MODELING OF REGID PAVEMENTS: JOINT SHEAR TRANSFER MECHANISMS AND FINITE ELEMENT SOLUTION STRATEGIES

WA-RD 455.1

Research Report
July 1998

Washington State Department of Transportation

Washington State Transportation Commission
Planning and Capital Program Management
in cooperation with:
U.S. DOT - Federal Highway Administration

Final Report
Research Project T9903, Task 54
Enhanced Finite Element Tools for Rigid Pavement Analysis

MODELING OF RIGID PAVEMENTS: JOINT SHEAR TRANSFER MECHANISMS AND FINITE ELEMENT SOLUTION STRATEGIES

by

William G. Davids, George M. Turkiyyah, and Joe P. Mahoney
Department of Civil and Environmental Engineering
University of Washington
Seattle, Washington 98195-2700

Washington State Transportation Center (TRAC)
University of Washington
Box 354802
University District Building
1107 NE 45th Street, Suite 535
Seattle, Washington 98105-4631

Washington State Department of Transportation
Technical Monitors

Robyn Moore
Pavement and Soils Engineer

Linda Pierce
Pavement Management Engineer

Prepared for

Washington State Transportation Commission
Department of Transportation
and in cooperation with
U.S. Department of Transportation
Federal Highway Administration

July 1998

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
WA-RD 455.1		

4. TITLE AND SUBTITLE	5. REPORT DATE
Modeling of Rigid Pavements: Joint Shear Transfer Mechanisms and Finite Element Solution Strategies	JULY 1998
	6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S)	8. PERFORMING ORGANIZATION REPORT NO.
William G. Davids, George M. Turkiyyah, and Joe P. Mahoney	
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. WORK UNIT NO.
Washington State Transportation Center (TRAC) University of Washington, Box 354802 University District Building; 1107 NE 45th Street, Suite 535 Seattle, Washington 98105-4631	11. CONTRACT OR GRANT NO. Agreement T9903-54
12. SPONSORING AGENCY NAME AND ADDRESS	13. TYPE OF REPORT AND PERIOD COVERED
Washington State Department of Transportation Transportation Building, MS 7370 Olympia, Washington 98504-7370	Final
	14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

This study was conducted in cooperation with the U.S. Department of Transportation, Federal Highway Administration.

16. ABSTRACT

This report documents the modeling strategy and associated verification which permits analysis of multiple PCC plain jointed slabs. Several subbase/soil layers below the slabs and separation of the slab and base layer are considered. Load effects include those associated with both multiple tire and linear temperature gradients. A technique for modeling aggregate interlock shear transfer across PCC pavement joints is developed and verified. The overall development is contained in the computer program EverFE. The computer program uses a graphical user interface which aids its use.

17. KEY WORDS	18. DISTRIBUTION STATEMENT
Finite elements, rigid pavement, joints load transfer, aggregate interlock, EverFE	No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22616

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
None	None	191	

DISCLAIMER

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Washington State Transportation Commission, Department of Transportation, or the Federal Highway Administration. This report does not constitute a standard, specification, or regulation.

Contents

List of Figures

List of Tables

List of Symbols

A	=	cross-sectional area of dowel
A_{sv}, A_{sw}	=	shear areas of dowel
$\overline{A}_x, \overline{A}_y$	=	probable horizontal and vertical contact areas of crack plane
a_i	=	tensor defining i^{th} dowel node's independent DOF
b_i	=	tensor defining i^{th} dowel node's constrained DOF
B	=	matrix of differential operators
c_{ij}	=	scalar distance between nodes i and j used in constraint equation
C	=	right-hand side vector in constraint equation
d_i	=	displacement vector at node i
ds	=	incremental vector along dowel
D	=	constitutive matrix
E	=	modulus of elasticity
e	=	error in solution vector
f, F	=	element nodal force vectors
G	=	shear modulus
G	=	coefficient matrix for constraints
h	=	scale factor for dowel coordinate mapping
h	=	residual vector in conjugate gradient iteration
H	=	matrix operator to compute relative element nodal displacements
(i, j, k)	=	Cartesian unit vectors
I_v, I_w	=	dowel moments of inertia
I	=	2^{nd} order identity tensor

J = dowel torsional moment of inertia

J = Jacobian matrix

k = modulus of subgrade reaction

K = system or element stiffness matrix

m = displacement magnitude of embedding element in gap direction

n = number of unknowns (nodal degrees of freedom)

N_i = i^{th} element shape function

n = unit normal vector

N = matrix of element shape functions

P, p = vector of applied nodal forces

$(\mathrm{q}, \mathrm{s}, \mathrm{t})$ = dowel element local unit vectors

Q = basis transformation matrix for interface

r = system residual force vector

R = restriction operator

s = dowel local coordinate

T = interpolation operator; transformation matrix for dowel

(u, v, w) = scalar displacement components at a node in (i,j,k) directions

u, U = system or element displacement vector

v = unit vector in direction of aggregate interlock shear

y = residual vector in constraint equation

z = displacement vector in conjugate gradient iteration

α = coefficient of thermal expansion

β = inner loop limit in multigrid algorithm

γ = magnitude of gap between dowel and slab

δ_n, δ_t = normal and tangential relative displacements across crack plane

δv = relative displacement between dowel and slab

Δ = vector of relative displacements

ϵ = residual solution tolerance

η = dowel element local coordinate

$(\theta_u, \theta_v, \theta_w)$ = scalar nodal rotations in (i,j,k) directions

Θ = nodal rotation vector

κ = penalty parameter

μ = coefficient of friction between paste and aggregate

ν = Poisson's ratio

(ξ, η, ζ) = element local coordinates

Π = internal virtual work

ρ = mass density

σ_n, σ_t = normal and tangential stresses on crack plane

σ_{pu} = ultimate strength of cement paste

σ = stress tensor or vector

τ_{pu} = friction at cement-aggregate interface

ϕ = constraint tolerance

Φ_i = i^{th} rotational shape function for dowel

Φ = vector of dowel rotational shape functions

ω = scalar perturbation parameter in Uzawa iteration

Abstract

This study addresses the need to better quantify and model load transfer across joints in rigid (concrete) pavement systems, and presents efficient solution strategies for making three-dimensional finite element analyses of rigid pavements and other large structural systems practical on desktop computers.

A modeling strategy is developed which permits the analysis of multiple slab systems having various geometries. Several soil layers below the slabs and separation of the slab and base layer are considered. Load effects include accurately located tire pressures and linear temperature gradients through the thickness of the slabs.

An embedded bending element is developed and verified that permits the precise location of dowels within the model without regard to meshing of the slab. Debonding and nonlinearities due to looseness of the dowels may be explicitly captured. Details of the inclusion of a general bond-slip law are also presented, extending the capabilities of the element to the modeling of more general reinforced concrete structures.

A technique for modeling aggregate interlock shear transfer across rigid pavement joints is developed that relies on a two-phase model for the concrete. The aggregate interlock model captures variations in shear stiffness and strength with aggregate size, concrete strength, and joint opening and relative tangential displacement.

An efficient iterative solution strategy for large-scale structural systems involving material and contact nonlinearities is developed and tested. This solution strategy relies on a multigrid-preconditioned conjugate gradient solver that permits the use of unstructured, unnested mesh sequences, the incorporation of nodal contact conditions, spatially varying materials, and the meshing of multiple element types within a single model.

Verification of the finite element models is accomplished through comparison with existing laboratory test data of doweled rigid pavements. A parametric study on the decrease in dowel load transfer due to dowel looseness is presented, as is a study on the effect of joint opening on aggregate interlock shear transfer. The potential for detrimental effects on rigid pavements due to poor load transfer is clearly demonstrated by these studies.

Acknowledgments

This study was supported in part by WSDOT under grant T9903-54 and by fellowships from the Valle Scholarship and Scandinavian Exchange Program and the Osberg Family Trust at the University of Washington.

Chapter 1

Introduction

1.1 Context

Jointed plain concrete pavement (JPCP) has been in use throughout the world for many years [39], and is common in both roadways and airport runways. In general, JPCP consists of several components as illustrated in Fig. 1.1:

- Multiple unreinforced concrete slabs, often tied to each other by dowels (round steel bars) that assist in transferring load between slabs.
- Base layers consisting of compacted gravel or cement-treated soils, which are often placed below the slabs to better distribute load to the natural subgrade.
- The natural subgrade below the base layers.

A significant feature of JPCP are the joints, which are of two primary types: contraction and construction (see Fig. 1.2). Contraction joints are usually spaced about every 3-5m to permit expansion and contraction of the slabs and control crack location. They are formed by sawcutting 1/4-1/3 the pavement thickness after curing, forcing a crack to occur at the joint through the pavement thickness. Aggregate interlock arising from the interaction of the two rough joint surfaces is an important load transfer mechanism at contraction joints. Construction joints are formed at the end of a pour, and in contrast to contraction joints, have relatively smooth joint surfaces with no aggregate interlock load transfer.

Rigid pavements are complex structural systems. To design them efficiently, engineers need to evaluate the effects of critical load cases and predict the corresponding failure modes, evaluate pavement lifetimes, and develop better retrofit strategies for damaged pavements.

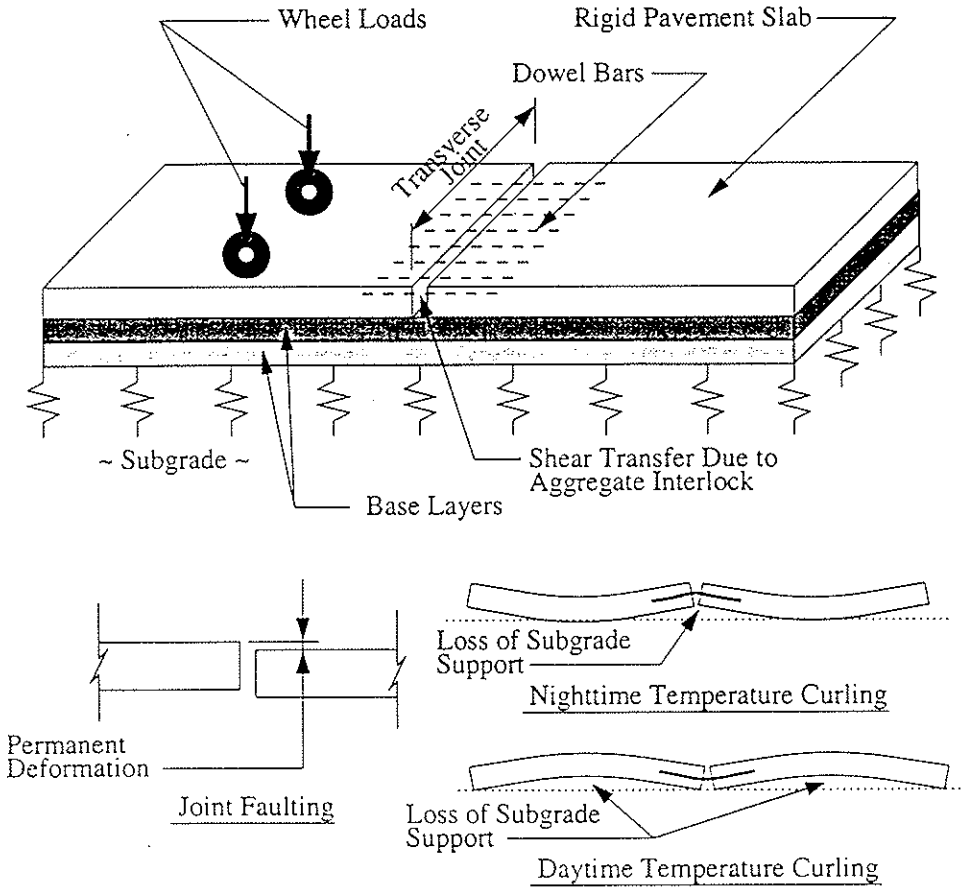


Figure 1.1: System of Two Doweled Rigid Pavement Slabs

Achieving these objectives relies on accurately predicting the structural response of the pavement system, which is affected by the following factors:

- Complex load transfer mechanisms at the joints.
- Temperature-induced stresses and loss of foundation support (see Fig. 1.1).
- Dynamic effects of wheel loading.
- Material behavior of both the slab and subgrade.

In recent years, there has been a movement toward mechanistic design procedures for pavements which holds promise for rigorously incorporating these factors in design. This movement has led researchers to use finite element techniques to analyze the response of rigid pavement systems to applied static, dynamic, and thermal loadings [84, 54, 96, 60]. However, despite the active research into the finite element analysis of rigid pavements, the

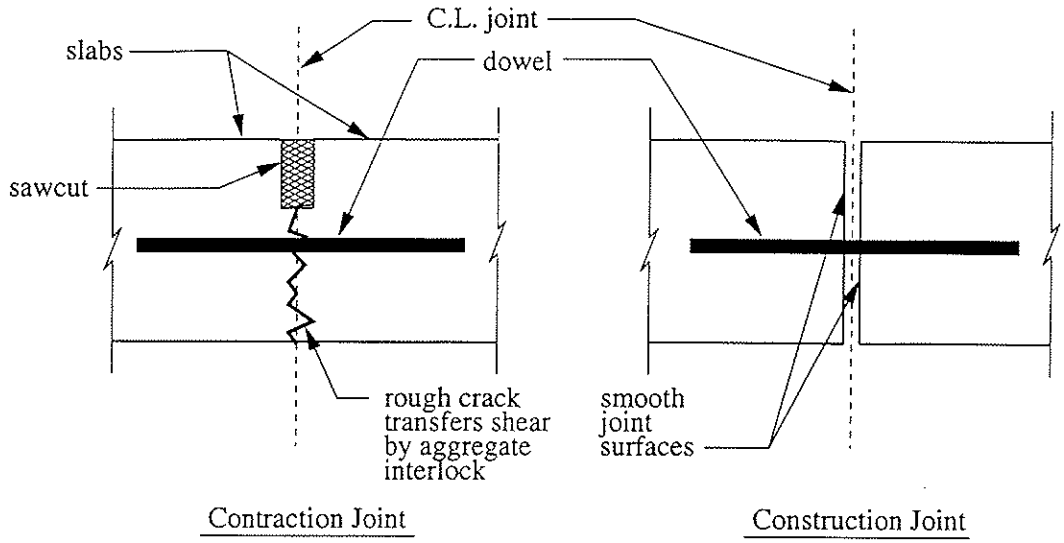


Figure 1.2: Typical Rigid Pavement Joints

methods commonly used to model joint response are simplistic, and do not accurately capture the complex behavior of aggregate interlock and dowel-slab interaction. This is of great concern, since serviceability failures of JPCP are often the result of poor joint performance. For example, faulting, where a permanent relative vertical displacement develops between adjacent slabs as shown in Fig. 1.1, is a common joint failure that is often attributed to poor joint shear transfer and localized degradation of the supporting soil. In addition to concerns about the modeling of joint shear transfer, solution of the three-dimensional (3D) finite element models needed to capture pavement response is not feasible on the desktop computers commonly available to designers.

1.2 Research Objectives

This study focuses on overcoming particular deficiencies in current state-of-the-art analysis of JPCP. In particular, this study focuses on the following topics:

- *Development of new techniques for modeling dowel load transfer.* These techniques permit the consideration of dowel looseness that is known to have a large effect on joint performance [10, 81], but has not been explicitly modeled in previous studies.

- *Investigation into new methods for modeling aggregate interlock shear transfer.* A modeling strategy is developed and implemented that rationally incorporates realistic crack constitutive relations that reflect variations in joint load transfer with joint

opening, load magnitude, and concrete properties.

- *Development of efficient solution strategies.* A nonlinear multigrid-preconditioned conjugate gradient solution method appropriate for use with unstructured finite element models of general, 3D geometries is developed. This method permits the rapid solution of modest finite element models of rigid pavement systems on desktop computers and will permit the investigation of detailed models on high-end workstations.

To aid in the implementation of new joint modeling techniques, a general, object-oriented finite element program suitable for 3D structural analysis is developed. This program incorporates the proposed nonlinear solution strategies, and is designed to serve as a valuable tool for the analysis of rigid pavement systems when integrated with an appropriate user interface. To verify the accuracy of the proposed modeling techniques and solution strategies, results from two finite element models of experimental systems are compared with measured data. Numerical studies are performed to study the changes in response of typical systems to model parameters, and to verify their applicability.

1.3 Organization of Study

The remainder of this study is organized as follows:

- In Chapter 2, a survey of the literature on the finite element modeling of rigid pavement slabs is presented. This serves to define the current state-of-the-art, and help identify what specific modeling strategies are needed.

- Chapter 3, drawing on the conclusions of the literature review of Chapter 2, presents an overview of the modeling strategies used in this study. Included are issues such as slab and subgrade discretization, material types, slab-subgrade loss of contact, and methods of loading. Dowel shear transfer, aggregate interlock, and multigrid solution strategies are individually addressed in Chapters 4-6.

- Chapter 4 covers the modeling of the dowels and dowel-slab interaction. An embedded bending element designed specifically for modeling dowel action is developed that explicitly incorporates contact conditions due to gaps between the dowel and slab. In addition, performance and convergence of the dowel element, and the effects of gaps between the dowel and the slab on joint response, are examined.

- Chapter 5 focuses on the mechanism of aggregate interlock and its inclusion in finite element models of rigid pavement systems. A detailed literature review is presented to explain the mechanics of aggregate interlock and provide an overview of past attempts to model this phenomenon. Based on this research and insight into the physical mechanism of aggregate interlock shear transfer, a modeling technique is developed that rigorously incorporates crack constitutive relations at joints.

- Chapter 6 addresses solution strategies, including the development and implementation of a nonlinear solution scheme appropriate for use with large-scale models incorporating material and contact nonlinearities. A multigrid-preconditioned conjugate gradient solver applicable to unstructured 3D finite element discretizations is developed to solve the linearized systems of equations arising at each iteration in the nonlinear solution. In addition to extensive performance studies, a model problem of a rigid pavement system is solved to illustrate the use of the solution strategy in the context of the present study.

- Chapter 7 examines the accuracy of the finite element models developed in this study by comparison with laboratory test data collected by Hammons [33]. The test data include scale models of doweled two-slab systems both with and without cement-treated base layers.

- Chapter 8 presents parametric studies designed to indicate the usefulness of the proposed joint shear transfer modeling strategies for pavement evaluation and design. The effect of dowel looseness on joint shear transfer efficiency and pavement response, including slab and subgrade stresses, is addressed. The effect of joint opening on aggregate interlock shear transfer efficiency is also examined in detail. Both wheel and temperature loadings are considered in these studies.

- Chapter 9 presents a summary of this study, along with conclusions and recommendations for future research.

Chapter 2

Literature Review

2.1 Early Finite Element Models of Rigid Pavement Systems

In an effort to overcome the limitations of early analytical methods, the finite element method was first employed to model the response of rigid pavements in the early 1970's. Wang, Sargious, and Cheung [80] studied the response of rigid pavements subjected to wheel loadings using a linear finite element model. The slab was modeled with medium-thick plate elements assuming Kirchoff plate theory. The foundation was considered to be either an elastic half-space, for which an equivalent foundation stiffness matrix was developed by inverting the foundation flexibility matrix determined using the Boussinesq equation, or a dense liquid (Winkler) foundation. Shear load only was transferred at the longitudinal joints assuming perfect load transfer efficiency. The effect of transverse joints was not considered, since only a portion of a single roadway slab was modeled due to the need to minimize the problem size. Slab stresses and deflections were computed using the finite element model with both a continuous foundation and a Winkler foundation, and were compared to stresses computed using Westergaard's equation. In general, Westergaard's solution agreed closely with the finite element method results assuming a Winkler foundation; however, the finite element model results assuming a continuous foundation yielded higher stresses and deflections.

Soon after this initial study, Huang [37] presented another finite element model for rigid concrete paving systems. In this model, the effect of an adjacent slab, connected by shear transfer devices at a transverse joint, was considered. Load transfer efficiency was assumed to be perfect. In addition, stresses due to temperature curling were considered. The foundation was modeled as an elastic continuum, and loss of contact was considered. The model was verified by comparison with analytical solutions and deflections and stresses

measured during the Arlington Road Test [87, 88, 89].

Following the development of these early models, Tabatabaie and Barenberg [82, 84] developed a more general finite element program called ILLI-SLAB which is still in use today. ILLI-SLAB utilizes the same medium-thick plate elements employed in earlier models, and the effect of a bonded or unbonded base can be incorporated using a second layer of plate elements below the slab. The subgrade is modeled as a Winkler foundation. Dowel bars at joints are modeled as discrete bar elements, and shear transfer via aggregate interlock or keyway is modeled with spring elements. Further, the relative displacement between dowel bars and the concrete pavement can be considered by placing a discrete spring element at each end of the dowel bar. Verification of models developed with ILLI-SLAB was achieved by comparison with theoretical solutions for stresses and displacements, as well as comparison with actual field test data from the AASHO Road Tests [1] and tests of concrete pavement on cement treated subbases. Studies were performed to examine the effects of load transfer efficiency on the behavior of plain jointed concrete pavement as well as the effect of transverse crack spacing on load stresses in continuously reinforced concrete pavement. Further, increased support provided by a tied concrete shoulder was examined, as was the the effect of a stabilized base when modeled as a discrete stiff layer. One of the main conclusions from the early studies using ILLI-SLAB was that joint load transfer efficiency must be modeled accurately to predict the possibility of permanent subgrade deformation and slab distress near the slab corners.

Tayabji et al. [86] developed the program JSLAB for analyzing pavements resting on a Winkler foundation. The model incorporates features similar to ILLI-SLAB, utilizing plate elements to model the slab and a bonded or unbonded base. Dowels were modeled with modified beam elements that incorporated the effect of shear deformation and the elastic support provided by the concrete. As in ILLI-SLAB, aggregate interlock and keyways were modeled with springs.

In 1983, Huang [38] extended his earlier model to allow the consideration of multiple slabs and various load transfer devices in a manner similar to ILLI-SLAB. It should be noted that dowels were modeled as having shear stiffness only across the joint, i.e. bending deformations of the dowels were not considered. The deformation of the concrete under the dowel was considered by treating the embedded portion of the dowel as an infinitely long beam on elastic foundation. The subgrade was modeled as an elastic half-space and loss of contact between the subgrade and the slab was considered. Comparisons between the

finite element model results and data from the Arlington Road Test [87, 88, 89] were made for various slab thicknesses. Further, the effects of mesh refinement and loss of subgrade contact on the computed deflections and stresses were studied. Huang concluded that loss of contact should be considered for wheel loads applied near the pavement edge where actual loss of subgrade support often occurs.

A 1988 study by Krauthammer [52] utilized a complex shear transfer mechanism as part of a 2D plane strain dynamic finite element model to examine the effects of shear transfer on pavement behavior. The shear transfer mechanism that was employed consisted of two dowel elements cantilevered from the ends of the slab which were connected by a pure shear element. Horizontal forces were transferred via horizontal truss elements. To model dowel looseness and slip, a previously developed shear transfer model [51] was employed for the pure shear element used at the joint. Model results were generated for falling weight deflectometer loads. Various levels of joint deterioration were considered in the study, as were simulated subgrade deterioration near the joint and the effects of asphalt concrete overlays. Variations in shear and horizontal stresses in the concrete slab, joint efficiency, deflections adjacent to the joint, and subgrade shear stresses were examined.

Following the initial development of these and other similar finite element models, investigators began detailed studies of load transfer, subgrade nonlinearity, and temperature-induced curling stresses (see [42, 44] for example). However, the models employed still relied on medium-thick plate bending elements; as a result, shear stresses and normal stresses through the thickness of the plate were not considered, limiting the effectiveness of the models for detailed stress analysis. In addition, the concrete was modeled as linearly elastic, and aggregate interlock shear transfer was incorporated with linear spring elements. Lastly, with the exception of [53], dynamic load effects were not considered in these early studies. To overcome these limitations, new finite element models have recently been developed. Some investigators have focused on material nonlinearities, while others have performed both linear and nonlinear dynamic finite element analyses to better capture the applied load effects. Most investigators have moved to 3D models, although plate models are still in use. The remainder of this Chapter details these more recent investigations into the structural response of rigid pavement systems.

2.2 Current FE Models of Rigid Pavement Systems

2.2.1 Ioannides and Donnelly (1988)

This study [43] was an early attempt to model a single rigid concrete pavement slab and subgrade using a 3D finite element model. The focus of the study was the effect of subgrade nonlinearity on pavement response. The effects of mesh refinement, vertical and lateral subgrade extent, and boundary conditions on the predicted response were also examined.

The existing 3D finite element program GEOSYS was adopted for the analysis. Linear, 8-noded brick elements were used to model the slab and subgrade, with varying degrees of mesh refinement. The effects of lateral and vertical subgrade extent were examined, with the investigators concluding that the subgrade depth should extend to the point where constant strain is reached. This depth varies depending on the lateral extent of the subgrade, but was generally about 40' for a lateral subgrade extent of between 25' and 35'. The boundary conditions used in the study were vertical supports on the bottom of the subgrade, and lateral restraints on the vertical sides of the subgrade.

Vertical mesh gradation studies showed that sufficient accuracy was achieved when the slab was divided into two layers. Horizontal mesh gradation and element aspect ratio were also studied, resulting in the conclusion that the mesh needs to be refined in the region of loading, and the maximum element aspect ratio should be kept below four.

Interior, edge, and corner wheel loadings were applied and the results compared with analytical solutions and 2D analyses reported previously by Ioannides et al. [42]. To accentuate the effects of subgrade nonlinearity, the foundation material assumed was a relatively soft cohesive material, and the wheel loads used were fairly large. The interior loading produced little subgrade nonlinearity. Comparison with the 2D results and analytical solutions showed some discrepancy in the deflections and stresses which the authors attributed to mesh refinement in the region of the load. The edge loading produced more pronounced subgrade nonlinearity, but the overall response of the slab was not significantly affected, with the upper 2.0 m of the foundation exhibiting significant nonlinear response. Comparison with 2D solutions showed relatively good agreement for slab deflections, with the discrepancies in slab stresses being somewhat larger. The differences in subgrade stresses were quite significant. In the case of corner loading, the effects of subgrade nonlinearity were more significant, with about 20% of the elements experiencing reduced moduli, with these elements concentrated in the upper 1.7 m of the subgrade. Comparison with the ILLI-SLAB 2D solutions for stresses and displacements showed discrepancies of less than

10% for slab stresses and deflections; the discrepancy in the foundation stresses was much larger.

The main conclusion of the authors [43] was that 3D analysis is a valid modeling tool for rigid concrete pavements; however, existing 2D finite element models are reasonably accurate. The effect of stress dependent subgrade stiffness may be significant for corner and edge loadings, but slab deflections and bending stresses are less affected than the subgrade stress.

2.2.2 Channakeshava, Barzegar, and Voyiadjis (1992)

This study [12] focused on the development of a nonlinear, 3D static finite element model of plain concrete pavements with doweled joints. The goal of the authors was to adequately model the 3D response, overcoming the limitations of simpler 2D models utilizing plate elements. To model cracking of the concrete pavement, a plastic constitutive model augmented by a smeared cracking model was implemented for the slab concrete. Loss of subgrade support adjacent to transverse joints was modeled, and the progressive softening of dowel load transfer was considered. The loadings included temperature gradients (curling) and wheel loads. The finite element model of the slab employed 20-noded, quadratic, isoparametric brick elements.

The subgrade was idealized as a dense liquid foundation with three discrete linear springs at each node on the base of the slab. The vertical spring resisted only compressive forces, allowing the slab to lift freely off the subgrade. The two horizontal springs were given a small stiffness to prevent numerical difficulties, but no attempt was made to model horizontal frictional effects. The loss of subgrade support adjacent to the joint was simulated by reducing the stiffness of the subgrade springs.

The dowel bars were modeled with discrete bending elements. To model the interaction between the dowels and the concrete, including looseness effects, three discrete nonlinear springs were used to connect each end of the dowel to the slab. The stiffnesses of these springs were determined in a separate local finite element model of a single dowel bar where the dowel was embedded in the slab, and cantilevered out from the face a distance equal to the joint opening. Increasing point loads (shear forces) were applied at the tip of the dowel, and the displacements at the face of the slab were determined, giving the force-deformation relationships for the nonlinear springs at the ends of the dowel. This allowed the effects of local stress concentrations and concrete nonlinearities to be accounted for. Varying concrete strengths were assumed for this analysis in an attempt to quantify the loss of shear transfer

efficiency due to fatigue.

Various loadings were considered, including wheel loads, nighttime curling (cooling of the top of the slab relative to its base), and daytime curling (heating of the top of the slab relative to its base). The main conclusions of the study were as follows:

- Dowel-concrete interface stiffness is reduced due to high stress concentrations near the dowel. This loss of stiffness increases with repeated loadings due to fatigue, reducing shear transfer efficiency.
- The dowel-concrete interface stiffness significantly affects deflection profiles for slabs with normal subgrade support.
- The shear transfer efficiency increases with loss of subgrade support near the joint due to pumping action.
- Nighttime curling causes a loss of support near the joints, as well as widening of the joint. Truck traffic is typically heaviest at night, so nighttime curling in combination with wheel loading is a critical load case.
- During nighttime curling and the resulting loss of support at the joint, the far end of a slab lifts off the base as heavy wheel loads pass. This may help explain joint faulting.
- Daytime curling is not a critical loading, since the slab lifts off the subgrade at its center as opposed to the joints.

2.2.3 Zaghloul, White, and Kuczek (1994)

This study [96] focused on the determination of load equivalence factors (LEFs) by the use of a 3D, nonlinear dynamic finite element model developed using the commercially available finite element program ABAQUS. The model was designed to overcome the inherent limitations of 2D plate finite element models and Westergaard analysis.

The slab and subgrade were modeled using 3D brick elements, and the dowels were explicitly modeled with bar elements that were unbonded over half their length. Joints were modeled with gap elements, and assumed to have an initial opening of 3/8 inch; contact friction was considered if the gap closed during loading.

Concrete, granular subgrade, and clay subgrade materials were all modeled as different nonlinear materials. The concrete was modeled as bilinearly elastic-plastic up until failure, at which point the stress-strain curve descended bilinearly. The granular base, subbase, and

subgrade were idealized using an elastic-plastic Drucker-Prager model. Clays were modeled with a Cam-Clay model that accounted for deformation rates and both elastic and plastic deformations.

The model was verified for static loading by comparing computed deflections to those determined using Westergaard's equations and a separate finite element program. Various slab thicknesses and subgrades were considered; agreement was found to be good in all cases. The dynamic modeling capabilities of the program were verified by comparing computed slab response with measured field data.

Following model verification, the authors computed LEFs for various axle loads, axle configurations, and slab thicknesses. These LEFs were compared to the AASHTO LEFs [2], and a close correlation between the two was found. However, the authors felt that their LEFs offered significant advantages over the AASHTO LEFs since realistic material properties and dynamic effects were considered. Further, the LEFs computed by the authors are based on an empirical model, and thus can be extended to cover unusual loadings or other factors.

2.2.4 Chatti, Lysmer, and Monismith (1994)

In this investigation [13], the authors studied the effects of truck dynamics on the response of rigid pavement slabs. A linear dynamic finite element model (DYNA-SLAB) was developed to model the slab/subgrade system, and was essentially an extension of the existing static model ILLI-SLAB [82, 84] which uses plate elements to model the slab.

Aggregate interlock shear transfer between slabs was modeled with springs, and dowels were modeled with bar elements. The foundation was treated either as a Winkler foundation, or a layered viscoelastic medium resting on a rigid base or a semi-infinite half-space. Further, the authors developed an analytical method for determining the frequency dependent stiffness and damping coefficients used in the Winkler model. In this method, the analytical solutions for displacement and velocity response of a massless slab resting on a layered soil profile when subjected to a harmonic loading were determined. These values were then substituted into the integral equation of dynamic equilibrium expressed in terms of the stiffness and damping coefficients of the Winkler foundation, and the stiffness and damping coefficients were calculated.

When a frequency independent Winkler foundation was employed, Newmark's constant acceleration method was used for an explicit solution in the time domain. For the case of the frequency dependent Winkler model or a layered viscoelastic model, the solution was achieved in the frequency domain using a complex response method. The solutions were

verified by comparing the results to an approximate solution for a point load on a infinite plate by a Winkler foundation, and to the exact solution for a transient point load on a beam of finite length supported by a viscoelastic Winkler foundation. Good agreement was found in both cases. In addition, results from a U.S. Army Corps of Engineers study were compared to those computed using DYNA-SLAB, with good agreement. Further, the foundation models were verified by comparing computed deflection profiles due to harmonic loadings to those determined using a separate 3D soil-structure interaction program. In general, the layered foundation gave good results for all loading conditions, whereas the Winkler foundation model was in good agreement for interior deflections, but overpredicted edge deflections.

Following verification of the model, the authors investigated whether dynamic analysis is necessary for the prediction of rigid pavement response. Parameters included vehicle velocity and pavement roughness, and several different slab thicknesses and load transfer efficiencies were considered. The effect of vehicle velocity on slab bending stress was found to be negligible in all cases. To study the effect of pavement roughness, equivalent loads were developed for various slab profiles that included the effects of curling, faulting, and breaks of varying severity using a separate truck simulation program.

The authors concluded that while it is important to use expected pavement roughness to determine the peak magnitudes and locations for the truck wheel load history, the dynamic effects on the slab response are not significant. The authors caution, however, that the existence of a stiff layer or bedrock at a shallow depth may increase dynamic effects to the point that they should be incorporated in the analysis.

2.2.5 Uddin, Hackett, Joseph, Pan, and Crawley (1995)

This study [90] documented the development of a 3D finite element model of a concrete pavement and subgrade using the general-purpose finite element program ABAQUS. The goal of the study was to examine the effect of pavement discontinuities on surface deflections of a pavement subjected to a standard falling weight deflectometer load (FWD).

The slab, cement-treated base, and subgrade were modeled using 3D elastic brick elements. The following model parameters were investigated in a previous study and employed:

- The model was quarter symmetric with rollers on the lateral sides of the subgrade. One and one-half pavement slabs were modeled.

- The subgrade was modeled well beyond the edge of the pavement slab and shoulder.

- Discontinuous shoulders of 10' foot were modeled along the pavement edges, with gap elements employed between the shoulder and the pavement slab.

Pavement cracks (discontinuities) were modeled using gap elements. Dowels at the transverse joint were modeled with beam elements which were free to move relative to one slab, and contact between the two faces of the transverse joint was modeled with gap elements. A sensitivity study was done in which the effect of crack opening and friction coefficient on surface deflection was examined.

Back-calculation of the elastic moduli of the concrete slab, cement-treated base, and subgrade was done based on both static load-deflection data and FWD results for both cracked and uncracked pavement sections. As expected, significantly lower modulus values were obtained for cracked pavements as opposed to uncracked pavements, as well as for dynamic loading as opposed to static loading. Using the back-calculated moduli, the model was capable of accurately predicting the deflection response of the slab subjected to a FWD.

2.2.6 Kuo, Hall and Darter (1995)

In this report [54], the efforts of the research team to develop a 3D elastic finite element model of a rigid pavement system are documented. The purpose of the model was to analyze various factors affecting concrete pavement support, including base thickness and stiffness, interface bond friction, slab curling and warping due to temperature and moisture gradients, load transfer at joints via dowels and aggregate interlock, and the improved support gained with a widened lane, widened base, or tied concrete shoulder. The model was developed using the widely available general-purpose finite element program ABAQUS.

Significant effort was spent determining the best element type and the required level of mesh gradation. Both 4-noded and 8-noded slab elements were considered, as well as 8-noded and 20-noded brick elements. Convergence tests were performed for both interior and exterior loading cases, with comparisons made to both Westergaard's deflection solutions and the program ILLI-SLAB. In general, it was found that the 2D (plate element) and 3D finite element solutions were in good agreement for the limited range of problems for which plate elements are applicable, i.e. thin slabs with fairly large loaded areas. The use of the ABAQUS plate element which incorporates transverse shear effects reduced the effect of load size. It should be noted that the 8-noded brick element did not converge to Westergaard's solutions for interior and edge loading, even with a fine mesh and multiple layers of elements through the thickness of the slab. As a result, a single layer of quadratic

(20-noded) brick elements was used to model the slab.

The subgrade was modeled as a dense liquid (Winkler) foundation. Interface friction between the slab and subgrade was modeled using a membrane element coupled with a special interface element designed to model contact and frictional movement. The membrane element was required to provide compatibility between the interface element and a dense liquid foundation. An additional layer of 3D elements between the slab and subgrade.was used to model cement-treated bases.

The dowel bars were modeled as discrete beam elements that were allowed to slip relative to the slab. No attempt was made to model the interaction between the dowel bars and the concrete. Aggregate interlock shear transfer across transverse joints was modeled using spring elements.

Model verification was achieved by comparison with full-scale field test data, specifically the AASHO Road Tests [1], PCA tests on cement-treated bases [14], and the Arlington Road Test [87]. In general, the comparisons between the 3D model and the test results was good. The ability of the model to predict crack initiation locations was verified by comparing the principal stress contours generated by ABAQUS with the actual observed crack locations for various slab thicknesses and loading conditions consistent with the AASHO road tests. The PCA tests on cement-treated bases provided load-deflection data for interior and free edge loading conditions. In general, the model accurately predicted deflections, whereas the conventional 2D plate models and Westergaard's solutions gave less accurate results. Comparison of computed curling stresses with data obtained from the Arlington road test showed good agreement.

2.2.7 Zaman and Alvappillai (1995)

This study [97] examined the effect of moving aircraft loads on systems of multiple, jointed rigid pavement slabs. The pavement slabs were modeled using 4-noded, rectangular, thin plate elements, and the underlying soil was considered a viscoelastic Winkler foundation. The dynamic interaction of the aircraft and the slab system was incorporated in the analysis, with each landing gear modeled as a parallel spring and dashpot with an associated mass.

The longitudinal joints were considered to be keyed or cracked, and were modeled as discrete displacement springs. The transverse joints were modeled as doweled, and debonding and gaps between the dowels and pavement were considered explicitly; this study represents the first attempt to rigorously consider this contact nonlinearity. The constraint conditions between the dowels and the pavement were:

- No penetration of nodal points
- No tensional contact forces
- Coulomb friction assumed between the dowel and pavement in the longitudinal direction

An initial gap was specified at the free end of the dowel, and the nodal constraints were enforced using Lagrange multipliers. The dowels were meshed explicitly with the slabs (the dowel and slab nodes were coincident). Model verification studies showed significant losses in joint transfer efficiency with relatively small values for the dowel looseness (up to 0.127 mm), as well as a dowel normal force distribution closely matching analytical results for a beam on elastic foundation.

Extensive parametric studies were completed to determine the effects of dynamic vs. static loading, as well as the effects of dowel looseness on joint performance. Based on the model results, it was determined that in general, static loading conditions are more critical for determining pavement thickness. However, dynamic simulations that incorporated aircraft-pavement interaction produced larger losses in joint efficiencies due to dowel looseness than both static and simple moving loads.

2.2.8 Masad, Taha, and Muhunthan (1997)

Masad et al. [60] focused on the examination of temperature effects on rigid concrete pavements through the used of 3D finite element models. The models had the following characteristics:

- The longitudinal joints between the slabs and shoulders were modeled as keyed, and the transverse joints as grooves. Friction and contact at the joint surfaces were considered.
- Both the slab and foundation were idealized as linearly elastic, and the foundation was discretized to a fixed depth.
- Eight noded brick elements were used to model the slab and subgrade.
- Friction and loss of contact between the slab and base layer were considered in the analyses.

Both linear and nonlinear temperature gradients through the thickness of the slab and uniform temperature changes were considered. Using a model consisting of four slabs and two concrete shoulders with three elements through their thickness, parametric studies were performed to examine the effects of friction, slab lengths and thickness on curling and thermal expansion stresses. The effect of temperature variation on joint openings was also examined. Comparisons were made between the model results and those from similar analyses employing common 2D finite element codes, including ILLI-SLAB and JSLAB. Nonlinear temperature gradients were modeled with a single slab model having nine elements through its thickness. The slab stresses determined from the linear and nonlinear temperature variations were compared.

The main conclusions of the study were as follows:

- Reasonable agreement for maximum curling stresses was found between the model and previous analyses performed using 2D FE models.
- The maximum temperature curling stresses for the entire range of linear temperature gradient analyses were about 28% of the modulus of rupture of concrete.
- The effect of friction on curling stresses was determined to be negligible; slab length had a moderate effect, and increasing the slab thickness markedly increased curling stresses.
- Thermal expansion stresses were found to be significantly smaller than those predicted by subgrade drag theory, although they increased with friction and slab length.
- Nonlinear temperature distributions caused higher slab tensile stresses than linear temperature distributions in general.
- The joint openings predicted by the finite element models due to uniform temperature changes were in general less than those predicted using common design equations.

2.2.9 Kim, Hjelmstad, and Zuo (1997)

This study examined the response of a single rigid pavement slab to the heavy multiple-wheel loading of typical aircraft landing gear [49]. Three dimensional finite element analyses were performed using radially graded meshes and infinite elements to represent the vertical and lateral extents of the natural soils in an attempt to minimize the number of unknowns

(and thus computational requirements) without compromising accuracy. Vertical mesh refinement and load application were based on a parallel study [34] that specifically examined these aspects of 3D finite element modeling of rigid pavement systems.

The slab, cement-treated base, and subgrade layers were modeled with linear hexahedral elements. Models with both bonded and unbonded base layers were considered, and the cement treated base was treated as both linearly elastic and as a nonlinear material obeying a Drucker-Prager elastoplastic constitutive law. All other components of the system were treated as linearly elastic. Since only single slab systems were modeled, aggregate interlock and dowel joint shear transfer were not considered.

Three different load cases (center, edge, and corner) were considered and separate meshes refined appropriately for each loading were generated. Single, dual, and triple tandem axle loadings were applied for all cases to examine the effect of wheel interaction. In general, it was concluded that the maximum displacements were nearly proportional to the total load regardless of gear configuration. However, the maximum slab stresses were governed by the bending curvature of the slabs, which depends greatly on the wheel spacing, i.e. more closely spaced wheels produce larger slab stresses due to higher interaction between wheels. The critical load case was found to be edge loading with the longitudinal axis of the gear aligned with the slab edge; this case also exhibited the largest degree of wheel load interaction.

2.2.10 Brill, Hayhoe, and Lee (1997)

This manuscript [9] details the development of a general-purpose 3D finite element package for the modeling of rigid pavements. The computational engine is the public domain finite element package NIKE3D; the graphical user interface was developed independently by the authors.

In contrast with most 3D finite element models of rigid pavements, the slabs were modeled with 4-noded thin plate elements, while the subgrade layers were meshed with 8-noded linear hexahedra. Both aggregate interlock and dowel shear transfer at the joints were modeled with linearly elastic 8-noded hexahedra where the shear modulus was chosen to give a specified spring stiffness. Separation between the slab and subgrade was considered through frictionless nodal contact; frictionless sliding with no separation may also be modeled with the program, as well as perfect bond between the slab and subgrade. Only static analyses were considered.

Finite element analyses of typical systems with and without stabilized base layers sub-

jected to edge loading by either a single wheel, Boeing 777 landing gear, or 727 landing gear were performed. Each wheel in the landing gear was modeled explicitly, with mesh refinement done in the region of the load. The model degrees of freedom varied from about 81,000 to 92,000 depending on the load case. The stress results from the finite element analyses were compared with theoretical solutions for a slab on elastic foundation; significant differences in the two solutions were observed for most combinations of loading and subgrade type. To analyze the effect of the stabilized base layer on stress response, further runs were computed for the Boeing 777 landing gear assuming that the stabilized base layer was either continuous or cracked at the joint. It was found that a cracked base layer results in significantly higher slab stresses than an uncracked layer; stress load transfer efficiency is also reduced by cracking of the base layer.

2.3 Summary of Previous Work

Clearly, significant progress has been made in the finite element modeling of rigid pavement slabs. The original, 2D plate models developed in the early 1970's [80, 37] have evolved into 2D finite element packages designed specifically for the modeling of rigid pavement systems [82, 86]. Subsequently, these tools have been extended to allow research into the response of rigid pavements to applied static and dynamic loading (see [42, 44, 97] for example).

At present, it is generally accepted that 3D modeling techniques are necessary to accurately capture the response of rigid pavement systems. Large 3D models are commonly used by researchers to study the dynamic response of rigid pavement systems [96, 13, 90]; the effect of material nonlinearities [12]; for comparison with field test data [54]; and to identify critical loading conditions [12, 49]. Table 2.1 summarizes the salient features of the 3D finite element studies reviewed in this Chapter.

Despite the large number of analytical studies on the structural response of jointed concrete pavements, some significant aspects of the problem have been neglected. Specifically:

- As seen in Table 2.1, modeling of the dowels has been accomplished by explicitly meshing beam elements with the slab in previous 3D finite element investigations. Beyond permitting dowel slip, dowel-slab interaction has not been considered except by Channakeshava et al. [12], where a separate analysis of a single dowel and localized slab region was performed to compute a nonlinear spring stiffness to place between the dowel and the slab in the larger finite element model. Similarly, most 2D models

Table 2.1: Summary of Features of 3D Finite Element Studies

	Loading			Slab(s)			Subgrade			Base Layer(s)		Dowels	Agg. Inter.	
	Static	Dynamic	Temperature	Multiple	Linear	Nonlinear	Winkler	Linear Solid	Nonlinear Solid	Linear Solid	Nonlinear Solid	Beams	Springs	Friction
Ioannides [43]	x				x				x					
Channakeshava [12]	x		x	x		x	x					x		
Zaghloul [96]		x		x		x			x		x	x		x
Uddin[a] [90]		x		x	x			x		x		x		x
Kuo [54]	x		x	x	x		x			x		x	x	
Masad[b] [60]			x	x	x			x						x
Kim [49]	x				x			x			x			
Brill[c] [9]	x				x			x		x				

[a]Falling Weight Deflectometer loading.
[b]Nonlinear temperature gradients considered.
[c]Slabs modeled with plate elements.

account for both localized compression of the slab and dowel looseness by placing displacement springs between the ends of the slab and the dowel. However, for 2D models, the calculation of the appropriate spring stiffness has typically been accomplished by assuming the embedded portions of the dowel to act as a beam on a Winkler foundation as shown in Fig. 2.1. A rational determination of an appropriate Winkler modulus is difficult, with reported values in the literature varying by two orders of magnitude [45]. Dowel looseness (see Fig. 2.1) was explicitly considered only by the 2D analyses of Zaman et al. [97].

- Table 2.1 indicates that aggregate interlock load transfer has been modeled using linear spring elements [54], or assuming classical frictional behavior [96, 90, 60]. This approach does not capture the actual mechanism of aggregate interlock (see Chapter 5). While an effective spring stiffness may be back-calculated from field measurements of joint deflections or a target load transfer efficiency [44], it is effectively a

secant stiffness, valid only for one geometric configuration, set of material properties, and loading. No recent study has addressed the rational determination of load transfer across pavement joints due to aggregate interlock under a variety of loadings, geometries, and joint openings.

- Previous 3D models have been used only in research applications due to computational requirements; realistic, practical solutions of 3D models in design settings are not feasible at present. This is due partially to the efficiency of the solution routines in available finite element programs for the 3D finite element modeling of structural systems.

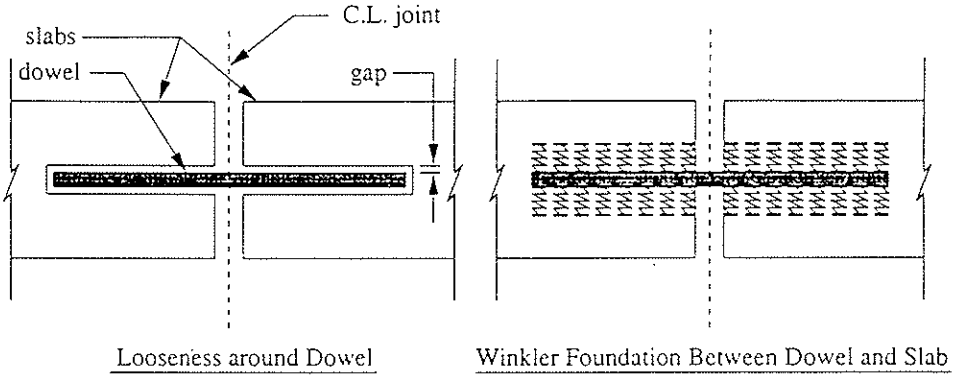


Figure 2.1: Dowel Looseness and Winkler Foundation Around Dowel

These three specific shortcomings of previous investigations are the main research topics of this study. The first two items are of particular importance in the design of rigid pavements, since many failures and shortened pavement lives can be attributed to joint problems caused at least in part by inadequate load transfer. Indeed, many undoweled concrete pavements are commonly retrofitted with dowels in an attempt to lengthen their service life and improve performance [32, 81]. Further, one significant 3D finite element study reviewed in Chapter 2 notes that poor joint load transfer has predictable detrimental effects on pavement behavior [12].

Addressing the third issue is of broad importance, as large, 3D models of any structural system require tremendous computer resources to achieve a reliable solution. Several of the models cited in this literature review employed relatively coarse meshes [12, 54] because of computational limitations. More efficient solution strategies will ease the problem of mesh refinement, as well as making the proposed joint modeling techniques accessible to more pavement researchers and designers to aid in their investigations.

Chapter 3

Finite Element Model Development

3.1 Introduction

The literature review presented in Chapter 2 indicates that three-dimensional (3D) finite element models are required to accurately capture the structural response of rigid pavement systems. Further, three main shortcomings of previous investigations were identified: (1) present methods for modeling dowel load transfer are inadequate; (2) aggregate interlock shear transfer has not been properly quantified; and (3) presently available solution strategies severely limit the size of the 3D finite element models that can be solved on the desktop computers and workstations commonly available to researchers and designers.

These three specific shortcomings of previous models and studies are the main research topics of this study, and are individually addressed in Chapters 4-6. However, implementation of these joint modeling techniques and solution strategies necessitates the development of general, 3D finite element models of rigid pavement systems. This requires that the following issues be resolved:

- Static vs. dynamic modeling.
- Modeling of the slab(s), including selection of element type and appropriate constitutive relations.
- Modeling of the subgrade and appropriate boundary conditions. The variable vertical and lateral extents of the subgrade make the choice of realistic boundary conditions difficult [43].
- Inclusion of both wheel and temperature loading in the finite element models.

- Loss of contact between the slab and subgrade due to temperature and wheel loading. This can lead to critical loadings as noted in Channakeshava et al. [12].

The remainder of this Chapter addresses these issues.

3.2 Modeling Issues

3.2.1 Dynamic vs. Static Modeling

The literature review in Chapter 2 indicates that several researchers have begun to use dynamic models of rigid pavement systems. However, only static models are employed in the present investigation. This restriction does not represent a severe limitation, and is justifiable for several reasons:

- The development and implementation of new joint shear modeling strategies should first be accomplished with static models that will allow detailed evaluation of their performance and applicability.

- Rigid pavement systems are typically designed assuming static equivalent single axle loads (ESALs) which attempt to account for damage and fatigue due to mixed traffic over the expected lifetime of the structure. The proposed models are not limited to any one type of axle or load magnitude, and any axle loading corresponding to an actual truck or appropriate design axle may be applied.

- One recent study has shown that the dynamic response of the slab may not be significant, and as long as the dynamic interaction between the slab and truck are accounted for in developing the effective wheel loads, static models are reasonably accurate [13]. Such wheel load histories may be generated with an independent model that incorporates the pavement profiles, and the stiffness and damping properties of the truck suspension, or via actual field data.

- The computational requirements of large, 3D dynamic models are significant. At present, implicit time stepping methods are far too computationally expensive for the proposed models, since they require solution of the system of equations arising from the system discretization at each time step. While less expensive explicit methods are available, the stability limitations of these methods can make them computationally inefficient as well.

It should also be noted that the solution of dynamic FE simulations requires only the selection and implementation of an appropriate time stepping scheme, for the which the static solver serves as the core computational unit. If dynamic simulations are to be a topic of future study, the nonlinear solution techniques developed in this study could serve as building blocks for the development of a dynamic modeling package.

3.2.2 Modeling of the Slab

The slabs are modeled as linearly elastic. While several researchers have incorporated nonlinear slab material models, this is not necessary to develop realistic rigid pavement models for the prediction of joint shear transfer. In fact, with the exception of the material immediately surrounding the dowels, rigid pavement slabs remain in a low stress, linearly elastic range under most common axle loadings; slab cracking and damage are usually the result of repeated loading and fatigue. Appropriate published values for the elastic properties of slab concrete are available and typically known to design practitioners, while nonlinear models usually require the specification of several parameters which may not be readily available.

The slab boundaries coincide with the physical slab boundaries; see Fig. 3.1 for an example of the discretized slabs in a typical two-slab model. The generality of the finite

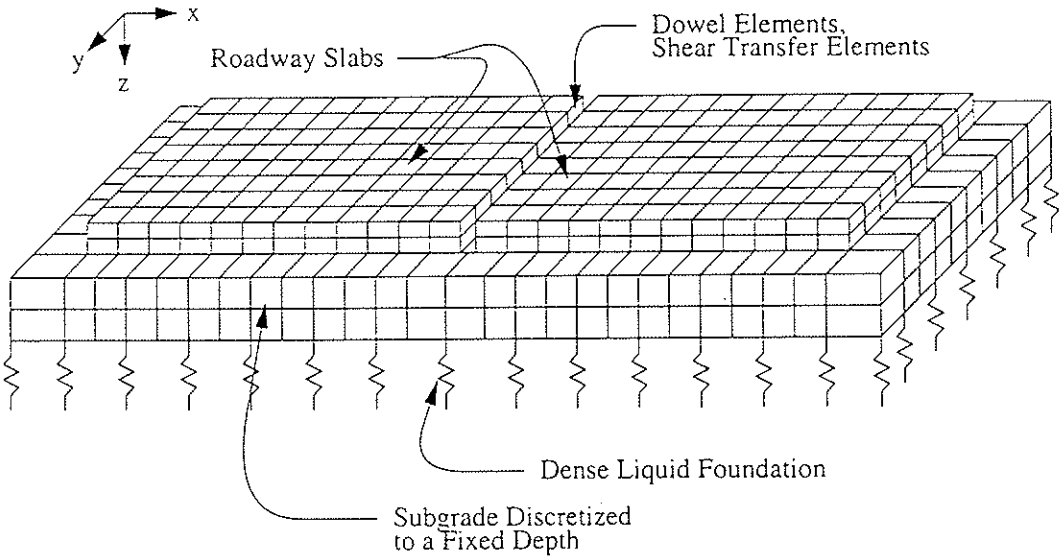


Figure 3.1: Three-Dimensional Model of Two Rigid Pavement Slabs

element method allows the modeling of skewed joints and as many slabs as required to capture loading effects.

3.2.3 Modeling of the Base Layers and Subgrade

Finite element models of rigid pavement systems consisting of one or more slabs require a reasonable representation of the subgrade. Several researchers have used continuum models for the entire subgrade [43, 96, 90]; however, this dictates that the subgrade be modeled to a depth where constant vertical strain exists, which may be quite large. This is often done in spite of the fact that the subgrade stresses decrease quite quickly with depth [43] to a point where a continuum model may not be required, unnecessarily increasing the size of the model. Another approach to modeling the subgrade is to use a dense liquid (Winkler) foundation directly below the slab or treated base layer to model the infinite vertical extent of the subgrade [54, 12]. While this greatly reduces computational complexity, it may not accurately model foundation response due to the lack of shear coupling between the discrete foundation springs generated with the dense liquid foundation model.

In the present study, the foundation model varies depending on the system being studied. In general, any number of linearly elastic base/soil layers can be modeled to any depth below the slab. Below the bottom-most layer, a Winkler foundation is used recognizing that foundation stresses at this depth are low and the use of a simplified model will have little effect on the slab response. The base layers can be extended laterally to account for extended shoulders if required as shown for a two-slab system in Fig. 3.1. Note that modeling the soil as linearly elastic is not entirely realistic. However, while nonlinear response of the subgrade is likely under heave axle loads, subgrade yielding may not greatly affect computed slab displacements and stresses [43]. The development of 3D constitutive relations for soils appropriate for use in the modeling of rigid pavements is an active area of research [35]. As better, more reliable models become available, they may be included in future analyses.

3.2.4 Discretization of the Slab and Subgrade

The regular geometry of the system to be modeled lends itself to hexahedral elements. Quadratic elements, while somewhat more difficult to implement than simpler linear elements, have excellent convergence rates and are more robust. Further, other researchers have explicitly demonstrated the applicability of quadratic hexahedra for modeling rigid pavement systems [54]. For these reasons, 20-noded quadratic hexahedra are used to model both the slab and the subgrade. An isoparametric element formulation is used to permit complex geometries such as skewed joints to be rigorously considered; the element stiffness matrices are computed using $3 \times 3 \times 3$ Gaussian integration. See Zienkiewicz and Taylor [98]

for details of the element formulation. The Winkler foundation models below the subgrade are implemented with an 8-noded quadratic interface element, which is compatible with the quadratic hexahedra.

3.2.5 Wheel, Temperature and Self-Weight Loading

The three major load effects on rigid pavements are wheel, temperature and self-weight loadings. Note that the slab self-weight is significant when applied in conjunction with wheel and/or temperature loads due to separation of the slab and subgrade. All three load effects are considered in this study; details of their inclusion are given in this section.

Wheel Loading

External load effects are typically applied within a finite element model by either point loads applied at nodes, or pressures (usually uniform) distributed over an element face. However, point loads are unrealistic, as wheel loads are distributed over the finite area of the tire footprint. Further, the use of a pressure load over an element face requires that an element face coincide exactly with the tire footprint.

To overcome these difficulties, individual wheel loads are applied as uniform rectangular pressure distributions independently of the slab mesh as shown in Fig. 3.2, which depicts a portion of the top surface of a typical slab. Determination of the equivalent nodal load

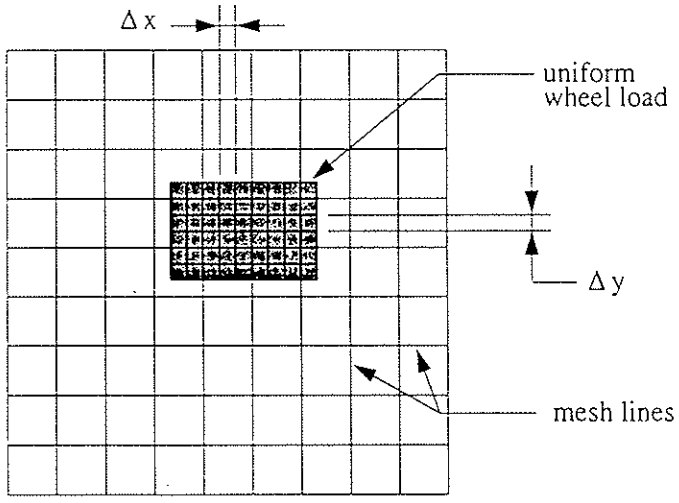


Figure 3.2: Application of Wheel Loads

vector, $\mathbf{p}$, requires summation of the integral of the product of the transpose of the element shape function matrix, $\mathbf{N}_i{}^{\mathrm{T}}$ by the pressure vector, $\mathbf{f}$ over the all loaded elements $i = 1 \ldots l$

[98]:

$$\mathbf{p} = \sum_{i=1}^{l} \int_{A_i} \mathbf{N}_i^T \mathbf{f} dA_i \tag{3.1}$$

In Eq. 3.1, Σ represents assembly and summation.

Evaluation of Eq. 3.1 may be accomplished using a rectangular rule where the wheel load is represented by n small point loads, each equal to the pressure acting over the area $\Delta A = \Delta x \Delta y$ as shown in Fig. 3.2. The evaluation of Eq. 3.1 is then reduced to:

$$\mathbf{p} \approx \sum_{j=1}^{n} \mathbf{N}_{ij}^{\mathrm{T}} \mathbf{f}_j \Delta A \tag{3.2}$$

where the subscript i has been retained as a reminder that for each ΔA, the element it lies within must be determined, and $\mathbf{N}_{ij}$ is the usual matrix of shape functions for the i^{th} element evaluated at the the j^{th} point. Again, Σ implies both assembly and summation.

Determination of the i^{th} element within which the j^{th} point lies is accomplished with an efficient grid searching algorithm developed for use with the multigrid solution strategies. The grid search relies on grouping all elements in the finite element mesh according to their location in the model, greatly reducing the number of elements that must be searched to determine which element any (x, y, z) point lies within. Calculation of the local element coordinates corresponding to the j^{th} point is done using Newton's method at the element level. Details may be found in Chapter 6 and Appendix A.

It should be noted that another approach to the evaluation of 3.1 would be to determine which elements are fully loaded, integrate that portion of Eq. 3.1 by conventional integrations over entire element faces, and approximately evaluate the remainder of the integral over the partially loaded elements. This approach was recently outlined by other researchers in [34].

Temperature and Self-Weight Loading

Temperature gradients through the thickness of the slab are treated as general prestrains on an element-by-element basis [98]. This allows an exact mathematical treatment of linear variations in temperature through each element, as the 20-noded brick captures linear variations in strain. Note that a model with multiple elements through the slab thickness allows the approximation of a nonlinear temperature gradient as a piecewise linear function. While nonlinear temperature gradients are not considered in this study, several researchers have noted their occurrence [60], and the ability to model this phenomenon may prove useful in future studies.

Self-weight is treated in the usual fashion as a uniformly distributed body force in the vertical direction. The equivalent nodal loads are determined element-by-element and mapped to the global vector of external forces. Self-weight is included in all analyses, as it induces additional stresses in the slab when the slab and base layer separate.

3.2.6 Separation of the Slab and Base, and Boundary Conditions

Modeling the loss of contact between the slab and the subgrade is critical when considering temperature-induced curling and wheel loading at the joint (Fig. 3.3). This loss of contact

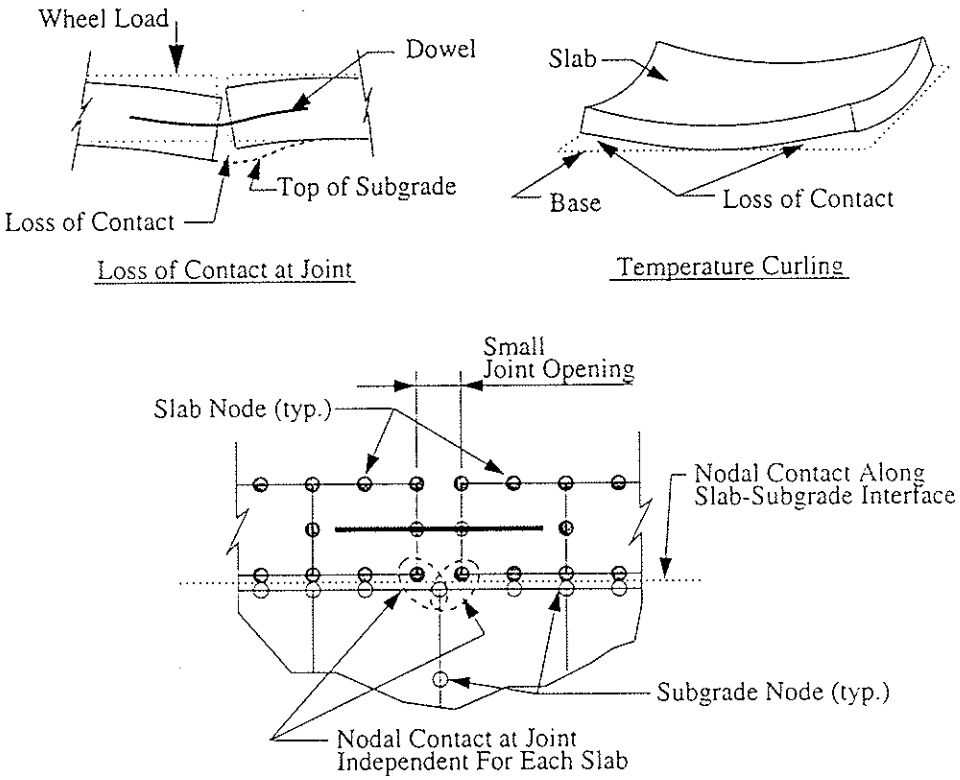


Figure 3.3: Loss of Contact between Slab and Subgrade

is modeled using a nodal contact approach, which requires the enforcement of inequality constraints for pairs of adjacent nodes. Near the joint, the bottom nodes of both slabs at the joint are independently constrained to or released from a single subgrade node depending on relative vertical displacements and stresses as shown in Fig. 3.3. This detail is required to allow the relative vertical displacements of the loaded and unloaded slab to be captured. Contact conditions may also need to be considered on the vertical joint face for small joint

openings.

The implementation of nodal contact modeling within a finite element model introduces inequality constraints, requiring a complex solution strategy. The complexity is heightened by the use of iterative solvers, as will be employed in this study. Details of the implementation are thus deferred until Chapter 6, where a comprehensive treatment of the nonlinear solution strategies employed in this study is presented.

Ideally the slab boundary conditions would include the effect of friction between the slab and subgrade. However, for the purposes of joint shear transfer modeling, simple displacement boundary conditions are sufficient. The appropriate boundary conditions are the minimum required to keep the model stable, i.e. prevent transverse displacements and rotation about the vertical axis. The models developed in this study use a single (x, y) displacement restraint on one side of each slab and a single x direction displacement constraint on the other side. The same constraints are applied to the base layers and subgrade as a unit; friction between the slab and base is not considered in the analyses.

3.3 Summary

The need for better finite element modeling of dowel and aggregate interlock joint shear transfer and more efficient solution strategies has been identified. Addressing these issues requires the development of specific finite element techniques appropriate for the modeling of rigid pavement systems. This Chapter has covered modeling of the slab and subgrade, incorporation of boundary conditions, slab-subgrade loss of contact, and methods for applying both temperature and wheel loadings. The next three Chapters focus individually on dowel joint shear transfer, aggregate interlock shear transfer, and nonlinear solution strategies.

Chapter 4

Dowel Modeling

4.1 Review of Current Dowel Modeling Techniques

Several different approaches have been used to model load transfer between rigid concrete paving slabs via dowel action. The actual interaction of the dowel and the slab is complex, but in general consists of two portions:

- Looseness (gaps) between the dowel and the slab
- Compression of the slab around the dowel

Laboratory research by several investigators has shown that dowel looseness has a significant effect on joint performance [10, 81]. Despite this concern, only one 2D finite element study employing plate elements has explicitly considered this phenomenon [97]. A recent study examines this problem with contact modeling using a 2D plane strain model [69], and demonstrates the potential for undesirable effects on pavement response.

Most attempts to model dowel-slab interaction approximately account for compression of the the slab around the dowel by assuming a Winkler foundation between the dowel and surrounding slab. Indeed, this fundamental approach was in use prior to the development of finite element analysis, with early models based on the explicit solution for an infinitely long beam resting on an elastic foundation [28]. With the advent of finite element analysis, investigators have typically modeled dowel bars with discrete beam elements between adjacent slabs having springs between the dowels and the slabs to account for the effect of dowel-slab interaction, as in ILLI-SLAB [82, 84] and JSLAB [86]. The spring stiffnesses are derived by considering the embedded portion of the dowel as an infinitely long beam on a Winkler foundation.

More recent dowel models have also been proposed which more rigorously incorporate the effect of dowel-slab interaction in addition to the beam deformation of the dowel between adjacent slabs; a detailed description of the formulations and limitations of the more recent models is given by Guo [31]. Guo developed a component dowel bar model that consistently incorporates a Winkler foundation model for embedded portions of finite length and a discrete shear beam element spanning the joint into a single element.

Of course, the Winkler foundation modulus can be reduced to account for gaps between the dowel and the slab. While this is computationally attractive, it is not always clear what modulus to use to model the interaction of the slab and the dowel. Similar models that reduce the Winkler modulus nonlinearly in a more rational manner, such the approach developed by Sittipunt [79] for the modeling of reinforcement at the base of concrete shear walls, require the determination of several parameters in addition to the spring stiffness which may not be readily determined.

As noted in Chapter 2, 3D finite element models generally employ beam elements meshed explicitly with the solid elements comprising the pavement slab and do not attempt to model dowel-slab interaction. An approach designed to remove the difficulty of meshing beam elements with plate or solid elements was developed by Ioannides and Korovesis [45], where a uniform joint shear stiffness is used to replace the individual dowels. This approach was employed by Hammons [33] in the development of 3D finite element models of scale model laboratory tests on doweled pavements. Implicit in the derivation of the joint stiffness, however, is the common assumption that the embedded portion of the dowel may be represented as a beam with a Winkler foundation sandwiched between it and the surrounding concrete. Again, the difficulty lies in determining an appropriate modulus of reaction for the Winkler foundation that properly accounts for both dowel/slab interaction and any dowel looseness that may exist. This model also assumes uniformly spaced dowels along the joint, which is common for newly constructed pavements but rarely the case for retrofitted joints where dowels are often located only in the wheelpaths. No 3D finite element models have explicitly considered the effect of dowel looseness on joint performance.

In this Chapter, a new technique for modeling dowel load transfer is proposed and developed which relies on an embedded formulation of the dowel, and allows the explicit and rigorous consideration of gaps between the dowel and the slab through nodal contact modeling. The necessary details for the inclusion of a general bond-slip law are also presented, as the dowel element has potential application in the modeling of many reinforced concrete

structures. Although the effect of compression of the slab around the dowel is not addressed explicitly, a linear or nonlinear bond-slip law would easily permit this effect to be considered in future studies.

4.2 Overview of Dowel Modeling Technique

The first issue which must be addressed is the choice of an appropriate representation of the dowels in the 3D finite element model. The finite element modeling of reinforced concrete structures indicates two practical methods to realistically incorporate the dowels.

- *Discrete representations.* The straightforward way of modeling the dowels is as discrete line elements in a global mesh. Unfortunately, this is at odds with the desire to discretize the concrete slabs—due to their regular geometry—using a mesh of regular solid brick elements. Modeling the dowel bars as discrete line elements requires that the dowels and the solid elements share common nodes, placing undesirable geometric constraints on the finite element mesh. This problem is exacerbated by the need to generate several meshes of the same physical system when using multigrid solution strategies as detailed in Chapter 6.

- *Embedded representations.* Embedded dowel elements (see Fig. 4.1) have the advan-

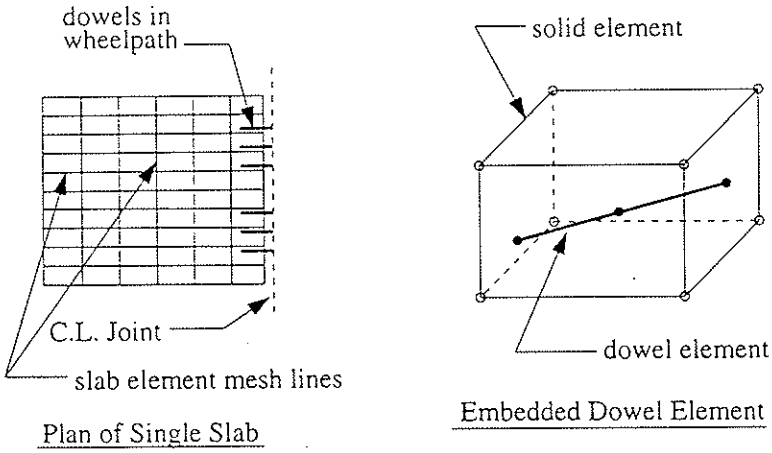


Figure 4.1: Embedded Dowel Element

tages of maintaining the exact mathematical representation of the line elements and encapsulating them in a regular solid embedding element. This allows exact location of the dowels while permitting the slab to be meshed independently of the dowel lines as shown in Fig. 4.1. Embedded formulations for both conventional and prestress-

ing axial-only reinforcement in finite element models of reinforced-concrete structures have been employed by several researchers [21, 95, 78]. However, these formulations typically require special integration techniques to compute the stiffness matrices of elements containing embedded reinforcing members.

Clearly, it is desirable to use an embedded formulation for the dowel elements similar to that developed for axial-only reinforcing, allowing the meshing of the slabs to be independent of the dowel locations. One of the major goals of this study is the development of a general formulation for an arbitrarily oriented, embedded, quadratic bending element. Provisions are made for axial debonding of the dowel, which is typical in rigid pavement slabs. Further, the formulation is extended to model gaps between the dowel and the surrounding slab which occur due to damage of the slab concrete under repeated loading. To model the portion of the dowel between the two slabs, a conventional two-noded shear beam is used that is constrained at its ends to displace compatibly with the elements it spans between.

The remainder of this Chapter is dedicated to the presentation of the dowel modeling techniques employed in this study. A brief description of the dowel element is presented in Section 4.3. In Section 4.4, the stiffness matrix of the embedded dowel is developed for three specific conditions: perfectly bonded and transversely constrained, debonded and transversely constrained, and debonded with gaps between the embedding element and the dowel. Section 4.5 examines how the dowel element could be extended to include a general bond-slip law in lieu of an explicit consideration of dowel looseness. This is followed in Section 4.6 by the development of the stiffness matrix of a shear beam element spanning the joint and compatible with the elements on either side of the joint. Section 4.7 details the constraint updating strategy required when dowel looseness is considered. Verification of the element formulation is achieved with a simple model problem in Section 4.8. Section 4.9 summarizes the dowel modeling techniques developed in this Chapter.

4.3 The Quadratic Beam Element

4.3.1 Element Definition

Higher order shear and bending elements have been in use for many years [40, 6], with more recent applications focusing on the modeling of curved structures [48]. Initially, the use of these elements was limited by their tendency to "lock", or fail to converge with increasing levels of mesh refinement. An excellent explanation of this phenomenon and proposed solutions are given by Prathap and Bhashyam [72]. It is demonstrated that the

phenomenon of locking may be explained in terms of spurious constraints arising when shape functions of the same order are used to interpolate both displacements and cross-sectional rotations.

For the study presently being conducted, a three noded, straight beam element is used to model dowels embedded in a mesh of quadratic solid elements (see Fig. 4.2). The shape

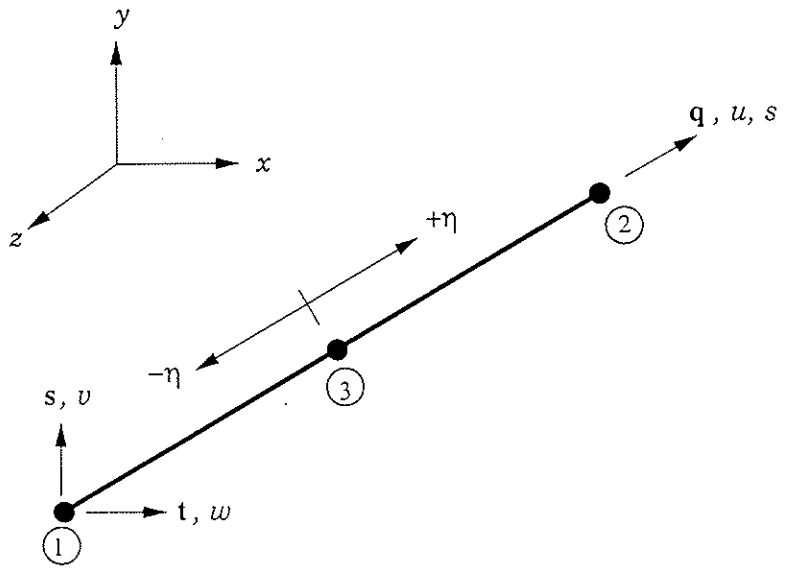


Figure 4.2: Dowel Element Local Coordinates and Unit Vectors

functions used to define the element displacements, N_i, are quadratic: those used to define the element cross-sectional and torsional rotations, Φ_i, are linear. As shown in [72], this choice of shape functions prevents locking. Any scalar displacement component, $d(\eta)$ or rotation, $\theta(\eta)$ may then be expressed as a combination of nodal values:

$$d(\eta) = \sum_{i=1}^{3} N_i(\eta) d_i \tag{4.1}$$

$$\theta(\eta) = \sum_{i=1}^{3} \Phi_i(\eta) \theta_i \tag{4.2}$$

Quadratic mapping of the element coordinates is assumed, which is isoparametric with respect to the displacement shape functions. Definitions for the shape functions may be found in most standard finite element texts (see [6], for example).

4.3.2 Unembedded Dowel Element Stiffness Matrix

The derivation of the 3D element stiffness matrix may be accomplished using virtual work principles. The strain energy contributions from axial displacements, shear displacements,

torsional rotation, and bending rotations can be written as:

$$SE_{du} = \frac{1}{2}\int_0^l EA\left(\frac{du}{ds}\right)^2 ds \tag{4.3}$$

$$SE_{dv} = \frac{1}{2}\int_0^l GA_v\left(\theta_w - \frac{dv}{ds}\right)^2 ds \tag{4.4}$$

$$SE_{dw} = \frac{1}{2}\int_0^l GA_w\left(\theta_v + \frac{dw}{ds}\right)^2 ds \tag{4.5}$$

$$SE_{tu} = \frac{1}{2}\int_0^l GJ(\theta_u)^2 ds \tag{4.6}$$

$$SE_{bv} = \frac{1}{2}\int_0^l EI_v\left(\frac{d\theta_v}{ds}\right)^2 ds \tag{4.7}$$

$$SE_{bw} = \frac{1}{2}\int_0^l EI_w\left(\frac{d\theta_w}{ds}\right)^2 ds \tag{4.8}$$

Evaluation of these integrals is most easily performed with respect to the dowel local coordinate, η. Referring to Fig. 4.2, we may write an incremental vector along the bar, $\mathbf{ds}$, as:

$$\mathbf{ds} = dx\mathbf{i} + dy\mathbf{j} + dz\mathbf{k} = \frac{dx}{d\eta}d\eta\mathbf{i} + \frac{dy}{d\eta}d\eta\mathbf{j} + \frac{dz}{d\eta}d\eta\mathbf{k} \tag{4.9}$$

where $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ refer to the usual unit vectors in the global coordinate system. The incremental distance along the bar is then computed as:

$$ds = \sqrt{\left(\frac{dx}{d\eta}\right)^2 + \left(\frac{dy}{d\eta}\right)^2 + \left(\frac{dz}{d\eta}\right)^2}\, d\eta = h d\eta \tag{4.10}$$

Using Eqs. 4.1 to 4.2, the required derivatives of the global coordinates with respect to the dowel local coordinate may be computed as:

$$\begin{bmatrix} \frac{dx}{d\eta} \\ \frac{dy}{d\eta} \\ \frac{dz}{d\eta} \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{bmatrix} \begin{bmatrix} \frac{dN_1}{d\eta} \\ \frac{dN_2}{d\eta} \\ \frac{dN_3}{d\eta} \end{bmatrix} \tag{4.11}$$

Note that for a straight bar with a centrally located third node, the value of h is constant and equal to half the length of the dowel.

The vector of nodal displacements and rotations may be written as:

$$\mathbf{U}^d = \begin{bmatrix} \mathbf{d}_1 & \Theta_1 & \mathbf{d}_2 & \Theta_2 & \mathbf{d}_3 & \Theta_3 \end{bmatrix}^{\mathrm{T}} \tag{4.12}$$

where $\mathbf{d}_i$ and Θ_i refer to generalized nodal displacement and rotation vectors; the superscript T denotes a transpose. Using the displacement and rotation basis functions developed in the previous section, the generalized displacement and rotation vectors at any point may

be expressed as functions of the dowel local coordinate, η:

$$\mathbf{d}(\eta) = \mathbf{N}(\eta)\mathbf{U}^{\mathbf{d}} \tag{4.13}$$

$$\Theta(\eta) = \Phi(\eta)\mathbf{U}^{\mathbf{d}} \tag{4.14}$$

where:

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \end{bmatrix} \tag{4.15}$$

$$\Phi = \begin{bmatrix} 0 & \Phi_1 & 0 & \Phi_2 & 0 & \Phi_3 \end{bmatrix} \tag{4.16}$$

Using these relations, the individual element displacements and rotations at any point may be expressed as:

$$u(\eta) = \mathbf{q} \cdot \mathbf{d}(\eta) \tag{4.17}$$

$$v(\eta) = \mathbf{s} \cdot \mathbf{d}(\eta) \tag{4.18}$$

$$w(\eta) = \mathbf{t} \cdot \mathbf{d}(\eta) \tag{4.19}$$

$$\theta_u(\eta) = \mathbf{q} \cdot \Theta(\eta) \tag{4.20}$$

$$\theta_v(\eta) = \mathbf{s} \cdot \Theta(\eta) \tag{4.21}$$

$$\theta_w(\eta) = \mathbf{t} \cdot \Theta(\eta) \tag{4.22}$$

Substituting Eqs. 4.13 through 4.22 into the expressions for the strain energy given by Eqs. 4.3 through 4.8 and changing variables from s to η yields:

$$SE_{du} = \frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}\frac{EA}{h}\left(\frac{d\mathbf{N}^{\mathrm{T}}}{d\eta}\right)\left(\frac{d\mathbf{N}}{d\eta}\right)d\eta\,(\mathbf{q}\otimes\mathbf{q})\right]\mathbf{U}^{d} \tag{4.23}$$

$$SE_{dv} = \frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}GA_{sv}h\Phi^{\mathrm{T}}\Phi d\eta\,(\mathbf{t}\otimes\mathbf{t})\right]\mathbf{U}^{d} - \tag{4.24}$$

$$\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{l}GA_{sv}\Phi^{\mathrm{T}}\left(\frac{d\mathbf{N}}{d\eta}\right)d\eta\,(\mathbf{t}\otimes\mathbf{s})\right]\mathbf{U}^{d} + \tag{4.25}$$

$$\frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}\frac{GA_{sv}}{h}\left(\frac{d\mathbf{N}^{\mathrm{T}}}{d\eta}\right)\left(\frac{d\mathbf{N}}{d\eta}\right)d\eta\,(\mathbf{s}\otimes\mathbf{s})\right]\mathbf{U}^{d} \tag{4.26}$$

$$SE_{dw} = \frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}GA_{sw}h\Phi^{\mathrm{T}}\Phi d\eta\,(\mathbf{s}\otimes\mathbf{s})\right]\mathbf{U}^{d} - \tag{4.27}$$

$$\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{l}GA_{sw}\Phi^{\mathrm{T}}\left(\frac{d\mathbf{N}}{d\eta}\right)(\mathbf{s}\otimes\mathbf{t})\right]\mathbf{U}^{d} + \tag{4.28}$$

$$\frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}\frac{GA_{sw}}{h}\left(\frac{d\mathbf{N}^{\mathrm{T}}}{d\eta}\right)\left(\frac{d\mathbf{N}}{d\eta}\right)d\eta\,(\mathbf{t}\otimes\mathbf{t})\right]\mathbf{U}^{d} \tag{4.29}$$

$$SE_{tu} = \frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}\frac{GJ}{h}\left(\frac{d\Phi^{\mathrm{T}}}{d\eta}\right)\left(\frac{d\Phi}{d\eta}\right)d\eta\,(\mathbf{q}\otimes\mathbf{q})\right]\mathbf{U}^{d} \tag{4.30}$$

$$SE_{bv} = \frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}\frac{EI_v}{h}\left(\frac{d\Phi^{\mathrm{T}}}{d\eta}\right)\left(\frac{d\Phi}{d\eta}\right)d\eta\,(\mathbf{s}\otimes\mathbf{s})\right]\mathbf{U}^{d} \tag{4.31}$$

$$SE_{bw} = \frac{1}{2}\mathbf{U}^{d\mathrm{T}}\left[\int_{-1}^{1}\frac{EI_w}{h}\left(\frac{d\Phi^{\mathrm{T}}}{d\eta}\right)\left(\frac{d\Phi}{d\eta}\right)d\eta\,(\mathbf{t}\otimes\mathbf{t})\right]\mathbf{U}^{d} \tag{4.32}$$

Taking the first variation of the strain energy with respect to $\mathbf{U}^d$ gives the total dowel stiffness matrix, $\mathbf{K}^d$, as the sum of the symmetric bracketed expressions and the symmetric part of the unsymmetric bracketed expressions. The integrals are computed numerically using two-point Gaussian quadrature, which is exact for a straight element.

4.4 Embedment of the Dowel

In this section, a procedure for embedding a dowel element within a solid, quadratic, isoparametric element is developed. In the following discussion, *bonding* will refer to axially constraining the dowel to the embedding element, and *constraining* will refer to the constraint of transverse dowel displacements to the embedding element. Three types of embedment are considered: (1) the case of a bonded, constrained embedded dowel is developed first to illustrate the embedding procedure; (2) the case of a debonded, constrained dowel, which has great practical applicability to the doweled systems being studied is then presented; and (3) the geometrically nonlinear case where gaps exists between the dowel and the embedding element is developed. Embedding the dowel requires that the locations of the dowel nodes in terms of the embedding element local coordinates be known. Procedures for determining the element local coordinates (ξ, η, ζ) from the nodal (x, y, z) coordinates using Newton's method are presented in [21] and [5]; a similar technique is used in this study. Details of the implementation may be found in Appendix A.

4.4.1 The Bonded, Constrained Case

Constraining the i^{th} node of a bonded dowel to an embedding element requires that the dowel nodal displacements be identical to the corresponding displacements of the embedding element at the node. Using the embedding element basis functions and nodal displacements, the i^{th} node of an unbonded dowel can be constrained to an embedding element in the following manner:

$$\mathbf{d}_i = \mathbf{N}_i^e\mathbf{U}^e \tag{4.33}$$

where $\mathbf{U}^e$ is the vector of nodal displacements of the embedding element, and $\mathbf{N}_i^e$ is the array of embedding element basis functions evaluated at the i^{th} dowel node.

If the vector of nodal displacements for the embedded dowel is expanded to include the displacements of the embedding element:

$$\mathbf{U}^{de} = \left[\begin{array}{cccccc} \mathbf{d}_1 & \Theta_1 & \mathbf{d}_2 & \Theta_2 & \mathbf{d}_3 & \Theta_3 & \mathbf{U}^e \end{array} \right]^{\mathrm{T}} \tag{4.34}$$

$\mathbf{U}^{de}$ may be transformed back to $\mathbf{U}^d$ as follows:

$$\mathbf{U}^d = \mathbf{T}\mathbf{U}^{de} \tag{4.35}$$

The transformation matrix, $\mathbf{T}$ is defined as:

$$\mathbf{T} = \begin{bmatrix} \mathbf{a}_1 & 0 & 0 & 0 & 0 & 0 & \mathbf{b}_1 N_1^e \\ 0 & \mathbf{I} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{a}_2 & 0 & 0 & 0 & \mathbf{b}_2 N_2^e \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{a}_3 & 0 & \mathbf{b}_3 N_3^e \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} & 0 \end{bmatrix} \tag{4.36}$$

where:

$$\mathbf{a}_i = \text{three dimensional second order zero tensor} \tag{4.37}$$

$$\mathbf{b}_i = \text{three dimensional second order identity tensor} \tag{4.38}$$

$$\mathbf{I} = \text{three dimensional second order identity tensor} \tag{4.39}$$

The definitions of $\mathbf{a}_i$ and $\mathbf{b}_i$ are trivial here, but are extended later to more complex cases. Note that $\mathbf{T}$ explicitly incorporates Eq. 4.33 on a node-by-node basis for the entire dowel, enforcing the constraint that the dowel nodal displacements be equal to the displacements of the embedding element at the dowel nodal locations. The identity matrix, $\mathbf{I}$, ensures that the dowel retains its nodal degrees of freedom.

It follows immediately from the previous derivation of the element stiffness matrix that the stiffness matrix of the embedded dowel element, $\mathbf{K}^{de}$, may be calculated as:

$$\mathbf{K}^{de} = \mathbf{T}^{\mathrm{T}}\mathbf{K}^d\mathbf{T} \tag{4.40}$$

The vector of embedded dowel nodal forces, $\mathbf{F}^{de}$, may be computed from the vector of unembedded dowel nodal forces $\mathbf{F}^d$ by:

$$\mathbf{F}^{de} = \mathbf{T}^{\mathrm{T}}\mathbf{F}^d \tag{4.41}$$

Note that the transformed stiffness matrix, $\mathbf{K}^{de}$, will have rows and columns corresponding to the dowel nodal displacements that are comprised entirely of zeros. These arise from

the fact that the dowel has lost its independent nodal displacement degrees of freedom. In the actual implementation of the element, these zero rows and columns can be retained, and values of one placed on the diagonal to avoid singularity. The dowel nodal displacements are then computed as zero during the solution of the system stiffness equations. Alternatively, the rows and columns of $\mathbf{K}^{de}$ that consist entirely of zeros can be eliminated, resulting in a reduced number of degrees of freedom for the debonded, constrained dowel. The dowel nodal displacements can of course be recovered using Eq. 4.35.

4.4.2 The Debonded, Constrained Case

The development of this case follows in the same manner as the previously presented bonded, constrained case. The i^{th} node of an unbonded dowel can be constrained to an embedding element in the following manner:

$$\mathbf{d}_i \cdot \mathbf{q} = \mathbf{d}_i \cdot \mathbf{q} \tag{4.42}$$

$$\mathbf{d}_i \cdot \mathbf{s} = (\mathbf{N}_i^e \mathbf{U}^e) \cdot \mathbf{s} \tag{4.43}$$

$$\mathbf{d}_i \cdot \mathbf{t} = (\mathbf{N}_i^e \mathbf{U}^e) \cdot \mathbf{t} \tag{4.44}$$

Physically, at a debonded node the dowel retains its independent axial degree of freedom, but displaces compatibly with the embedding element in any transverse direction. The displacement vector at the i^{th} dowel node can then be expressed as:

$$\mathbf{d}_i = (\mathbf{q} \otimes \mathbf{q})\, \mathbf{d}_i + (\mathbf{s} \otimes \mathbf{s} + \mathbf{t} \otimes \mathbf{t})\, (\mathbf{N}_i^e \mathbf{U}^e) \tag{4.45}$$

The transformation matrix, $\mathbf{T}$ is still defined by Eq. 4.36, but $\mathbf{a}_i$ and $\mathbf{b}_i$ are now given by:

$$\mathbf{a}_i = \mathbf{q} \otimes \mathbf{q} \tag{4.46}$$

$$\mathbf{b}_i = \mathbf{s} \otimes \mathbf{s} + \mathbf{t} \otimes \mathbf{t} \tag{4.47}$$

As in the bonded case, the unembedded dowel element stiffness matrix and nodal force vector can be transformed according to Eqs. 4.40 and 4.41. Note that the only difference between the debonded and bonded cases lies in the definition of the submatrices $\mathbf{a}_i$ and $\mathbf{b}_i$. Also note that no consideration has been given to friction between the dowel and the embedding element.

4.4.3 The Debonded Case with a Gap

The inclusion of a gap, γ, between the dowel element and the surrounding slab is of particular interest (see Fig. 4.3). Unlike the previously considered cases, this situation presents a

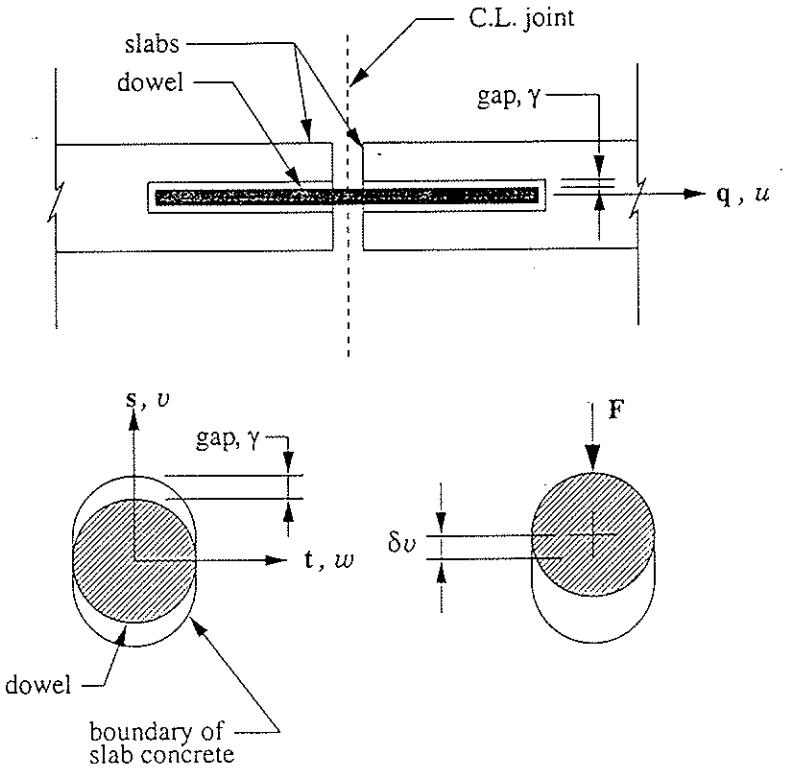


Figure 4.3: Gap Between Dowel and Surrounding Slab

nonlinear nodal contact problem where the embedded dowel nodes may either be constrained to the embedding element, or retain their independent displacement degrees of freedom. The goal is to express this nonlinearity in the transformation matrix, $\mathbf{T}$. Only the case for a gap oriented in one transverse direction will be considered here. Extension to include gaps in both transverse directions is straightforward once the case for a single direction has been developed, and the techniques presented here are useful for modeling dowel action of reinforcing in many structures.

Consider the transverse displacement of a dowel node i. For the sake of discussion, the gap γ_i will be defined in the local s direction, with corresponding scalar displacement component v as shown in Fig. 4.3. The subscript i is a reminder that the gap is defined on a node-by-node basis, and may vary along the length of the dowel. If there is no contact between the dowel and the embedding element, v is independent of the embedding element. If the dowel node is in contact with the embedding element, v is now a function of both the embedding element displacement and γ_i. For the case where the dowel node is in contact with the embedding element (gap closed), the dowel nodal displacement vector, $\mathbf{d}_i$ can be

expressed as:

$$\mathbf{d}_i = \mathbf{a}_i \mathbf{d}_i + \left(\mathbf{N}_i^e \mathbf{U}^e \cdot \mathbf{s} + \gamma_i\right) \mathbf{s} + \left(\mathbf{t} \otimes \mathbf{t}\right) \left(\mathbf{N}_i^e \mathbf{U}^e\right) \tag{4.48}$$

where $\mathbf{a}_i$ is given by Eq. 4.46. In Eq. 4.48, the term $\mathbf{a}_i \mathbf{d}_i$ is the axial displacement of the dowel, the term $\left(\mathbf{N}_i^e \mathbf{U}^e \cdot \mathbf{s} + \gamma_i\right) \mathbf{s}$ represents the displacement in the local s direction, and $\left(\mathbf{t} \otimes \mathbf{t}\right) \left(\mathbf{N}_i^e \mathbf{U}^e\right)$ gives the displacement in the local $\mathbf{t}$ direction; the sum of the three is the dowel nodal displacement vector. Note that γ_i is a signed quantity: when the magnitude of the dowel transverse displacement exceeds that of the embedding element, γ_i is positive; otherwise, γ_i is negative.

Assuming that $\mathbf{N}_i^e \mathbf{U}^e$ has a component in the direction of s gives:

$$\mathbf{s} = \frac{\mathbf{N}_i^e \mathbf{U}^e \left(\mathbf{s} \otimes \mathbf{s}\right)}{\left\| \mathbf{N}_i^e \mathbf{U}^e \left(\mathbf{s} \otimes \mathbf{s}\right) \right\|} \tag{4.49}$$

If m_i is defined according to:

$$m_i = \left\| \mathbf{N}_i^e \mathbf{U}^e \left(\mathbf{s} \otimes \mathbf{s}\right) \right\| \tag{4.50}$$

Substitution of Eqs. 4.49 and 4.50 into Eq. 4.48 gives:

$$\mathbf{d}_i = \mathbf{a}_i \mathbf{d}_i + \left(\left(1 + \frac{\gamma_i}{m_i} \right) \left(\mathbf{s} \otimes \mathbf{s}\right) + \left(\mathbf{t} \otimes \mathbf{t}\right) \right) \mathbf{N}_i^e \mathbf{U}^e \tag{4.51}$$

This gives a new definition for the tensor, $\mathbf{b}_i$, at a constrained, debonded node with a gap of any magnitude:

$$\mathbf{b}_i = \left(\left(1 + \frac{\gamma_i}{m_i} \right) \left(\mathbf{s} \otimes \mathbf{s}\right) + \left(\mathbf{t} \otimes \mathbf{t}\right) \right) \tag{4.52}$$

The tensor $\mathbf{b}_i$ reverts to its previous definition for the debonded, constrained case if $\gamma_i = 0$. The transformation matrix, $\mathbf{T}$, follows the same definition given by Eq. 4.36, and now encapsulates the contact nonlinearity. Several points must be noted, however:

1. $\mathbf{T}$ as defined by Eq. 4.36 when using Eq. 4.52 is valid only for transforming displacements. Nodal forces must be computed using $\mathbf{b}_i$ as defined by Eq. 4.47, and then applying Eq. 4.41.

2. The stiffness matrix of the embedded dowel is still computed according to Eq. 4.40. However, computation of a tangent stiffness, appropriate for use in a Newton iteration, requires that $\mathbf{b}_i$ be defined by Eq. 4.47.

Inclusion of the embedded dowel with gaps in a general nonlinear solution strategy requires that a constraint updating scheme be employed; this is the topic of Section 4.7.

4.5 Inclusion of a General Bond-Slip Law

To extend the potential applicability of the embedded dowel formulation to the modeling of general reinforced concrete, provision should be made for the incorporation of a general bond-slip law. Consider the case where no gaps are assumed to exist between the dowel and the slab, and a general bond-slip law relates incremental interface stress (force per unit length) between the dowel and the surrounding slab, $d\mathbf{f}'$, with the incremental relative displacement of the dowel and the embedding element, $d\boldsymbol{\Delta}'$ as follows:

$$d\mathbf{f}' = \mathbf{D}d\boldsymbol{\Delta}' \tag{4.53}$$

The vectors $d\mathbf{f}'$ and $d\boldsymbol{\Delta}'$ are of size 3×1; the prime indicates that they are defined as having components in the $\mathbf{q}, \mathbf{s}$ and $\mathbf{t}$ directions.

For this situation, constraint of the dowel exists only through stresses at the dowel-concrete interface, and the dowel element itself contributes no stiffness terms to the embedding element. However, is it is convenient to consider the dowel as embedded with corresponding displacement vector $\mathbf{U}^{de}$ as done in the previous section, i.e. Eq. 4.35 is still valid, with $\mathbf{a}_i$ equal to $\mathbf{I}$ and $\mathbf{b}_i$ equal to $\mathbf{0}$. Note, however, that the transformations given by Eqs. 4.40 and 4.41 are not necessary: the only non-zero terms in $\mathbf{K}^{de}$ are those associated with the dowel's original degrees of freedom. The vector of incremental relative displacements, $d\boldsymbol{\Delta}$, may then be expressed as:

$$d\boldsymbol{\Delta} = \mathbf{B}d\mathbf{U}^{de} \tag{4.54}$$

in the global Cartesian coordinates. The matrix operator $\mathbf{B}$ is analogous to the matrix of linear differential operators common in continuum elements, and is defined as:

$$\mathbf{B} = \begin{bmatrix} \mathbf{N}_1 & \mathbf{0} & \mathbf{N}_2 & \mathbf{0} & \mathbf{N}_3 & \mathbf{0} & -\mathbf{N}^e \end{bmatrix} \tag{4.55}$$

$$\mathbf{N}_i = \begin{bmatrix} N_i & 0 & 0 \\ 0 & N_i & 0 \\ 0 & 0 & N_i \end{bmatrix} \tag{4.56}$$

$$\mathbf{0} = 3 \times 3 \text{ array of zeros} \tag{4.57}$$

$$\mathbf{N}^e = \text{the usual array of embedding element shape functions} \tag{4.58}$$

Transforming to the dowel local coordinate system where the incremental stress-strain relations given by $\mathbf{D}$ are assumed to be defined may be accomplished by:

$$d\boldsymbol{\Delta}' = \mathbf{Q}d\boldsymbol{\Delta} \tag{4.59}$$

where $\mathbf{Q}$ is a 3×3 matrix whose rows are the dowel element local unit vectors $\mathbf{q}$, $\mathbf{s}$, and $\mathbf{t}$.

The stiffness contribution due to bond-slip is formulated using virtual work principles. Writing the *incremental* internal virtual work, $d\Pi^e$, yields:

$$d\Pi^e = \int_l d\mathbf{f}' \cdot d\boldsymbol{\Delta}' dl \tag{4.60}$$

Using Eqs. 4.53, 4.54, and 4.59, and transforming the variable of integration to the local element coordinate, η, Eq. 4.60 can be written as:

$$d\Pi^e = d\mathbf{U}^{de\mathrm{T}} \left(\int_{-1}^{1} \mathbf{B}^{\mathrm{T}} \left(\mathbf{Q}^{\mathrm{T}} \mathbf{D} \mathbf{Q} \right) \mathbf{B} h d\eta \right) d\mathbf{U}^{de} \tag{4.61}$$

The stiffness contribution due to bond-slip, $\mathbf{K}^{db}$, is then given by:

$$\mathbf{K}^{db} = \int_{-1}^{1} \mathbf{B}^{\mathrm{T}} \left(\mathbf{Q}^{\mathrm{T}} \mathbf{D} \mathbf{Q} \right) \mathbf{B} h d\eta \tag{4.62}$$

which may be easily evaluated. The total dowel stiffness matrix, $\mathbf{K}^{dt}$ is then:

$$\mathbf{K}^{dt} = \mathbf{K}^{de} + \mathbf{K}^{db} \tag{4.63}$$

and is of size 78×78. Note again that in this case $\mathbf{K}^{de}$ has non-zero values only in the rows and columns pertaining to the dowel element's degrees of freedom, and only $\mathbf{K}^{d}$ need be calculated and added to the upper 18×18 portion of $\mathbf{K}^{db}$. The vector of nodal forces due to bond at the dowel-concrete interface, $\mathbf{F}^b$, is computed as:

$$\mathbf{F}^b = \int_{-1}^{1} \mathbf{B}^{\mathrm{T}} \mathbf{f} h d\eta \tag{4.64}$$

4.6 Shear Beam Spanning the Joint

Modeling the portion of the dowel spanning the joint can be achieved using a conventional shear beam [73]. To enforce displacement compatibility between the shear beam and the embedded dowels on either side of the joint, the end nodes of the shear beam must be constrained to the solid elements between which it spans. This is easily done in the same manner that the dowel element was constrained to an embedding element. Following the development of the matrix $\mathbf{T}$ given in Eq. 4.36 for the embedded dowel, we can specify the expanded vector of nodal displacements as:

$$\mathbf{U}^{be} = \begin{bmatrix} \mathbf{d}_1 & \Theta_1 & \mathbf{d}_2 & \Theta_2 & \mathbf{U}_1^e & \mathbf{U}_2^e \end{bmatrix}^{\mathrm{T}} \tag{4.65}$$

where $\mathbf{U}_1^e$ and $\mathbf{U}_2^e$ refer to the displacement vectors for the elements the shear beam connects at nodes 1 and 2. The transformation matrix, $\mathbf{T}^b$, is defined as:

$$\mathbf{T}^b = \begin{bmatrix} \mathbf{a}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{b}_1\mathbf{N}_1^e & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{a}_2 & \mathbf{0} & \mathbf{0} & \mathbf{b}_2\mathbf{N}_2^e \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.66}$$

The definitions of the submatrices $\mathbf{a}_i, \mathbf{b}_i$ and $\mathbf{N}_i^e$ are the same as for the embedded dowel, but the basis functions must be evaluated for the appropriate solid element at nodes 1 and 2.

4.7 Constraint Updating

Computation of the tangent system stiffness matrix at each iteration in the global solution strategy (see Chapter 6) requires an algorithm specifying the constraint updating logic. This section presents the development of the constraint updating algorithm employed in this study.

With reference to Fig. 4.3, the transverse displacement of the dowel relative to the slab at node i, δv_i, may be expressed as:

$$\delta v_i = \mathbf{s} \cdot (\mathbf{d}_i - \mathbf{N}_i^e \mathbf{U}_i^e) \tag{4.67}$$

The corresponding relative displacement vector for an unconstrained node, Δ_i^u, may then be written as:

$$\Delta_i^u = (\mathbf{s} \otimes \mathbf{s}) (\mathbf{d}_i - \mathbf{N}_i^e \mathbf{U}^e) \tag{4.68}$$

In the case of a constrained node, the transverse displacement of the dowel relative to the embedding element follows directly from Eq. 4.48 as

$$\Delta_i^c = \gamma_i \mathbf{s} \tag{4.69}$$

The criteria for maintaining/releasing constraints at any iteration during the solution are then:

1. For currently unconstrained nodes, if $\|\Delta_i^u\| > \|\gamma_i\|$ the dowel is transversely constrained to the embedding element.

2. For a currently constrained node, if $\mathbf{F}_i \cdot \Delta_i^c > 0$ the constraint is released; otherwise, the transverse constraint is maintained. The vector $\mathbf{F}_i$ represents the force applied on the dowel by the surrounding concrete at node i as shown in Fig. 4.3.

Physically, the inequality in (2) is false if $\mathbf{F}_i$ has a component opposite the direction of the transverse relative nodal displacement, i.e. the dowel is compressing the surrounding slab in the s direction at the point of contact.

The simplest constraint updating algorithm follows directly from this criteria for maintaining/releasing constraints. Based on the current displacements and constraints, the transverse nodal forces, $\mathbf{F}_i$, acting on the dowels are found. For currently unconstrained nodes, if the relative displacements exceed the gap, the node is constrained. The currently constrained i^{th} node is released if $\mathbf{F}_i$ does not oppose the relative displacement.

This simple node-by-node updating scheme can be improved upon, however. At the end of an iteration in the global solution scheme, the displaced shape of the model has just been incremented, and is fixed while constraints are checked. We can treat all quadratic dowel and cubic beam elements as a single substructure within the model that has enforced non-zero displacement constraints at all points of contact with the slabs. Defining:

$\mathbf{U}^t$ = nodal displacement vector for dowel/beam substructure

$\mathbf{U}^c$ = enforced displacements at contact nodes for dowel/beam substructure

$\mathbf{U}^u$ = vector of unknown displacements at free dowel/beam nodes

$\mathbf{K}^t$ = assembled stiffness matrix for dowel/beam substructure

$\mathbf{P} = \mathbf{K}^t\mathbf{U}^c$

$\mathbf{F}$ = reactions at enforced displacement nodes

we can solve $\mathbf{K}^t\mathbf{U}^u = -\mathbf{P}$ for $\mathbf{U}^u$; the total displacement vector for the dowel/beam substructure is $\mathbf{U}^t = \mathbf{U}^u + \mathbf{U}^c$. After computing the reactions at the constrained nodes, $\mathbf{F}$, the direction of the individual reaction at each enforced displacement node, $\mathbf{F}_i$, may be dotted with the corresponding relative displacement, Δ_i, and released if required. This process is then applied repeatedly until no more constrained nodes are released ($n_f = 0$) or a maximum number of times, max_f, whichever occurs first.

Of course, we must also update constraints based on relative displacements at the free nodes. This is performed in an outer loop within which the previous iteration is contained. As with the inner loop where constraints are maintained/released based on nodal forces, the outer loop terminates when no more constraints are updated based on relative displacements ($n_d = 0$) or the maximum iteration limit, max_d, is reached. This leads to an updating

scheme with two levels of iteration as given in Algorithm 4.1.

```
update(U)
 1  ▷ outer loop for updating free nodes
 2  while (i < max_d) and (n_d > 0)
 3  do j = 1
 4      ▷ inner loop for updating constrained nodes
 5      while (j < max_f) and (n_f > 0)
 6      do U^c = get_fixed_displ(U)
 7          P = K^t U^c
 8          ▷ solve for unknown displacements
 9          U^u = linear_solve(K^t, -P)
10          U = U^c + U^u
11          ▷ compute fixed node reactions
12          F = compute_F(U, K^t)
13          n_f = update_fixed_nodes(F)
14          j^++
15      n_d = update_free_nodes(U^u)
16      i^++
17  return
```

Algorithm 4.1: Pseudo-code for Dowel Constraint Updating

For the analyses performed in this study, both max_d and max_f were fixed at five which generally allowed Algorithm 4.1 to converge to an unchanging set of constraints at each iteration in the global solution strategy. Preliminary comparisons indicated that Algorithm 4.1 provides significantly improved performance over the simpler updating strategy presented earlier.

4.8 Element Convergence Study

This Section presents results of a convergence study designed to verify the element formulation and the constraint updating strategy. Both the number of dowel elements and the ability of the model to predict joint response with varying amounts of dowel looseness are studied.

4.8.1 Finite Element Model

Fig. 4.4 illustrates the geometry and finite element discretization of the two slab system chosen as a model problem. A single layer of elements was used to model the slab, which previous studies have shown to be adequate to capture the displaced shape of the slab. The modulus of elasticity of the slab was chosen as 28 kN/mm^2. The dowel diameter used was

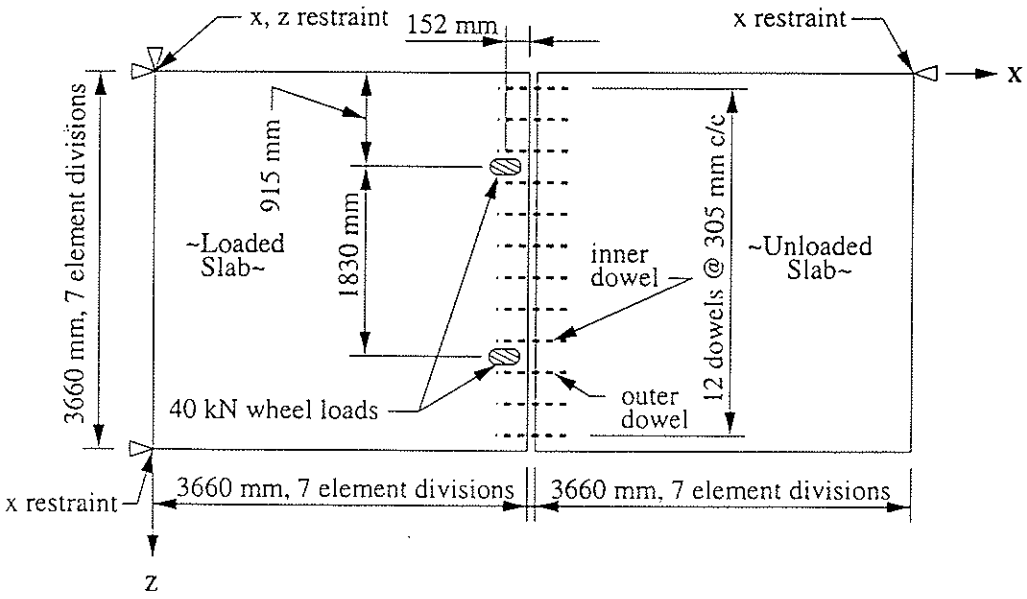


Figure 4.4: Plan View of Two Slab Model Problem

32 mm, and its modulus of elasticity taken as 200 kN/mm^2. The subgrade was modeled as a dense liquid foundation with a modulus, k, of 0.0407 N/mm^3. The value of k is known to vary with the magnitude and velocity of the wheel loading as well as the subgrade type [16, 54]; the value chosen here is considered representative for an untreated gravel subgrade. The only loading considered in this study was an equivalent single axle load (ESAL) of 80 kN applied as point loads near the joint as shown in Fig. 4.4. The slab boundary conditions are displacement constraints sufficient to prevent rigid body displacements and rotations as shown in Fig. 4.4.

The gap between the embedded dowels and the slab was assumed to be concentric, varying parabolically in magnitude along half the dowel length for both the loaded and unloaded slabs as shown in Fig. 4.5. Each dowel was considered to be debonded along its length, except at a single end node as required to prevent rigid body displacement of the dowel in its axial direction. The zero gap between the slab and the dowel near the dowel ends ensures that the dowel is constrained to the slab at these nodal locations, providing sufficient transverse dowel restraint. The magnitude of the gap at the joint was varied between 0 mm and 0.3 mm, as larger values had little effect on pavement response. Note that laboratory studies have indicated that the magnitude of the gaps formed under cyclic damage vary from about 0.3 mm to 0.6 mm, although no information about the variation in magnitude along the dowel length is known [10, 81]. The parabolic variation was assumed

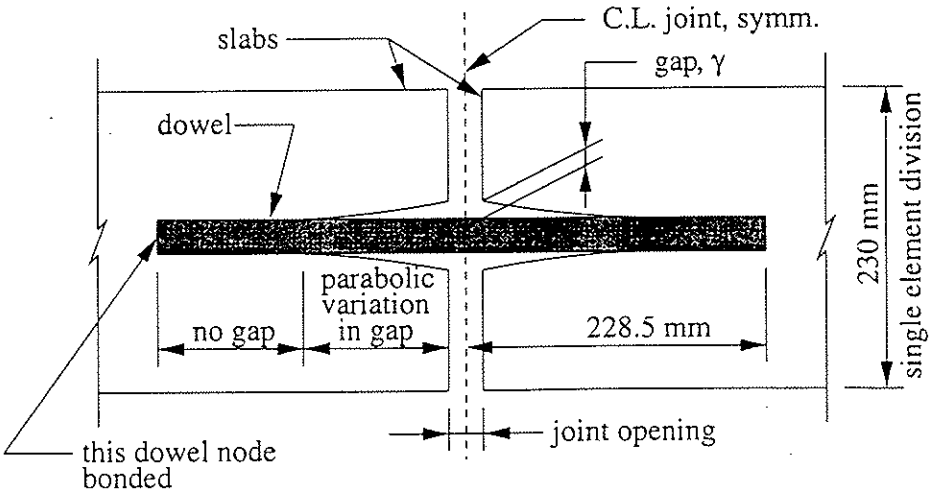


Figure 4.5: Assumed Gap Profile in Convergence Study

based on the likely development of the gap throughout the loading history of the joint. The first appearance of a gap can be expected at the point of the highest contact stresses near the joint. As the gap widens near the joint, the point of highest contact stresses will move back along the length of the dowel, resulting in the region of the gap propagating back along the length of the dowel. Although the dowels closest to the wheel loads transfer the most shear across the joint, and thus can be expected to develop larger gaps than the other dowels, the gap was assumed to be identical for all dowel lines. The joint width was fixed at 5 mm; it was not necessary to consider nodal contact between the vertical faces of the joint due to the small magnitudes of the displacements.

4.8.2 Results of the Convergence Study

To verify the element formulation, the level of discretization of the dowels in each slab was varied, while the gap was fixed at an intermediate value of 0.15 mm. The sensitivity of both load transfer efficiency (LTE) and dowel shears to the number of dowel elements was examined. Load transfer efficiency, commonly used to measure joint condition [83, 61], is defined as:

$$\text{LTE} = \frac{d_u}{d_l} \times 100\% \tag{4.70}$$

where d_u is the vertical (downward) deflection of the unloaded slab at the joint, and d_l is the vertical deflection of the loaded slab at the joint. The dowels closest to the applied wheel loads are denoted as "inner" and "outer" as shown in Fig. 4.4; LTE values are reported for both the inner and outer dowels. The results of the convergence study are shown in

Fig. 4.6. From these plots, it is clear that the LTE values and dowel shears converge with increasing numbers of dowel elements. Over the number of dowel elements considered, the LTE varies less than 3% for both the the inner and outer dowels, and the dowel shears vary approximately 2.5% in magnitude. Based on this convergence study, six elements per dowel line in each slab were used to discretize the dowels in the remaining runs. Note that this corresponds to three elements over the region of the gap.

Further studies examine the effect of the magnitude of the gap on the LTE and dowel shears. Figure 4.7 shows, for the outer dowel, the effect of the gap on LTE and dowel shears. Note that a gap of 0.3 mm reduces the LTE by over 50%, and that the outer dowel shear is reduced by approximately 500%. The variation in LTE is is in good agreement with measured values reported in [10], which showed a loss in LTE of over 50%. These results indicate that even relatively small gaps may have a large effect on the ability of the dowels to effectively transfer loads from the loaded slab to the unloaded slab.

4.9 Summary

An embedded formulation for a quadratic beam element incorporating both bending and shear deformations suitable for modeling dowel action in rigid concrete pavements has been proposed and developed. The embedded formulation has significant practical advantages over a discrete formulation, allowing both an exact mathematical representation of the dowels and regular meshes for the slabs. The specific advantages of the embedded formulation are as follows:

- Slab mesh divisions are not restricted to coincide with dowel lines.
- The element may be debonded at selected nodes.
- The nonlinear effect of gaps between the dowel and surrounding slab concrete may be explicitly modeled.
- The element stiffness is easily computed using a matrix transformation, and does not require special integration techniques. Both the effects of debonding and gaps are incorporated in this transformation.
- The dowel element permits the incorporation of a general bond-slip law, making it appropriate for modeling reinforcement in conventional reinforced concrete structures.

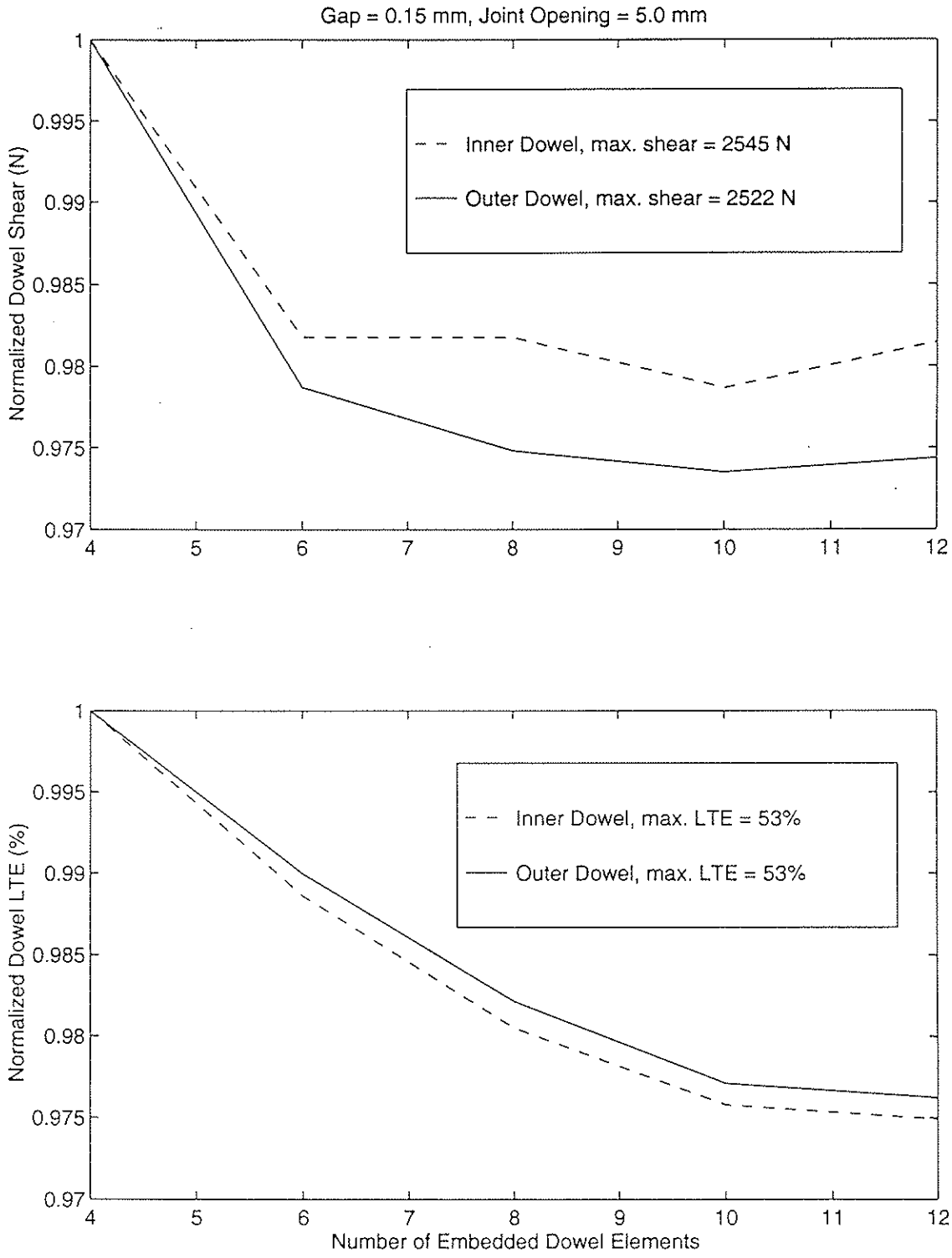


Figure 4.6: Dowel Shear and LTE vs. Number of Embedded Dowels for $\gamma = 0.15$ mm

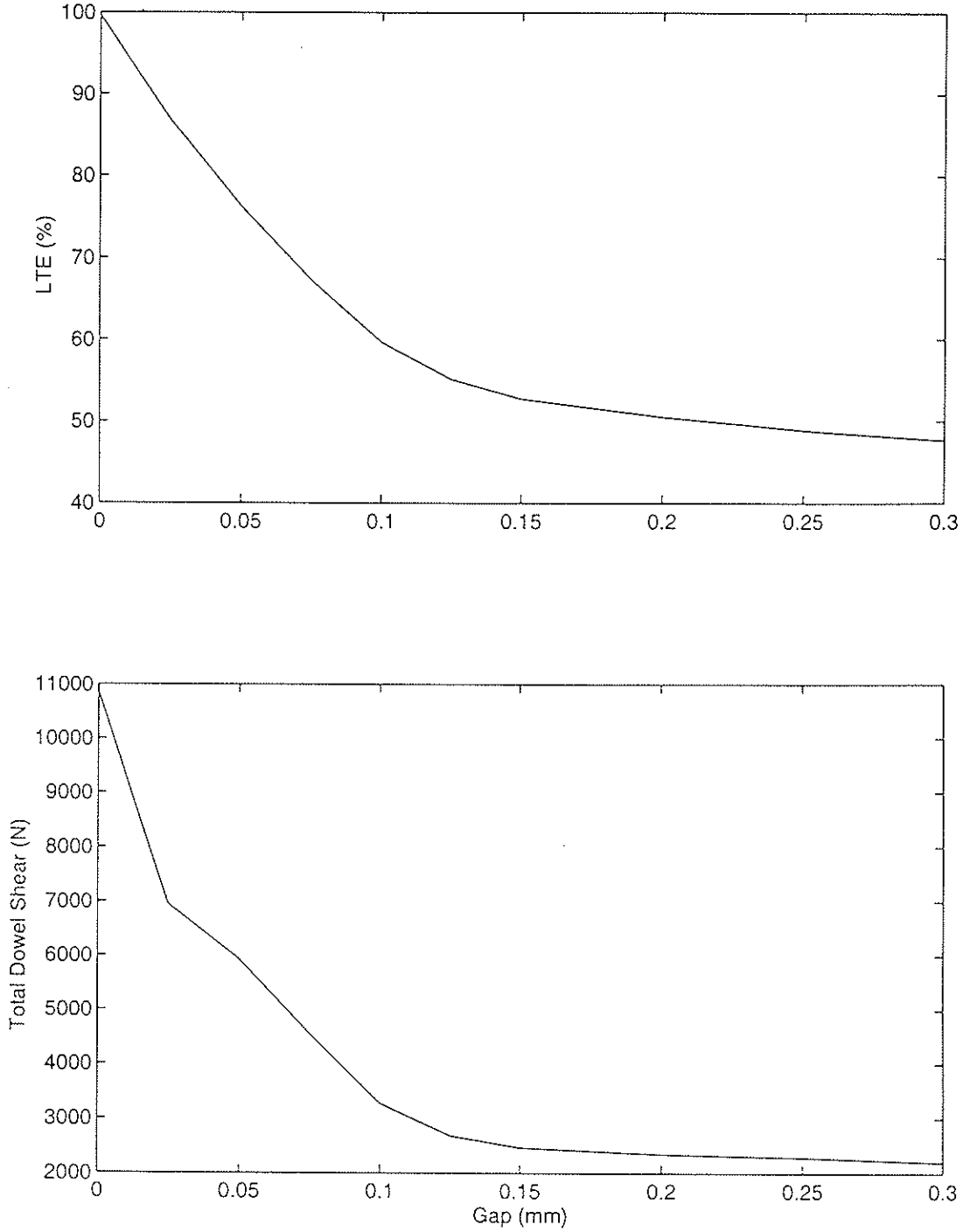


Figure 4.7: LTE and Joint Shear vs. Gap for Outer Dowel

A convergence study was performed to verify the element formulation. The study indicated that the element performance is as expected, giving rapidly converging joint displacements and dowel shears with increasing levels of discretization of the embedded dowel when gaps exist. Relatively small gaps between the dowel and the slab (less than 0.3 mm) appear to have significant effects on the joint displacements and the ability of the dowels to transfer shear from the loaded to the unloaded slab. This is in agreement with existing laboratory data [81, 10], and serves to justify the need to model explicitly dowel looseness.

Chapter 5

Aggregate Interlock

5.1 Introduction

Early experimental studies on aggregate interlock shear transfer in rigid pavements demonstrated that joint shear transfer effectiveness and endurance depend on many factors including joint width, slab thickness, load magnitude, foundation type, subgrade modulus, and aggregate shape [15].

Research into the more general problem of shear transfer across discrete cracks in concrete has shown the mechanics of aggregate interlock shear transfer to be highly complex. In addition to contact between asperities on joint surfaces, there may be localized crushing of both the cement paste and the aggregate (see Fig 5.1) [7, 91]. The amount of crushing

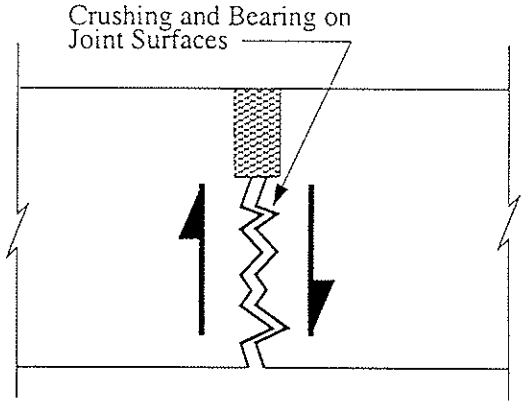


Figure 5.1: Aggregate Interlock Shear Transfer

and the bearing area of the surfaces depends on the joint opening, normal restraint of the joint, the strength of the concrete (both the paste and the aggregate), and the size and distribution of the aggregate particles [91]. A predictive model of aggregate interlock shear transfer at rigid pavement joints should take all of these factors into consideration. The

complexity is further heightened by cumulative damage to the joint due to cyclic loading, which reduces the ability of the joint to transfer shear [85, 92]. A recent experimental study is underway which attempts to quantify the degradation in aggregate interlock shear transfer with high cycle, low stress conditions common to rigid pavements [3].

Despite the complexity of aggregate interlock shear transfer, current practice is to use discrete linear spring elements spanning the joint when modeling rigid pavement systems (see Chapter 2). While the use of linear springs may be reasonable for an examination of the effect of aggregate interlock shear transfer effectiveness on the global slab response (see [44] or [45], for example), it does not permit accurate modeling of local response at the joint. Even when the use of linear springs is appropriate, the rational choice of a spring stiffness (constitutive relation) may be difficult, if not impossible. Moreover, the appropriate spring value is effectively a secant stiffness, valid only for one model geometry, set of material properties, and loading.

Clearly, there is a need for more realistic finite element modeling of aggregate interlock shear transfer; the development of such a technique is the focus of this Chapter. Common approaches to the finite element modeling of aggregate interlock shear transfer are discussed next, with consideration given to both element type and incorporation of constitutive relations. Section 5.3 addresses the mechanics of aggregate interlock, and contains a detailed literature review that serves to provide a basis for the choice of crack constitutive relations appropriate for the present problem. Lastly, a technique for the finite element modeling of aggregate interlock shear transfer in rigid pavement slabs is developed.

5.2 Review of Aggregate Interlock Modeling

The finite element modeling of discrete cracks in concrete requires that the crack constitutive relations be chosen and an appropriate element be used. Element types commonly used in the finite element modeling of shear transfer across discrete joints in concrete can be grouped into three categories:

- Discrete linkage elements between nodes.
- Continuum finite elements with reduced shear stiffness.
- Special zero-thickness isoparametric joint elements.

Some representative 2D elements are illustrated in Fig. 5.2.

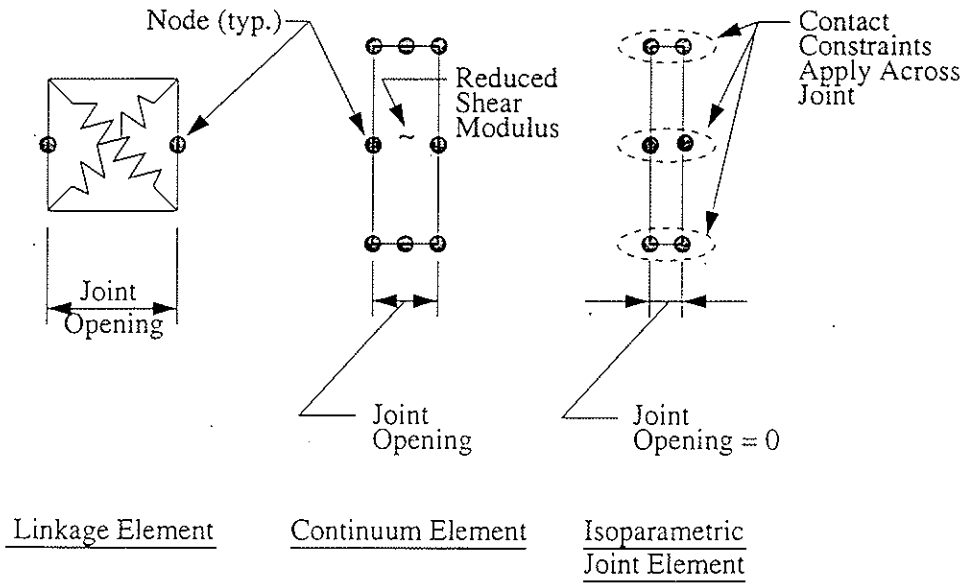


Figure 5.2: Common Elements for Modeling Aggregate Interlock Shear Transfer

Linkage elements have the obvious advantage of simplicity and ease of implementation, and are often used in finite element models of rigid pavement systems. However, it has been shown [7] that linkage elements are incapable of modeling joint shear transfer because of the indefiniteness of the constitutive relations. The error induced by this approach depends on the magnitude of the normal forces at the joint; as long as these normal forces are small, discrete springs may be acceptable. The use of a continuum finite element with a reduced shear stiffness is also unable to model the normal forces occuring at the joint. In addition, large relative tangential displacements at the joint may induce large element strains, violating small strain theory and requiring that higher-order strain approximations be employed. The third approach, while more complex, is desirable since any constitutive relations based directly on (δ_n, δ_t) may be incorporated [36]. This also avoids any potential difficulties that may arise from large strains occuring at the joint.

Attempts to rigorously incorporate complex aggregate interlock constitutive models within finite element models are uncommon, but this issue was tackled by Feenstra et al. [25, 26]. In these studies, various aggregate interlock constitutive models, including those developed by Walraven [91, 74] and Bazant [7], were implemented using isoparametric joint elements. Several significant computational difficulties were noted: (1) the element constitutive relations are quite complex, and (2) the incremental constitutive matrix is not symmetric (discussed in detail in the following Section). As shown by Feenstra et al. [25, 26], this lack of symmetry can result in unstable and/or poorly conditioned systems that require

very small load steps to converge.

Another issue which must be addressed for the present problem is the potential for contact between the opposing joint faces when the slabs displace as shown in Fig. 5.3. Note

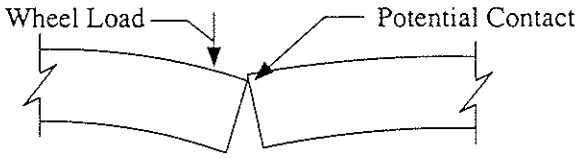


Figure 5.3: Contact Between Opposing Joint Surfaces

that when discrete cracks are modeled in cracked concrete beams or slabs, this is often not a consideration since the cracks are continually growing and opening while the load is increased. This is fundamentally different from the current problem, where a full-depth crack exists, and the crack opening varies significantly due to seasonal contraction/expansion of the slabs. This is a similar situation to that which occurs in the modeling of jointed rock [36].

Joint contact conditions may be incorporated within the isoparametric joint element stiffness matrix using penalty parameters [36]. The penalty parameters may be viewed as shear and normal stress constitutive relations, and calibrated accordingly. For example, consider a 2D case at a node where contact conditions require a constraint, but the applicable constitutive model predicts a non-zero shear stiffness with no coupling between normal and shear stresses. For this case, the element constitutive matrix at the node would be:

$$\mathrm{D} = \begin{bmatrix} \kappa_{nn} & 0 \\ 0 & D_{tt} \end{bmatrix} \tag{5.1}$$

where κ_{nn} is the penalty parameter (effectively a very large elastic modulus), and $D_{tt} \ll \kappa_{nn}$ the effective shear stress-displacement modulus.

The large difference in magnitude between D_{tt} and κ_{nn} highlights a problem identified with penalty parameter approaches in general: they yield a poorly conditioned stiffness matrix. The solution strategies used in this study rely on iterative techniques for solving the system stiffness equations, the convergence of which is greatly slowed by poor conditioning. To avoid this difficulty, the contact conditions at the joint are modeled independently of the joint constitutive relations and enforced with Lagrange multipliers. Details may be found in Chapter 6.

5.3 Mechanics of Aggregate Interlock

5.3.1 Background

Early research on the empirical development of shear stress–displacement relationships was often conducted in relation to the design of nuclear containment structures. A summary of proposed shear stress–displacement relationships may be found in [11]. These models are largely empirical, relying on laboratory test data for determination of appropriate parameters, and generally apply to reinforced concrete structures. They may be divided into two broad categories:

- Isolated crack stiffness models that provide an interface shear stiffness per unit area given parameters such as concrete strength and the initial crack opening. These models are generally derived from fitting curves to experimental data of shear stress-displacement, and are limited to constant crack openings and/or crack restraint.

- Crack stiffness models that may be incorporated directly in element constitutive relations. These models may be analytical or based on extensions of isolated crack stiffness models. They are generally applicable for use with continuum elements, and involve reduction of the shear modulus based on crack width and stress.

More recent research on the mechanics of aggregate interlock has highlighted physical phenomena that these early models are not capable of rationally capturing. One of the most important and earliest studies of this type was that of Bazant [7], who examined the mechanical aspects of shear and normal stress transfer across cracks in plain concrete as part of a study on shear stress transfer across rough cracks in reinforced concrete. This work is significant since it represents an early attempt to comprehensively and rationally incorporate micromechanics and the effect of crack asperities in an aggregate interlock model.

Bazant [7] noted that shear slip across a crack requires an increase of the crack opening, which will increase the normal compressive stresses across the crack if it is restrained. Based on this observation, the incremental stress-strain relations for rough cracks were formulated in the context of monotonic loading using total displacements as follows:

$$\sigma_n = f(\delta_n, \delta_t) \tag{5.2}$$

$$\sigma_t = f(\delta_n, \delta_t) \tag{5.3}$$

$$d\sigma = \mathrm{D}d\delta \tag{5.4}$$

where:

$$d\boldsymbol{\sigma} = \begin{bmatrix} d\sigma_n \\ d\sigma_t \end{bmatrix} \tag{5.5}$$

$$\mathbf{D} = \begin{bmatrix} D_{nn} & D_{nt} \\ D_{tn} & D_{tt} \end{bmatrix} \tag{5.6}$$

$$d\boldsymbol{\delta} = \begin{bmatrix} d\delta_n \\ d\delta_t \end{bmatrix} \tag{5.7}$$

Here, (δ_n, δ_t) represent the relative normal and tangential crack displacements, respectively; σ_n is the normal stress on the crack plane; and σ_t the shear stress.

Bazant made several important observations regarding the nature of stress transfer across discrete cracks:

- Crack slip at a constant opening must induce normal stresses across the crack.
- The initial deformation at a crack must begin with a normal displacement, which is then followed by a tangential displacement inducing shear stresses.
- For δ_t = constant and > 0 and increasing δ_n, both σ_t and σ_n must decrease in magnitude; thus $D_{nn} < 0$ and $D_{tn} < 0$.
- Similarly, for δ_n = constant and > 0, an increase in δ_t produces increases in σ_n and σ_t, meaning that $D_{tt} > 0$ and $D_{nt} > 0$.

The last two observations lead to the immediate conclusion that the crack constitutive matrix is never positive definite except when $\delta_t = 0$.

Since this initial work by Bazant [7], a large number of studies, both experimental and theoretical, have been conducted on stress transfer across discrete cracks in concrete. The remainder of this Section provides a detailed literature review of these studies, organized by author. Empirical models, where crack constitutive relations are derived from or compared to test data, are covered first. This is followed by a review of micromechanics-based models where constitutive relations are developed from basic assumptions of crack morphology and the physical properties of the concrete.

5.3.2 Recent Empirical and Semi-Empirical Models

Reinhardt and Walraven (1982)

In an attempt to increase the existing experimental database on aggregate interlock shear transfer and to resolve conflicting conclusions from previous tests by other researchers, Reinhardt and Walraven [74] conducted a series of experimental investigations. Tests on both

specimens with both internal restraint (conventional reinforcing bars intersecting the crack plane) and external restraint were conducted. The effects of concrete strength and aggregate gradation and maximum particle size were considered. In addition, bilinear curves were fitted to the experimental shear stress–shear displacement curves, with the main parameters being crack opening and concrete strength. Some conclusions of the experimental study related to plain (externally restrained) specimens are:

- For externally restrained cracks, crack opening path (relation between δ_t and δ_n) is influenced by the external restraining stiffness.
- High restraining stiffness keeps the crack opening small, giving a large total contact area and high shear resistance.
- The shear resistance depends on contributions from all aggregate particles with diameters larger than the crack width.

Millard and Johnson (1984)

This study [65] examined both aggregate interlock shear transfer and dowel shear transfer through a series of experiments. The primary goal of the aggregate interlock tests was comparison with previously developed theoretical models, including that developed by Walraven [91] (reviewed later in this Chapter).

A total of 19 aggregate interlock specimens were tested, with the main parameters being initial crack width (0.063 mm – 0.75 mm), concrete strength (29 N/mm^2 – 52 N/mm^2 cube strength), and specimen stiffness normal to the crack plane (1.2 N/mm^3 – ∞). One test was conducted with constant crack opening (infinite restraining stiffness). Tensile loading, inducing axial tension in the reinforcing bars, was applied through flexible straps to minimize the stiffness of the test rig. The specimens consisted of two concrete blocks with a pre-formed crack. Crack restraining stiffness was provided by two reinforcing bars extending through the specimen that were sleeved to prevent any contributing stiffness due to dowel action. All tests were monotonic, and loading was applied until failure.

The test results gave relationships between crack width, relative shear displacement, and normal and shear stresses on the crack. The following general trends were observed:

- Nominally identical specimens showed nearly identical behavior.
- Shear stiffness and ultimate shear stress decrease with increasing initial crack width. Cracks widen (open further) with increasing shear regardless of the initial value of the

crack opening.

- Increasing axial stiffness (restraint) increases the shear stiffness and ultimate shear stress.
- For tests where the crack width was allowed to increase under increasing shear stress, the concrete strength had only a small effect on the ultimate shear stress.
- The test with constant crack opening exhibited significantly higher ultimate shear stress and stiffness than similar tests which permitted progressive crack opening.

Comparison of the test results with theoretical models was also done. The primary conclusion of the author is that shear is resisted by a combination of crushing and sliding; the comparison with Walraven's two-phase model, which captures this behavior, was generally quite good. The conclusion of Laible et al. [55] that bearing/crushing predominates for small crack widths ($\leq$ 0.25 mm) and sliding predominates for larger crack widths was not supported by the test results.

Tassios and Vintezeleou (1987)

Tassios and Vintezeleou [85] performed a series of cyclic shear-displacement tests on plain concrete specimens. Three types of interfaces were considered: smooth (as typically found in pre-cast connections), rough (natural cracks induced by pre-cracking the specimens), and sand-blasted interfaces. Three concrete strengths were used (16 MPa, 30 MPa, and 40 MPa), and three different normal stress levels. All tests were displacement controlled, and the monitored data included the crack dilatancy, shear displacement, and shear stress.

The main observations of the test data regarding the rough interfaces are as follows:

- There is an increase in maximum shear stress transfer with increasing normal stress. This increase is not proportional.
- For small initial crack widths (<0.10 mm) which were typical for this testing program, there is no significant free shear slip before large shear stresses are mobilized. The authors qualitatively state that larger initial crack widths should allow for larger initial free shear slip.
- With regard to the hysteretic behavior, the response is markedly asymmetric during the first reversal of loading, i.e. the maximum shear stress transferred drops significantly after the first load reversal. This effect decreases during subsequent load cycles.

- Due to the smoothing of the interfaces during cycling, there is a large drop in shear stiffness, and the force-response degradation is large. Further, the maximum crack dilatancy decreases with cycling, and the residual shear slip after unloading is nearly equal to the maximum shear slip for each loading cycle.

Tassios and Vintezeleou also presented empirical expressions which were developed to fit the shear hysteresis loops for both smooth and rough interfaces. The details of these expressions will not be repeated here. The authors caution that scale effects may be significant.

Divakar, Fafitas, and Shah (1987)

Divakar et al. [19] performed a series of monotonic shear tests on pre-cracked plain concrete specimens. The concrete strength of all specimens was 35 MPa, and the maximum aggregate size was 12.7 mm. Measured data for shear stress versus slip and slip versus dilation were taken at four different values of constant normal stress.

The test results and data previously published by other researchers were then used to develop analytical models to predict crack stiffness using total deformation theory in a manner similar to that of Bazant [7]. Empirically determined factors for crack shear stiffness, coefficient of friction, dilatancy, and normal stiffness were proposed. These factors were incorporated into expressions relating incremental displacements and stresses assuming that shear stress, σ_t, and normal displacement, δ_n, were functions of δ_t and σ_n. Expressions for the coefficients D_{nn}, D_{nt}, D_{tn}, and D_{tt} were then derived; the resultant matrix relating incremental displacements and stresses, D, was not symmetric positive definite, which is consistent with previous work by others [7]. The parameters defining the coefficients of D were optimized with respect to the author's and other published test data using nonlinear least squares. The accuracy of these analytical expressions was tested by comparison with published test data from several other sources that incorporated different degrees of crack normal restraint stiffness. Agreement with the test data was generally good, with the model somewhat overpredicting shear stress vs. slip in most cases.

5.3.3 Micromechanics-Based Models

Several models that allow prediction of crack stiffness and response based on a theoretical micromechanics-based approach have been developed. These models are reviewed in detail in this section.

Fardis and Buyukozturk (1979)

In this study [23], an examination of the micromechanical aspects of both aggregate interlock and dowel shear transfer was undertaken. Underlying the aggregate interlock model is the assumption that a crack surface can be decomposed into a "local" roughness and a "general" roughness as originally proposed by Laible et al. [55], and that the concrete can be represented as a single-phase medium. The model decomposes the local roughness into a variable friction angle, and the general roughness is defined as a stochastic surface composed of parabolic segments.

The crack is assumed to be able to open non-uniformly, with a linear variation in δ_n along the crack length. This, coupled with the prior assumptions as to the variations in the crack surface and joint opening, results in a unique solution for two contact points on any crack of finite size. The model does not allow the prediction of normal and tangential stresses given a displaced shape; instead, the relative displacement of a finite two-dimensional crack area gives rise to shear and normal forces and a moment.

Walraven (1981)

Walraven [91] addressed the fundamental physical relations between shear and normal stresses and relative tangential and normal displacements across discrete cracks in concrete. This study dealt largely with monotonic loading, i.e., both the shear and normal stress are functions of the tangential and normal displacements only: the loading history is not considered. The stresses and constitutive relations were formulated as first identified by Bazant [7] (see Eqs. 5.2-5.4) based on a representation of concrete as constituted by two separate materials, the cement paste and the aggregate. The weakest link in the concrete matrix was assumed to be the bond between the cement paste and the aggregate, and thus cracking occurs along the aggregate boundaries. The microroughness, caused by the aggregate particles intersecting the crack plane, was assumed to dominate the macroroughness of the crack, defined as the overall variation in path of the crack face. Because of the large plastic deformations of the cement paste due to pore-volume reduction, the cement paste was idealized as obeying a rigid-plastic stress-strain law. The aggregate was idealized as incompressible.

The aggregate particles were modeled as spheres of varying size, distributed according to a Fuller curve. The aggregate particles intersect the crack face at various depths depending on their statistical distribution within the concrete matrix (see Fig. 5.4). As a result of these

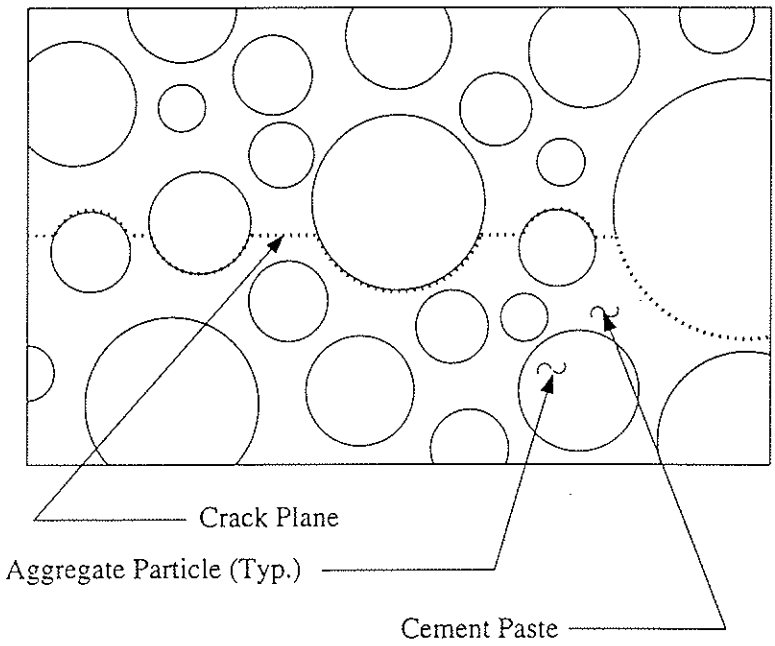


Figure 5.4: Distribution of Aggregate Particles

assumptions, the crack may be reduced to a two-dimensional problem of finite width. The shear and normal stresses at the crack result from the tangential and normal projections of the stresses produced when the cement paste deforms plastically as it bears on the aggregate particles (see Fig. 5.5). These stresses are related by assumption that the contact areas are about to slip, and thus:

$$\tau_{pu} = \mu \sigma_{pu} \tag{5.8}$$

where μ is the coefficient of friction between the paste and the aggregate, and σ_{pu} is the ultimate strength of the paste. The equilibrium of the crack plane requires that the net forces be balanced as follows:

$$\sigma_n = \sigma_{pu}(\overline{A}_x - \mu \overline{A}_y) \tag{5.9}$$

$$\sigma_t = \sigma_{pu}(\overline{A}_y + \mu \overline{A}_x) \tag{5.10}$$

where:

$\overline{A}_x$ = x-projection of the sum of the most probable contact areas

$\overline{A}_y$ = y-projection of the sum of the most probable contact areas

The determination of $\overline{A}_x$ and $\overline{A}_y$ is based on the statistical distribution of the aggregate in the concrete matrix, and the geometry of the spherical aggregate particles intersecting the crack plane at a given tangential and normal displacement.

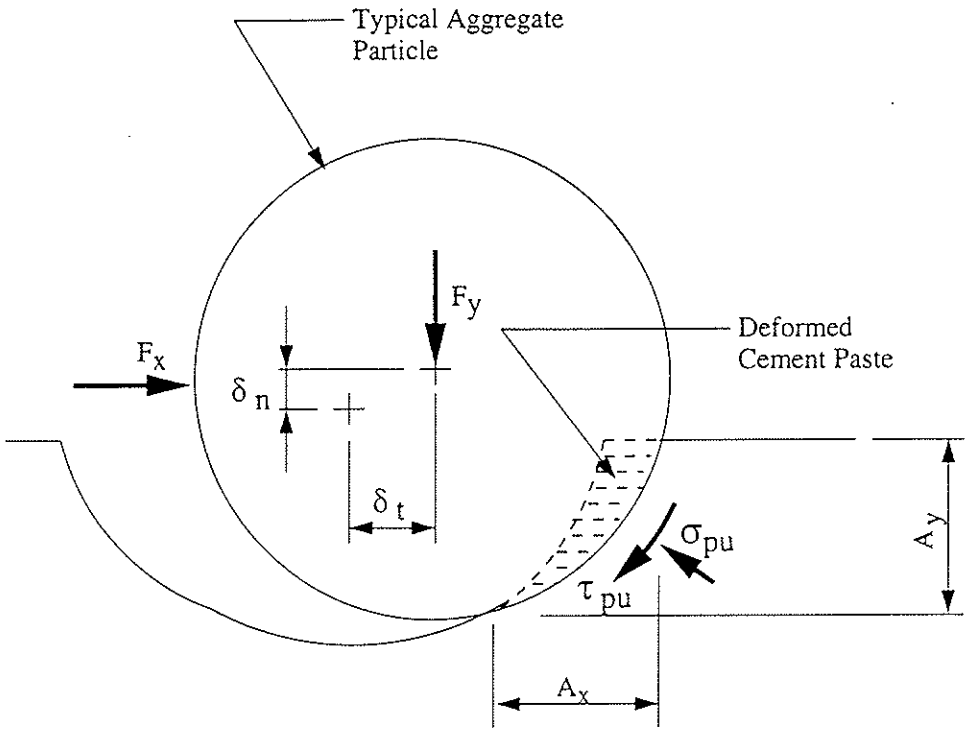


Figure 5.5: Deformed Cracked Plane

In an attempt to verify the theory, a series of tests were run in which the crack width was maintained at a constant value, and the shear stress, normal stress, and shear displacement were measured. The values of σ_{pu} and μ were determined to provide the best fit to the experimental data for a crack opening of 1.0 mm; these values were then used to calculate the stress-displacement curves for other crack opening values. It was found that the theory was generally in good agreement with the experimental data for all values of crack openings. In addition, parametric studies were conducted which allowed the influence of the friction coefficient, the size of the aggregate, and the grading curve to be studied. The contribution of different aggregate fractions to stress transmission was examined, and the possibility of extending the theory to cyclic loading was examined in a qualitative manner. Some of the relevant conclusions of Walraven's initial study were:

- The experimental results can be adequately described by the adopted physical model.
- A friction coefficient of 0.4 between the aggregate particles and the matrix results in the best fit between experimental data and the model. This value is independent of the concrete quality. The yield strength of the cement paste which gives the best results is a function of the concrete strength. Both parameters are in good agreement

with experimental data described in the literature.

- The resistance to shear deformations is governed by the concrete strength, with the diameter of the aggregate playing a secondary role.
- The aggregate grading curve does influence the shear transfer characteristics at larger crack widths. The influence of the grading curve on the normal stresses is minimal.
- The behavior of the cracks subjected to cyclic loading can be qualitatively described by the model.

Li, Maekawa, and Okamura (1989)

The work by Li et al. [56] represents a unique approach to the fundamental analysis of aggregate interlock shear transfer. The crack is assumed to be a rough surface with a number of areas, or contact units, having various angles of inclination, θ, that vary from $-\pi/2$ to $\pi/2$. A probability density function $\Omega(\theta)$ is assumed to give the probability of occurrence of any θ. The crack surface is shown schematically in Fig. 5.6. The second major assumption of the theory is that the direction of contact is fixed, and assumed normal to the initial contact direction, θ.

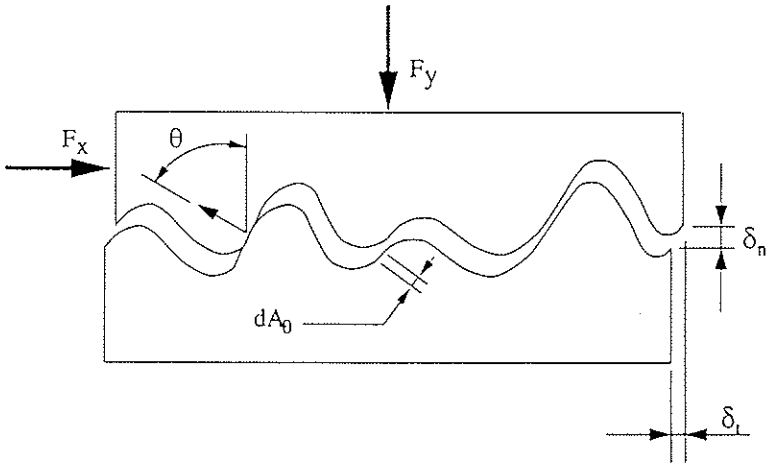


Figure 5.6: Surface Roughness - Model of Li et al.

A series of experiments were conducted at fixed crack widths. Based on the histogram of the measured surface profile of the crack, the following analytical form for $\Omega(\theta)$ was proposed:

$$\Omega(\theta) = 0.5cos(\theta) \tag{5.11}$$

An elastic-plastic material model was assumed to govern stress transfer at the contact points. An experimentally verified expression for the ratio of the contact area to the total crack area, K, was defined, permitting the calculation of the contact area as a function of δ_n and the maximum aggregate diameter. Based on these assumptions and proposals, analytical incremental constitutive relations were derived. Note that the elastic-plastic material allows cumulative damage to be accounted for, as well as permitting cyclic loading.

Following the development of the model, comparison with the experimental results was done. It was found that the model generally predicted cyclic and monotonic behavior well. However, the unloading/reloading paths computed with the proposed model are identical, whereas the experimental loading paths were significantly different. For large numbers of cycles, the authors note that the effects of smoothing and fracturing of the asperities becomes significant and that the model may not accurately capture this behavior. The sensitivity of the model to different forms of $\Omega(\theta)$ was checked by assuming different functions (i.e. linear, elliptic, uniform) and comparing computed monotonic response with the test data; the proposed form was found to give the best response, although the predicted response is sensitive to the assumed functional form of $\Omega(\theta)$.

Divakar and Fafitas (1992)

In [18] and [22], Divakar developed a micromechanics-based constitutive model for interface shear. The model is similar to that developed by Walraven in that it explicitly incorporates asperities representing the crack microroughness; however unlike Walraven's model it treats the concrete as a single phase material. The basis for the model is shown in Fig. 5.7. The model parameters are defined as follows:

σ_i = the interface strength

τ_i = the interface shear stress

σ_s = the closing pressure

a_c = the actual area of contact (function of displacements)

α_i = the asperity angle (function of displacements)

The interface shear stress, τ_i is related to σ_i through the coefficient of friction μ similarly to Walraven's model. The value of σ_s depends on the ultimate tensile strength of the concrete, and decreases exponentially for larger crack openings. The interface strength is related to f'_c through an equation similar to that developed for the parameter σ_{pu} in Walraven's model.

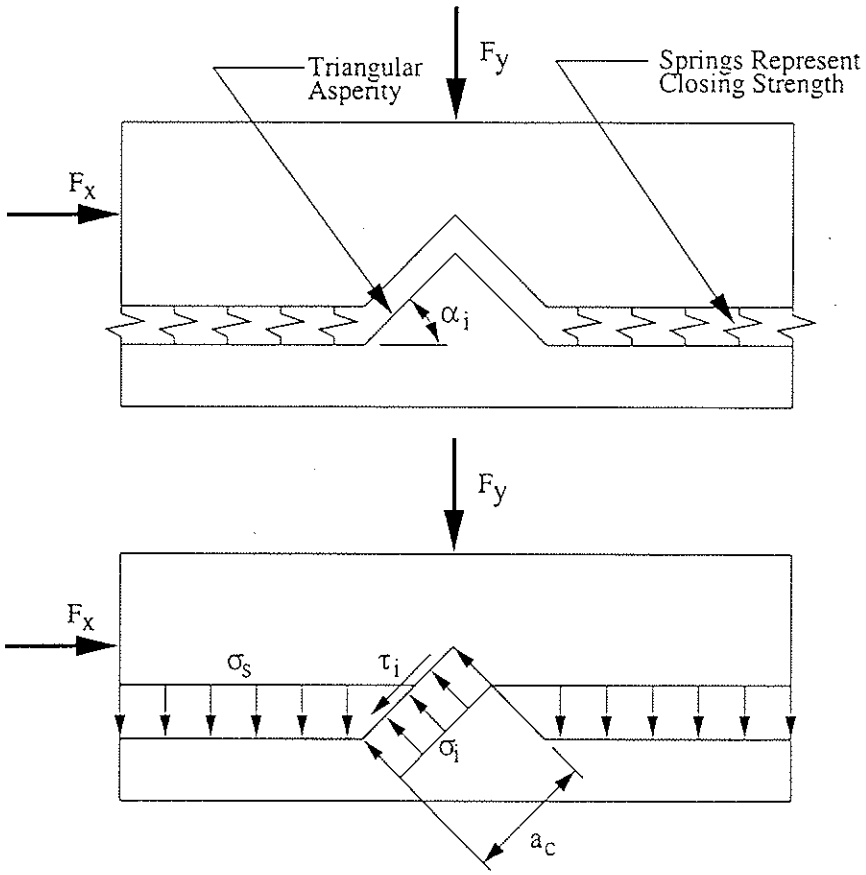


Figure 5.7: Equivalent Micromechanical Model of Crack (Divakar, 1992)

Writing the equations of equilibrium across the crack plane for a unit crack area gives

$$\sigma_t = \sigma_i a_c(\sin\alpha_i + \mu\cos\alpha_i) \tag{5.12}$$

$$\sigma_n = \sigma_i a_c(\cos\alpha_i - \mu\sin\alpha_i) - \sigma_s \tag{5.13}$$

where σ_n and σ_t represent the normal and tangential crack stresses corresponding to F_y and F_x acting over unit crack areas in Fig. 5.7.

Following this formal statement of the model, Divakar performed several regression analyses based on experimental data to determine expressions for α_i and a_c. The resulting equations were then verified by comparing computed values with existing constant normal stress and variable normal stress test data. In general, the model performed well, accurately predicting monotonic stress-displacement response.

Walraven (1994)

In a more recent study, Walraven [92] focused on the shear and normal stress transfer across cracks in concrete subjected to cyclic loading. The same physical and mechanical concepts developed in his initial study [91] were assumed for modeling the stress transfer; however, the calculations for the projected contact areas were modified.

The statistical basis for the particle distribution is identical to that developed previously by Walraven. To simplify calculations, both the distribution of particle diameters and all possible embedment depths were discretized into a finite number, with the embedment depth assuming to be uniformly distributed between a minimum of zero and a maximum of the radius of a given particle. This allows the representation of an aggregate particle D_{ij} with diameter D_i and embedment depth d_j as shown in Fig. 5.8. For each embedment/diameter combination, D_{ij}, the cement paste is discretized into a finite number of layers, allowing computation of the projected contact area by summing the contact areas of the individual paste layers. See Fig. 5.8 for a graphical representation of this discretization. For a given

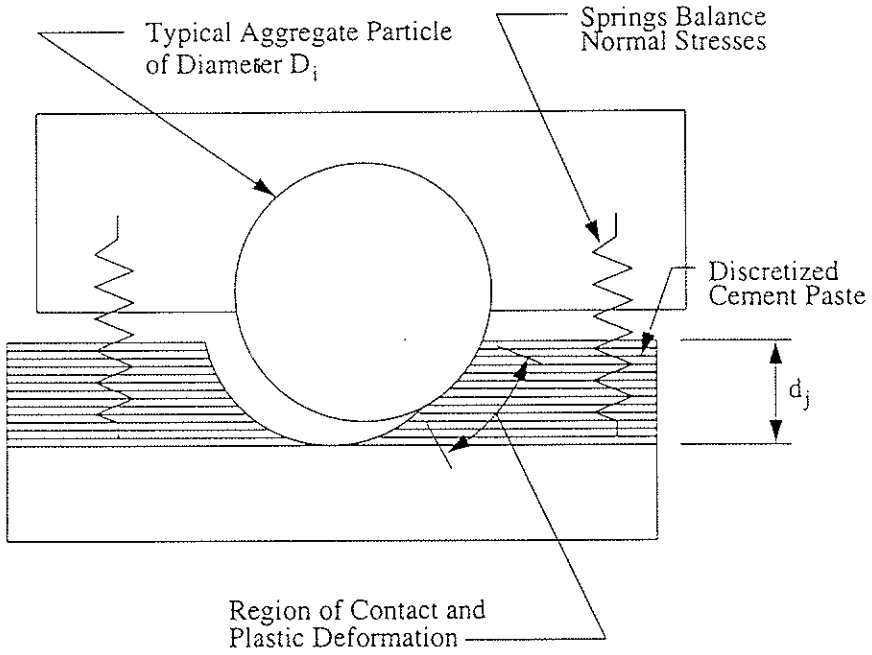


Figure 5.8: Discretization of Aggregate Particles and Cement Paste

normal and tangential displacement of the crack plane, the likely contact area for any particle diameter is determined by summing the contact areas for all embedments and multiplying by n_i, the likely number of occurrences of D_i. The total contact area is then determined by summing over all particle diameters, and the shear and normal stresses may

be computed using Eqs. 16 and 17.

The normal stresses generated across the crack are balanced by the springs assumed to cross the crack plane as shown in Fig. 5.8. These springs may represent some external restraint mechanism, or reinforcing bars intersecting the crack plane. The discretization of the cement paste also allows the damage to the cement paste to be updated throughout the loading history by tracking the geometry of the paste layers for each particle diameter/embedment depth combination. A simple program may be written to perform simulations of both monotonic and cyclic loading for different parameters such as initial crack opening, restraining spring stiffness, etc. Another further addition to the theory as initially presented in the context of monotonic loading is the inclusion of the fracture index, C_f. This is used to proportionally reduce both the shear and normal stress transferred across the crack, and accounts for the fracturing of aggregate particles. Walraven notes that the value of C_f is characteristic of the specific concrete mixture, and must be determined from monotonic testing of a specimen and curve-fitting the resulting data.

Verification of the theory was accomplished by comparing computed shear stress vs. displacements with those from existing laboratory tests found in the literature [55]. In general, agreement between the tests and the theory was very good, and the simulations accurately capture the significant physical phenomena observed from testing. In contrast with the model of Li et al [56], Walraven's model is capable of predicting different loading/unloading paths for adjacent cycles.

5.4 Proposed Modeling Technique

5.4.1 Choice of Aggregate Interlock Constitutive Model

Due to the generality and extensibility of the micromechanics-based constitutive models, they are the best choice for modeling aggregate interlock shear transfer. The model of Fardis [23] is the least physically defensible of the micromechanics-based constitutive models, and is also computationally complex. Its resolution of aggregate interlock shear transfer into global shear and normal forces and a moment acting on the entire crack plane is not ideal for rigid pavement modeling.

The remaining models are all viable for the modeling of aggregate interlock shear transfer in rigid pavements. However, the two-phase constitutive model developed by Walraven [91, 92] has been chosen to model aggregate interlock shear transfer. The model was selected for several reasons:

- Several researchers have independently shown that experimental results can be adequately described using this model with relatively few rational physical parameters. The model of Divakar relies more heavily on empirical data due to its conceptualization of the concrete as a single-phase medium. The model of Li [56] relies on assumptions about the statistical distribution of the crack surface asperities and contact area which have limited experimental verification.

- The model will allow reasonably accurate estimates to be made of the behavior of the joint. Although laboratory testing is required to determine optimal parameters, this is not necessary for initial implementation of the model and verification of its applicability.

- Unlike the model of Divakar [18], the model has been extended to cyclic loading, and cumulative damage at the joint is important due to the large number of load cycles experienced by concrete pavements. Current experimental research into this phenomenon has been undertaken by other researchers [3]; the development and calibration of analytical models that capture this effect would be a valuable topic for future research.

- The model is relatively simple to implement in its discretized form as presented in [92].

The model has been implemented and verified by duplicating some of the results published by Walraven. Details of the implementation are given in the following Section.

5.4.2 Incorporation of Constitutive Relations in Finite Element Model

Walraven's model will be incorporated in the finite element models via the generation of curves similar to those shown in Fig. 5.9 using appropriate parameters. Note that the stress state at any point on the joint surfaces can be determined given (δ_n, δ_t) by interpolating on and between curves. Similarly, numerical derivatives of these curves are easily evaluated with a finite difference equation to give required values for the constitutive matrix $\mathbf{D}$.

The approach used to generate these curves is similar to the discrete calculation proposed by Walraven [92]. However, Walraven assumed ten discrete particle diameters, paste layers, and embedment depths for his example calculations with no consideration that a more refined discretization might improve accuracy. To determine an appropriate number of discrete particle diameters, embedment depths, and paste layers to be used in the calculation of A_x and A_y, a small convergence study has been performed.

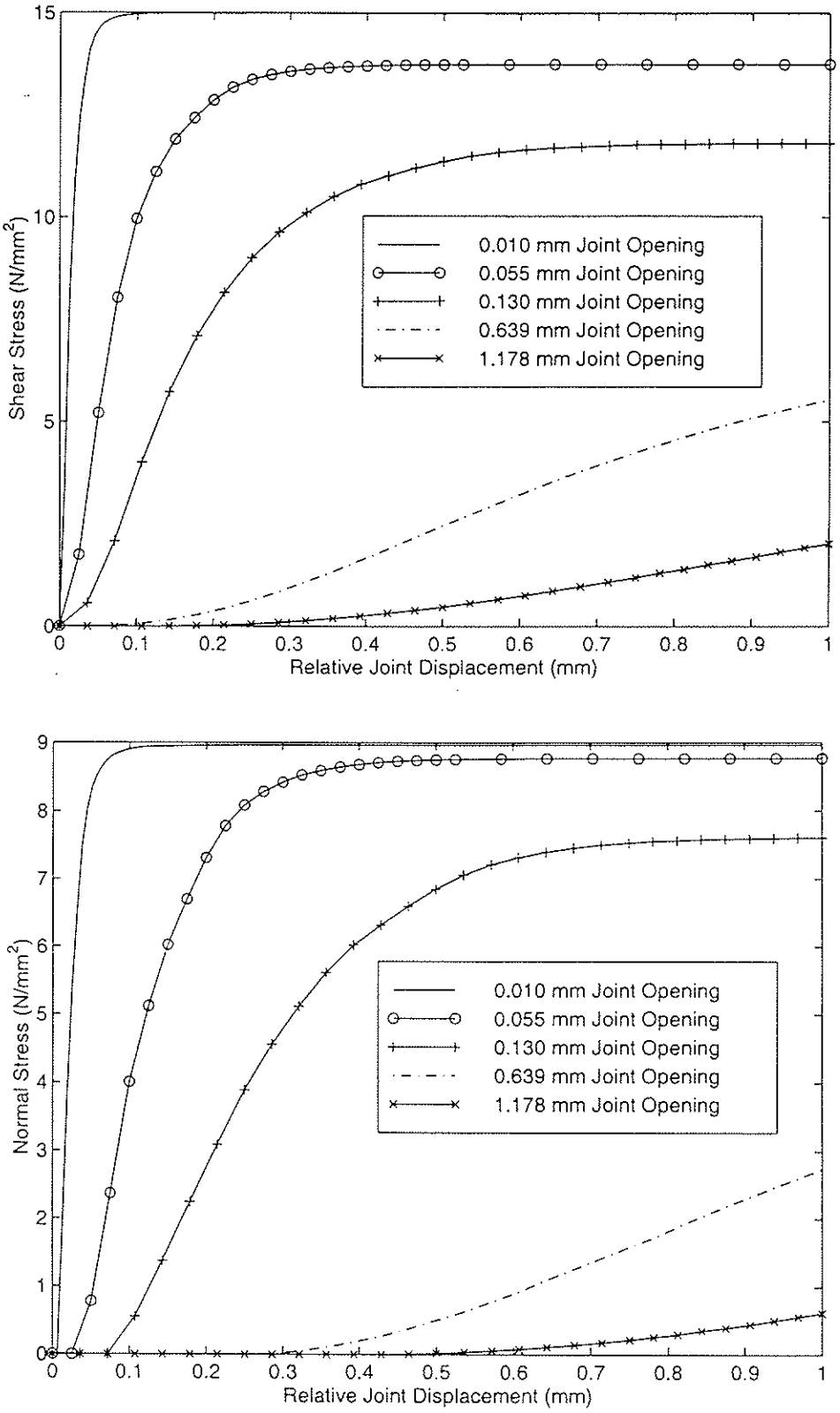


Figure 5.9: Shear and Normal Stresses vs. δ_t for Varying Crack Widths

Consider a concrete mix with the following typical properties: $\sigma_{pu} = 45$ N/mm^2, $\mu = 0.4$, a fraction of aggregate particles in the concrete of 0.75, and a maximum particle diameter of 20 mm. For simplicity, the number of diameters, embedments, and paste layers will be assumed equal. Computing σ_t vs. δ_t and σ_n vs. δ_t for varying levels of discretization yields the curves shown in Fig. 5.10 for a constant δ_n of 0.2 mm. Note that the calculation is convergent; however, the value of 10 diameters, layers, and embedments suggested by Walraven introduces significant error in the calculation of the shear and normal stresses. Based on this convergence study, the number of diameters, layers, and embedments was fixed at 30 for the remainder of this study.

5.4.3 Joint Interface Element

In the present investigation, an isoparametric interface element will be used to model aggregate interlock shear transfer. The particular element implemented is a zero-thickness, 16-noded quadratic element designed to deform compatibly with the 20-noded brick elements used to model the slab as shown in Fig. 5.11. The element stiffness can be formulated using a variational approach. The *incremental* internal virtual work, $d\Pi^e$, may be written as [98]:

$$d\Pi^e = \int_A d\boldsymbol{\delta}^{\mathrm{T}} d\boldsymbol{\sigma}\, dA \tag{5.14}$$

where $\boldsymbol{\delta}$ and $\boldsymbol{\sigma}$ are as defined previously. Assuming an arbitrary orientation for the shear direction, indicated by the vector $\mathbf{v}$ shown in Fig. 5.11, the components of $d\boldsymbol{\delta}$ may be written in terms of an incremental relative displacement vector at any point in the global coordinate system, $d\boldsymbol{\Delta}$:

$$d\boldsymbol{\delta} = \begin{bmatrix} \mathbf{n} \cdot d\boldsymbol{\Delta} \\ \mathbf{v} \cdot d\boldsymbol{\Delta} \end{bmatrix} = \mathrm{Q}\, d\boldsymbol{\Delta} \tag{5.15}$$

where Q is a 2×3 array with the unit vectors $\mathbf{n}$ and $\mathbf{v}$ as rows, i.e.:

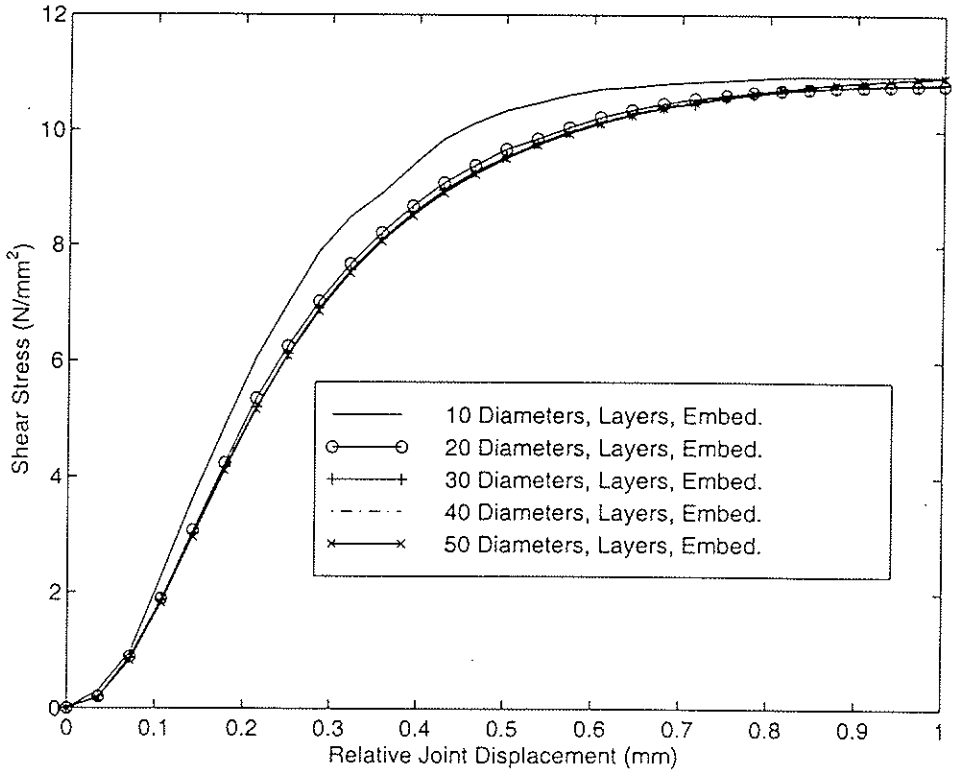
$$\mathrm{Q} = \begin{bmatrix} \mathbf{n} \\ \mathbf{v} \end{bmatrix} \tag{5.16}$$

Substituting Eq. 5.15 and the incremental stress-strain relationship $d\boldsymbol{\sigma} = \mathrm{D}\, d\boldsymbol{\delta}$ into Eq. 5.14 gives:

$$d\Pi^e = \int_A \left[\mathrm{Q}\, d\boldsymbol{\Delta}\right]^{\mathrm{T}} \left[\mathrm{D}\mathrm{Q}\, d\boldsymbol{\Delta}\right] dA = \int_A \left[d\boldsymbol{\Delta}^T \mathrm{Q}^{\mathrm{T}}\right] \left[\mathrm{D}\mathrm{Q}\, d\boldsymbol{\Delta}\right] dA \tag{5.17}$$

Expressing the vector of incremental relative nodal displacements, $d\boldsymbol{\Delta}$, in terms of the incremental element nodal displacement vector, $d\mathbf{U}^e$, gives:

$$d\boldsymbol{\Delta} = \mathrm{N}\mathrm{H}\, d\mathbf{U}^e = \mathrm{B}\, d\mathbf{U}^e \tag{5.18}$$



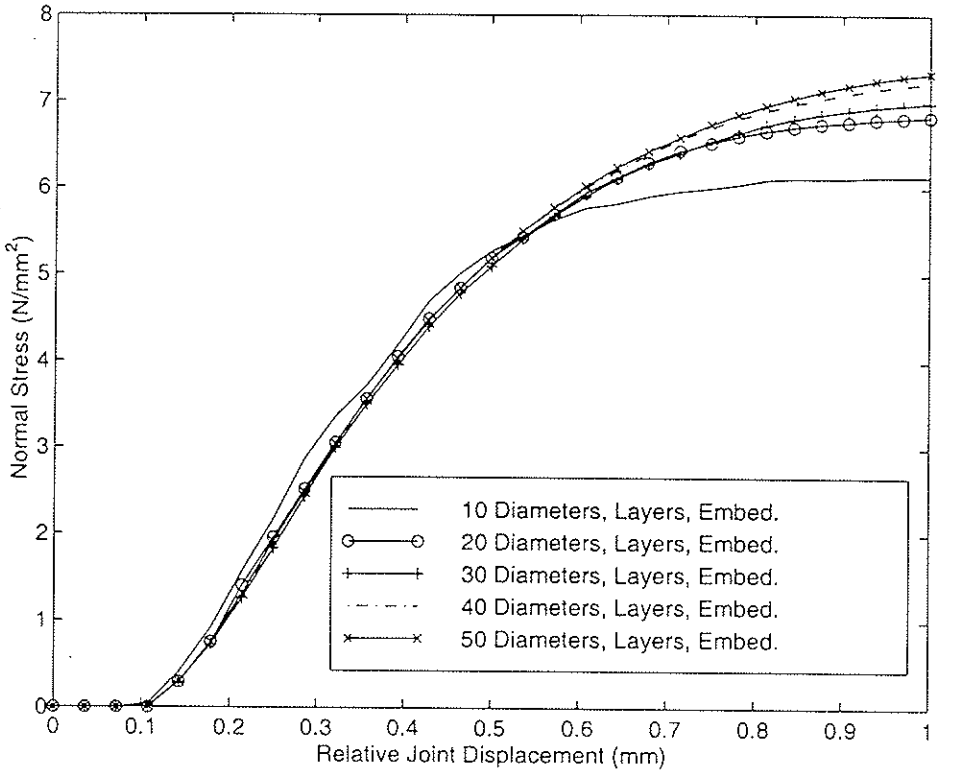


Figure 5.10: Normal and Shear Stress vs. δ_t for Varying Levels of Discretization

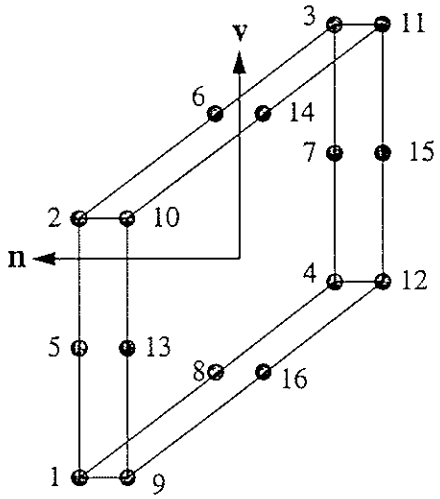


Figure 5.11: Quadratic Isoparametric Interface Element

where:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & \cdots & -1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & -1 & 0 & \cdots & 0 & 0 & 0 \\ & & & & & \textit{etc.} & & & & & \end{bmatrix} \tag{5.19}$$

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & \cdots & N_8 & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \cdots & 0 & N_8 & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & \cdots & 0 & 0 & N_8 \end{bmatrix} \tag{5.20}$$

$$\mathbf{U}^e = \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_{16} \end{bmatrix} \tag{5.21}$$

The matrix $\mathbf{H}$ is of size (24×48) and the product $\mathbf{H}d\mathbf{U}^e$ merely gives the relative incremental displacements at each node given the nodal displacement vector. $\mathbf{N}$ is of size (3×24), and is a matrix of standard 2D quadratic element shape functions for an 8-noded element [98] that interpolates the relative displacement at any point in the element from the nodal values. The matrix $\mathbf{B} = \mathbf{NH}$ is analogous to the matrix of differential operators common to continuum elements. Each vector $\mathbf{d}_i$ represents the 3-element displacement vector at element node i. Substituting 5.18 into 5.17 gives the element tangent stiffness matrix, $\mathbf{K}^e$:

$$\mathbf{K}^e = \int_A \mathbf{B}^{\mathrm{T}}\mathbf{Q}^{\mathrm{T}}\mathbf{D}\mathbf{Q}\mathbf{B}dA \tag{5.22}$$

The integral expression for $\mathbf{K}^e$ given in Eq. 5.22 is evaluated using 3×3 Gauss point integration performed in terms of the element local coordinates using the usual rules for

coordinate mapping [98]. The values of σ_n and σ_t are updated and stored at each integration point to allow the computation of the element nodal forces, $\mathbf{f}^e$ by the following integral:

$$\mathbf{f}^e = \int_A \mathbf{B}^{\mathrm{T}} \left[\mathbf{n}\sigma_n + \mathbf{v}\sigma_t\right] dA \tag{5.23}$$

5.4.4 Symmetry and Computational Issues

The final issue which must be addressed is the non-symmetry of the constitutive matrix predicted by Walraven's model. As noted in [25, 26], this may pose computational difficulties. Further, the efficient multigrid-preconditioned conjugate gradient solution techniques used in this study (see Chapter 6) may only be used with symmetric positive definite systems. The details of the proposed approach to overcoming this difficulty are presented in the remainder of this Section.

Consider the general form of the equilibrium statement for a finite element model:

$$\mathbf{P} - \mathbf{F} = 0 \tag{5.24}$$

where $\mathbf{P}$ is a constant vector of applied forces, and $\mathbf{F}$ is the vector of nodal forces determined by:

$$\mathbf{F} = \sum_{elements} \int_V \mathbf{B}^{\mathrm{T}} \sigma dV + \int_{A_j} \Gamma dA_j = \mathbf{F}_e + \mathbf{F}_j \tag{5.25}$$

The contribution to the nodal force vector, $\mathbf{F}$, by summation/assembly over the elements includes the slab, subgrade, and dowel elements. The contribution to $\mathbf{F}$ by joint shear and normal stress transfer is accounted for in $\mathbf{F}_j$, computed by separately integrating the tractions on the joint surfaces predicted by Walraven's model, Γ, over the area of the joint, A_j. The evaluation of $\mathbf{F}_j$ is accomplished by numerically integrating Eq. 5.23 for each joint element and summing/assembling.

The nonlinear solution strategy developed in Chapter 6 is similar to Newton's method in that it requires the formation of the Jacobian, $\mathbf{J}$, which is simply the tangent stiffness matrix:

$$\mathbf{J} = \frac{\partial \mathbf{F}_e}{\partial \mathbf{U}} + \frac{\partial \mathbf{F}_j}{\partial \mathbf{U}} = \mathbf{K}_e + \mathbf{K}_j \tag{5.26}$$

Strict evaluation of Eq. 5.26 results in an unsymmetric stiffness matrix, where $\mathbf{K}_e$ and $\mathbf{K}_j$ are the symmetric and unsymmetric contributions, respectively. However, we need not evaluate J exactly; in fact, approximations to J are often employed in quasi-Newton methods, and in modified Newton approaches J is not updated at every iteration. In the present study, a symmetric approximation to $\mathbf{K}_j$ will be used, denoted by $\mathbf{K}'_j$, which is computed by

including only the diagonal element of the crack constitutive matrix corresponding to the shear stress-displacement, D_{tt}.

To illustrate further, consider Algorithm 5.1, appropriate for the case where the only source of nonlinearity is aggregate interlock:

```
solve()
1  r = || P − F ||
2  while || r ||> 0.0
3  do K'_j = update_K(U)
4      U = U + (K_e + K'_j)^{-1} r
5      F_j = get_joint_forces(U)
6      F_e = get_element_forces(U)
7      r = P − F_j − F_e
8      end while
9  return
```

Algorithm 5.1: Solution of Model with Aggregate Interlock

This approach is easily included in a general nonlinear solution strategy based on Newton's method, and allows the use of efficient solution techniques that require symmetric positive definiteness. Preliminary studies indicate that it is feasible, most likely due to the small magnitudes of the normal stresses occuring at the joint.

5.5 Summary

In this Chapter, a comprehensive literature survey and investigation into the mechanics of aggregate interlock shear transfer has been presented and techniques for the incorporation of the constitutive relations predicted by these models in finite element discretizations have been reviewed. Based on this research, the micromechanics-based aggregate interlock model developed by Walraven [91, 92] was chosen for implementation in the present study. Crack constitutive relations are computed using a special program that produces shear stress and normal stress versus shear displacement curves for various joint openinings. Interpolation within and between curves allows the state of stress on the crack faces to be defined for any relative crack displacement, and numerical differentiation of the computed relations permits calculation of the required tangent crack stiffness. A 16-noded isoparametric joint element is used to incorporate the crack constitutive relations in the finite element models; details of its development have also been presented.

Chapter 6

Solution Strategies

6.1 Introduction

In Chapter 3, a finite element modeling strategy for rigid pavement systems was outlined and the following three major goals of this study were identified:

- Develop better methods for modeling dowel load transfer.
- Improve techniques for modeling aggregate interlock shear transfer.
- Develop computationally efficient solution strategies for solving 3D finite element models of rigid pavement systems.

Discretization of the slab and subgrade, constitutive properties, and application of loads were also addressed in Chapter 3; Chapters 4 and 5 covered dowel and aggregate interlock modeling, respectively. At this point, all aspects of the proposed finite element models of rigid pavement systems have been defined. Fig. 6.1 shows a typical system of two rigid pavement slabs with all components of the system identified; Fig. 6.2 illustrates a typical finite element mesh of such a system.

The focus of this Chapter is on achieving the third major goal of this study: the development of efficient and robust solution strategies for finite element models of rigid pavement systems. This is a particularly challenging undertaking for several reasons:

- *Material nonlinearities.* The proposed aggregate interlock constitutive relations are nonlinear. In addition, while gaps between the dowels and the slabs give rise to contact nonlinearities, this phenomenon is actually treated as a material nonlinearity due to the embedded formulation of the dowel element.

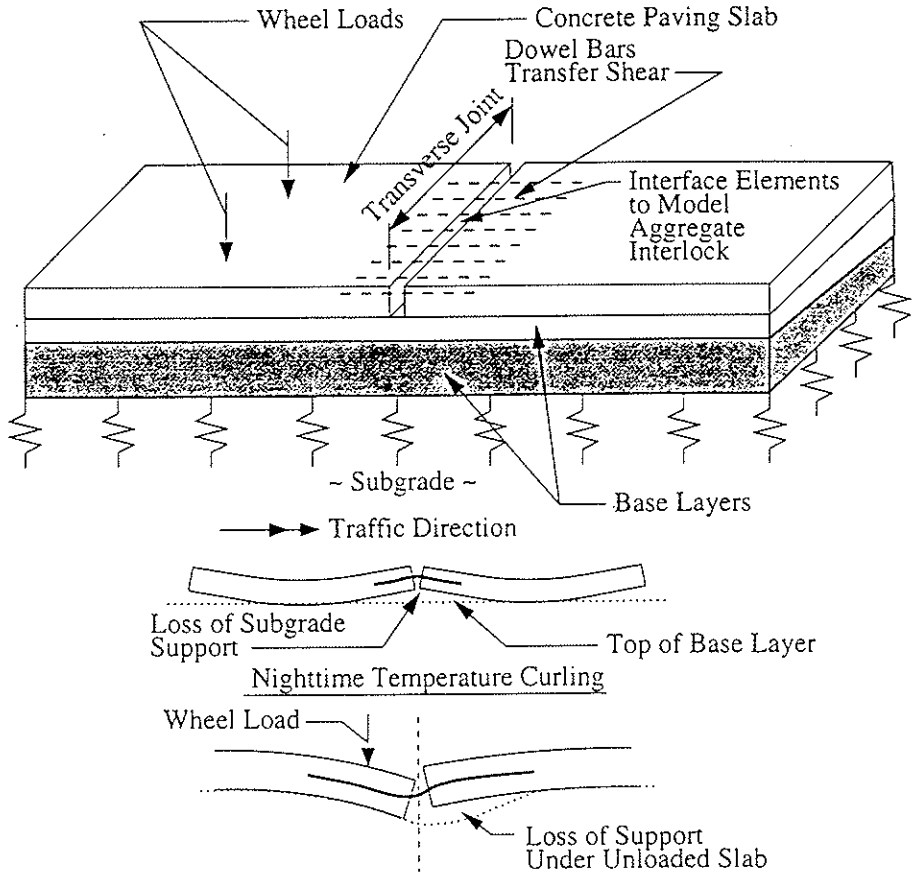


Figure 6.1: System of Two Rigid Pavement Slabs

- *Contact nonlinearities.* Separation of the slab and subgrade under temperature and wheel loading (see Figs. 6.1 and 6.3) must be treated as a contact nonlinearity.

- *Model size.* The proposed 3D finite element models are large, dictating that memory and computational requirements of the solution strategy be minimized.

The following Section addresses the first two difficulties by the development of a global solution strategy which allows for general material and contact nonlinearities. The primary source of computational effort in this nonlinear solution strategy is the solution of a linearized system of equations in the inner kernel, the optimization of which is crucial to efficiency. This is accomplished by the use of an iterative multigrid-preconditioned conjugate solver, discussed in Section 6.3. Section 6.4 presents the development of the multigrid methods used to precondition the conjugate gradient iteration. In Section 6.5, performance studies testing important aspects of the proposed multigrid and multigrid-preconditioned conjugate gradient solution techniques are presented. In Section 6.6 the effectiveness of the

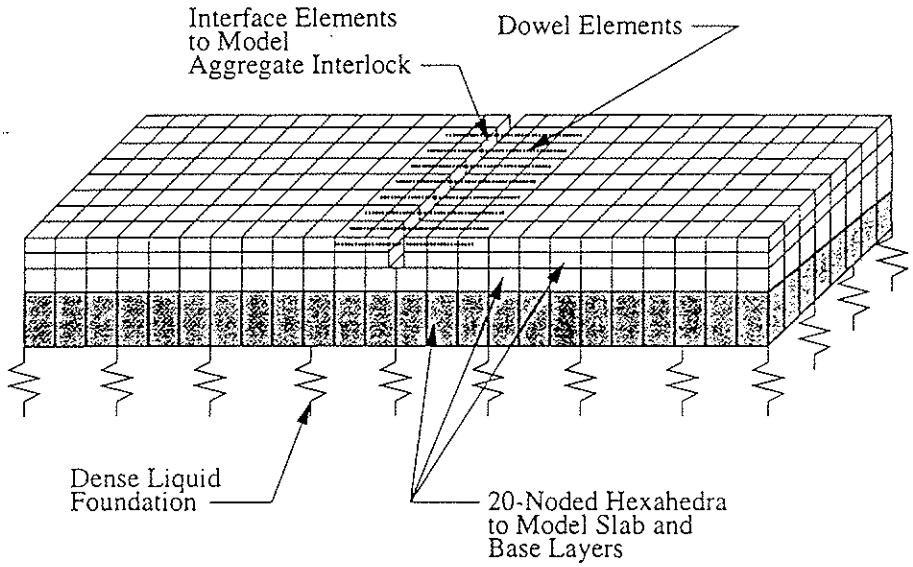


Figure 6.2: FE Mesh of System of Two Rigid Pavement Slabs

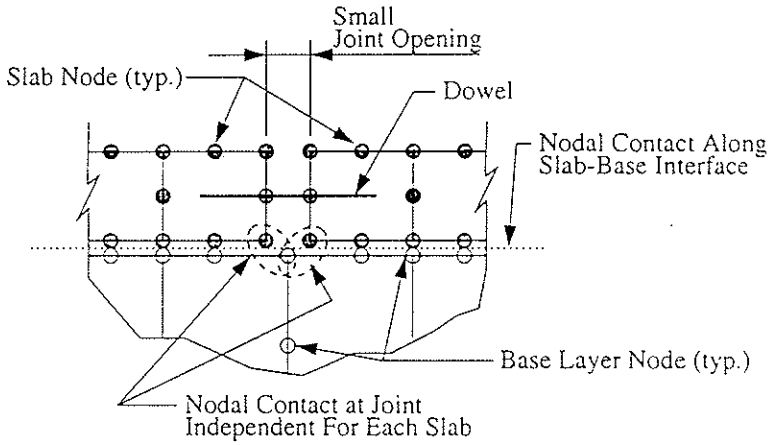


Figure 6.3: Detail of Mesh Near Joint Showing Double Nodes at Slab-Base Layer Interface

global solution strategy is evaluated using a model of a typical rigid pavement system incorporating several base layers and slab-subgrade separation. Section 6.7 presents a summary of this Chapter.

6.2 Global Solution Strategy

The global solution strategy must account for both material and contact nonlinearities. Here, an algorithm is developed that is analogous to a full Newton iteration: it is based on successive linearizations of the constrained system that employ a tangent system stiffness matrix and solution for a displacement increment at each iteration. Slab/subgrade

separation is modeled assuming frictionless nodal contact.

6.2.1 Formulation of the Nonlinear Nodal Contact Problem

For frictionless nodal contact problems, inequality constraints are required. To illustrate, consider Fig. 6.4 which depicts a portion of a larger finite element mesh. Two surfaces (the

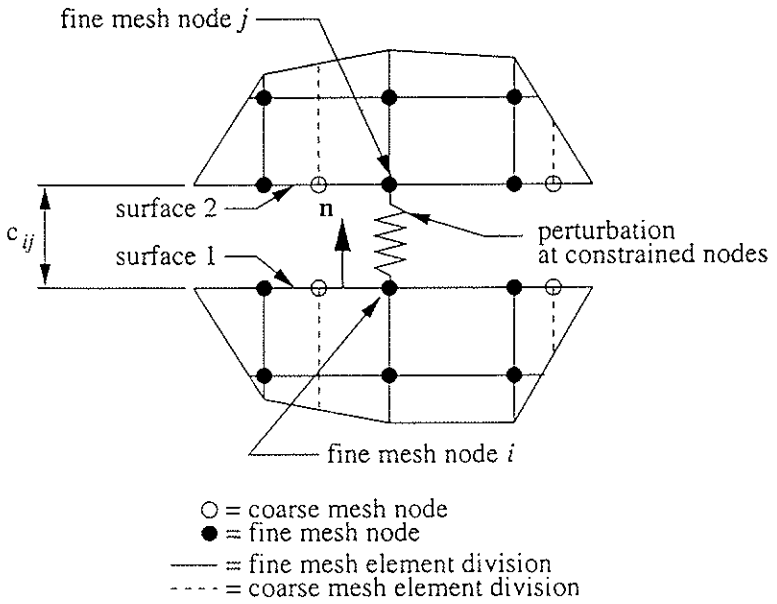


Figure 6.4: Nodal Contact

slab and base layer, for example) are separated by a scalar distance c_{ij}; the outward normal for surface 1 is given by $\mathbf{n}$. Mathematically the constraint can be expressed as:

$$(\mathbf{d}_i - \mathbf{d}_j) \cdot \mathbf{n} \leq c_{ij} \tag{6.1}$$

where $\mathbf{d}_i$ and $\mathbf{d}_j$ are nodal displacement vectors. Eq. 6.1 merely states that surface 1 may not interpenetrate surface 2 at the location of nodes i and j.

Inequality constraints require that appropriate criteria for maintaining/releasing constraints be employed. In the case of frictionless contact constraints, the criteria are straightforward: for a pair of currently unconstrained nodes, if Eq. 6.1 is false, the nodes are constrained; for a pair of constrained nodes, they must be released when:

$$\mathbf{n} \cdot (\boldsymbol{\sigma} \cdot \mathbf{n}) > 0 \tag{6.2}$$

where $\boldsymbol{\sigma}$ is the average of the stress tensors at nodes i and j.

When constraints are imposed and Newton's method is used to solve the materially nonlinear system, solution of the following constrained system is required at any iteration

k:

$$\mathbf{K}^k \delta \mathbf{U}^k = \mathbf{r}^k \tag{6.3}$$

$$\text{s.t. } \mathbf{G}^{k\mathrm{T}} \mathbf{U} = \mathbf{c}^k \tag{6.4}$$

where:

$$\begin{aligned} \delta \mathbf{U}^k &= \text{displacement increment to be solved for} \\ \mathbf{r}^k &= \text{the current residual} \end{aligned}$$

The tangent system stiffness matrix, $\mathbf{K}^k$, constraint matrix, $\mathbf{G}^k$, and $\mathbf{c}^k$ are superscripted as they change at each iteration due to material nonlinearity and constraint updating. Recognizing that the current solution vector, $\mathbf{U}$, may be written as:

$$\mathbf{U}^k = \mathbf{U}^0 + \delta \mathbf{U}^1 + \ldots + \delta \mathbf{U}^k \tag{6.5}$$

the constraints may be expressed as:

$$\mathbf{G}^{k\mathrm{T}} \left(\mathbf{U}^{k-1} + \delta \mathbf{U}^k \right) = \mathbf{c} \tag{6.6}$$

or:

$$\mathbf{G}^{k\mathrm{T}} \delta \mathbf{U}^k = \mathbf{c}^{*k} \tag{6.7}$$

where:

$$\mathbf{c}^{*k} = \mathbf{c}^k - \mathbf{G}^{k\mathrm{T}} \mathbf{U}^{k-1} \tag{6.8}$$

The constrained system given by Eqs. 6.3 and 6.4 may now be rewritten with the constraints expressed in terms of the displacement increment as:

$$\mathbf{K}^k \delta \mathbf{U}^k = \mathbf{r}^k \tag{6.9}$$

$$\text{s.t. } \mathbf{G}^{k\mathrm{T}} \delta \mathbf{U}^k = \mathbf{c}^{*k} \tag{6.10}$$

6.2.2 Solving the Constrained System

Several alternatives are available for enforcing constraints, the most common of which are penalty parameters and Lagrange multipliers [6]. Application of penalty parameters to Eqs. 6.9 and 6.10 leads to the following system:

$$\left(\mathbf{K}^k + \alpha \mathbf{G}^k \mathbf{G}^{k\mathrm{T}} \right) \delta \mathbf{U}^k = \mathbf{r}^k + \alpha \mathbf{G}^k \mathbf{c}^{*k} \tag{6.11}$$

where the penalty parameter, α, is chosen to be large enough to ensure that the constraints are satisfied, i.e. $\alpha \gg max(K_{ii})$. There are several problems associated with the penalty parameter method, however:

- The appropriate magnitude of α must be determined for a particular problem.
- Due to the size of α, the perturbed system stiffness matrix becomes ill-conditioned.

In spite of these problems, the penalty parameter method may be quite attractive when using direct methods (factorization) to solve for $\delta \mathrm{U}^k$ because the bandwidth, size, and positive definiteness of the original system of equations is maintained. However, the problems posed in this study are too large to be solved by direct methods; since iterative methods will be used to solve the systems of linear equations, the problem of ill-conditioning makes the penalty parameter infeasible.

In the present study, Lagrange multipliers are used to enforce the constraints in lieu of a penalty parameter approach to avoid problems associated with poor conditioning. Enforcement of the constraints with Lagrange multipliers leads to the expanded system:

$$\begin{bmatrix} \mathrm{K}^k & \mathrm{G}^k \\ \mathrm{G}^{k\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \delta \mathrm{U}^k \\ \delta \lambda^k \end{bmatrix} = \begin{bmatrix} \mathrm{r}^k \\ \mathrm{c}^{*k} \end{bmatrix} \tag{6.12}$$

which is expressed in terms of both incremental displacements and Lagrange multipliers. The solution of Eq. 6.12 for $\delta \mathrm{U}^k$ and $\delta \lambda^k$ (which satisfy Eqs. 6.9 and 6.10) is analogous to the solution of a linear system within a Newton iteration.

6.2.3 Solution Strategy

Clearly, if the solution of 6.11 by direct methods is not feasible due to memory and computational requirements, the solution of 6.12 by factorization of the coefficient matrix is not possible. This implies that an iterative solution method must be used. However, while the coefficient matrix in Eq. 6.12 is symmetric, it is not positive definite; this precludes the direct implementation of most multigrid methods which rely on standard iterations requiring positive definiteness (Gauss-Seidel, SSOR). This fact also prevents the direct application of the conjugate gradient method, which is probably the most widely used iterative technique for symmetric, positive-definite problems. While other iterative methods are available that do not require symmetry or positive definiteness (bi-conjugate gradient, generalized minimum residual, etc. [77]), their efficiency depends greatly on preconditioning the system 6.12, which is made difficult by its lack of positive definiteness.

To circumvent these problems, a block iteration will be used that reduces the problem one step further to the solution of a symmetric positive definite system at each iteration. In particular, an adaptation of Uzawa's method will be employed as detailed in [77, 99]. Note that $\mathbf{K}^k$ itself is singular, since the only vertical support provided to the slab is through contact with the upper base layer, and methods based on the Schur complement of $\mathbf{K}^k$, $\mathbf{G}^{\mathrm{T}}(\mathbf{K}^k)^{-1}\mathbf{G}$ are not feasible. The variation of Uzawa's method employed here requires that Eq. 6.12 be modified by the following perturbation:

$$\begin{bmatrix} \mathbf{K}^k + \mathbf{K}' & \mathbf{G}^k \\ \mathbf{G}^{k\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \delta\mathbf{U}^k \\ \delta\lambda^k \end{bmatrix} = \begin{bmatrix} \mathrm{r}^k + \omega\mathbf{G}^k\mathrm{c}^{*k} \\ \mathrm{c}^{*k} \end{bmatrix} \tag{6.13}$$

where $\mathbf{K}' = \omega\mathbf{G}^k\mathbf{G}^{k\mathrm{T}}$ is a perturbation to the system stiffness matrix ($\mathbf{K}^k + \mathbf{K}'$ is nonsingular). Note that once the constraints are satisfied, the perturbation does not modify the original system. The scalar perturbation parameter, ω, is mathematically equivalent to a penalty parameter, and may be interpreted as a spring stiffness. However, it is typically chosen to be a relatively small value so as not to cause numerical difficulties. The perturbed system is then solved iteratively as follows:

$$\mathrm{y} = \mathbf{G}^{k\mathrm{T}}\delta\mathbf{U} - \mathrm{c}^{*k} \tag{6.14}$$

$$\delta\lambda = \delta\lambda + \omega\mathrm{y} \tag{6.15}$$

$$\delta\mathbf{U} = (\mathbf{K}^k + \mathbf{K}')^{-1}\left((\mathrm{r}^k + \omega\mathbf{G}^k\mathrm{c}^{*k}) - \mathbf{G}^k\delta\lambda\right) \tag{6.16}$$

For convergence, ω must be sufficiently small [99]; numerical studies indicate that if ω is the same order as $avg(\mathbf{K}^k_{ii})$, convergence is not impaired and is fairly rapid. More complex perturbations can be employed in an attempt to speed convergence, but in general the simple approach presented is sufficient.

Algorithm 6.1 gives the global solution strategy employed in this study. Note that it employs three levels of iteration: (1) at the outer level, the nodal constraints and tangent system stiffness matrix (appropriate for use in a Newton iteration) are updated; (2) the constrained problem is solved using Uzawa's method; and (3) the innermost kernel requires

an iterative solution of the resultant linear system.

```
solve(U^0, r^1)
 1  k = 1
 2  ▷unchanging constraints and small residual error
 3  while (ε < tol_ε) and (G^k ≠ G^{k-1})
 4  do c^{*k} = c^k − G^{kT} U^{k-1}
 5     K' = ω G^k G^{kT}
 6     ▷update all stiffness matrices
 7     update_K(K^k)
 8     ▷Uzawa iteration - satisfy constraints
 9     while (φ < tol_φ)
10     do y = G^{kT} δU^k − c^{*k}
11        δλ = δλ + ωy
12        F = r^k + ωG^k c^{*k} − G^k δλ
13        ▷solve linear system
14        linear_solve(K^k + K', δU^k, F)
15        φ = ‖ G^{kT}(U^{k-1} + δU^k) − c^k ‖_2
16        end while
17     U^k = U^{k-1} + δU^k
18     λ^k = λ^{k-1} + δλ
19     k++
20     compute_stresses()
21     update_constraints(G^k)
22     compute_residual(r^k)
23     ε = ‖r^k‖_2 / ‖P‖_2
24  return
```

Algorithm 6.1: Solution of Nonlinear Constrained System

6.3 Conjugate Gradient Solver

The inner kernel of the nonlinear solution strategy reduces to the solution of Eq. 6.16. To simplify notation, the following description will be based on the solution of the typical symmetric positive definite system:

$$\mathbf{KU} = \mathbf{P} \tag{6.17}$$

Of course, in the nonlinear solution, $\mathbf{K}$ and $\mathbf{P}$ are augmented as in Eq. 6.16, and $\mathbf{U}$ is replaced by a displacement increment, $\delta\mathbf{U}$. As noted previously, direct methods for the solution of Eq. 6.17, based on Gauss elimination, are not feasible due the large amount of memory required to store the matrix factorizations. Further, as the size of the problem increases, the number of operations required to factor the matrix increases superlinearly with

the number of unknowns, making factorization techniques inefficient for large problems. To circumvent these problems, iterative solution techniques must be employed.

One of the most effective iterative solution techniques for symmetric positive definite (SPD) systems such as Eq. 6.17 is the conjugate gradient method. Details of the development and basis of the conjugate gradient algorithm will not be presented here, but may be found in many sources [29, 77]. The basic preconditioned conjugate gradient algorithm is given in Algorithm 6.2.

The effectiveness of the conjugate gradient method is largely dependent on preconditioning the original system of equations, which is accomplished on line 12 of Algorithm 6.2. Note here that the matrix M is an approximation to the system stiffness matrix, K. Many alternatives are available for preconditioning; the simplest rely on standard iterations, such as Jacobi or symmetric SOR. However, these are generally not effective for large problems [77]. Another alternative is to use an incomplete factorization for the system stiffness matrix, i.e. compute M as:

$$\mathbf{M} = \mathbf{L}'\mathbf{U}' \tag{6.18}$$

where $\mathbf{L}'$ and $\mathbf{U}'$ have been determined assuming some specified degree of fill-in relative to K, often fixed at zero. Incomplete LU factorization (or Choleski factorization for symmetric positive definite systems) is a viable preconditioner for large problems, although it requires extra memory for the storage of $\mathbf{L}'$ and $\mathbf{U}'$. The effectiveness of zero fill-in incomplete Choleski preconditioned conjugate gradient (ILU-PCG) will be briefly examined later in

this Chapter to provide baseline performance data.

```
solve(U^0, P)
 1  r^0 = P - K U^0
 2  z^0 = M^{-1} r^0
 3  ▷get initial search direction
 4  p^0 = z^0
 5  k = 0
 6  while || r ||_2 > 0
 7  do w^k = K p^k
 8     s^k = (U^k, r^k)/(p^k, w^k)
 9     U^{k+1} = U^k + s^k p^k
10     r^{k+1} = r^k - s^k w^k
11     ▷preconditioning step
12     z^{k+1} = M^{-1} r^{k+1}
13     ▷compute new search direction
14     β^k = (z^{k+1}, r^{k+1})/(z^k, r^k)
15     p^{k+1} = z^{k+1} + β^k p^k
16     k^{++}
17  return
```

Algorithm 6.2: Conjugate Gradient Iteration

Another preconditioning alternative is to approximately solve Eq. 6.17 with an iterative multigrid method, leading to the multigrid-preconditioned conjugate gradient method (MG-PCG). MG-PCG has been shown to be an effective strategy for solving various problems, including discretizations of the semiconductor equations [63] and groundwater flow simulations with spatially varying conductivities [4]. More recently, this idea was extended to the preconditioning of the generalized minimum residual and bi-conjugate gradient algorithms [77] for the solution of singularly perturbed unsymmetric problems [68]. The solution of nonlinear heat conduction problems with large discontinuities in material properties using a multigrid preconditioned Newton-Krylov method was investigated in [75].

Multigrid methods are themselves iterative solution techniques which rely on a hierarchy of discretizations of the domain coupled with standard iterations – typically Jacobi, Gauss-Seidel, or successive over-relaxation (SOR) – to achieve a solution. The effectiveness of MG-PCG for solving finite element discretizations in structural mechanics (such as the rigid pavement systems presently of interest) hinges on the development of a general multigrid solver, which is addressed in the following Section.

6.4 Multigrid Methods

6.4.1 Review of Multigrid Methods

To solve a SPD system of equations such as that of Eq. 6.17 (or equivalently, precondition a linear system within a MG-PCG iteration), the multigrid (MG) method relies on a few applications of a standard iterative technique, called smoothing, which is intended to reduce the high frequency components of the solution error, $\mathbf{e}$, coupled with coarse grid approximations to the smoothed error [8].

The error in the solution, $\mathbf{e} = \mathbf{U}^* - \mathbf{U}$, is related to the residual, $\mathbf{r}$, by $\mathbf{r} = \mathbf{Ke}$. Here $\mathbf{U}^*$ denotes the exact (unknown) solution and $\mathbf{r}$ is computed as:

$$\mathbf{r} = \mathbf{P} - \mathbf{KU} \tag{6.19}$$

Using these these definitions, the general, recursive multigrid algorithm is as shown in Algorithm 6.3 [47]:

```
multi_grid(level, r, e)
 1  if level = 1
 2     then ▷direct solution at coarsest level
 3          e = K^{-1} r
 4     else for i = 1, ..., β
 5          do
 6             e = smooth(e, r)  ▷ pre-smoothing
 7             r = r − K_level e
 8             r_{level−1} = R_level r
 9             ▷recursive call to next coarsest level
10             e = multi_grid(level − 1, r_{level−1}, e_{level−1})
11             e = T_level e_{level−1}
12             e = smooth(e, r)  ▷ post-smoothing
13             end for
14          end if
15  return
```

Algorithm 6.3: General Recursive Multigrid Algorithm

In the above algorithm, the operator $\mathbf{R}$ restricts from the fine mesh to the coarse mesh, and $\mathbf{T}$ interpolates the coarse mesh error to the fine mesh. This subroutine is placed in an outer loop, and called repeatedly until an appropriate convergence criteria is met. If the inner loop limit, β, is greater than one, the algorithm corresponds to a 'W-cycle', where multiple processes of smoothing/restricting and interpolating/smoothing at coarser levels

are implemented. A 'V-cycle' is implied by $\beta = 1$, where smoothing and restricting are applied sequentially from the finest to the coarsest grid followed by sequential interpolation and smoothing back to the finest grid.

Early research on multigrid methods focused on two-dimensional (2D) finite difference approximations to general elliptic boundary value problems using regular grids [8]. When used to solve such regular finite difference discretizations, multigrid methods require an amount of work proportional to the number of unknowns, making them very attractive for large problems [8]. Also note that the definitions for $\mathbf{R}$ and $\mathbf{T}$ are simplified by a regular grid structure.

Recently, multigrid techniques have begun to find widespread application in finite element modeling of structures and solid mechanics problems. Much of the groundwork for the application of multigrid techniques in solid mechanics was performed by Parsons and Hall [70, 71]. In these studies, the convergence properties of the multigrid method were investigated for structured 2D and 3D meshes. Important parameters were identified, for which optimal values were determined; in addition, both Gauss-Seidel and conjugate gradient methods were investigated as possible iterative smoothers. Another application of multigrid techniques to the 3D finite element modeling of structures may be found in [67], where a two grid solver using a conjugate gradient smoother was developed. The smoother was implemented on an element-by-element basis, eliminating the need to assemble the system stiffness matrices, and making the multigrid method more amenable to parallel processing. More recently, the use of unstructured meshes with poorly conditioned problems arising from the discretization of shell structures was addressed in [27]. To overcome the inherent poor conditioning of shell discretizations, a modified Incomplete Choleski Factorization smoother was employed, along with weighted acceleration schemes based on a line-search minimization of potential energy. In [58], unstructured multigrid methods were developed for 2D discretizations of the Poisson problem with a variable time-stepping Jacobi smoother intended to alleviate poor conditioning.

Other, more indirect applications of the multigrid method may also be found in the literature. For example, in [24] a conjugate gradient preconditioner based on a nested coarse mesh approximation was developed that can be interpreted as a multigrid method employing a conjugate gradient smoother. Multigrid methods in solid mechanics have also been integrated with adaptive solution techniques, where a series of meshes of increasing refinement are generated automatically based on error estimates [50].

Despite the progress in developing multigrid methods for 3D finite element modeling of structures and solid mechanics problems, there are still many issues which must be addressed before they can be used routinely and reliably for the solution of complex structures such as the rigid pavement systems of interest in this study.

- Multigrid methods should be generalized to allow multiple element types and varying numbers of nodal degrees of freedom within the same model. This is clearly of interest in the present study which requires that dowel elements be meshed with solid elements.

- The use of unstructured sequences of meshes should be allowed. It is often impractical in 3D to generate nested sequences of meshes due to constraints on element aspect ratios and limits on the problem size. When local mesh refinement is required or complex geometries are considered, this problem becomes even more difficult as automatic mesh generation codes do not readily generate nested sequences of meshes.

- Multigrid methods need to be generalized to handle nonlinear problems with contact constraints. Contact constraints cause difficulties because of the need for defining appropriate intergrid restriction and interpolation operators for them.

- There is a need for multigrid methods that can be conveniently integrated with conventional, displacement-based finite element methods to avoid expensive replication of existing FE codes.

This section presents the development of multigrid methods that fulfill these needs. Only V-cycle multigrid will be used ($\beta = 1$ in Alg. 6.3); as shown by Parsons and Hall [70], there is not a significant difference in computational effort between the V-cycle and the W-cycle for well-conditioned problems without singularities. Point-wise symmetric Gauss-Seidel smoothing will be employed in all analyses – note that a symmetric smoother is required when using multigrid as a preconditioner within a conjugate gradient iteration. Coarse grid solutions are achieved using SuperLU [17, 57], a state-of-the-art sparse direct solver. General methods for the intergrid transfer of information are developed, including restriction and interpolation operators and techniques for handling spatially varying materials. Details of the searching algorithms required to allow the efficient implementation of these techniques are also discussed. Finally, the incorporation of contact nonlinearities required for the present problem is presented.

6.4.2 Intergrid Transfer of Information

When developing any multigrid method, methods for the intergrid transfer of information, including consistent and computationally efficient definitions of $\mathbf{R}$ and $\mathbf{T}$, are of primary importance. In particular, it is desirable that the intergrid operations uniformly handle different element types (volume, surface, and line elements) and different element topologies (tetrahedral, hexahedral, etc.) without any alignment constraints between successive meshes. The definitions presented in this section possess the following:

- $\mathbf{R}$ and $\mathbf{T}$ meet the virtual work requirement that $\mathbf{R} = \mathbf{T}^{\mathbf{T}}$.
- Both $\mathbf{R}$ and $\mathbf{T}$ are defined and applied in the same manner for any sequence of meshes, including unstructured and unnested meshes.
- $\mathbf{R}$ and $\mathbf{T}$ permit the use of multiple isoparametric element types within the same model, including the meshing of beam or plate elements with solids.
- $\mathbf{R}$ and $\mathbf{T}$ may be easily computed using subroutines available in existing finite element codes.

Work and Energy Considerations

The requirement that $\mathbf{R} = \mathbf{T}^{\mathbf{T}}$ is easily derived from the consideration that the work done by the external forces be equal on each mesh [70]; details will not be repeated here. Similarly, by requiring that the fine and coarse mesh strain energies be equal and using $\mathbf{R} = \mathbf{T}^{\mathbf{T}}$, it follows that:

$$\mathbf{U}^{f\mathbf{T}}\mathbf{K}^{f}\mathbf{U}^{f} = \mathbf{U}^{c\mathbf{T}}\left(\mathbf{T}^{\mathbf{T}}\mathbf{K}^{f}\mathbf{T}\right)\mathbf{U}^{c} \tag{6.20}$$

which defines the coarse mesh stiffness, $\mathbf{K}^{c^*}$, as:

$$\mathbf{K}^{c^*} = \mathbf{T}^{\mathbf{T}}\mathbf{K}^{f}\mathbf{T} = \mathbf{R}\mathbf{K}^{f}\mathbf{R}^{\mathbf{T}} \tag{6.21}$$

where the superscripts f and c denote fine and coarse, respectively. The computation of $\mathbf{K}^{c^*}$ is expensive for sequences of unstructured meshes, since it involves two (albeit sparse) matrix products. Alternatively, $\mathbf{K}^{c}$ may be assembled from the coarse mesh element stiffness matrices in the usual manner and used in lieu of $\mathbf{K}^{c^*}$. This is the approach used exclusively in this study to simplify incorporation of the proposed multigrid methods in existing finite element code; details are presented later in this section.

The Interpolation Operator

A natural way of determining $\mathbf{T}$ when using the finite element method is to use the element shape functions to interpolate the displacements from the coarse to the fine mesh. This also guarantees that the interpolation has the same degree of accuracy as the solution. Considering the general case, defining $\mathbf{T}$ for two unstructured meshes then involves the following as illustrated in Fig. 6.5:

- For every fine mesh node, the coarse mesh element it is located within is determined.
- The natural (local) coarse mesh element coordinates corresponding to the physical fine mesh nodal coordinates are computed for each node/element pair.
- The element shape functions evaluated at these local coordinates allow interpolation from the coarse mesh nodal values to the value at the fine mesh node.

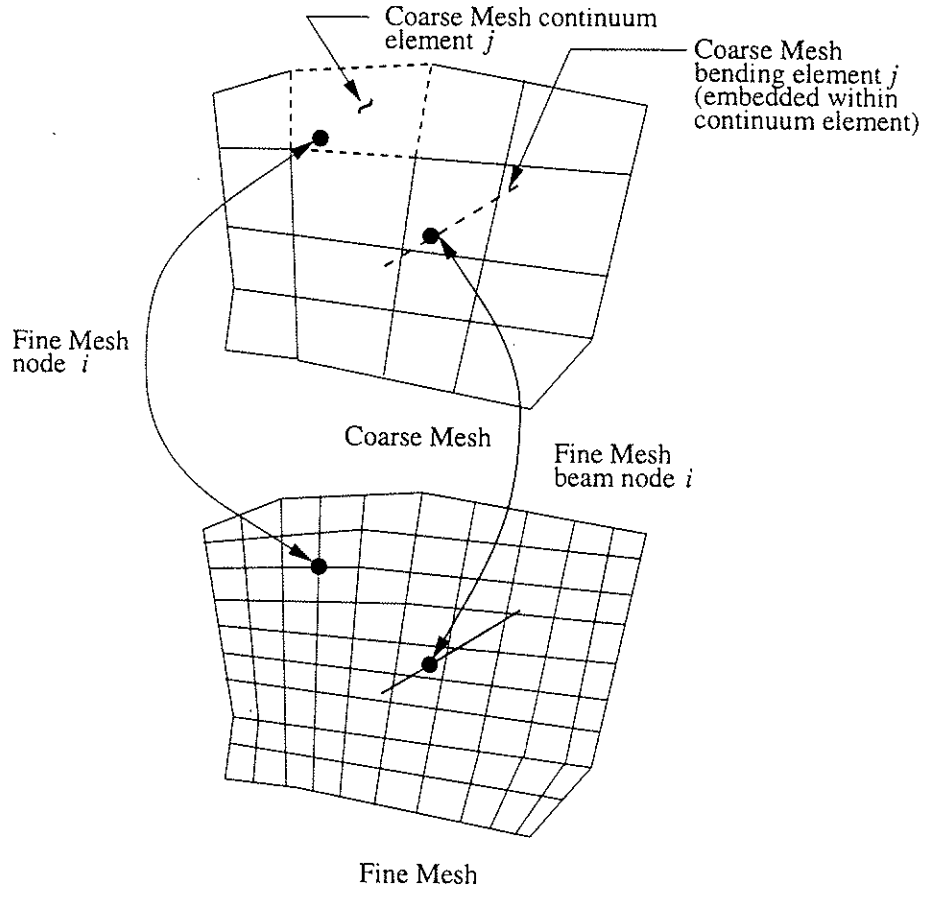


Figure 6.5: Interpolation and Restriction

Interpolation for each fine mesh node may be written as:

$$\mathbf{e}_i^f = \mathbf{N}_j^c \mathbf{e}_j^c \tag{6.22}$$

where $\mathbf{e}_j^c$ is the nodal error vector for coarse mesh element j, $\mathbf{N}_j^c$ is the usual array of shape functions for element j evaluated at fine mesh node i, and $\mathbf{e}_i^f$ represents the error vector at fine mesh node i. This process may also be written in matrix form for the entire mesh as:

$$\mathbf{e}^f = \mathbf{T}\mathbf{e}^c \tag{6.23}$$

where the element shape function matrices, $\mathbf{N}_j^c$, have been appropriately mapped to $\mathbf{T}$ which is of size n^f by n^c, where n denotes the number of nodal unknowns. In this study, the matrix representation of $\mathbf{T}$ is never formed explicitly, and the process of interpolation is carried out node-by-node as shown in Eq. 6.22 with $\mathbf{e}^f$ assembled from each $\mathbf{e}_i^f$. Note that this definition of $\mathbf{T}$ is quite general, allowing solid (or planar) elements of different type to be meshed together, as well as permitting the meshing of beam or plate elements with solid elements as shown in Fig. 6.5.

The Restriction Operator

Since the residual is actually a vector of unbalanced forces, $\mathbf{R}$ must be chosen so that it converts a fine mesh force vector to a statically equivalent force vector on the coarse mesh. This simple requirement is complicated by the use of general, unstructured meshes. Energy principles require that a residual force vector at a fine mesh node i, $\mathbf{r}_i^f$, applied at any point within a coarse mesh element j be converted to element nodal forces as:

$$\mathbf{r}_j^c = \mathbf{N}_j^{c\mathbf{T}} \mathbf{r}_i^f \tag{6.24}$$

where $\mathbf{r}_j^c$ represents the portion of $\mathbf{r}^c$ corresponding to the contribution from fine mesh node i, and $\mathbf{N}_j^{c\mathbf{T}}$ is the transpose of the matrix of coarse mesh element shape functions evaluated at the location of $\mathbf{r}_i^f$ [98]. It is apparent that the information required to form $\mathbf{R}$ is identical to that required for $\mathbf{T}$. Further, restriction may be represented by the following matrix multiplication:

$$\mathbf{r}^c = \mathbf{R}\mathbf{r}^f \tag{6.25}$$

In contrast to $\mathbf{T}$, the size of $\mathbf{R}$ is n^c by n^f. Clearly, this definition of $\mathbf{R}$ is consistent with the energy requirement that $\mathbf{R} = \mathbf{T}^{\mathbf{T}}$.

Searching Algorithm

A difficulty with the proposed definitions of $\mathbf{R}$ and $\mathbf{T}$ is the determination of the node/element pairs for general unstructured meshes. A naïve search would require $\mathcal{O}(n^f n^c)$ operations and be prohibitively expensive. To maintain the linear proportionality between n^f and solution time, this information must be determined with computational effort proportional to n^f. To achieve this goal, a searching algorithm based on a grid projection technique similar to that described in [58] is employed, which also proves useful for the intergrid transfer of material properties as shown later in this Section. Only elements with straight edges/surfaces are considered.

To illustrate the method with a 2D example, consider Fig. 6.6 which shows fine and coarse meshes, $\mathcal{M}^f$ and $\mathcal{M}^c$, discretizing the same physical domain. Superimposed over $\mathcal{M}^f$ and $\mathcal{M}^c$ is a rectangular grid, $\mathcal{G}$, defined by the maximum dimensions of the domain. For any element of $\mathcal{M}^c$ with straight edges, it is simple matter to determine what cells of $\mathcal{G}$

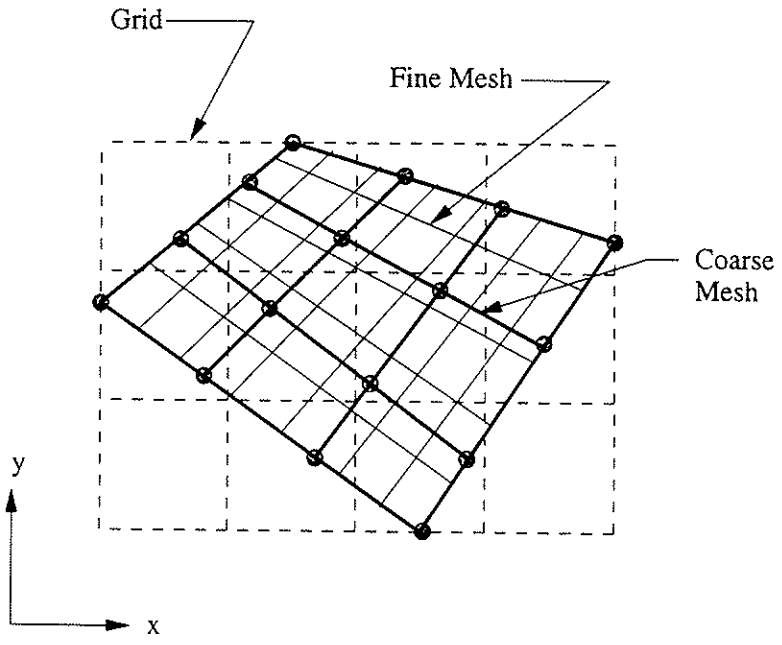


Figure 6.6: Grid Search

it overlaps by checking its maximum and minimum coordinates against the bounds of the cells. Given a list of elements of $\mathcal{M}^c$ for each cell in $\mathcal{G}$, the coarse mesh element containing any (x, y) point may be found by searching only the list of coarse mesh elements associated with the cell that the point lies within. The determination of whether or not a point lies within an element involves simple vector operations for an element with straight edges/faces;

the inclusion of general curved domains would require extension of the algorithm to handle the more complex geometry.

This searching procedure first requires a single loop over all elements in $\mathcal{M}^c$, followed by a single loop over all nodes n^f within which a small list of coarse mesh elements must be checked to determine which element n^f lies within. Thus, as long as the refinement of $\mathcal{G}$ follows the refinement of $\mathcal{M}^c$, the amount of work required is proportional to $n^f + n^c$. Further details on this grid searching strategy and its implementation may be found in Appendix A. Other geometric search techniques may also be used for building the correspondence between successive meshes [30].

The final issue which must be tackled is the determination of the coarse mesh element local coordinates corresponding to the fine mesh nodal locations. For a distorted isoparametric element, the determination of the local element coordinates requires the solution of a system of nonlinear equations, whose size is the number of spatial dimensions of the element [20]. Note that this problem has already arisen with the embedded dowel element presented in Chapter 4. Again, Newton's method is used to solve for the local element coordinates given the (x, y, z) nodal location; the Jacobian matrix needed for the iteration is simply the transpose of the Jacobian used in coordinate transformations during the integration of the element stiffness matrix. Given the element local coordinates, the element shape functions can then be evaluated when required for each instance of restriction and interpolation with minimal computational overhead. Details of the implementation may be found in Appendix A.

Determination of the Coarse Mesh Stiffness

As stated earlier, the coarse mesh stiffness matrices are determined by the usual assembly of the coarse mesh element stiffnesses. To maintain generality, it is assumed that the material properties can vary over the domain being modeled; this may be due to nonlinearities or simply differing linear materials used in a single model (i.e. multiple base layers below the slab). This variation in material properties is assumed to be defined in the usual fashion, element-by-element, but only on the fine mesh. This information must then be transferred to the coarse mesh elements to allow the assembly of $\mathbf{K}^c$. Computation of any coarse mesh element stiffness, $\mathbf{k}_i^c$, requires that constitutive properties be determined at the element integration points, which depend on the strain increment and the current stress state known only for the fine mesh. Using the searching algorithm detailed previously, the coarse mesh element integration points are located within the fine mesh elements, for

which the constitutive relations are defined. The computation of the strain increments and subsequent updating of the stress states at the coarse element integration points is then performed by the appropriate fine mesh elements. One potential problem with this technique must be noted: if the coarse mesh is not defined appropriately in regions where large changes in material properties occur, $\mathbf{K}^c$ may not provide a sufficiently accurate coarse grid approximation. This issue will be explored numerically with a model problem in the next Section.

Incorporation of Contact Conditions

In this study, MG-PCG methods are used in the solution of the linear system in Algorithm 6.1. However, since constraint imposition and updating are done only for the fine mesh where a solution is achieved, the perturbation, $\mathbf{K}'$, must be restricted from the fine mesh to all coarser meshes. This poses a difficulty: since the constrained fine mesh nodes do not necessarily coincide with the coarse mesh nodes, a fine mesh nodal contribution to the perturbation cannot be merely added to the coarse mesh stiffness. This may be overcome by forming the coarse mesh perturbations as:

$$\mathbf{K}^{tc} = \mathbf{K}^c + \mathbf{R}\mathbf{K}'\mathbf{R}^{\mathbf{T}} \tag{6.26}$$

where $\mathbf{K}^{tc}$ refers to the total coarse mesh stiffness and $\mathbf{K}^c$ is assembled from the coarse mesh stiffness matrices as detailed previously. The term $\mathbf{R}\mathbf{K}'\mathbf{R}^{\mathbf{T}}$ represents the coarse mesh perturbation that is consistent with the virtual work requirement that the fine and coarse mesh strain energies be equal; this term will be denoted as $\mathbf{K}'^c$ for the remainder of this discussion.

To simplify the calculation of $\mathbf{K}'^c$, it is computed separately for each pair of constrained nodes. Refer to Fig. 6.4, which shows the fine mesh nodes i and j as being constrained; the nodal contribution to the fine mesh perturbation is represented by a spring, whose stiffness may be expressed as:

$$\omega(\mathbf{n} \otimes \mathbf{n}) \tag{6.27}$$

Using the searching algorithm detailed previously, the coarse mesh elements within which these fine nodes lie may easily be determined. Defining $\mathbf{N}_i$ and $\mathbf{N}_j$ as the usual arrays of shape functions of the coarse mesh elements that the fine mesh nodes i and j lie in, the coarse mesh perturbation corresponding to the constraint at nodes i and j is easily

determined using the principle of virtual work as:

$$\mathbf{k}'^{c} = \omega \begin{bmatrix} \mathbf{N}_i^{\mathrm{T}} \\ -\mathbf{N}_j^{\mathrm{T}} \end{bmatrix} (\mathbf{n} \otimes \mathbf{n}) \begin{bmatrix} \mathbf{N}_i & -\mathbf{N}_j \end{bmatrix} \tag{6.28}$$

Adding/assembling the contribution of each $\mathbf{k}'^{c}$ gives $\mathbf{K}'^{c}$.

6.5 Performance Studies

In this section, the effectiveness of the proposed MG and MG-PCG solution methods is verified via several model problems that test various aspects of the intergrid transfer of information. V-cycle multigrid is used exclusively with four pre- and post-relaxation pointwise Gauss-Seidel smoothing steps; a single V-cycle is used to precondition the conjugate gradient solver. Again, it is important to note that a symmetric preconditioner is necessary for the conjugate gradient method. Symmetry of the multigrid preconditioner can be accomplished by choosing the post-smoother to be the transpose of the pre-smoother (line 5 of Algorithm 6.3). A Jacobi iteration has this property if used for pre- and post-smoothing; use of forward Gauss-Seidel for pre-smoothing and backward Gauss-Seidel for post-smoothing as done in this study also provides symmetry [77].

The first model problem is designed to provide performance baselines for well-conditioned, linearly elastic problems and allows comparison with efficient sparse matrix factorization techniques and ILU-PCG. Although the model geometry at the finest level is regular, unnested mesh sequences are used at the coarser levels to illustrate the applicability of the proposed definitions of $\mathbf{R}$ and $\mathbf{T}$. Following this, the effect of localized mesh refinement is examined through two different models, one discretized with quadratic hexahedra and the other with quadratic tetrahedra. These models demonstrate the generality of the proposed definitions of $\mathbf{R}$ and $\mathbf{T}$, and show that MG and MG-PCG suffer no performance losses when mesh sequences consisting of a locally refined fine mesh and uniform coarse meshes are employed. The final model problem studies spatially varying material properties and the resulting effects on convergence. It is demonstrated that MG-PCG performs quite well even when sharp material boundaries not captured explicitly by the coarse mesh are present.

6.5.1 Thick Plate Model Problem – Baseline Performance Studies

Prior studies by Parsons and Hall [70, 71] have shown that achieving a linear relationship between the number of floating point operations (N_{ops}) and the number of unknowns (n) requires that: (1) the work required to solve at the coarsest level must be negligible; (2)

β (defined previously) must be less than κ, a parameter denoting a uniform level of mesh refinement (typically $\kappa = 2^3$ in 3D); and (3) the total number of meshes, m, is a function of $\frac{\beta}{\kappa}$. Implicit in item (3) is that the level of discretization of the finest mesh must be chosen so that m is large enough.

The requirements for linearity between N_{ops} and n when using sequences of unstructured, unnested meshes are not so easily defined. For example, when automatic mesh generation is used to generate a sequence of meshes, refinement is often done only in selected regions of the mesh where solution errors are high. Further, complex geometry often requires the use of distorted elements or different element types, making it impossible to define a general level of refinement. For these reasons, the relationship between N_{ops} and n is studied numerically using different discretizations of the domain shown in Fig. 6.7.

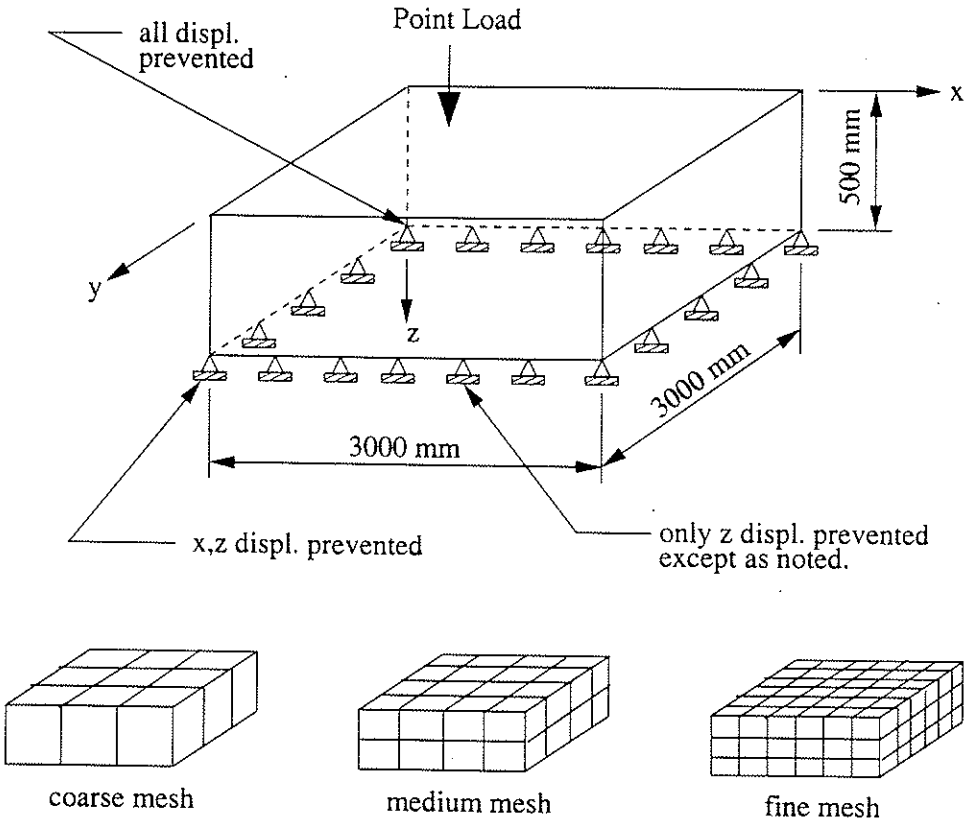


Figure 6.7: Thick Plate Model and Typical Mesh Sequence

Table 6.1 defines the sequences of meshes used to study convergence, along with the number of degrees of freedom, n, for each mesh. Uniform meshes with reasonable element aspect ratios are used to minimize numerical effects on convergence. The modulus of elasticity, E, was fixed at 100, and $\nu = 0.20$ for all cases.

Table 6.1: Sequences of Meshes Used to Study Convergence

Sequence	Mesh	Number of Elements			n
		n_x	n_y	n_z	
Sequence 1	coarse	3	3	1	288
	medium	4	4	2	735
	fine	6	6	3	2037
Sequence 2	coarse	3	3	1	288
	medium	5	5	2	1080
	fine	8	8	3	3429
Sequence 3	coarse	3	3	1	288
	medium	6	6	2	1491
	fine	10	10	3	5181
Sequence 4	coarse	3	3	1	288
	medium	8	8	2	2511
	fine	13	13	3	8484
Sequence 5	coarse	3	3	1	288
	medium	10	10	2	3795
	fine	16	16	3	12597
Sequence 6	coarse	3	3	1	288
	medium	11	11	2	4536
	fine	18	18	3	15789
Sequence 7	coarse	3	3	1	288
	medium	13	13	2	6216
	fine	22	22	3	23253
Sequence 8	coarse	3	3	1	288
	medium	15	15	2	8160
	fine	25	25	3	29796
Sequence 9	coarse	3	3	1	288
	medium	17	17	2	10368
	fine	30	30	3	42501
Sequence 10	coarse	3	3	1	288
	medium	19	19	2	12840
	fine	35	35	3	57456
Sequence 11	coarse	3	3	1	288
	medium	19	19	2	18576
	fine	42	42	3	82173

The chosen mesh sequences have the following characteristics:

1. The number of meshes, m, is fixed at three for each sequence.
2. The coarsest mesh is identical for each sequence of meshes.
3. The level of discretization of the intermediate mesh is approximately midway between the finest and coarsest meshes.
4. Although the discretizations at the finest mesh level are regular, the mesh sequences are unnested, and general definitions of the interpolation and restriction operators are required.
5. No attempt was made to optimize the mesh sequences.

Fig. 6.8 shows the real time required for each aspect of the MG and MG-PCG solutions. All runs were done on a Sun UltraSparc 1/200 with 256 MB of RAM, using only core memory. The number of MG or MG-PCG iterations required to achieve convergence is taken as the first iteration where

$$\epsilon = \frac{\| \mathbf{r} \|_2}{\| \mathbf{P} \|_2} \leq 10^{-05} \tag{6.29}$$

$\mathbf{P}$ is defined as the vector of applied nodal forces and $\mathbf{r}$ is the vector of fine mesh residual forces. The following observations can be made from Fig. 6.8:

- MG-PCG clearly outperforms MG for all levels of discretization, and exhibits a nearly linear increase in solution time as the mesh is refined.
- MG suffers fairly severe performance losses as the mesh is refined, with a superlinear relationship between solution time and n.
- The amount of computational effort required to build the auxiliary data structures, which include $\mathbf{R}$, $\mathbf{T}$, and the coarse mesh stiffness matrices, is about half that required to assemble $\mathbf{K}$.

Fig. 6.9 compares the total solution times for MG-PCG, ILU-PCG, and sparse direct factorization. For the direct sparse solution [17], minimum-degree reordering was employed, but no pivoting was used as it is not required for well-conditioned symmetric positive definite systems. The largest problem which could be factored in-core corresponded to a mesh with $15 \times 15 \times 3$ elements and 11136 nodal DOF.

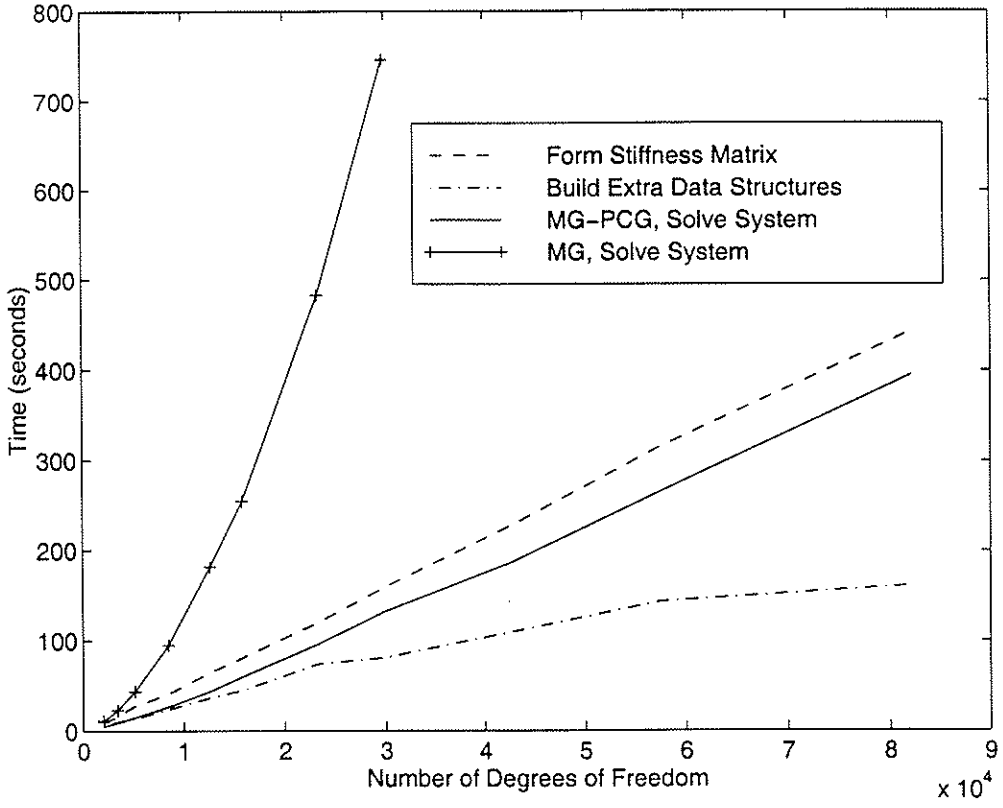


Figure 6.8: Breakdown of MG and MG-PCG Solution Components

The comparison of total solution times shown in Fig. 6.9 indicates that for $n >\approx 3000$, MG-PCG outperforms direct factorization. Note that the performance of the direct solver is affected by node ordering. For example, the worst-case scenario for a direct solver would be a cube meshed with equal numbers of elements in all directions, while the meshes used here have significantly fewer elements through the thickness of the plate and optimally numbered nodes. The performance of ILU-PCG is also shown for comparison purposes; although symmetry was not accounted for in storing the partial factors $\mathbf{L}'$ and $\mathbf{U}'$, it has been considered in determining computational time. For $n >\approx 40000$, MG-PCG outperforms ILU-PCG, and there is a superlinear increase in solution time for ILU-PCG. For these reasons, MG-PCG will be used used exclusively for the remainder of this study; for smaller problems, however, ILU-PCG may well be the most efficient solution method.

6.5.2 Performance on Locally Refined Unstructured Meshes

In order to verify that the proposed intergrid transfer strategies are effective on general unstructured meshes, two model problems will be examined. The first is the previously

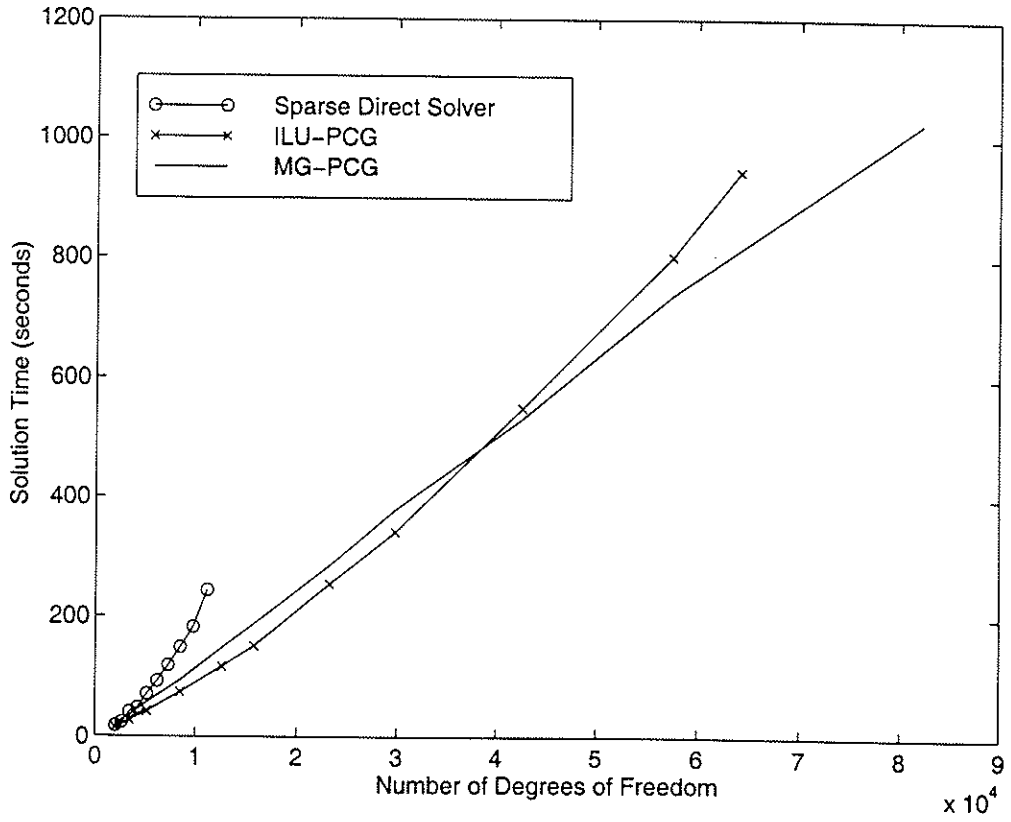


Figure 6.9: Relative performance of MG, MG-PCG, ILU-PCG, and Sparse Direct Solver

analyzed thick plate, but the fine mesh is refined over the center 750 mm by 750 mm region as shown in Fig. 6.10. The total number of elements is 960, and $n = 14949$, which nearly equal the values for the fine mesh in sequence (6). The coarse and medium meshes used in the refined mesh solution are the same as those in sequence (6), and are shown in Fig. 6.10.

The amount of time required to achieve a solution using MG-PCG was 185 seconds. Comparing this value with the total solution times in Fig. 6.9 indicates that there is effectively no loss in efficiency for MG-PCG when compared to regular mesh solutions with the same number of unknowns. We note here that regular coarse meshes were used to achieve a solution on an irregularly refined mesh with no difficulty.

The second model problem is an equilateral triangular plate discretized as shown in Fig. 6.11. All the meshes in the multigrid hierarchy are irregular (generated using the automatic mesh generator QMG [66]). The boundary conditions consist of vertical supports on one edge and at the opposite vertex, and horizontal supports sufficient to prevent rigid body motion. A point load is applied near the center of the plate where the mesh is refined. The total time required to solve this problem using the MG-PCG method was approximately

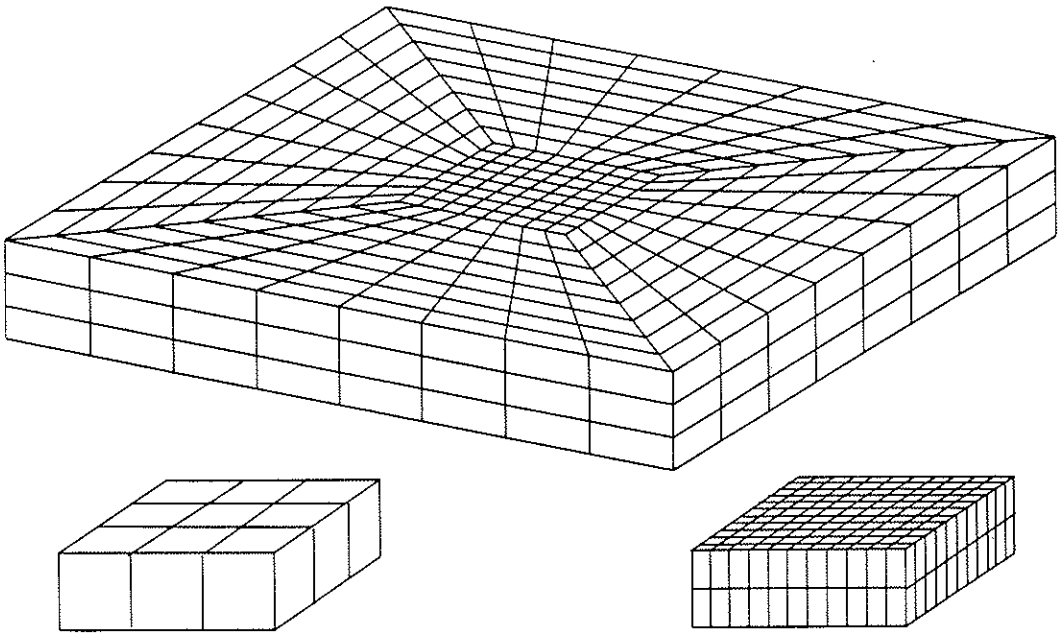
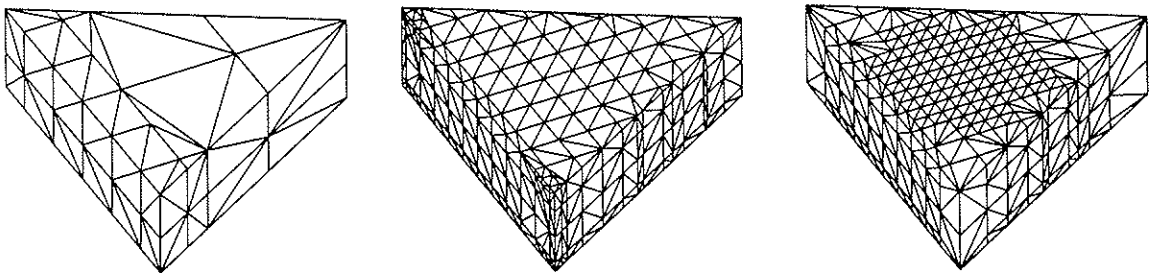


Figure 6.10: Refined Mesh Sequence



144 elements, 945 DOF 1760 elements, 9057 DOF 5518 elements, 24663 DOF

Figure 6.11: Mesh Sequence – Triangular Plate

202 seconds; a comparison with Fig. 6.9 indicates that this is approximately three-quarters the time required for a thick rectangular plate model with the same number of degrees of freedom. No generalizations can be made, however, due to the differences in element type, problem, and mesh sequence between the rectangular and triangular plate models.

6.5.3 Performance on Meshes with Spatially Varying Material Properties

The proposed method of dealing with spatially varying material properties will be illustrated with a model problem. Recall that material properties are defined element-by-element on the fine mesh, and this information is transferred to coarser meshes only at the the coarse mesh element integration points. This presents a possible source of error when evaluating

the coarse mesh element stiffness matrices, $\mathbf{k}$, by the usual Gaussian integration of:

$$\mathbf{k} = \int_V \mathbf{B}^{\mathrm{T}}\mathbf{D}\mathbf{B}dV \tag{6.30}$$

where $\mathbf{B}$ is a matrix of linear operators and $\mathbf{D}$ is the constitutive matrix. Gaussian integration is exact for polynomials, the order of which depends on the number of integration points used. In this study, $3 \times 3 \times 3$ Gaussian integration is used to evaluate Eq. 6.30, which is exact for a quadratic, hexahedral element with uniform, linearly elastic material properties. If a coarse mesh element covers a region with more than one applicable constitutive law, a sharp "jump" in $\mathbf{D}$ may occur, possibly introducing significant error in the evaluation of $\mathbf{k}$. This is studied with the following model problem, which has been purposely constructed to introduce significant errors due to sharp material boundaries.

Consider the mesh shown in Fig. 6.12, which follows the same pattern as the refined mesh presented in the previous section, but with fewer elements. The center $4 \times 4 \times 2$ block of rectangular elements is assumed to have a modulus of elasticity, E_c, differing from the remainder of the domain, where E = 100 N/mm^2. To solve the problem using MG or MG-PCG, a single coarse mesh consisting of 9 elements is used. Fig. 6.12 shows that if the regular, 9 element coarse mesh presented previously (denoted by $\mathcal{M}_c$) is used, only three of the 27 center element integration points lie within the center 750 mm$\times$ 750 mm refined region of the fine mesh, which can be expected to introduce significant error in the evaluation of $\mathbf{k}$. However, if the coarse mesh element boundaries coincide with the material property boundary, the evaluation of $\mathbf{k}$ is exact (see Fig. 6.12); this mesh is denoted by $\mathcal{M}_e$.

Fig. 6.13 shows the effect of using $\mathcal{M}_e$ as opposed to $\mathcal{M}_c$. Note that no results are presented for MG with $\mathcal{M}_c$ and $E_c/E > 20$, as it diverges (convergence can be achieved for $E_c/E > 20$, but only by increasing the number of Gauss-Seidel smoothing operations within each MG cycle). The following observations are based on the results presented in Fig. 6.13:

- When using $\mathcal{M}_e$, the solution times for both MG and MG-PCG are essentially constant, verifying that the only source of error is the variation in the coarse mesh element boundaries.

- MG-PCG appears to be stable even if significant errors are introduced by the use of $\mathcal{M}_c$, and the increase in solution time is only doubled for $E_c/E = 100$.

- The performance of both MG and MG-PCG is improved by the use of $\mathcal{M}_e$ as opposed to $\mathcal{M}_c$, indicating that errors in integrating the coarse mesh stiffness can have

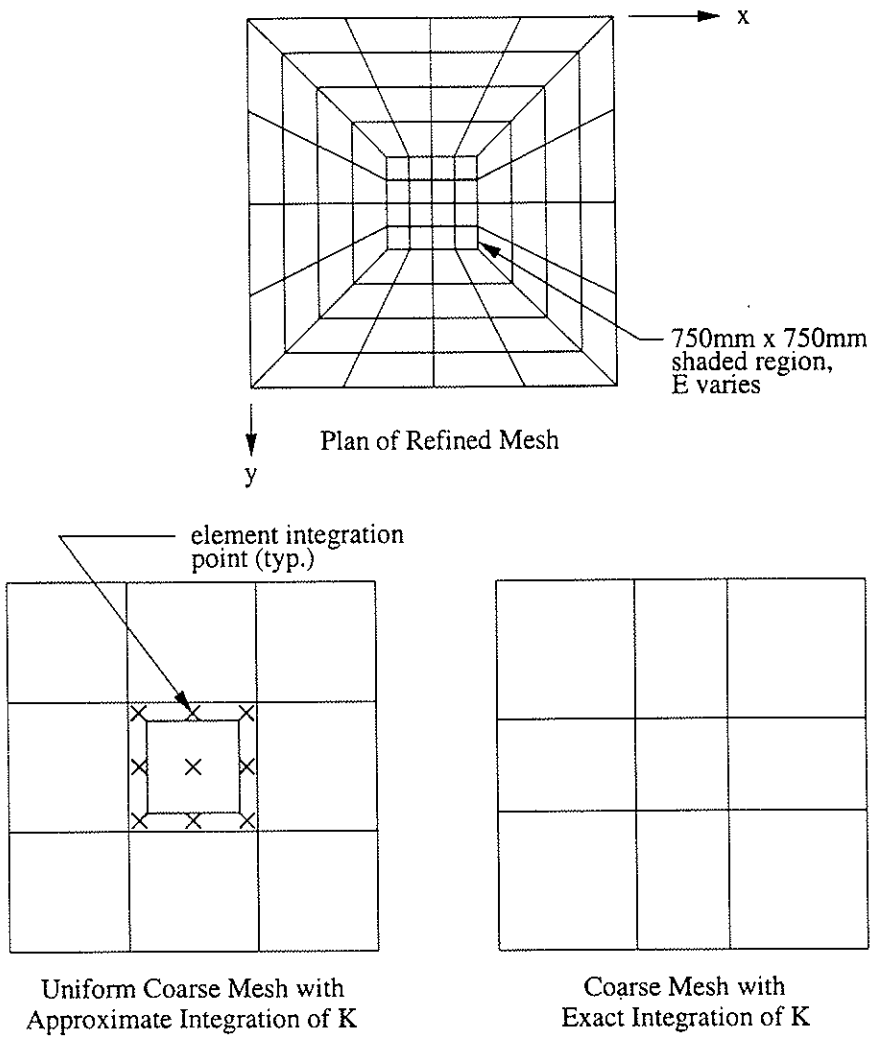


Figure 6.12: Meshes Used to Study Spatially Varying Material Properties

significant detrimental effects.

The approach for handling spatially varying materials presented here appears to be viable when MG-PCG solution methods are used and MG, even when non-convergent itself, is still a very effective preconditioner. A similar conclusion was drawn in [4] for finite difference simulations of groundwater flow, where increasing degrees of subsurface heterogeneity resulted in non-convergence of MG, but MG-PCG remained stable and efficient. However, care should be taken when developing coarse meshes where sharp discontinuities in material properties exist. Alternatively, more accurate integration techniques could be employed in the evaluation of Eq. 6.30 in these regions.

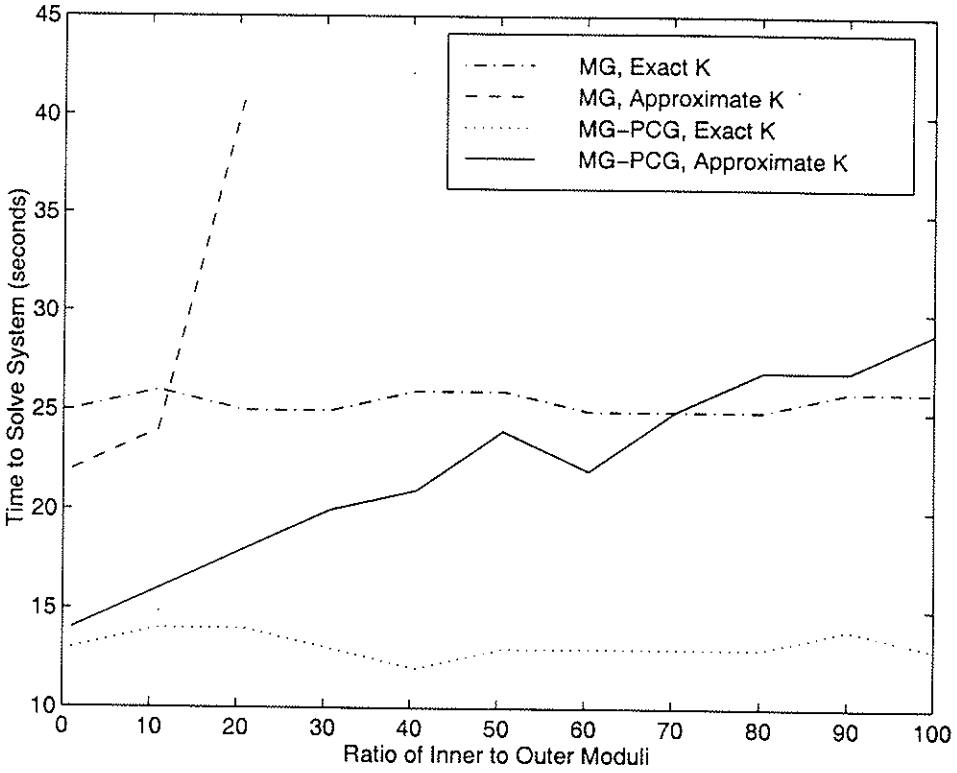


Figure 6.13: Spatially Varying Material Properties and Element Integration Error

6.6 Rigid Pavement Model Problem

In this Section, a typical model of a rigid pavement system will be solved to illustrate the applicability of the global nonlinear solution strategy to problems involving multiple element types and nodal contact.

6.6.1 Model Description

The system to be modeled is typical, consisting of two doweled, 254 mm thick rigid pavement slabs resting on two 300 mm thick base layers above the natural subgrade – similar to the system illustrated in Fig. 6.1. The slabs are 3350 mm long by 3660 mm wide and are separated by a transverse construction joint skewed at an angle of 10°. Three 485 mm long, 32 mm diameter dowels spaced at 300 mm are located in each wheelpath at the mid-thickness of the slabs. The slabs and base layers are meshed with 20-noded quadratic brick elements. The dowels are modeled using the embedded dowel element developed in Chapter 4. The natural soil below the solid elements is treated as a dense liquid foundation, and meshed with 8-noded quadratic interface elements that share nodes with the elements

of the lower base layer. The boundary conditions are the minimum necessary to prevent rigid body motion. Surface plots of the meshes used in the multigrid solution are shown in Fig 6.14; note the dowel elements shown as thick lines at the skewed joint. The number of solid elements in the coarse, medium, and fine models are 36, 216, and 1152 respectively; the corresponding numbers of degrees of freedom are 1314, 4929, and 21285. The slab and

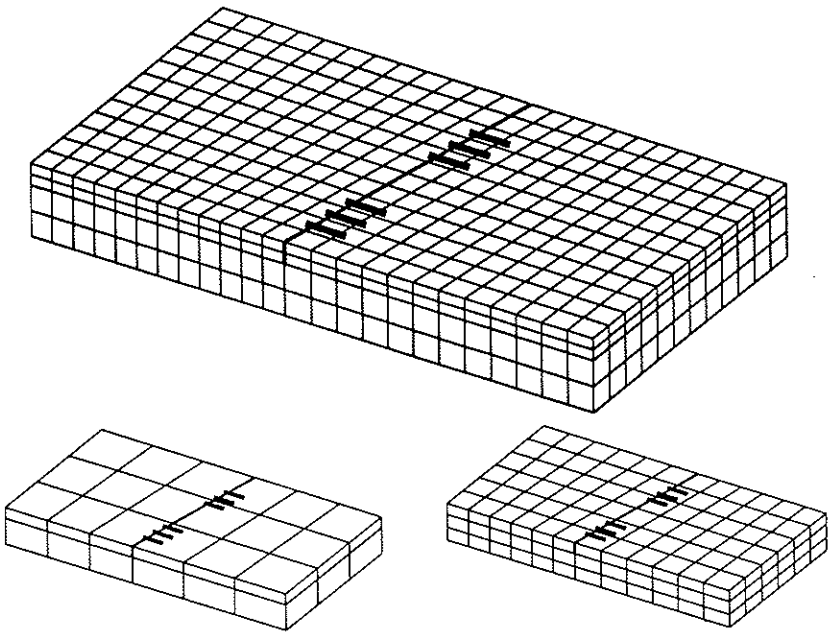


Figure 6.14: Mesh Sequence Used for Rigid Pavement Model Problem

base layers were modeled as linearly elastic with material properties as given in Table 6.2, where α is the coefficient of thermal expansion and ρ is the mass density. The values of E for the upper and lower base layers represent typical values for a cement treated soil and a strong gravel, respectively [64]. The dense liquid was assumed to have a modulus of 0.06MPa/mm. Note the extreme material boundary between the two base layers which is not captured by the coarsest mesh.

Table 6.2: Material Properties of Slab and Subgrade

Portion of System	E MPa	ν –	α $^{\circ}C^{-1}$	ρ kg/m^3
Slabs	28000	0.25	1.1×10^{-05}	2400
Upper Base Layer	3500	0.20	–	–
Lower Base Layer	150	0.20	–	–

The left-most slab was subjected to a single axle load of 80 kN near the joint. The axle was centered transversely on the slab with a wheel spacing of 1830 mm, and each wheel load was distributed over a 500mm×250mm area. In addition, the slabs were subjected to a linear temperature gradient through their thickness corresponding to a -5°C temperature change at the top of the slab and a +5°C temperature change at the bottom. This temperature differential required that the loss of contact between the unbonded upper base layer and the slab be modeled using the nodal contact approach presented in the previous Section. This dictated that the slab and upper base layers be meshed independently; details near the transverse joint are depicted in Fig. 6.3. The value of the parameter ω required for the Uzawa iteration was computed at run-time as the average value for all constrained nodes of $\mathbf{K}_{ii}$ corresponding to the constraint direction, $\mathbf{n}$. To illustrate the analysis results, a 3D perspective of the displaced shape is shown in Fig. 6.15 (displacements are scaled by factor of 500); note the slab lifting off the base layer due to the temperature gradient.

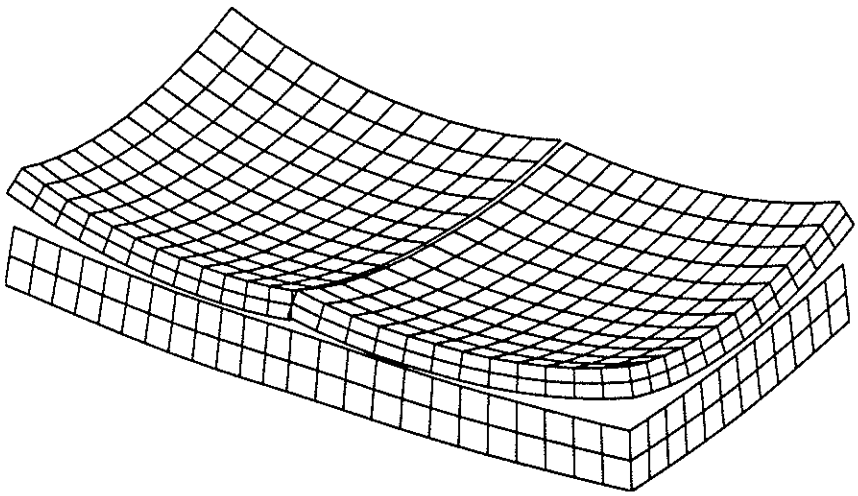


Figure 6.15: Displaced Shape of Rigid Pavement System

6.6.2 Performance of Solver

The iterative nonlinear solution strategy presented in Section 2 was used to solve the model problem. Both the global convergence tolerance, ϵ, and the constraint tolerance, ϕ were fixed at 10^{-05}. The solution required 12 global iterations and 2731 seconds of CPU time to solve on a Sun UltraSparc 1/200 workstation, and was achieved using only core memory. The following table gives a breakdown of the solution components. Note that the total number of Uzawa iterations required to achieve convergence for the constrained system

Table 6.3: Breakdown of Rigid Pavement Model Solution Results

Newton Iteration	Total Uzawa Iterations	Total MG-PCG Iterations	Number of Constr. Updated	Residual
1	4	12	264	8.22×10^{-03}
2	3	30	155	7.32×10^{-03}
3	3	30	70	6.71×10^{-03}
4	3	29	89	6.50×10^{-03}
5	3	32	61	5.13×10^{-03}
6	3	30	40	4.44×10^{-03}
7	3	27	29	3.15×10^{-03}
8	3	29	23	2.41×10^{-03}
9	2	22	19	1.58×10^{-03}
10	2	22	15	7.80×10^{-04}
11	2	22	3	2.35×10^{-04}
12	2	14	0	1.11×10^{-07}

never exceeds four, and averages about three. While at each Uzawa iteration the MG-PCG method is used to solve the resulting linear system, the number of MG-PCG iterations per linear solve varies significantly.

To put these results in perspective, comparisons must be made with the solution when a sparse direct solver is used in lieu of MG-PCG. This was only done for the medium mesh (4923 DOF), as a fine mesh solution would require too much memory with a direct solver. Note that when using a direct solver, $\mathbf{K}$ may be factored outside the Uzawa iteration, and each linear solve then becomes a back substitution. This advantage for the direct solver is offset by the fact that the auxiliary data structures required for the MG-PCG method need only be constructed once. In fact, even on the medium mesh using a two mesh sequence, the total solution time for MG-PCG is 407 seconds, less than the 503 seconds required when a sparse solver is used. Further, the MG-PCG solver required 21 MB of RAM, while the sparse direct solver required about 85 MB (with a minimum degree reordering).

6.7 Summary

In this Chapter, a nonlinear solution strategy was developed that is capable of solving materially nonlinear problems with inequality constraints arising from nodal contact. It employs three levels of iteration, where the outer level is analogous to a Newton iteration, the second

level is an iterative solution of the constrained system for increments of displacement and Lagrange multipliers, and the the inner kernel requires the iterative solution of a linearized, symmetric positive definite system.

To minimize the computational effort required in the inner kernel of the global solution algorithm, appropriate multigrid (MG) and multigrid-preconditioned conjugate gradient (MG-PCG) methods have been developed. These methods are quite general, relying on definitions for the interpolation and restriction operators that allow multiple element types and the meshing of bending members with solid elements within a single model. Spatially varying material properties are handled in a manner that relies on similar principles used to develop the restriction and interpolation operators. The MG and MG-PCG methods have also been extended to handle difficulties in forming the coarse mesh perturbations to the system stiffness matrices required by the nonlinear solver when modeling problems involving contact nonlinearities.

Baseline performance studies were conducted, providing comparisons between MG and MG-PCG methods vs. a state-of-the-art sparse direct solver and ILU-PCG. The MG-PCG method was shown, as expected, to be significantly more efficient than direct solution methods for large problems and exhibits a nearly linear increase in solution time with the number of unknowns. When compared with ILU-PCG, MG-PCG is more efficient for well-conditioned linear problems with more than approximately 40000 unknowns, and ILU-PCG exhibits a superlinear increase in solution time with the number of unknowns. Memory requirements beyond the storage of the stiffness matrix (or the element stiffness matrices) are small. The effectiveness of the proposed intergrid transfer techniques was demonstrated in the presence of mesh irregularities, local refinement, spatially varying material properties, and sharp material boundaries that may not be captured by all grids in the multigrid hierarchy. Further, the ability to precondition problems involving inequality constraints via restriction of the required fine mesh perturbation to unnested coarse meshes was shown to be effective in the general nonlinear solution strategy.

Lastly, a model problem of a rigid pavement system was solved that incorporated multiple element types, spatially varying materials, and nodal contact conditions to illustrate the application of the nonlinear solution strategy employing MG-PCG.

Chapter 7

Model Verification

This Chapter details the verification the finite element models developed in this study by comparison with laboratory data from scale-model tests of doweled rigid pavement systems. In particular, finite element models of two systems tested by Hammons [33] are solved, and displacement profiles and selected strains in the vicinity of the load are compared with measured values.

7.1 Laboratory and Analytical Study by Hammons

The basic system tested by Hammons is illustrated in Figs. 7.1 and 7.2 [33]. It consisted

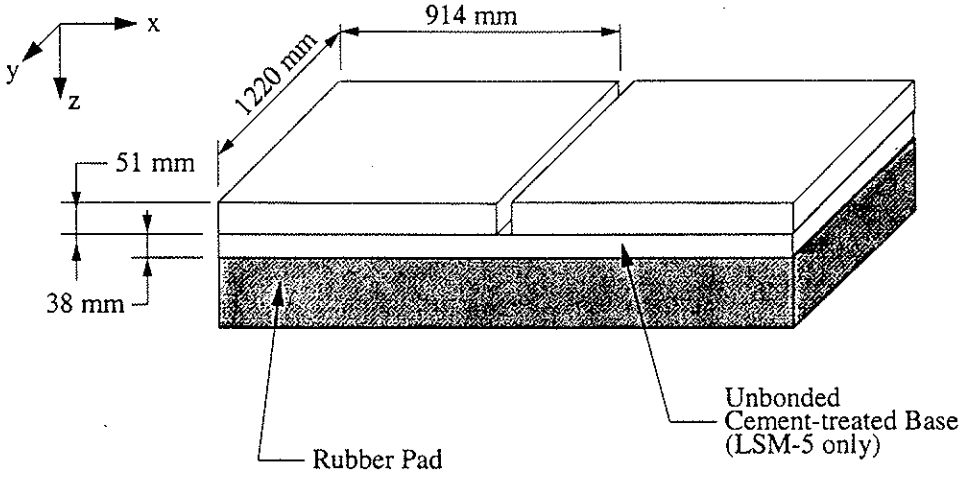


Figure 7.1: Experimental Model [33]

of two concrete pavement slabs separated by a 1.6 mm smooth joint constructed to prevent aggregate interlock. The 12 dowels were purposely debonded in one slab to simulate typical installed conditions. The model was cast in a steel reaction box, and the ends of the slabs were clamped between two angles to prevent displacement and increase their effective

length. Of the six tests completed, the first two (LSM-1A and LSM-1B) had an undoweled joint and were essentially tests of a single slab. Test LSM-2 consisted of two doweled slabs resting directly on the rubber pad; tests LSM-3 through LSM-5 employed a cement-treated base. For further details on all tests, see the report by Hammons [33].

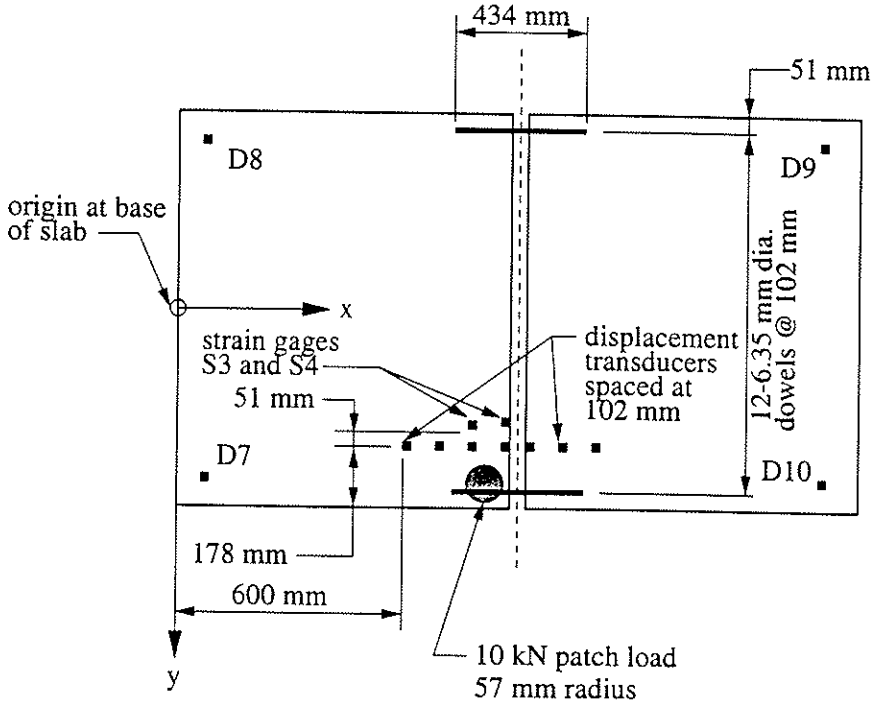


Figure 7.2: Dowel Locations, Loading, and Instrumentation [33]

Hammons also conducted finite element analyses for comparison with his experimental data using the commercially available program ABAQUS. Material properties of the slab and treated base were determined by testing specimens taken when the models were constructed in accordance with ASTM specifications. For analytical purposes, the rubber pad was idealized as a dense liquid. Its modulus, k, was based on the value back-calculated by Hammons using the laboratory data from the first two tests, LSM-1A and LSM-1B. Test LSM-1A consisted of a single slab resting directly on the rubber pad and subjected to an edge load; back-calculation using a finite element model yielded a value of $k = 0.070$ MPa/mm for best-fit displacements. It must be noted that the value of k is a secant stiffness that is dependent upon the configuration and loading of the system. This was experimentally observed by back calculating $k = 0.09$ MPa/mm for the same single slab resting on the rubber block when subjected to a corner load (test LSM-1B). The higher value of k for LSM-1B may be attributed to the higher and more localized stresses applied to the rubber

pad under corner loading, as it is a strain-hardening material.

The finite element models of Hammons employed 27-noded hexahedral elements to discretize the slab and base layer; separation of the slab and base was modeled assuming frictional contact conditions. The dowels were not explicitly discretized, but replaced by linear displacement springs spanning and evenly distributed across the joint. The stiffness of these springs was determined using the methods outlined in [45], which rest on the conventional assumption that the embedded portion of the dowel may be considered as a beam on elastic foundation with a specified modulus of reaction. Further details of these finite element studies may be found in [33]; pertinent differences between the finite element models developed in this study and the analyses by Hammons will be discussed later in this Chapter.

7.2 Finite Element Model Development

7.2.1 Discretization of the System

Tests LSM-2 and LSM-5 have been selected for comparison with the finite element models developed in this study. These tests were selected as bounding realistic conditions: LSM-2 had no treated base, with the slabs founded directly on the rubber pad; LSM-5 had a debonded, monolithic cement-treated base, requiring that separation of the slabs and base be modeled as detailed in Chapter 6. The finite element model developed for comparison with the test data of Hammons matches the experimental conditions as closely as possible. The rubber reaction block is treated as a dense liquid, while the slab and treated base are modeled as elastic continua. Figure 7.2.1 shows the mesh sequence for the finite element model of test LSM-5; an identical model without the base layer was used for LSM-2.

The material properties used in the finite element model are given in Table 7.1. The

Table 7.1: Model Material Properties

Portion of System	E MPa	ν –
Slabs	27600	0.18
Treated Base	1410	0.20
Dowels	200000	0.30

rubber pad has a modulus of 0.090 MPa/mm. As noted earlier, this value was back-

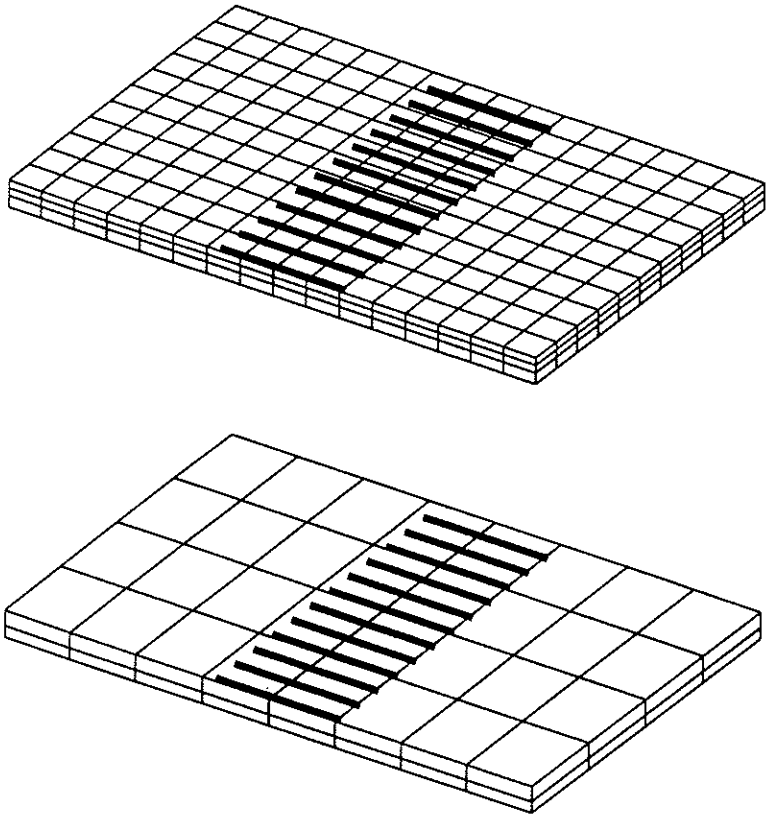


Figure 7.3: Mesh Sequence for Test LSM-5

calculated by Hammons using a 3D finite element model developed using ABAQUS to give good deflection profile comparisons with tests of a single undoweled slab founded directly on the rubber pad and subjected to corner loading. The moduli of the slab and subgrade are representative of the mean values determined from cylinder tests of specimens taken when the slabs and base layer were cast. All material properties are identical to those used by Hammons in his finite element verification studies [33].

7.2.2 Model Boundary Conditions

The model boundary conditions were chosen to reflect those of the laboratory tests. The reaction box in which the slabs were cast and tested was designed to prevent displacement of the ends of the slabs, increasing their effective length. On the sides of the slabs perpendicular to the joint, the reaction box was coated with a form release agent, and the slabs were poured against the box. Displacement transducers D7 through D10 near the slab corners (see Fig. 7.2) were monitored throughout the tests to check displacements near the slab ends.

Hammons reported that for test LSM-2, the displacements were indeed negligibly small, and the appropriate boundary conditions in the finite element model are fixed zero displacements at the slab ends. However, for test LSM-5, the ends of the slab underwent significant vertical displacements [33]. Therefore, for the finite element model of test LSM-5, the boundaries of the tops of the ends of the slabs were subjected to fixed vertical displacements that varied linearly across their width in accordance with measured displacements at transducers D7 through D10. The two horizontal (x and y direction) displacements were not fixed on the assumption that if vertical displacements were not adequately restrained by the clamping angles, neither were the horizontal displacements.

The sides of the slabs and base layers ($+/-y$ faces) were unrestrained even though the slabs were cast against the reaction box walls. The reasons for this were threefold: (1) the slabs were only restrained when displacing in the $+y$ direction on the $+y$ face and in the $-y$ direction on the $-y$ face; (2) even when this contact between the slabs and box walls occured, the restraint provided by the box walls was not that of a rigid support; and (3) shrinkage of the slab may be expected to have resulted in a gap between the slabs and box walls prior to loading which had to be overcome before any contact could occur.

It should be noted that these boundary conditions are significantly different than those assumed by Hammons in his ABAQUS models, who fixed all displacements at the slab ends for each model regardless of the measured displacements at transducers D7-D10. Further, Hammons fixed the y-direction (lateral) displacements on the sides of the slabs (and base layer for LSM-5) on the premise that the reaction box restrained lateral movement of the slabs.

7.2.3 Dowel Modeling

Modeling of the dowels is an issue that requires special treatment. As will be seen, the tests showed significant variations in vertical displacement across the joint near the load. If the dowels are assumed to have no looseness, the finite element model will predict nearly perfect joint efficiency, i.e. very little relative vertical displacement between the two slabs at the joint. However, it is probable that there is some looseness between the dowels and the slabs due to construction techniques. The dowels were held in place with a wooden template while the first slab was poured; prior to placing the second slab, the dowels were greased and plastic drinking straws slipped over them to prevent bonding. In the present finite element models, a constant gap, γ, was assumed between the dowels and the unloaded slab along their embedded length, while perfect contact was maintained between the dowels

and the loaded slab as shown in Fig. 7.4. Note that four dowel elements were located near the joint to better capture potential contact, with two more dowels used to discretize the remaining embedded portions of the dowel. This gives a total of 13 potential contact points for each dowel. The odd dimension of 114.25 mm is necessary since each dowel element must be completely embedded within a single solid element, and the hexahedra used to discretize the slabs have a dimension of 114.25 mm in the x-direction.

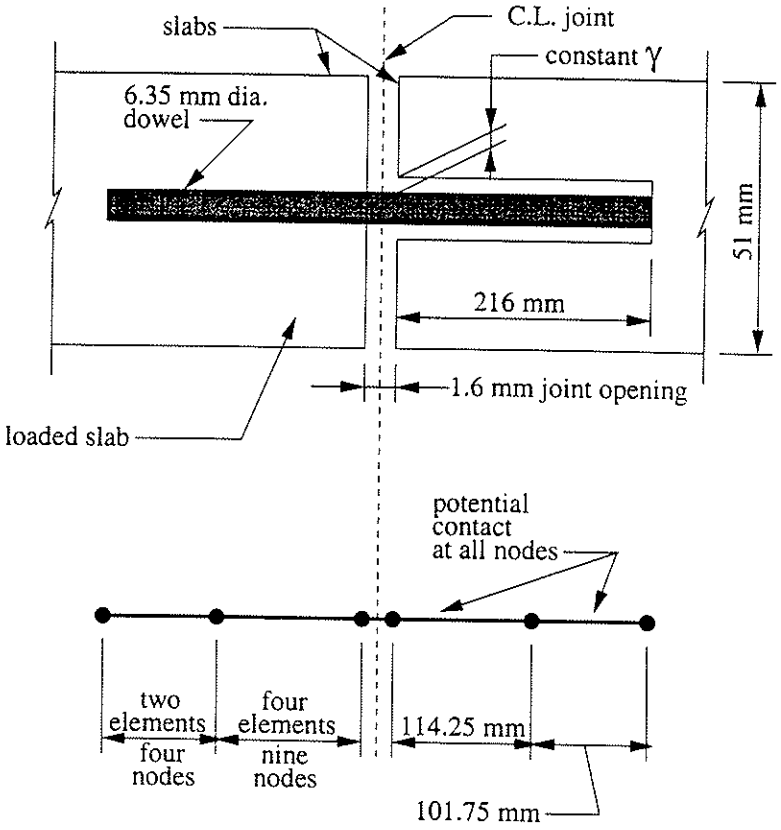


Figure 7.4: Assumed Profile of Gap around Dowels

7.3 Model Results

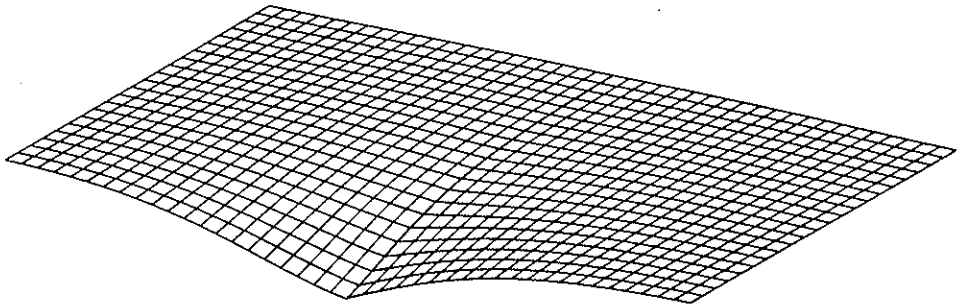
7.3.1 Results for LSM-2

Two different finite element analyses were run for LSM-2: the first assumed no dowel looseness; the second run assumed that a uniform gap, γ, of 0.08 mm existed between the dowel and the unloaded slab. This value of γ was chosen to give a relative vertical displacement between the two loaded and unloaded slabs that closely matches the experimentally measured values. Note that the measured wall thickness of a typical drinking straw is about

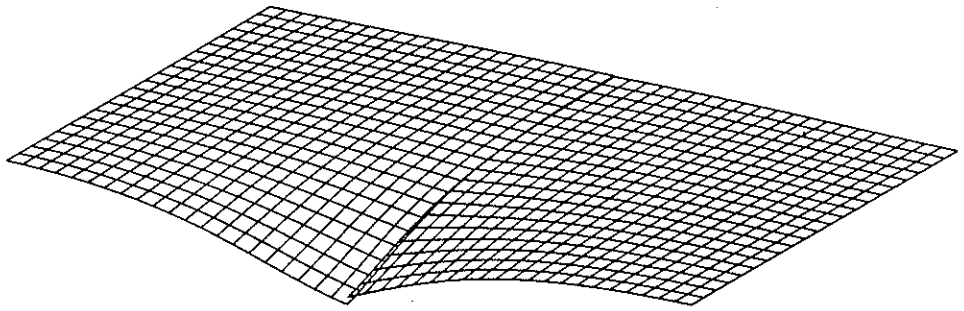
0.13 mm – 59% greater than the assumed gap. Figure 7.5 shows the displaced shape of the top surface of the slabs predicted by both finite element models.

The surfaces defined by these plots were interpolated along the line $y = 432$ mm to give the analytical deflection profile corresponding to that measured experimentally as shown in Fig. 7.6. Note that if $\gamma = 0$, there is nearly perfect load transfer efficiency at the joint; this result is consistent with the convergence study results in Chapter 4. Assuming that $\gamma = 0.08$ mm gives an accurate relative vertical displacement at the joint as well as a better overall prediction of the displacement basin. In general, the agreement between the finite element model and the experimental data is good, with a maximum discrepancy of about 13% observed at $x = 600$ mm.

Figure 7.7 compares the strain (ϵ_{yy}) measured at gages S3 and S4 (see Fig. 7.2) with the values computed by the finite element model assuming $\gamma = 0.08$ mm. While the agreement is reasonably good, the discrepancies between measured and computed strains are significantly larger than those between measured and computed displacements, with relative differences at gage S3 varying from 16% to 24% and from 19% to 35% at gage S4. Note that the maximum relative discrepancies occur at small loads, with the absolute difference between the measured and computed strains remaining relatively constant after the initial divergence. The relatively large error in strains may be attributed to several factors, including: (1) the relatively low level of mesh refinement used in the models; (2) the fact that computed strains are not as accurate as computed displacements when using displacement-based finite element models; (3) differences in actual material properties and those determined by testing; (4) possible inaccuracies in the strain measurements.



Gap = 0.00 mm



Gap = 0.08 mm

Figure 7.5: Displaced Shape of Top of Slab for LSM-2

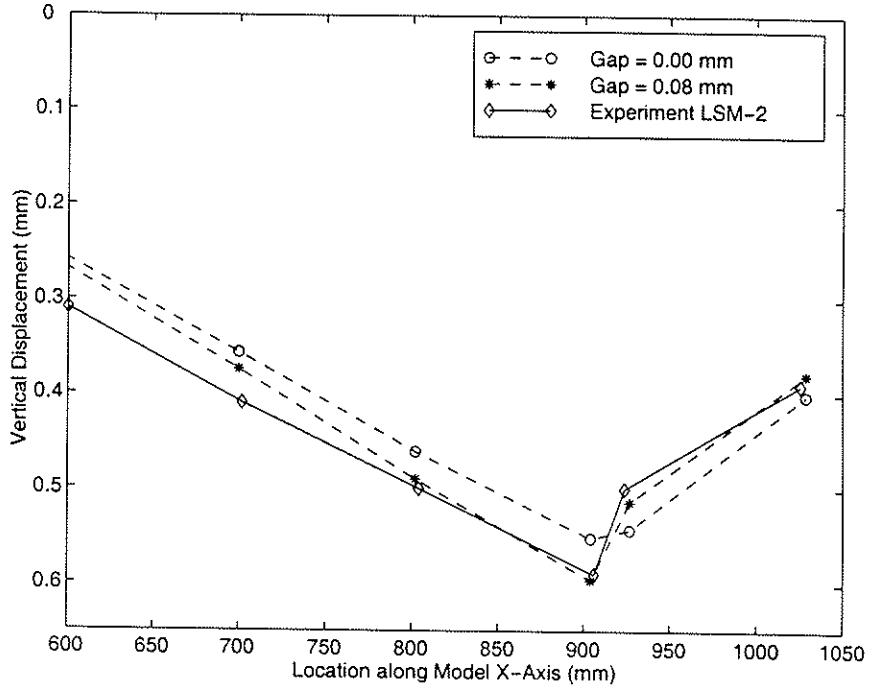


Figure 7.6: Deflection Basin Comparison for LSM-2 at $y = 432$ mm

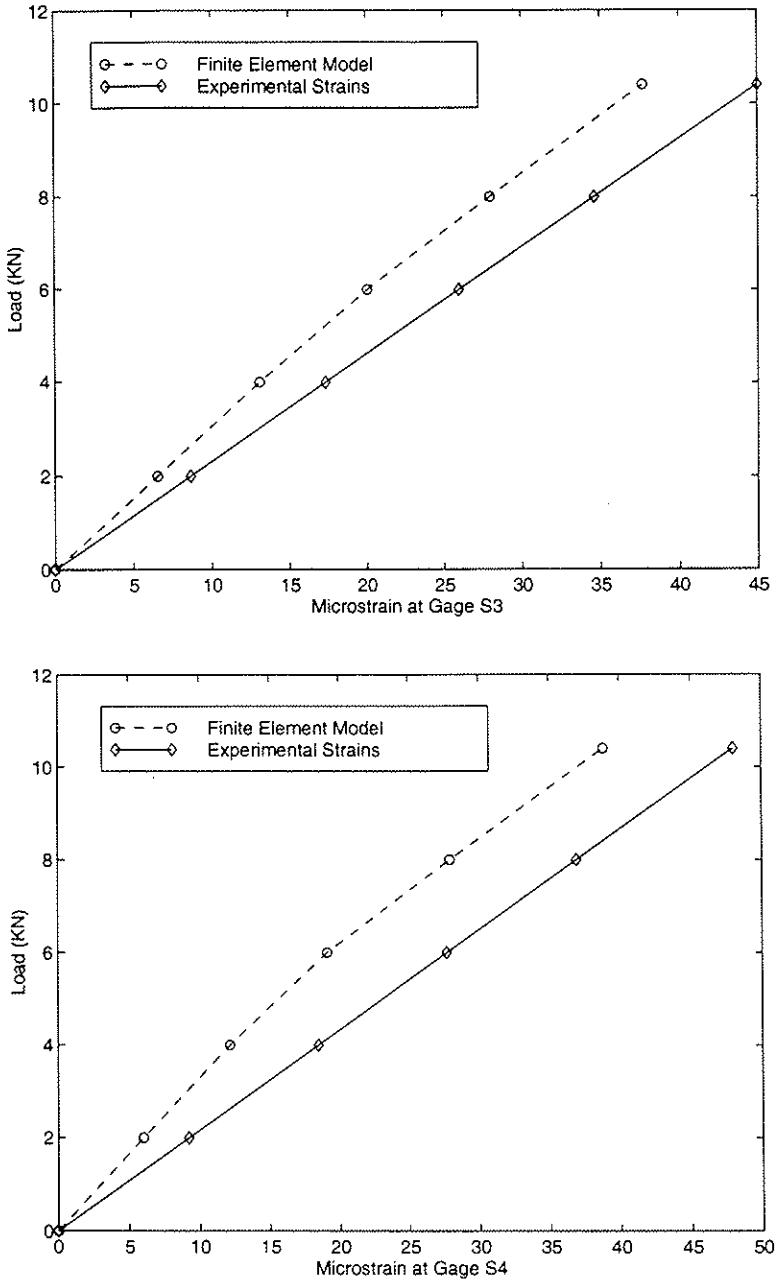


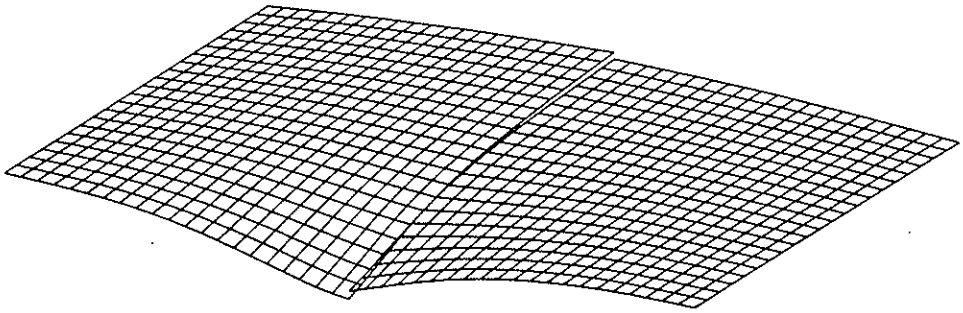
Figure 7.7: Strain Comparison for LSM-2

7.3.2 Results for LSM-5

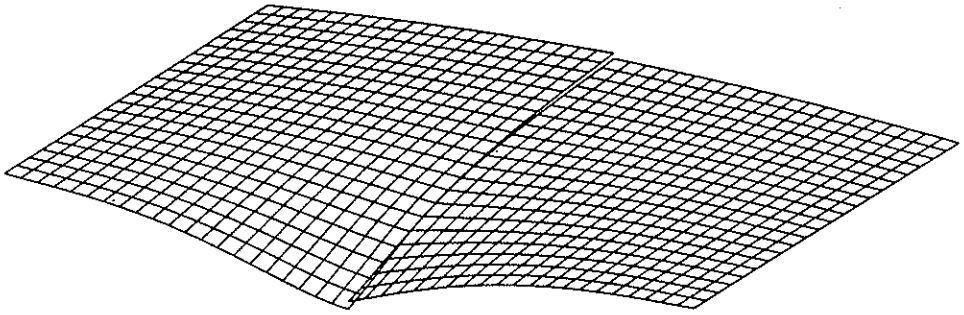
As for LSM-2, two different finite element analyses were run for LSM-5. However, in both analyses, the value of γ was fixed at 0.08 mm based on the results of LSM-2. The first analysis was conducted with an assumed modulus of subgrade reaction for the rubber pad, k, of 0.09 MPa/mm. The second analysis assumed $k = 0.070$ MPa/mm. This reduction is plausible: the value $k = 0.09$ MPa was determined from comparison with a test of a single corner loaded slab resting directly on a dense liquid. Lack of a treated base layer will increase the stresses applied to the rubber pad; since it is a strain-hardening material, the stiffness back-calculated from this test will be too high for case LSM-5, where the stresses are smaller and more uniformly distributed to the rubber pad by the cement-treated base. Figure 7.8 shows the displaced shape of the top surface of the slabs predicted by both finite element models.

Figure 7.9 compares the predicted and experimental deflection basins. Note that for both finite element models, the predicted relative displacement at the joint is quite accurate, validating the 0.08 mm gap determined from the analysis of LSM-2. Fig. 7.9 shows significantly better prediction of displacements when $k = 0.070$ MPa/mm is assumed, in particular near the joint region. However, the maximum error observed is about 17%, larger than that for LSM-2. Also note that the predicted slope of the deflection basin is significantly steeper than that observed experimentally. Possible factors for this include inaccurate modeling of boundary conditions as well as nonlinear and creep response of the rubber block not accounted for in the finite element model. In particular, since the rubber is a strain-hardening material, it will be softer further away from the load and stiffer nearer the load, tending to decrease the predicted slope of the deflection basin.

Figure 7.10 compares the strain (ϵ_{yy}) measured at gages S3 and S4 with the values computed by the finite element model. These results assume a k of 0.070 MPa/mm for the rubber pad. The experimental strain traces are plotted as reported by Hammons, and show compressive strain. This is inconsistent with the observed failure mode, however, which was the development of several corner cracks near the gage locations that would clearly result in tensile strain at gages S3 and S4. This is also inconsistent with the results of LSM-2 which showed measured tensile strains at gages S3 and S4. It is not known if the measured compressive values were actually tensile and were incorrectly reported due to experimental error, or if another explanation exists for this anomaly. If the measured strains are actually tensile, the computed strains are reasonably accurate at the maximum load of 10.2 kN,



Gap = 0.08 mm, k = 0.09 MPa/mm



Gap = 0.08 mm, k = 0.07 MPa/mm

Figure 7.8: Displaced Shape of Top of Slab for LSM-5

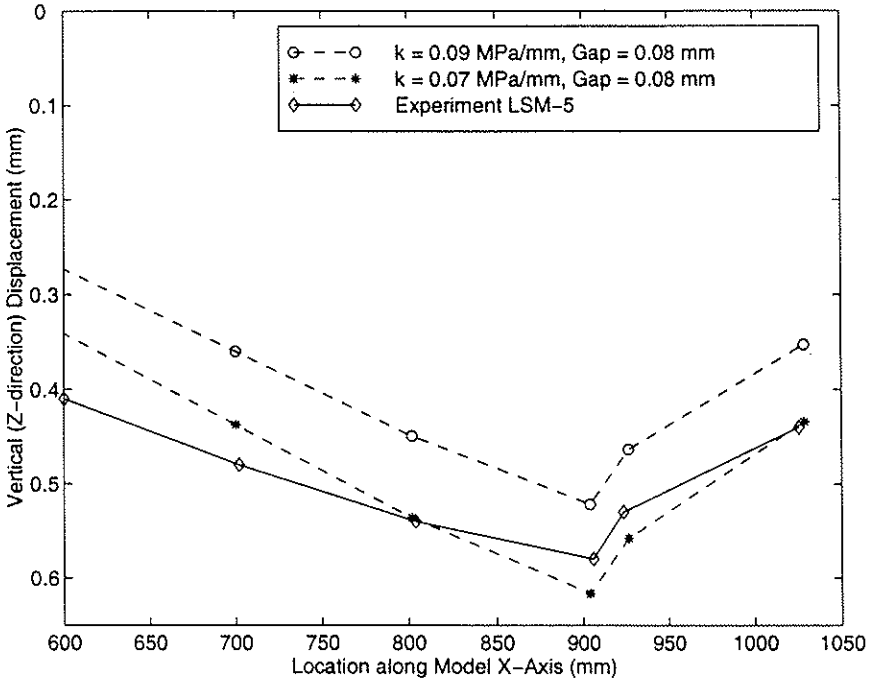


Figure 7.9: Deflection Basin Comparison for LSM-5 at $y = 432$ mm

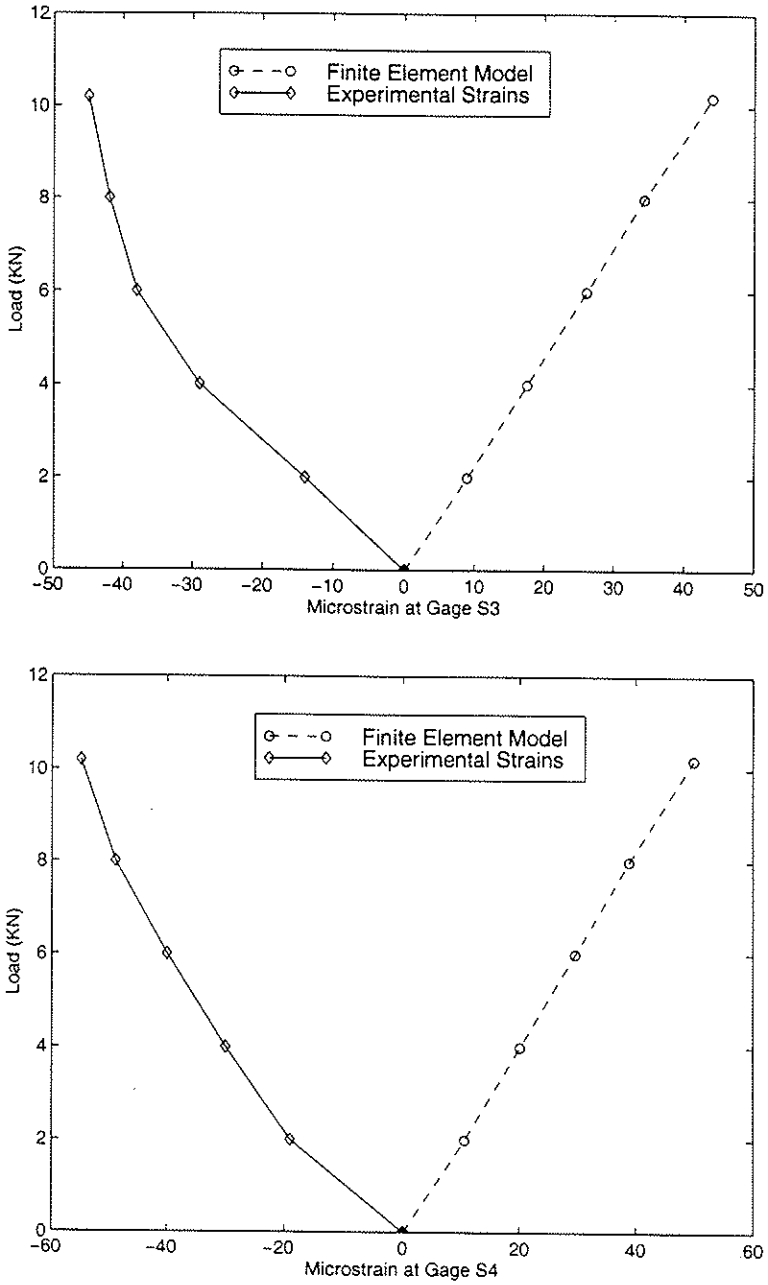


Figure 7.10: Strain Comparison for LSM-5

with an error of 2% at gage S3 and 9% at gage S4. However, the significant nonlinearity in the measured strain response is not captured by the finite element model, leading to large errors at lower loads. In any case, no conclusions can be drawn due to the discrepancy in sign between the measured and computed strains.

7.4 Summary

In this Chapter, the proposed finite element modeling techniques have been verified by comparison with experimental data from scale-model tests of doweled rigid pavement systems [33]. It has been shown that displacement response of typical systems with and without base layers can be predicted with reasonable accuracy. In addition, it was postulated that dowel looseness arising from construction tolerances is a probable explanation for observed variations in relative joint displacements. The dowel modeling techniques developed in Chapter 4 have proved to be capable of capturing this effect, and a small uniform gap of 0.08 mm around the dowel was shown to produce relative joint displacements very close to those observed in both experiments. Comparisons between measured and computed strains were less accurate than displacements. Further, for the second model the computed strains were tensile and the measured strains reported as compressive, with no apparent explanation.

Chapter 8

Parametric Studies

8.1 Introduction

This Chapter presents two parametric studies on joint shear transfer effectiveness. The first study examines the effect of dowel looseness on pavement response; the second focuses on the changes in aggregate interlock shear transfer efficiency and pavement response due to variations in contraction joint opening. The intent of these studies is three-fold: (1) they illustrate the effectiveness of the proposed techniques for modeling dowel and aggregate interlock shear transfer; (2) they provide valuable information on the response of rigid pavements under applied wheel and thermal loadings due to realistic variations in joint shear transfer efficiency; (3) they indicate what future research needs are required to more completely model rigid pavement systems.

8.2 Description of Parametric Studies

8.2.1 Idealized System and Loads

Both the dowel and aggregate interlock parametric studies employ the same basic system shown in Fig. 8.1. For simplicity, the joint is not skewed, and the boundaries of the subgrade have not been extended beyond the edges of the slabs. The 150 mm thick base layer is assumed to be a compacted gravel, below which 300 mm of natural soil are modeled as a linearly elastic continuum. A dense liquid is used to represent the remaining natural soil with a modulus of 0.054 MPa/mm. The elastic material properties of the slab and soil are given in Table 8.1, where α is the coefficient of thermal expansion and ρ is the mass density. These values are typical of many designs [64].

Fig. 8.2 shows the layout of the dowels, which is consistent with a typical retrofit design by the Washington State Department of Transportation (WSDOT) and similar to those

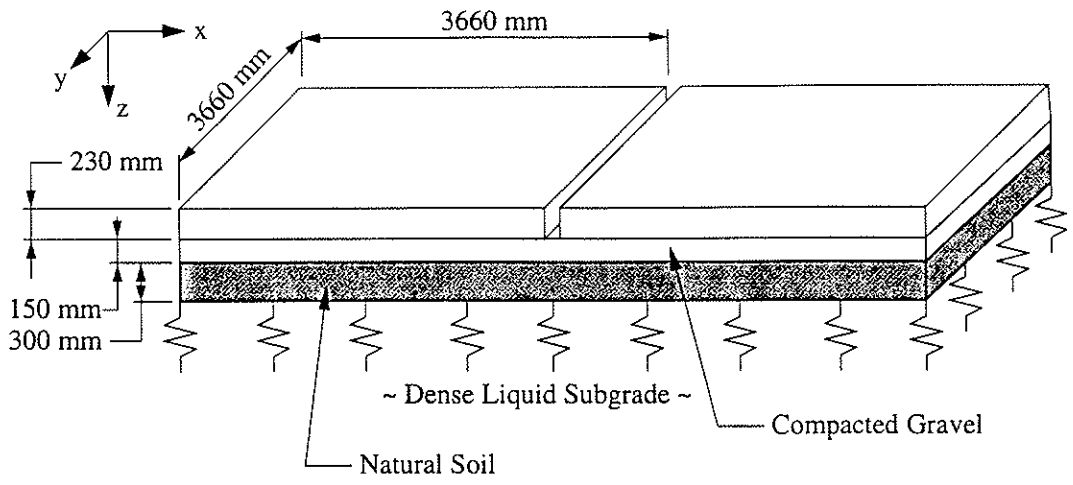


Figure 8.1: Model of Two Rigid Pavement Slabs

Table 8.1: Material Properties of Slab and Subgrade

Portion of System	E MPa	ν –	α $^{\circ}C^{-1}$	ρ kg/m^3
Slabs	28000	0.25	1.1×10^{-05}	2400
Gravel Base Layer	150	0.20	–	–
Natural Soil	75	0.20	–	–

used by other agencies [32]. The dowels are assumed to be debonded along their entire length, except at a single node as required to prevent rigid-body motion. The horizontal boundary conditions used to prevent rigid-body motion of the slabs are also indicated in Fig. 8.2. The finite element meshes used in the analyses are shown in Fig. 8.3. Note that the dowels are not included in the aggregate interlock study and that the coarser meshes are required by the multigrid solver.

For each study, a single 80 kN axle load – the usual equivalent single axle load assumed in pavement design – will be applied near the joint as shown in Fig. 8.2. The axle has dual wheels, and each wheel is idealized as a uniform pressure acting over a 180 mm long by 180 mm wide rectangular area. The wheel spacing of the axle is typical [59], and the tire contact region was selected to give a realistic uniform pressure of approximately 90 psi. The transverse axle position was chosen based on a typical design case used by WSDOT [94]. Since separation of the slab and base will occur, the slab self-weight must also be included in the analyses.

The doweled model will be subjected to a second load case consisting of the same axle

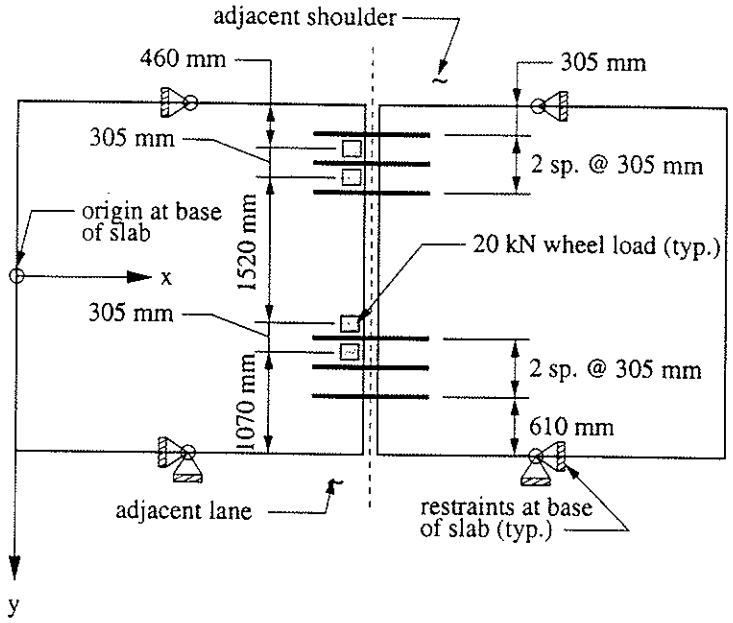
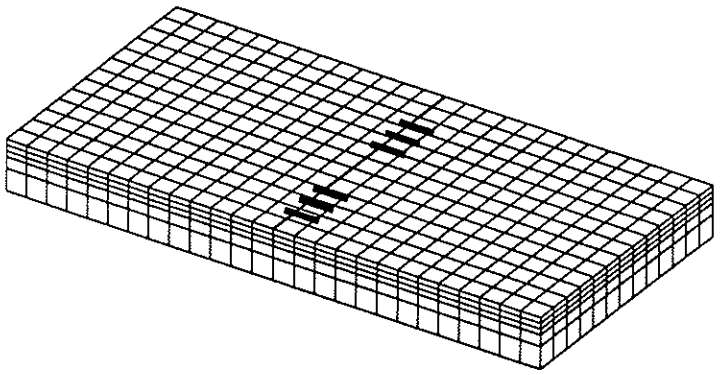
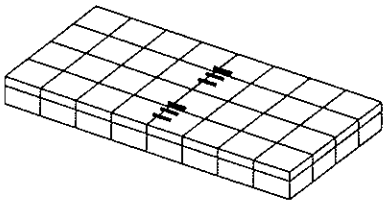


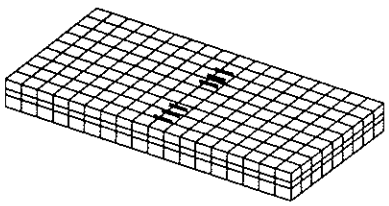
Figure 8.2: Plan View of Slabs Showing Dowels and Wheel Loads



Mesh with 29721 degrees of freedom



Mesh with 2439 degrees of freedom



Mesh with 10617 degrees of freedom

Figure 8.3: Finite Element Meshes used in Parametric Study

load acting in conjunction with a linear temperature gradient through the slab thickness varying from $-4°C$ at the top surface to $4°C$ on the bottom.

8.2.2 Model Details: Dowel Looseness

The model used to study the effects of dowel looseness on pavement response is typical of a retrofitted system as shown in Fig. 8.2. No aggregate interlock shear transfer is assumed across the joint, which is reasonable for pavement slabs which have been in service for a number of years and require joint retrofit. The single parameter considered in the study on dowel looseness is the value of the gap between the dowel and surrounding slab at the joint face. As discussed in Chapter 4, dowel looseness can arise from poor construction techniques [81] as well as damage to the surrounding concrete under cyclic loading [10], and magnitudes of the gap have been measured at up to 0.6 mm. Fig. 8.4 details the assumed gap, γ, which tapers parabolically from a maximum value to zero over one-half the embedded length of the dowel. The 12.7 mm joint opening is artificial, and is used only to ensure that the dowel element spanning the joint is not so stiff that it causes conditioning problems. The

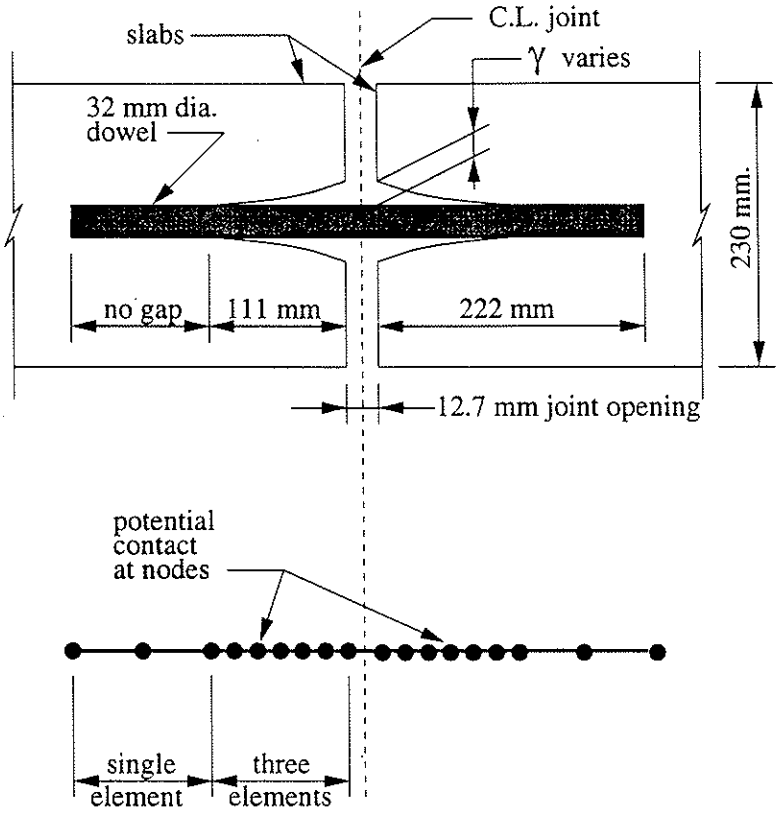


Figure 8.4: Cross-Sectional Detail of Dowel at Joint and Discretization of Dowel

magnitude of the gap at the joint face varies from 0 mm up to the value beyond which there is no significant effect on the pavement response. The number of elements in the region of the gap is fixed at three, giving six potential points of contact – note the nodal locations shown in 8.4. Recall that the convergence studies presented in Chapter 4 indicated that a total of six dowel elements in each slab, corresponding to three elements in the region of the gap, was sufficient to ensure convergence of displacements and dowel shears.

8.2.3 Model Details: Aggregate Interlock

This study examines the effect of initial joint opening, w, on aggregate interlock shear transfer. This is a very practical concern, as seasonal temperature changes can cause significant variations in w. The constitutive relations used to model aggregate interlock shear transfer across the joint are developed using a discretized version of Walraven's model as detailed in Chapter 5. The value of σ_{pu} was set at 45 N/mm^2, and $\mu = 0.4$. The fraction of aggregate particles in the concrete was fixed at 0.75, and a maximum particle diameter of 20 mm was assumed. The number of different embedments and diameters assumed in the analysis was 30. A total of 50 curves, each defined by 31 discrete points, was generated for constant crack openings of between 0.05 mm and 3.2 mm, with each curve capturing relative shear displacements between 0 mm and 1.0 mm. Plots of the shear and normal stresses corresponding to five selected values of crack opening are given in Fig. 8.5.

One modeling detail which must be addressed is the effect of the sawcut used to form rigid pavement contraction joints. As shown in Fig. 8.6, the upper quarter to third of the slab thickness is sawcut and filled with a joint sealant; as a result, there is no interlock shear transfer between slabs in this region. To properly model this phenomenon, the 16-noded interface element spanning the joint between the upper slab elements (covering the upper third of the slab thickness) is assigned a large initial joint opening and zero strength. Note that the 12.7 mm joint opening is artificial, and is only used to provide separation of the mesh between the slab faces.

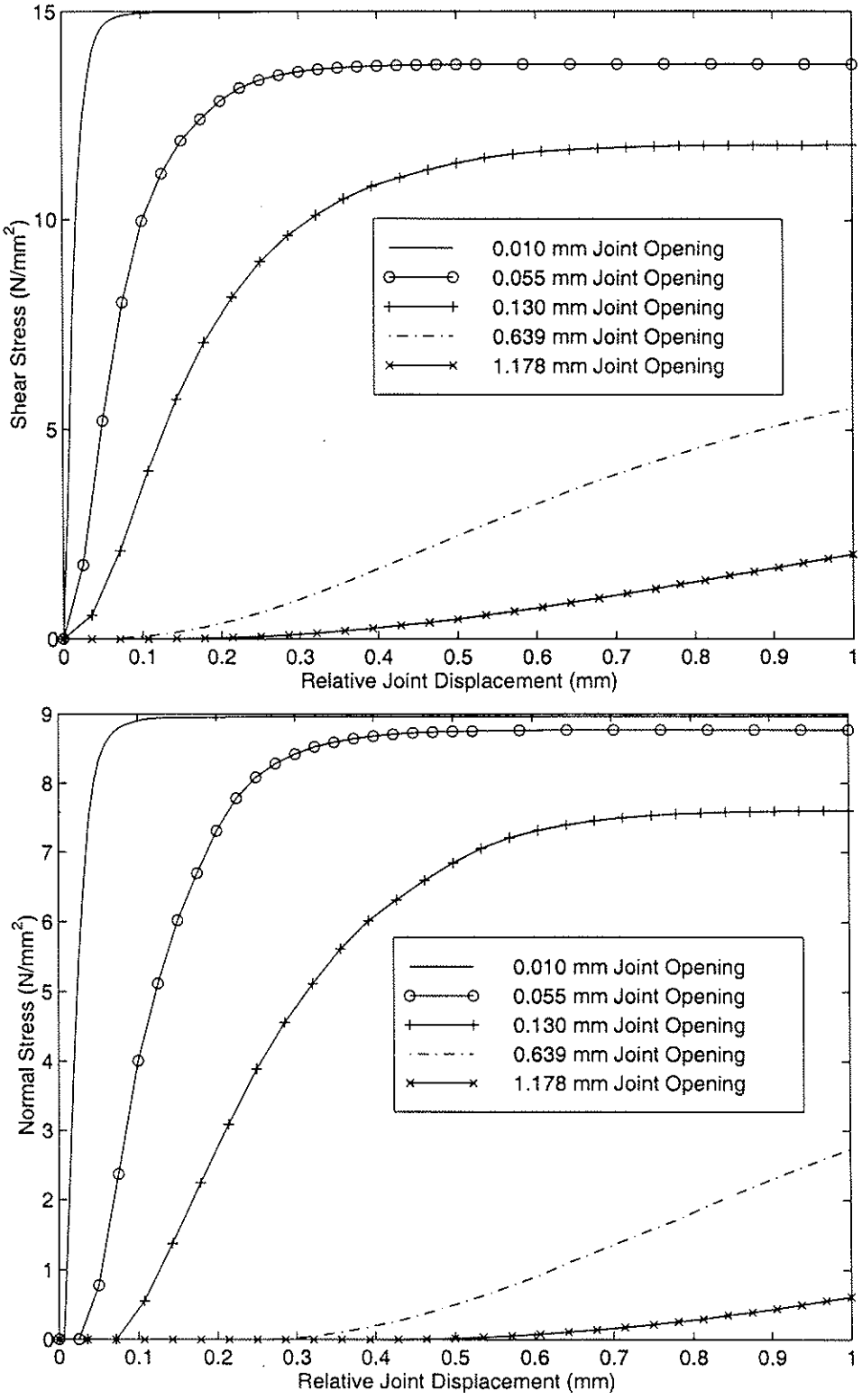


Figure 8.5: Aggregate Interlock Parametric Study: Joint Stress-Displ. Relations

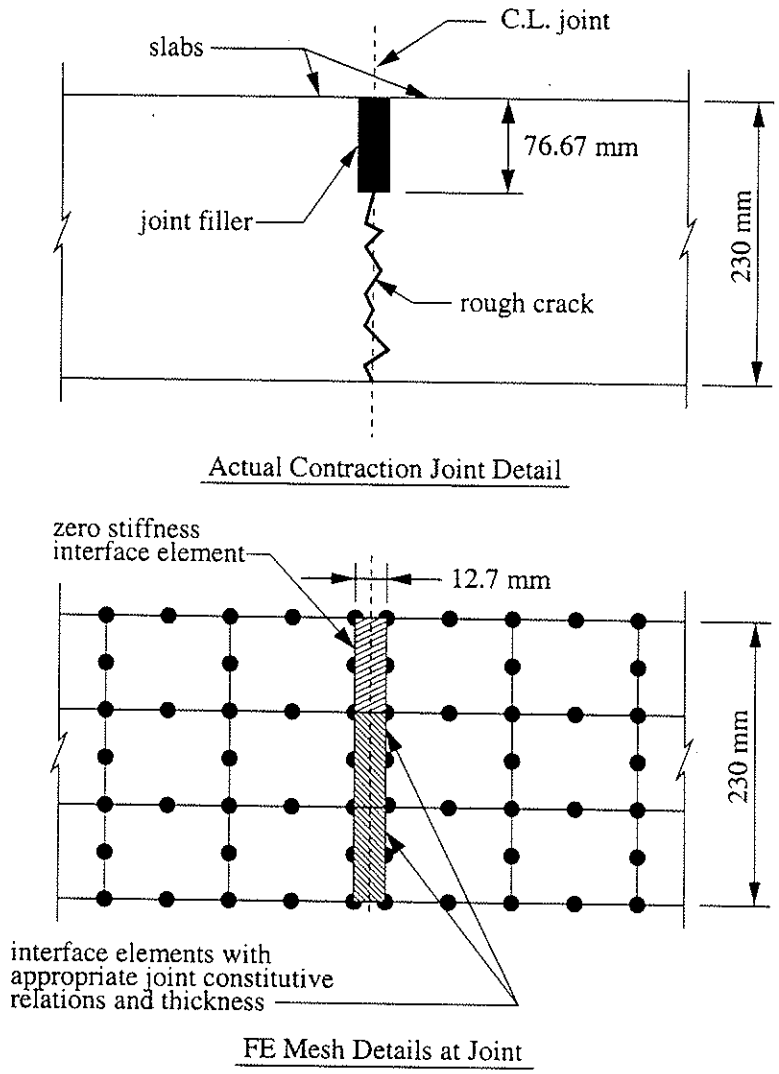


Figure 8.6: Aggregate Interlock Parametric Study: Detail of Joint

8.3 Results: Doweled Joint, Axle Loading

8.3.1 Displacement Response

As expected, the relative vertical displacements of the two slabs near the joint grow with an increase in γ. Figure 8.7 shows the variation in displacement load transfer efficiency (LTE) as defined in Chapter 4 with the magnitude of the gap. The LTE has been computed after subtracting the vertical displacement of the slabs due to their self-weight of 0.10 mm. Note that since the axle and dowel locations are unsymmetric, the slab displacements vary significantly across the joint; for this reason, the LTE has been measured at $(x, y, z) = (3660, -1266.9, -230)$ and $(3660, 563.1, -230)$, which correspond to the nodes closest to the center of each dual wheel load at the joint face.

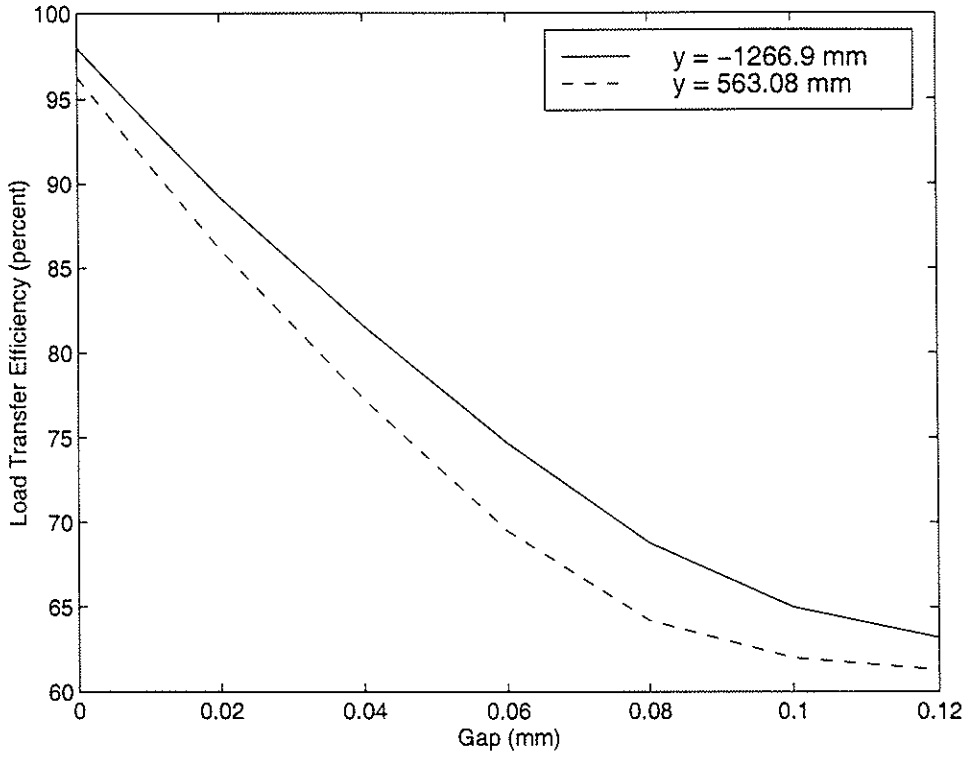


Figure 8.7: Load Transfer Efficiency vs. Gap – Doweled Joint, Axle Loading

The largest gap considered in the analyses is only 0.12 mm – beyond this value, there is no significant change in the model response. This contrasts with the maximum gap of 0.30 mm determined in the preliminary studies of Chapter 4 for a similar system with the same gap profile. This discrepancy is due largely to the subgrade models: the subgrade of the system presently under consideration is stiffer than the dense liquid foundation used

previously, so a smaller gap is required to produce the maximum decrease in LTE. Also note that the maximum reduction in LTE is about 37%, as opposed to 52% determined in Chapter 4. This difference can also be attributed to the different subgrade models. In the present model, as the loaded slab contacts the subgrade and displaces it downward, there is a loss of support under the unloaded slab. With a dense liquid directly below the slab as used in Chapter 4, the subgrade under the unloaded slab provides continuous support that is independent of the displacement of the loaded slab, tending to increase relative vertical displacements between the two slabs.

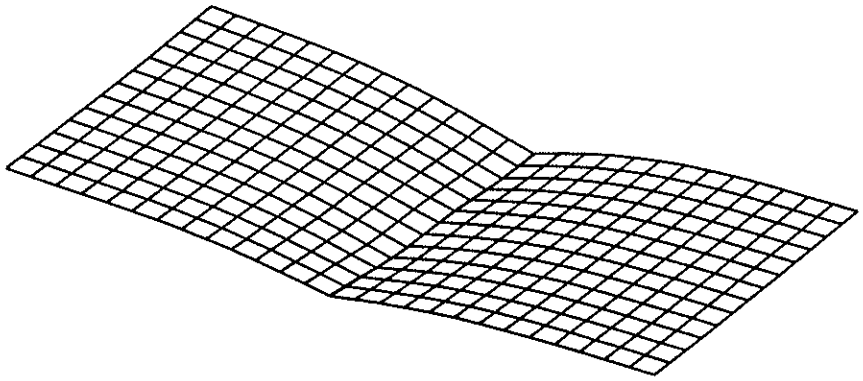
Figure 8.8 shows the displaced shape of the top surface of the slabs for selected values of the gap; the vertical displacement component has been scaled by a factor of 1000. Note the larger vertical displacements of the slab on the side where the wheel loads are closer to the edge. The vertical displacement of the loaded slab at $(x, y, z) = (3660, -1266.9, -230)$ varies from 0.56 mm for a gap of 0.00 mm to 0.64 mm for a gap of 0.12 mm; the corresponding vertical displacements at $(3660, 563.1, -230)$ vary from 0.48 mm to 0.55 mm.

8.3.2 Dowel Shears and Contact Forces

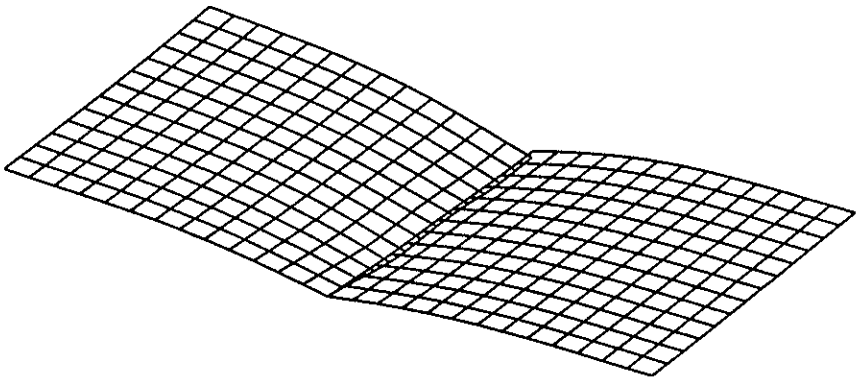
The loss in load transfer efficiency with gap is naturally paralleled by decreasing shear transfer due to dowel action. Fig. 8.9 details the change in shear transmitted across the joint for each dowel. Note the ineffectiveness of the dowel at $y = 1220$ mm, which is 610 mm from the nearest wheel load. For smaller gaps, the largest shears are transmitted by the dowel at $y = 610$ mm, which is centered between two wheels. This illustrates the localized nature of dowel shear transfer also noted by other researchers [41]. At larger values of γ, the dowel shears converge to nearly constant values with magnitudes ranging between about 0.7 kN and 2 kN.

It is also worthwhile to examine the nodal forces applied to the dowels by the surrounding slab for various values of γ. Figs. 8.10 through 8.13 provide this information for each dowel at selected values of γ; refer to Fig. 8.4 for the nodal locations corresponding to the x axis coordinates in Figs 8.10 through 8.13.

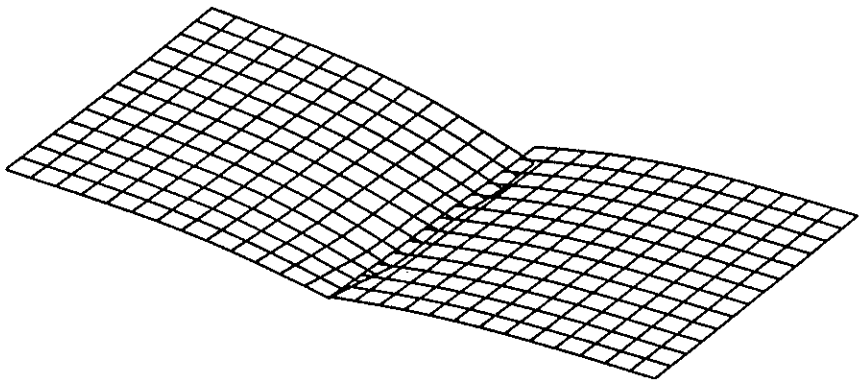
For $\gamma = 0.00$ mm, the largest contact forces occur at the joint face, and decrease rapidly in an oscillatory manner similar to a beam on elastic foundation. When $\gamma = 0.04$ mm, the point of contact is at the joint face for the loaded slab, but it has decreased significantly in magnitude; the portion of the dowel in the loaded slab sees more uniformly distributed contact along its length with smaller nodal forces. At $\gamma = 0.08$ mm, the contact forces are distributed over the dowel within both the unloaded and loaded slabs. As the gap increases



Gap = 0.00 mm



Gap = 0.04 mm



Gap = 0.12 mm

Figure 8.8: Displaced Shape of Top of Slab – Doweled Joint, Axle Loading

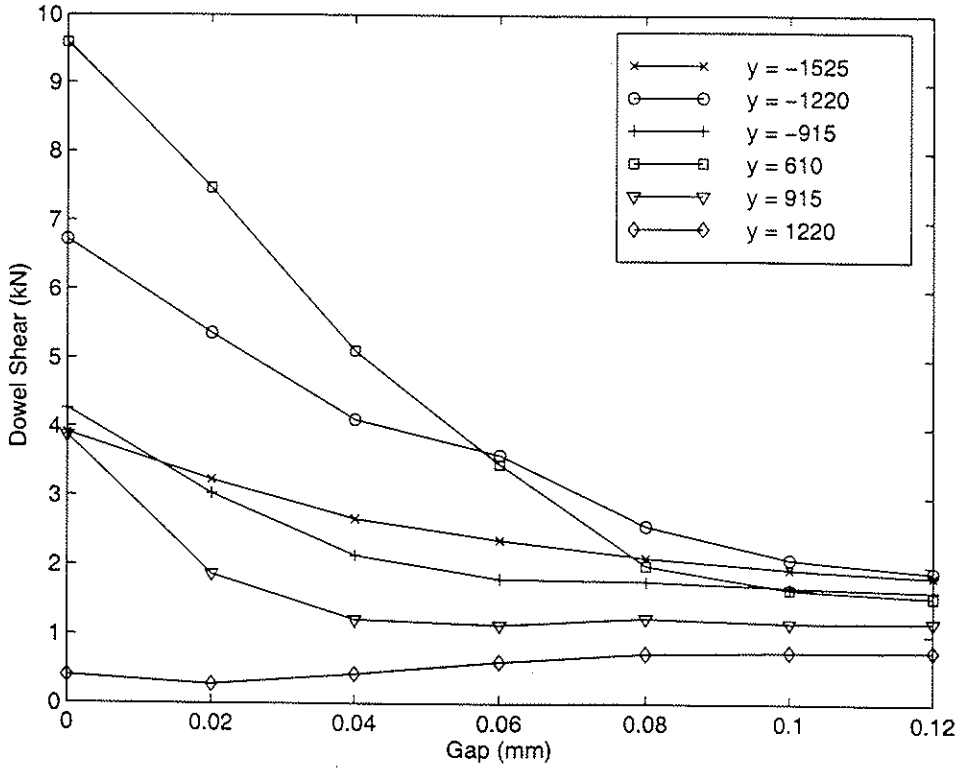


Figure 8.9: Dowel Shear vs. Gap – Axle Loading

to 0.12 mm, the points of contact are fewer and further back from the joint face, but the contact forces increase in magnitude vs. $\gamma = 0.08$ mm.

It must be noted that different shapes for the gap profile (i.e. linear, cubic, etc.) could be expected to produce significantly different contact patterns and forces. However, assuming that the formation of the gaps is due to degradation of the concrete surrounding the dowels under cyclic loading, and that the degradation is a function of the magnitude of the compressive stresses between the dowel and the slab, the assumption of a parabolic gap is not unreasonable given the oscillatory decrease in contact forces predicted when $\gamma = 0$ mm.

8.3.3 Slab and Subgrade Stresses

Given that increasing γ significantly decreases the dowel shears, we would expect an increase in subgrade stresses under the loaded slab adjacent to the joint with a concomitant decrease in subgrade stresses under the unloaded slab. Fig. 8.14 shows the magnitude of the vertical stress between the slab and subgrade for selected values of γ. Note the localized high stresses under the edge of the loaded slab for all values of the gap. As γ increases, the unloaded slab transfers less load to the subgrade, and the region of separation between the unloaded

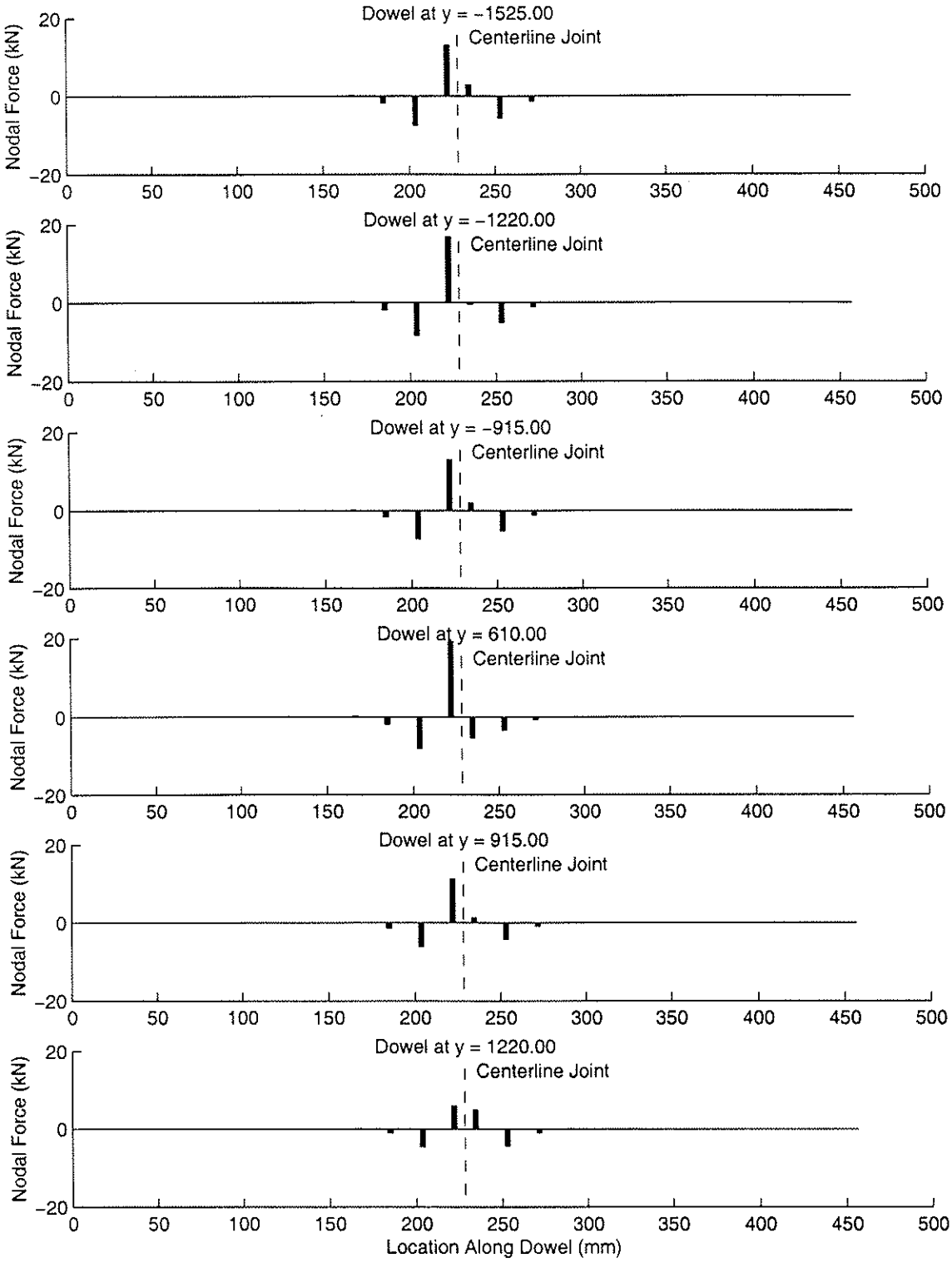


Figure 8.10: Nodal Forces on Dowels for 0 mm Gap – Axle Loading

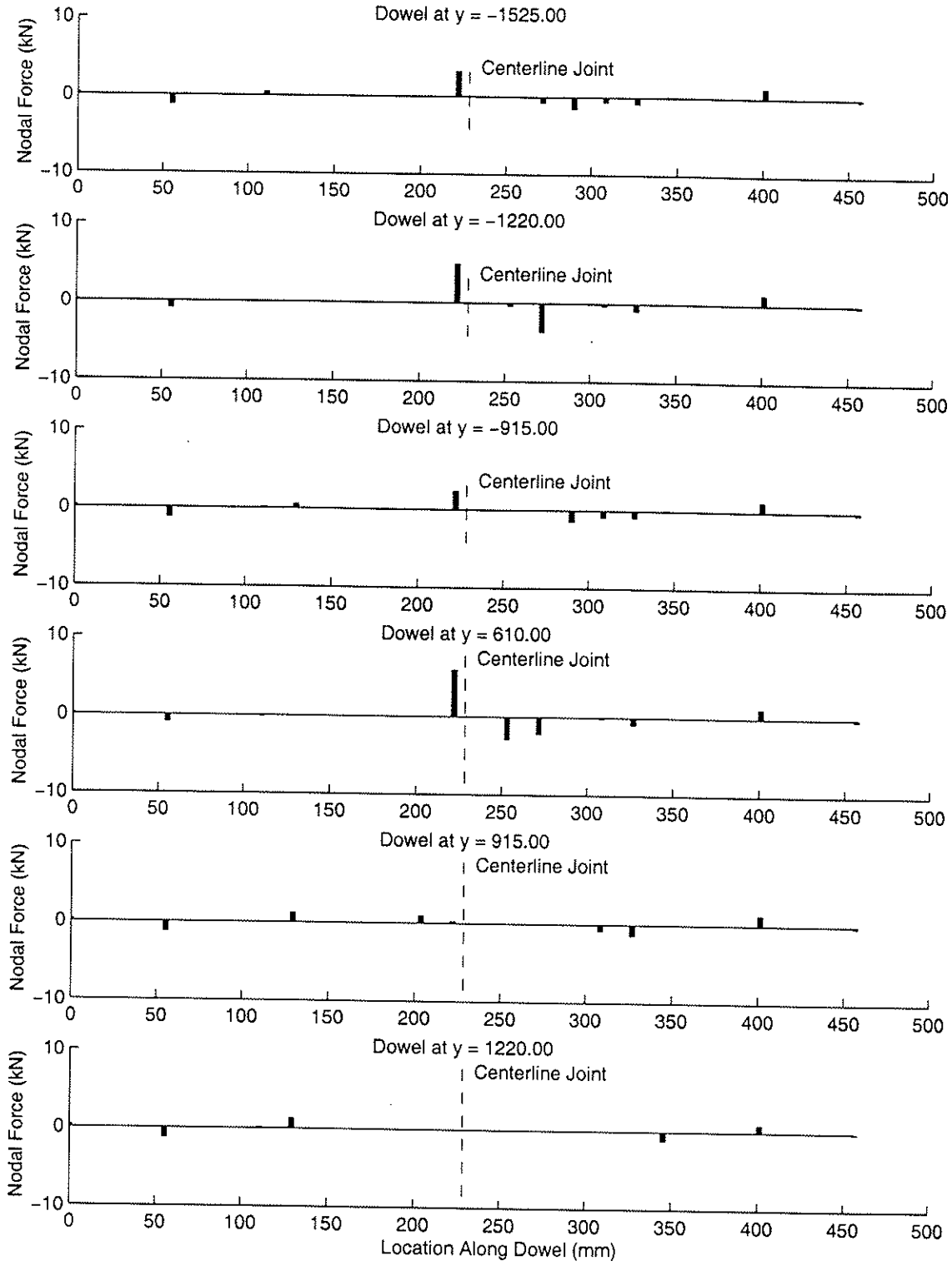


Figure 8.11: Nodal Forces on Dowels for 0.04 mm Gap – Axle Loading

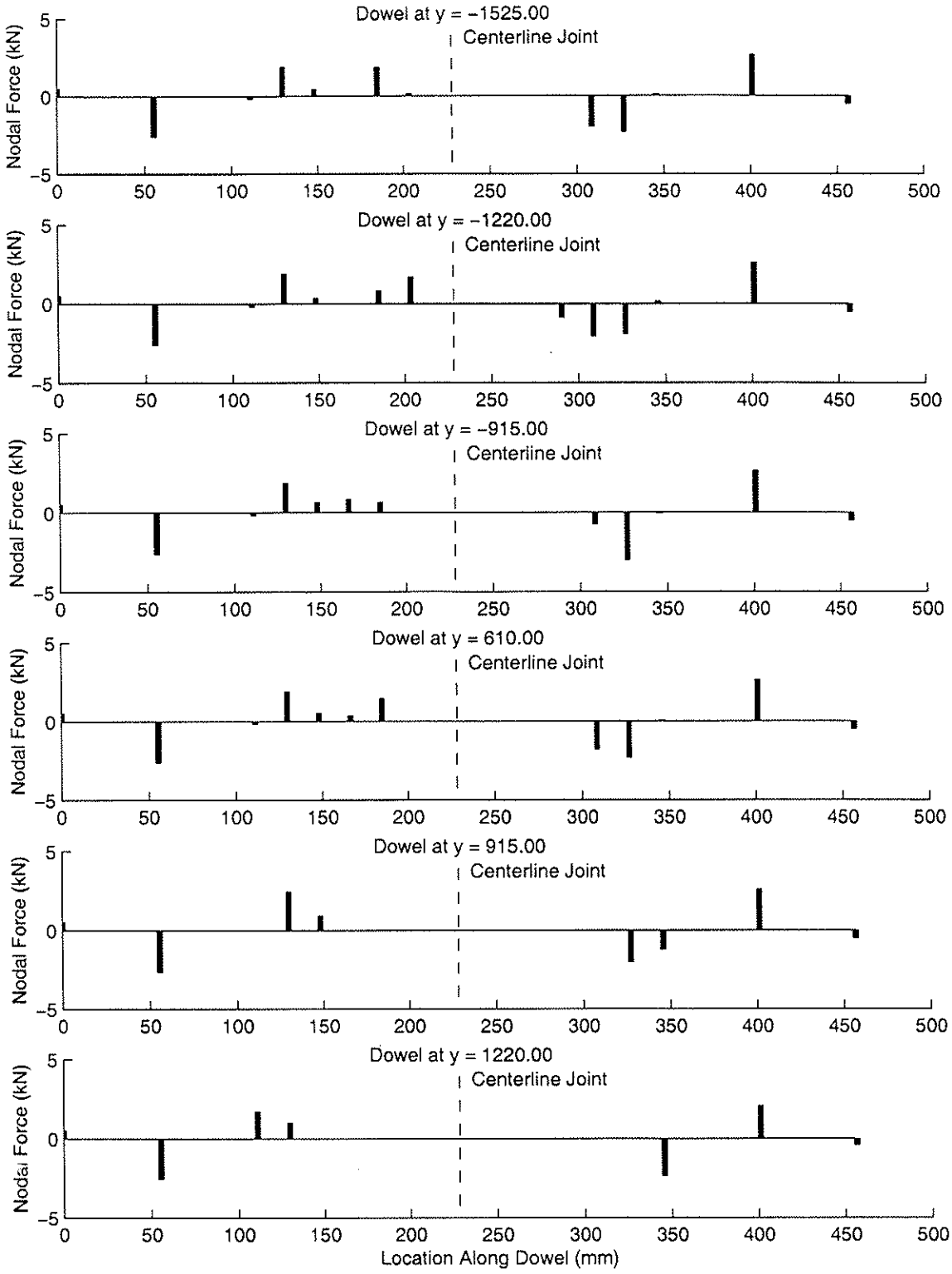


Figure 8.12: Nodal Forces on Dowels for 0.08 mm Gap - Axle Loading

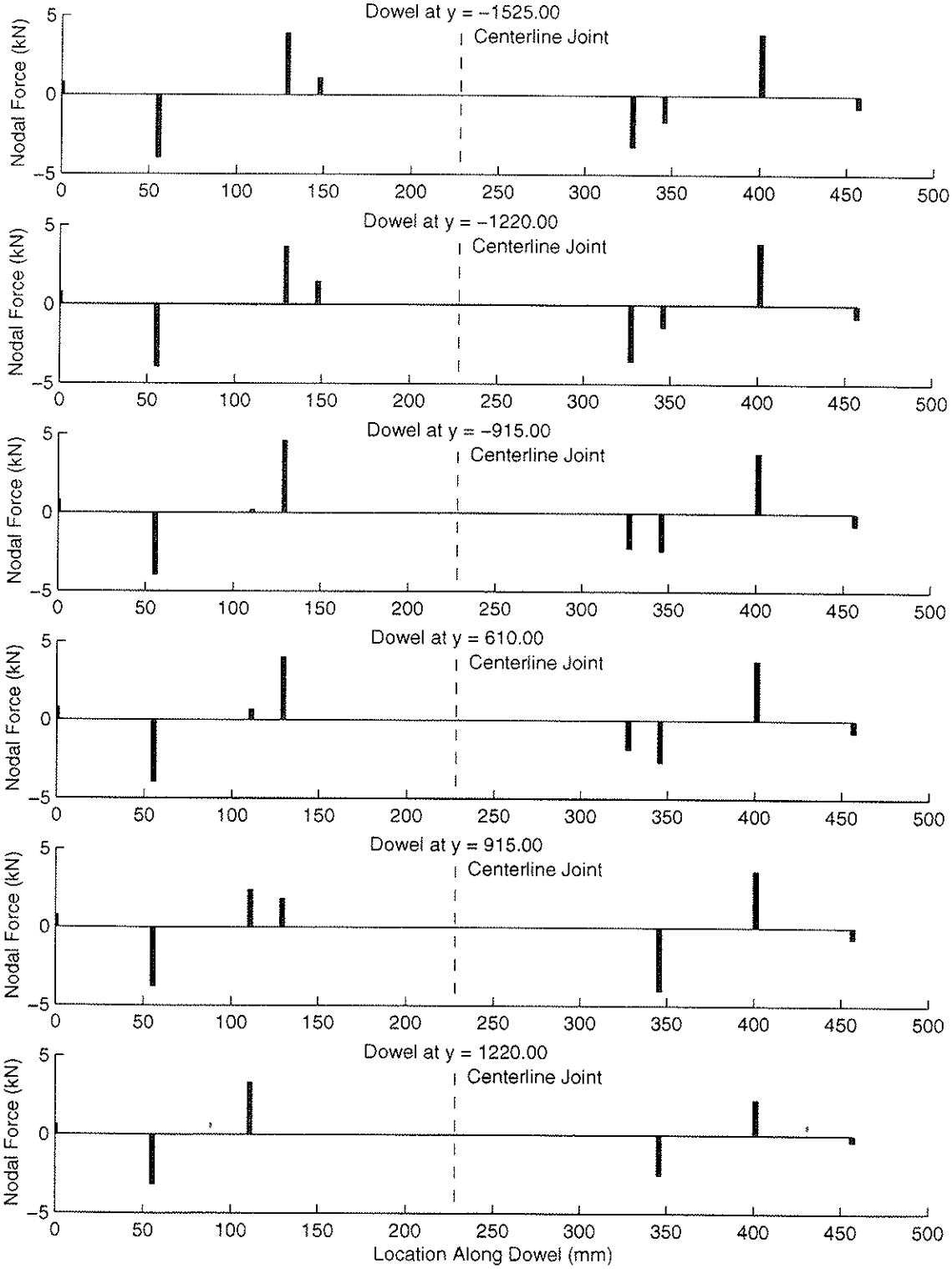


Figure 8.13: Nodal Forces on Dowels for 0.12 mm Gap - Axle Loading

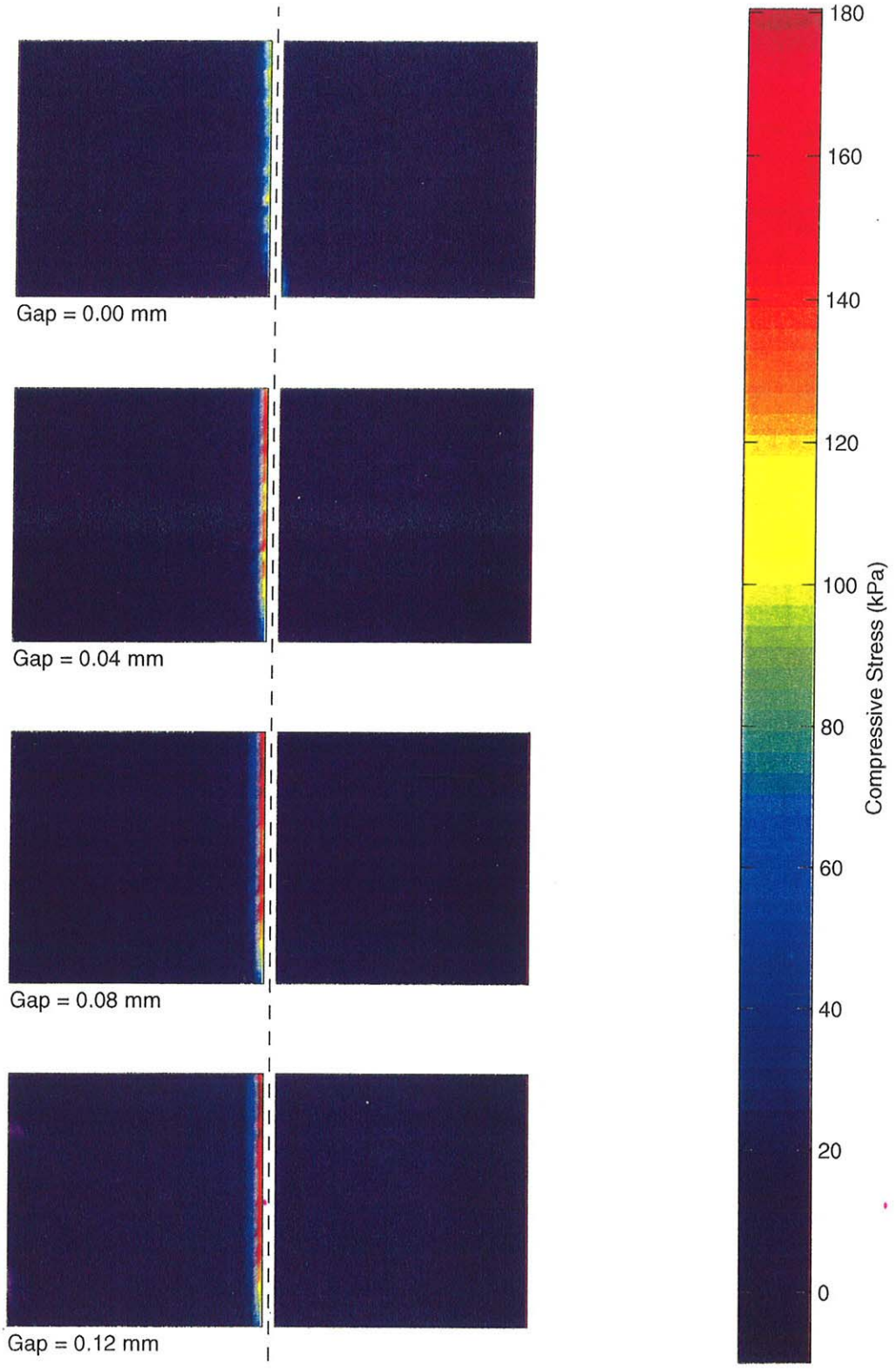


Figure 8.14: Vertical Stress on Top of Base Layer - Doweled Joint, Axle Loading

slab and subgrade increases as evidenced by larger regions of zero or near-zero stresses. The maximum stress applied to the subgrade by the loaded slab varies from 107 kPa to 180 kPa as the gap varies from 0.00 mm to 0.12 mm – an increase of 68%. It must be noted that these stress values are somewhat unrealistic due to the likelihood of subgrade yielding which is not captured by the linearly elastic model. However, the fact that they increase dramatically with the presence of small values of γ is significant: as detailed in Chapter 1, loss of subgrade support and strength due to pumping action and subgrade deterioration under high stresses is suspected of being a significant component of many joint failures [46].

It is also illuminating to examine the principal tensile stresses in the bottom of the slab, as they increase significantly with increasing gaps. Figure 8.15 shows the maximum principal stresses on the bottom of the slab, with the maximum values occuring under each dual wheel at the joint face. The maximum values vary from 667 kPa to 1037 kPa with gaps of between 0.00 mm and 0.12 mm, an increase of 55%. The orientation of the maximum principal stress is in the y direction (transverse): effectively, as γ grows and the unloaded slab provides less support to the loaded slab, the stresses in the loaded slab approach those due to an unprotected edge loading condition.

8.4 Results: Aggregate Interlock, Axle Loading

8.4.1 Displacement Response

A similar relationship between LTE and joint opening, w, for an undoweled joint with aggregate interlock exists as between LTE and γ for a doweled joint. As w increases, the effectiveness of aggregate interlock as a shear transfer mechanism is reduced, leading to smaller LTEs as shown in Fig. 8.16. The values of w observed in practice do not typically reach 3.20 mm (the maximum value assumed in this study). The following equation is often used to predict joint opening, ΔL [60, 93]:

$$\Delta L = DL(\alpha \Delta T + \epsilon) \tag{8.1}$$

where D = adjustment factor for slab-subgrade friction (0.8 for gravel base layers), L = the transverse joint spacing (3660 mm), ΔT = the uniform change in temperature, and ϵ = strain due to drying shrinkage (typically 1.5×10^{-04}). For the present study, assuming a reasonable value for ΔT of 35°C, Eq. 8.1 gives $\Delta L = 1.6$ mm. Fig. 8.16 indicates that a maximum seasonal change in LTE would be approximately 50%.

The global displacement response of the undoweled system is illustrated in Fig. 8.17,

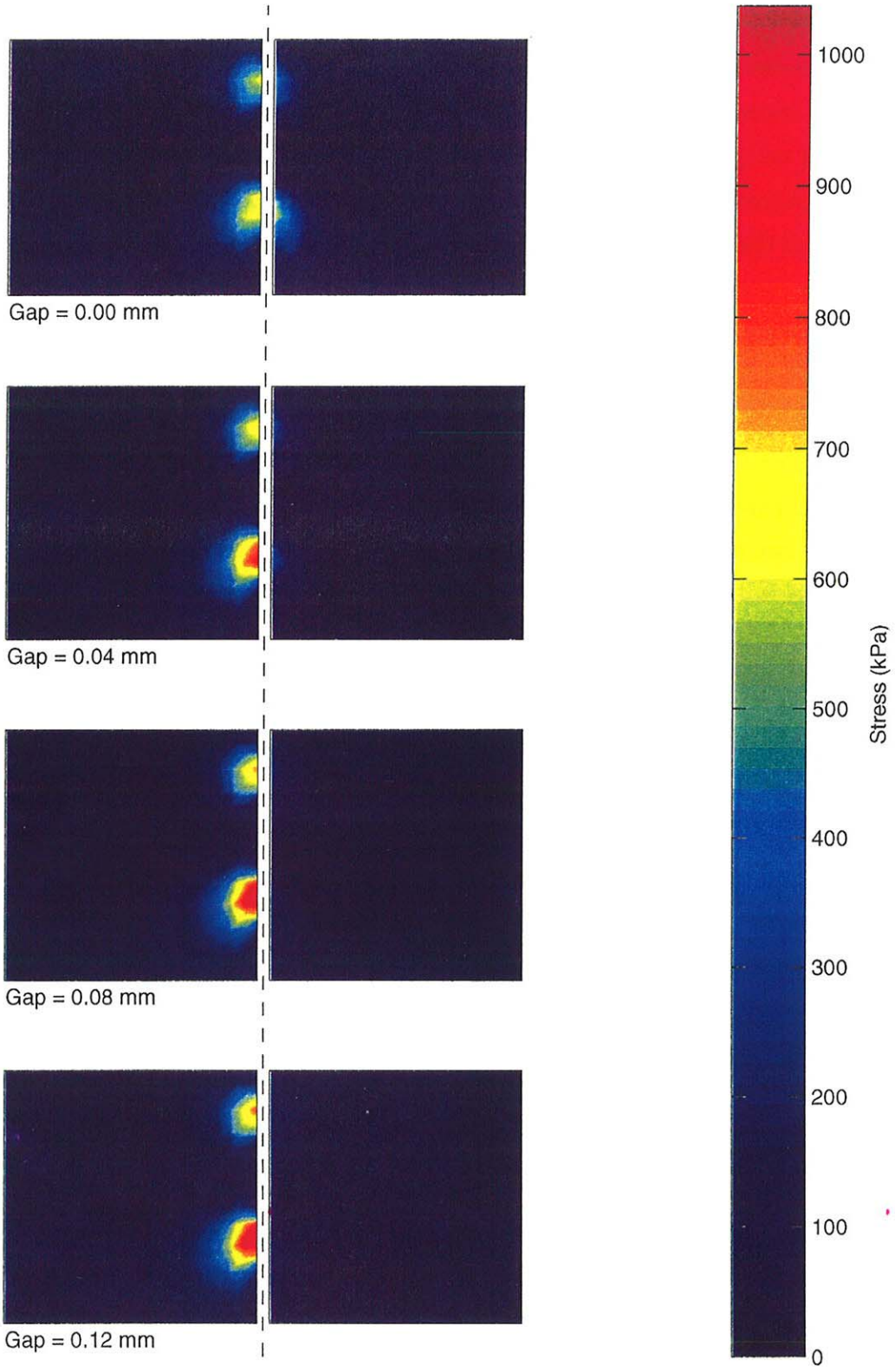


Figure 8.15: Max. Principal Stress on Bottom of Slab – Doweled Joint, Axle Loading

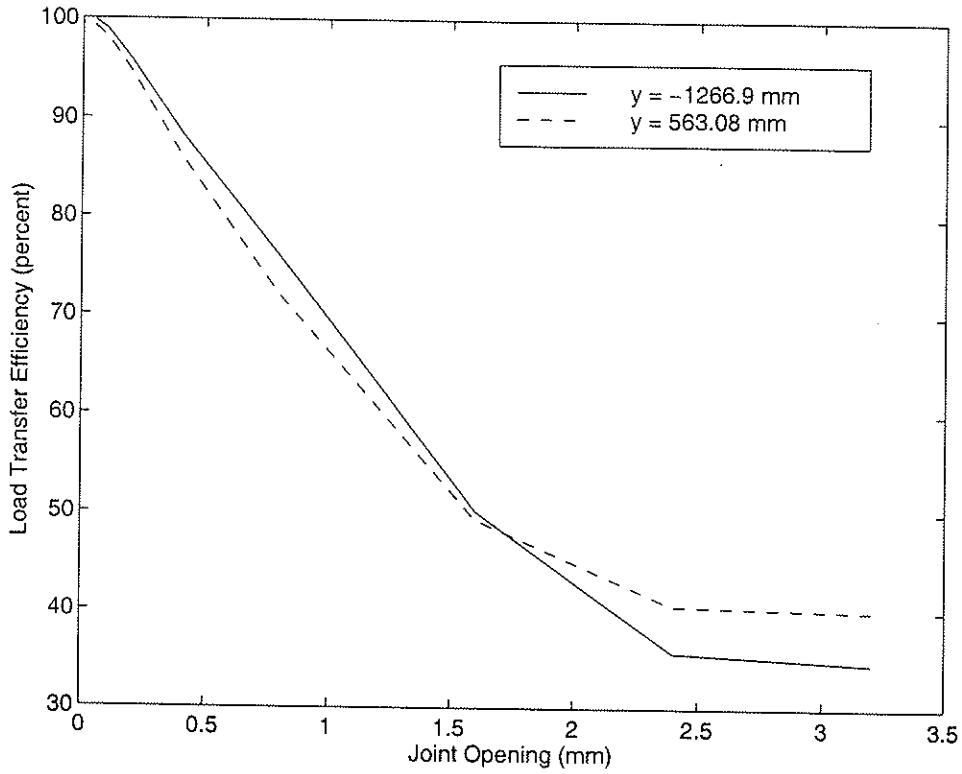


Figure 8.16: Load Transfer Efficiency vs. Joint Opening – Undoweled Joint, Axle Loading

which depicts the displaced shape of the top surface of the slabs for selected values of w. Note the growing discrepancy between the vertical displacements of the loaded and unloaded slabs at the joint with increasing w.

8.4.2 Aggregate Interlock Stresses on Joint Faces

It is particularly interesting to examine the distribution of shear stresses on the joint faces arising from aggregate interlock as shown in Fig. 8.18 for different values of w. The shear stresses are negative, as they are shown acting on the vertical joint face of the loaded slab; of course, identical stresses in the opposite direction exist on the face of the unloaded slab. Note the region of zero and near-zero stresses in the upper third as required by the sawcut detailed in Fig. 8.6; the small non-zero stresses are artifacts of the smoothing and nodal averaging of stresses and subsequent quadratic interpolation within elements. As with dowel joint shear transfer, aggregate interlock shear transfer is a highly localized phenomenon, with only the portions of the joint faces nearest the wheels seeing significant stress. At large joint openings, the shear stresses acting on the joint face are effectively zero, and the unloaded slab provides no support for the loaded slab as shown in Fig. 8.17. The

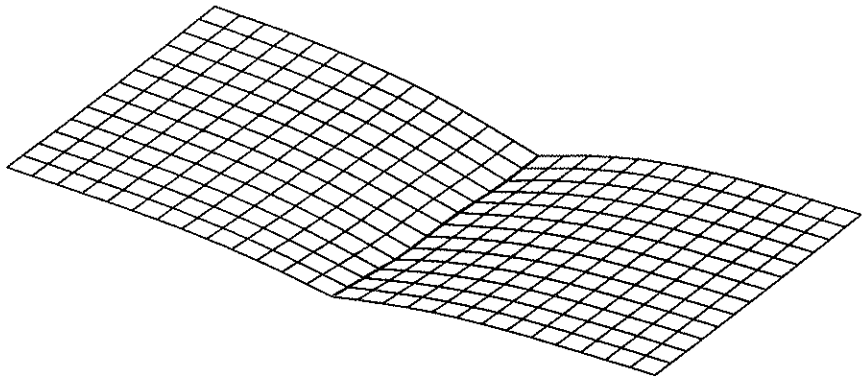
slight curvature of the unloaded slab is due only to self-weight and the separation between the unloaded slab and the base layer near the joint.

The normal stresses occuring on the joint faces have not been shown, as their magnitudes are negligibly small. This is due to the fact that the relative vertical displacements between the loaded and unloaded slabs are quite small, ranging from a few thousandths of a millimeter for a joint opening of 0.05 mm to 0.4 mm for a joint opening of 3.20 mm. Examination of Fig. 8.5 indicates that the normal stresses at the joint would indeed be zero (or negligibly small) for these values of displacement. If the sawcut were not explicitly modeled (see Fig. 8.6), contact between the two joint faces would likely occur at the top of the slab for smaller values of w, which would significantly affect shear and normal stresses.

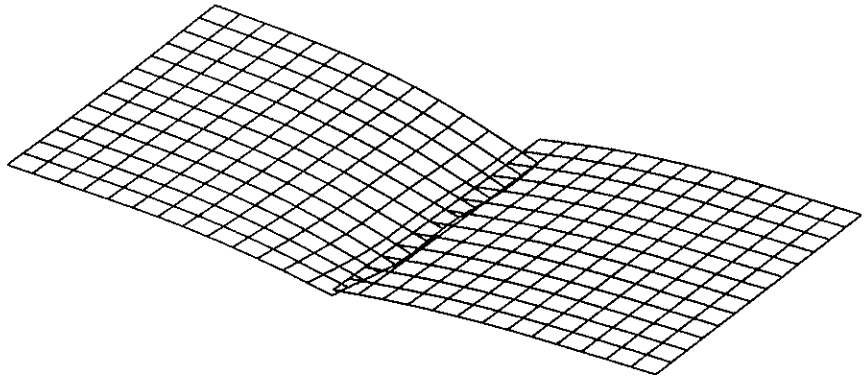
8.4.3 Slab and Subgrade Stresses

As w increases, the vertical stresses applied to the subgrade by the loaded slab increase dramatically as shown in Fig. 8.19. For $w = 0.05$ mm, the vertical subgrade stress under the loaded slab is only 95 kPa; however, for $w = 1.60$ mm, this stress reaches a value of 200 kPa, an increase of 111%. This is a larger range of values than predicted for the doweled joint, since the dowels still transfer shear even for large gaps as no looseness has been assumed over half the dowel length.

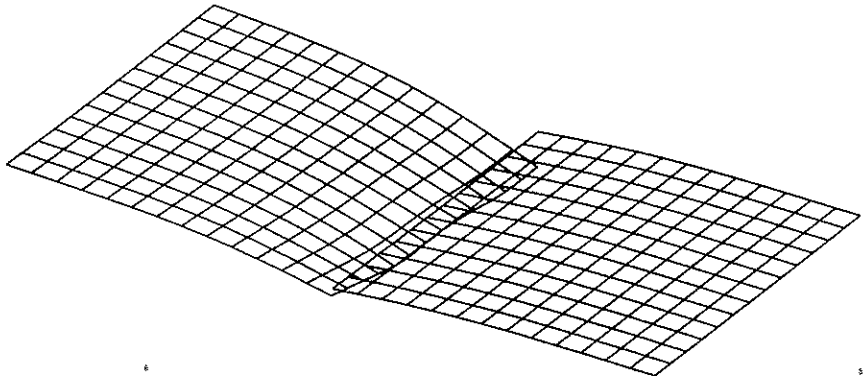
The maximum tensile stresses on the bottom of the slab are shown in Fig. 8.20. They range from 674 kPa to 1107 Kpa in the loaded slab and from 507 kPa to 32 kPa in the unloaded slab. As with the subgrade stresses, this range of stresses is greater than that observed for the doweled joint.



Joint Opening = 0.05 mm



Joint Opening = 1.60 mm



Joint Opening = 3.20 mm

Figure 8.17: Displaced Shape of Top of Slab – Undoweled Joint, Axle Loading

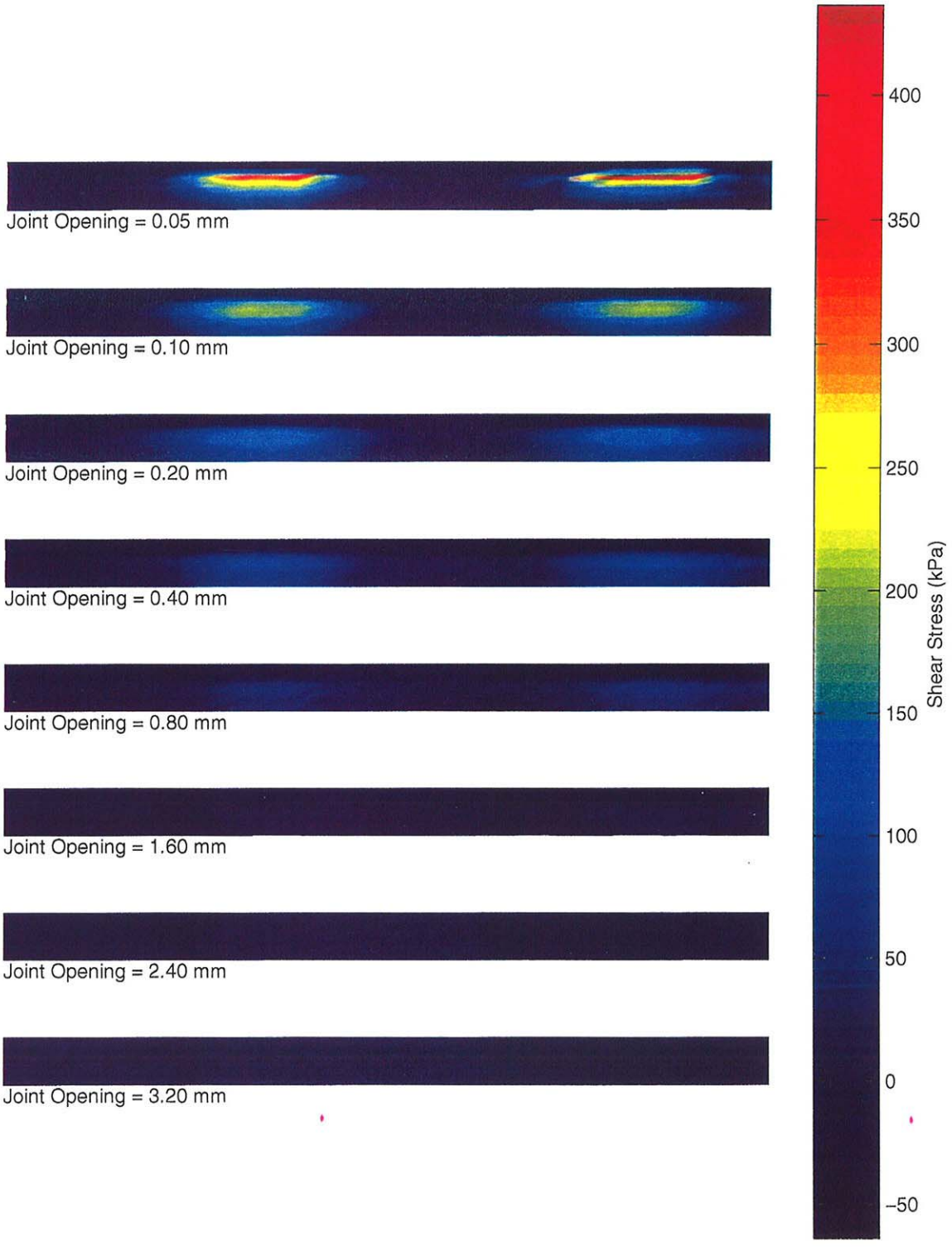


Figure 8.18: Aggregate Interlock Shear Stresses on Joint Faces – Axle Loading

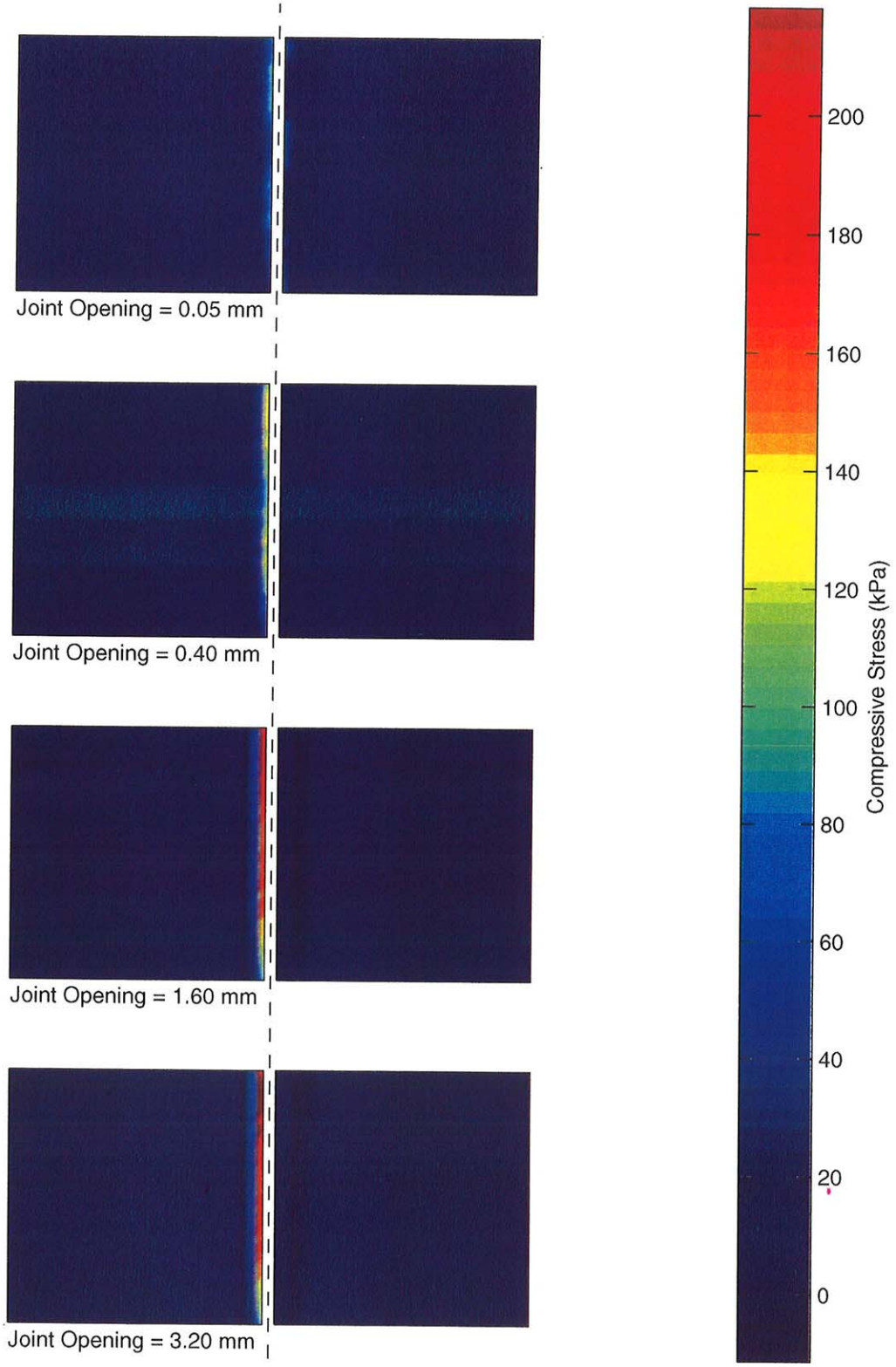


Figure 8.19: Vertical Stress on Top of Base Layer – Undoweled Joint, Axle Loading

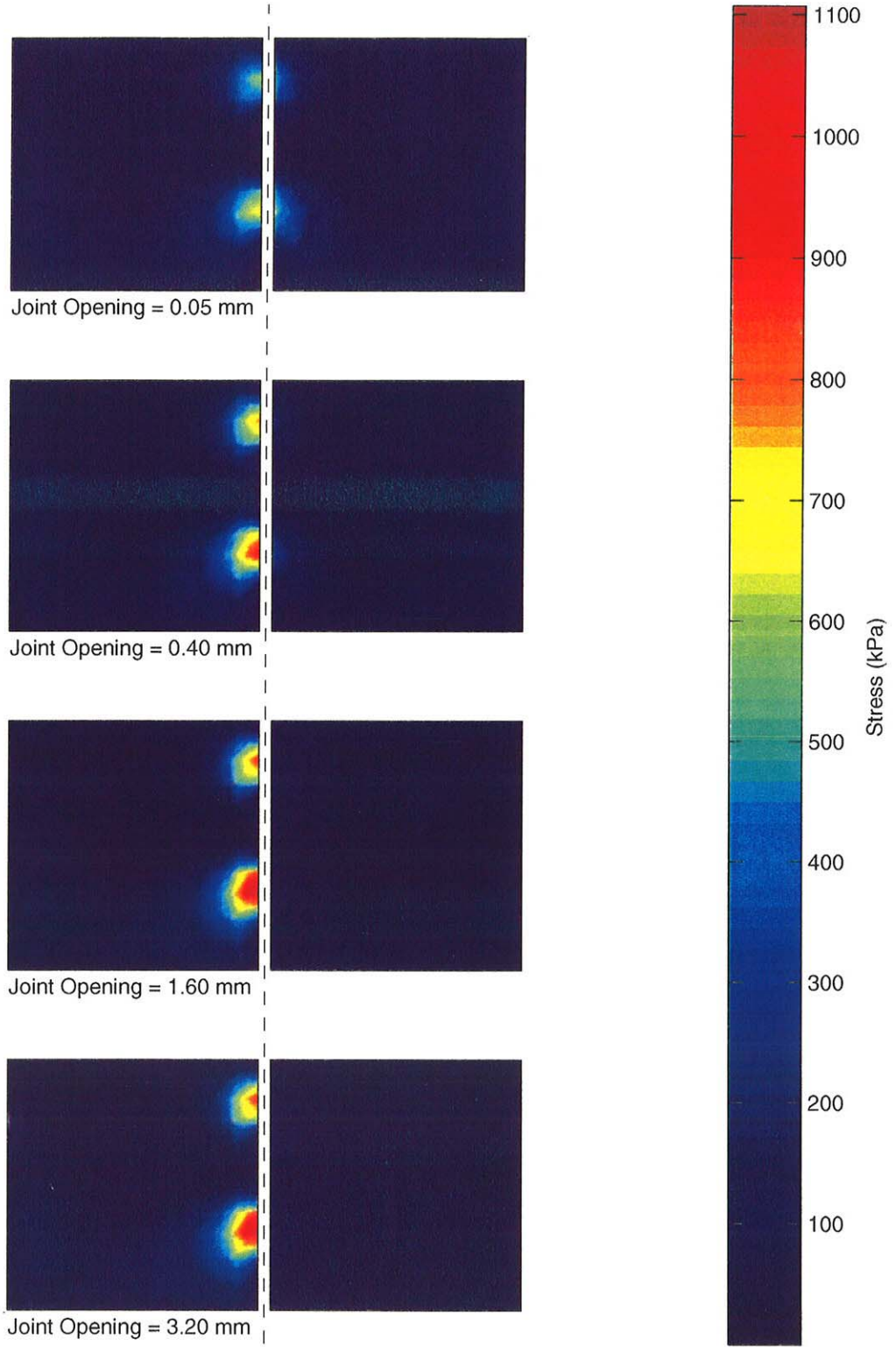


Figure 8.20: Max. Principal Stress on Bottom of Slab – Undoweled Joint, Axle Loading

8.5 Results: Doweled Joint, Axle and Temperature Loading

8.5.1 Displacement Response

Similar to the case of axle loading only, the relative vertical displacements of the two slabs near the joint grow with increasing γ. The maximum value assumed for γ was 0.24, beyond which there is not a significant change in the model response. When computing the LTE, the displacements due to temperature and self-weight alone were determined from separate analyses and subtracted from the displacements due to self-weight, temperature, and axle loading. As for axle loading alone, the LTE has been plotted at $(x, y, z) = (3660, -1266.9, -230)$ and $(3660, 563.1, -230)$, which correspond to the nodes closest to the center of each dual wheel load (see Fig. 8.21). Figure 8.22 shows the displaced shape of the top surface of the slabs for selected values of the gap; the vertical displacement component has been scaled by a factor of 1000.

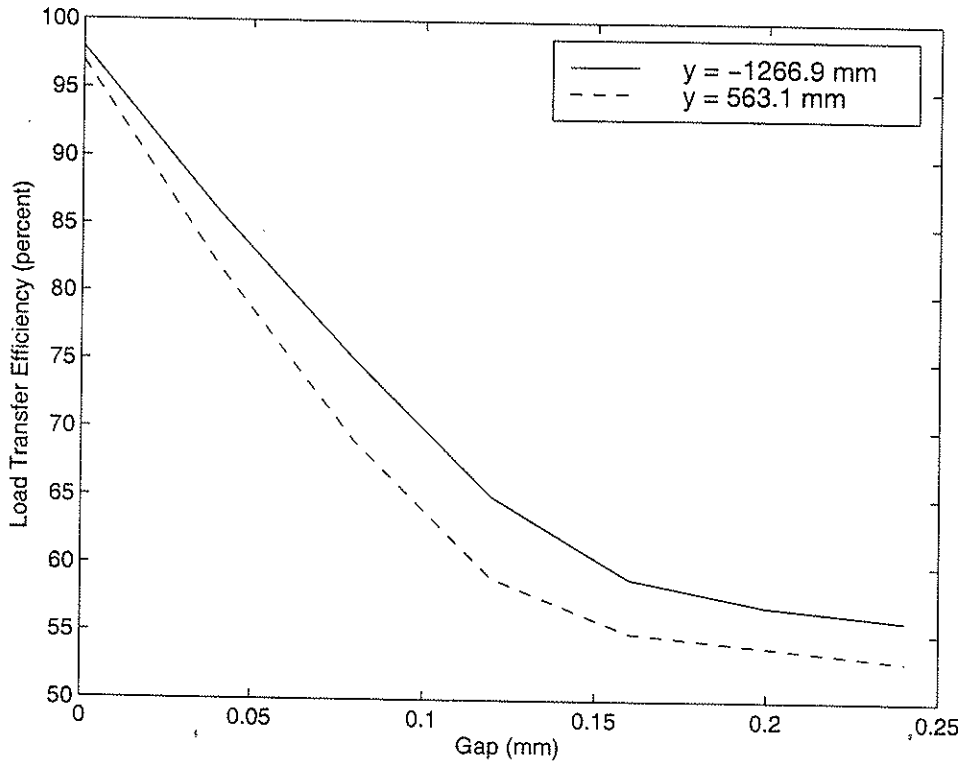
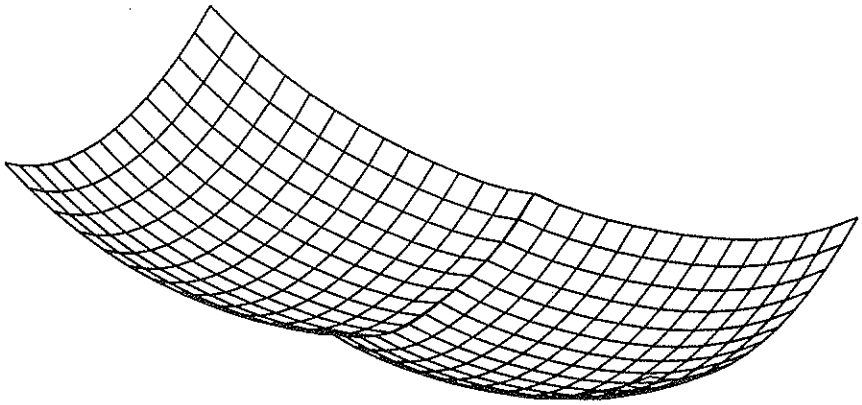
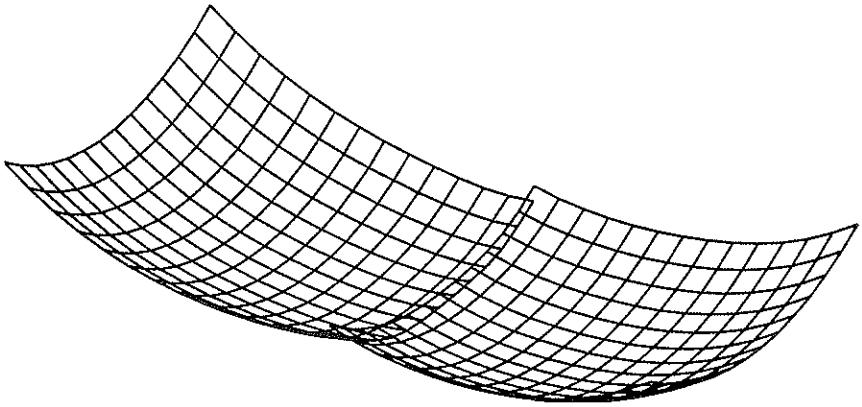


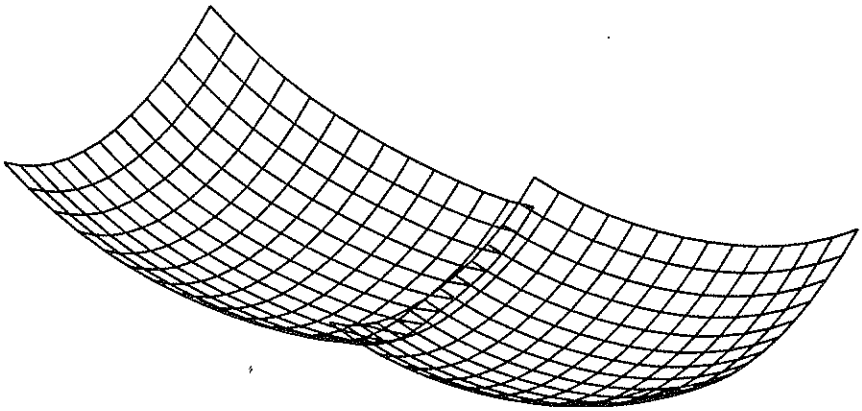
Figure 8.21: Load Transfer Efficiency vs. Gap – Axle and Temperature Loading



Gap = 0.00 mm



Gap = 0.08 mm



Gap = 0.24 mm

Figure 8.22: Displaced Shape of Top of Slab – Doweled Joint, Axle and Temperature Loading

8.5.2 Dowel Shears and Contact Forces

The dowel shears follow much the same pattern with respect to dowel location observed for axle loading only, but with increased magnitudes as shown in Fig. 8.23. This increase in dowel shears is due to the fact that under temperature loading, the slab and base separate and less of the axle load is transferred to the subgrade in the immediate vicinity of the wheels. These larger dowel shears translate into generally larger contact forces between the

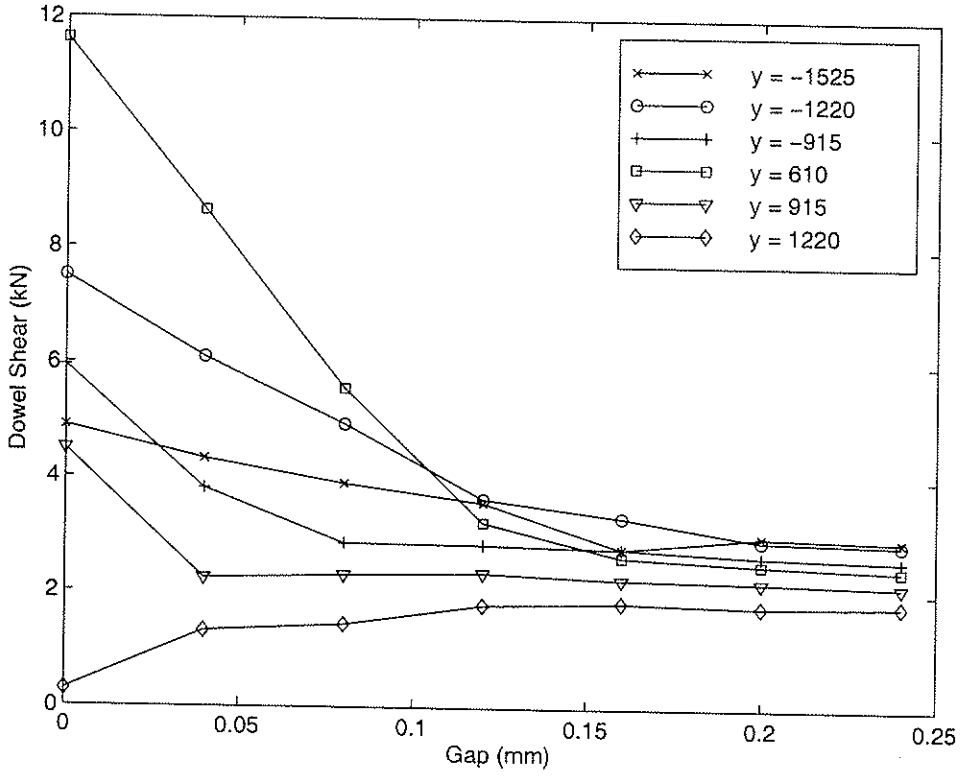


Figure 8.23: Dowel Shear vs. Gap – Axle and Temperature Loading

dowel and the surrounding slab as illustrated in Figs. 8.24 through 8.27.

8.5.3 Slab and Subgrade Stresses

Fig. 8.21 shows increasing LTE with increasing γ, and Fig. 8.22 illustrates the tendency of the loaded slab to "tip" in the direction of the joint for larger values of γ. This results in different contact regions between the slab and subgrade and increasing contact stresses as illustrated in Fig. 8.28. In fact, the maximum stress applied to the base layer varies from 34 kPa to 104 kPa as γ increases from 0.00 mm to 0.24 mm, an increase of 206%. These stresses are still significantly lower than those observed for axle loading only, however.

Nighttime temperature curling generally constitutes a critical load case for tensile stresses on top of the slab. Fig. 8.29 shows maximum principal stress on top of the slab for this condition. Note that the maximum principal stress in each slab is nearly equal when $\gamma = 0.00$ mm. As the gap increases, the maximum principal tensile stress in the loaded slab increases from 890 kPa to 943 kPa with a concomitant decrease in tensile stress in the unloaded slab from 924 kPa to 763 kPa. Note, however, that these stresses are less than the maximum principal tensile stresses observed in the bottom of the loaded slab for axle loading only and the largest value of γ.

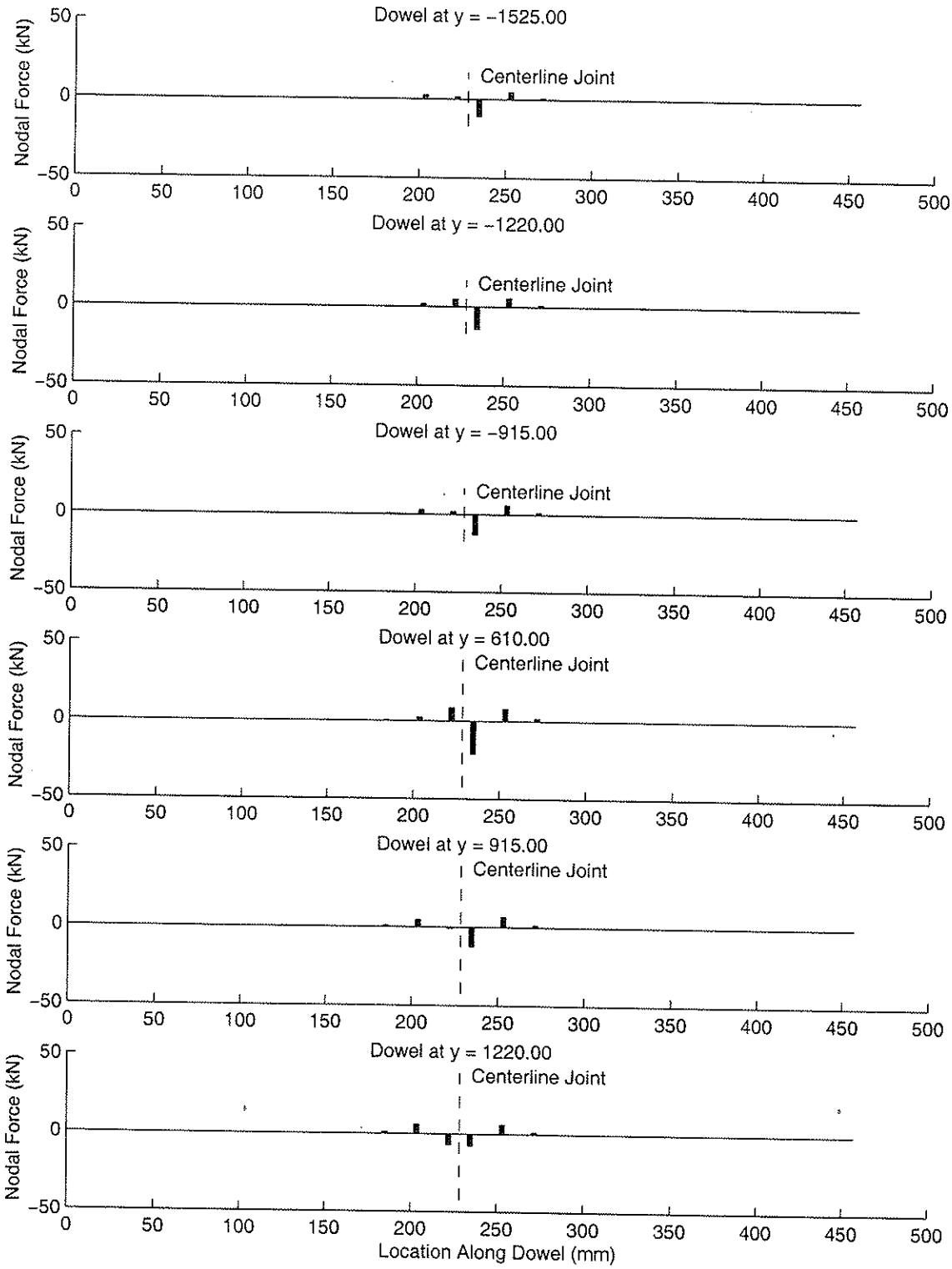


Figure 8.24: Nodal Forces on Dowels for 0 mm Gap – Axle and Temperature Loading

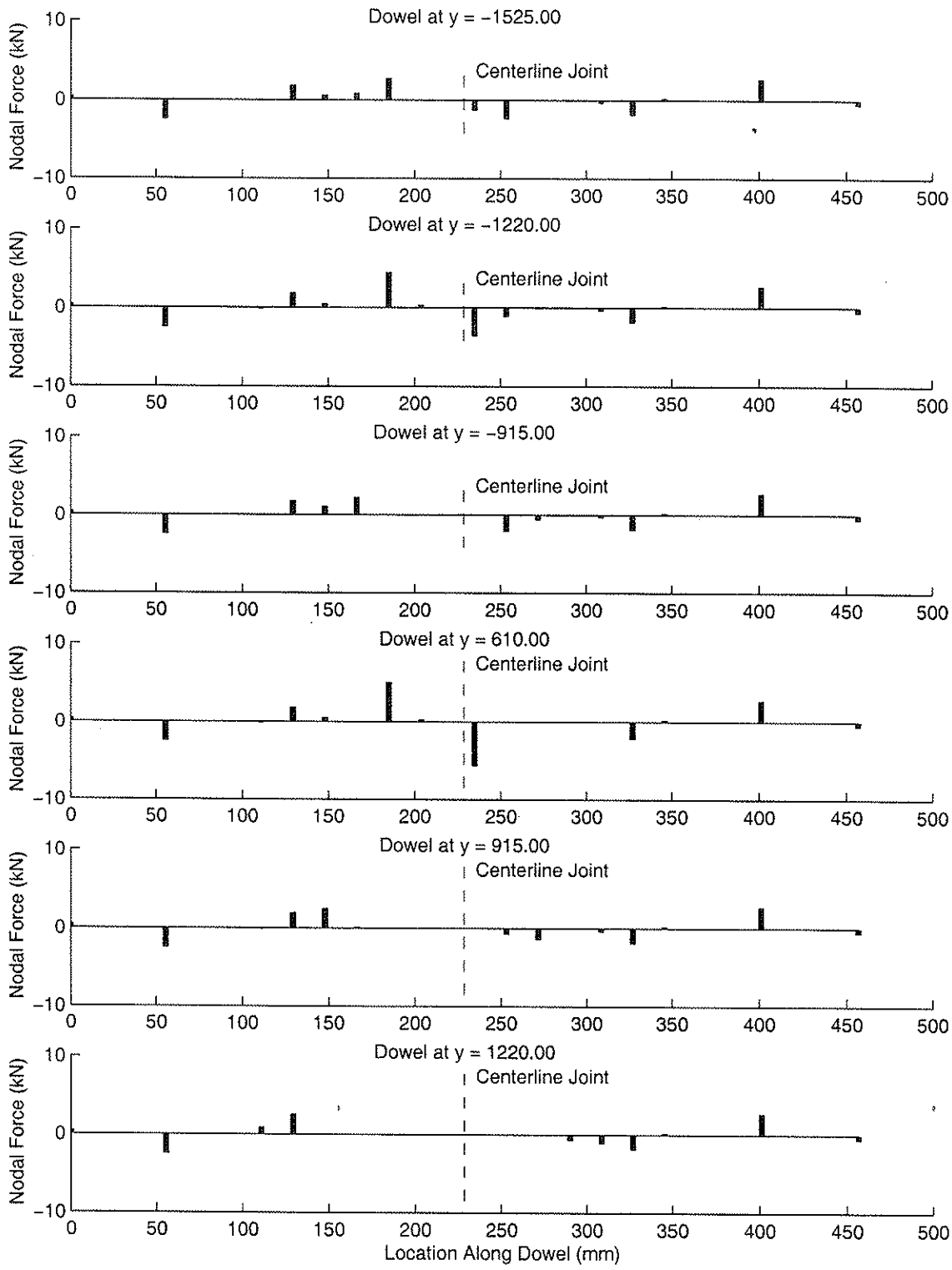


Figure 8.25: Nodal Forces on Dowels for 0.08 mm Gap – Axle and Temperature Loading

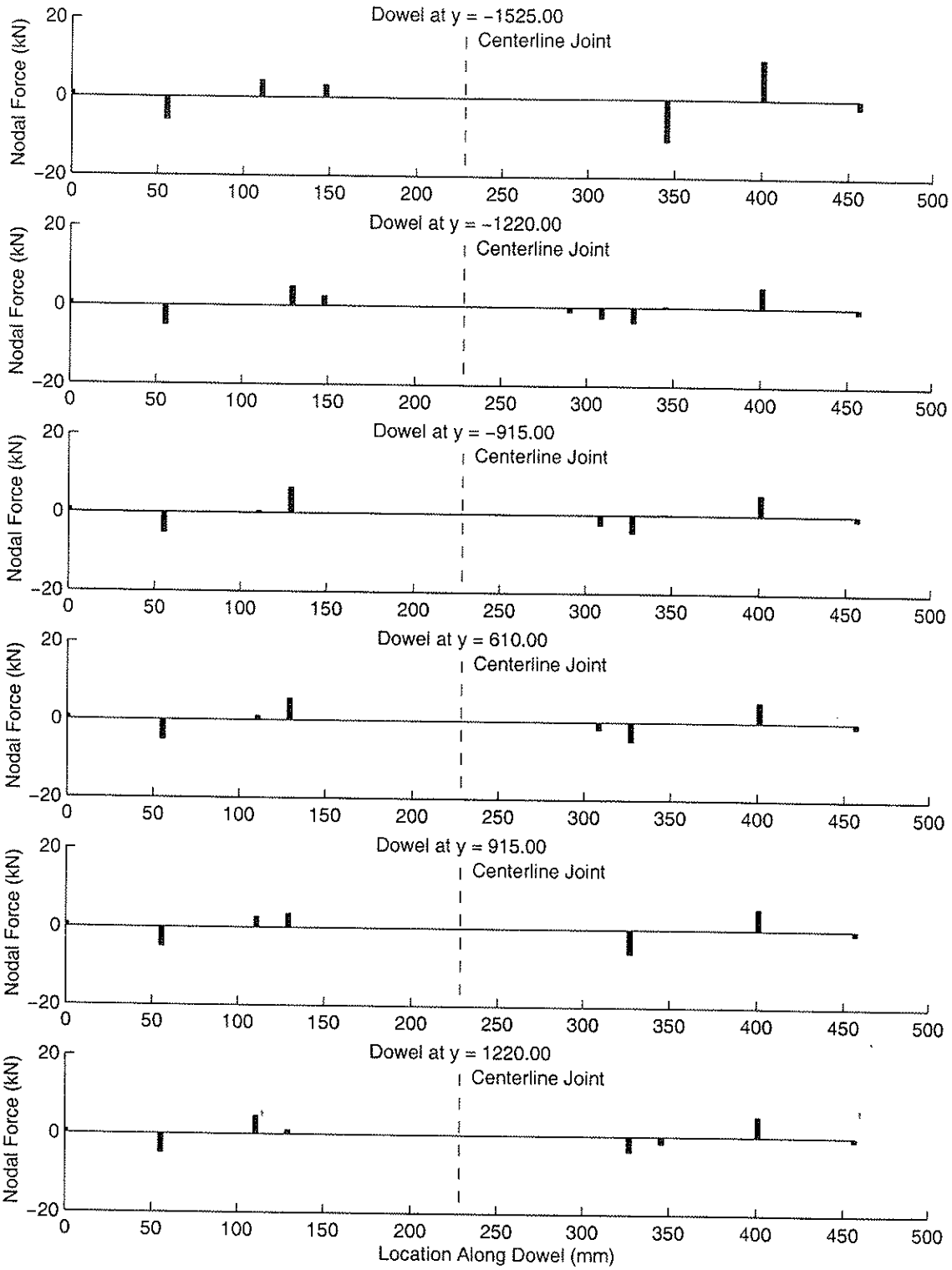


Figure 8.26: Nodal Forces on Dowels for 0.16 mm Gap – Axle and Temperature Loading

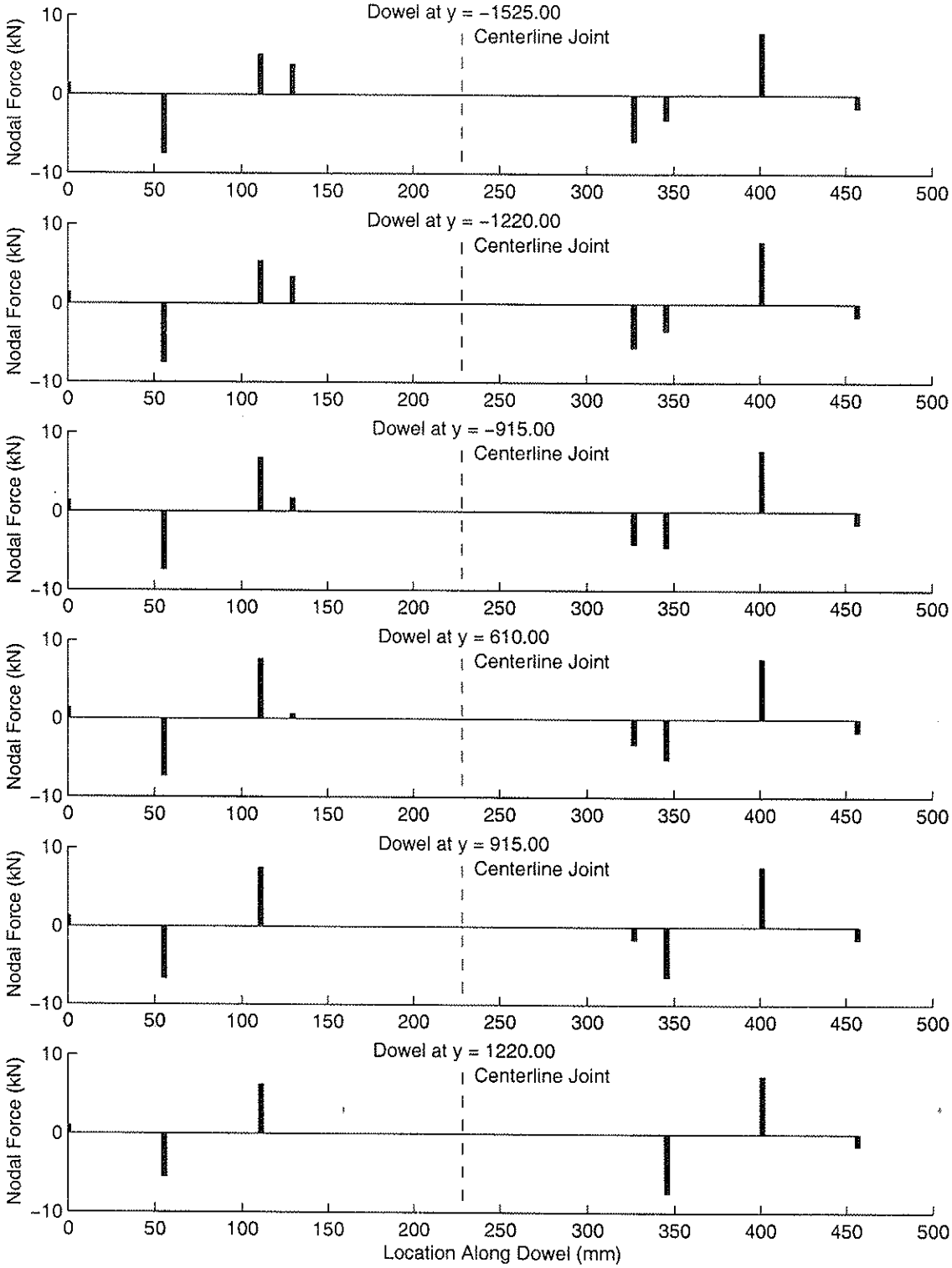


Figure 8.27: Nodal Forces on Dowels for 0.24 mm Gap – Axle and Temperature Loading

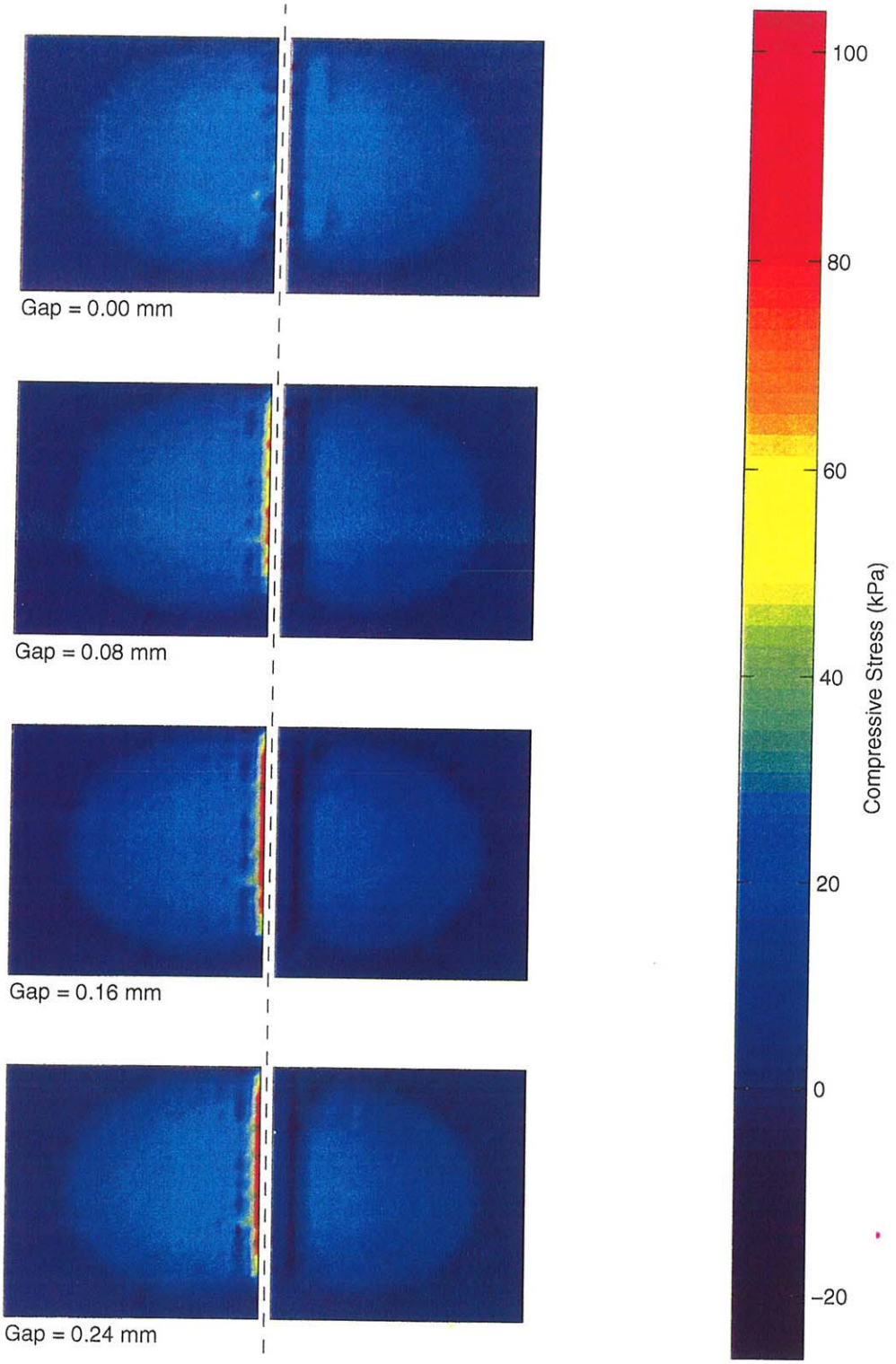


Figure 8.28: Vertical Stress on Top of Base Layer – Doweled Joint, Axle and Temperature Loading

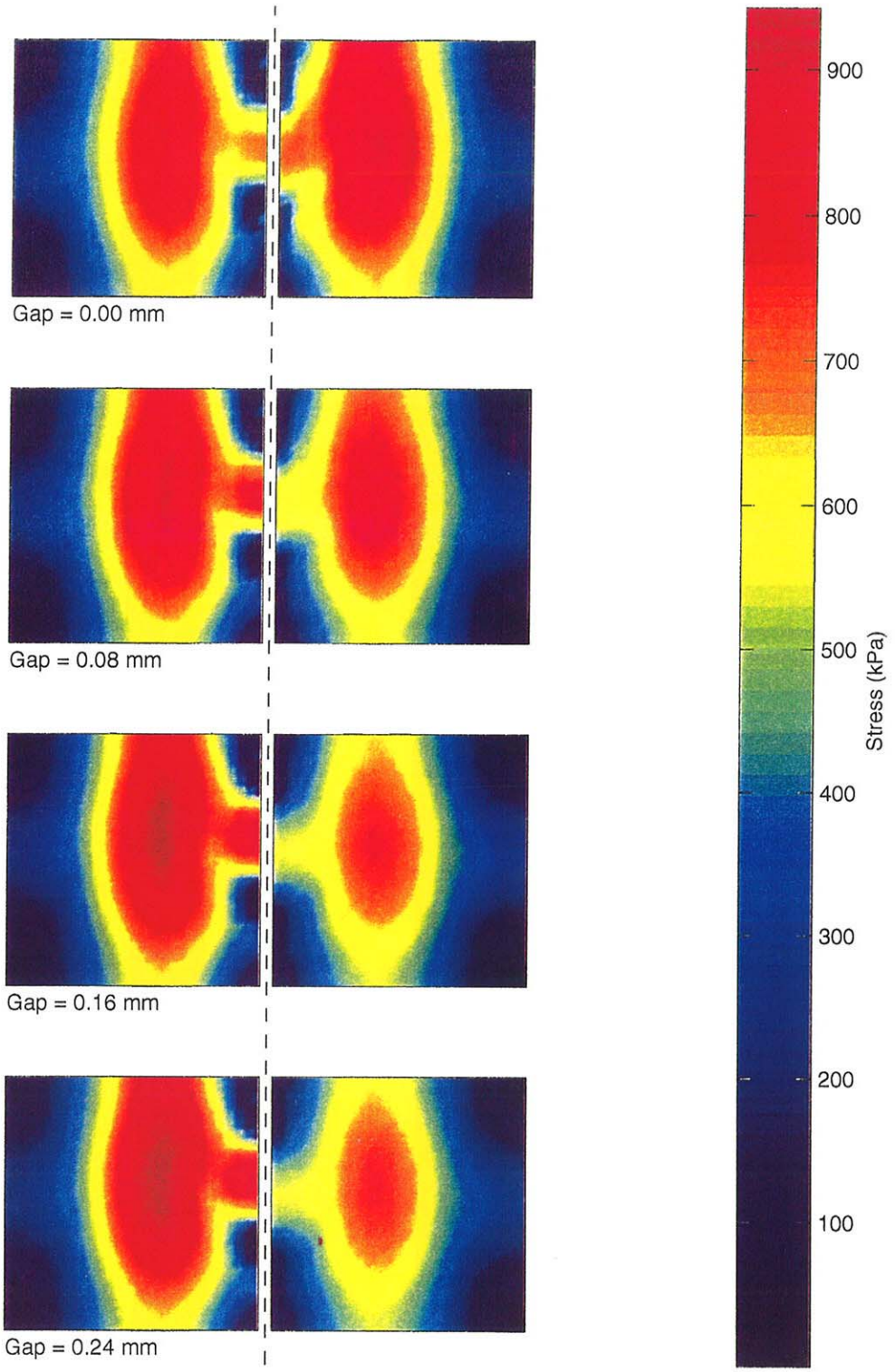


Figure 8.29: Max. Principal Stress on Top of Slab – Doweled Joint, Axle and Temperature Loading

8.6 Summary

This Chapter presented two parametric studies on joint shear transfer effectiveness. The first study examined the effect of dowel looseness on pavement response; the second study focused on the effect of the variation in initial joint opening on aggregate interlock shear transfer effectiveness. Axle loading was considered in both studies, while axle loading combined with a negative temperature gradient was also considered for the doweled model.

It was shown that small values of dowel looseness (gaps between the dowel and the slab of less than 0.12 mm) and typical seasonal variations in initial undoweled joint opening (less than 1.60 mm) can have large effects on pavement response to axle loads near the joint, with losses in joint shear transfer efficiency of up to 50% and significant increases in the slab tensile stresses. Perhaps more important, however, is the large increase in vertical stress applied to the subgrade by the loaded slab near the joint as joint shear transfer effectiveness decreases. This indicates significant potential for subgrade yielding and degradation under repeated loading, which could hasten the development of joint damage and faulting.

Under the action of combined temperature and wheel loading, gaps between the dowels and the slabs resulted in relatively modest increases in tensile stresses on top of the slabs. However, the displacement response of the system varied significantly with differing amounts of joint shear transfer. The models captured these effects quite well, illustrating the variations in slab/subgrade separation and differential joint displacements with dowel looseness.

Based on these studies, it appears that both dowel looseness and aggregate interlock nonlinearities due to commonly experienced joint openings can have large effects on the response of rigid pavement systems. The finite element models used in this study are capable of capturing these variations in response, making them valuable tools for the evaluation of existing and proposed design configurations under a variety of loadings. Several areas of future research can also be identified based on these studies, including the formation of dowel looseness and reduction of aggregate interlock load transfer due to cyclic loading, and the development of better base/subgrade models that capture soil yielding and damage under high localized stresses.

Chapter 9

Conclusions

9.1 Summary

This study was undertaken with the objectives of (1) developing a method for modeling dowel joint load transfer in rigid pavement systems that rigorously incorporates dowel looseness; (2) investigating new methods for the rational modeling of aggregate interlock load transfer; and (3) developing efficient solution strategies for making three dimensional (3D) finite element analysis of rigid pavements practical. A better understanding of joint shear transfer mechanisms was identified as a critical research need due to the high frequency of joint failures and their detrimental effects on pavement serviceability. Achieving the second objective is crucial to making realistic 3D finite element analysis of rigid pavement systems a viable tool for routine mechanistic design.

A comprehensive literature review of past research on the finite element modeling of rigid pavement systems was conducted, including detailed summaries of recent 2D and 3D models. This literature review served to verify that 3D analysis is preferable to 2D analysis using plate elements due to the inherent limitations of 2D analysis. The modeling techniques employed by various researchers were summarized, including slab/subgrade discretization, material models, and methods of modeling dowel and aggregate interlock shear transfer. This summary indicated that previous efforts to model dowel transfer have been somewhat unrealistic; in particular, the presence of dowel looseness has not been accounted for explicitly in prior 3D models, even though it has been demonstrated as having significant detrimental effects on rigid pavement performance [10]. A similar conclusion was drawn regarding aggregate interlock shear transfer, which has typically been modeled using linear springs or frictional contact at contraction joints. Further, there is no evidence of the use of 3D finite element modeling techniques in design settings due in part to computational

requirements.

These specific shortcomings were addressed in the development of modeling techniques and the simulations of this study.

- Multiple-slab systems of various geometries were considered. The slabs were considered as linearly elastic.
- The inclusion of multiple and debonded linearly elastic base layers was addressed.
- The models included accurately located rectangular and circular tire contact areas as well as linear temperature gradients through the slab thickness.
- Dowels were located precisely, and debonding and dowel looseness were considered explicitly.
- A two-phase model for aggregate interlock was implemented, permitting the effects of joint opening and concrete properties to be captured.
- Efficient iterative solution strategies for large-scale systems involving material and contact nonlinearities were developed and thoroughly tested.

To test the accuracy and applicability of the proposed finite element models developed in this study, a comparison of deflection and strain data with measured values from a recent laboratory study was done [33]. In particular, doweled two-slab systems subjected to corner loading were analyzed: the first model rested directly on a rubber pad (idealized as a dense liquid), the second on an unbonded cement treated base above the rubber pad. To capture the relative vertical displacements at the joint observed for both systems, a small gap was assumed to exist around the embedded portion of the dowels in the unloaded slab; this assumption was deemed reasonable based on the methods of construction. Displacement basin comparisons were reasonably accurate with excellent agreement near the joint, verifying dowel looseness as a potential cause of the observed relative joint displacements. Strain comparisons were less accurate, but reasonable agreement between measured and computed values was observed for the first test. Straïn comparisons for the second test were inconclusive, with no obvious explanation.

To illustrate the applicability of the aggregate interlock and dowel modeling techniques, a series of parametric studies of a typical rigid pavement system was conducted. The system studied consisted of two rigid pavement slabs on a gravel base layer; dowel locations were based on a typical retrofit design configuration used by the Washington State Department

of Transportation. For the doweled model, the parameter considered was dowel looseness, while for the undoweled joints where aggregate interlock provides the only means of shear transfer, joint opening was the single parameter. Both axle and combined axle and temperature loadings were considered in these studies. In addition to load transfer efficiency, slab stresses and displacements were examined, as well as vertical stresses applied to the base layer. Variations in dowel shears and dowel contact forces as a function of dowel looseness were illustrated for the doweled joint; for the undoweled model, the variation in aggregate interlock stresses on the joint faces was examined.

9.2 Contributions

The major contributions of this study have been (1) the development of rational finite element methods for the modeling of joint load transfer in rigid pavements due to dowel action and aggregate interlock, and (2) the development of a multigrid preconditioner applicable for unstructured 3D finite element models incorporating contact nonlinearities. The features of these contributions are summarized as follows:

- *Dowel load transfer.* A quadratic embedded bending element was developed that can be meshed independently of the solid elements it is embedded within. This formulation permits the dowels to be located precisely within the finite element model without distorting the mesh of solid elements comprising the slab. In addition to easing the task of mesh generation and minimizing errors due to mesh distortion, this is a significant advantage when using multigrid solution techniques which require meshes of the same domain having varying levels of refinement. The formulation also permits debonding of the dowel on a node-by-node basis, as well as the modeling of gaps between the dowel and surrounding slab. Contact nonlinearities arising from dowel looseness are captured in the transformation of the element stiffness matrix, making the element amenable to direct inclusion in a nonlinear solution scheme without the use of geometric constraints. The embedded bending member should prove useful for modeling reinforcing in many concrete structures; towards this end, the inclusion of a general bond-slip law was discussed.

- *Aggregate interlock shear transfer.* A rational method for the modeling of aggregate interlock shear transfer was developed. It relies on a two-phase model for the concrete at the joint as initially developed by Walraven [91, 92]. The constitutive relations predicted by this model capture the nonlinear variation in shear stress transfer

with normal and tangential joint displacements. Particularly significant is the large variation in shear transfer with joint opening, which may undergo significant seasonal changes. The aggregate interlock model was implemented using a sixteen-noded quadratic interface element that displaces compatibly with the solid elements it is meshed between.

- *Solution strategies.* A solution strategy appropriate for the analysis of large-scale structural systems incorporating spatially varying or nonlinear materials and contact nonlinearities was developed. A significant feature of this solution strategy is a multigrid preconditioner that permits the use of unstructured/unnested mesh sequences. General restriction and interpolation operators that rely on the evaluation of element shape functions were developed that permit multiple isoparametric element types and varying nodal degrees of freedom. These multigrid methods permit the coarse mesh stiffness matrices to be assembled in the usual fashion even in the presence of spatially varying materials with sharp boundaries, greatly simplifying implementation.

9.3 Future Work

While this study has made significant contributions to the finite element modeling of rigid pavement systems, there are several areas that would benefit from further research.

9.3.1 Formation of Dowel Looseness

The ability to predict the formation of gaps between the dowels and surrounding slab would be a particularly important topic of future research. One prior experimental study has shown this to occur under repeated load cycles [10], but at present a predictive analytical model does not exist. Prediction of long-term joint response and a better understanding of the associated mechanics of dowel-slab interaction would aid new design efforts as well as assist in the development of retrofit strategies and associated life-cycle cost analyses. This would require the development of constitutive relations that relate cumulative damage to repeated application of bearing stresses between the dowel and concrete and local finite element models incorporating these constitutive models in the regions around the dowels.

Once a viable and realistic dowel looseness prediction model has been developed, back-calculation of in-situ looseness of doweled joints might be possible. Ideally, such an evaluation would rely on standard non-destructive tests. Once the in-situ properties have been determined, the damage model could be used to predict remaining joint life.

9.3.2 Aggregate Interlock Shear Transfer – Cumulative Damage

The method for predicting aggregate interlock shear transfer developed in this study is strictly applicable only to newly constructed joints. In reality, repeated axle loading results in damage to the joint, and the effectiveness of aggregate interlock shear transfer degrades significantly; this is the subject of at least one recent and ongoing study [3]. The two-phase model of Walraven can predict cumulative damage under high stresses and low load cycles. Refinement of this model to allow the accurate prediction of low-stress, high cycle damage would be a first step in the development of an aggregate interlock cumulative damage model. Such a research effort would likely rely on relevant experimental data.

Although few pavement joints are presently constructed without additional load transfer devices (such as dowels), many existing pavements rely entirely on aggregate interlock for joint load transfer. As with the dowel looseness model, an aggregate interlock cumulative damage model would be particularly useful in conjunction with back-calculation of in-situ joint properties. Development of cost-effective long-term retrofit strategies for these pavements would benefit greatly from the ability to estimate current performance and predict the remaining life of the joint.

9.3.3 Nonlinear Base/Subgrade Response and Damage

Closely related to the first two areas of future research discussed here are nonlinear subgrade response and damage. The parametric studies of Chapter 8 highlighted the increased vertical stresses applied to the base layer due to dowel looseness and poor aggregate interlock shear transfer arising from large joint openings. These stress increases might lead to localized yielding and plastic flow of the base/subgrade material, and increase the potential for mechanical fatigue and pumping-related deterioration. Cumulative damage and the formation of dowel looseness and wear on the joint surfaces will likely be exacerbated by soil yielding and softening, making its modeling critical to any finite element-based damage predictions.

Bibliography

[1] *The AASHO Road Test - Pavement Research.* Special Report 61E, Highway Research Board, 1962.

[2] AASHTO. *Guide for Design of Pavement Structures*, 1988.

[3] M.G. Abdel-maksoud, N. Hawkins, and E. Barenberg. Behavior of Concrete Joints under Cyclic Shear. In *Aircraft/Pavement Technology: In the Midst of Change*, pages 190–204. ASCE, 1997.

[4] S.F. Ashby and R.D. Falgout. A Parallel Multigrid Preconditioned Conjugate Gradient Algorithm for Groundwater Flow Simulations. *Nuclear Science and Engineering*, 124:145–159, 1996.

[5] F. Barzegar and S. Maddipudi. Generating Reinforcement in FE Modeling of Concrete Structures. *Journal of Structural Engineering*, 120(5):1656–1662, 1994.

[6] K.-J. Bathe. *Finite Element Procedures.* Prentice-Hall, Inc., 1996.

[7] Z.P. Bažant and P. Gambarova. Rough Cracks in Reinforced Concrete. *Journal of the Structural Division, ASCE*, 106(ST4):819–842, 1980.

[8] A. Brandt. Multi-Level Adaptive Solutions to Boundary-Value Problems. *Mathematics of Computation*, 31(138):333–390, 1977.

[9] D.R. Brill, G.F. Hayhoe, and X. Lee. Three-Dimensional Finite Element Modeling of Rigid Pavement Structures. In *Aircraft/Pavement Technology: In the Midst of Change*, pages 151–165. ASCE, 1997.

[10] Neeraj Buch and Dan G. Zollinger. Development of Dowel Looseness Prediction Model for Jointed Concrete Pavements. In *Transportation Research Record 1525*, pages 21–27, Washington, D.C., 1996. TRB, National Research Council.

[11] O. Buyukozturk, J.J. Connor, and P. Leombruni. Research on Modeling Shear Transfer in Reinforced Concrete Nuclear Structures. *Nuclear Engineering and Design*, 59:67–83, 1980.

[12] C. Channakeshava, F. Barzegar, and G. Voyiadjis. Nonlinear FE Analysis of Plain Concrete Pavement with Doweled Joints. *Journal of Transportation Engineering*, 119(5):763–781, 1993.

[13] K. Chatti, J. Lysmer, and C.L. Monismith. Dynamic Finite-Element Analysis of Jointed Concrete Pavements. *Transportation Research Record*, 1449:79–90, 1994.

[14] L.D. Childs. Tests of Concrete Pavement Slabs on Cement-Treated Subbases. In *Highway Research Record 60*, Washington, D.C., 1964. HRB, National Research Council.

[15] B.E. Colley and H.M. Humphrey. Aggregate Interlock at Joints in Concrete Pavements. In *Bulletin 189*, pages 1–18, Washington, D.C., 1967. HRB, National Research Council.

[16] M. I. Darter, K. T. Hall, and C.-M. Kuo. NCHRP Report No. 372: Support Under Portland Cement Concrete Pavements. Washington, D.C., 1995. TRB, National Research Council.

[17] James W. Demmel, John R. Gilbert, and Xiaoye S. Li. SuperLU Users' Guide. Technical report, Computer Science Division, University of California, Berkeley, Feb. 1997.

[18] M.P. Divakar and A. Fafitas. Micromechanics-Based Constitutive Model for Interface Shear. *Journal of Engineering Mechanics*, 118(7):1317–1337, 1992.

[19] M.P. Divakar, A. Fafitas, and S.P. Shah. Constitutive Model for Shear Transfer in Cracked Concrete. *Journal of Structural Engineering*, 113(5):1046–1062, 1987.

[20] M.C. Dracopoulos and M.A. Crisfield. Coarse/Fine Mesh Preconditioners for the Iterative Solution of Finite Element Problems. *International Journal for Numerical Methods in Engineering*, 38:3297–3313, 1995.

[21] A. Elwi and T. Hrudi. Finite Element Model for Curved Embedded Reinforcement. *Journal of Engineering Mechanics*, 115(4):740–754, 1988.

[22] A. Fafitas and M.P. Divakar. A Constitutive Model for Aggregate Interlock Shear Based on Micromechanics. *Journal of Energy Resources Technology*, 114(9):235–243, 1992.

[23] M.N. Fardis and O. Buyukozturk. Shear Transfer Model for Reinforced Concrete. *Journal of Engineering Mechanics*, 105(EM2):255–275, 1979.

[24] C. Farhat and N. Sobh. A Coarse/Fine Mesh Preconditioner for Very Ill-Conditioned Finite Element Problems. *International Journal for Numerical Methods in Engineering*, 28:1715–1723, 1989.

[25] P.H. Feenstra, R. Boorst, and J.G. Rots. Numerical Study on Crack Dilatancy. I: Models and Stability Analysis. *Journal of Engineering Mechanics*, 117(4):733–753, 1991.

[26] P.H. Feenstra, R. Boorst, and J.G. Rots. Numerical Study on Crack Dilatancy. II: Applications. *Journal of Engineering Mechanics*, 117(4):754–769, 1991.

[27] J. Fish, L. Pan, V. Belsky, and S. Gomaa. Unstructured Multigrid Method for Shells. *International Journal for Numerical Methods in Engineering*, 39:1181–1197, 1996.

[28] B.F. Friberg. Design of Dowels in Transverse Joints of Concrete Pavements. *Proceedings of the ASCE*, 64(9), 1938.

[29] G.H. Golub and C.F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, 1989.

[30] J. E. Goodman and J. O'Rourke. *Handbook of Discrete and Computational Geometry.* CRC, 1997.

[31] H. Guo, J.A. Sherwood, and M.B. Snyder. Component Dowel-Bar Model for Load Transfer Systems in PCC Pavements. *Journal of Transportation Engineering*, 121(3):289–298, 1995.

[32] K. T. Hall, M. I. Darter, and J. M. Armaghani. Performance Modeling of Joint Load Transfer Restoration. In *Transportation Research Record 1388*, pages 129–139, Washington, D.C., 1993. TRB, National Research Council.

[33] Michael I. Hammons. Development of an Analysis System for Discontinuities in Rigid Airfield Pavements. Technical Report GL-97-3, US Army Corps of Engineers, Apr. 1997.

[34] K.D. Hjelmstat, J. Kim, and Q.H. Zuo. Finite Element Procedures for Three-Dimensional Pavement Analysis. In *Aircraft/Pavement Technology: In the Midst of Change*, pages 125–137. ASCE, 1997.

[35] K.D. Hjelmstat and E. Taciroglu. A Coupled Hyperelastic Constitutive Model for Resilient Response of Granular Materials. In *Aircraft/Pavement Technology: In the Midst of Change*, pages 178–189. ASCE, 1997.

[36] J.-M. Hohberg. Concrete Joints. In A.P.S. Selvadurai and M.J. Boulon, editors, *Studies in Applied Mechanics 42 - Mechanics of Geomaterial Interfaces*, pages 421–446. Elsevier, 1995.

[37] Yang H. Huang. Finite Element Analysis of Slabs on Elastic Solids. *Transportation Engineering Journal, ASCE*, 100(TE2):403–415, 1974.

[38] Yang H. Huang and Xue-Jun Deng. Finite Element Analysis of Jointed Concrete Pavements. *Journal of Transportation Engineering*, 109(5):689–705, 1983.

[39] Y.H. Huang. *Pavement Analysis and Design.* Prentice-Hall, Inc., 1993.

[40] T.J.R. Hughes. *The Finite Element Method.* Prentice-Hall, Inc., 1987.

[41] A. M. Ioannides and G. T. Korovesis. Analysis and Design of Doweled Slab-on-Grade Pavement Systems. *Journal of Transportation Engineering*, 118(6):745–769, 1992.

[42] A.M. Ioannides, E.J. Barenberg, and M.R. Thompson. Finite Element Model with Stress Dependent Support. In *Transportation Research Record 954*, pages 10–16, Washington, D.C., 1984. TRB, National Research Council.

[43] A.M. Ioannides and J.P Donelly. Three-Dimensional Analysis of Slab on Stress-Dependent Foundation. In *Transportation Research Record 1196*, pages 72–84, Washington, D.C., 1988. TRB, National Research Council.

[44] A.M. Ioannides and G.T. Korovesis. Aggregate Interlock: A Pure-Shear Load Transfer Mechanism. In *Transportation Research Record 1286*, pages 14–24, Washington, D.C., 1990. TRB, National Research Council.

[45] A.M. Ioannides and G.T. Korovesis. Analysis and Design of Doweled Slab-on-Grade Pavement Systems. *Journal of Transportation Engineering*, 118(6):745–768, 1992.

[46] A.M. Ioannides, Y.-H. Lee, and M.I. Darter. Control of Faulting Through Joint Load Transfer Design. In *Transportation Research Record 1286*, pages 49–56, Washington, D.C., 1992. TRB, National Research Council.

[47] D.C. Jesperson. Multigrid Methods for Partial Differential Equations. In G.H. Golub, editor, *MAA Studies in Mathematics*, volume 24, pages 270–317, 1984.

[48] Jong keun Choi and Jang keun Lim. General Curved Beam Elements Based on the Assumed Strain Fields. *Computers and Structures*, 55(3):379–386, 1995.

[49] J. Kim, K.D. Hjelmstat, and Q.H. Zuo. Three-Dimensional Finite Element Study of Wheel Load Interaction. In *Aircraft/Pavement Technology: In the Midst of Change*, pages 138–150. ASCE, 1997.

[50] M. Kŏcvara. An Adaptive Multigrid Technique for Three-Dimensional Elasticity. *International Journal for Numerical Methods in Engineering*, 36:1703–1716, 1993.

[51] T. Krauthammer, N. Bazeos, and T.J. Holmquist. Modified SDOF Analysis of RC Box-type Structures. *Journal of Structural Engineering*, 112(4):726–744, 1986.

[52] T. Krauthammer and K.L. Western. Joint Shear Transfer Effects on Pavement Behavior. *Journal of Transportation Engineering*, 114(5):505–529, 1988.

[53] A.R. Kukreti, M.R. Taheri, and Ragnar H. Ledesma. Dynamic Analysis of Rigid Airport Pavements with Discontinuities. *Journal of Transporation Engineering*, 118(3):341–360, 1992.

[54] C.-M. Kuo, K.T. Hall, and M.I. Darter. Three-Dimensional Finite Element Model for Analysis of Concrete Pavement Support. In *Transportation Research Record 1505*, pages 119–127, Washington, D.C., 1996. TRB, National Research Council.

[55] J.P. Laible, R.N. White, and P. Gergely. Experimental Investigation of Seismic Shear Transfer Across Cracks in Concrete Nuclear Containment Structures. In *Reinforced Concrete Structures in Seismic Zones*, ACI Special Publication SP-53, pages 203–226, Detroit, Mi., 1977. American Concrete Institute.

[56] B. Li, K. Maekawa, and H. Okamura. Contact Density Model for Stress Transfer across Cracks in Concrete. *Journal of the Faculty of Engineering, the University of Tokyo*, 40(1):9–52, 1989.

[57] Xiaoye S. Li. Sparse Gaussian Elimination on High Performance Computers. Technical report, Computer Science Division, University of California, Berkeley, Sep. 1996.

[58] R. Löhner and K. Morgan. An Unstructured Multigrid Method for Elliptic Problems. *International Journal for Numerical Methods in Engineering*, 24:101–115, 1987.

[59] Joe P. Mahoney, Brian C. Winters, Karim Chatti, Thomas J. Moran, Carl L. Monismith, and Steven L. Kramer. Vehicle Pavement Interaction at the Paccar Test Site. Technical Report GC87195, Task 42, Washington State Transportation Commission, 1995.

[60] E. Masad, R. Taha, and B. Muhunthan. Finite-Element Analysis of Temperature Effects on Plain-Jointed Concrete Pavements. *Journal of Transporation Engineering*, 122(5):388–398, 1997.

[61] K.H. McGhee. NCHRP Synthesis No. 211: Design, Construction, and Maintenance of PCC Pavement Joints. Washington, D.C., 1995. TRB, National Research Council.

[62] F.T. McKenna. *Object-Oriented Finite Element Programming: Frameworks for Analysis, Algorithms, and Parallel Computing.* PhD thesis, The University of California, Berkeley, 1997.

[63] J.C. Meza and R.S. Tuminaro. A Multigrid Preconditioner for the Semiconductor Equations. *SIAM Journal on Scientific Computing*, 17(1):118–132, 1996.

[64] C.M. Kuo M.I. Darter, K.T. Hall. NCHRP Report No. 372: Support under Portland Cement Concrete Pavements. Technical report, TRB, National Research Council, Washington, D.C., 1995.

[65] S.G. Millard and R.P. Johnson. Shear Transfer across Crack in Reinforced Concrete due to Aggregate Interlock and to Dowel Action. *Magazine of Concrete Research*, 36(126):9–21, 1984.

[66] S.A. Mitchell and S. Vavasis. Quality Mesh Generation in Three Dimensions. In *Proceedings of the ACM Computational Geometry Conference*, pages 212–221, 1992.

[67] F. Muttin and J.-L. Chenot. On a Conjugate-Gradient Two-Grid Method for Three-Dimensional Elasticity. *Engineering Computations*, 12:3–20, 1995.

[68] C.W. Oosterlee and Y. Washio. An Evaluation of Parallel Multigrid as a Solver and A Preconditioner for Singularly Perturbed Equations. *SIAM Journal on Scientific Computing*, 19(1):87–110, 1998.

[69] I.D. Parsons, I.-S. Eom, and K.D. Hjelmstad. Numerical Simulations of Load Transfer Between Doweled Pavement Slabs. In *Aircraft/Pavement Technology: In the Midst of Change*, pages 166–177. ASCE, 1997.

[70] I.D. Parsons and J.F. Hall. The Multigrid Method in Solid Mechanics: Part I - Algorithm Description and Behavior. *International Journal for Numerical Methods in Engineering*, 29:719–737, 1990.

[71] I.D. Parsons and J.F. Hall. The Multigrid Method in Solid Mechanics: Part II - Practical Applications. *International Journal for Numerical Methods in Engineering*, 29:739–753, 1990.

[72] G. Prathap and G.R. Byashyam. Reduced Integration and the Shear-Flexible Beam Element. *International Journal for Numerical Methods in Engineering*, 18:195–210, 1982.

[73] J.S. Przemieniecki. *Theory of Matrix Structural Analysis.* McGraw-Hill, Inc., 1968.

[74] H.W. Reinhardt and J.C. Walraven. Cracks in Concrete Subject to Shear. *Journal of the Structural Division, ASCE*, 108(ST1):207–224, 1982.

[75] W.J. Rider and D.A. Knoll. Solving Nonlinear Heat Conduction Problems with Multigrid Preconditioned Newton-Krylov Methods. Technical Report LA-UR-97-2929, Los Alamos National Lab., Feb. 1997.

[76] M.D. Rucki. *An Algorithmic Framework for Flexible Finite Element Modelling.* PhD thesis, The University of Washington, May 1996.

[77] Y. Saad. *Iterative Methods for Sparse Linear Systems.* PWS Publishing Co., Boston, MA, 1996.

[78] G. Sathurappan, N. Rajagopalan, and C.S. Krishnamoorthy. Nonlinear Finite Element Analysis of Reinforced and Prestressed Concrete Slabs with Reinforcement (Inclusive of Prestressing Steel) Modelled as Discrete Integral Components. *Computers and Structures*, 44(3):575–584, 1992.

[79] C. Sittipunt and K.L. Wood. Finite Element Analysis of Reinforced Concrete Shear Walls. Technical report, Department of Civil Engineering, University of Illinois, Dec 1993.

[80] S.K.Wang, M.Sargious, and Yau K. Cheung. Advanced Analysis of Rigid Pavements. *Transportation Engineering Journal, ASCE*, 98(TE1):37–44, 1972.

[81] M. B. Snyder. Cyclic Shear Load Testing of Dowels in PCC Pavement Repairs. In *Transportation Research Record 1215*, pages 246–257, Washington, D.C., 1989. TRB, National Research Council.

[82] A.M. Tabatabaie and E.J. Barenberg. Finite-Element Analysis of Jointed or Cracked Concrete Pavements. In *Transportation Research Record 671*, pages 11–19, Washington, D.C., 1978. TRB, National Research Council.

[83] A.M. Tabatabaie and E.J. Barenberg. Finite-Element Analysis of Jointed or Cracked Concrete Pavements. In *Transportation Research Record 671*, pages 11–19, Washington, D.C., 1978. TRB, National Research Council.

[84] A.M. Tabatabaie and E.J. Barenberg. Structural Analysis of Concrete Pavements. *Transportation Engineering Journal, ASCE*, 106(TE5):832–849, 1980.

[85] T.P. Tassios and E.N. Vintezeleou. Concrete-to-Concrete Friction. *Journal of Structural Engineering*, 113(4):832–849, 1987.

[86] S.D. Tayabji and B.E. Colley. Analysis of Jointed Concrete Pavements. Technical Report FHWS-RD-86-041, FHWA, 1986.

[87] L.W. Teller and E.C. Sutherland. The Structural Design of Concrete Pavements. *Public Roads*, 16:145–158, 169–197, 201–221, 1935.

[88] L.W. Teller and E.C. Sutherland. The Structural Design of Concrete Pavements. *Public Roads*, 17:143–171, 175–192, 1936.

[89] L.W. Teller and E.C. Sutherland. The Structural Design of Concrete Pavements. *Public Roads*, 23:167–211, 1936.

[90] W. Uddin, R.M. Hackett, A. Joseph, Z. Pan, and A.B. Crowley. Three-Dimensional Finite-Element Analysis of Jointed Concrete Pavement Having Discontinuities. In

Transportation Research Record 1482, pages 26–32, Washington, D.C., 1996. TRB, National Research Council.

[91] J.C. Walraven. Fundamental Analysis of Aggregate Interlock. *Journal of the Structural Division, ASCE*, 107(ST11):2245–2270, 1981.

[92] J.C. Walraven. Rough Cracks Subjected to Earthquake Loading. *Journal of Structural Engineering*, 120(5):1510–1524, 1994.

[93] Washington State DOT. *WSDOT Pavement Guide for Design, Evaluation and Rehabilitation*, 1993.

[94] Personal communication with Joe P. Mahoney, Professor, University of Washington Deparment of Civil Engineering, Dec., 1997.

[95] E. Yamaguchi and T. Ohta. Accurate and Efficient Method of Analysis for Reinforced Concrete Structures. *Journal of Structural Engineering*, 119(7):2017–2035, 1992.

[96] S.M. Zaghloul, T.D. White, and T. Kuczek. Evaluation of Heavy Load Damage Effect on Concrete Pavements Using Three-Dimensional, Nonlinear Dynamic Analysis. In *Transportation Research Record 1449*, pages 123–133, Washington, D.C., 1994. TRB, National Research Council.

[97] M. Zaman and A. Alvappillai. Contact-Element Model for Dynamic Analysis of Jointed Concrete Pavements. *Journal of Transporation Engineering*, 121(5):425–433, 1995.

[98] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method - Volume I.* McGraw-Hill Book Company, 1994.

[99] O.C. Zienkiewicz, J.P. Vilotte, S. Toyoshima, and S. Nakazawa. Iterative Methods for Constrained and Mixed Approximation. An Inexpensive Improvement of FEM Performance. *Comp. Meth. Appl. Mech. Eng.*, 51:3–29, 1985.

Appendix A

Object-Oriented Implementation

A.1 Introduction

The immediate use of the finite element program developed for this study is the modeling of rigid pavement systems; however, the program has been written to allow for general 3D displacement-based finite element analysis. The impetus for developing general code is to provide for future extensions, namely the inclusion of different material models, additions to the element library, and the use of other solution routines. With the ability to solve general problems and easy extensibility, the program may be expected to be a valuable research tool for future studies.

To achieve this goal of easy extensibility, the finite element package was developed using the object-oriented C^{++} programming language. The following Section presents a general overview of the program architecture, including descriptions of major classes and their interdependence. Section A.3 is dedicated to the implementation of the embedded dowel element developed in Chapter 4. Details of the classes developed for implementation of the multigrid methods of Chapter 6 are covered in Section A.4, and modeling of the geometric constraints required to capture frictionless nodal contact are detailed in Section A.5.

A.2 Fundamental Classes and Architecture

A.2.1 Overview

The goals of object-oriented program design are to mirror the natural data required to represent a model and implement necessary algorithms through the use of classes that minimize repeated code, facilitate object interaction, and allow for easy future extensibility. Research into the development of object-oriented finite element analysis programs has centered around these objectives; recent reviews of such research may be found in [62] and [76].

McKenna lists four broad categories of classes that have been used in most object-oriented finite element programs [62]:

1. Modeling classes for generation of the finite element model.

2. Finite element model classes used to describe the model and store analysis results. Classes in this category include representations for nodes, elements, boundary conditions, and loads.

3. Numerical classes that handle computational operations, generally basic matrix, vector and tensor operations.

4. Analysis classes, used to form and solve the governing equations. Generally, analysis classes interact with the finite element model classes to assemble the system stiffness matrix and load vectors, compute the residual force vector, etc.

Model generation has not been considered in this study, and model geometry and properties are defined in separately developed input files. Categories (2) - (4) are detailed in the remainder of this Section. With this framework in hand, the object-oriented implementations of the more complex computational aspects of the present study (i.e. the embedded dowel element, multigrid solution strategies, etc.) will be discussed in later sections.

A.2.2 Finite Element Model Classes

The finite element model classes may be grouped into three major categories:

1. Model geometry, including nodal locations, element connectivities, and zero-displacement boundary conditions.

2. Element definitions, which include both material properties and computational aspects of the element discretization.

3. Loads.

The objects representing these three categories are depicted in Fig. A.1.

Model Geometry

Model geometry is represented by the *Mesh* class, which primarily provides storage for and access to lower level objects. Included in the *Mesh* class are dynamically-sized arrays of *Node* objects used to represent nodes in the finite element model. Each *Node* stores an

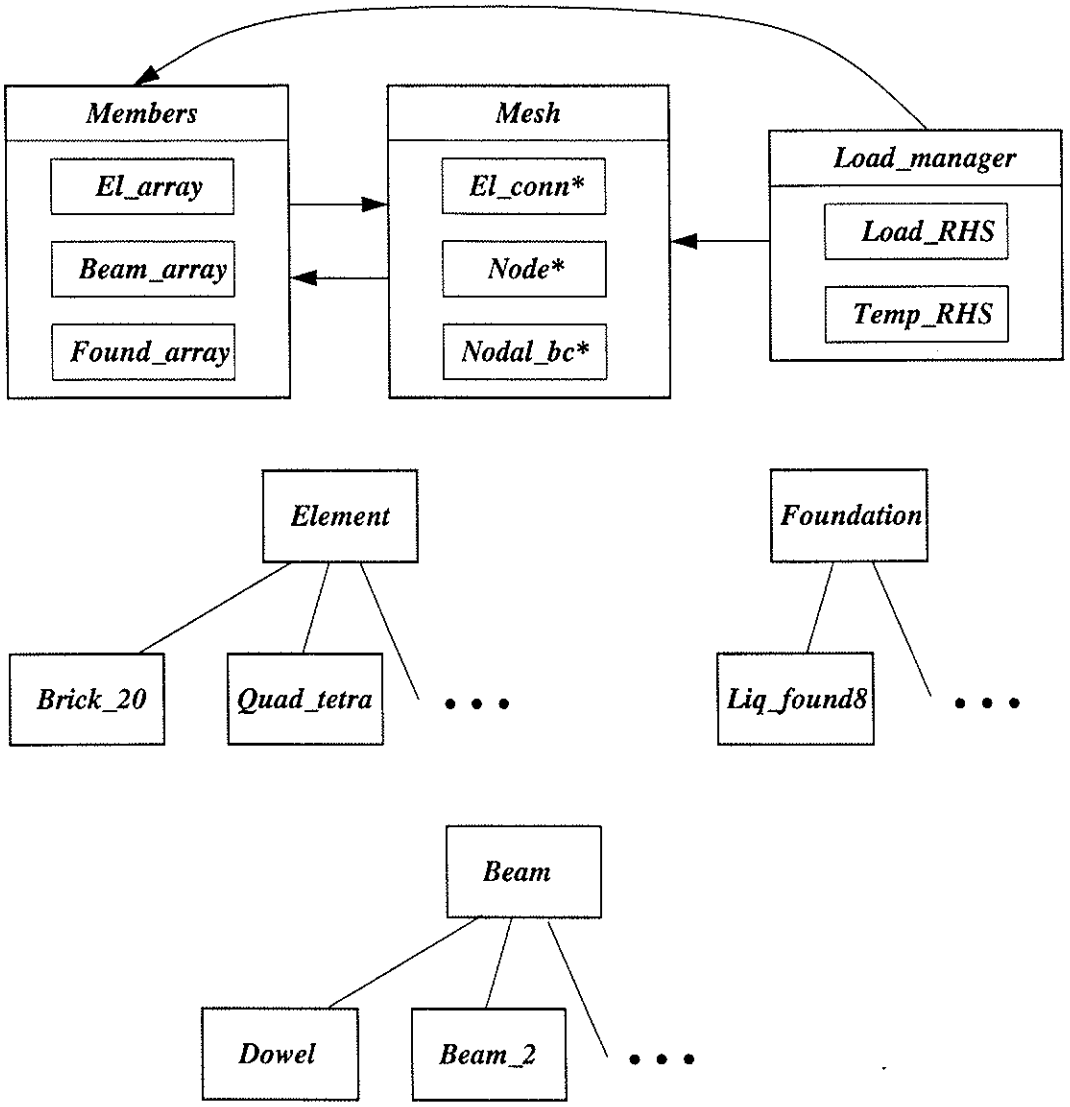


Figure A.1: Finite Element Model Classes

(x, y, z) location, the elements of which may be modified and retrieved via public member functions. The *Mesh* class also contains a dynamically-sized array of *El_conn* objects used to store element connectivities; each *El_conn* contains as member data an integer array of node numbers which may be individually accessed and modified much as elements of a vector. Nodal boundary conditions (free/fixed displacement specifications) are stored by *Nodal_bc* objects, one of which exists for every *Node*. Aside from storing and providing access to the model geometry and boundary conditions, the *Mesh* class also stores the current system displacement vector. The main computational public member function contained in the *Mesh* class is that used to form the graph of the system stiffness matrix prior to its assembly.

Element Representation

Three natural divisions were made with regard to element type, and corresponding virtual base classes were developed from which specific elements can be derived:

- The *Element* base class was designed to represent displacement-based solid and planar (in 2D) continuum elements. Both the 20-noded brick element (*Brick_20*) and 10-noded tetrahedral element (*Quad_tetra* – limited implementation) are derived from this class.

- The *Beam* class represents line elements; the 3-noded *Dowel* element and 2-noded *Beam_2* element are derived from this class.

- The *Foundation* class was developed to implement the Winkler foundation model; at present, only the quadratic *Liq_found8* element has been implemented.

This element hierarchy is illustrated in Fig. A.1. In general, each element contains public member functions for the formation and retrieval of element stiffness matrices and computation of nodal stresses and/or member forces which are required by the solution routines. Each element is also responsible for storing its nodal degrees of freedom, used in mesh traversal and mapping element vectors/matrices to system vectors/matrices. In addition, the *Element*, *Beam*, and *Foundation* classes have special functions specifically required for the analysis methods and solution strategies developed in this study which are detailed later in this Appendix. Each element contains a pointer to a material property object created independently at the time of model generation which defines appropriate constitutive relations. A single material property object may be shared by elements of the same material.

The *El_array*, *Beam_array*, and *Found_array* objects shown in Fig. A.1 are storage classes for each of the three basic element types. They permit dynamically-sized arrays of each base class to be used for element storage and integer-indexed element access. These three objects are created by and contained within a single *Members* object which provides a top-level interface through which other objects may interact with individual elements.

Loading Methods

Figure A.1 depicts the major class objects used to represent various loadings. Note that the *Load_Manager* class is largely a storage class through which the analysis classes retrieve specified load vectors, which may be any combination of equivalent nodal and temperature loads.

The *Load_RHS* class is responsible for handling externally applied tractions, which may be of several types:

- Patch loads applied over arbitrarily located rectangular or circular areas as specified in Chapter 3. These loads are assumed normal to the planar area over which they are applied.
- Point loads in any direction located anywhere within the model.
- Nodal loads.

Recall from Chapter 3 that the integration of the surface load over a specified area is performed using the rectangular rule, which reduces to the summation of a series of point loads applied within elements. As a result, the main computational routine for load application is that for applying an arbitrarily located point load within a model. This does require that the element within which the load is applied be determined, and the element shape functions be computed. This is accomplished through element member functions originally developed to aid in the implementation of the embedded dowel and the multigrid solution strategies which are detailed later in this Appendix. The *Load_RHS* class also implements self-weight loading; the integration necessary to determine equivalent nodal loads due to self weight is performed by individual *Element* objects.

The *Load_Temp* class was developed to generate equivalent nodal loads due to temperature. Temperatures are specified on a nodal basis and stored in the *Load_Temp* class; appropriate private member functions are provided for generating *Element* nodal temperature

vectors. As with self-weight, the individual *Element* objects are responsible for integrating temperature-induced stresses over their volume to determine equivalent nodal loads.

A.2.3 Numerical Classes

Dense Array and Vector Classes

To facilitate element-level operations such as forming element stiffness matrices and computing element residual force vectors and loads, a set of class objects was coded specifically for linear algebra using dense matrices. These objects support zero-based parenthetical indexing through overloading of the "()" operator. Storage is allocated upon construction, and the objects may be dynamically resized at any time. To maximize performance, no checking for out-of-bound indices is supported; the programmer is responsible for sizing all arrays/vectors prior to element assignment/retrieval and matrix operations. All of the computations performed in this study were done in double precision. However, the class objects reference the global type definition `typedef double Number`, which would permit recompilation of a single precision version if desired. Throughout this Appendix, *Number* and *double* are used interchangeably.

These classes support symbolic linear algebra operations (matrix-vector multiplication, vector dot products, etc.) through public member functions employing overloaded operators. For example, the product of matrix $\mathbf{A}$ with vector $\mathbf{x}$ giving the vector result $\mathbf{b}$ is coded as $\mathbf{b} = \mathbf{A} * \mathbf{x}$. Results of computations are returned by reference to object member data to increase efficiency, i.e. the vector result $\mathbf{b}$ is returned by reference from $\mathbf{A}$ in the previous example.

A brief description of the individual objects is given below:

- *dubVector* is a simple vector class containing an array of *Numbers*. Vectors may be created using an empty constructor, as having a specified number of elements, or as defined by two *Nodes*. Public member functions support vector dot and cross products, vector-matrix multiplication, vector tensor products, assignment of a scalar or another vector (with dynamic reallocation), retrieval of maximum or minimum values, and sorting of values.

- *dub2Darray* is a dense matrix class. Arrays may be sized upon construction or dynamically resized after instantiation. Overloaded operators permit matrix-matrix and matrix-vector operation, etc. Individual columns or rows may be assigned or extracted, as may block matrices.

- *dub_symm_array* provides for storage and operation on dense symmetric matrices. Storage is allocated using individually sized *dubVectors* to represent each row. In addition to supporting matrix operations with overloaded operators, Choleski factorization and substitution are supported via public member functions. This class was created primarily for optimal storage and manipulation of element stiffness matrices.

Sparse Representation of the System Stiffness Matrix

For the 3D models solved in this study, the system stiffness matrix, **K**, is large. This requires that it be stored in a sparse format to minimize memory usage; the *Sparse_Col_K* class was developed for this purpose. To permit the use of existing numerical libraries for the solution of the linear system of equations arising from the finite element discretization, **K** is stored in a Harwell-Boeing column-major format. The storage required is member data consisting of:

```
int nnz;
double* nzval;
int* rowind;
int* colptr;
```

The value of `nnz` is the number of non-zero values, the array `nzval` stores the non-zero coefficients of **K** ordered by column, `rowind` the corresponding row indices, and `colptr` the starting and ending indices of each column within `nzval`.

In addition to storing **K**, the *Sparse_Col_K* object is responsible for allocating storage space for and assembling **K**. The graph of **K** is computed prior to assembly by a *Mesh* public member function, ensuring that `nzval` and `rowind` are sized exactly and the values of `colptr` are known. This eliminates the need for expensive memory reallocations and shifting of the columns of **K** during assembly.

The major public member functions of *Sparse_Col_K* permit assembly of **K**, the removal of equations corresponding to zero displacements, and adding or removing individual element stiffness matrices:

```
int form_K()
void apply_bc()
void remove_element(int elnum)
void add_element(int elnum)
```

Member functions are also provided for access to `nzval`, `rowind`, and `colptr` as required by the *LU_Solver* object (described later). Other *Sparse_Col_K* public member functions are required for the multigrid solution strategies and are detailed in later sections.

A.2.4 Analysis Classes

The *Solver* object is the top level analysis object, and implements the solution strategy outlined in Algorithm 6.1. As shown in Fig. A.2, *Solver* relies on a *Constraint* class for enforcing fixed non-zero displacement and frictionless contact constraints (detailed in a later section of this Appendix). Description of the *Multi_grid* and *Conj_grad* classes used in

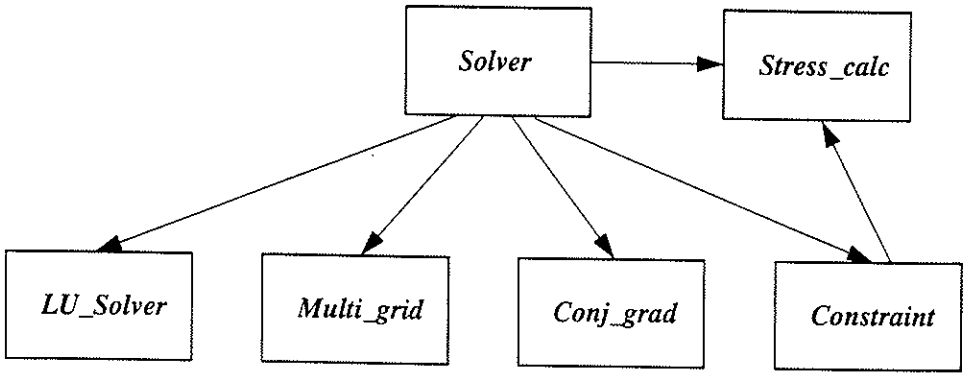


Figure A.2: Analysis Classes

the solution of the linearized system stiffness equations are also deferred until later in this Appendix.

The *LU_Solver* object is merely a wrapper providing an interface to the SuperLU sparse direct solver library routines [17, 57]. The *Multi_grid* class relies on *LU_Solver* for achieving the coarse grid solution; *LU_Solver* may also be used to directly solve the system stiffness equations for smaller problems that do not require the use of iterative solution methods. Public member functions are provided for factoring $\mathbf{K}$, and substitution for the solution. Multiple substitutions may be performed after a single factorization to increase efficiency.

Smoothed nodal stresses, along with nodal displacements, are the primary output from the finite element analysis. In addition, they are required for updating nodal contact constraints as detailed in Chapter 6. The *Stress_calc* object averages and stores the smoothed nodal stresses over the entire mesh. Each *Element* object is responsible for computing its contribution to the smoothed nodal stresses based on extrapolating the integration point stresses.

A.3 Implementation of Embedded Dowel

The object-oriented implementation of the embedded dowel element detailed in this Section mirrors the element development as closely as possible. Following a description of the *Dowel* object, details of the *Dowel_gap* object, responsible for storing bond and gap/constraint information as well as implementing the constraint updating strategy, are presented. Note that specifics of the embedded formulation of the *Dowel* object are also applicable to the 2-noded *Beam_2* element used to span the joint between the two slabs.

A.3.1 Modifications to the *Dowel* Class

With the usual (unembedded) formulation of the 3-noded quadratic *Dowel* element in hand, only the transformation matrix, $\mathbf{T}$ (see Chapter 4) need be computed to embed the dowel in a solid element. The information required to form $\mathbf{T}$ is as follows:

- The solid element which the dowel is embedded in must be specified.
- The matrix of shape functions of the embedding element, $\mathbf{N}_i^e$, must be evaluated for each dowel node $i = 1 \ldots 3$.
- The dowel must be specified as bonded or debonded at each node.
- At each dowel node, i, the gap, γ_i and current constraint information must be known.

Given the small amount of information required to embed the dowel and to avoid code replication, the member data and functions necessary to embed the dowel were merely added to the existing unembedded *Dowel* class. The additional member data required for the embedded dowel are:

```
Element* embedder;
Dowel_gap* gap_bond;
dubVector* embed_coords;
```

Embedment is specified by the assignment of `embedder` and `gap_bond` objects through the following public member function:

```
void assign_embed(Element* embed, Dowel_gap* gapper)
```

Element public member functions permit the computation of the local embedding element coordinates (detailed later) and evaluation of $\mathbf{N}_i^e$. Only a few additional private member functions were necessary to permit the generation of $\mathbf{T}$ once access to the `embedder` public member functions was provided.

One other issue which must be addressed upon embedment of the dowel is its additional nodal connectivity and number of degrees of freedom. As mentioned earlier, each *Element*, *Beam*, and *Foundation* object stores its nodal degrees of freedom for purposes of mesh traversal, sizing of data structures, and mapping element matrices/vectors to the global matrices/vectors. Therefore, upon embedment the *Dowel* simply appends the nodal degree of freedom information as retrieved from `embedder` to its original unembedded nodal degree of freedom data. This implies that the dowel has *six* degrees of freedom at its three original nodes and *three* degrees of freedom at the nodes corresponding to `embedder` – see Chapter 4 for details. The nodal connectivity of the embedded dowel is then modified by a call to a *Mesh* public member function. These two steps, accomplished with a few lines of code in the *Dowel* `assign_embed` member function, ensure that interaction with other objects such as *Mesh* and *Sparse_Col_K* through existing *Dowel* public member functions is unaffected.

A.3.2 The *Dowel_gap* Class

The *Dowel_gap* object was created to facilitate the storage and updating of nodal constraint and bond information for all embedded *Beam* elements in a model. Member data for the *Dowel_gap* object include the following:

```
Member* ptr_to_members;
Mesh* ptr_to_mesh;
Sparse_Col_K* K;
El_conn bond_info;
El_conn constraint_info;
dubVector nodal_gaps;
```

Access to the *Beams* is provided through `ptr_to_members`, and bond, constraint on/off, and gap magnitudes are stored in `bond_info`, `constraint_info`, and `nodal_gaps`, respectively. The following public member functions provide the *Beam* objects with information required to form **T**:

`int get_bond_info(int node_num)`
Returns a 1 if node is bonded, 0 if not.

`int get_constr_info(int node_num)`
Returns a 1 if node is constrained, 0 otherwise.

`void get_gap_ratio(Number& gap_ratio, dubVector& dowel_displ,`
`dub2Darray& N)`

Given the dowel nodal displacement vector and the matrix of shape functions of the embedding element, $\mathbf{N}_i^e$, this function returns by reference the gap ratio, γ/m.

A review of Chapter 4 will verify that this is the only information required to form $\mathbf{T}$.

Constraint updating and modification of the tangent system stiffness stiffness matrix are also the responsibility of the *Dowel_gap* object. See Algorithm 4.1 for details of the constraint updating algorithm. The *Dowel_gap* object stores as member data a pointer to the *Sparse_Col_K* object permitting it to add/remove *Beam* element stiffness matrices. Additional *Dowel_gap* public member functions used to accomplish these tasks are:

```
int update(dubVector& nodal_displ)
int remove_K()
int add_K()
```

A.4 Multigrid Implementation

As noted earlier, the *Solver* object implements the nonlinear solution algorithm detailed in Chapter 6. The inner kernel of this algorithm requires the solution of a system of linear equations, which is usually accomplished with the multigrid (MG) or multigrid-preconditioned conjugate gradient (MG-PCG) method. Presentation of the MG implementation is the primary focus of this section; with this in hand, the implementation of MG-PCG is briefly discussed.

A.4.1 Implementation of the Multigrid Algorithm

The top level object in the multigrid implementation is the *Multi_grid* class, which simply implements the recursive multigrid algorithm given in Alg. 6.3. Its main public member function takes a force vector and returns a displacement vector after a specified solution tolerance or maximum number of iterations has been reached. Member data for this object include pointers to the *Mesh*, *Members*, and *Sparse_Col_K* objects corresponding to each mesh in the grid hierarchy as well as a pointer to the *LU_Solver* for determination the coarse grid error correction. See Fig. A.3 for an illustration.

A review of Alg. 6.3 indicates that the multigrid method relies on smoothing and methods for the intergrid transfer of information, including the restriction and interpolation

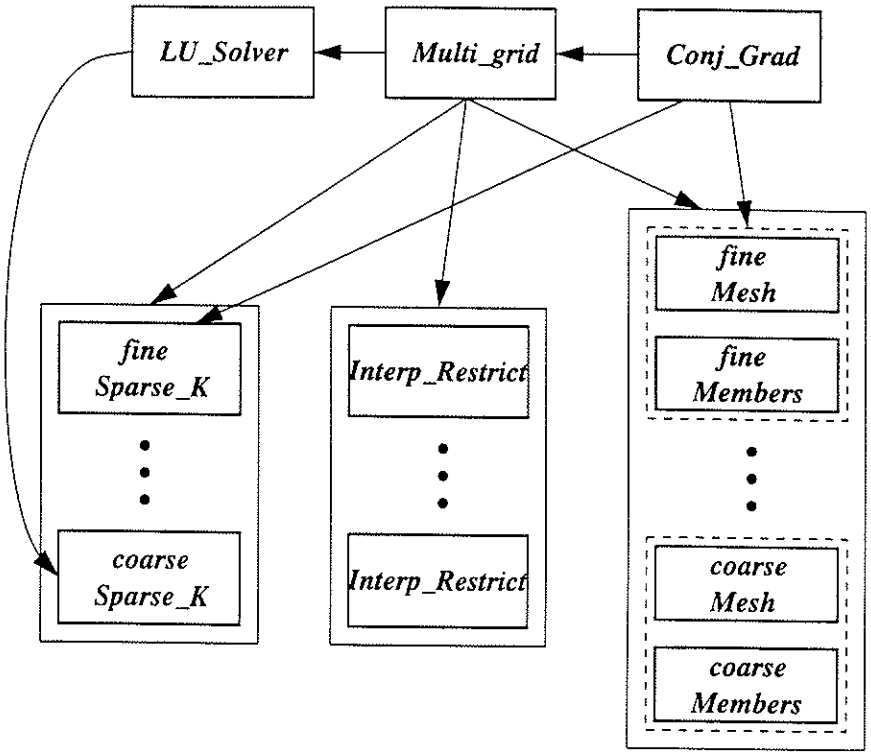


Figure A.3: Multigrid Class Structure and Implementation

operators developed in Chapter 6. Smoothing is the simplest fundamental component of the multigrid implementation. Recall that point-wise Gauss-Seidel smoothing is used, which requires only the system stiffness matrix and the current residual and displacement vectors. As such, additional functions were simply added to the *Sparse_Col_K* object to perform forward and backward Gauss-Seidel smoothing operations. Implementation of the intergrid transfer of information concepts presented in Chapter 6 is more detailed, and is the focus of the following discussion.

A.4.2 Intergrid Transfer of Information

At the heart of the intergrid transfer of information concepts developed in Chapter 6 lies the need to efficiently determine for any *Node*: (1) the element it is contained within, and (2) the local (natural) coordinates of the containing element. This information is required for restriction/interpolation as well as the intergrid transfer of material properties, where all coarse mesh elements must access the finest mesh at their integration points. As detailed in Chapter 6, a grid searching procedure is needed to reduce the number of elements which must be checked for each *Node*. This grid search has been implemented using the *Grid_sort*

class, detailed next. Following this, the modifications to the *Members* class and various element classes that permit the determination of (1) and (2) will be discussed. Finally, the specifics of the restriction/interpolation and intergrid transfer of material properties will be covered.

Grid Searching Procedure

The *Grid_sort* class implements the grid searching algorithm outlined in Chapter 6. In addition to constructors and a destructor, it contains only two public member functions:

```
void build_sort()

El_conn& get_list(Node& node)
```

The `build_sort` function is responsible for constructing a 3D grid over the domain of the *Mesh*, and assigning to each cell in the grid a list of all elements which lie partially or completely within it. Assuming straight element edges, the only information required to generate this data are the maximum and minimum extents of the *Mesh* in the (x, y, z) Cartesian coordinates, and the maximum and minimum (x, y, z) extents of each element. This information is computed and stored by the *Mesh* during the assignment of elements and nodes, and accessed by the *Grid_sort* object via *Mesh* public member functions. The algorithm implemented by `build_sort` is given in Alg. A.1 with reference to Fig. A.4 and the following definitions. Only the 2D case is outlined for brevity.

$$
\begin{aligned}
X_{min}, X_{max} &= \text{minimum and maximum } x \text{ coordinates of mesh} \\
Y_{min}, Y_{max} &= \text{minimum and maximum } y \text{ coordinates of mesh} \\
x^i_{min}, x^i_{max} &= \text{minimum and maximum } x \text{ bounds of } i^{th} \text{ element} \\
y^i_{min}, y^i_{max} &= \text{minimum and maximum } y \text{ bounds of } i^{th} \text{ element} \\
n_{dx}, n_{dy} &= \text{number of divisions of grid in } (x, y) \text{ directions}
\end{aligned}
$$

```
build_sort()
 1  [X_min, X_max, Y_min, Y_max] = Mesh → get_mesh_limits()
 2  δx = (X_max − X_min)/n_dx
 3  δy = (Y_max − Y_min)/n_dy
 4  ▷ loop over all elements
 5  for i = 1 to number of elements
 6  do [x^i_min, x^i_max, y^i_min, y^i_max] = Mesh → get_el_limits(i)
 7      ▷ starting and ending cell indices
 8      low_x = (int) (x^i_min − X_min) /δx
 9      high_x = (int) (x^i_max − X_min) /δx
10      low_y = (int) (y^i_min − Y_min) /δy
11      high_y = (int) (y^i_max − Y_min) /δy
12      ▷ loop over each row in grid
13      for j = low_y to high_y
14      do k_s = j * n_dx + low_x
15          k_e = j * n_dy + high_x
16          ▷ loop over each column in grid
17          for k = k_s to k_e
18          do el_nums[k].add_val(i)
19          end
20      end
21  end
```

Algorithm A.1: Pseudo-code for the `build_sort` Function

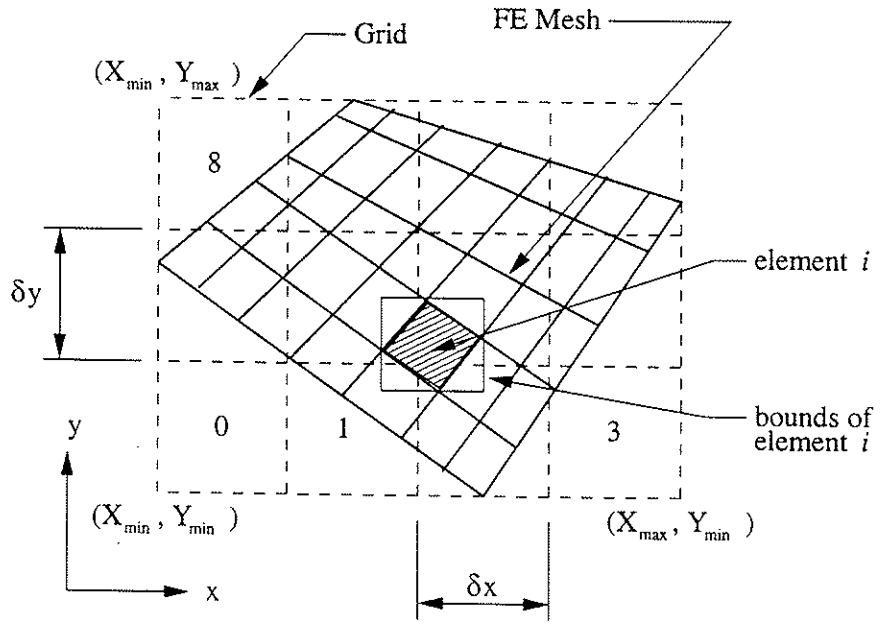


Figure A.4: Definitions for Grid Searching Algorithm

Note that this algorithm allows for an element to span multiple cells in the grid in both the x and y directions. The computational time required by this algorithm is negligibly small, although it does grow nonlinearly with increasing n_{dx} and n_{dy}. The integer list of elements lying at least partially in the k^{th} each grid cell are stored in $el\_nums[k]$, a dynamically sized *El_conn* object. The *Grid_sort* member function `get_list(Node& node)` merely computes the k^{th} cell that `node` lies within and returns $el\_nums[k]$.

Although the *Grid_sort* class was originally constructed to aid in the multigrid implementation, it is used in other parts of the program for operations that would otherwise require linear searches over all elements. For example, the application of arbitrarily located point and patch loads described in Chapter 3 requires that the element within which a load is applied be known. The formation of the graph of the system stiffness matrix, $\mathbf{K}$, prior to assembly requires that all elements connected to each node be determined. Given that there are several uses for the *Grid_sort* object and that it interacts primarily with the elements, it is constructed by and stored as member data for each instance of *Members*.

Calculation of Element Local Coordinates

Two public member functions were added to the *Members* class which provide the primary interface for the intergrid transfer of information:

```
int in_element(int nodal_dof, Node& node, El_conn& el_groups)
int get_local(int i, Node& node, dubVector& loc_coords)
```

The first member function triggers a call to *Grid_sort*, which returns the list of elements to search. The input `nodal_dof` and `el_groups` are necessary to ensure that the proper element type is searched, i.e. *Element*, *Foundation*, or *Beam*. The returned integer is the global index of the element. The second function takes this global index, `i`, along with `node`, and returns by reference the element local coordinates.

To avoid code replication, all calculations required to determine the element local coordinates are performed by the elements themselves, which already contain most of the required data and member functions. The determination of whether or not a node lies in an element reduces to simple vector dot products. The calculation of element local coordinates given a nodal location, while more complicated, requires only a straightforward application of Newton's method. For an isoparametric, 3D continuum element with b nodes, we know that:

$$\mathbf{G} = \mathbf{N}(\boldsymbol{\Upsilon})\mathbf{x}^* - \mathbf{x} = \mathbf{0}$$

where:

$$\begin{aligned}
\mathbf{x} &= \text{3-element vector containing the given } (x, y, z) \text{ location} \\
\mathbf{x}^* &= \text{vector of element nodal locations, } [x_1, y_1, z_1, \ldots x_b, y_b, z_b]^{\mathrm{T}} \\
\mathbf{N} &= \text{the matrix of element shape functions} \\
\boldsymbol{\Upsilon} &= \text{vector of element local coordinates, } (\xi, \eta, \zeta)
\end{aligned}$$

Applying Newton's method for the solution of this equation requires the Jacobian, $\mathbf{J}$, given by:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial N_i}{\partial \xi} x_i & \frac{\partial N_i}{\partial \eta} x_i & \frac{\partial N_i}{\partial \zeta} x_i \\ \frac{\partial N_i}{\partial \xi} y_i & \frac{\partial N_i}{\partial \eta} y_i & \frac{\partial N_i}{\partial \zeta} y_i \\ \frac{\partial N_i}{\partial \xi} z_i & \frac{\partial N_i}{\partial \eta} z_i & \frac{\partial N_i}{\partial \zeta} z_i \end{bmatrix}$$

where summation over i is implied. A review of isoparametric element definitions reveals that $\mathbf{J}$ is the transpose of the Jacobian used in coordinate mapping operations [98, 20], and no extra element functionality is required for its computation. Solution for the unknown element local coordinate vector, $\boldsymbol{\Upsilon}$, using Newton's method then follows as:

```
int get_local(x, Υ)
 1  ▷ start at center of element
 2  Υ = (0, 0, 0)
 3  while ‖G(Υ)‖ > 0
 4  do J = form_Jacobian()
 5       ▷ solve for increment
 6       δΥ = J^-1 G
 7       Υ+ = δΥ
 8  end
```

Algorithm A.2: Solve for Element Local Coordinates

Restriction and Interpolation

Since restriction and interpolation as defined in Chapter 6 employ the same information about adjacent meshes in the hierarchy, it was logical to develop a single class, *Restrict_interp*, to perform these operations between adjacent *Mesh/Members* pairs as shown Fig. A.5. This implies that the number of *Restrict_interp* objects is always one less than the number of meshes in the hierarchy. In addition to constructors and a destructor, the *Restrict_interp* class provides only three public member functions:

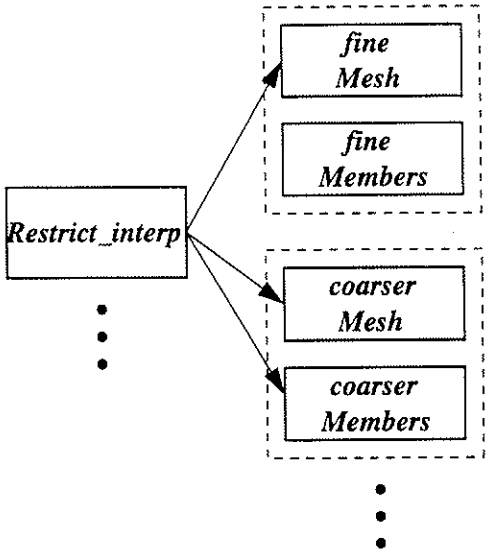


Figure A.5: *Restrict_interp* Class

`build_data()`

This function is called only at the start of the simulation. It is responsible for determining the integer list of coarse mesh elements within which each fine mesh node lies, and the corresponding list of coarse mesh element local coordinates. This information is stored as member data by the *Restrict_interp* object.

`restrict(dubVector& fine_res, dubVector& coarse_res)`

Restricts the fine mesh residual vector to the coarse mesh. Requires a loop over all fine mesh nodes, and assembly/summation of the equivalent coarse mesh contributions to get `coarse_res`.

`interp(dubVector& coarse_displ, dubVector& fine_displ)`

Performs interpolation using the node-by-node procedure detailed in Chapter 6. Requires a loop over all fine mesh nodes, and mapping of interpolated fine mesh nodal displacements to `fine_displ`.

The public member functions for restriction and interpolation are called by the *Multi_grid* object during the solution. Restriction and interpolation are actually performed on a node-by-node basis by the individual *Element*, *Beam*, and *Foundation* objects where the required definitions of the element shape functions are already provided. In addition to minimizing code replication, this definition has the significant advantage that explicit representations of the operators $\mathbf{R}$ and $\mathbf{T}$ need not be formed. The additional computational overhead

incurred by the evaluation of the element shape functions for each restriction/interpolation operation is minimal.

Intergrid Transfer of Material Properties

The intergrid transfer of material properties is accomplished at the element level with the following additional required element member functions:

```
void assign_members(Members* fine_members)
void get_D(Node& gauss_nodes[i], dub2Darray& D)
```

After the elements of all coarser meshes in the multigrid hierarchy have been constructed, they are assigned pointers to the fine *Members* object. The fine mesh element numbers and local coordinates corresponding to the coarse mesh element integration points are computed and stored as coarse element member data. This permits the constitutive matrix, **D**, relating stress and strain to be retrieved from the fine mesh elements during the calculation of the element stiffness matrix. Currently, these member functions have been added to each class in the *Element* and *Foundation* class hierarchies; the *Beam* elements do not account for spatially varying material properties when a multigrid solution is employed. Note that at present, this implementation is limited to total displacement nonlinearities (such as the aggregate interlock constitutive relations developed in Chapter 5). More complex constitutive models would require the tracking of more loading history parameters. In principle, however, a class could be developed to compute and update required parameters at each integration point, allowing the mechanisms for the intergrid transfer of material properties to remain largely unchanged.

A.4.3 Multigrid-Preconditioned Conjugate Gradient Implementation

Implementing the multigrid-preconditioned conjugate gradient solver (MG-PCG) is simple once the multigrid solver is in place. A *Conj_grad* class was coded specifically to implement the preconditioned conjugate gradient algorithm. Once the algorithm was coded, all that was required to implement MG-PCG was to give the *Conj_grad* object pointer access to *Multi_grid*.

At present, three other options are also available for preconditioning: diagonal, symmetric Gauss-Seidel, and incomplete LU factorization (ILU). The diagonal preconditioner retrieves the vector of diagonal elements from the *Sparse_Col_K* object, and the symmetric Gauss-Seidel preconditioner relies on the forward and backward smoothing routines coded

for the *Multigrid* class and implemented by the *Sparse_Col_K* object. The ILU factorization is computed and stored by the *Sparse_Col_K* object, and substitution performed via an additional public member function.

A.5 Enforcing Geometric Constraints

The nonlinear solution strategy developed in Chapter 6 employs an Uzawa iteration to enforce the geometric constraints required to model loss of contact between the slab and base layer. The *Constraint* class, described in this Section, was developed specifically to aid in implementing these nodal contact constraints.

A.5.1 The *Constraint* Class

The *Constraint* class is an analysis object that interacts directly with the *Solver* class, and in addition to nodal contact, permits the specification of non-zero displacement constraints (not used in this study). The *Constraint* class contains the following public member functions that apply directly to the Uzawa iteration; refer to Chapter 6 for notation.

`void get_perturb(dubVector& perturb_force)`
Responsible for returning the current perturbation to the right hand side of the constrained system, $\omega \mathbf{G}^k \mathbf{c}^{*k}$. This function is called once at the start of each global iteration.

`double get_constr_err(dubVector& G_transp_U)`
Returns by reference the current value of the residual in the constraints, $\mathbf{y} = \mathbf{G}^{k\mathbf{T}} \delta \mathbf{U} - \mathbf{c}^{*k}$, as well as the magnitude of $\mathbf{y}$, which is used as the convergence criteria in the Uzawa iteration.

`void get_G_lambda(dubVector& G_lambda, dubVector& del_lambda)`
Takes the current estimate for $\delta \boldsymbol{\lambda}$ and returns the matrix-vector product $\mathbf{G}^k \delta \boldsymbol{\lambda}$ by reference.

`int update_constraints()`
Updates constraints based on the current displaced shape and smoothed nodal stresses. At present, updating is based on a straightforward application of the criteria presented in Chapter 6 for frictionless nodal contact conditions.

`void add_constraints(int add)`
This function computes and adds/removes the perturbation to the system stiffness

matrix, $\mathbf{K}' = \omega \mathbf{G}^k \mathbf{G}^{k\mathbf{T}}$, based on the currently active constraints. The perturbation $\mathbf{K}'$ is added if add = 1, and removed if add = -1.

The implementation of the above member functions is straightforward when a direct solver is used in the inner kernel of the solution algorithm. An additional level of complexity is added when solving constrained models using multigrid methods; this is discussed next.

A.5.2 Multigrid Methods for Constrained Problems

The only difficulty in solving the constrained system with a multigrid method (or multigrid-preconditioned conjugate gradient) lies in restricting $\mathbf{K}'$ to the coarser meshes in accordance with Chapter 6. Recall that this process of restriction is performed node-by-node, requiring that the coarse mesh element within which each fine mesh nodal stiffness perturbation, $\omega(\mathbf{n} \otimes \mathbf{n})$, be known. However, with the element search capabilities previously added to the *Members* class for the multigrid implementation, determination of this information becomes a trivial task. Similarly, the functions required for the evaluation of the coarse mesh element shape functions are already in place, permitting the calculation of Eq. 6.30 with little work. The only additional modifications involve notifying the coarse *Mesh* that there is additional coupling between nodes not defined by element connectivities to ensure that storage for the system stiffness matrix is properly allocated.

Given this simplicity, the required functionality for perturbing the coarse mesh stiffness matrices was added to the *Constraint* class. During model generation, a single *Constraint* object is constructed for each mesh in the hierarchy; however, the coarse mesh *Constraint* objects are each assigned a pointer to the finest mesh *Constraint* object. This permits each coarse mesh *Constraint* to compute its equivalent perturbation, $\mathbf{K}'$, based on the current constraint information known and updated only on the fine mesh.