns StressState
	UpdateState	StrainState, StressState	updates the stress and strain values
Material	SetLaw	ConstitutiveLaw*	defines the Material behavior
	GetLaw	-	returns ConstitutiveLaw*

The parameters of these models are mostly constant; however, some of these parameters evolve with the state information. It is efficient to include the constant model parameters in the definition of the constitutive laws, while the evolving parameters are encapsulated in a separate class structure beneath an abstract class`Parameters`. As these parameters are evolving and dependent on the state information, each material state requires an associated `Parameters` instance. The soil law `SBLaw` has such evolving model parameters, which are defined in a derived class `SBLawParameters`. Therefore, two different treatments of model parameters are suggested in this study. First, the constant parameters can be implemented in the constitutional law's definition itself, while the evolving parameters are encapsulated in a separate class structure beneath an abstract class`Parameters`. The soil law `SBLaw` has such evolving model parameters, which are defined in a derived class `SBLawParameters` (Figure (6.11)). As in general the model parameters are functions of the state parameters, they are included in the definition of the `MaterialState` class. The data abstraction outlined above allows for modeling with arbitrary constitutive behavior for different elements. The essential functions in the classes defined in this subsection are summarized in the Table (6.6).

6.1.5 Loads and Constraints

The treatment of loads and constraints in a finite element analysis could be viewed as the external influences on degrees of freedom associated with a particular node. The differing aspects of these influences are encapsulated in abstract classes `TimeDependentLoad` and `Constraint`, and their subclasses.

The abstract base class `TimeDependentLoad` primarily encapsulates the time dependent behavior of load allowing a general treatment of arbitrary loading conditions. The abstract class is responsible for keeping track of the magnitude and direction of the load at each point in time, as well as the degrees of freedom to which it is applied. Depending on the type of load, this force updating might be determined by direct calculation, or by reading and interpolating data from an input file, such as an earthquake record. These specific type

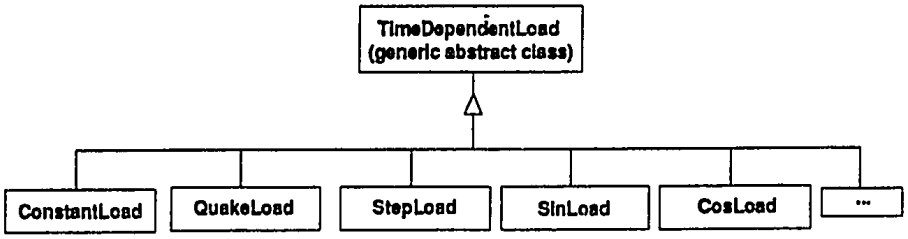


Figure 6.12: Inheritance diagram for `TimeDependentLoad` class

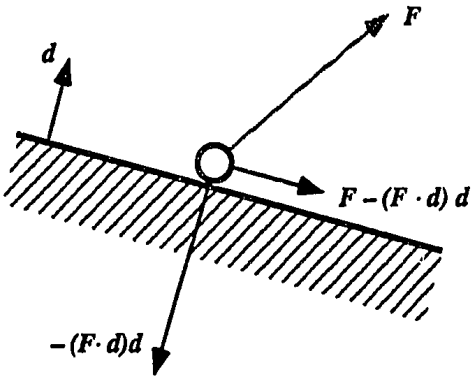


Figure 6.13: Force filtering by a constraint (Miller et al., 1995)

of loads such as, constant loads, step loads, quake loads, etc. are defined as subclasses of the abstract `TimeDependentLoad` class as shown in Figure (6.12). The specifics related to the actual loads themselves are encapsulated in these classes, while the primary functionality is encapsulated in `TimeDependentLoad` class as summarized in Table (6.7).

The primary responsibility of the `Constraint` class is to encapsulate the force and displacement boundary conditions, except the external loads represented by `TimeDependentLoad` class. The abstract base class defines functions to filter the externally applied forces, internal forces and displacements associated with the degrees of freedom at the corresponding node consistent with a specified constraint. Directional constraints operate by

Table 6.7: Protocol for abstract `Constraint` and `TimeDependentLoad`

CLASS	OPERATION	ARGUMENTS	EFFECT
`TimeDependentLoad`	Update	time	updates itself
`Constraint`	FilterAppliedLoad	-	filters the applied forces
	FilterForce	-	filters the internal forces
	FilterDisplacement	-	filters the displacements

applying the necessary filters in the constrained directions as shown in the example of a planar constraint in Figure (6.13). Thus, the node experiences the appropriate kinematic restraint, while the corresponding reaction associated with the constraint is always available as the net force absorbed by the `Constraint`. Constraint directions and applied loads are expressed in terms of coordinate-free vector quantities and the resulting generalized forces satisfy the dynamic equilibrium conditions of the particular degrees of freedom. This type of force-based constraint handling is similar to a technique used in physically-based computer graphics models of rigid and deformable media (Barzel & Barr 1988).

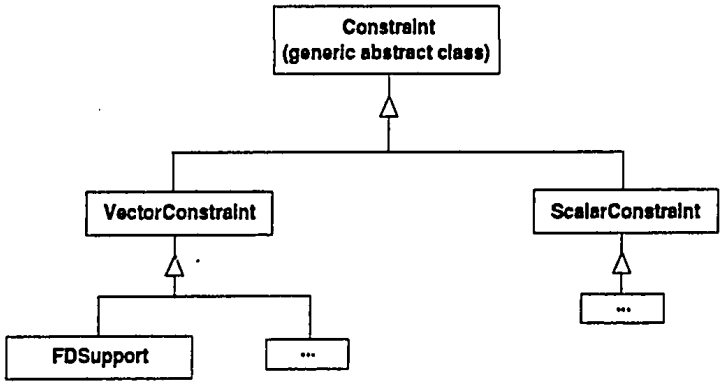


Figure 6.14: Inheritance diagram for `Constraint` class

Reflecting the type of degrees of freedom, constraints can be grouped into two major subclasses, namely `ScalarConstraint` and `VectorConstraint` encapsulating the scalar and vector data, respectively. Different types of constraints are implemented beneath these classes implementing the virtual functions defined in the `Constraint` class. In this study, `FDSupport` class encapsulates several such displacement and force constraints. Figure (6.14) shows the inheritance diagram for the above classes and the main functions are summarized in Table (6.7).

6.1.6 Data Recorders

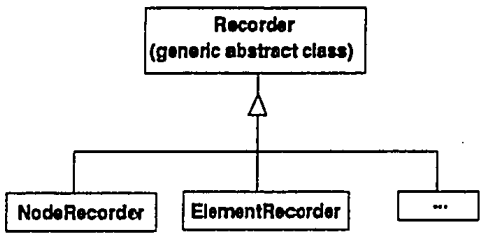


Figure 6.15: Inheritance diagram for `Recorder` class

In a dynamic analysis of a finite element model, only a small portion of the vast amount of numerical data is of any practical use. Monitoring the particular behavior of a selected object is the primary role of an abstract `Recorder` class. Conceptually, recorders can be attached to selected objects such as nodes and elements to monitor the behavior of interest corresponding to the attached object. As these behaviors are dependent on the type of object, different classes such as `NodeRecorder`, and `ElementRecorder` are derived as in Figure (6.15), enclosing the functionality to record relevant physical quantities. Presentation of the recorded results is included in the definition of the graphical representation of `Recorder` described in the next section.

6.1.7 *Assembly of the Analysis System*

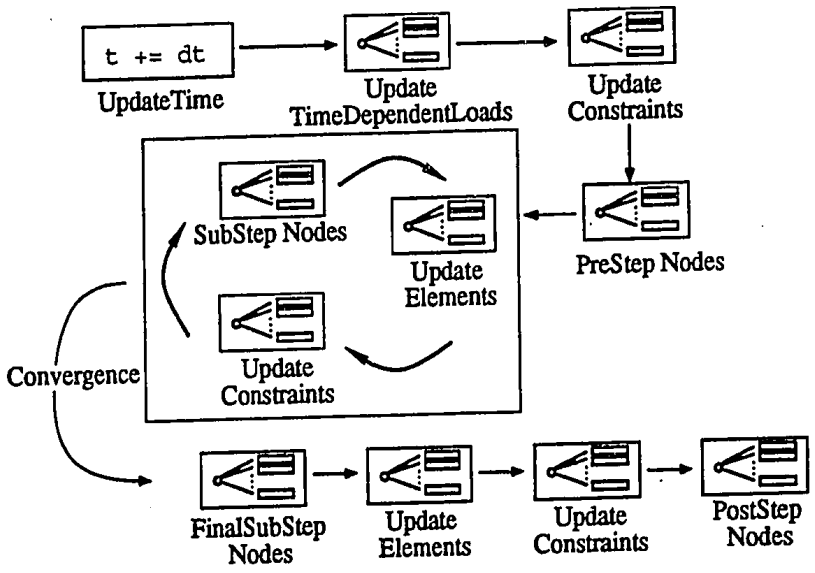


Figure 6.16: Time-step process for an Assemblage (Miller et al., 1995)

Some essential classes conceptualizing the finite element analysis have so far been presented in this section. Assembly of these conceptual classes to produce a coherent analysis system and efficient creation, manipulation, and management of the instances of these various classes are discussed in this subsection. These tasks are quite exhaustive, and elaborate definitions of data types encapsulating different management issues and analysis situations may be necessary to produce an efficient general system. However, for simplicity these functionalities are enclosed here in the `Assemblage` class, assembling various modeling and analysis classes in this study.

The principal function of the `Assemblage` class, besides the management functions, is to encapsulate the sequence of computations involved in the time history modeling of a structure. This is accomplished by means of a basic time-stepping function, `DoStep`, as outlined in Figure (6.16). This function is intended to provide a generic implementation

Table 6.8: Protocol for abstract `Assemblage`

CLASS	OPERATION	ARGUMENTS	EFFECT
`Assemblage`	AddNode	`Node`	Node management
	RemoveNode	`Node`	
	NumberofNodes	-	
	AddElement	`Element`	`Element` management
	RemoveElement	`Element`	
	NumberofElements	-	
	AddConstraint	`Constraint`	`Constraint` management
	RemoveConstraint	`Constraint`	
	NumberofConstraints	-	
	AddConstitutiveLaw	`ConstitutiveLaw`	`ConstitutiveLaw` management
	RemoveConstitutiveLaw	`ConstitutiveLaw`	
	NumberofConstitutiveLaws	-	
	AddLoad	`TimeDependentLoad`	`TimeDependentLoad` management
	RemoveLoad	`TimeDependentLoad`	
	NumberofLoads	-	
	DoStep	Maximum Number of iterations	iterate until convergence

of the shared features of any element-by-element algorithm for integrating the equations of motion for general nonlinear problems. The main feature of this implementation is that it does not rely on any explicit, structured global data structure.

Basic management functions such as maintaining the records of the instances such as `Node`, `Element`, `Material`, `Constraint`, and `TimeDependentLoad` are handled in the definition of `Assemblage` class. This class is composed of collections of instances of these principal abstractions enclosed in a simple list data structure (`SimpleList`). The essential management functions pertaining to the records of each of these objects are de-

fined with the use of access functions to the `SimpleList` data structure as summarized in Table (6.8). The use of these management functions such as creation, deletion, modification of the above objects are handled efficiently through graphical user interfaces. This necessitates that these objects are graphically represented in the analysis system. Classes defining such graphical representations are presented in the following section.

6.2 Classes for Graphic Representations

The previous section presents the details of the development and refinement of some essential classes encapsulating the numerical analysis. The primary aim of this section is to present the details of the design of classes representing analytical objects graphically. The graphical representation classes are efficiently designed by using the classes designed in the above section and extending them to include the graphical aspects inherited from graphical user interface libraries.

Interactive modeling of numerical analysis problems is possible by defining graphical objects such as `NodeRep`, `ElementRep`, `ConstraintRep`, `RecorderRep`, and `LoadRep` representing the corresponding real world analysis objects such as , `Node`, `Element`, `Constraint`, `Recorder`, and `TimeDependentLoad` defined in the above section. Actions such as creation, deletion, modification, presentation, and management of these analysis objects can be invoked by performing similar operations on their graphical representations respectively. Graphics libraries, such as `Unidraw`, provide graphical objects and functions for basic graphical manipulations. The functionalities of the analysis object and the corresponding graphics object can be enclosed in the graphical representations either by multiple inheritance or by polymorphism. This latter mechanism is used in this study and the corresponding inheritance diagram is shown in Figure (6.17).

Interactive modeling, analysis, and presentation is feasible by displaying particular views of the graphical representations of the analysis objects and by manipulating those representations. A primary definition of such a view involves projection of the graphical object to a

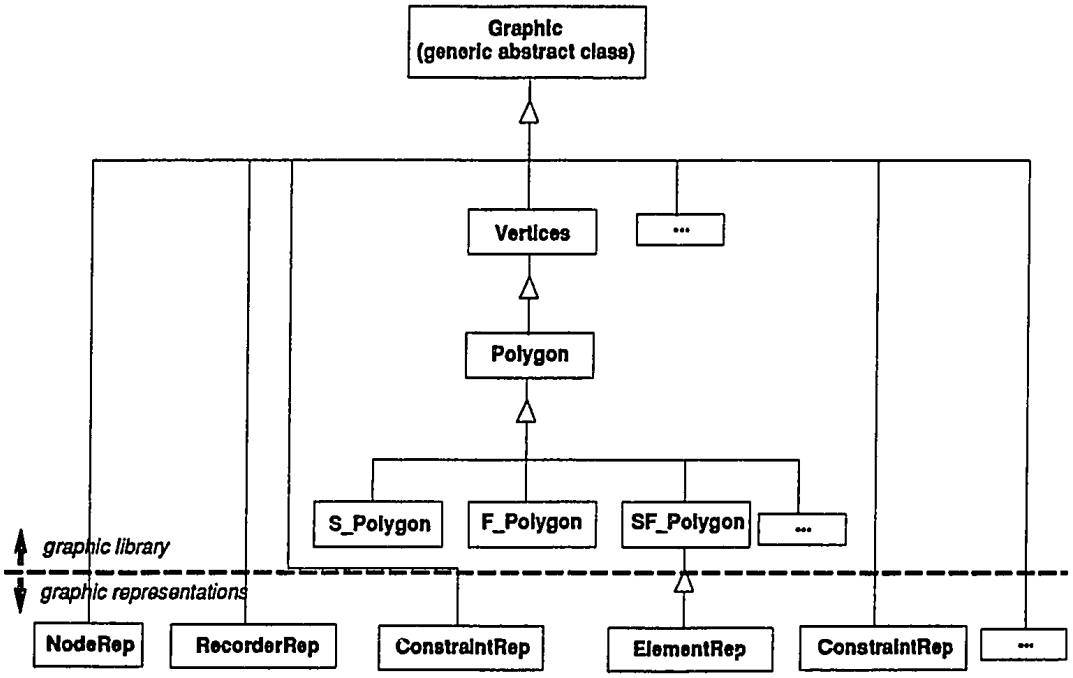


Figure 6.17: Inheritance diagram for graphic representation classes

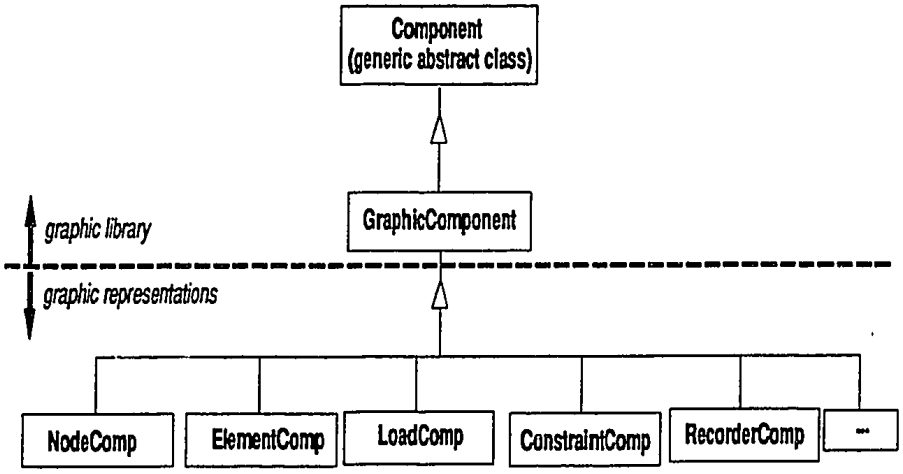
two dimensional screen space which is uniquely represented by an origin, a local coordinate system defined by a `GFrame`, scale, and deformation scale. This concept of view and the associated data are included in the definition of class `View`. As the class `Node` contains the geometry related data, its representation, `NodeRep`, will also depend on the geometry details. Therefore, of all the graphical representation objects, only `NodeRep` will include the details of the projection by including `View` in its definition.

In developing software using the `Unidraw` architecture, the primary goal of data abstraction is to design a context based graphics editor, consisting primarily of the graphical representations mentioned above. These components represent the real world behavior of the objects of interest in such an editing domain. The architecture of the editing domain consists of: i) components; ii) tools; iii) commands; and iv) external representations. Components encapsulate the appearance and the semantics of the real world analysis objects and

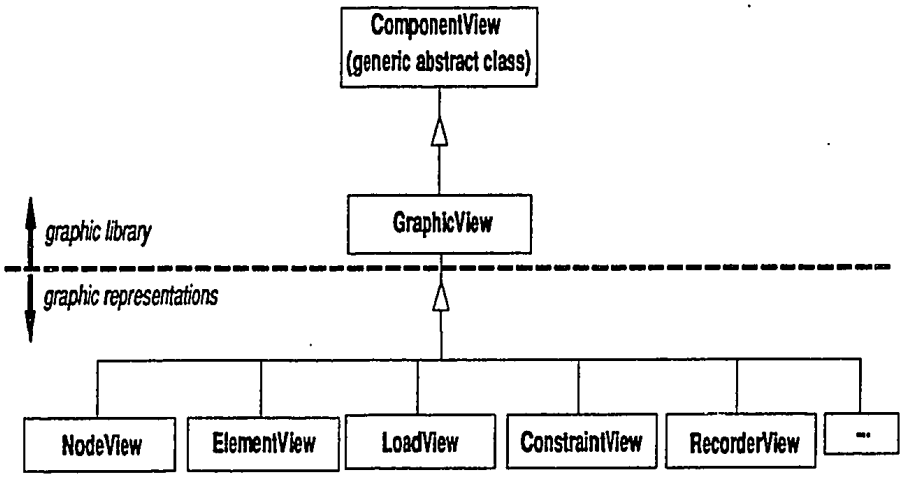
represent them in the domain-specific editor (SandEditor). The role of this domain specific editor is to allow the end-user to arrange these components to convey information regarding the modeling in the domain of interest. These components can maintain one or more graphical representations as well as non-graphical representations. Components also define their responses to manipulations by tools and commands.

In a well defined user interface such as Unidraw, the distinction is made between the subject and view of the graphics components. A subject encapsulates the context-independent operations of a component, while a view allows the context-dependent presentation of the subject. That is, a component subject may have any number of component views, each presenting a different representation of and interface to the subject. The Unidraw library provides the definitions for base classes of these components and the implementations for the graphic objects provided in the library. For each additional graphic representation introduced, the user supplies the corresponding subject and view protocols by defining newer classes as summarized in Figure (6.18).

Manipulations of the above graphical objects in the graphical object editor, SandEditor, could be handled with commands or tools protocol. Tools allow direct manipulation of the graphical objects. Tools such as NodeCompTool, ElementCompTool, RestrainTool, RecorderTool, MaterialTool, CLawTool, XSectionTool, LoadTool, ViewTool, DOFConnectorTool are defined in this study in order to manipulate the respective graphical representations and thereby the associated analysis object. For example, a left mouse click on the drawing area after engaging the NodeCompTool creates a pair of instances of NodeComp and Node classes. On the other hand, commands are analogous to messages as they can be interpreted by graphic components. Also, the command data structure maintains enough information to execute one operation and, if that operation is reversible, it also maintains enough information to reverse the effects of such an operation. Commands can be made directly available to the user through menus. A detailed discussion of the Unidraw architecture can be found in the technical report written by Vlissides (1990).



(a) Subject



(b) View

Figure 6.18: Inheritance diagrams for subjects and views of graphic components

Chapter 7

ILLUSTRATIVE EXAMPLES

The details of the design of the proposed computational framework and its implementation for dynamic soil-structure interaction problems have been presented in the previous chapter. A prototype finite element program (SAND) based on the design and the concepts outlined in the earlier chapters has been developed during the study. In order to illustrate the uses and accuracy of the prototype, three simple soil-structure interaction problems are analyzed and the results are presented in the following three sections of this chapter. In the first section an analysis of an uplift problem involving steel frames subjected to earthquake loading is presented. The next section presents a narrowly scoped example involving interface elements. In the third section an analysis of a tunnel problem using both a linear elastic material model and the elasto-plastic model proposed in Chapter 3 is presented. The final section summarizes the significance of the proposed computational work.

7.1 Uplift of steel frames

In standard dynamic analysis of structures the foundation of the structure is generally assumed to be anchored rigidly to the ground. However. in most actual situations the foundations are not firmly attached to the ground. When the overturning moment exceeds the overturning resistance due to gravity such a structure will experience uplift. In general, analysis of this problem requires a model with foundation-soil interface elements allowing discontinuities between columns and foundations. Various researchers (e.g. Priestley et al. 1992, Smith 1995) have modeled this foundation-soil discontinuities with non-linear spring/truss elements successfully. Results from such an analysis obtained using the non-

linear finite element program DRAIN-2DX (Prakash et al. 1993) are presented in this section along with the results from this study for comparison purposes.

Figure (7.1) shows a concentrically braced steel frame with all connections pinned to model shear connections. As noted above, the foundation-soil interface behavior is modeled with non-linear spring type elements with load-displacement behavior given by Figure (7.2a), while the structural elements are modeled with linear elastic truss elements. In DRAIN-2DX, non-linear foundation behavior can be modeled with non-linear truss elements with behavior depicted in Figure (7.2b). This is acomplished by letting the structure hang from the foundation elements, as opposed to letting the structure rest on the foundation elements, with zero compressive stiffness and large tensile yield strength. The inelastic behavior of the foundation elements can also be modeled successfully with the inelastic cable elements provided in the DRAIN-2DX element library. In SAND, however, the foundation elements are modeled with the `FDSupport` class described in the previous chapter. This has been made possible by including the non-linear spring type behavior of the foundation elements in the definition of `FDSupport` class.

The applicability of this model hinges on the determination of the equivalent stiffness of the foundation elements representing the soil properties. A method to determine this equivalent stiffness was proposed by Gazetas (1991) for machine loaded and earthquake loaded foundations. Utilizing this method and assuming soft soil properties, an equivalent stiffness of the non-linear truss elements was found to be of 2,350 kip/in (Smith 1995). Zero soil damping is used for the analyses presented in this section.

Uniform cross-sections for structural members (W14x38) for beams and columns are assumed for simplicity. An equivalent density of the structural members is computed to correspond to the load distribution (dead load 50 psf; live load 50 psf), developed in this study does not include provisions for additional dead load and live load computations yet. The equivalent density is found to be 0.4017 kips-sec^2/ft^4 for the exterior frame considered for the analysis presented in this section. Damping of the structural elements is assumed to

be of the Rayleigh damping type given by

$$C = \alpha_c M + \beta_c K \tag{7.1}$$

where α_c and β_c are material constants. In the following analyses, the structure is subjected to an earthquake (Pacoima dam) loading shown in Figure (7.3) and the uplift displacements of the node **A** are observed. Values of α_c and β_c used in these analyses are 0.00 and 0.00163 respectively.

Unlike the present day analysis programs such as DRAIN2DX, the program SAND provides interactive modeling and visual feedback. In the program SAND, the analysis can be stopped interactively at any particular point in time and the results can be viewed and modifications of the model can be made as needed. Also, the program continually updates the deformed shape of the structure graphically, providing an effect of animation. The effect of animation can be improved by reducing the time interval between successive updates which primarily depends on the speed of the computation and the size of the problem. Alternate algorithmic schemes may be employed to improve the speed of the computation. Figure (7.4) shows a display of a deformed shape of the structure, with a deformation scale of 10.0, showing uplift at the node **A**. A `NodeRecorder` attached to node **A** monitors the displacement vector of this node. Uplift displacements of this node with and without iteration of recurrence equations within each time step is shown in Figure (7.5). In both situations the error correction scheme suggested in Chapter 5 is used and the results are not significantly different. The small differences observed may be attributed to different choices of algorithms and time steps in the respective analyses.

The finite element formulations used in SAND are in fact large displacement formulations, while DRAIN-2DX does not handle large displacement analysis (Prakash et al. 1993). Therefore, large displacement and $P-\delta$ effects are neglected in generating Figure (7.6) displaying the uplift displacements of the node **A** from both programs. The agreement between these two results is excellent. This verifies the consistency of the prototype finite element scheme in the computer program SAND with traditional tools.

7.2 *Interface Elements*

As discussed in the previous chapter, the implementation of interface elements (`IElement`) differs from traditional interface elements. In order to verify this implementation, two plane strain elements each subjected to different pressure loadings as shown in Figure (7.7a) are considered. The displacement pattern of the loading surface is calculated using the newly developed SAND program with and without interface element(s) connecting these two elements. In order to obtain better results and to obtain results at more points on the loading surface these elements are further subdivided as shown in Figure (7.7a). The interface behavior is defined in `ILaw` as a linear force-displacement behavior given by Figure (7.7b). The elastic properties of the plane strain elements are given by, G = 10 ksi and ν = 0.3. As the objective of this analysis is to study the interfacial behavior due to the differential loading on these elements the gravitational forces are turned off during the analysis. The analyses presented in this section were performed without interface, with smooth interface, and with an interface modulus of 200 psi. The results in these three cases are shown in Figure (7.7c). The results conform to the anticipated interfacial behavior. The deformed shape of the mesh with and without interface elements are shown in Figure (7.8).

7.3 *Surface settlement due to tunnel construction*

In this section, a surface settlement analysis of a twin tunnel system is chosen to demonstrate the applicability of the prototype finite element program developed to practical geomechanics problems. The layout of this twin tunnel is shown in Figure (7.9a). This twin tunnel system built to construct subway lines during the rapid urbanization of Nagoya city (Kawamoto & Okuzono 1977) is analyzed with different material models to expound the the implementation and use of different numerical models in the program. The analyses were done using both a linear elastic material model and the elasto-plastic model proposed in this study. For simplicity, uniform material properties are assumed in the analysis presented here.

Due to gravitational loads, the soil is subjected to an initial stress distribution. This dis-

tribution may be found by subjecting the structure to gravitational loads and then resetting the strains to zero values. For elastic and most of the plastic models this provides an accurate estimation of the initial stress distribution and for the rest this provides a satisfactory estimation. Alternatively, for problems of simple geometry the initial stress distributions may be estimated directly by assuming a value for the coefficient of lateral earth pressure, k_o. For elastic materials, the value of k_o can be derived as

$$k_o = \frac{\nu}{1-\nu} \tag{7.2}$$

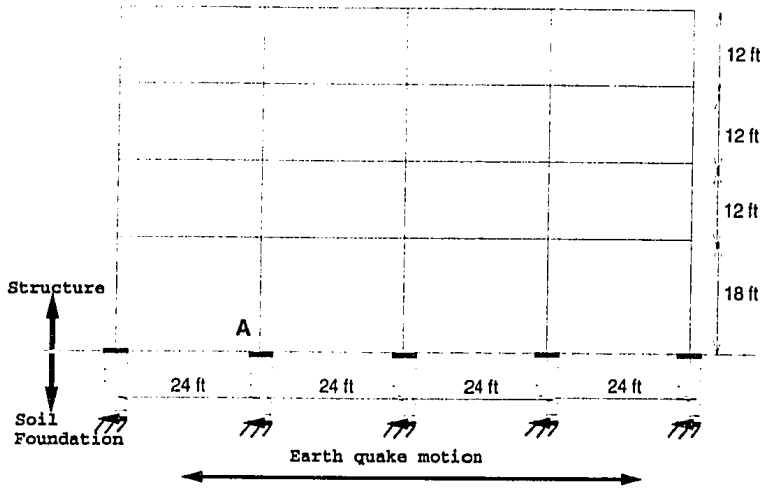
where ν is Poisson ratio of the soil.

Considering the symmetry of the model, the analysis is performed only on one tunnel as shown in Figure (7.9b). This figure displays the deformed shape of the finite element mesh with an elastic model(E=35.0 MPa, ν=0.33). The analysis was repeated with the elasto-plastic model proposed in this study with model parameters, p_o=100.0 kPa, G_o=13.125 MPa, K_o=35.0 MPa, M_c=1.35, M_p=1.5, α=.0515, β=1.0, ζ=1.0, e_o=0.7, and e_{min}=0.55. The value of k_o is assumed to be 0.5, as in the case of elastic model. The displacement profile of the ground surface obtained from both these analyses are satisfactory as shown in Figure (7.10). These results from the prototype program demonstrate the successful implementation of the computational framework and the elasto-plastic model developed in this study.

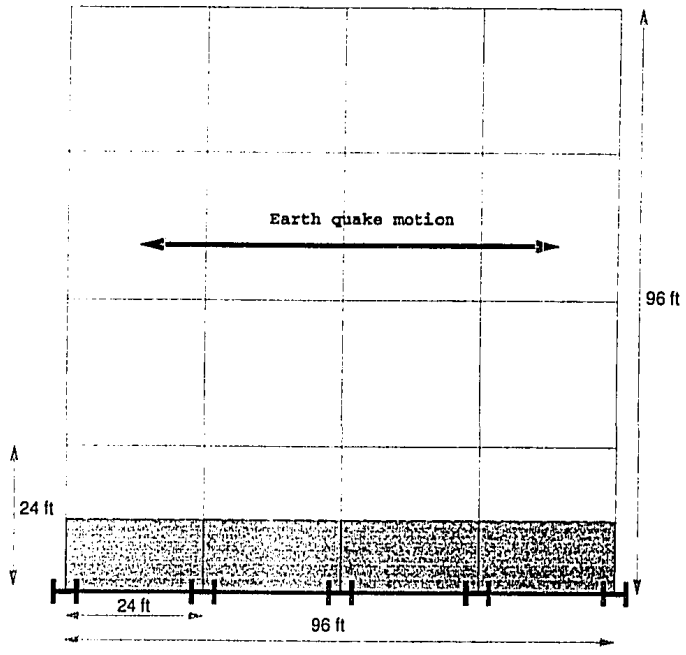
7.4 Summary

The prototype finite element program SAND developed based on the computational framework presented in the previous chapter has been tested in the above examples. The accuracy of the numerical schemes implemented in the program have been verified. Interactive modeling and visual feedback of the structure during the analysis have also been demonstrated. In addition, the use of interface elements developed as a consequence of the design of the computational framework has been illustrated. The elasto-plastic model developed in this study has been implemented in the program SAND and its use in practical soil problems has

been demonstrated with the tunnel example. Problems with initial stress and complicated interface behaviors can be analyzed efficiently without losing accuracy.

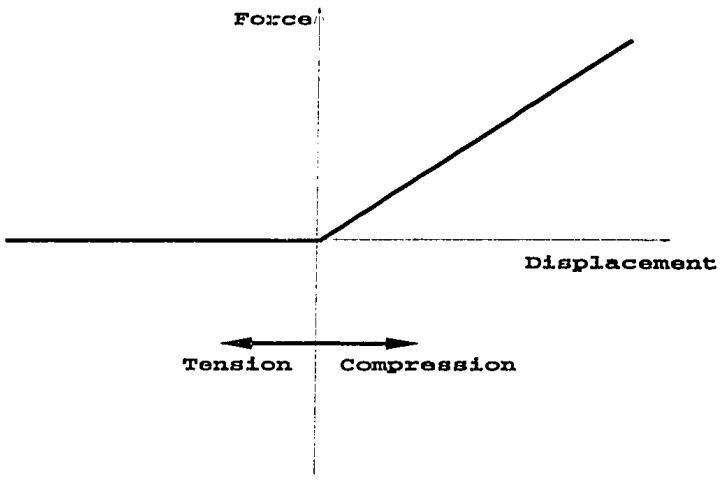


(a) Elevation view

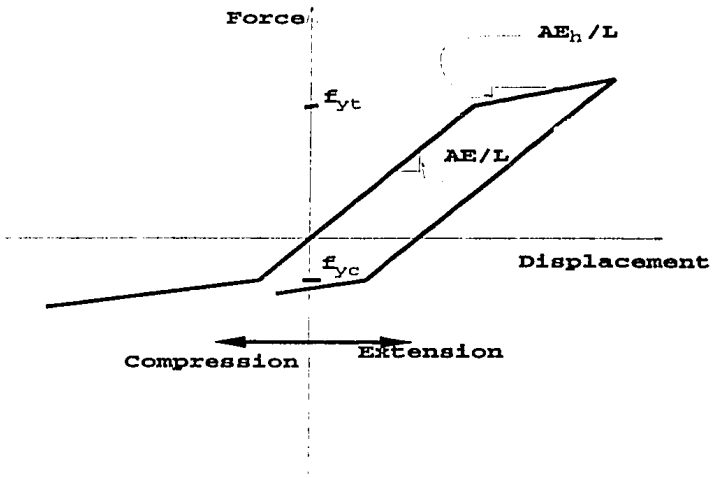


(b) Plan view

Figure 7.1: Structural model for the uplift problem



(a) Spring support



(b) Truss element (Prakash et al., 1993)

Figure 7.2: Force-displacement behavior of ground support

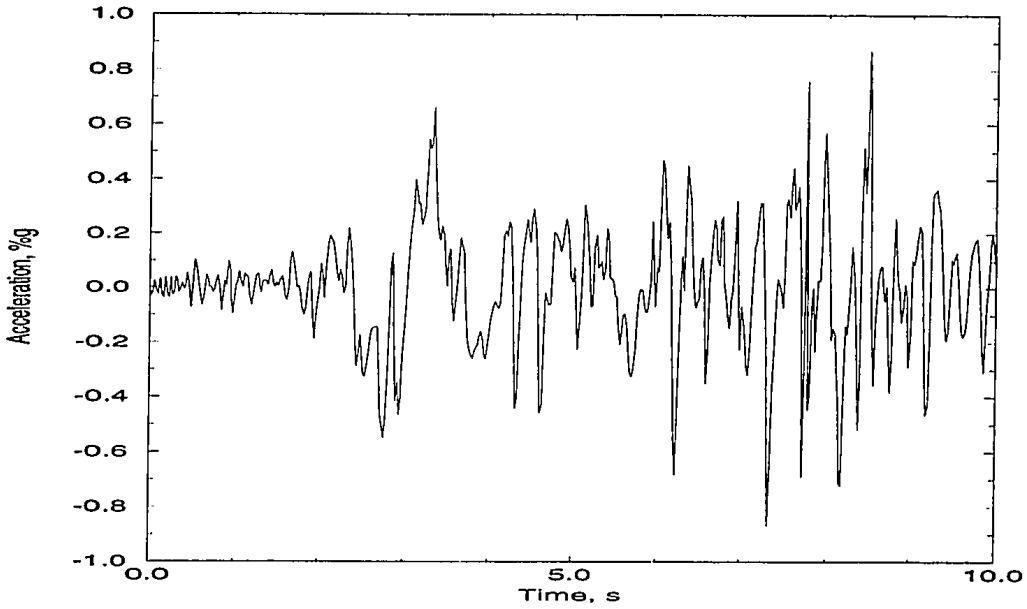


Figure 7.3: Acceleration time history, Pacoima dam

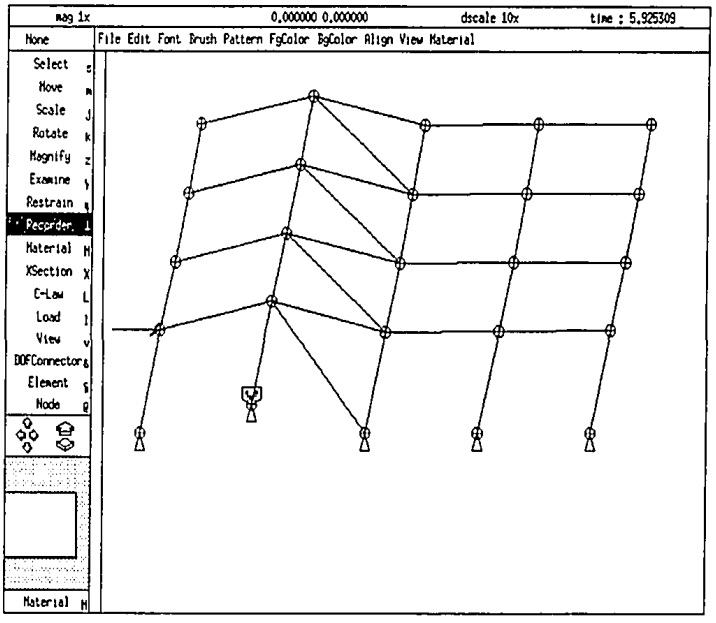


Figure 7.4: Deformed structure showing uplift of the column

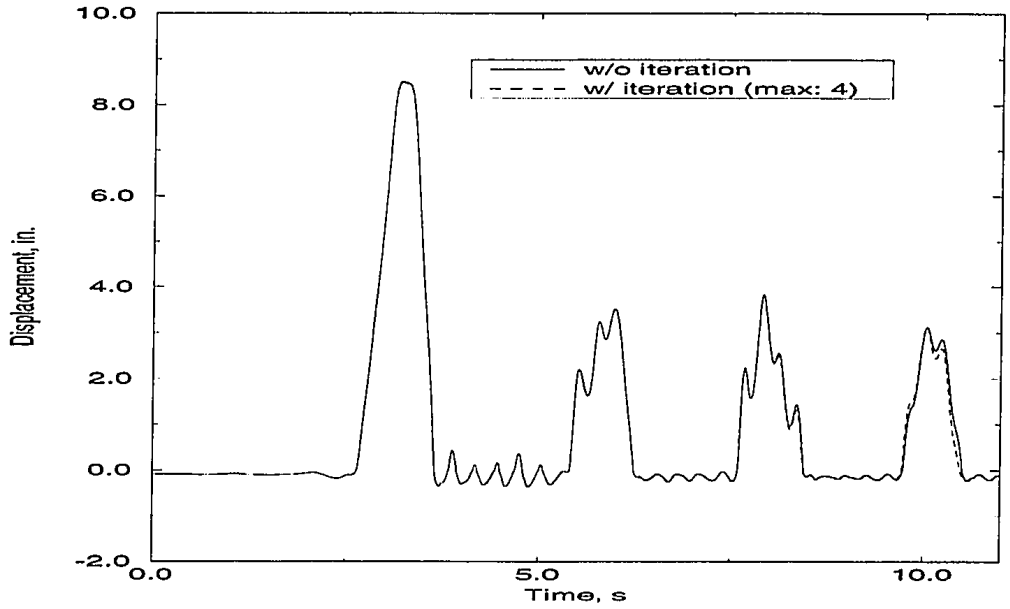


Figure 7.5: Comparision of uplift results with and without iteration

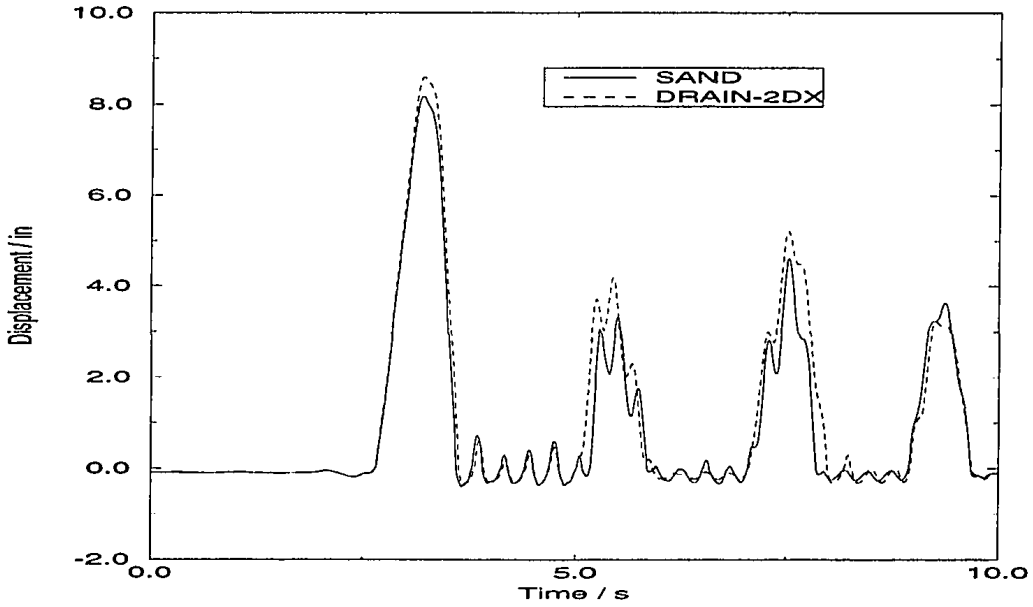
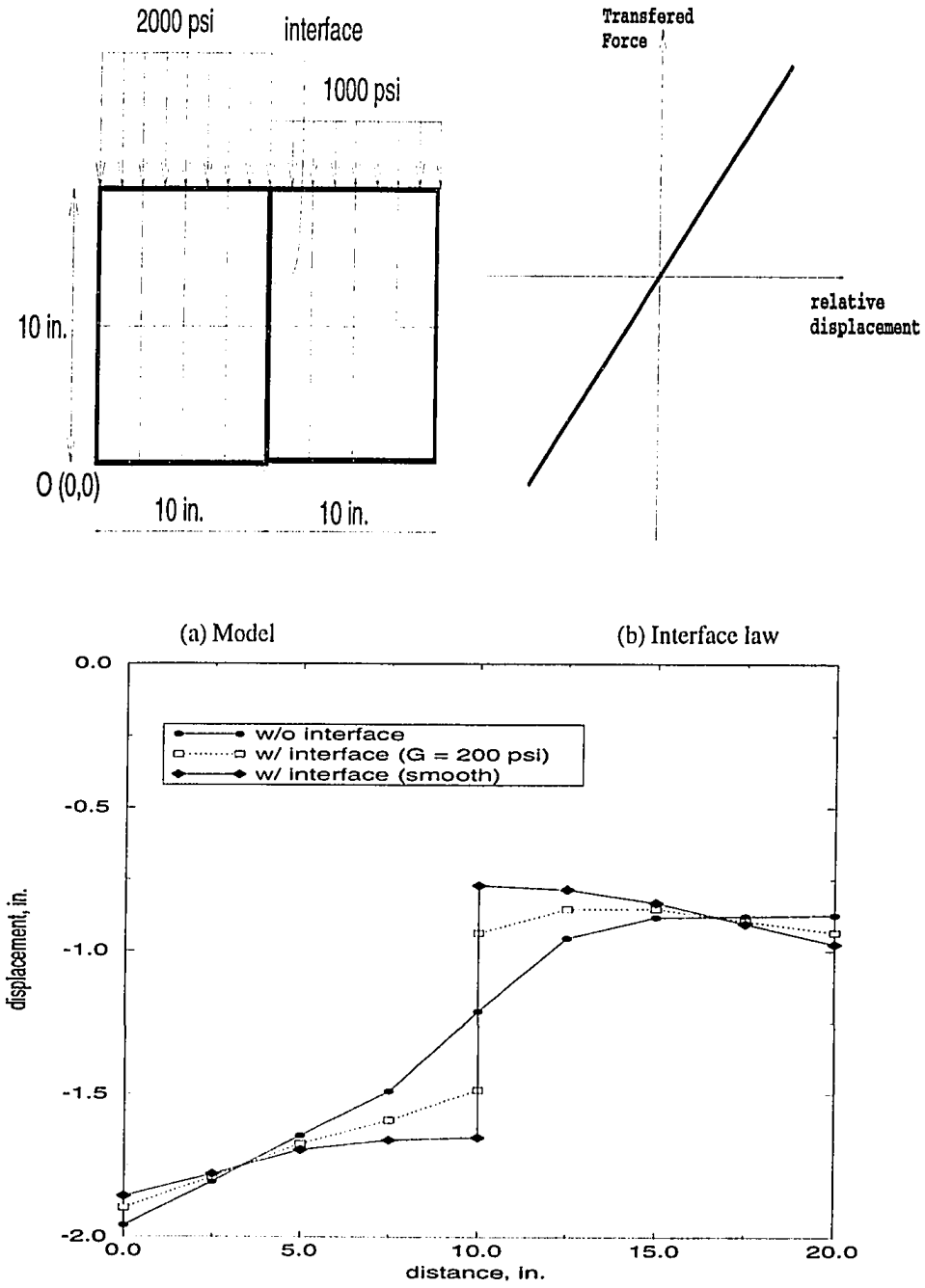
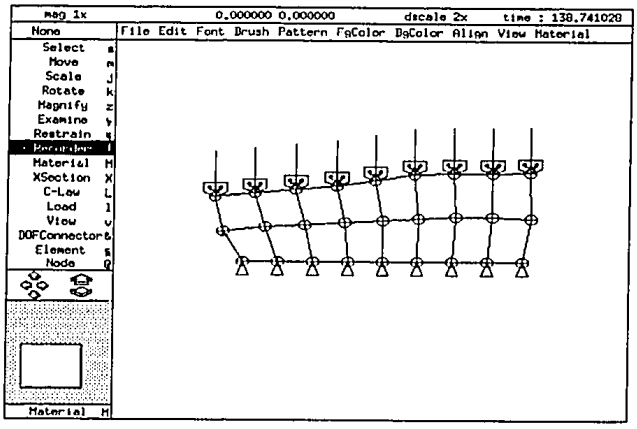


Figure 7.6: Comparision of uplift with DRAIN-2DX results

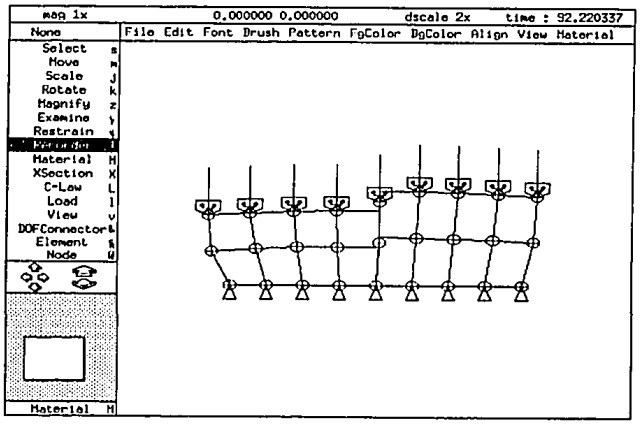


(c) deformation of the loading surface

Figure 7.7: Interface example

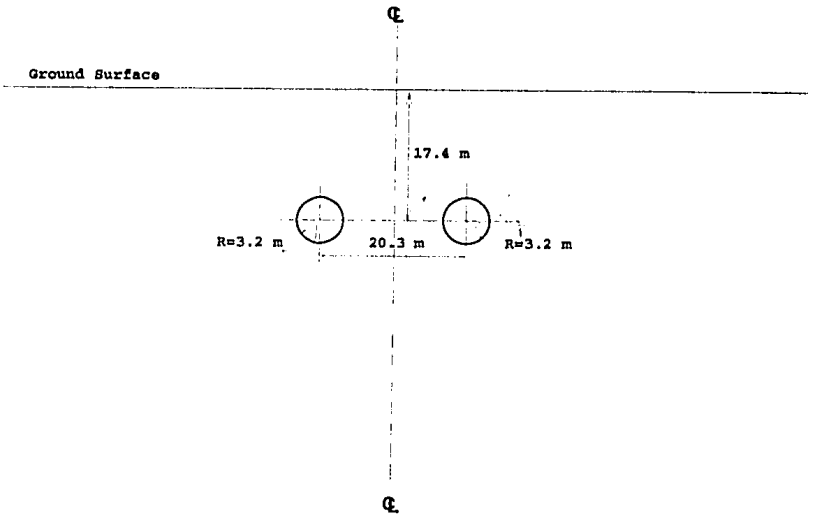


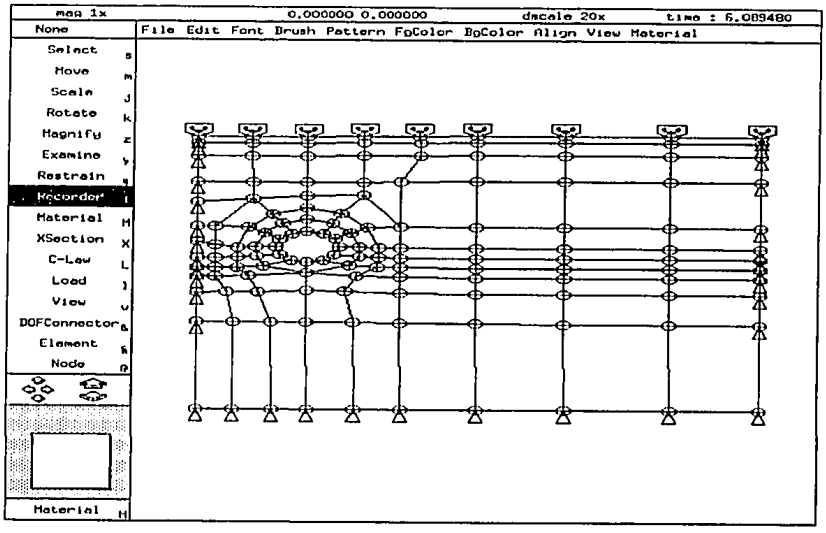
(a) without interface



(b) with interface

Figure 7.8: Comparision of displacements with and without interface elements





(b) Deformed mesh

Figure 7.9: Layout and model of the twin-tunnel

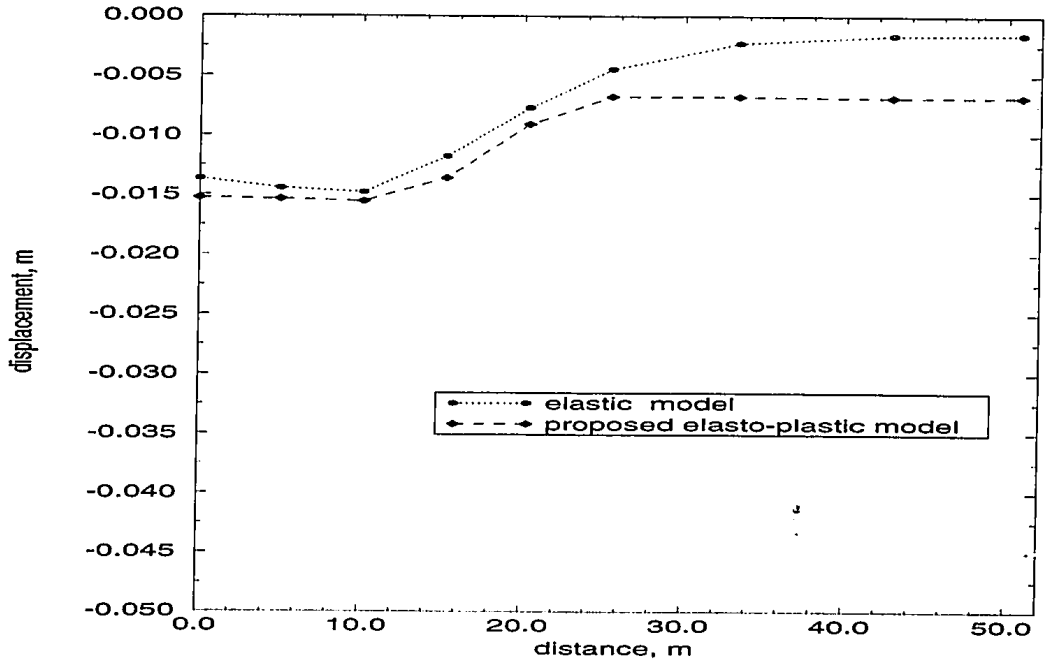


Figure 7.10: Surface settlement due to the construction of circular tunnel

Chapter 8

SUMMARY, CONCLUSIONS, AND RECOMMENDADTIONS FOR FURTHER RESEARCH

A new constitutive model for geomaterials and a computational framework for dynamic soil-structure interaction problems have been presented in this dissertation. The conclusions drawn from this study and recommendations for further research are summarized in the following sections.

8.1 Constitutive Model

The proposed constitutive model is based on non-associative, elasto-plastic, and double hardening concepts. The shear and volumetric mechanisms of the geomaterials are modeled separately, each with distinct yield and potential functions. The potential functions for shear and volumetric mechanisms coincide with the yield functions used in the original and modified Cam-Clay models respectively. The yield function for the shear mechanism is expressed in the form of constant stress ratio, while the yield function for the volumetric mechanism is expressed in the form of constant mean stress. The model also includes an alternate representation of volume change behavior of soils under isotropic compression. The isotropic compression and extension behaviors are characterized by exponential and power laws respectively. The following conclusions can be drawn from the development of this model.

- The proposed model is general; the model parameters are physically meaningful; and the number of model parameters is small.

- The model formulation representing the volume change behavior under isotropic compression ensures the existence of minimum void ratio, which corresponds to the state at which a geomaterial becomes incompressible. The validity of this formulation for sands as well clays has been established in the examples.

- The overall model is valid for general loading conditions, including monotonic and cyclic loading with drained and undrained drainage conditions. The validity of the model has been verified under monotonic and cyclic triaxial loading conditions.

- The model reproduces the complex essential features of sands such as path dependent dilatancy, hardening and softening, liquefaction, initial and induced anisotropy, and the behavior of soft-clay under compression.

- Simplified model equations have been presented for easy and efficient implementation of the model in a numerical scheme. The model has been implemented in the prototype finite element program developed in this study.

- The parameters of the proposed model for a given soil can be estimated from generally available experimental data using optimization techniques. The feasibility of using optimization schemes to estimate model parameters has been demonstrated by the numerical examples considered in Chapter 4.

- Simple experimental studies influencing selective parameters can be devised to estimate the model parameters with increased confidence.

8.2 Computational Framework

A computational framework for dynamic soil-structure interaction problems has been proposed in this study using object-oriented programming concepts. Based on this framework, a prototype finite element program has been developed implementing a direct integration

dynamic solution scheme. A basic graphical user interface has been implemented utilizing the UNIDRAW library based on the InterViews graphics library. The following conclusions can be drawn from the development of this computational framework.

- The design of the proposed framework is flexible, allowing convenient development and implementation.

- The framework provides a convenient tool for interactive modeling of soil-structure interaction problems with its graphical user interface capabilities. Visual feedback from the analysis can be displayed during and after the analysis.

- The treatment of element connectivity through connector objects allows interface elements to be modeled effectively without physically constructing stiffness matrices.

- The accuracy of the direct solution scheme, correction scheme, etc., implemented in the prototype finite element program, is checked and verified for non-linear problems.

8.3 Recommendations for Further Research

The conclusions drawn from both aspects of the study are very remarkable. Thus, further research based on this study will be useful in finding wider practical use of the ideas expounded in this dissertation.

With regard to the soil model, this study has clearly demonstrated the qualitative performance of the model with a scheme to estimate the model parameters using optimization schemes. Although the model parameters are estimated from experimental studies, the complexity of such a numerical exercise does not improve the chances for wider practical use of this model. On the other hand, practicing soil engineers are comfortable in estimating model parameters with simple physically meaningful laboratory experiments. To estimate the individual parameters in a unique manner a set of existing and new simple experiments have to be identified.

With regard to the computational framework, the study has clearly illustrated the improvements in increased user interaction, in the use of visualization of the model and results while retaining the accuracy of the traditional approaches. The design of the framework could be expanded to include other solution algorithms, numerical methods, and material models.

BIBLIOGRAPHY

Banerjee, S., Davis, R. O. & Sribalaskandarajah, K. (1992). Simple double-hardening model for geomaterials, *Journal of Geotechnical Engineering* **118**(6): 889–901.

Banerjee, S. & Pan, Y. W. (1986). A transitional yielding model for clay, *Journal of the Geotechnical Engineering Division, ASCE* **112**(2): 170–186.

Banerjee, S. & Sribalaskandarajah, K. (1994). An alternative formulation for volume-change behavior of soils, *in* A. T. Yeung & G. Y. Félio (eds), *Vertical and Horizontal Deformations of Foundations and Embankments; Proceedings of Settlement '94, Texas*, Vol. 1, ASCE, pp. 652–662. Geotechnical Special Publication No. 40.

Barzel, R. & Barr, A. H. (1988). A modelling system based on dynamic constraints, *Computer Graphics* **22**: 179–188.

Bazant, Z. P. (1977). Endochronic and classical theories of plasticity in finite element analysis, *International Conference on Finite Elements in Nonlinear Solid and Structural Mechanics*, Vol. 1, Geilo, Norway, pp. 151–164.

Bazant, Z. P. & Kriezek, R. J. (1976). Endochronic constitutive law for liquefaction of sand, *Journal of Engineering Mechanics* **102**(EM2): 225–238.

Biot, M. A. (1955). Theory of elasticity and consolidation for a porous anisotropic solid, *Journal of Applied Physics* **26**: 182–185.

Bonnet, A. & Dahan, C. (1983). Oil-well data interpretation using expert system and pattern recognition technique, *Proceedings of the eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, West Germany, pp. 185–189.

Butterfield, R. (1979). A natural compression law for soils (an advance on $e - \log p$), *Geotechnique* **29**(4): 469–480.

Casagrande, A. (1936). The determination of the pre-consolidation load and its practical significance; discussion d–34, *Proceedings of the First International Conference on Soil Mechanics and Foundation Engineering*, Vol. III, Cambridge, pp. 60–64.

Christian, J. T. (1991). Geotechnical engineering design in the age of the modern computer, *in* F. G. McLean, D. A. Campbell & D. W. Harris (eds), *Geotechnical engineering congress 1991 Vol.1*, ASCE, pp. 468–478.

Christie, I. F. & Tonks, D. M. (1985). Developments in the time lines theory of consolidation, *Proceedings of the Eleventh International Conference in Soil Mechanics and Foundations*, Vol. 2, San Francisco, pp. 423–426.

Clough, R. W. & Woodword III, R. J. (1966). Analysis of embankment stresses and deformations, *ASCE Speciality Conference on Stability and Performance of Slopes and Embankments*.

Couthino, A. L. G. A., Alves, J. L. D., Landau, L., Ebecken, N. F. F. & Troina, L. M. (1991). Comparison of lanczos and conjugate gradients for the element-by-element soluition of finite element equations on the IBM 3090 vector computer, *Computers and Structures* **39**: 47–56.

Dafalias, Y. F. (1986). Bounding surface plasticity. I: Mathematical foundation and hypoplasticity, *Journal of the Engineering Mechanics Division (ASCE)* **112**(9): 117–141.

Dafalias, Y. F. & Popov, E. P. (1975). A model of nonlinearly hardening materials for complex loading, *Acta Mech.* **21**(3): 173–192.

Davis, R. & Mullenger, G. (1978). A rate-type constitutive model for soil with a critical state, *International Journal of Numerical and Analytical Methods in GeoMechanics* **2**: 225–282.

Den Haan, E. J. (1992). The formulation of virgin compression of soils, *Geotechnique* **42**(3): 465–483.

Dennis, J. E., Gay, D. M. & Welsch, R. E. (1981a). An adaptive nonlinear least-squares algorithm, *ACM Transactions on Mathematical Software* **7**(3): 348–368.

Dennis, J. E., Gay, D. M. & Welsch, R. E. (1981b). Algorithm 573 NL2SOL - an adaptive nonlinear least-squares algorithm [E4], *ACM Transactions on Mathematical Software* **7**(3): 369–383.

DeRose, T. D. (1989). A coordinate-free approach to geometric programming, *Technical report*, Department of Computer Science, University of Washington, Seattle.

Desai, C. S. (1972). Non-linear analysis using spline functions, *Journal of Soil Mechanics and Foundation Engineering* **97**(SM10).

DiMaggio, F. L. & Sandler, I. S. (1971). Material model for granular soil, *Journal of the Engineering Mechanics Division, Proc. ASCE* **97**(EM3): 935–950.

Drucker, D. C., Gibson, R. E. & Henkel, D. J. (1957). Soil mechanics and work hardening theories of plasticity, *Transactions, ASCE* **122**: 338–346.

Duncan, J. M. & Chang, C. Y. (1970). Non-linear analysis of stresses and deformations, *Journal of Soil Mechanics and Foundation Engineering, ASCE* **96**(SM4): 1629–1653.

Gazetas, G. (1991). Foundation vibrations, *in* H.-Y. Fang (ed.), *Foundation Engineering Handbook*, 2nd edn, Van Nostrand Reinhold.

Hardin, B. O. & Drnevich, V. P. (1972). Shear modulus and damping in soils: Design equations and curves, *Proceedings of ASCE*, Vol. 98(SM6), pp. 603–624.

Hashiguchi, K. (1986). Elastoplastic constitutive model with a subloading surface, *Proceedings of the International Conference on Computational Mechanics*, Springer–Verlag.

Henkel, D. J. (1959). The relationships between the strength, pore-water pressure and volume change characteristics of saturated clays, *Geotechnique* **9**(2): 119–135.

Hill, R. (1950). *Mathematical Theory of Plasticity*, Oxford University Press, London.

Ishihara, K., Tatsuoka, F. & Yasuda, S. (1975). Undrained deformation and liquefaction of sand under cyclic stresses, *Soils and Foundations* **15**(1): 29–44.

Janbu, N. & Senneset, K. (1979). Interpretation procedures for obtaining soil deformation procedures, *Proceedings of Seventh European Conference in Soil Mechanics, Brighton*, Vol. 1, pp. 19–25.

Joyner, W. B. & Chen, A. T. F. (1975). Calculation of nonlinear ground response in earthquakes, *Bulletin of the Seismological Society of America* **65**(5): 1315–1336.

Kawamoto, T. & Okuzono, K. (1977). Analysis of ground surface settlement due to shallow shield tunnels, *International Journal for numerical and analytical methods in Geomechanics* **1**: 271–281.

Koiter, W. T. (1953). Stress-strain relations, uniqueness, and variational theorems for elastoplastic materials with a singular yield surface, *Quart. Appl. Math.* **11**(3): 350–.

Konder, R. L. (1963). Hyperbolic stress-strain response: Cohesive soils, *Journal of the soil Mechanics and Foundation Division, ASCE* **89**(SM1): 115–143.

Lade, P. (1977). Elasto-plastic stress-strain theory for cohesive soil with curved yield surfaces, *International Journal of Solids and Structures* **13**(11): 1019–1035.

Linton, M. A., Calder, P. R. & Vlissides, J. M. (1988). Interviews: A C++ graphical interface toolkit, *Technical Report CSL-TR-88-358*, Stanford University, Computer Systems Laboratory, Stanford University.

Masing, G. (1926). Eigenspannungen and verfestigung beim messing, *Proceedings of the Second International Congress of Applied Mechanics, Zurich, Switzerland.*

Matsuoka, H. & Nakai, T. (1974). Stress-deformation and strength charateristics of soil under three different principal stresses, *Transactions of JSCE* **6**: 108–109.

Miller, G. R., Banerjee, S. & Sribalaskandarajah, K. (1992). Interactive analysis of evolving earth structures, *in* G. N. Pande & S. Pietruszczak (eds), *Proceedings of the Fourth International Symposium on Numerical Models in Geomechanics-NUMOG IV, Swansea*, Balkema, Rotterdam, pp. 567–576.

Miller, G. R., Banerjee, S. & Sribalaskandarajah, K. (1995). A framework for interactive computational analysis in geomechanics, *Computers and Geotechnics* **17**(1): 17–37.

Mroz, Z. (1980). On hypoelasticity and plasticity approaches to constitutive modelling of inelastic behavior of soils, *International Journal for Numerical and Analytical Methods in GeoMechanics* **4**: 45–55.

Mroz, Z., Norris, V. A. & Zienkiewicz, O. C. (1981). An anisotropic critical state model for soils subjected to cyclic loading, *Geotechnique* **31**(4): 451–469.

Mroz, Z. & Pietruszczak, S. (1983). A constitutive model for sand with anisotropic hardening rule, *International Journal for Numerical and Analytical Methods in Geomechanics* **7**(3): 305–320.

Murayama, S. (ed.) (1985). *Constitutive laws of soil, report of ISSMFE subcommittee on constitutive laws of soils and proc. discussion session 1A*, Eleventh International Conference on Soil Mechanics and Foundation Engineering, San Francisco. Tokyo: Japanese Society of Soil Mechanics and Foundation Engineering.

Negussey, D. (1992). Private files of test data on isotropic consolidation of sands.

Newmark, N. M. (1959). A method of computation for structural dynamics, *Journal of engineering mechanics division, ASCE* **85**: 67–94.

Nishida, Y. (1956). A brief note on compression index of soil, *Journal of the Soil Mechanics and Foundations Division, Proceedings of ASCE* **82**(SM3): 1–14.

Ohmaki, S. (1979). A mechanical model for the stress-stain behavior of normally consolidated cohesive soil, *Soils and Foundations* **3**(19): 29–44.

Pietruszczak, S. & Mroz, Z. (1983). On hardening of anisotropy of Ko-consolidated clays, *International Journal for Numerical and Analytical Methods in Geomechanics* **7**(1): 19–38.

Prager, W. (1959). *Introduction to Plasticity*, Addison-Wesley, Reading, Mass.

Prakash, V., Powell, G. H. & Campbell, S. (1993). DRAIN-2DX base program description and user guide, *Technical Report UCB/SEMM-93/17*, Department of Civil Engineering, University of California, Berkeley, CA.

Prevost, J. H. (1987). Dynamics of porous media, *in* S. M. Sayed (ed.), *Geotechnical Modelling and Applications*, Gulf Publishing Company, pp. 76–146.

Prevost, J. H. & Hoeg, K. (1975). Effective stress-strain-strength model for soils, *Journal of Geotechnical Engineering* **101**(3): 257–288.

Priestley, M. J. N., Seible, F. & Chai, Y. H. (1992). Design guidelines for assessment of retrofit and repair of bridges for seismic performance, *Structural system research report Report No. SSRP-92/01*, Department of applied and engineering sciences, University of California, San Diego.

Pyke, R. (1979). Nonlinear soil models for irregular cyclic loadings, *Journal of the Geotechnical Engineering Division* **105**(GT6): 715–726.

Ramberg, R. M. & Osgood, W. R. (1943). Description of stress-strain curves by three parameters, *Technical Report Tech. Note 902*, National Advis. Comm. Aeronaut., Washington, D.C.

Rehak, D., Christiano, R. & Norkin, D. (1985). Sitechar: An expert system component of a geotechnical site characterization work bench, *in* C. L. Dym (ed.), *Applications of Knowledge-Based Systems to Engineering analysis and design*, ASCE, pp. 117–133.

Robinson, V. & Frank, A. (1987). Expert systems for geographic information systems, *Photogrammertric engineering and remote sensing* **53**(10): 1435–1441.

Romano, M. (1974). A continuum theory for granular media with a critical state, *Arch. of Mech.* **26**: 1011–1028.

Roscoe, K. H. & Burland, J. B. (1968). On the generalized stress-strain behavior of "wet" clay, *Engineering Plasticity*, Cambridge University Press, Cambridge, U.K., pp. 535–609.

Roscoe, K. H., Schofield, A. N. & Wroth, C. P. (1958). On the yielding of soils, *Geotechnique* **8**(1): 22–53.

Saada, A. & Bianchini, G. (eds) (1987). *Proceedings of International Workshop on Constitutive Equations for Granular Non-Cohesive Soils*, Balkema, Cleveland.

Saleeb, A. & Chen, W. (1980). Nonlinear hyperelastic (green) constitutive models for soils, part i - theory and calibration, part ii - predictions and comparisions, *Proceedings of the North American Workshop on Limit Equlibrium, Plasticity and Generalized Stress-Strain in Geotechnical Engineering*, McGill University, Montreal, Canada.

Scott, R. F. (1985). Plasticity and constitutive relations in soil mechanics, *Journal of Geotechnical Engineering* **111**(5): 563–605.

Seed, H. B. & Idriss, I. M. (1970). Soil moduli and damping factors for dynamic response analysis, *Report 70–10*, EERC, University of California, Berkeley.

Smith, R. & Baker, J. (1983). The dipmeter advisor system: A case study in commercial expert system development, *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, West Germany, pp. 122–129.

Smith, S. K. (1995). *Parametric analysis of dynamically loaded concentrically braced steel frames allowd to uplift*, Master's thesis, University of washington.

Sribalaskandarajah, K. & Banerjee, S. (1996). An improved double-hardening model for geomaterials (tentatively accepted for publication), *Journal of Geotechnical Engineering* .

Tatsuoka, F. & Ishihara, K. (1973). Stress path and dilatancy performance of a sand, *Proceeding of Eighth International Conference in Soil Mechanics and Foundation Engineering*, Vol. 1, pp. 419–424.

Tatsuoka, F. & Ishihara, K. (1974a). Drained deformation of sand under cyclic stresses reversing direction, *Soils and Foundations* **14**(3): 51–65.

Tatsuoka, F. & Ishihara, K. (1974b). Yielding of sand in triaxial compression, *Soils and Foundations* **14**(2): 63–76.

Terzaghi, K. (1943). *Theoretical soil mechanics*, Wiley, New York.

Truesdell, C. (1955). Hypo-elasticity, *Journal of Rational Mechanics and Analysis* **4**(1): 83–133.

Truesdell, C. & Toupin, R. (1960). The classical field theories, *in* S. Flugge (ed.), *Handbuch der Physik*, Vol. III, Springer-Verlag, Berlin.

Valanis, K. C. (1971). A theory of viscoplasticity without a yield surface; part I: General theory; Part II: Application to the mechanical behavior of metals, *Arch. of Mech.* **23**: 517–555.

Valanis, K. C. & Read, H. E. (1982). A new endochronic plasticity model for soils, *in* Pande & Zienkiewicz (eds), *Soil Mechanics – Transient and Cyclic Loads, Constitutive Relations and Numerical Treatment*, Wiley, pp. 375–417.

Vermeer, P. A. (1978). A double hardening model for sand, *Geotechnique* **28**(4): 413–433.

Vlissides, J. M. (1990). Generalized graphical object editing, *Technical Report CSL-TR-90-427*, Stanford University, Computer Systems Laboratory, Stanford University.

Whitman, R. V. & Bailey, W. A. (1966). Use of computers for slope stabilty analysis, *ASCE Speciality Conference on Stability and Performance of Slopes and Embankments*, Berkeley, CA, pp. 519–542.

Wong, I. (1971). *Analysis of Braced Excavations*, Sc.D thesis, Department of Civil Engineering, MIT, Cambridge, Mass.

Zienkiewicz, O. C. (1977). *The finite element method*, McGraw-Hill Book Company (UK) Ltd.

Curriculum Vitae

KANDIAH SRIBALASKANDARAJAH

Business Address:
233, More Hall FX-10
University of Washington
Seattle WA 98195
(206) 543-6778

Home Address:
1120 NE 43rd St. #404
Seattle WA 98105
(206) 633-1301

Degrees

Ph.D. in Civil Engineering (Geotechnical) — Mar. 1996
University of Washington, Seattle, WA

M.Eng. in Civil Engineering (Structural) — Apr. 1988
Asian Institute of Technology, Bangkok, Thailand

B.Sc.Eng.(Hons.) in Civil Engineering — Dec. 1984
First Class Honors
University of Peradeniya, Peradeniya, Sri Lanka

Publications

"An Improved Double-Hardening Model for Geomaterials", Tentatively accepted for publication in Journal of Geotechnical Engineering, ASCE, 1996 (with S. Banerjee)

"A Framework for Interactive Computational Analysis in Geomechanics", Computers and Geotechnics, Vol. 17(1), pp. 17-37, 1995 (with G. R. Miller and S. Banerjee)

"An Alternative Formulation of Volume-Change Behavior of Soils", in "Vertical and Horizontal Deformations of Foundations and Embankments, Proceedings of Settlement '94", ASCE, vol. 1, pp. 652-662, June 1994 (with S. Banerjee)

"Interactive Analysis of Evolving Earth Structures", in "Numerical Models in Geomechanics, Proceedings of the Fourth International Symposium on Numerical models in Geomechanics - NUMOG IV", pp. 567-576, August 1992 (with G. R. Miller and S. Banerjee)

"Simple Double-Hardening Model for Geomaterials", Journal of Geotechnical Engineering, ASCE Vol. 118(6), pp 889-901, June 1992 (with S. Banerjee and R. O. Davis)

"Boundary Element Scheme for Multi-Layered Porous Halfspace", Masters Thesis, Asian Institute of Technology, Bangkok, Thailand, 1988

"Computer Aided Design of Pin-Jointed Plane Trusses", Research Report, University of Peradeniya, Sri Lanka, 1984.

March, 1996