HIERARCHICAL FINITE ELEMENT MODELLING OF PLATES AND SHELLS

By

Ravi Guttal

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

DOCTOR OF PHILOSOPHY

Major Subject: Civil Engineering

Approved by the
Examining Committee:

Dr. J. Fish, Thesis Adviser

Dr. J. Flaherty, Member

Dr. M. Shephard, Member

Dr. R. Spilker, Member

Rensselaer Polytechnic Institute
Troy, New York

July 1996
(For Graduation August 1996)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

I dedicate the thesis to my parents Mr.and Mrs C. B. Guttal. I am grateful to my thesis advisor Dr. Jacob Fish for his guidance and support during the course of my research. I am honored to have had Prof. Flaherty, Prof. Shephard and Prof. Spilker in my doctoral committee and I thank them for their valuable suggestions. I am particularly grateful to Prof. Shephard for his suggestions during my finite element programming project, which helped me immensely in my research. I also thank Dr. Belsky for his comments and suggestions during my research.

I take this opportunity to acknowledge the encouragement and support received from my wife Dr. Rajeshwari and my friends during my stay at Rensselaer. It is a pleasure to acknowledge my gratitude to all the people involved directly or indirectly in the production of this manuscript.

ABSTRACT

The thesis is dedicated to exploring and investigating methodologies to enhance the performance and computational efficiency of hierarchical finite element methods for plate and shell models. A novel quadrature scheme and a family of hierarchical assumed strain elements have been developed to enhance the performance of the displacement-based hierarchical shell elements.

The *s*-version of finite element method is developed for laminated plates and shells. By this technique the global domain is idealized using 2D Equivalent Single Layer (ESL) model. The regions where the ESL model errs badly in capturing localized phenomena are superimposed by a stack of 3D elements. Assumed strain formulation and selective polynomial order escalation in the two models are employed to maintain high level of computational efficiency.

For an efficient solution of large scale hierarchic finite element systems an adaptive solver has been developed. A decision making methodology aimed at selecting an optimal solution strategy on the basis of estimated conditioning, sparsity and memory requirements for a given problem has been devised. Numerical experiments have been conducted on selected shell and 3D problems in the range of 1,000-100,000 degrees of freedom.

CHAPTER 1
Introduction

There has been a disagreement between various developers in the finite element community over the computational efficiency of higher order elements. On one hand there was a clear mathematical evidence of the superior theoretical rate of convergence (measured in terms of the problem size) of the p-type methods for properly designed meshes as demonstrated by Babuska, Szabo, and Katz [5]. However, it was commonly believed, primarily in the engineering community, that the h-method is computationally more efficient due to its superior sparsity. The disagreement has peaked in the early nineties. For example, in the First US Congress on Computational Mechanics, Bathe presented numerical results conducted on Floyd pressure vessel showing the superior performance in terms of CPU time of the h-method even for problems for which the exact solution is analytic. At the same conference Carnevali reported IBM research division findings on similar problems suggesting exactly an opposite trend.

In practice, computational efficiency of various finite element versions depends not only on sparsity and theoretical rate of convergence, but is a function of several other factors including adaptivity and quality control, conditioning, distortion sensitivity, locking, model preparation and model improvement, utilization of previous computations and coding simplicity. Ironically, there is no general consensus on the relative merits of some of these factors. For example, it has been argued that for p-type methods the finite element mesh is simpler, and thus the time required for data preparation is substantially smaller. Unfortunately in automated computational environment the cost of automatic mesh generation of higher order elements is not necessarily lower than that of the h-method [49].

The p-method has been commended for its versatility in the adaptive process

due to its ability to exploit previous computations and the elegance of hierarchical error estimation process [62]. However, it is often overlooked that the sequence of lower order finite element meshes generated in the adaptive process can be utilized for both solution and quality control processes by utilizing multigrid technology [12].

Contradicting observations were reported regarding the sensitivity to element distortion. Holzer, Rank, and Werner [29] present experimental results indicating that higher order elements are less sensitive to mesh distortion, while Ramm, Stander and Matzenmiller [44] in their review article on assumed strain shell formulation report that 4-node bilinear shell elements are less sensitive to mesh distortion than their quadratic counterparts.

In the realm of opposing views, there is a sound theoretical evidence on superior conditioning of matrices arising from orthogonal basis functions [62], and circumvention of locking with higher order elements as shown by Szabo, Babuska, and Chayapaty [55]. Nevertheless, since the overall computational efficiency is strongly linked to the program architecture, it is not obvious what are the contributing factors of these aspects. One of the focuses of the thesis is on computational aspects of the p-version of finite element method.

Laminated composites present to the analyst a hard nut to crack due to inadequacy of Equivalent Single Layer (ESL) theories in resolving 3D phenomena on one hand, and computational complexity of layer-wise or 3D models on the other hand. A natural remedy is a global-local approach in which different regions of the problem domain are described with different types of models. The author refers to [45] for comprehensive review of global-local techniques for composite laminates and to [21] and [6] for various aspects of reliability, convergence and accuracy of global-local techniques.

The focus of the thesis is only on the class of global-local techniques that advocates hierarchical solution strategy in the sense that information from the analysis

of an ESL model is exploited in resolution of local effects using Discrete Layer (DL) mesh. Among the most popular hierarchical global-local strategies are various forms of multigrid and composite grid methods [39] [11] [24] [58] [22] [12], as well as methods based on hierarchical decomposition of approximation space [15] [16] [17] [18] [10] [60] [45]. Recently the composite grid method originated for displacement-based linear systems has been extended to hybrid systems [23]. Engineering global-local approaches which approximate a detailed response by means of post processing techniques, such as subjecting refined discrete layer model to the boundary conditions extracted from the global ESL model, can be viewed as a single iteration within the composite grid procedure. For various improvements of this simple "zoom" technique the author refers to [41] [35] [38].

A robust computationally efficient solver for a myriad of large scale problems has been an elusive goal for the finite element community. The concept of such a solver poses a dilemma for solution method developers, *"should one try to devise a single strategy which can handle all problems in a computationally efficient manner, or develop problem specific computationally efficient strategies?"*. This dilemma is not new, just as in other branches of computational mathematics, both approaches are being pursued by researchers. In the current investigation the latter approach has been adopted. The adaptive solver proposed in the thesis, utilizes the properties of the given problem to select a computationally optimal strategy rather than applying a single solution strategy to every problem.

The outline of the thesis is as follows:

- The computational aspects of the p-version for shell analysis are explored in Chapter 2. The focus is on techniques to enhance the performance of shell elements and speed up the computation of element matrices.

- A hierarchic global-local method for analysis of laminated composites is investigated in Chapter 3.

- An adaptive solver for hierarchic systems, which selects an optimal solution strategy based on problem data is elucidated in Chapter 4.

- Finally the thesis is concluded with a summary and future work directions.

CHAPTER 2
Hierarchic p-type Shell Finite Element Formulation.

2.1 Introduction

This chapter focuses on the computational aspects of the p-version for shell analysis. The primary research efforts focus on enhancing element level computations, since the cost of analysis in the p-method is often dominated by the formation of finite element matrices (as opposed to the h-method, where the solution process is generally dominant even for small to medium size systems). The following aspects are studied:

- How to enhance the performance of shell elements up to the polynomial order of 4-5 using assumed strain formulation.
- How to speed up the computation of element matrices by utilizing previous computations and how to exploit hierarchiality of the p-method via special quadrature scheme.

2.2 Element Formulation

2.2.1 Preliminaries

Consider the geometry of a typical quadrilateral shell element defined by the following relation:

$$\mathbf{X} = \frac{1}{2}[(1+\xi_3)\,\mathbf{X}^{top}(\xi_1,\,\xi_2) \;+\; (1-\xi_3)\,\mathbf{X}^{bot}(\xi_1,\,\xi_2)] \tag{2.1}$$

where $\mathbf{X}$ denotes the position vector of a generic point of the shell in the global Cartesian coordinate system, $\mathbf{X}^{top}$ and $\mathbf{X}^{bot}$ are position vectors at the top and bottom surfaces, respectively. It is common in practice to interpolate the bottom and the top surfaces either using Lagrange polynomials [53], blending functions

[26] or even Legendre polynomials [13]. The displacement field of a higher order plate/shell theory can be approximated [54].

$$u_i = \sum_{a=1}^{n_i} f_a(\xi_3)\, u_{i|a}(\xi_1,\, \xi_2) \tag{2.2}$$

where $f_a(\xi_3)$ in equation (2.2) represents through-the-thickness variation of the displacement components. Typically, Legendre polynomials of order p are chosen as basis functions for in-plane displacement components $u_{i|a}(\xi_1,\, \xi_2)$ to ensure numerical stability and hierarchiality. q $(q \leq p)$ is the polynomial order of basis functions in transverse direction identified with the polynomial order of $f_a(\xi_3)$.

Shell elements formulated on the basis of equations (2.1) and (2.2) will be referred to as **H3SOL** - Hierarchical $(3-D)$ Solid element. To enhance the performance of **H3SOL** for $q = 1$ plane stress assumption and shear correction factors are usually employed.

H3SOL is known to have the following drawbacks:

1. Locking for lower order elements and hence poor convergence. To circumvent this problem we propose to employ, an assumed natural strain formulation by introducing (see section 2.2.2) the

   - Hierarchical $(3-D)$ Assumed Natural Strain element that will be referred to as **H3ANS**. For $q = 1$ it employs plane stress assumption and shear correction factors.

2. In thin shell limit, retention of 3 degrees-of-freedom at each node on top and bottom surfaces leads to large stiffness coefficients for relative displacements corresponding to shell thickness [52]. This leads to deterioration of the rate of convergence for lower polynomial orders. This phenomenon is especially prominent when thickness is small compared to the in-plane dimensions. To alleviate this drawback, we propose either to reduce the stiffness in the transverse direction, or to employ hierarchical assumed natural strain degenerated

element formulation with rotational degrees-of-freedom. Thus in following sections, the following types of elements are described:

- Hierarchical ($3-D$) Reduced transverse stiffness, Assumed Natural Strain element denoted by **H3RANS**.
- Hierarchical ($2-D$) degenerated Assumed Natural Strain element with rotational degrees-of-freedom denoted by **H2ANS**.

H2-type elements have 5 degrees-of-freedom per hierarchical mode. They are used for thin and moderately thick shells. H3-type elements are intended to model very thin shell behavior as well as general compact solid behavior. They enable a uniform $3-D$ idealization of the problem domain that avoids $2-D/3-D$ transitions.

2.2.2 H3ANS Hierarchical $(3-D)$ Assumed Natural Strain element

Definition of Coordinate Systems and Basis Vectors. Let $\mathbf{X}$ be the Global Cartesian coordinate system where geometry of the shell domain is defined and basis vectors $\mathbf{e}_i$ be the unit Cartesian vectors. Let ξ represent a Natural Element curvilinear coordinate system. The covariant basis vectors $\mathbf{a}_i$ and their contravariants $\mathbf{a}^i$ are defined as follows:

$$\mathbf{a_i} = \frac{\partial X_j}{\partial \xi_i}\,\mathbf{e_j} \qquad \mathbf{a^i} = \frac{\partial \xi_i}{\partial X_j}\,\mathbf{e_j} \qquad \mathbf{a_i}\cdot\mathbf{a^j} = \delta_i^j \tag{2.3}$$

Let $\mathbf{x}$ be the Material Cartesian coordinate system where material properties of the element are defined. Unit basis vectors for the Material coordinate system are denoted by $\mathbf{p}_i$. They are defined such that, $\mathbf{p}_3$ is perpendicular to mid-surface and $\mathbf{p}_1$, $\mathbf{p}_2$ are as close as possible to $\mathbf{a}_1$, $\mathbf{a}_2$.

Strain Field in Natural Coordinate System of the Element. By the principle of invariance:

$$E_{ij}\ \mathbf{e_i}\ \mathbf{e_j} = \Xi_{ij}\ \mathbf{p_i}\ \mathbf{p_j} = \varepsilon_{ij}\ \mathbf{a}^{\mathbf{i}}\ \mathbf{a}^{\mathbf{j}} \tag{2.4}$$

where E_{ij}, Ξ_{ij}, ε_{ij} are components of Global, Material and Natural covariant strain tensor. Let $\mathbf{u} = u_i\ \mathbf{e}_i$ be the displacement vector. Then the strain tensor $\mathbf{E}$ is obtained by taking the symmetric gradient of the displacements, that is,

$$\mathbf{E} = \frac{1}{2}\left[\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i}\right]\ \mathbf{e_i}\ \mathbf{e_j} = \varepsilon_{ij}\ \mathbf{a}^{\mathbf{i}}\ \mathbf{a}^{\mathbf{j}} \tag{2.5}$$

Contracting $\mathbf{E}$ by covariant curvilinear vectors $\mathbf{a_i}$, $\mathbf{a_j}$ and using equation (2.3), yields the natural covariant strain tensor components

$$\varepsilon_{ij} = \frac{1}{2}\left[\frac{\partial u_m}{\partial \xi_i}\frac{\partial X_m}{\partial \xi_j} + \frac{\partial u_n}{\partial \xi_j}\frac{\partial X_n}{\partial \xi_i}\right] = \frac{1}{2}\left[\mathbf{u}_{,\xi_i}\mathbf{a}_j + \mathbf{u}_{,\xi_j}\mathbf{a}_i\right] \tag{2.6}$$

Let $\mathbf{N} \in S^{(p,q)}$ be the shape functions obtained using a tensor product of Legendre polynomials [54], then

$$\mathbf{u} = \sum_{A=1}^{NMDS} N_A\ \mathbf{d}^A \tag{2.7}$$

where $\mathbf{d}$ is the displacement vector representing the amplitudes of hierarchical modes in the global coordinate system; NMDS is the number of modes. Using equation (2.6), the natural strain-displacement relation can now be recast as:

$$\varepsilon = \sum_{A=1}^{NMDS} \mathbf{B}_A^{nat}\ \mathbf{d}^A \tag{2.8}$$

where

$$\mathbf{B}_A^{nat} = \{b_{Aij}^{nat}\} = \frac{1}{2}\left(N_{A,\xi_i}\mathbf{a}_j\ +\ N_{A,\xi_j}\mathbf{a}_i\right) \tag{2.9}$$

Assumed Natural Strain Field. In order to alleviate membrane and shear locking primarily at lower polynomial we define an assumed natural strain interpolants $\bar{\mathbf{B}}_A^{nat} = \{\bar{b}_{Aij}^{nat}\}$ in the following manner: Let (NG_1, NG_2, NG_3) be the number of quadrature points for the displacement based formulation. To enhance

the element performance, a special set of one-dimensional shape functions are introduced $[\phi_J(\xi_1), \phi_K(\xi_2), \phi_M(\xi_3)]$ defined with nodes at reduced quadrature points $(\bar{\xi}_{1I}, \bar{\xi}_{2J}, \bar{\xi}_{3K})$ where $J \in [1, NG_1 - 1], K \in [1, NG_2 - 1], M \in [1, NG_3 - 1]$.

The general form of $\{\bar{b}^{nat}_{Aij}\}$ is given by:

$$\begin{aligned} \bar{b}^{nat}_{Aii} &= \sum_{I=1}^{NG_I-1} b^{nat}_{Aii}(\xi_i = \bar{\xi}_{iI}, \xi_j, \xi_k)\ \phi_I(\xi_i) \quad \text{no sum on i} \\ \bar{b}^{nat}_{Aij} &= \sum_{I=1}^{NG_I-1} \sum_{J=1}^{NG_J-1} b^{nat}_{Aij}(\xi_i = \bar{\xi}_{iI}, \xi_j = \bar{\xi}_{jJ}, \xi_k)\ \phi_I(\xi_i)\phi_J(\xi_j)\ \ i \neq j \end{aligned} \tag{2.10}$$

no sum on i,j.

Stiffness Matrix Calculations. Since the constitutive relations are expressed in material coordinate system, the natural strains are transformed to material coordinate system. From equations (2.4) and (2.5), the strain components in material coordinate system are defined as:

$$\Xi_{kl} = \frac{\partial \xi_i}{\partial x_k}\ \frac{\partial \xi_j}{\partial x_l}\ \varepsilon_{ij} = \left[\frac{\partial x_k}{\partial \xi_i}\right]^{-1}\ \left[\frac{\partial x_l}{\partial \xi_j}\right]^{-1}\ \varepsilon_{ij} \equiv T_{klij}\ \varepsilon_{ij} \tag{2.11}$$

or

$$\mathbf{\Xi} = \mathbf{T}\ \varepsilon \tag{2.12}$$

and the element stiffness matrix can be cast into the classical form:

$$\mathbf{K^e} = \int_\Omega \bar{\mathbf{B}}^{nat^T}\ \mathbf{D}^\varepsilon\ \bar{\mathbf{B}}^{nat}\ d\Omega$$

where $\bar{\mathbf{B}}^{nat}$ is defined by equation (2.10) and

$$\mathbf{D}^\varepsilon = \mathbf{T}^T\ \mathbf{D}^\Xi\ \mathbf{T} \tag{2.13}$$

$\mathbf{D}^\Xi$ is the constitutive matrix defined in the Material coordinate system.

2.2.3 H3RANS - Hierarchical $(3-D)$ Reduced Transverse Stiffness, Assumed Natural Strain Element.

For the purpose of examining the causes of somewhat stiffer behavior of H3-type elements compared to their degenerated counterparts [47], consider a beam problem. For an elastic isotropic beam the strain energy is given by,

$$U = \frac{1}{2}\int_L (D_B\kappa^2 + D_M\epsilon^2 + D_S\gamma^2)dx \tag{2.14}$$

where L is the element length; ϵ, κ, andγ are the membrane strain, curvature and transverse shear strain respectively; D_B, D_M, and D_S are the bending, membrane and shear stiffness constants given by,

$$D_B = \frac{E\,t^3}{12} \quad D_M = E\,t \quad D_S = k_s\,G\,t \tag{2.15}$$

where t is the thickness of the beam of a unit width; E the Youngs modulus; G the shear modulus and k_s the shear correction factor.

In the classical beam formulation the normal strains μ are *a posteriori* calibrated to maintain zero normal stress (plane stress assumption), and thus have no contribution to the strain energy in equation (2.14). It can be seen that as $t \rightarrow 0$ the bending energy becomes negligible in comparison to shear and membrane energy giving rise to shear and membrane locking, if the element cannot represent deformed state in which shear and membrane strains vanish through out the element [9].

In H3-type beam elements normal strains are computed directly from kinematics. These values are not arbitrary and cannot be calibrated to maintain the plane stress condition. Thus if two dimensional state of stress is considered, the resulting strain energy takes the following form:

$$U = \frac{1}{2}\int_L (\bar{D}_B\kappa^2 + \bar{D}_M\epsilon^2 + \epsilon\,D_C\,\mu + D_S\gamma^2 + D_M\mu^2)dx \tag{2.16}$$

It can be seen that in H3-type flexural elements spurious coupling between membrane and normal deformation exists giving rise to a parasitic transverse normal

strain energy, which is of the same order of magnitude as that of the membrane strain energy if the strains are of equal order. This phenomenon is referred here as the transverse normal locking of H3-type flexural elements.

To ameliorate the locking caused by the transverse normal strains we propose to calibrate the constitutive behavior of H3-type elements to match the strain energy corresponding to H2-type elements without introducing zero energy modes. This is accomplished by modifying coefficients in the constitutive tensor in the following way:

$$\bar{D}_M = D_M \qquad \bar{D}_B = D_B \quad D_C = 0 \quad D_\mu = \chi \; D_M \tag{2.17}$$

where χ is a stabilization parameter aimed at stabilizing the zero transverse normal energy modes of H3-type flexural elements. The feasible numerical values for χ are given in section 2.3

2.2.4 H2ANS - Hierarchic $(2-D)$ Degenerated Assumed Natural Strain Element with Rotational Degrees-Of-Freedom

In this section we formulate a degenerated assumed strain shell element, which employs blending functions or Lagrangian basis for geometry mapping and Legendre polynomials for solution interpolation.

As a starting point, the displacement field is expressed in terms of mid-point translations $u_i^t(\xi_1,\ \xi_2)$ and mid-point rotations $\theta_\alpha(\xi_1,\xi_2)$ which are defined with respect to the fiber coordinate system:

$$\left\{ \begin{array}{c} u_1 \\ u_2 \\ u_3 \end{array} \right\}_{(\xi_1,\xi_2,\xi_3)} = \left\{ \begin{array}{c} u_1^t \\ u_2^t \\ u_3^t \end{array} \right\}_{(\xi_1,\xi_2)} + \frac{\xi_3}{2} \; [-t\mathbf{e}_2^f,\ t\mathbf{e}_1^f]_{(\xi_1,\xi_2)} \left\{ \begin{array}{c} \theta_1 \\ \theta_2 \end{array} \right\} \tag{2.18}$$

where the unit vectors $(\mathbf{e}_1^f,\ \mathbf{e}_2^f,\ \mathbf{e}_3^f)$ of the fiber coordinate system are defined as in [32].

For the purpose of discussion here the iso-parametric shell element discretization is viewed of consisting of the following two steps:

1. Evaluate the displacement field at the finite element nodes ($\xi_1 = \xi_1^A,\ \xi_2 = \xi_2^A$) using equation (2.18): $u_i^A(\xi_3) = u_i(\xi_1 = \xi_1^A, \xi_2 = \xi_2^A, \xi_3)$

2. Interpolate the displacement field using two-dimensional Lagrangian basis functions: $u_i(\xi_1, \xi_2, \xi_3) = N_{iA}^{lag}(\xi_1, \xi_2)\, u_i^A(\xi_3)$

The extension of this approach to the hierarchical p-method where degrees-of-freedom do not represent the value of the solution at a specific location within the element is not trivial. It involves sampling for the solution value at some arbitrary points, say, finite element nodes corresponding to Lagrangian interpolation followed by inverse mapping aimed at finding corresponding amplitudes of hierarchical modes. Unfortunately, for higher polynomial orders this will involve inversion of large element matrices, which may overshadow the benefit from a hierarchical degenerated shell element formulation.

In the present work we propose a different route by which the fields $u_i^t(\xi_1, \xi_2)$ and $\theta_\alpha(\xi_1, \xi_2)$ are directly discretized using Legendre polynomials. The resulting solution approximation takes the following form:

$$\left\{ \begin{array}{c} u_1 \\ u_2 \\ u_3 \end{array} \right\} = \sum_{A=1}^{NMDS} N_A(\xi_1, \xi_2) \left\{ \begin{array}{c} u_1^{t^A} \\ u_2^{t^A} \\ u_3^{t^A} \end{array} \right\} + \sum_{A=1}^{NMDS} N_A(\xi_1, \xi_2)\ \frac{\xi_3}{2}\ [-t\mathbf{e}_2^f,\ t\mathbf{e}_1^f]_{(\xi_1,\xi_2)} \left\{ \begin{array}{c} \theta_1^A \\ \theta_2^A \end{array} \right\} \tag{2.19}$$

Note that there is a fundamental difference between equation (2.19) and its iso-parametric counterpart. In the classical iso-parametric formulation the variable vector functions $\mathbf{e}_i^f(\xi_1,\ \xi_2)$ in equation (2.19) are replaced by a set of constant vectors $\mathbf{e}_i^f(\xi_1^A,\ \xi_2^A)$ representing the fiber coordinate system at the node A. The present formulation gives rise to an additional term in the displacement gradient

evaluation, resulting from the derivatives of $\mathbf{e}_i^f$:

$$\frac{\partial \mathbf{u}}{\partial \xi_\alpha} =$$

$$\sum_{A=1}^{NMDS} \frac{\partial N_A(\xi_1,\xi_2)}{\partial \xi_\alpha} \begin{Bmatrix} u_1^{t^A} \\ u_2^{t^A} \\ u_3^{t^A} \end{Bmatrix} + \sum_{A=1}^{NMDS} \left[\frac{\partial N_A(\xi_1,\xi_2)}{\partial \xi_\alpha} \frac{\xi_3}{2} [-t\mathbf{e}_2^f, t\mathbf{e}_1^f] + \Delta_A \right] \begin{Bmatrix} \theta_1^A \\ \theta_2^A \end{Bmatrix}$$

and

$$\Delta_A = N_A(\xi_1,\xi_2) \ \frac{\xi_3}{2} \ [-\frac{\partial (t\mathbf{e}_2^f)}{\partial \xi_\alpha}, \ \frac{\partial (t\mathbf{e}_1^f)}{\partial \xi_\alpha}] \tag{2.20}$$

where $\alpha \in [1,2]$. Derivatives of $((t\mathbf{e}_1^f), \ (t\mathbf{e}_2^f))$ are obtained by differentiation of appropriate mapping functions:

$$\frac{\partial t}{\partial \xi_i} = \frac{1}{t}[(\mathbf{X}^{top} - \mathbf{X}^{bot}) \cdot (\mathbf{X}^{top}_{,\xi_i} - \mathbf{X}^{bot}_{,\xi_i})] \tag{2.21}$$

$$\frac{\partial \mathbf{e}_3^f}{\partial \xi_i} = [\frac{1}{t} \ (\mathbf{X}^{top}_{,\xi_i} - \mathbf{X}^{bot}_{,\xi_i}) - \frac{(\mathbf{X}^{top} - \mathbf{X}^{bot})}{t^2} \ \frac{\partial t}{\partial \xi_i}] \tag{2.22}$$

and from the definition of fiber vectors:

$$\frac{\partial \mathbf{e}_2^f}{\partial \xi_i} =$$

$$\frac{1}{\| \ \mathbf{e}_3^f \times \mathbf{e}_j \ \|} (\frac{\partial \mathbf{e}_3^f}{\partial \xi_i} \times \mathbf{e}_j) - \frac{(\mathbf{e}_3^f \times \mathbf{e}_j)}{\| \ \mathbf{e}_3^f \times \mathbf{e}_j \ \|^3} \left\{ \mathbf{e}_3^f \cdot \frac{\partial \mathbf{e}_3^f}{\partial \xi_i} - (\frac{\partial \mathbf{e}_3^f}{\partial \xi_i} \cdot \mathbf{e}_j) \ (\mathbf{e}_3^f \cdot \mathbf{e}_j) \right\}$$

$$\frac{\partial \mathbf{e}_1^f}{\partial \xi_i} = (\mathbf{e}_2^f \times \frac{\partial \mathbf{e}_3^f}{\partial \xi_i}) + (\frac{\partial \mathbf{e}_3^f}{\partial \xi_i} \times \mathbf{e}_2^f) \tag{2.23}$$

Derivative w.r.t ξ_3 is given as:

$$\frac{\partial \mathbf{u}}{\partial \xi_3} = \sum_{A=1}^{NMDS} \left[N_A(\xi_1,\xi_2) \ \frac{1}{2}[-t\mathbf{e}_2^f, \ t\mathbf{e}_1^f] \right] \begin{Bmatrix} \theta_1^A \\ \theta_2^A \end{Bmatrix} \tag{2.24}$$

The assumed natural strain - displacement matrix for this element is formulated by projecting out parasitic modes as in equation (2.10).

2.3 Quadrature schemes for hierarchical systems.

In the h-version, all the terms in the stiffness matrix are integrated with the same number of quadrature points, which is dictated by the maximum polynomial order of the integrand. This scheme labeled here as the Uniform Gauss quadrature does not exploit the hierarchical structure of the stiffness matrix. To take advantage of the hierarchiality of the p-method, the scheme labeled as Hierarchic Gauss quadrature, integrates each matrix entry separately with a number of quadrature points depending on the polynomial order of each integrand. Hierarchic Gauss quadrature is optimal in terms of number of quadrature points but it drastically increases the number of function evaluations. Numerical experiments conducted by Hinnant [30] indicate that the benefit from reducing the number of function evaluations overshadows the cost savings obtained from reducing the number of quadrature points. A variant of the Uniform quadrature scheme, which is partially adapted to take advantage of the hierarchic structure of the stiffness matrix is labeled here as *Hierarchic Block* quadrature. By this technique sub-matrices corresponding to different polynomial orders are integrated with an integration rule corresponding to the maximum polynomial order of the appropriate block.

To obtain a nearly optimal performance in terms of CPU time a new quadrature scheme for rectangular and hexahedral elements, which exploits hierarchiality of the p-method and at the same time minimizes the number of function evaluations, is presented in the subsequent subsections.

2.3.1 Dot product integral decomposition

Consider the integral of the form $\int_\Omega g(\xi_1,\ \xi_2,\ \xi_3)\ h(\xi_1,\ \xi_2,\ \xi_3)\ d\Omega$ in the natural coordinate system such that $d\Omega = d\xi_1\ d\xi_2\ d\xi_3$ and $-1 \leq \xi_i \leq 1$. The integral under consideration may represent a typical stiffness matrix component, where g is a strain-displacement matrix component and h is a strain-displacement matrix term

multiplied by a jacobian and a corresponding component of the constitutive tensor. In the classical Gauss quadrature the integrand (gh) is implicitly curve fit with a polynomial and then integrated in a close form. It will be shown that for hierarchical systems it is more efficient to curve fit separately each term of the integrand (g and h) with orthonormal polynomial:

$$g = \sum_{i=0}^{n_1}\sum_{j=0}^{n_2}\sum_{k=0}^{n_3} \underbrace{a_{ijk}}_{a_I} \overbrace{\hat{P}_i(\xi_1)\,\hat{P}_j(\xi_2)\,\hat{P}_k(\xi_3)}^{\phi_I} \equiv \sum_{I=1}^{N} a_I \phi_I(\xi_1,\,\xi_2,\,\xi_3) \qquad (2.25)$$

$$h = \sum_{s=0}^{m_1}\sum_{t=0}^{m_2}\sum_{r=0}^{m_3} \underbrace{b_{str}}_{b_J} \overbrace{\hat{P}_s(\xi_1)\,\hat{P}_t(\xi_2)\,\hat{P}_r(\xi_3)}^{\phi_J} \equiv \sum_{J=1}^{M} b_J \phi_J(\xi_1,\,\xi_2,\,\xi_3) \qquad (2.26)$$

where $\hat{P}_i$ are normalized Legendre polynomials. For convenience, concise notation is introduced with $a_I \equiv a_{ijk}$, $\phi_I(\xi_1,\xi_2,\xi_3) \equiv \hat{P}_i(\xi_1)\hat{P}_j(\xi_2)\hat{P}_k(\xi_3)$, $N = (n_1+1)(n_2+1)(n_3+1)$ and $M = (m_1+1)(m_2+1)(m_3+1)$.

Using (2.25) and (2.26) the integral of $g\ h$ can be decomposed into a dot product of two vectors:

$$\begin{aligned}\int_\Omega g\ h\ d\Omega &= \sum_{i=0}^{l_1}\sum_{j=0}^{l_2}\sum_{k=0}^{l_3}\int_\Omega g\ \hat{P}_i(\xi_1)\hat{P}_j(\xi_2)\hat{P}_k(\xi_3)d\Omega \cdot \int_\Omega h\ \hat{P}_i(\xi_1)\hat{P}_j(\xi_2)\hat{P}_k(\xi_3)d\Omega \\ &= \sum_{I=1}^{L}\int_\Omega g\ \phi_I d\Omega \ \int_\Omega h\ \phi_I d\Omega \end{aligned} \qquad (2.27)$$

where $l_i = \min(n_i, m_i)$ and $L = (l_1+1)(l_2+1)(l_3+1)$.

Proof:

Substituting (2.25) and (2.26) into left hand side of (2.27) yields:

$$\begin{aligned}&\int_\Omega (gh)d\Omega = \\ &\quad \sum_{i=0}^{n_1}\sum_{j=0}^{n_2}\sum_{k=0}^{n_3}\sum_{s=0}^{m_1}\sum_{t=0}^{m_2}\sum_{r=0}^{m_3} a_{ijk}b_{rst} \\ &\quad \cdot \underbrace{\int_{-1}^{+1}\hat{P}_i(\xi_1)\hat{P}_r(\xi_1)d\xi_1}_{\delta_{ir}} \underbrace{\int_{-1}^{+1}\hat{P}_j(\xi_2)\hat{P}_s(\xi_2)d\xi_2}_{\delta_{js}} \underbrace{\int_{-1}^{+1}\hat{P}_k(\xi_3)\hat{P}_t(\xi_3)d\xi_3}_{\delta_{kt}} \\ &= \sum_{i=0}^{l_1}\sum_{j=0}^{l_2}\sum_{k=0}^{l_3} a_{ijk}\ b_{ijk} = \sum_{I=1}^{L} a_I\ b_I \end{aligned} \qquad (2.28)$$

Likewise, the right hand side of (2.27) gives :

$$\sum_{I=1}^{L} \int_{\Omega} g\phi_I d\Omega \int_{\Omega} h\phi_I d\Omega =$$

$$\sum_{i=0}^{l_1}\sum_{j=0}^{l_2}\sum_{k=0}^{l_3}\sum_{s=0}^{n_1}\sum_{t=0}^{n_2}\sum_{r=0}^{n_3}$$

$$\left(a_{str}\int_{-1}^{1}\hat{P}_i(\xi_1)\hat{P}_s(\xi_1)d\xi_1\int_{-1}^{1}\hat{P}_j(\xi_2)\hat{P}_t(\xi_2)d\xi_2\int_{-1}^{1}\hat{P}_k(\xi_3)\hat{P}_r(\xi_3)d\xi_3\right)$$

$$\cdot \sum_{p=0}^{m_1}\sum_{q=0}^{m_2}\sum_{v=0}^{m_3}\left(b_{pqv}\int_{-1}^{1}\hat{P}_i(\xi_1)\hat{P}_p(\xi_1)d\xi_1\int_{-1}^{1}\hat{P}_j(\xi_2)\hat{P}_q(\xi_2)d\xi_2\int_{-1}^{1}\hat{P}_k(\xi_3)\hat{P}_v(\xi_3)d\xi_3\right)$$

$$= \sum_{i=0}^{l_1}\sum_{j=0}^{l_2}\sum_{k=0}^{l_3} a_{ijk}\, b_{ijk} = \sum_{I=1}^{L} a_I\, b_I \tag{2.29}$$

The dot product of integral decomposition was originally proposed by Hinnant [30]. The quadrature based on dot product integral decomposition is optimal in terms of number of integrand evaluations for hierarchical systems. To clarify this point consider a one-dimensional case. Let $\mathbf{g} = \{g_i\}$ and $\mathbf{h} = \{h_j\}$ be vectors whose terms represent the hierarchical sequence with increasing polynomial order, where subscripts on g and h denote the polynomial orders and $i,\ j \in [0, p]$. In evaluating integrals of the form $G_{ik} = \int g_i\ \hat{P}_k(\xi)d\xi$ and $H_{jl} = \int h_j\ \hat{P}_l(\xi)d\xi$, where $k \in [0, i]$ and $l \in [0, j]$, the number of function evaluations for (g_i, h_j) is $(i+1)$ and $(j+1)$, respectively. Thus the total number of function evaluations for computing all the integrals of the form G_{ik} and H_{jl} is $(p+1)(p+2)$ as opposed to $2(p+1)^2$ for uniform quadrature. It can be shown that this estimate grows exponentially with the increase in the number of space dimensions.

The major drawback of Dot product integral decomposition is the lack of symmetry, which leads to:

1. Non-symmetric stiffness matrix if $\mathbf{g}$ and $\mathbf{h}$ are of different polynomial orders (such a situation may arise in the case of material or geometric nonlinearity).

2. Redundancy in evaluating each of the two integrals G_{ik} and H_{jl}, which, except for the term involving constitutive tensor, should be identical.

2.3.2 Symmetric dot product integral decomposition for shells.

In this section a variant of Dot product integral decomposition which preserves the symmetry of the stiffness matrix is presented. Consider a typical stiffness term given by

$$k_{AB} = \int_{\Omega} \underbrace{\mathbf{B}_A^T \, \mathbf{D} \, \mathbf{B}_B \, J}_{\mathbf{g}_A \, \mathbf{h}_B} \, d\Omega \tag{2.30}$$

In an attempt to obtain a symmetric dot product integral decomposition, we decompose the integrand $(\mathbf{g}_A \, \mathbf{h}_B)$ as follows:

$$\mathbf{h}_A^T \; = \; \mathbf{g}_A \; = \; \mathbf{B}_A^T \, \mathbf{L} \, J^{1/2} \tag{2.31}$$

where $\mathbf{L}$ is a lower triangular Cholesky factor of the constitutive matrix $\mathbf{D}$. The resulting stiffness matrix is given by

$$k_{AB} = \sum_{I=1}^{L} \int_{\Omega} \left(\mathbf{L}^T \, \mathbf{B}_A \right)^T \, \phi_I \, J^{1/2} d\Omega \;\; \int_{\Omega} \mathbf{L}^T \mathbf{B}_B \, \phi_I \, J^{1/2} d\Omega \tag{2.32}$$

Each of the integrals is integrated using Gauss quadrature. The number of quadrature points as well as the maximum polynomial order of the interpolating Legendre polynomials in each direction depends on how well the integrand $\mathbf{L}^T \; \mathbf{B}_A \; J^{1/2}$ is approximable by polynomials and what is their polynomial order. We refer to this integration scheme as Symmetric Dot Product (SDP) Gauss quadrature.

In case when the constitutive tensor $\mathbf{D}$ is not positive definite an alternative integrand decomposition is employed. Let

$$\mathbf{g}_A \; = \; \mathbf{B}_A^T \, J^{1/2} \quad \mathbf{h}_B \; = \; \mathbf{D} \, J^{1/2} \, \mathbf{B}_B \tag{2.33}$$

yielding

$$k_{AB} = \sum_{I=1}^{L} \int_{\Omega} \mathbf{B}_A^T \, J^{1/2} \, \phi_I d\Omega \;\; \int_{\Omega} \overbrace{\mathbf{D} \, \phi_I}^{\mathbf{g}_I} \underbrace{\mathbf{B}_B \, J^{1/2}}_{\mathbf{h}_B} d\Omega \tag{2.34}$$

and further dot product integral decomposition of the second term in (2.34) yields

the following symmetric form:

$$k_{AB} = \sum_{I=1}^{L}\sum_{J=1}^{L} \int_{\Omega} \left(\mathbf{B}_A^T\, J^{1/2}\right)\, \phi_I d\Omega \cdot \underbrace{\int_{\Omega} (\mathbf{D})\, \phi_I \phi_J d\Omega}_{\bar{\mathbf{D}}_{IJ}} \cdot \int_{\Omega} \left(\mathbf{B}_B\, J^{1/2}\right)\, \phi_J d\Omega \qquad (2.35)$$

Note that if the constitutive tensor is constant, $\bar{\mathbf{D}}_{IJ} = \int_{\Omega}(\mathbf{D})\, \phi_I \phi_J d\Omega = \mathbf{D}\delta_{IJ}$ reducing equation(2.35) to

$$k_{AB} = \sum_{I=1}^{L} \int_{\Omega} (\mathbf{B}_A^T J^{1/2})\, \phi_I d\Omega \cdot \mathbf{D} \cdot \int_{\Omega} (\mathbf{B}_B J^{1/2})\, \phi_I d\Omega \qquad (2.36)$$

It can be easily shown that if $\mathbf{D} \neq$ constant, stiffness matrix evaluations by means of equation (2.35) is more computationally intensive because of the double summation involved. Nevertheless, the triple integral decomposition (2.35) might be useful in the following two scenarios:

- Thick laminated composite shells with multiple layers and variable jacobian through the thickness.
- Small deformation nonlinear material analysis, where the first term in (2.35) can be computed only once and then reused in the nonlinear incremental iterative process.

2.3.3 Symmetric dot product integral decomposition for assumed strain shell elements.

In general, implementation of assumed natural strain element is computationally more expensive than its displacement based counterpart, mainly due to the additional cost involved in the interpolation of natural strain components from the reduced quadrature points to the regular quadrature points. In order to perform a more realistic comparison based on the CPU time we now focus on the development of efficient quadrature scheme for hierarchical assumed strain shell elements.

Consider a typical stiffness matrix term for the assumed natural strain shell element:

$$k_{AB} = \int_{\Omega} \bar{\mathbf{B}}_A^{nat^T} \mathbf{T}^T \mathbf{D}\ \mathbf{T}\ \bar{\mathbf{B}}_A^{nat} J d\Omega \tag{2.37}$$

where $\mathbf{B}_A^{nat}$ is a matrix relating displacement degrees of freedom to natural strain tensor components; $\bar{\mathbf{B}}_A^{nat}$ is an assumed natural strain-displacement matrix, which is obtained by selectively interpolating natural strain components between reduced quadrature points [47]. $\mathbf{T}$ is the natural-to-material strain transformation matrix.

Decomposing the integrand $(\mathbf{g}_A\ \mathbf{h}_B)$ as

$$\mathbf{h}_A^T\ =\ \mathbf{g}_A\ =\ \left(\mathbf{L}^T\mathbf{T}\ \bar{\mathbf{B}}_A^{nat}\right)^T J^{1/2} \tag{2.38}$$

yields the following symmetric form:

$$k_{AB} = \sum_{I=1}^{L} \int_{\Omega} \left(\mathbf{L}^T\mathbf{T}\bar{\mathbf{B}}_A^{nat}\right)^T\ \phi_I\ J^{1/2} d\Omega \quad \int_{\Omega} \mathbf{L}^T\mathbf{T}\ \bar{\mathbf{B}}_A^{nat}\ \phi_I\ J^{1/2} d\Omega \tag{2.39}$$

To further enhance the computational efficiency of assumed strain shell elements we will attempt to obtain the very same effect of selective polynomial order reduction by simply projecting selected components of the material strain-displacement matrix $\mathbf{B}^{mat} = \mathbf{T}\mathbf{B}^{nat} = \hat{\mathbf{T}}\mathbf{B}^{global}$ onto a reduced space spanned by Legendre polynomials. In the above equation $\hat{\mathbf{T}}$ is the global-to-material strain transformation matrix. By this technique the typical stiffness matrix term is given by:

$$k_{AB} = \sum_{i=1}^{l_1}\sum_{j=1}^{l_2}\sum_{k=1}^{l_3} \int_{\Omega} \left(\begin{bmatrix} b_{A11}^{mat} \hat{\hat{P}}_i(\xi_1)\hat{P}_j(\xi_2)\hat{P}_k(\xi_3) \\ b_{A22}^{mat} \hat{P}_i(\xi_1)\hat{\hat{P}}_j(\xi_2)\hat{P}_k(\xi_3) \\ b_{A33}^{mat} \hat{P}_i(\xi_1)\hat{P}_j(\xi_2)\hat{\hat{P}}_k(\xi_3) \\ b_{A23}^{mat} \hat{P}_i(\xi_1)\hat{\hat{P}}_j(\xi_2)\hat{\hat{P}}_k(\xi_3) \\ b_{A12}^{mat} \hat{\hat{P}}_i(\xi_1)\hat{\hat{P}}_j(\xi_2)\hat{P}_k(\xi_3) \\ b_{A13}^{mat} \hat{\hat{P}}_i(\xi_1)\hat{P}_j(\xi_2)\hat{\hat{P}}_k(\xi_3) \end{bmatrix} \right)^T \mathbf{L}\sqrt{J} d\Omega$$

$$\cdot \int_{\Omega} \mathbf{L}^T \begin{bmatrix} b_{B11}^{mat} \hat{\hat{P}}_i(\xi_1)\hat{P}_j(\xi_2)\hat{P}_k(\xi_3) \\ b_{B22}^{mat} \hat{P}_i(\xi_1)\hat{\hat{P}}_j(\xi_2)\hat{P}_k(\xi_3) \\ b_{B33}^{mat} \hat{P}_i(\xi_1)\hat{P}_j(\xi_2)\hat{\hat{P}}_k(\xi_3) \\ b_{B23}^{mat} \hat{P}_i(\xi_1)\hat{\hat{P}}_j(\xi_2)\hat{\hat{P}}_k(\xi_3) \\ b_{B12}^{mat} \hat{\hat{P}}_i(\xi_1)\hat{\hat{P}}_j(\xi_2)\hat{P}_k(\xi_3) \\ b_{B13}^{mat} \hat{\hat{P}}_i(\xi_1)\hat{P}_j(\xi_2)\hat{\hat{P}}_k(\xi_3) \end{bmatrix} \sqrt{J} d\Omega \tag{2.40}$$

where

$$\hat{\hat{P}}_m(\xi_t) = \begin{cases} \hat{P}_m(\xi_t) & m < l_t \\ 0 & m = l_t \end{cases} \tag{2.41}$$

The polynomial order of ϕ_I in equation (2.40) is chosen to selectively project out higher order terms from the strain field. Equation (2.41) is consistent with equation (2.10).

The shell element formulation based on above equations (2.40) differs from HANS type element in the following two respects:

- Eliminates the need for explicit strain interpolation between reduced quadrature points.
- Provides selective reduced polynomial order reduction in material strain tensor components as opposed to reduction in natural strain tensor components. Hereafter we will refer to such element formulation as Hierarchical Assumed Material Strain shell element or simply HAMS.

2.4 Numerical Experiments

Our numerical experimentation agenda includes investigation of various

- element types
- quadrature schemes

A comparison of various quadrature schemes is a difficult task since they provide different results and their performance is implementation dependent. Focusing primarily on the computational efficiency aspects and assuming that each quadrature scheme has been implemented with the optimal efficiency we comprise the following basis for comparison : Solution accuracy (measured in terms of error in the energy norm) versus CPU time required to form a stiffness matrix of the single element. We will conduct numerical examples for two problems quarter of each modeled with a single element: Pinched Cylinder (constant in-plane jacobian) and a frustum of a cone (variable in-plane jacobian). All the experiments are conducted on a Sun Sparc 10 workstation with 128 MB internal memory.

The problems considered are,

- The *Pinched Cylinder* with rigid end diaphragms subjected to strip load (with the width of strip equal to thickness of the shell) as shown in Figure 2.1. Due to symmetry only one eighth of the problem is solved.

- Frustum of a cone with rigid end diaphragms under self weight as shown in Figure 2.1.

Figures 2.1-2.2 show the performance of two types of shell elements: H2-type (degenerated hierarchic shell with 5 dofs per mode) in Figure 2.1, H3R-type (hierarchic shell element with $q = 1$ and reduced stiffness in transverse normal direction) in Figure 2.2.

For the problems considered the stabilization parameter χ equal to $\frac{h}{l}$, where h and l are the smallest and largest dimensions of the shell element is used for **H3R**-type elements. The range of $\chi = 10^{-1} - 10^{-3}$ the element behavior was found to be close to that of the degenerated element with rotational degrees-of-freedom.

Each figure contains six plots :

- (o) HSOL-SDP corresponding to Symmetric Dot Product Gauss quadrature for displacement based element.
- ($\Box$) HANS-SDP corresponding to Symmetric Dot Product Gauss quadrature for assumed natural strain element.
- ($\triangle$) HAMS-SDP corresponding to Symmetric Dot Product Gauss quadrature for assumed material strain element.
- (+) HSOL-UNIF corresponding to Uniform Gauss quadrature scheme for displacement based element.
- (*) HANS-UNIF corresponding to Uniform Gauss quadrature scheme for assumed natural strain element.
- (x) HSOL-HBLOCK corresponding to Hierarchic Block Gauss quadrature scheme for displacement based element.

To preserve hierarchical structure of the stiffness matrix the displacement based shell element has been integrated to accommodate for highly varying metric tensor components $\frac{\partial x_i}{\partial \xi_j}$. For numerical examples considered, the number of integration points for Block Gauss quadrature was selected as $p^{max}+3$ in inplane direction and $q^{max}+1$ in transverse direction, where p^{max} and q^{max} are the maximum polynomial orders of the corresponding block in inplane and transverse directions, respectively. These integration orders ensure that the lower order blocks are integrated with the same accuracy and the higher order coupling stiffness terms.

Similarly, for SDP-Gauss quadrature applied to displacement based elements, the order of interpolating Legendre polynomials $\hat{P}_i(\xi_i)$ is selected as $(p_i + l)$ $i \in [1,2]$ in inplane directions and $(q + m)$ in transverse direction. The corresponding number of integration points are $(p_i + l + 1)$ and $(q + m + 1)$ in inplane and transverse directions, respectively, where $(p_1,\ p_2)$ are the polynomial orders of the integrand in

inplane directions (ξ_1, ξ_2) and q is the polynomial order in transverse direction (ξ_3). Selection of integers l and m is dictated by the variation of the metric tensor $\frac{\partial x_i}{\partial \xi_j}$ in inplane and transverse directions respectively. For example, in case of constant inplane jacobian (the pinched cylinder) we used $l = 1, m = 0$ and $l = 2, m = 1$ for the case of variable inplane jacobian (frustum of the cone).

In case of HAMS elements the order of interpolating Legendre polynomials is selected such that their polynomial order does not exceed the maximum polynomial order of the basis functions to ensure effectivity of selective polynomial order reduction. On the other hand for lower order blocks the polynomial order for Legendre polynomials is selected the same as for displacement based elements to partially preserve hierarchiality. Thus the order of Legendre polynomials for HAMS element is defined using the following rule:

- For a given integrand with polynomial orders (p_i, q)

$$\text{The inplane polynomial order of } \hat{P}_i = \begin{cases} p_i + l & \text{if } p_i + l < p^{max} \\ p^{max} & \text{otherwise} \end{cases} \tag{2.42}$$

  The order of Legendre polynomials in transverse direction is selected as $q + m$.

- The number of inplane integration points is selected as $p^{max} + 1$, and $q^{max} + 1$ in transverse direction.

It is evident from figures(2.1-2.2) that among the displacement (HSOL) based elements, SDP and HBLOCK quadrature schemes are computationally more efficient than the uniform (UNIF) quadrature. The difference between HSOL-SDP and HSOL-HBLOCK is not significant and it can be deduced that for displacement based elements SDP and HBLOCK have a comparable performance. It is apparent from the figures(2.1-2.2) that HAMS-SDP has higher computational efficiency than HANS-SDP and HANS-UNIF. Numerical tests considered indicate that the

assumed strain element HAMS-SDP has faster rate of convergence than its displacement based counterpart HSOL-SDP, and is more efficient than its uniform quadrature counterpart HANS-UNIF.

The effect of mesh distortion for the cantilever plate is illustrated in Figure 2.3. It is evident that mesh distortion degrades the performance of both p and h-versions of finite element method, although the p-version was found to be significantly less sensitive to mesh distortion.

Figure 2.4 depicts the rate of convergence of various elements for the pinched cylinder and the 3 cylinder assembly problems. Percentage relative error in the energy norm is plotted versus the total CPU time required to solve the problem. SDP quadrature scheme for integration of element stiffness matrices and the best solution procedure for a given polynomial order are adopted for all elements. It is evident that H2AMS and H3RAMS have the best performance in degenerated and 3-D categories, respectively.

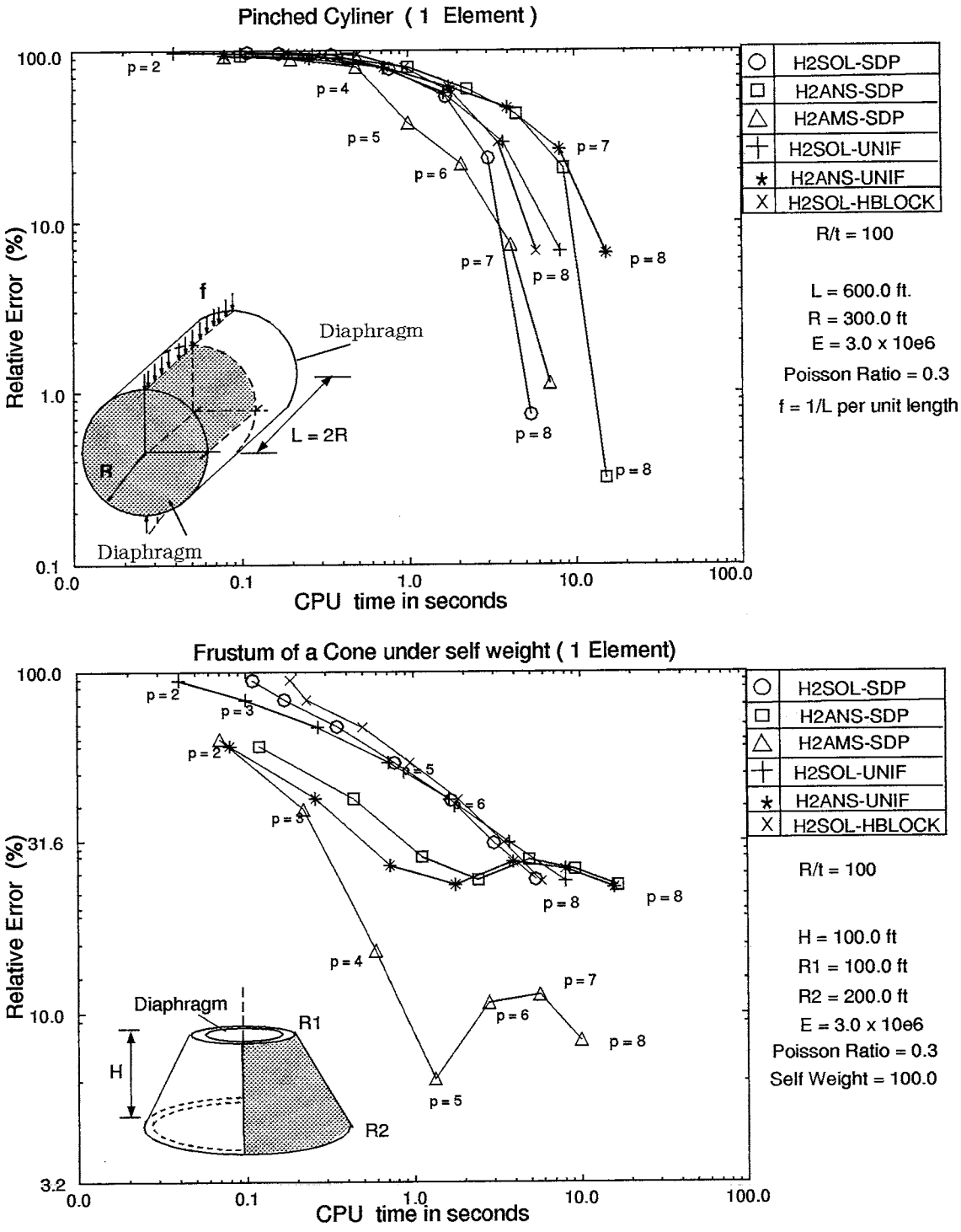


Figure 2.1: Comparison of Quadrature Schemes for H2-type Elements

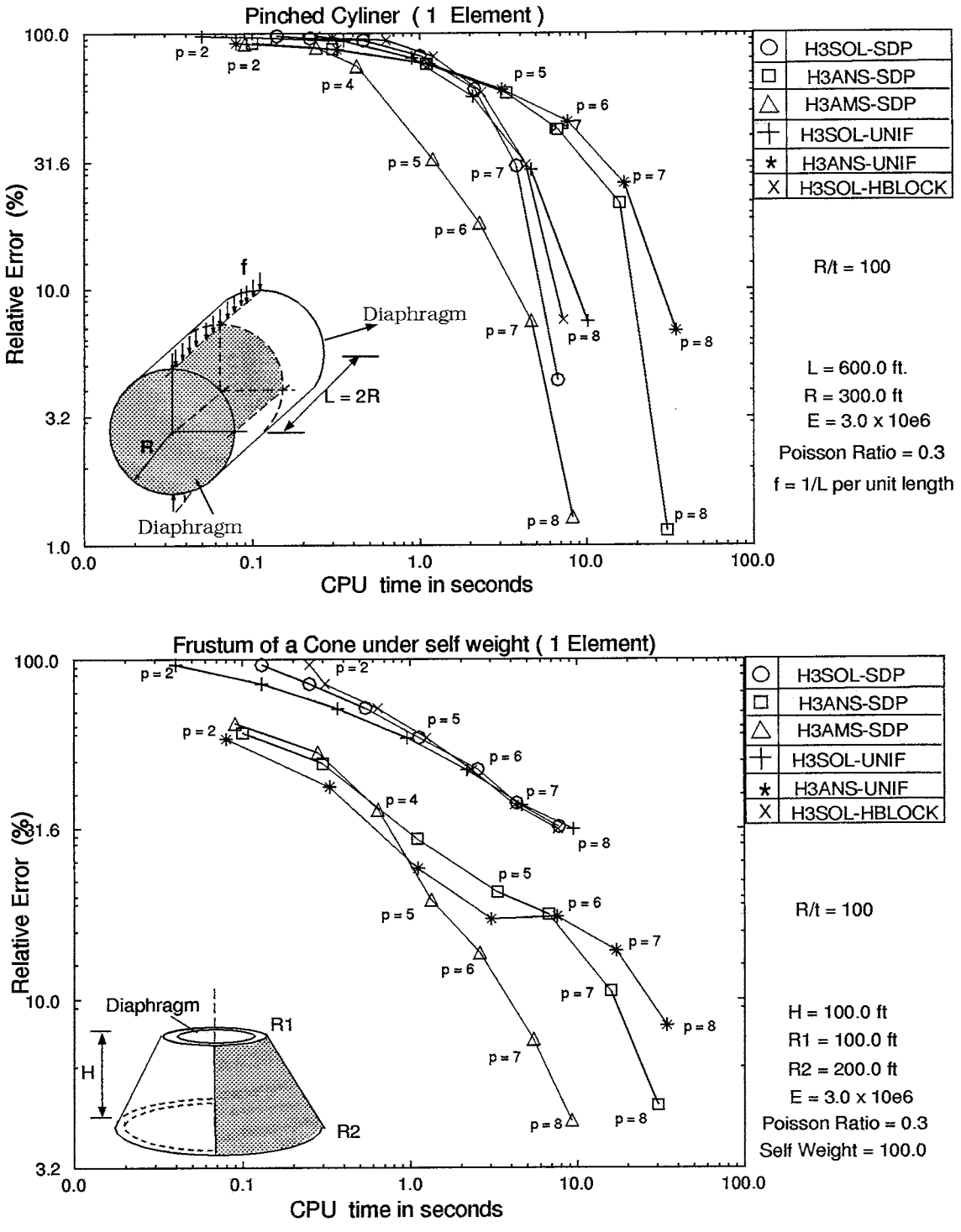
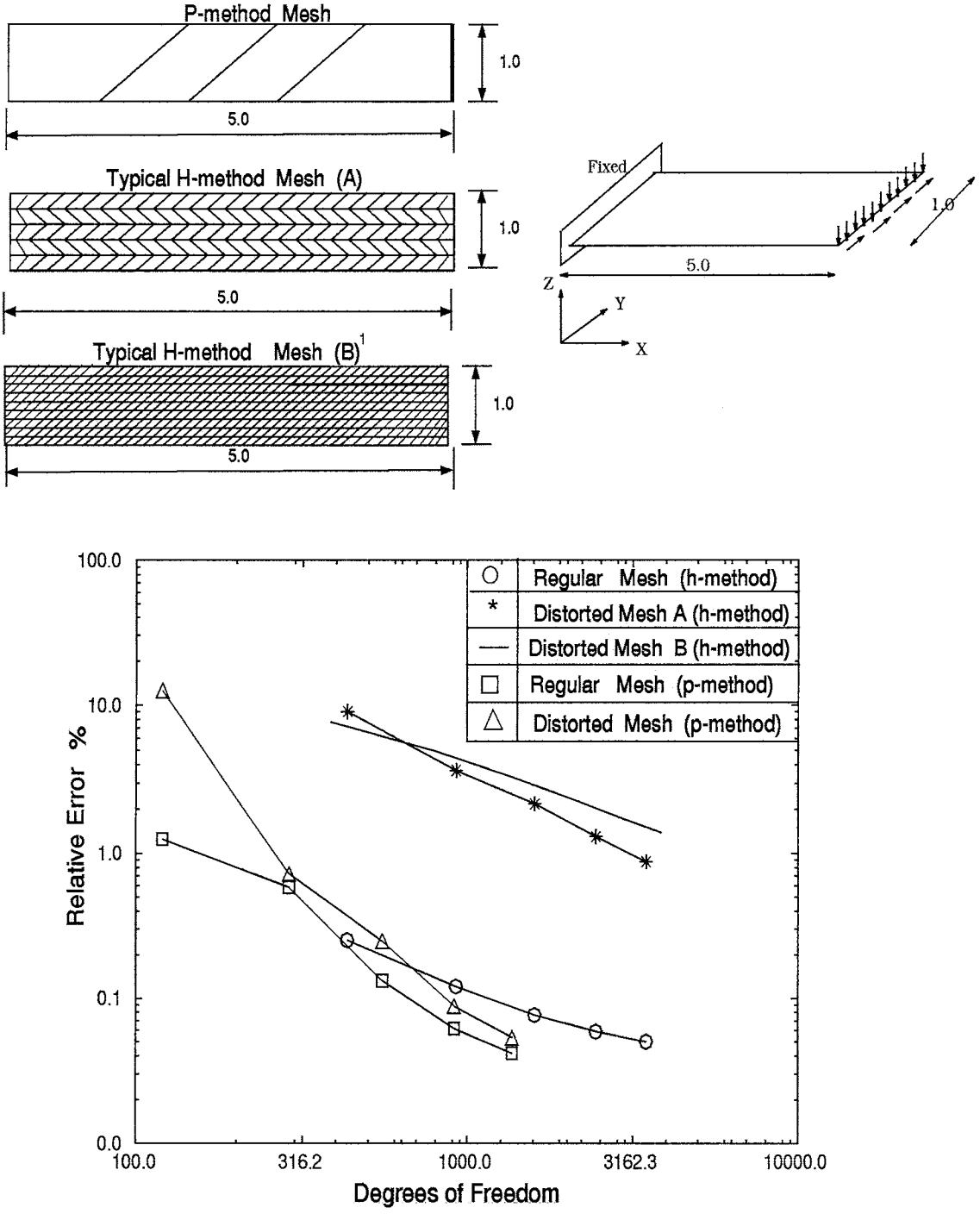


Figure 2.2: Comparison of Quadrature Schemes for H3R-type Elements



1. suggested by T. Belytschko Northwestern Univ.

Figure 2.3: Effect of Mesh Distortion on h and p versions of Finite Element Methods

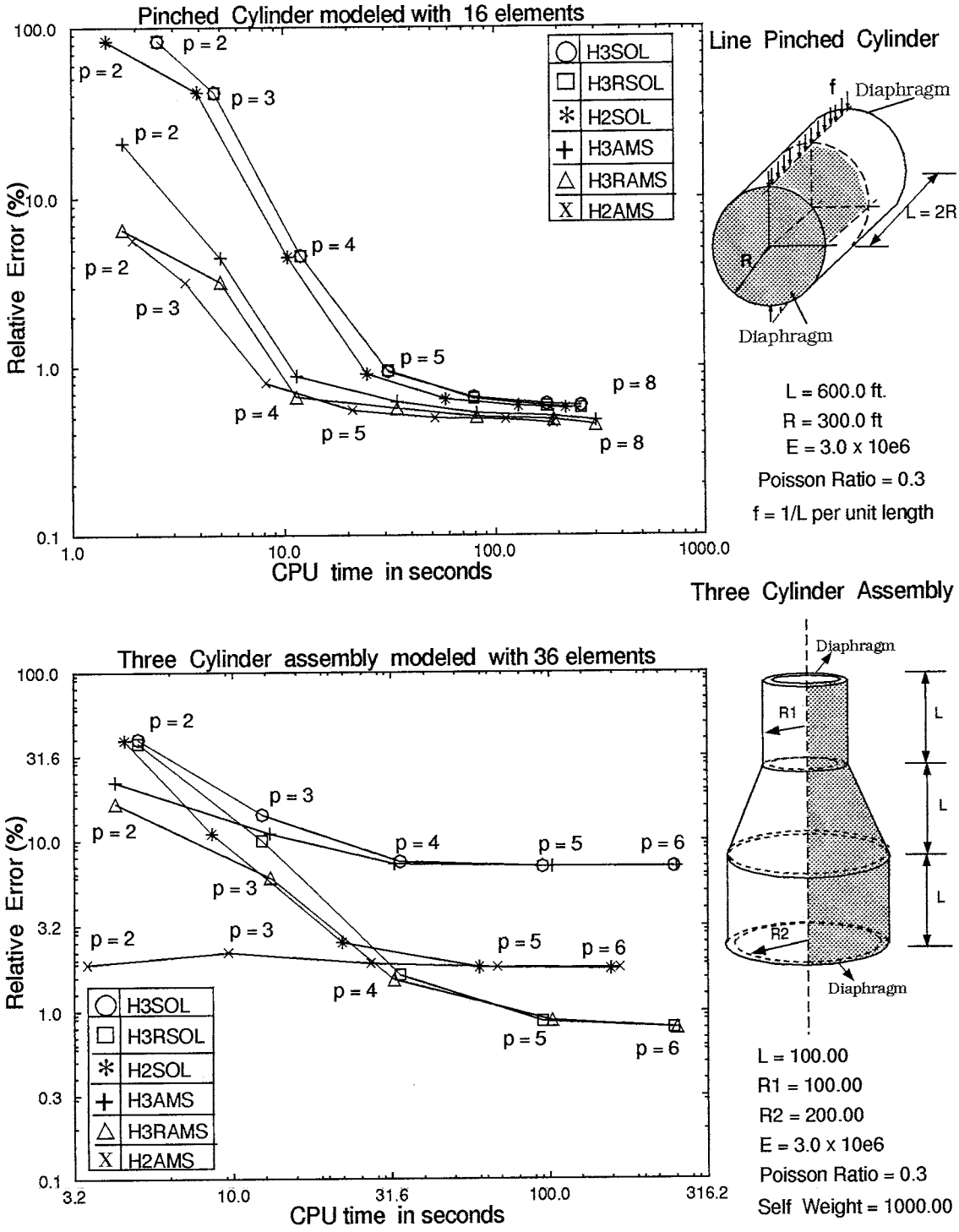


Figure 2.4: Rate of Convergence in terms of CPU time for various elements

CHAPTER 3

Hierarchic Global-Local Method for Laminated Composites

3.1 Introduction

The work presented in this chapter is an extension of the *s*-version of the finite element method [15] [16] [18] to laminated plates and shells. In this investigation we emphasize the utility of the mesh superposition technique to:

- Identify the location of the critical regions where DL model is needed using Dimensional Reduction Error (DRE) indicators.
- Accurately and efficiently predict both local and global effects using assumed strain formulation and selective polynomial order escalation.

The outline of the chapter is as follows: In section 3.2 the *s*-method is introduced in the context of laminated plates and shells. Assumed strain formulation and SDP quadrature scheme in the context of laminated composites are briefly discussed in section 3.3. Dimensional Reduction Error indicators and quality control techniques are described in section 3.4. Numerical examples conclude the chapter.

3.2 Mesh superposition for laminated composites

Consider a heterogeneous medium on domain Ω with boundary Γ as shown in Figure 3.1. The boundary Γ is decomposed as $\Gamma = \Gamma_u \cup \Gamma_t$, where Γ_u is the boundary with prescribed displacements and Γ_t with prescribed traction. The critical region to be superimposed by a local mesh is denoted by Ω^L, $\Omega^L \subset \Omega$. Domains Ω and Ω^L are subdivided independently into element domains such that $\cup\Omega_e^G = \Omega$ and $\cup\Omega_e^L = \Omega^L$. Let Γ^{GL} be the boundary between the two meshes such that $\Gamma \cap \Gamma^{GL} = \emptyset$. Let $\mathbf{u^G}$ be the global displacement field defined on Ω and $\mathbf{u^L}$ be

the local displacement field defined on the local region Ω^L. The total displacement field is constructed by superposition:

$$\mathbf{u} = \mathbf{u^G} + \mathbf{u^L} \tag{3.1}$$

In the case of laminated plates and shells, the global displacement field $\mathbf{u}^G$ is represented by the Equivalent Single Layer (ESL) model. The ESL model on Ω^G is discretized using a hierarchic degenerated plate/shell finite elements.

In the notation employed in this chapter, indices $(1,2)$ correspond to inplane coordinates and index '3' corresponds to the transverse direction as shown in Figure 3.1. The Greek subscripts represent the inplane spatial components (1,2) only. The lower case Latin subscripts indicate spatial components (1,2,3) except for subscripts a, b and e which are reserved for element numbers. Uppercase subscripts indicate degrees of freedom. Summation convention is employed for repeated indices unless otherwise specified.

The displacement field $\mathbf{u}^G$ is expressed in terms of mid-point translations $u_i^t(\xi_1,\ \xi_2)$ and mid-point rotations $\theta_\alpha(\xi_1,\xi_2)$ which are defined with respect to the fiber coordinate system as:

$$\left\{ \begin{array}{c} u_1^G \\ u_2^G \\ u_3^G \end{array} \right\} = \sum_{A=1}^{NMDS} N_A^G(\xi_1,\xi_2) \left\{ \begin{array}{c} u_{1A}^t \\ u_{2A}^t \\ u_{3A}^t \end{array} \right\} + \sum_{A=1}^{NMDS} N_A^G(\xi_1,\xi_2) \frac{\xi_3}{2} [-t\mathbf{e}_2^f, t\mathbf{e}_1^f]_{(\xi_1,\xi_2)} \left\{ \begin{array}{c} \theta_{1A} \\ \theta_{2A} \end{array} \right\} \tag{3.2}$$

Where $\{N_A^G\}$ are the hierarchic basis of the interpolation space $S^G(\Omega^G, p^G, q)$ (defined as in [54]), p^G is the maximum inplane polynomial order, q the polynomial order in transverse direction and $\Omega^G = \Omega$. NMDS are the number of hierarchic modes in the superimposed mesh. Note that there is a fundamental difference between equation (3.2) and its iso-parametric counterpart. In the classical iso-parametric formulation the variable vector functions $\mathbf{e}_i^f(\xi_1,\ \xi_2)$ in equation (3.2) are replaced by a set of constant vectors $\mathbf{e}_i^f(\xi_1^A,\ \xi_2^A)$ representing the fiber coordinate system at

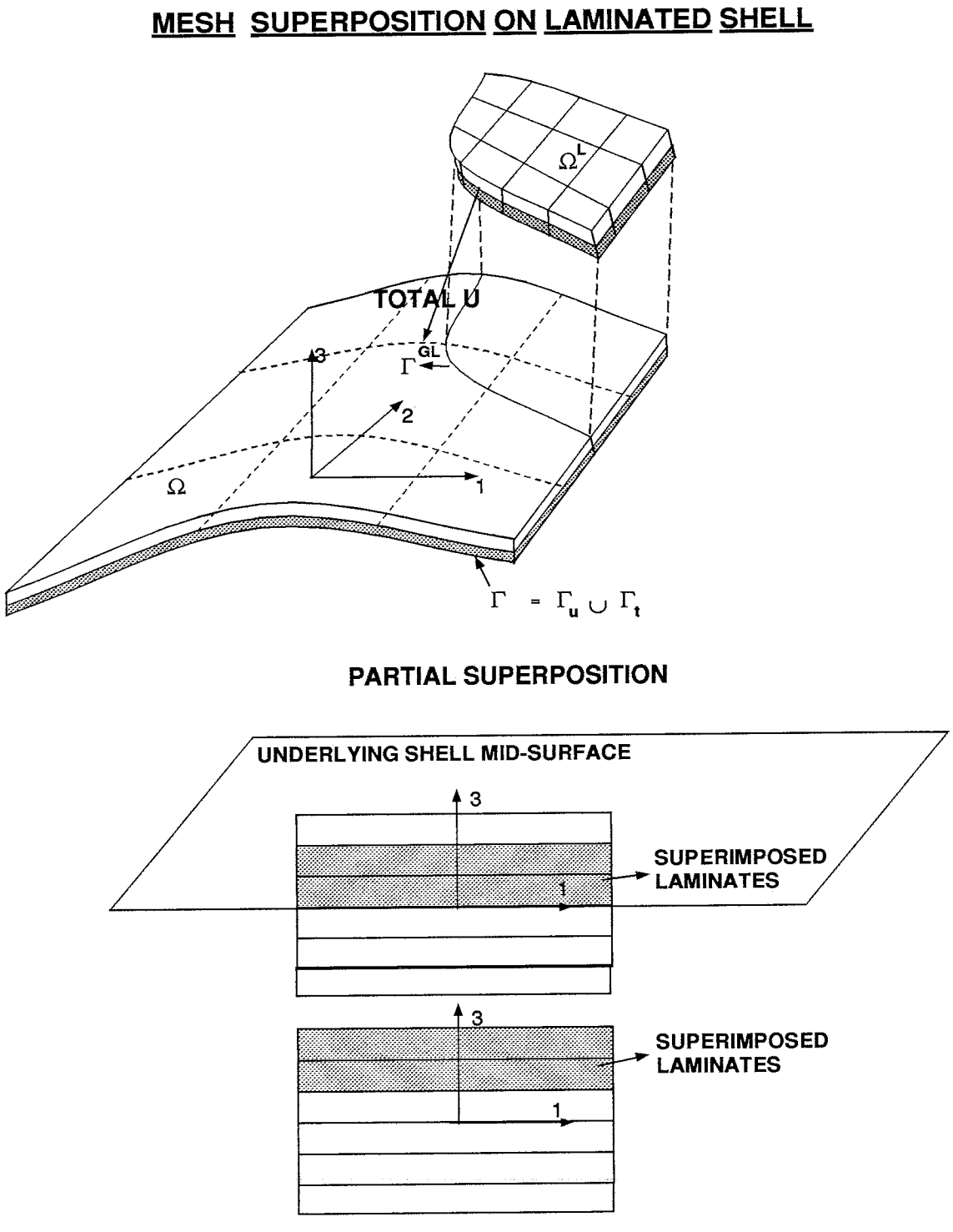


Figure 3.1: Mesh Superposition on Laminated Shell

the node A. The present formulation gives rise to an additional term in the displacement gradient evaluation, resulting from the derivatives of $\mathbf{e}_i^f$. For a detailed formulation of hierarchic degenerated plate/shell element see Chapter 2.

To resolve the localized phenomena a Discrete Layer model is superimposed in the critical region Ω^L. The DL model is discretized in terms of 3D hierarchical C^0 continuous shape functions $\{N_i^L\}$, spanning the interpolation space $S^L(\Omega^L, p^L)$ (defined in [54]), where p^L is the maximum polynomial order. Thus the local displacement field is expressed as:

$$u_m^L = \sum_{A=1}^{NMDS} N_{mA}^L a_A \tag{3.3}$$

where $\mathbf{a}$ is the displacement vector representing the amplitudes of the hierarchic modes in the superimposed mesh.

To satisfy the C^0 displacement compatibility between the global (ESL) and local (DL) meshes, the following homogeneous boundary conditions are imposed on the local mesh at the boundary between the two meshes (Γ^{GL}):

$$u_\alpha^L = 0 \ \ on \ \ \Gamma^{GL} \alpha \in [1,2] \tag{3.4}$$

where α corresponds to the inplane coordinates of the shell. Note that the local displacement field is not constrained in the transverse (X_3) direction on Γ^{GL} as the underlying shell model has no extension mode in this direction. These constraints are equivalent to the so called telescopic constraints in finite element jargon.

The inhomogeneous displacement boundary conditions $\mathbf{u}^P$ on Γ_u can be approximated with the global mesh, although better resolution of prescribed fields can be obtained by prolongating the remainder of the field

$$\mathbf{u}^L = I^L(\mathbf{u}^P - \mathbf{u}^{PG}) \ \ on \ \ \Gamma_u \tag{3.5}$$

to the local mesh [15] [16]. In (3.5) I^L is the prolongation operator from the local to global mesh and $\mathbf{u}^{PG} = \mathbf{u}^G$ on Γ_u.

To demonstrate the basic structure of the discrete equations, a linear elastostatics problem is considered. The discrete equations are obtained using Galerkin approximation of the weak form which states:

Given the spaces,

$$\begin{aligned}
U^G &= \{\mathbf{u}^G | \mathbf{u}^G \in S^G(\Omega^G, p^G, q, l^G);\ \mathbf{u}^G \in C^0(\Omega^G);\ \ \mathbf{u}^G = \mathbf{u}^{PG}\ on\ \Gamma_u\} \\
W^G &= \{\mathbf{w}^G | \mathbf{w}^G \in S^G(\Omega^G, p^G, q, l^G);\ \mathbf{w}^G \in C^0(\Omega^G);\ \ \mathbf{w}^G = 0\ on\ \Gamma_u\} \\
U^L &= \{\mathbf{u}^L | \mathbf{u}^L \in S^L(\Omega^L, p^L, l^L); \mathbf{u}^L \in C^0(\Omega^L); \\
\mathbf{u}^L &= I^L(\mathbf{u}^P - \mathbf{u}^{PG})\ on\ \Gamma_u; u^L_\alpha = 0\ on\ \Gamma^{GL}\} \\
W^L &= \{\mathbf{w}^L | \mathbf{w}^L \in S^L(\Omega^L, p^L, l^L);\ \mathbf{w}^L \in C^0(\Omega^L); \\
\mathbf{w}^L &= 0\ on\ \Gamma_u;\ \ w^L_\alpha = 0\ on\ \Gamma^{GL}\}
\end{aligned} \tag{3.6}$$

where $\alpha \in [1,2]$ and $i \in [1,3]$.

Find

$$\mathbf{u}^G \in U^G,\ \ \mathbf{u}^L \in U^L, \mathbf{u} = \mathbf{u}^G + \mathbf{u}^L \tag{3.7}$$

such that for all $\mathbf{w}^G \in W^G,\ \ \mathbf{w}^L \in W^L, \mathbf{w} = \mathbf{w}^G + \mathbf{w}^L$

$$a(\mathbf{w}^G + \mathbf{w}^L,\ \mathbf{u}^G + \mathbf{u}^L)_\Omega = (\mathbf{w}^G + \mathbf{w}^L,\ b_i)_\Omega + (\mathbf{w}^G + \mathbf{w}^L,\ t_i)_\Gamma \tag{3.8}$$

where S^G is the aforementioned interpolation space for the global domain Ω^G, and l^G is the list of active degrees of freedom to be determined in the adaptive process on the basis of the contribution of the corresponding basis functions towards reduction of discretization errors (see section 3.4). Similarly S^L is the interpolation space for the local domain Ω^L and l^L is the corresponding list of active degrees of freedom determined adaptively. b_i and t_i are prescribed body and traction forces respectively. $a(\cdot,\cdot)$, and $(\cdot,\cdot)$ are bilinear symmetric forms defined by

$$a(\mathbf{P}, \mathbf{Q}) = \int_\Omega P_{(i,j)} D_{ijkl} Q_{(i,j)} d\Omega \tag{3.9}$$

$$(\mathbf{P}, \mathbf{Q})_\Gamma = \int_\Gamma P_i\ Q_i d\Gamma \tag{3.10}$$

$$(\mathbf{P},\mathbf{Q})_\Omega = \int_\Omega P_i\, Q_i d\Omega \tag{3.11}$$

D_{ijkl} is the constitutive tensor. The parenthesis about lower case subscripts designate the symmetric part of a second order tensor.

Discrete equations are obtained by substituting interpolants (3.2,3.3) for the both trial and test functions into the weak form (3.8) and requiring arbitrariness of the local and global variations. The structure of the resulting equation is given as:

$$\mathbf{K}\,\mathbf{d} = \mathbf{f} \tag{3.12}$$

where

$$\mathbf{K} = \begin{bmatrix} K^G_{AD} & K^C_{AE} \\ K^C_{BD} & K^L_{BE} \end{bmatrix} \quad \mathbf{d} = \begin{Bmatrix} d_D \\ a_E \end{Bmatrix} \quad \mathbf{f} = \begin{Bmatrix} f^{ext}_A \\ h^{ext}_B \end{Bmatrix} \tag{3.13}$$

and,

$$K^G_{AD} = a(\mathbf{N}^G_A,\mathbf{N}^G_D) = \int_\Omega B^G_{ijA}\hat{D}_{ijkl}B^G_{klD}d\Omega \tag{3.14}$$

$$K^L_{BE} = a(\mathbf{N}^L_B,\mathbf{N}^L_E) = \int_{\Omega^L} B^L_{ijB}D_{ijkl}B^L_{klE}d\Omega \tag{3.15}$$

$$K^C_{AE} = a(\mathbf{N}^G_A,\mathbf{N}^L_E) = \int_{\Omega^L} B^G_{ijA}D_{ijkl}B^L_{klE}d\Omega \tag{3.16}$$

$$f^{ext}_A = (N^G_A,\mathbf{b})_\Omega + (N^G_A,\mathbf{t})_{\Gamma_t} = \int_\Omega N^G_{iA}b_i d\Omega + \int_\Gamma N^G_{iA}t_i d\Gamma \tag{3.17}$$

$$h^{ext}_B = (N^L_B,\mathbf{b})_{\Omega^L} + (N^L_B,\mathbf{t})_{\Gamma_t} = \int_{\Omega^L} N^L_{iB}b_i d\Omega + \int_\Gamma N^L_{iB}t_i d\Gamma \tag{3.18}$$

$\mathbf{B}^G$ and $\mathbf{B}^L$ are the strain-displacement matrices corresponding to the ESL and DL models, respectively. The **B**-matrix is modified using assumed strain method [47] to enhance the element performance as described in the next section. Effective (or smeared) laminate material properties $\hat{D}_{ijkl}$ are used for calculation of global stiffness matrix and plane stress assumption is incorporated in $\Omega\backslash\Omega^L$. 3D material properties are used for the matrices in the local region.

To maintain rank sufficiency of the stiffness matrix **K** the global and local interpolation spaces must be mutually exclusive. For structured mesh superposition (S_s-version of FEM [15]) exclusively considered here, this can be accomplished explicitly as demonstrated below:

From the definition of spaces $S^G(\Omega^L, p^G, q)$ and $S^L(\Omega^L, p^L)$ it is evident that:

$$S^L(\Omega^L, p^L) \cap S^G(\Omega^L, p^G, q) \neq \emptyset \tag{3.19}$$

Hence redundant modes are encountered in the system (3.12), since a linear combination of underlying mesh basis functions can replicate a superimposed mesh basis function or vice versa.

Define an auxiliary space

$$\hat{S}(\Omega^L) \;=\; S^L(\Omega^L, p^L) \cap S^G(\Omega^L, p^G, q) \tag{3.20}$$

so that the rank sufficiency of the stiffness matrix $\mathbf{K}$ can be maintained using the following interpolation space

$$S(\Omega) \equiv S^G(\Omega^G, p^G, q) \cup \left(S^L(\Omega^L, p^L) \backslash \hat{S}(\Omega^L)\right) \tag{3.21}$$

or

$$S(\Omega) \equiv S^L(\Omega^L, p^L) \cup \left(S^G(\Omega^G, p^G, q) \backslash \hat{S}(\Omega^L)\right) \tag{3.22}$$

The two decompositions of the space $S(\Omega)$ (3.21,3.22) in the local region denoted by, $S(\Omega^L)$, represent a complete space of polynomial order p^L. The decomposition given in (3.21) is more convenient from the computational efficiency stand point because it allows to exploit previous computations in certain cases discussed in Section 3.5.

In order to construct the space $S(\Omega^L)$ we decompose it as a tensor product of two spaces:

$$S(\Omega^L) = S_{2D}(A^L, p_A^L) \otimes S_{1D}(T, t, p_t^L) \tag{3.23}$$

where S_{2D} is spanned by 2D hierarchical basis of order p_A^L [54] defined over the mid-surface of the shell over the local region A^L and S_{1D}^L is spanned by 1D hierarchical basis functions of order p_t^L defined within each layer $t_i \subset T$, such that $\cup t_i = T$.

Each of the two spaces is constructed from a set of basis functions in the global and local meshes:

$$S_{2D}(A^L, p_A^L) \;=\; S_{2D}^G(A^L, p_A^G) \cup [S_{2D}^L(A^L, p_A^L) \backslash \hat{S}_{2D}^L(A^L)] \tag{3.24}$$

$$S_{1D}(T,t,p_t^L) \;=\; S_{1D}^G(t=T,q) \cup [S_{1D}^L(T,t,p_t^L) \backslash \hat{S}_{1D}^L(T,t)] \tag{3.25}$$

For the lower order plate/shell theories based on either Kirchhoff or Reissner hypothesis the through the thickness variation of inplane displacements u_α are linear, that is, $q = 1$ and $dim[S_{1D}^G(t = T, q)] = 2$, while the transverse displacement u_3 variation is constant, that is, $q = 0$ and $dim[S_{1D}^G(t = T, q)] = 1$.

To define the subspace in the local mesh which is to be constrained, $\hat{S}^L$ we limit ourselves to the case where the global space defined on the local region is a subspace of the local superimposed space, that is, $S^G(\Omega^L, p^G, q) \subset S^L(\Omega^L, p^L)$. In this scenario it is sufficient to constrain dofs of the local mesh corresponding to the following subspace:

$$\hat{S}^L = S_{1D}^G \otimes S_{2D}^G \tag{3.26}$$

This implies that it is necessary to constrain a number of shell surfaces or layer interfaces in the superimposed (DL) mesh equal to $dim[S_{1D}^G]$. In numerical examples we have constrained the top and bottom surfaces for inplane displacement and a bottom (or top) surface for the transverse displacement.

The second case of practical interest is that of partial superposition in the transverse direction as shown in Figure 3.1. This is especially important for multilayer composite with hundreds of layers, where it is desirable to superimpose only in the vicinity of most critical layers. In this scenario it is necessary to enlarge the constraint set $\hat{S}^L$ to maintain displacement compatibility between global and local meshes by constraining the dofs in the superimposed mesh at the global-local interface Γ^{GL}.

For the case of unstructured superposition (S_u-version) the redundant modes are eliminated during the solution phase as elucidated in [16]. In the current work only the structured mesh superposition methodology has been considered.

REMARK: The s-method can be utilized to hierarchically model discontinuous fields [17]. By this technique a mathematical model free of discontinuities is

superimposed by an additional field able to represent the discontinuity. The mathematical validation of this technique was given in [17]. In the present study the global ESL mesh free of delamination is superimposed by a local 3D mesh with delamination (discontinuity) modeled by double nodes along the delamination surface. Numerical investigation is presented in section 3.5.

3.3 Assumed Strain formulation and SDP quadrature scheme

To enhance the performance of the degenerated plate/shell elements primarily at lower spectral orders ($p \leq 4$) assumed strain formulation is employed. The formulation of the enhanced strain-displacement matrix $\bar{\mathbf{B}}$ can be obtained by interpolating B_{ijA} between a set of reduced quadrature points [32] or by selectively projecting out higher order modes in the quadrature process as described in Chapter 2.

To facilitate the analysis of laminated composite shells, where it is desirable to model multiple material layers within a single element, without having to use a corresponding number of thickness integration points, symmetric dot product integral decomposition can be adapted to perform thickness pre-integration. Thus symmetric dot product integral decomposition scheme can be used directly for analysis of laminated composite shells. The procedure for thickness pre-integration with SDP is outlined below:

Consider equation

$$k_{AB} = \int_{\Omega} (B_A^T J^{1/2})\ \phi_I d\Omega \cdot \int_{\Omega} (D)\ \phi_I \phi_J d\Omega \cdot \int_{\Omega} (B_B^T J^{1/2})\ \phi_J d\Omega \tag{3.27}$$

In case of laminated composites D is varying only through thickness, that is $D = D(\zeta)$. Thus we have,

$$\begin{aligned} \bar{D}_{IJ} &= \int_{\Omega} D(\zeta)\ \phi_I \phi_J d\Omega \\ \phi_I &= \hat{P}_i(\xi)\hat{P}_j(\eta)\hat{P}_k(\zeta) \end{aligned}$$

$$\begin{aligned}
\phi_J &= \hat{P}_l(\xi)\hat{P}_m(\eta)\hat{P}_n(\zeta) \\
\bar{D}_{IJ} &= \int_{-1}^{+1} \hat{P}_i(\xi)\hat{P}_l(\xi)d\xi \int_{-1}^{+1} \hat{P}_j(\eta)\hat{P}_m(\eta)d\eta \int_{-1}^{+1} D(\zeta)\ \hat{P}_k(\zeta)\hat{P}_n(\zeta)d\zeta \\
&= \int_{-1}^{+1} D(\zeta)\ \hat{P}_k(\zeta)\hat{P}_n(\zeta)d\zeta
\end{aligned}$$

where $0 \leq (k,n) \leq Q$; Q is the maximum polynomial order of shape functions in ζ direction. The integral $\int_{-1}^{+1} d\zeta$ is calculated as sum of integrals, where number of integrals depends on number of layers. The outlined method is computationally efficient and is implemented within the framework of symmetric dot product integral decomposition for HAMS and HSOL elements.

3.4 Adaptive Strategy

Based on the information from *a posteriori* error estimators and indicators the adaptive procedure aims to achieve the the following objectives:

- Demarcation of critical regions where DL model should be superimposed.
- Polynomial refinement in both global(ESL) and local(DL) meshes until the desired level of accuracy is obtained.

The error indicators presented in this section are based on the earlier work by Fish et.al. [15]. The estimated error in the ESL model, referred as Dimensional Reduction Error (DRE), $\mathbf{E}$, is approximated by a linear combination of some basis functions ϕ.

$$\mathbf{E} = \phi\ \beta \in E(\Omega^\star) \tag{3.28}$$

where β are determined by solving the minimization auxiliary problem:

$$\frac{\partial}{\partial\beta}\{\frac{1}{2}a(\mathbf{u}^{FE}+\mathbf{E},\mathbf{u}^{FE}+\mathbf{E})_\Omega - (\mathbf{u}^{FE}+\mathbf{E},\mathbf{b})_\Omega - (\mathbf{u}^{FE}+\mathbf{E},\mathbf{t})_\Gamma\} = 0 \tag{3.29}$$

and the functional space E is defined to maintain C^0 continuity of the augmented field $\mathbf{u}^{FE}+\mathbf{E}$ and to satisfy essential boundary conditions, that is

$$E = \{\mathbf{E}|\mathbf{E} = \phi\beta \in C^0(\Omega^\star);\ \ \mathbf{E} \in S^E(\Omega^\star, p^E);\ \ \mathbf{E} = 0\ \ on\ \ \Omega\backslash\Omega^\star\} \tag{3.30}$$

We now define the subspace S^E. For the process of identification of critical regions in the ESL plate/shell model the $\mathbf{u}^{FE}$ corresponds a given ESL finite element solution and ϕ represents layer-wise 3D finite element interpolants. To estimate the dimensional reduction error indicators, the space S^E is chosen as a 3D discrete layer interpolation space defined on the entire domain $\Omega^\star = \Omega$, with maximum polynomial order $p^E = 1$. It is desirable that the computational effort of the error estimation process be only a fraction of the finite element solution, hence in practice only an approximation for β is calculated. We will be seeking for an approximation of β by replacing the Hessian matrix $a(\phi, \phi)$ resulting from (3.29) by its diagonal. The resulting energy norm of the estimated error for each element is obtained by:

$$\eta_e = \sum_{A}^{NDOFS} \{\frac{1}{2}\beta_A^2 a(\phi_A,\ \phi_A)_{\Omega^e}\}^{\frac{1}{2}} \tag{3.31}$$

where NDOFS are the total number of degrees of freedom in the element 'e'. The critical elements which need to be superimposed are selected such that:

$$\eta_e \geq \gamma_e \max_e(\eta_e) \tag{3.32}$$

where parameter $\gamma_e \in [0, 1]$ controls the number of elements to be superimposed with $\gamma = 0$ corresponding to uniform superposition over the entire domain.

We now consider adaptive strategy based on selective polynomial escalation in the two meshes. Consider a given finite element space consisting of underlying and superimposed meshes

$$S^{FE} = S^G((\Omega^G \backslash \Omega^L), p^G, q, l^G) \cup [S^L(\Omega^L, p^L, l^L) \backslash \hat{S}^L(\Omega^L)] \tag{3.33}$$

and let us define a higher order space S^E obtained by increasing the polynomial order of both global and local interpolants.

$$S^E = [S^G((\Omega^G \backslash \Omega^L), p^G + 1, q) \cup S^L(\Omega^L, p^L + 1) \backslash \hat{S}^L(\Omega^L)] \setminus S^{FE} = SPAN\{\phi_A\} \tag{3.34}$$

As before approximation for β are found by replacing the Hessian matrix $a(\phi, \phi)$ by its diagonal. Error indicators corresponding to each higher order degree of freedom are calculated as:

$$\eta_A = \{\frac{1}{2}\beta_A^2 a(\phi_A,\ \phi_A)\}^{\frac{1}{2}} \quad \text{no sum on A} \tag{3.35}$$

in both meshes. The degree-wise error indicators η_A is used in selection of critical basis functions which reduce the error similar to the element-wise error indicators (3.32). The list of active degrees of freedom l^G and l^L is appended with those degrees-of-freedom which correspond to the error indicators η_A (3.35) such that:

$$\eta_A \geq \gamma \max_A(\eta_A) \tag{3.36}$$

where parameter $\gamma \in [0, 1]$ controls the speed of convergence with $\gamma = 0$ corresponding to uniform polynomial escalation.

3.5 Numerical Experiments

Numerical experiments are conducted to investigate the performance of s-method for modeling laminated composites in terms of accuracy and computational efficiency. The following classical test problems are considered:

- $(45/-45)_s$ laminate in axial tension [59] [57] [46]
- $(45/0/-45/90)_s$ laminate in axial tension [59]
- Edge delamination test for $(0/35/-35/90)_s$ laminate [46]
- Composite laminate Scordelis-Lo roof.

Geometry, boundary conditions and material properties for the axial tension problems are illustrated in Figures 3.2 and 3.3.

Our numerical experimentation agenda involves investigation of the following aspects:

- Ability of Dimensional Reduction Errors to capture critical regions.
- The influence of mesh gradation, selective polynomial escalation and assumed strain formulation on the rate of convergence.

Contour lines of the estimated error for $(45/-45)_s$ and $(45/0/-45/90)_s$ laminates subjected to axial tension are given in Figures 3.2 and 3.3 respectively. The contour lines were obtained by interpolating estimated elemental errors. The estimated error distribution substantiates the well known fact that edge effects are prominent near the free edges of the laminates. A layer-wise finite element mesh is superimposed over the critical elements.

In our current study identical inplane mesh subdivision for underlying and overlying meshes has been employed. Numerical comparison of meshes with different gradation in resolving 3D stress fields in $(45/-45)_s$ laminate is illustrated in Figure 3.5. The uniform superimposed mesh shown in Figure 3.4 is obtained by layer-wise subdivision in the transverse direction. The two geometric meshes in Figure 3.4 have layers of elements graded towards the free edge and the ply interface with a common factor of 0.15. Normalized stress distribution ($\bar{\sigma_{i3}} = \sigma_{i3}/\epsilon_0$ where $\epsilon_0 =$ axial strain) near the free edges for each type of mesh is shown in Figure 3.5. The reference stress distributions at the free edges are depicted from [57], [46] [59]. It is evident from the plots in Figure 3.5 that geometric meshes are vital to accurately estimate the stress field near free edges and polynomial order escalation alone in uniform meshes is not sufficient.

Assumed strain and displacement based elements (for both underlying and superimposed models) as well as the quadrature schemes are compared in Figure 3.6. The CPU times required for the calculation of total (global and local) stiffness employing SDP and uniform quadrature schemes are compared for the $(45/-45)_s$ problem modeled with two level geometric mesh (Figure 3.4). Plots of absolute maximum stress and error in energy norm versus the total CPU time for stiffness

calculation are shown in Figure 3.6. Both formulations are quite competent in estimating maximum stresses, although the assumed strain formulation has faster convergence. Moreover it is evident that SDP quadrature is computationally more efficient than the uniform quadrature scheme for both assumed strain and displacement based elements.

To study the effect of selective polynomial order escalation on the resolution of free edge stresses, the $(45/-45)_s$ and $(45/0/-45/90)_s$ laminates modeled with meshes as shown in Figures 3.7 and 3.8 respectively, are considered. In Figures 3.7 and 3.8 plots of maximum stresses with and without selective polynomial order escalation for the local mesh are presented for $(45/-45)_s$ and $(45/0/-45/90)_s$ laminates. It is evident that selective polynomial order escalation speeds up the rate of convergence. In the problems considered the parameter γ (3.36) was set equal to 0.01 and selective polynomial order escalation was performed only on the local mesh.

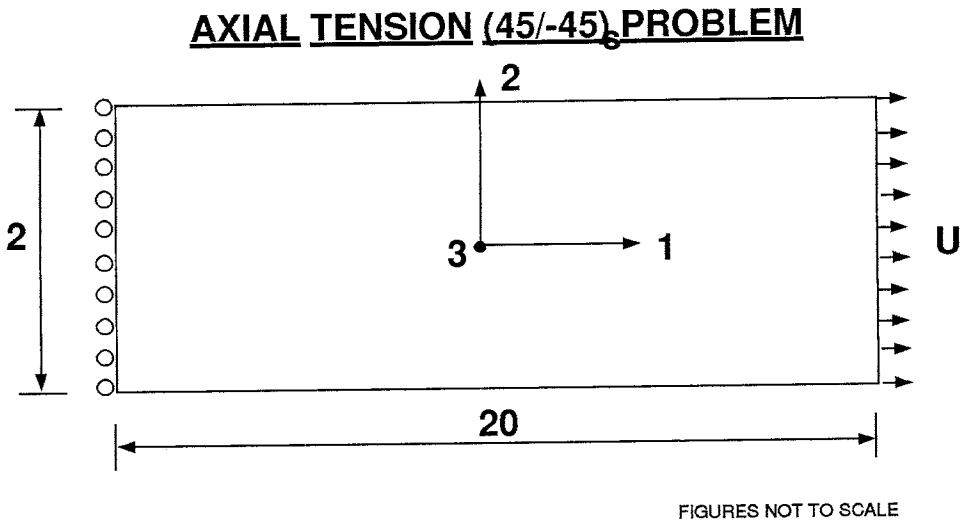
In Figure 3.9 the composite laminate Scordelis-lo roof problem is investigated. The elemental error densities in a 4x4 uniform mesh are illustrated. The two elements with maximum errors are superimposed with discrete layer mesh. The parameter γ_e was chosen as 0.75. The convergence in energy norm and in the maximum normal stress with uniform and selective polynomial escalation are depicted in the plots. Selective polynomial escalation is carried out in both local and global meshes. It is evident from the plots that selective polynomial escalation simultaneously in both meshes has to be carried out, selective polynomial escalation in the local mesh alone is not sufficient unless the global mesh is sufficiently refined.

Partial through-the-thickness mesh superposition where only two (0,-45) lamina are superimposed in the $(45/0/-45/90)_s$ laminate, is presented in Figure 3.10. It can be seen that the accuracy of stresses at the (0/-45) interface is not significantly compromised due to the partial superposition.

The edge delamination tension (EDT) problem is used to test the ability of the model to predict energy release rates. The particular laminate chosen for this experiment is $(0/35/-35/90)_s$, since this problem has been analyzed using layer-wise theory in [46] and using quasi-3D finite element model in [43]. The geometry, boundary conditions, symmetry and material properties are illustrated in Figure 3.11. The specimen has four delaminations two along each free edge at the $-35/90$ and $90/-35$ interfaces. The delamination surface interpenetration has been ignored in the current study. Three dimensional Virtual crack closure method [51] is used to calculate the total energy release rate and its components. In Figure 3.11 normalized energy release rates are plotted versus the specimen length. The energy release rates are normalized as follows:

$$(\bar{G},\ \bar{G}_I,\ \bar{G}_{II}) = \frac{(G,\ G_I,\ G_{II})}{\epsilon_0^2\ 0.5t\ L} \tag{3.37}$$

where L is the length of the specimen, t the total thickness and ϵ_0 the uniform axial strain. It was observed that the contribution of mode III towards the total energy release rate was negligible and mode II was the prominent contributor. The energy release rates are fairly constant along the central portion of the specimen and show a decrease near the ends, where the delamination surfaces are constrained against relative motion to simulate friction grips. The results presented match closely with the results presented in [46] in the central portion of the specimen.



Thickness of each lamina = 0.125

$E_1 = 20.0 \times 10^6$ psi,
$E_2 = E_3 = 2.10 \times 10^6$ psi
$G_{12} = G_{23} = G_{13} = 0.85 \times 10^6$ psi,
$\nu_{12} = \nu_{23} = \nu_{13} = 0.21$.

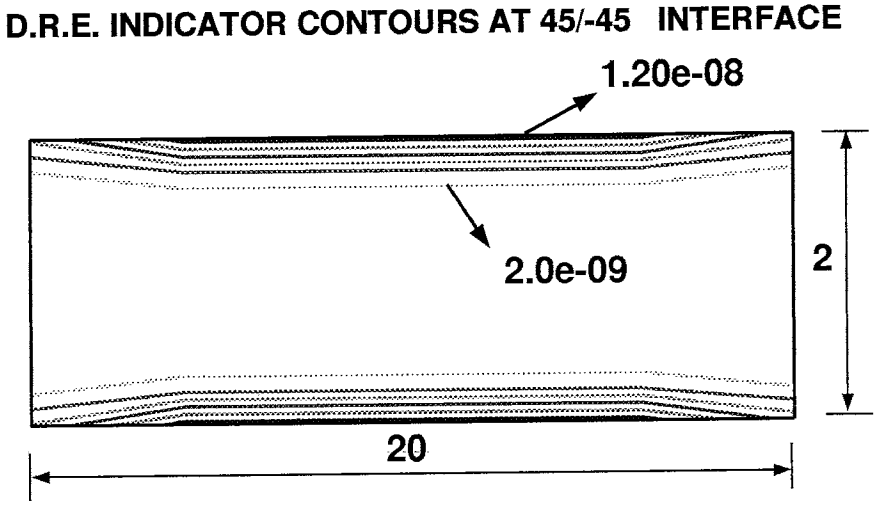
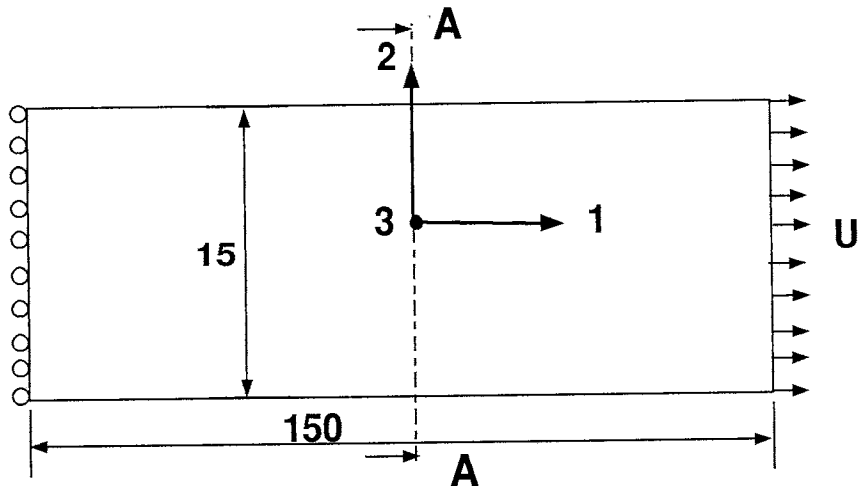


Figure 3.2: Axial Tension $(45/-45)_s$ Laminate

AXIAL TENSION PROBELM (90/-45/0/45)s COMPOSITE LAMINATE



Thickness of each lamina = 0.125

E_1 = 19.5 x 10^6 psi,
$E_2 = E_3$ = 1.48 x 10^6 psi
$G_{12} = G_{23} = G_{13}$ = 0.8 x 10^6 psi,

$\nu_{12} = \nu_{23} = \nu_{13} = 0.30.$

FIGURES NOT TO SCALE

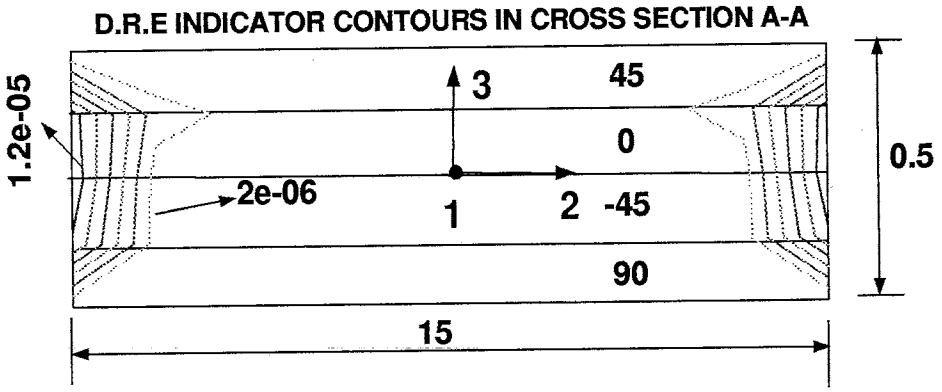


Figure 3.3: Axial Tension $(90/-45/0/45)_s$ Laminate

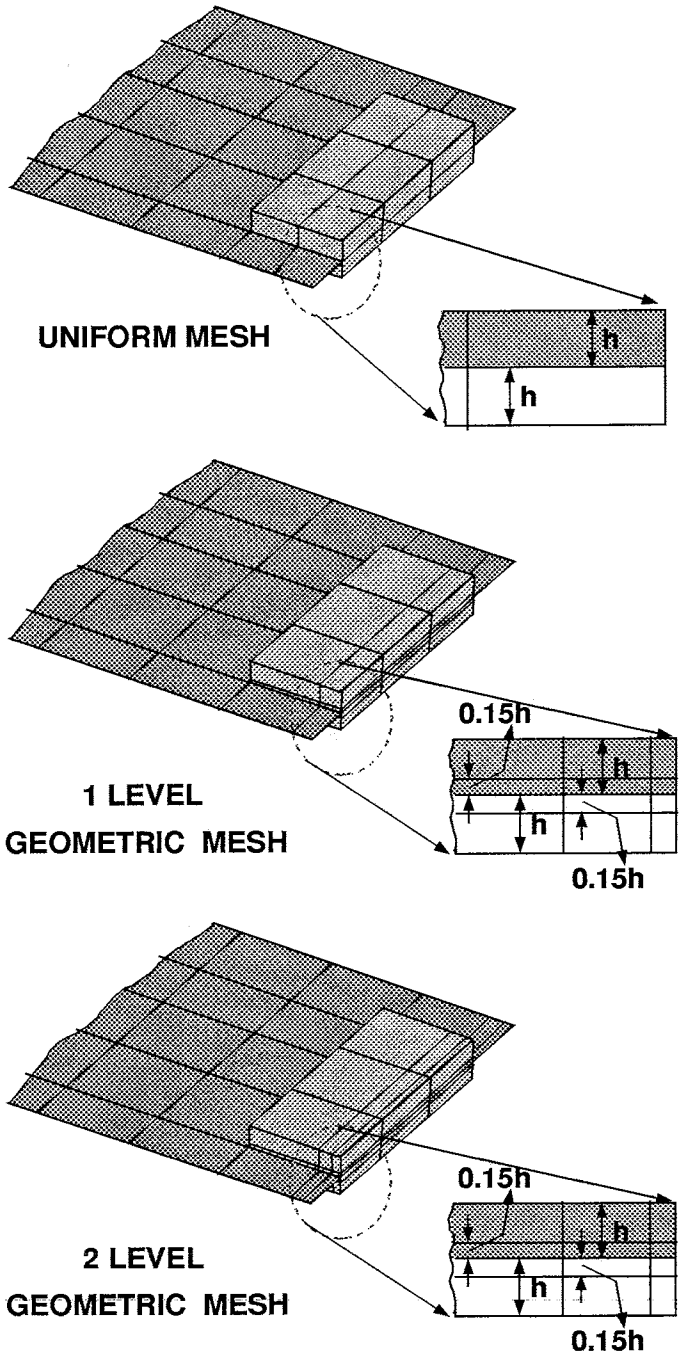


Figure 3.4: Meshes used for Free Edge Stress Estimation

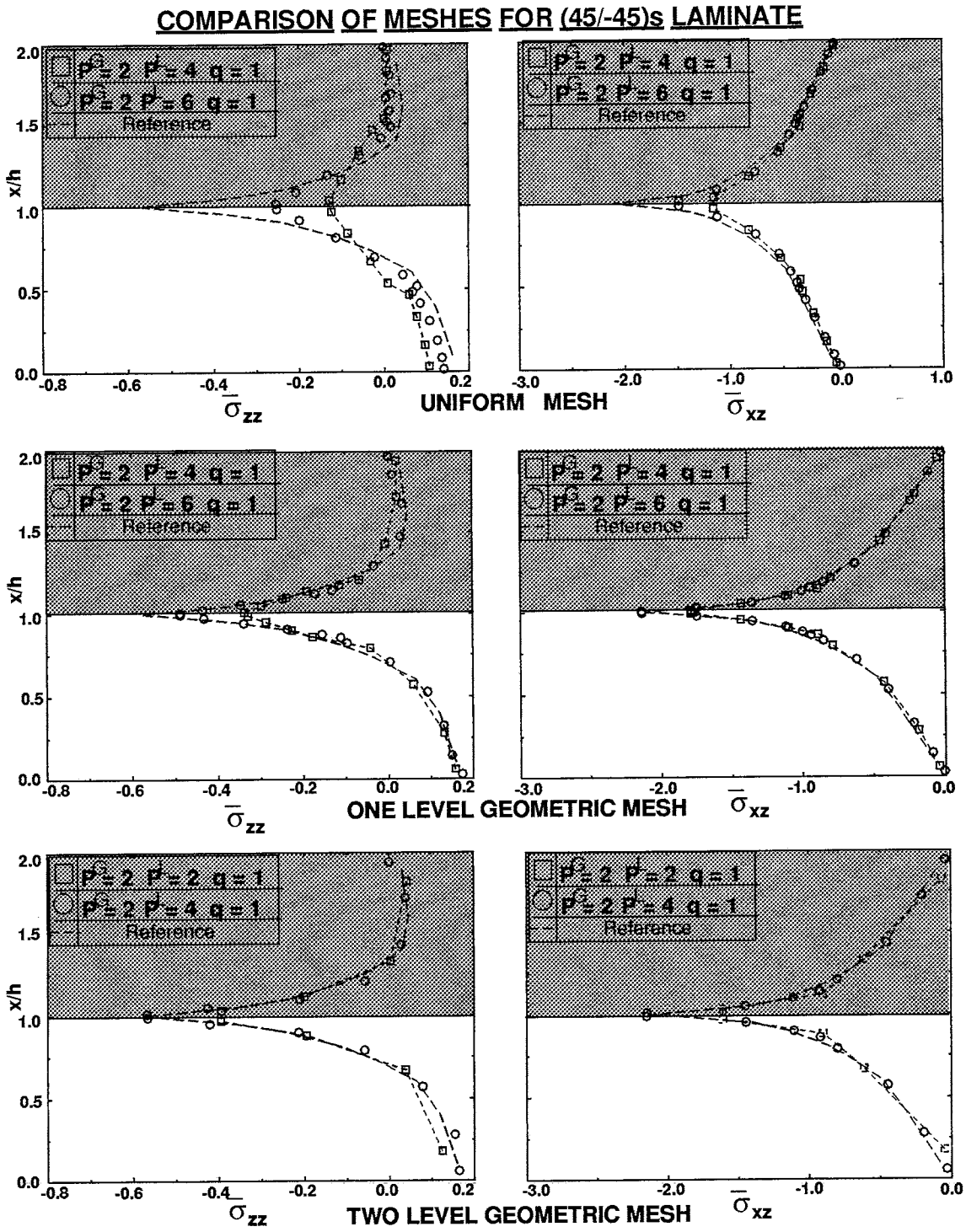


Figure 3.5: Comparison of Meshes for $(45/-45)_s$ Laminate

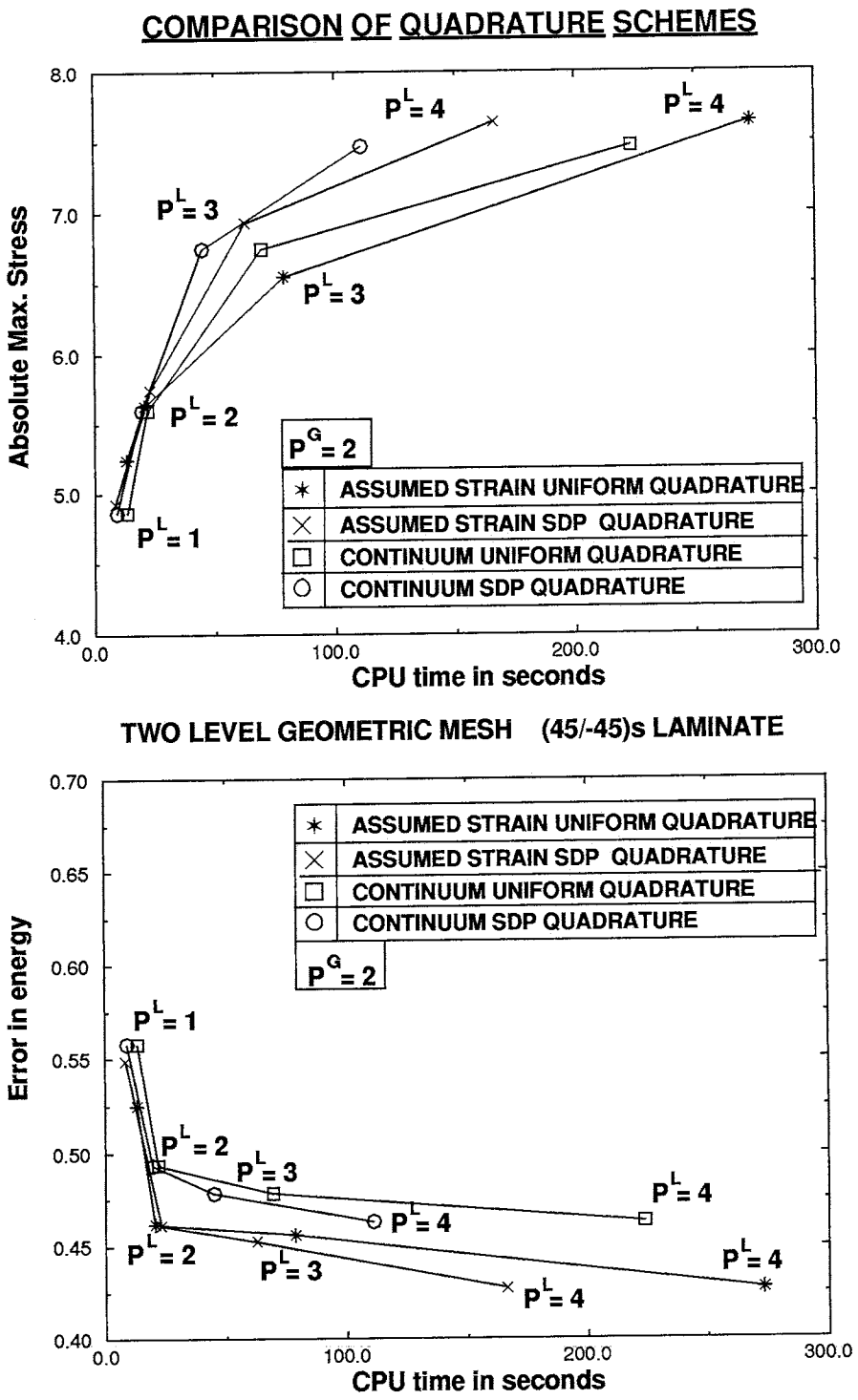


Figure 3.6: Comparison of Different Elements and Quadrature schemes

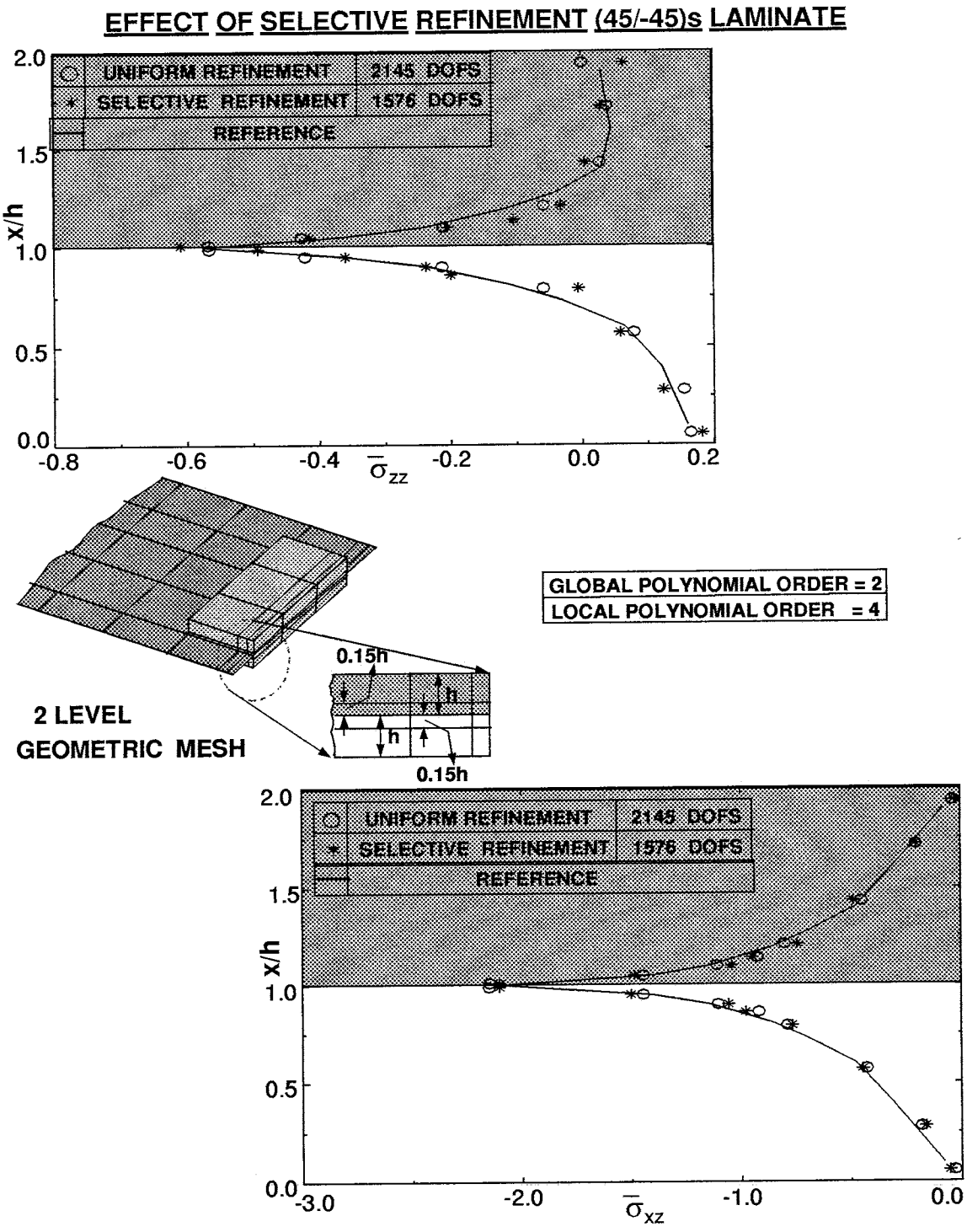


Figure 3.7: Effect of Selective Refinement $(45/-45)_s$ Laminate

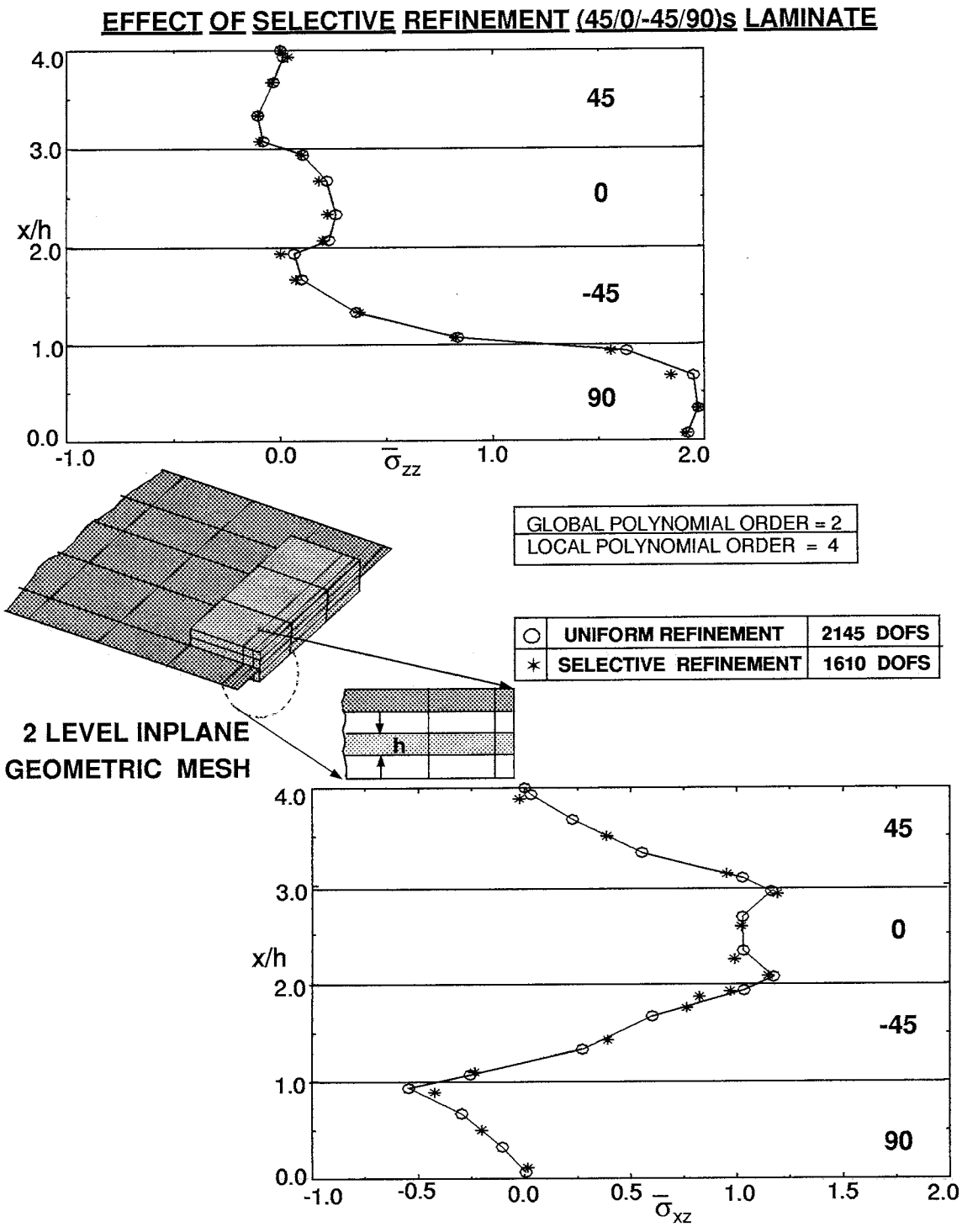


Figure 3.8: Effect of Selective Refinement $(45/0/-45/90)_s$ Laminate

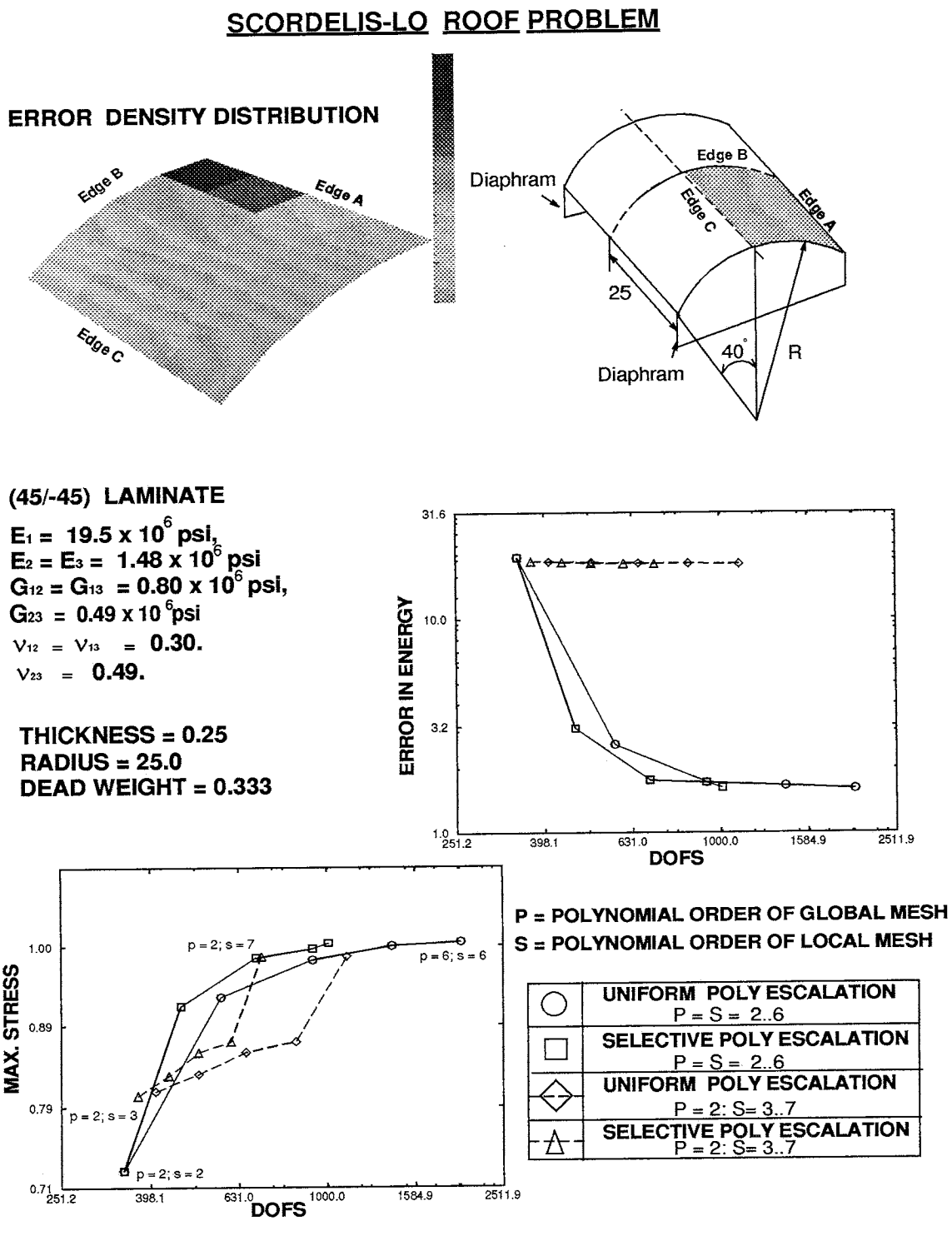


Figure 3.9: Scordelis-Lo Roof Problem

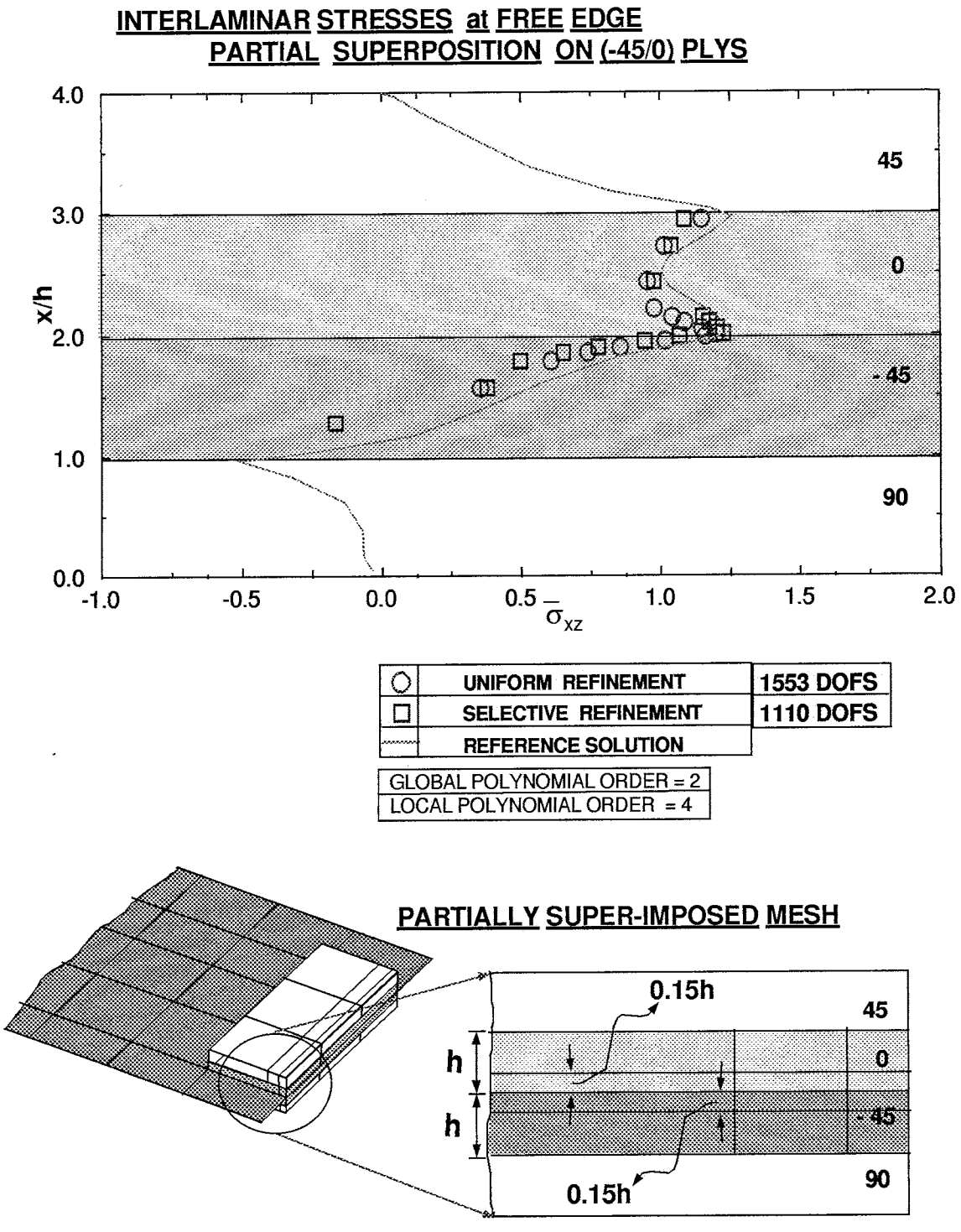


Figure 3.10: Interlaminar Stresses at Free Edge Partial Superposition

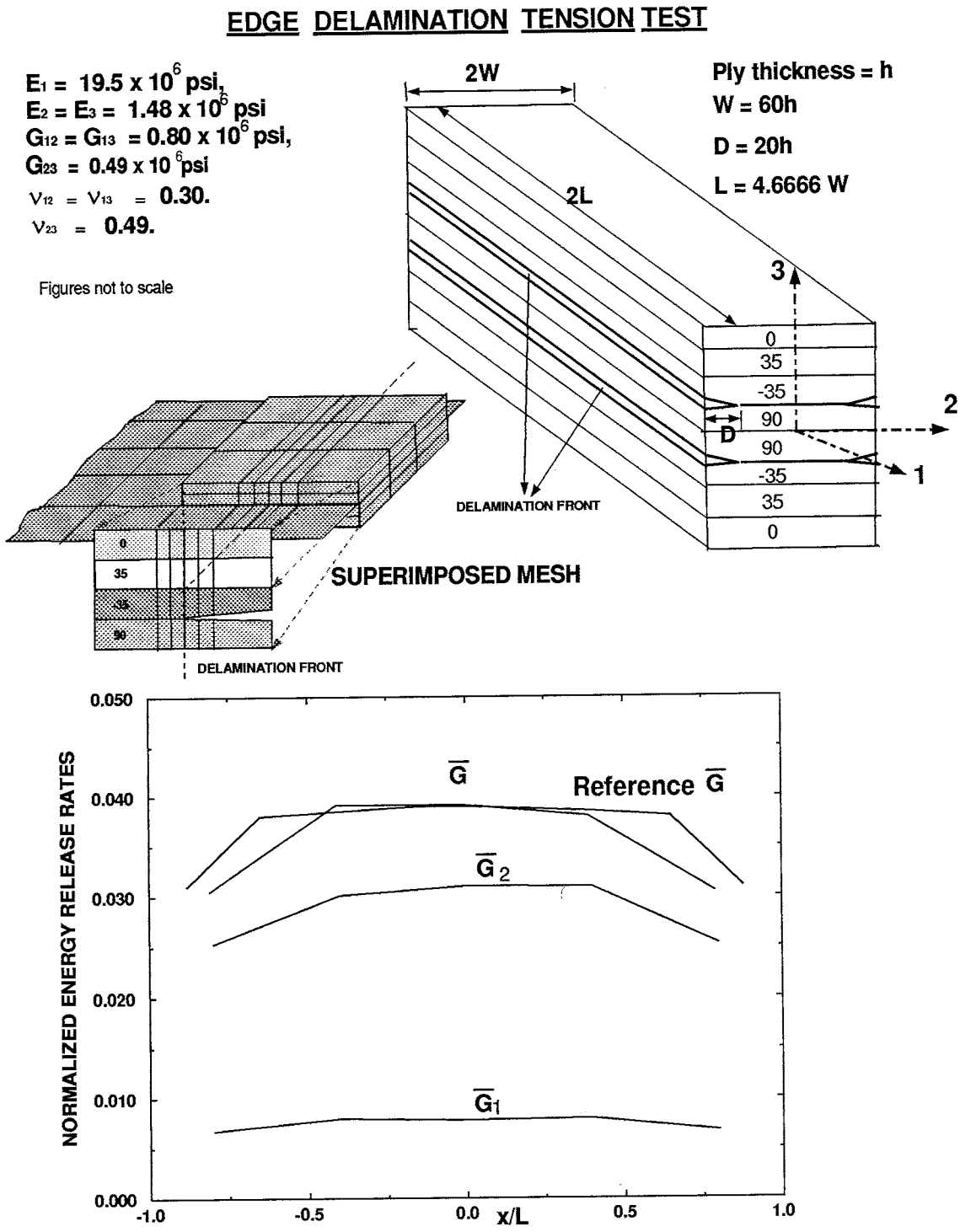


Figure 3.11: Edge Delamination Tension Test

CHAPTER 4
Adaptive Solver for Hierarchic Finite Element Methods

4.1 Introduction

The focus of this chapter is on developing an adaptive multilevel preconditioned solver for symmetric positive definite hierarchic system resulting from the p and s-versions of finite element method. Adaptive single level preconditioned conjugate gradient solvers for such systems have been investigated by researchers at I.B.M. [40] and Mandel [37]. An ideal choice of the preconditioner, that minimizes computational effort depends on computer architecture, software considerations, size of the problem, sparsity pattern and condition number of the system to be solved. The vital constituents of the multilevel preconditioned method are the number of auxiliary levels, their discretization, and the technique used to process individual levels. A methodology to determine an ideal choice of these constituents based on the problem data is explored.

Theoretical guidelines for choosing the number of levels and their discretization resulting in optimal number of iterations have been reported in [28]. However, such choices may not necessarily result in optimal CPU times. In large-scale problems, besides floating point operations, matrix computations involve a large amount of integer and logical operations, which can take a significant portion of the total computer time. In addition memory traffic can be a bottleneck. For optimal computational efficiency it is important to pay as much attention to the flow of data and the logical and integer operations as to the amount of floating point arithmetic. Hence in the present work some practical strategies which result in nearly optimal CPU times are devised.

In section 4.2 the multilevel preconditioned algorithm and solution strategy

for hierarchic systems is elucidated. Adaptive selection strategies and numerical results are presented in section 4.3.

4.2 Multilevel Preconditioned Methods for Hierarchic Systems

The hierarchic nature of the stiffness matrix resulting from the p-method is well suited for a multilevel preconditioned iterative method. The hierarchic levels utilized during the multilevel iteration process play the role of nested grids employed in the traditional multigrid method [12]. Multigrid iterations reduce the errors in different frequencies by using auxiliary grids and in case of the p-version of finite element method the frequency decomposition is directly available in terms of spectral orders. For an extension of the algorithm to the s-method we refer to [20].

Consider the hierarchic system:

$$\mathbf{K}^m \, \mathbf{d}^m \; = \; \mathbf{f}^m \qquad m = 1, 2 \ldots NUM - LEVELS$$

where

$$\mathbf{K}^m = \begin{bmatrix} \mathbf{K}^{m-1} & \mathbf{K}^m_{12} \\ \mathbf{K}^m_{21} & \mathbf{K}^m_{11} \end{bmatrix} \quad \mathbf{d} = \begin{Bmatrix} \mathbf{d}^{m-1} \\ \mathbf{d}^m_1 \end{Bmatrix} \quad \mathbf{f} = \begin{Bmatrix} \mathbf{f}^{m-1} \\ \mathbf{f}^m_1 \end{Bmatrix} \tag{4.1}$$

m is the maximum number of hierarchic levels used in multilevel iterations. $\mathbf{K}^0$ the stiffness matrix on the initial level; $\mathbf{K}^m$ is of order $n_m > n_{m-1}$, where n_{m-1} is the order of the block $\mathbf{K}^{m-1}$; $\mathbf{d}^m_1 \in \Re^{(n_m - n_{m-1})}$ and $\mathbf{d}^{m-1} \in \Re^{n_{m-1}}$.

Let $\mathbf{Q}^{m-1}_m$ and $\mathbf{Q}^m_{m-1}$ be the restriction and prolongation operators, which transfer the data from level (m) to level $(m-1)$ and vice versa. For the p-method it has a very simple form:

$$\mathbf{Q}^{m-1}_m \; = \; [\mathbf{I} \;\; \mathbf{0}] \; = \; {\mathbf{Q}^m_{m-1}}^T \tag{4.2}$$

where $\mathbf{I}$ is the order n_{m-1} identity matrix, and $\mathbf{0}$ is order $(n_m - n_{m-1})$ zero matrix. A single multilevel iteration has a compact recursive definition given by:

$$\mathbf{z}^m := ML^m \left(\mathbf{r}^m, \mathbf{K}^m \right). \tag{4.3}$$

Table 4.1: Multilevel Algorithm

1. Loop $i = 0, 1, 2 \ldots$ until convergence
   if $i = 0 \leftarrow \mathbf{d}^m = 0$
2. perform γ_1 pre-smoothing operations

$$_{\gamma_1}{}^{i}\mathbf{d}^m := \text{smooth}(\gamma_1, {}_0{}^{i}\mathbf{d}^m, \mathbf{K}^m, \mathbf{f}^m)$$

   where the left superscript and subscript denote the cycle number and smoothing count respectively.
3. Restrict residual from level m to $m-1$

$$\mathbf{r}^{m-1} = \mathbf{Q}_m^{m-1}(\mathbf{f}^m - \mathbf{K}^m {}_{\gamma_1}{}^{i}\mathbf{d}^m)$$

4. Coarse level correction

   If $(m-1)$ = lowest level, solve directly
   $\mathbf{z}^{m-1} = (\mathbf{K}^{m-1})^{-1}\mathbf{r}^{m-1}$,

   Else $\mathbf{z}^{m-1} := ML^{m-1}(\mathbf{r}^{m-1}, \mathbf{K}^{m-1})$
5. Prolongate from level $m-1$ to m

$$_{\gamma_1+1}{}^{i}\mathbf{d}^m = {}_{\gamma_1}{}^{i}\mathbf{d}^m + \mathbf{Q}_{m-1}^m \mathbf{z}^{m-1}$$

6. Perform γ_2 post-smoothing operations

$$^{i+1}{}_0\mathbf{d}^m := \text{smooth}(\gamma_2, {}_{\gamma_1+1}{}^{i}\mathbf{d}^m, \mathbf{K}^m, \mathbf{f}^m)$$

where $\mathbf{r}^m$ is the residual vector. The details of a V-cycle multilevel iteration process are given in Table 4.1.

The hierarchic multilevel preconditioned method involves three crucial steps:

- Coarse Level Correction (CLC).
- Smoothing.
- Acceleration.

In case of a 2-level scheme, CLC is performed by a direct solution. In the 1-level scheme, CLC corresponds to the direct solution of the entire system. The

computational complexity of CLC is influenced by the sparsity pattern of the stiffness matrix. On the other hand, the rate of convergence of the iterative method is governed by condition number of the problem. Thus the computational work of the multilevel preconditioned method is dependent on both, the sparsity pattern and the condition number of the stiffness matrix. Memory considerations also play a decisive role in the selection of a solution strategy.

4.2.1 Smoothing

The performance of the multilevel preconditioned method is influenced by the type of smoothing technique chosen, we have identified three efficient smoothing techniques:

- Block Diagonal Smoothing.
- Incomplete Cholesky factorization(ICC).
- Symmetric Gauss Seidel(SGS).

In case of Block Diagonal smoothing, a smaller than the normal subset of unknowns can be updated during the smoothing phase at a given level by taking advantage of the fact that smoothing mainly affects highest oscillatory modes of error. Thus relaxation sweeps (smoothing) can be performed on block by block level keeping the rest of the degrees of freedom fixed. The multilevel method resulting from such block diagonal smoothing was denominated by Bank, Dupont and Yserentant [8] as hierarchical basis multigrid technique (HBM). It has been shown [8] that the rate of convergence of HBM has logarithmic dependence on the problem size as opposed to the multilevel method with regular smoothing which has an optimal rate of convergence independent of the mesh size and spectral order.

An incomplete Cholesky factor is a popular smoother and results in robust multilevel methods [3]. Unfortunately incomplete factor exists only for some special

cases of symmetric positive definite matrices (*M-matrices*). For general class of symmetric positive definite matrices rejection (dropping) of nonzero terms during incomplete factorization often leads to an unstable factorization process resulting in small pivots in the factor or in an indefinite system. In such cases the diagonal entries are modified by adding a small positive number when required before elimination takes place to guarantee that the resulting incomplete factor is positive definite.

Based on the type of rejection (dropping) criteria, two types of incomplete factorizations are possible:

- Incomplete factorization by position ICC(P): In this case any nonzero entry generated during factorization is not retained if it does not fit into the sparsity pattern adopted [34], which in our case coincides with that of the matrix $\mathbf{K}$.

- Incomplete factorization by magnitude or value ICC(V): In this case the nonzero entry generated during factorization is retained only if its magnitude satisfies a specified criterion [2].

The incomplete factorization by value ICC(V) is computationally expensive compared to that by position ICC(P), but results in higher rate of convergence of the multilevel method especially for ill-conditioned problems. For incomplete factorization of the dense hierarchic stiffness matrix a combination of ICC(V) with no fill-in for lower polynomial orders and ICC(P) for higher orders involves significantly less computational work than ICC(V) for the entire stiffness matrix. This strategy emulates a block diagonal preconditioner with larger blocks for lower polynomial orders [4].

The Symmetric Gauss Seidel (SGS) smoothing is particularly suited for relatively well-conditioned problems. The SGS smoothing is very attractive from memory considerations as it requires no extra storage.

The type of smoothing ($level > 1$) will be adaptively selected based on the problem data.

4.2.2 Acceleration Schemes

For ill-conditioned problems, such as thin shells, it is desirable to accelerate the rate of convergence of the multilevel methods. In this subsection we present two acceleration schemes which require a small fraction of the total computational effort, but at the same time are efficient in expediting the convergence of the multilevel methods.

Two Parameter Acceleration scheme.

Let ${}^{i}\mathbf{r}^{m}$ be the residual vector at the end of i^{th} multilevel (m-level) iteration. The incremental multilevel solution for the next iteration ${}^{i}\mathbf{z}^{m} = ML^{m}({}^{i}\mathbf{r}^{m}, \mathbf{K}^{m})$ is used as a predictor in the two parameter acceleration scheme. The solution in the correction phase is then updated as follows:

$$ {}^{i+1}\mathbf{v} = {}^{i}\alpha \; {}^{i}\mathbf{z}^{m} + {}^{i}\beta \; {}^{i}\mathbf{v} \tag{4.4} $$

$$ {}^{i+1}\mathbf{d}^{m} = {}^{i}\mathbf{d}^{m} + {}^{i+1}\mathbf{v} \tag{4.5} $$

where parameters $({}^{i}\alpha, \; {}^{i}\beta)$ are obtained by minimizing the potential energy functional:

$$ \frac{1}{2}\,({}^{i}\mathbf{d}^{m} + {}^{i}\alpha {}^{i}\mathbf{z}^{m} + {}^{i}\beta {}^{i}\mathbf{v})^{T}\mathbf{K}^{m}({}^{i}\mathbf{d}^{m} + {}^{i}\alpha {}^{i}\mathbf{z}^{m} + {}^{i}\beta {}^{i}\mathbf{v}) - ({}^{i}\mathbf{d}^{m} + {}^{i}\alpha {}^{i}\mathbf{z}^{m} + {}^{i}\beta {}^{i}\mathbf{v})^{T}\mathbf{f}^{m} \longrightarrow \min_{{}^{i}\alpha\,{}^{i}\beta} \tag{4.6} $$

The resulting multilevel preconditioned algorithm is summarized in Table 4.2.

Conjugate Gradient Acceleration Scheme.

The conjugate gradient method can be used as an acceleration scheme for the multilevel method. The acceleration parameters ${}^{i}\alpha, {}^{i}\beta$ are found from the line search and $\mathbf{K}$-orthogonality $(\mathbf{K} \; {}^{i}\mathbf{v}, {}^{i+1}\mathbf{v}) = 0$ conditions respectively. The resulting multilevel

Table 4.2: Two Parameter Acceleration Algorithm

Step 1.

$$
\begin{aligned}
{}^0\mathbf{d} &= 0, \quad {}^0\mathbf{r} = \mathbf{f} \\
{}^0\mathbf{z} &:= ML({}^0\mathbf{r}, \mathbf{K}) \\
{}^0\mathbf{v} &= {}^0\mathbf{y} = 0 \\
{}^0\mathbf{x} &= \mathbf{K}\ {}^0\mathbf{z}
\end{aligned}
$$

$$
{}^0\beta = 0\,; \quad {}^0\alpha = \frac{(\mathbf{f}, {}^0\mathbf{z})}{({}^0\mathbf{x}, {}^0\mathbf{z})} \tag{4.7}
$$

Step 2.
Do $i = 0, 1, 2 \ldots$ until convergence

$$
\begin{Bmatrix} {}^i\alpha \\ {}^i\beta \end{Bmatrix} = \begin{bmatrix} ({}^i\mathbf{x}, {}^i\mathbf{z}) & ({}^i\mathbf{x}, {}^i\mathbf{v}) \\ ({}^i\mathbf{x}, {}^i\mathbf{v}) & ({}^i\mathbf{y}, {}^i\mathbf{v}) \end{bmatrix}^{-1} \begin{Bmatrix} ({}^i\mathbf{r}, {}^i\mathbf{z}) \\ ({}^i\mathbf{r}, {}^i\mathbf{v}) \end{Bmatrix} \quad \forall i > 0 \tag{4.8}
$$

$$
\begin{aligned}
{}^{i+1}\mathbf{v} &= {}^i\alpha\ {}^i\mathbf{z} + {}^i\beta\ {}^i\mathbf{v} \\
{}^{i+1}\mathbf{d} &= {}^i\mathbf{d} + {}^{i+1}\mathbf{v} \\
{}^{i+1}\mathbf{y} &= {}^i\alpha\ {}^i\mathbf{x} + {}^i\beta\ {}^i\mathbf{y} \\
{}^{i+1}\mathbf{r} &= {}^i\mathbf{r} - {}^{i+1}\mathbf{y} \\
{}^{i+1}\mathbf{z} &:= ML({}^{i+1}\mathbf{r}, \mathbf{K}) \\
{}^{i+1}\mathbf{x} &= \mathbf{K}\ {}^{i+1}\mathbf{z}
\end{aligned}
$$

Convergence Criteria.

$$
\sqrt{\frac{({}^i\mathbf{r}, {}^i\mathbf{r})}{({}^0\mathbf{r}, {}^0\mathbf{r})}} < \varepsilon
$$

preconditioned conjugate gradient algorithm is outlined in Table 4.3.

It can be shown that the two schemes are mathematically equivalent in absence of roundoff errors. However for well-conditioned problems conjugate gradient acceleration is superior because it involves fewer scalar product evaluations. On the other hand for poor conditioned problems the two parameter acceleration is less sensitive to roundoff errors, resulting in fewer iterations (see section 4.3). Note that the two acceleration schemes require no additional matrix-vector multiplication and

their benefit clearly overshadows the cost involved in vector product evaluations[19].

4.2.3 Adaptive Multilevel Preconditioned Solution Method.

In this subsection the adaptive strategy for solving symmetric positive definite hierarchic systems is outlined, which includes:

1. Estimate condition number (κ) and sparsity (ϖ) of the system.
2. Estimate memory requirements (μ_i) for alternative solution methods.
3. Given κ, ϖ, μ_i and maximum available memory select an optimal multilevel method. A particular choice of the multilevel method includes selection of:
   - Type of smoothing.
   - Number of levels.
   - Acceleration scheme.
4. Monitor the iterative algorithm for localized divergence.

The iteration process is monitored by the values of parameters (α, β)(4.7,4.8) (4.9,4.10). A localized divergence is indicated by negative or very small values of acceleration parameter α. Localized divergence indicates that a stronger multilevel preconditioner is required. Localized divergence can be circumvented either by selecting a stronger smoothing strategy or by increasing the size of the coarse level. When incomplete factorization smoothing is used in the multilevel scheme, divergence is attributed to incomplete factor being close to singular. In this case incomplete factor is recalculated on a diagonally scaled system or by performing ICC(V) on a larger lower order system. Another possibility is allowing additional fill-in during factorization. In the present case a combination of diagonal scaling and ICC(V) on a larger lower order system is employed.

Table 4.3: CG Acceleration Algorithm

Step 1.

$$
\begin{aligned}
{}^{0}\mathbf{d} &= 0, \quad {}^{0}\mathbf{r} = \mathbf{f} \\
{}^{0}\mathbf{z} &:= ML({}^{0}\mathbf{r}, \mathbf{K}) \\
{}^{0}\mathbf{v} &= {}^{0}\mathbf{z}
\end{aligned}
$$

Step 2.
Do $i = 0, 1, 2 \ldots$ until convergence

$$
\begin{aligned}
{}^{i}\alpha &= \frac{{}^{i}\mathbf{r}^{T}\,{}^{i}\mathbf{v}}{{}^{i}\mathbf{v}^{T}\,\mathbf{K}\,{}^{i}\mathbf{v}} \qquad (4.9) \\
{}^{i+1}\mathbf{d} &= {}^{i}\mathbf{d} + {}^{i}\alpha\ {}^{i}\mathbf{v} \\
{}^{i+1}\mathbf{r} &= {}^{i}\mathbf{r} - {}^{i}\alpha\,\mathbf{K}\,{}^{i}\mathbf{v} \\
{}^{i+1}\mathbf{z} &= \mathbf{P}^{-1}\,{}^{i+1}\mathbf{r} = ML({}^{i+1}\mathbf{r},\ \mathbf{K}) \\
{}^{i+1}\beta &= \frac{{}^{i+1}\mathbf{r}^{T}\,{}^{i+1}\mathbf{z}}{{}^{i}\mathbf{r}^{T}\,{}^{i}\mathbf{z}} \qquad (4.10) \\
{}^{i+1}\mathbf{v} &= {}^{i+1}\mathbf{z} + {}^{i+1}\beta\ {}^{i}\mathbf{v}
\end{aligned}
$$

Convergence Criteria.

$$
\sqrt{\frac{({}^{i}\mathbf{r}, {}^{i}\mathbf{r})}{({}^{0}\mathbf{r}, {}^{0}\mathbf{r})}} < \varepsilon
$$

An intelligent choice of the multilevel schedule and adaptive smoothing are vital for computational efficiency of the multilevel preconditioned method. The ability to reliably predict the optimal solution strategy for a given problem by purely theoretical means is questionable, and thus numerical experiments are utilized in the decision making process.

4.3 Selection of Optimal Multilevel Preconditioned Method

In this section we elucidate the decision making methodology to select an optimal multilevel preconditioned method. Implementation issues of the adaptive multilevel preconditioned method are also outlined. Numerical experiments are conducted on shell problems with varying thickness, number of elements, and polynomial orders. For the 3D problems only the number of elements, and polynomial orders are changed. Large-scale shell problems considered here are: the Canoe and Car examples are illustrated in figure 4.1. These problems are modeled with a hierarchic shell elements with six dofs per node (3 global translations and 3 global rotations) with the geometry mapped by cubic Lagrange interpolation functions. The Canoe is meshed with 288 and 512 elements, the Car with 198 and 252 elements respectively. The 3D problems considered are the flange and the V-block illustrated in figure 4.2. Both the 3D problems are modeled with hexahedral elements. The geometry of the hexahedral elements is mapped by cubic Lagrangian functions. The V-block is meshed with 63 and 128 elements and the Flange is modeled with 82 and 135 elements, respectively. The degrees of freedom corresponding to interior modes are statically condensed for both shell and hexahedral elements. Static condensation of interior modes results in a better conditioned system without affecting the sparsity. The condition number of the statically condensed system grows as $O(log^2 p)$ as compared to $O(p^2)$ of the system without static condensation for 2D problems [7]. For convergence ε is selected as 10^{-5}.

An efficient multilevel preconditioned method requires a computationally efficient coarse level correction method at the lowest level. A sparse direct solver [25], which exploits the sparsity of the stiffness matrix is employed. In [25] the equations are renumbered using the minimum degree algorithm [56]. The sparse [25] solver is superior to most envelope methods in terms of computational work and has been widely used in industrial applications.

4.3.1 Estimation of Condition Number, Sparsity and Storage requirements

Estimation of the condition number, the sparsity and storage requirements play an important role in the decision making process of selecting the optimal solution strategy. The exact calculation of the condition number is not practical and thus an estimate to the condition number is evaluated. The condition number estimate $\hat{\kappa}$ is evaluated as follows:

$$\hat{\kappa} = \frac{\hat{\lambda}_{max}}{\hat{\lambda}_{min}} \tag{4.11}$$

where $\hat{\lambda}_{max}$ and $\hat{\lambda}_{min}$ are the estimates of maximum and minimum eigenvalues of $\mathbf{K}$, respectively. For a symmetric positive definite matrix $\mathbf{K}$, the maximum eigenvalue is bounded by the maximum matrix norm $\| \mathbf{K} \|_\infty$ and is used as an estimate:

$$\hat{\lambda}_{max} = \| \mathbf{K} \|_\infty = \max_i \sum_j |k_{ij}| \tag{4.12}$$

To evaluate $\hat{\lambda}_{min}$, we estimate $\lambda_{max}(\mathbf{K}_0^{-1})$ using Lanczos method, where $\mathbf{K}_0^{-1}$ is a stiffness matrix of a lower order system. A plot of polynomial order versus the minimum eigenvalue (λ_{min}) illustrated in figure 4.3 for four problems(Car, Canoe,V-Block, and Flange) indicates that the smallest eigenvalue of a lower order system is a good estimate for the entire system.

The sparsity of the stiffness matrix $\mathbf{K}$ is quantified in terms of the average bandwidth. The average bandwidth of the system is defined as the ratio of estimated

total number of non-zeros in the Cholesky factor of $\mathbf{K}$, to the total number of equations. In case of an envelope method, such as the skyline solver, the total number of non-zeros in the Cholesky factor can be estimated as the size of the skyline storage. For a sparse solver the total number of non-zeros can be estimated from the graph model of the symmetric matrix [25]. The average bandwidth is calculated as:

$$\varpi = \frac{nonzero(\mathbf{L})}{NDOFS} \tag{4.13}$$

where $\mathbf{L}$ is the Cholesky factor of stiffness matrix $\mathbf{K}$, $NDOFS$ are the total number of equations and $nonzero(\mathbf{L})$ is the total number of nonzero terms in the matrix $\mathbf{L}$. The computation of $nonzero(\mathbf{L})$ for a sparse direct solver involves considerable symmetric graph manipulations, and hence is computationally expensive. Instead, an estimate to $nonzero(\mathbf{L})$ can be expeditiously calculated. Consider $\mathbf{L}_0$, the Cholesky factor of $\mathbf{K}_0$, the estimate for $nonzero(\mathbf{L})$ is then calculated as:

$$nonzero(\mathbf{L}) = \frac{nonzero(\mathbf{L}_0)}{nonzero(\mathbf{K}_0)} \times nonzero(\mathbf{K}) \tag{4.14}$$

This estimate for $nonzero(\mathbf{L})$ has been found to be remarkably accurate, since for hierarchic systems the ratio $\frac{nonzero(\mathbf{L})}{nonzero(\mathbf{K})}$ remains practically constant for all polynomial orders. A plot of $\frac{nonzero(\mathbf{L})}{nonzero(\mathbf{K})}$ versus polynomial order for four problems(Car, Canoe,V-Block, and Flange) (shown in figure 4.4) corroborates this fact.

The total storage requirements (μ_i) for different multilevel methods are estimated as follows:

- Multilevel method with SGS smoothing :

  $\mu_1 \approx$ storage for CLC + sparse storage of $\mathbf{K}$.

- Multilevel method with incomplete factorization smoothing (no fill ins) :

  $\mu_2 \approx$ storage for CLC + sparse storage of $\mathbf{K}$ + sparse storage for ICC.

- Multilevel method with block diagonal smoothing :

$\mu_3 \approx$storage for CLC + sparse storage of $\mathbf{K}$ + sparse storage for block diagonal $\mathbf{K}$.

- Single level method :

  $\mu_4 \approx$storage for Cholesky factor of $\mathbf{K}$ + sparse storage of $\mathbf{K}$.

The storage for CLC includes sparse storage of the Cholesky factor of size $nonzero(\mathbf{L}_0)$ and an integer index array of a smaller size. The sparse storage of stiffness matrix is given in terms of sorted Block Sparse Row (BSR) format, where the blocks correspond to the dofs associated with each mode. Hence the storage for $\mathbf{K}$ consists of storage for $nonzero(\mathbf{K})$ and a node by node index integer array of size $\{\frac{nonzero(\mathbf{K})}{dofs\ per\ node}\}$. Storage for ICC is given in terms of standard Compressed Sparse Row (CSR) format and consists of sparse storage of incomplete factor of size $nonzero(\mathbf{K})$ and an integer pointer array of same size. The estimated storage does not take into consideration the work arrays temporarily allocated during the program run. A plot of $\frac{max.\ heap\ memory}{estimated\ memory}$ versus maximum heap memory of all methods for the four representative problems is illustrated in figures 4.5 and 4.6. The maximum heap memory ($max.\ heap\ memory$) is the largest memory allocated during the program run. It is evident from the plot that the estimation of memory requirements is quite accurate especially for large problems.

4.3.2 Optimal Multilevel Preconditioned method

To aid in the selection of an optimal multilevel preconditioned method decision graphs are constructed by conducting numerical experiments for three factors: sparsity pattern, condition number and maximum available memory. These factors are considered simultaneously enabling interaction between them. The decision graphs are scatter plots with superimposed discriminant functions. The discriminant functions classify the set of sample problems into subsets, each of which can be solved in

optimal CPU times with a single solution method. The discriminant functions can be determined graphically, or by carrying out complete discriminant analysis [31].

An important attribute of the multilevel preconditioned method is the number of hierarchic levels. For the p-version the number of levels can be chosen in the range of [1, P], where P is the maximum polynomial order of the hierarchic stiffness matrix. In the current investigation only one and two level schemes are considered.

First, we estimate what must be the size of the coarse level in a two-level scheme which would result in minimum total CPU time. A plot of normalized CPU time (total cpu time/min. total cpu time) versus normalized coarse level polynomial order (CLC polynomial order/highest polynomial order) is constructed for the four representative problems. It is evident from the plot (shown in figure 4.7) that the selection of the coarse level polynomial order as one half the highest polynomial order is nearly optimal in terms of total CPU time. For the set of Legendre interpolation functions used, this results in a coarse level that has approximately one half the total number of equations for maximum polynomial order P in the range of $2 \leq P \leq 12$.

The selection of acceleration scheme depends on the condition number of the system to be solved. A plot of condition number versus the ratio of number of cycles with CG acceleration to the number of cycles with two parameter acceleration is shown in figure 4.7. It is evident from the plot that the two parameter acceleration scheme results in lesser number of iterations for ill-conditioned problems ($\hat{\kappa} > 10^7$). On the other hand for well posed problems both the acceleration schemes result in same number of iterations. However the CG acceleration scheme involves fewer scalar product evaluations.

Finally decision graphs are developed (figures 4.8, 4.9) for selecting an optimal multilevel solution strategy for four maximum memory limits:

- 50 MB.
- 100 MB.

- 200 MB.
- 600 MB.

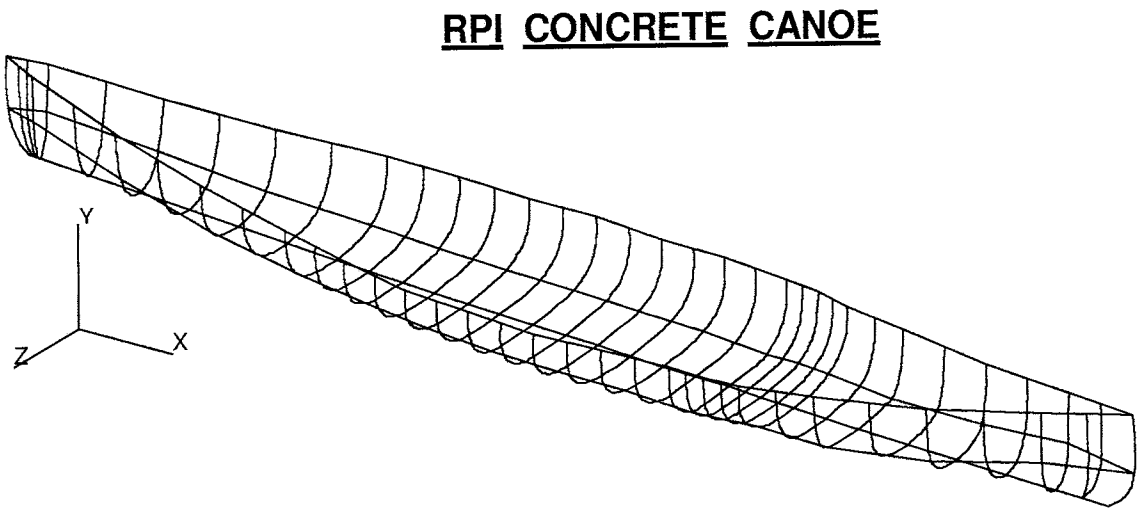
The following 11 solution techniques have been considered:

- 2-level scheme with Block diagonal incomplete factorization[ICC(P)] smoothing.
- 2-level method with Block diagonal incomplete factorization[ICC(V)] smoothing.
- 2-level method with Block diagonal full factorization smoothing.
- 2-level method with Incomplete factorization by value [ICC(V)] smoothing up to polynomial order 1 and ICC(P) for higher orders ($p > 1$).
- 2-level method with Incomplete factorization by value ICC(V) smoothing up to polynomial order 2 and ICC(P) for higher orders ($p > 2$).
- 2-level method with Incomplete factorization by value ICC(V) up to polynomial order 3 and ICC(P) for higher orders ($p > 3$).
- 2-level method with Incomplete factorization by value ICC(V) smoothing for all polynomial orders.
- 2-level method with ICC(P) smoothing for all polynomial orders.
- 2-level scheme with Symmetric Gauss Seidel (SGS) smoothing.
- 1-level sparse direct solver.
- 1-level skyline solver.

The optimal solution method in terms of total CPU time for each problem characterized by average bandwidth and condition number, is plotted in figures 4.8

and 4.9. Only five solution techniques amongst the ten listed above were found to be optimal for at least one problem considered. Thus the space of decision graph is divided into five regions corresponding to the five optimal solution strategies with some minor overlap along the boundaries. The Single level method (sparse direct solver) is represented by a circle, the two level method with (ICC(V) up to $p \leq 2$ and ICC(P) for $(p > 2)$) smoothing scheme is represented by square. The two level preconditioned method with incomplete factorization (ICC(V) for $p \leq 1$ and ICC(P) for $(p > 1)$)smoothing scheme is denoted by triangle. The two level preconditioned method with block diagonal incomplete factorization (ICC(P)) smoothing is represented by diamond, and finally the 2-level scheme with SGS smoothing is represented by an inverted triangle. When a new problem is encountered the optimal solver is determined based on the estimates of the condition number, average bandwidth and memory considerations for the given problem.

Finally a plot of the problem size versus CPU time in seconds is illustrated in figures 4.10, 4.11 for the four representative problems solved using different 2-level, 1-level and the adaptive solver. The runs were made on an Sun Sparc 5, 110 MHz workstation with 600 MB memory.



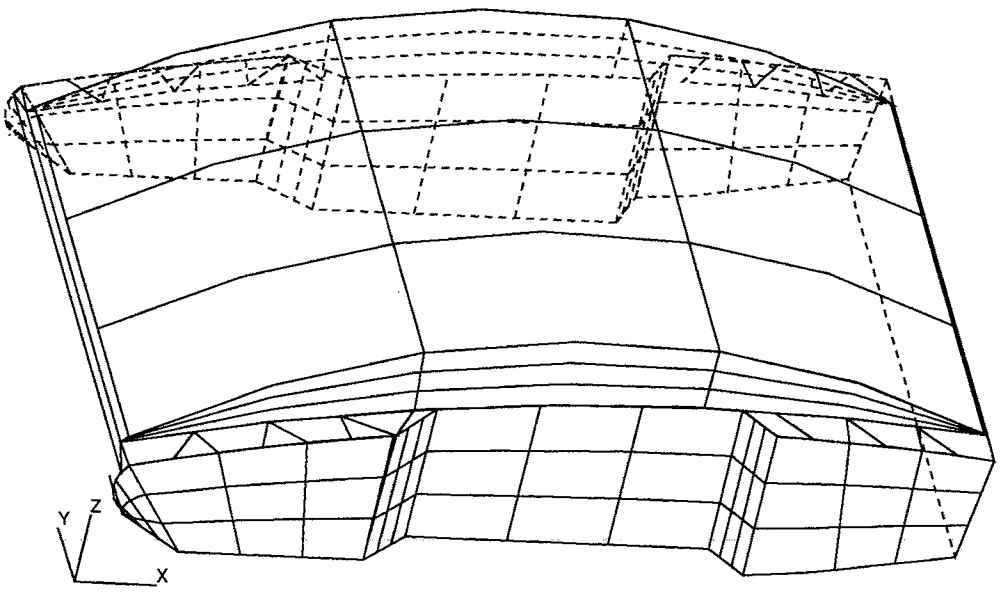


Figure 4.1: Shell Models : RPI Concrete Canoe and RPI Hybrid Car

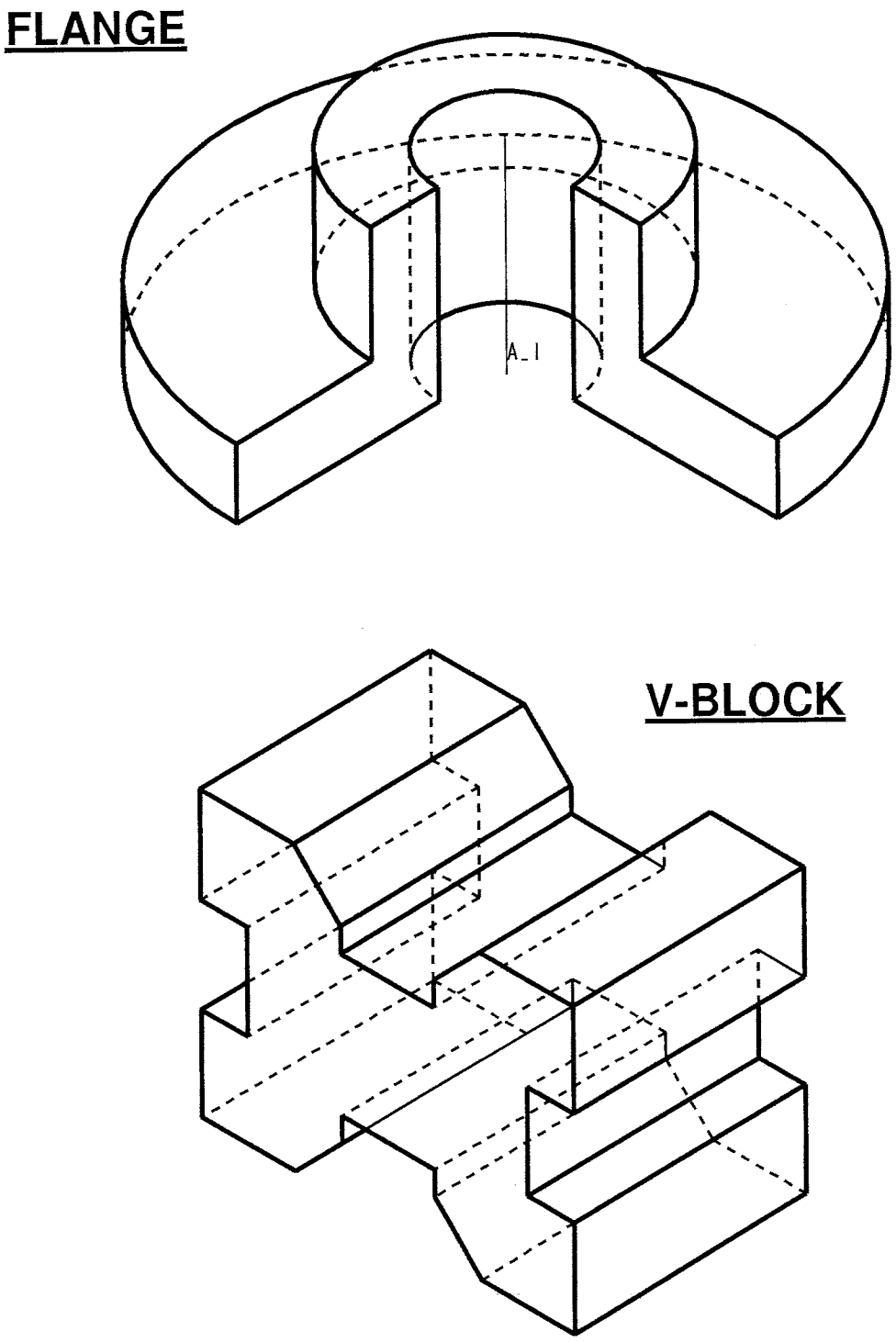


Figure 4.2: 3-D Models : Flange and V-Block

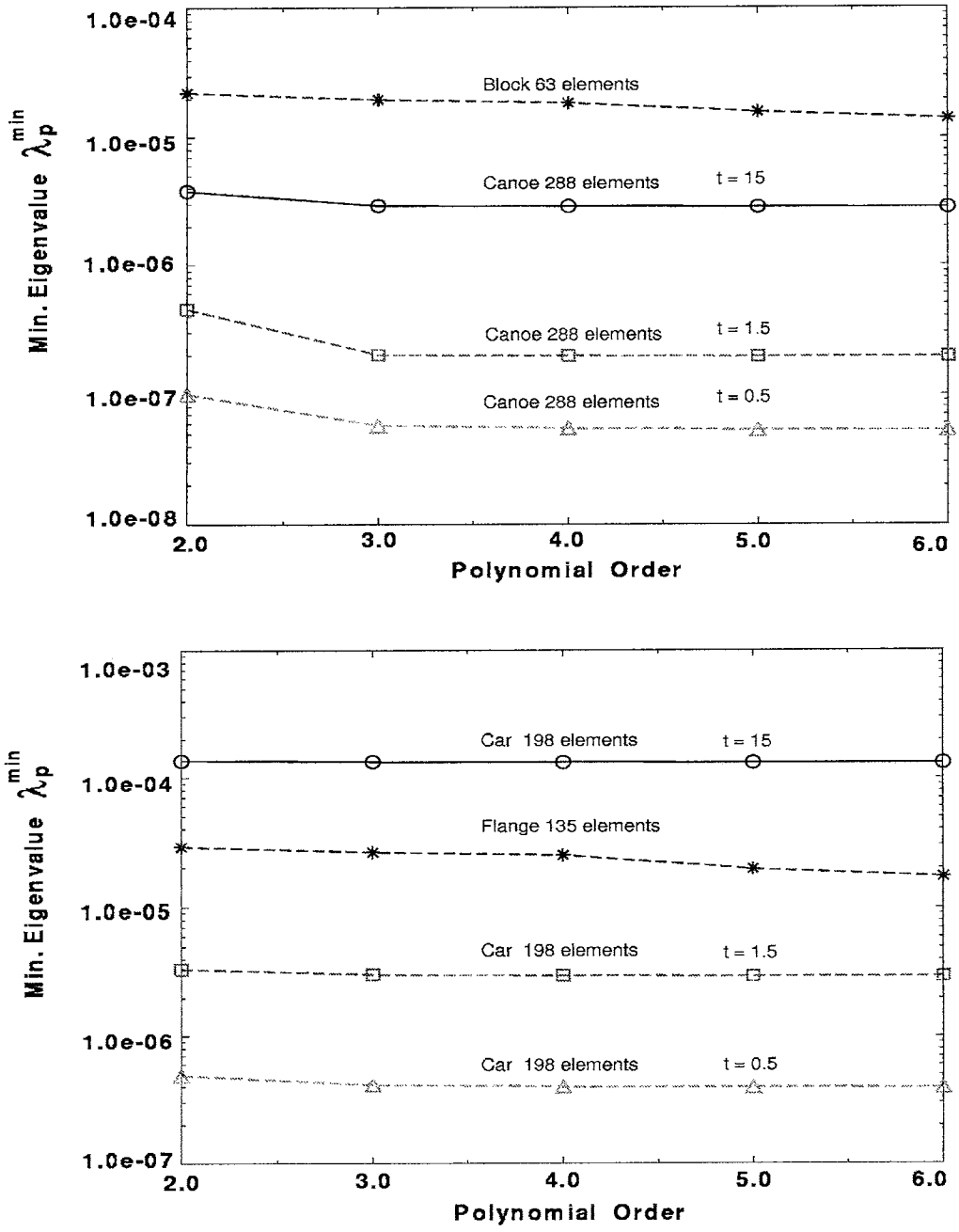


Figure 4.3: Plot of Min. Eigenvalue versus Polynomial Order

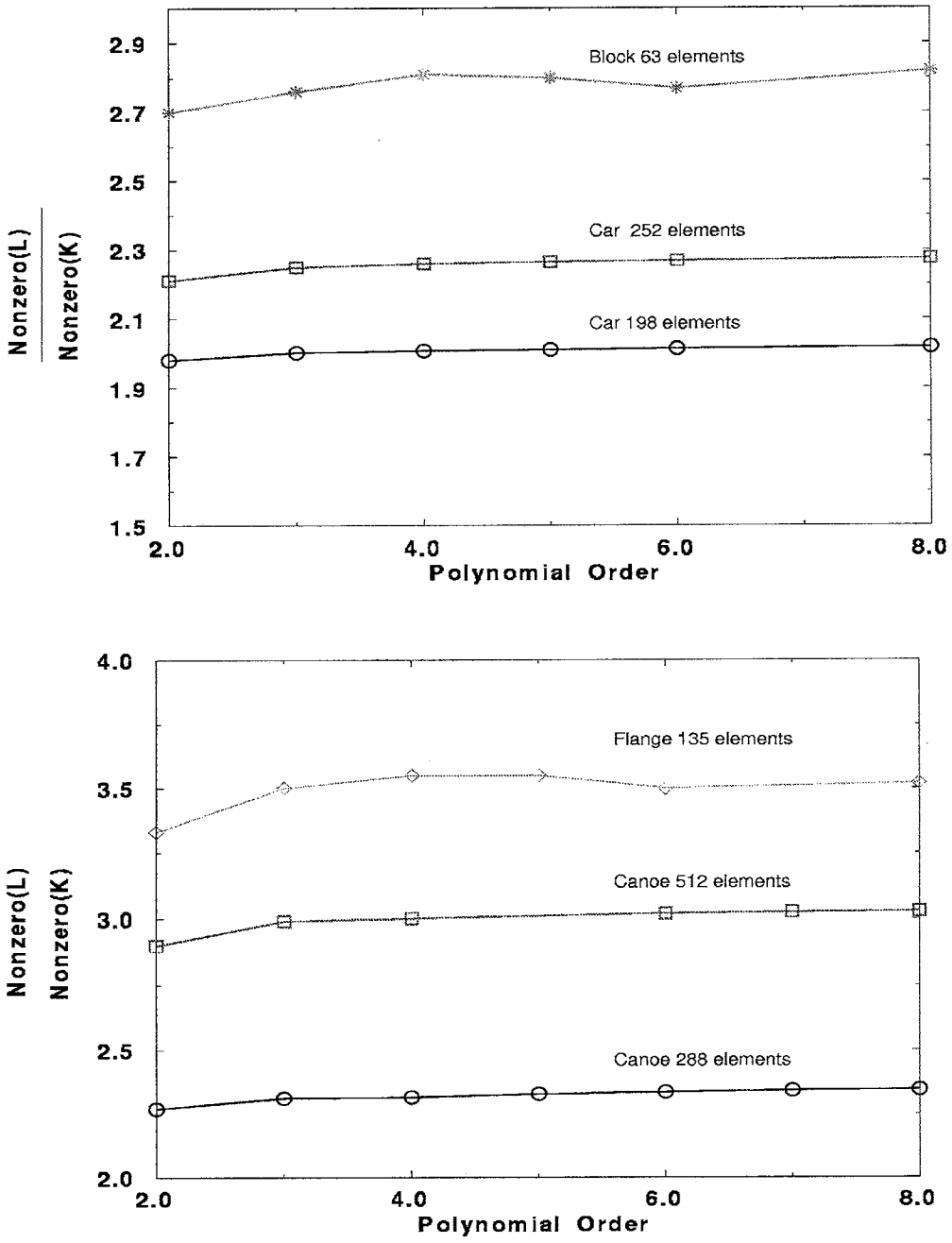


Figure 4.4: Plot of Sparsity Ratio to Polynomial Order

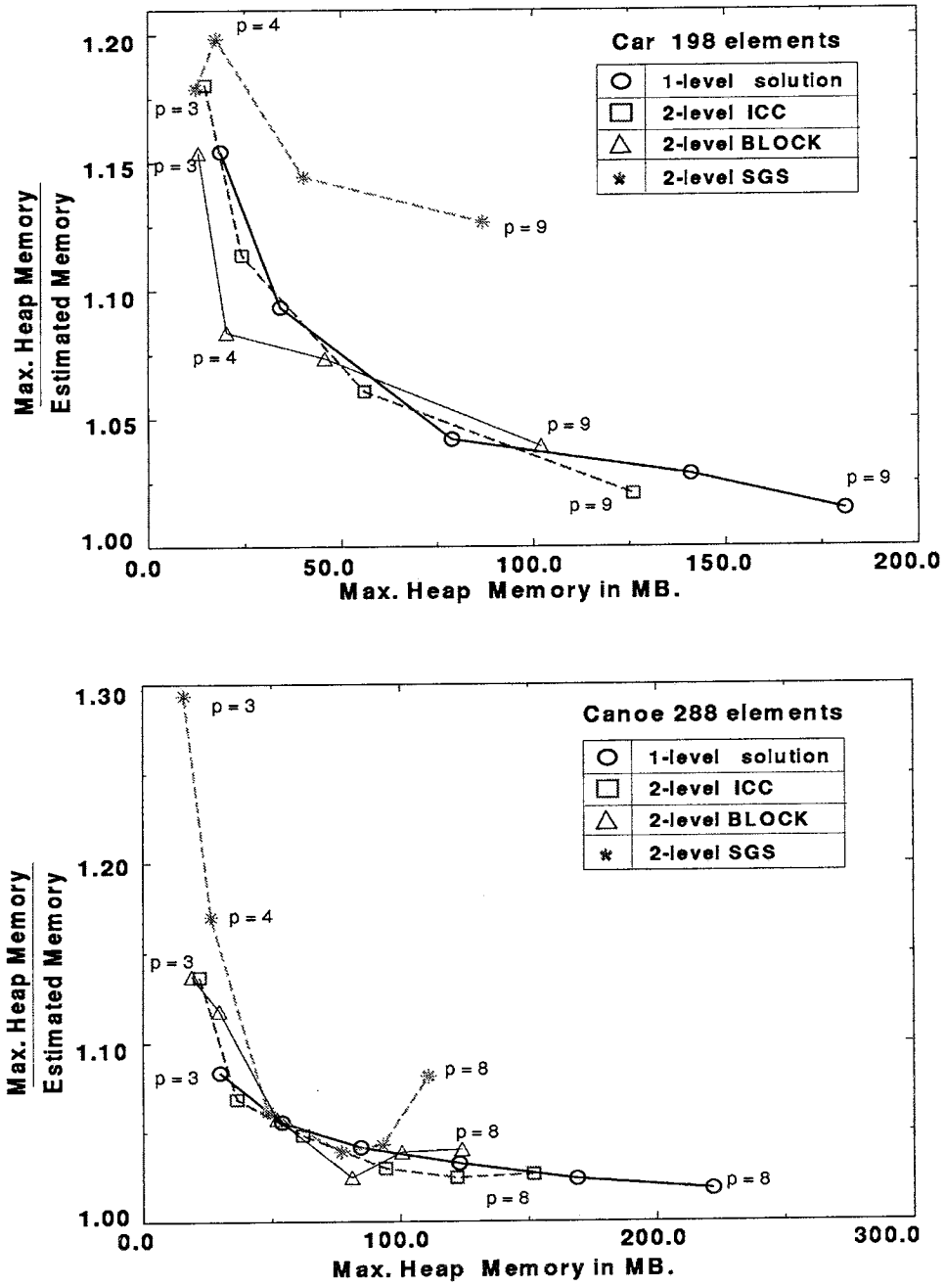


Figure 4.5: Plot of Estimated Memory versus Max. Heap Memory for Shell problems

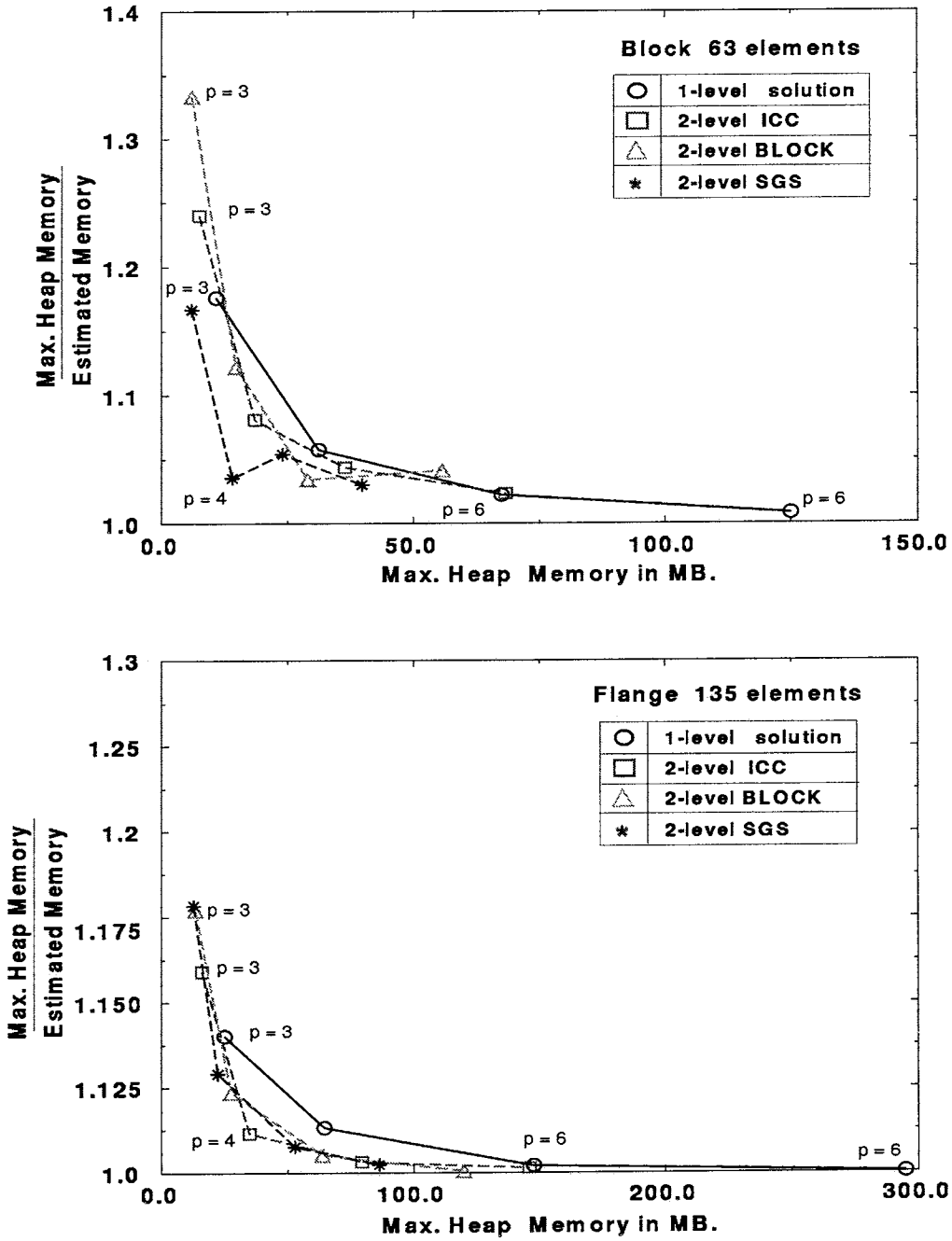


Figure 4.6: Plot of Estimated Memory versus Max. Heap Memory for 3-D problems

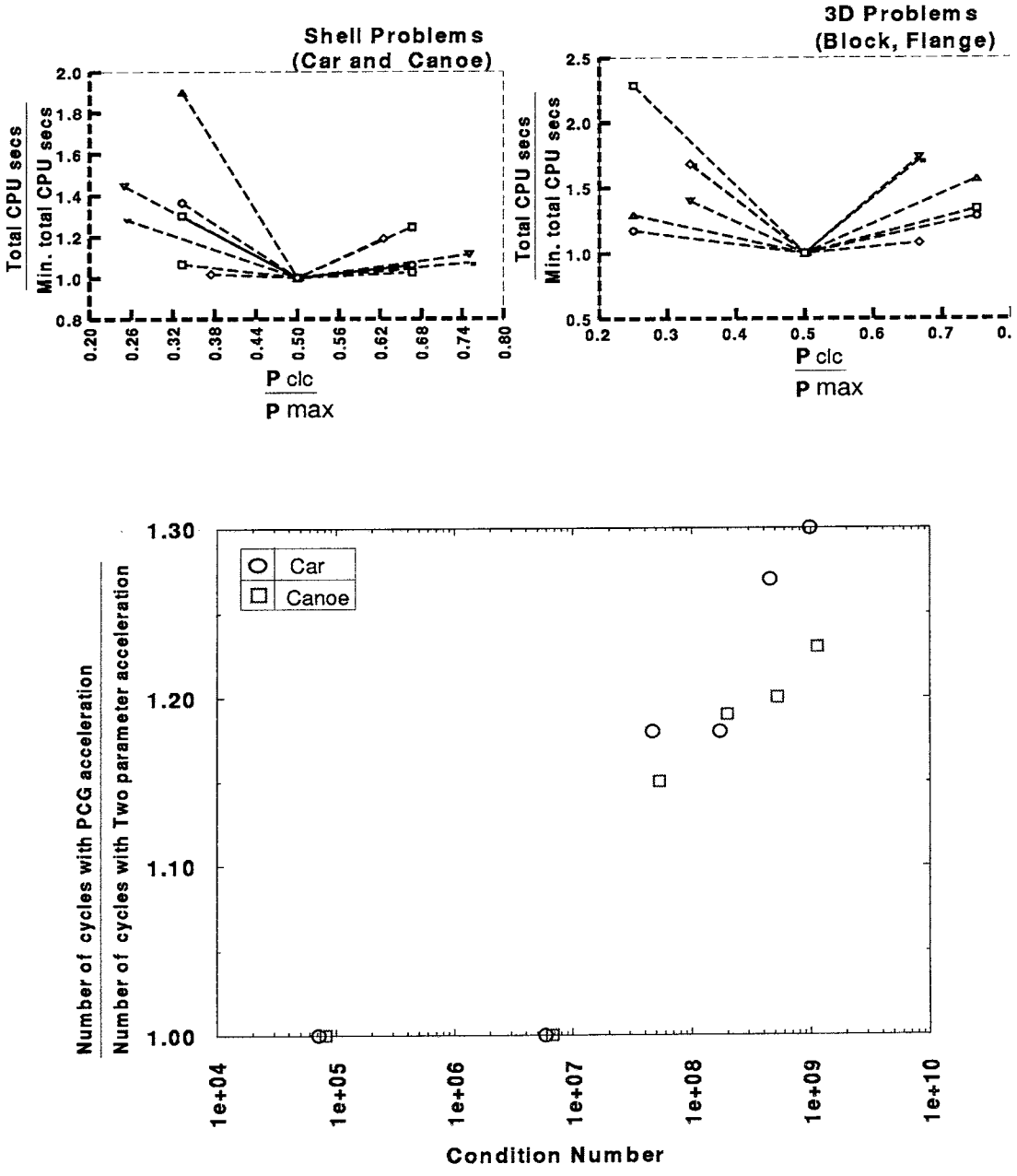


Figure 4.7: Selection of CLC Polynomial Order and Type of acceleration

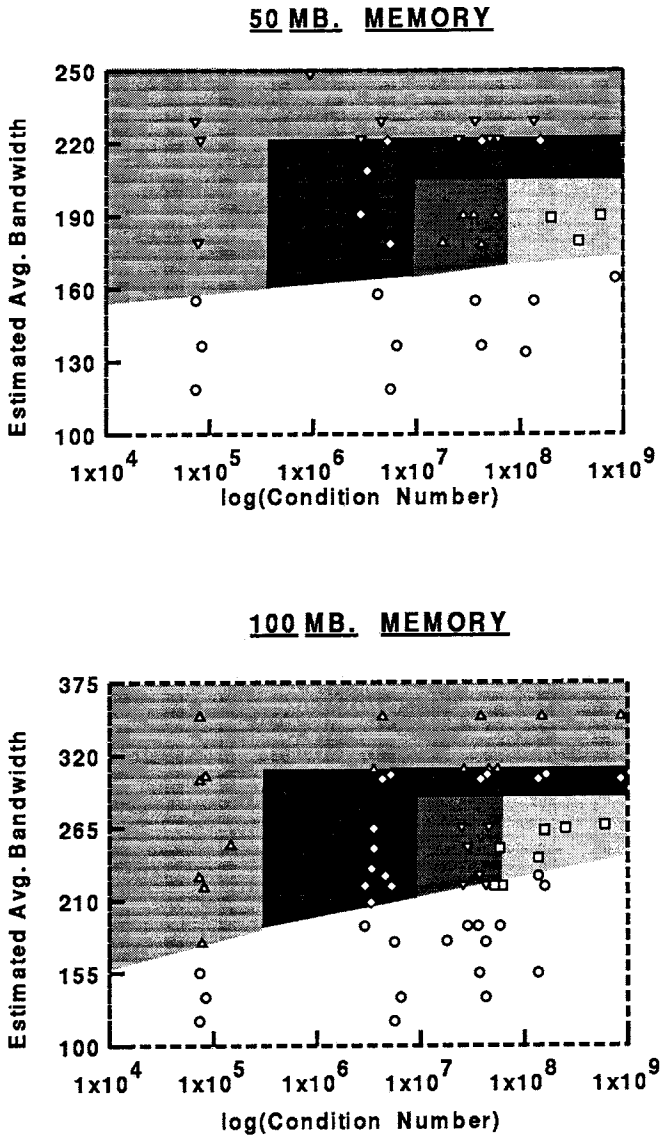


Figure 4.8: Decision Graphs

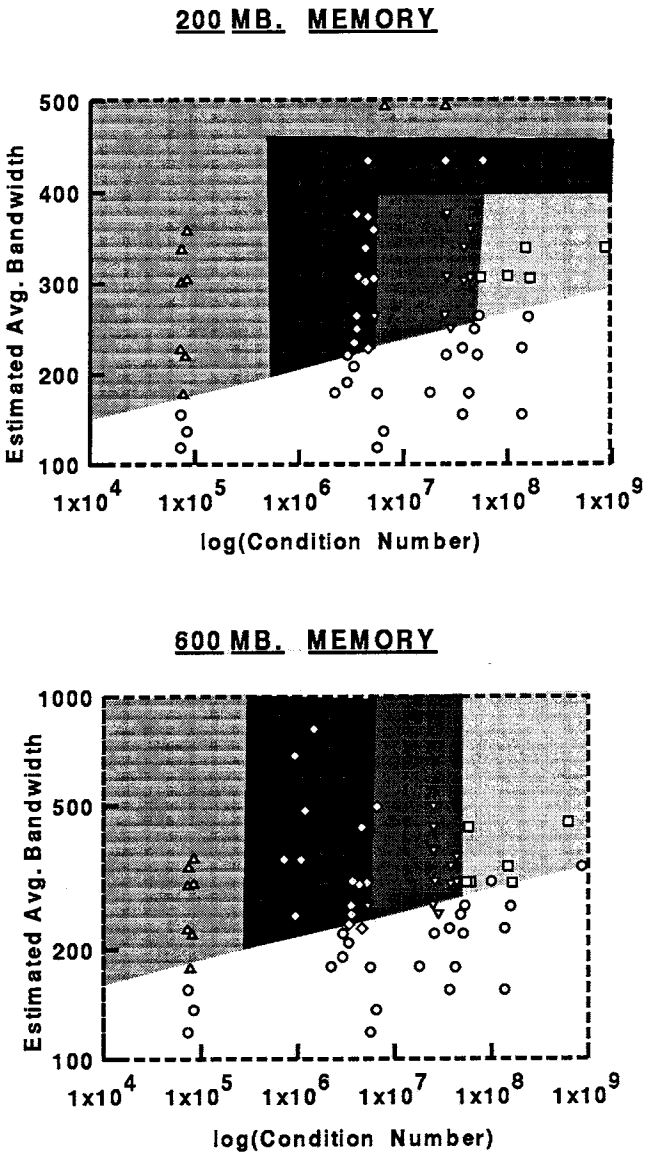


Figure 4.9: Decision Graphs

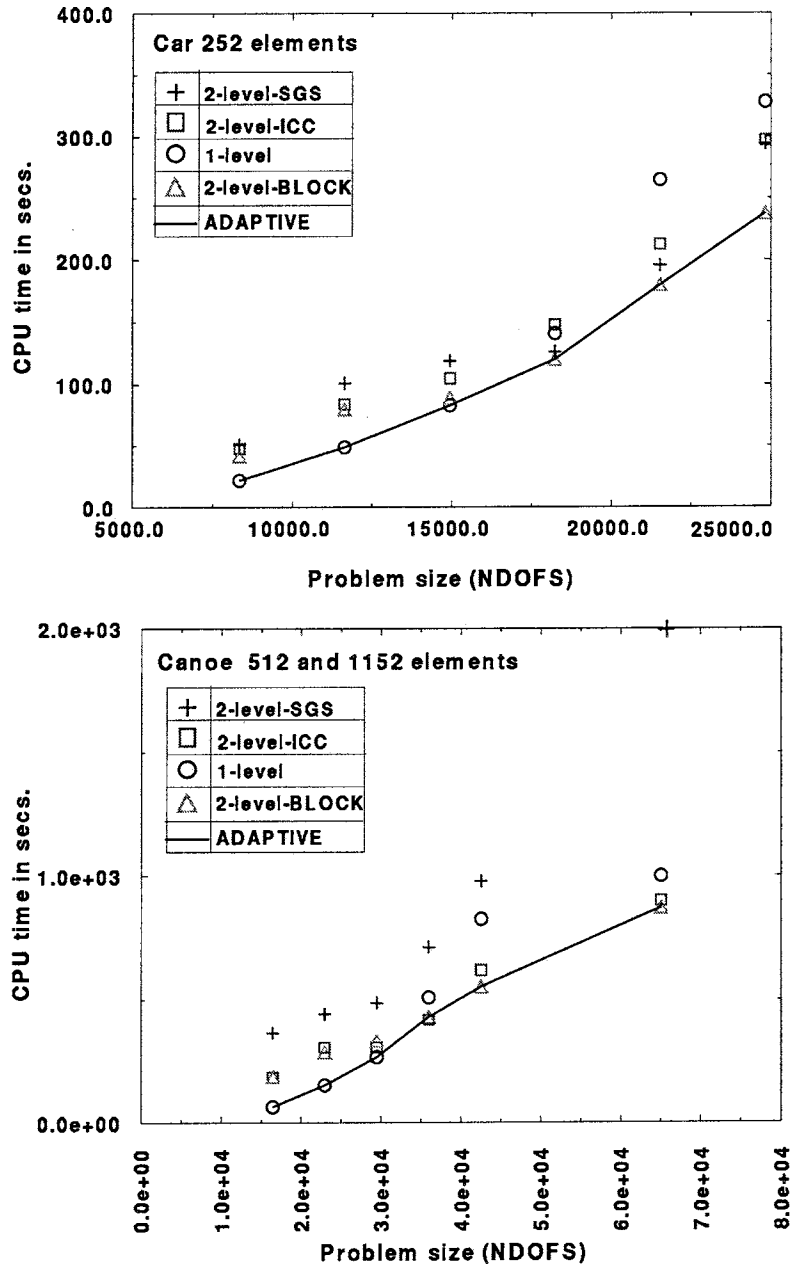


Figure 4.10: Performance of various Linear Solvers for Shell Problems

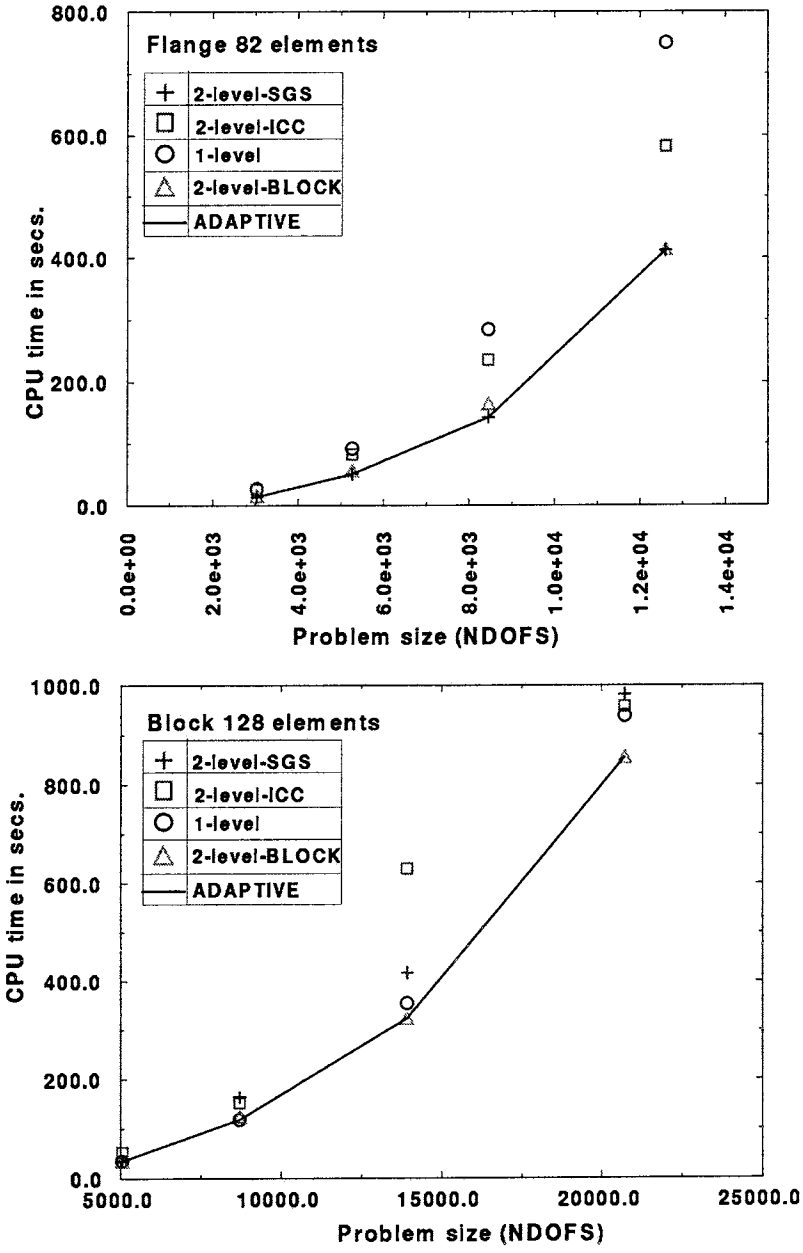


Figure 4.11: Performance of various Linear Solvers for 3-D Problems

CHAPTER 5
Summary and Future work

5.1 Summary and conclusions

Research efforts were made to optimize the computational efficiency of the p-method for shell analysis. A new quadrature scheme for rectangular and hexahedral elements and a family of hierarchical assumed strain based shell elements were introduced. Numerical experiments were conducted to evaluate the computational performance of the novel quadrature scheme. The p-type assumed strain shell elements were numerically tested. It was observed that the lower order ($p \leq 4$) p-type shell elements were not devoid of membrane and shear locking.

A hierarchical global-local strategy based on the s-version of finite element method for analysis of 3-D laminated plate and shell models was investigated. Advances in terms of computational efficiency were introduced. *A posteriori* error estimators were used for demarcation of critical local regions and polynomial refinement was carried out in both local and global regions. Numerical examples highlighted the utility of the s-method for hierarchic global-local analysis of laminated plates and shells.

Research efforts were conducted to develop an adaptive multilevel solution strategy for solving hierarchic systems resulting from the p-version of finite element discretization. The decision graph methodology aimed at determining an optimal multilevel solution strategy was developed.

5.2 Future work directions

Although an in-depth study of computational efficiency aspects of the hierarchic finite element methods was conducted, clearly there is a scope for future

enhancements. The quadrature scheme needs to be extended to account for material and geometric nonlinearities, as well as triangular and tetrahedral element domains. A hierarchic shell element with drilling degrees-of-freedom has also been developed [27].

Only in-core solution methods were considered as possible alternative methods. Clearly an ultimate solution engine needs to include an efficient out-of-core solver, since for very large problems, it is not usually possible to keep the stiffness matrix in RAM. Both iterative and direct out-of-core solvers can be investigated in the context of the p-method.

LITERATURE CITED

[1] Ahmed,N.U.;Basu,P.K.(1993):Higher Order Modeling Of Plates By p-Version Of Finite Element Method. J. Engg. Mech. ASCE. 119, 1228-1242.

[2] Ajiz,M.A.;Jennings,A.(1984):A robust incomplete Choleski-conjugate gradient algorithm. Int. J. for Num. Meth. in Engg. 20, 949-966.

[3] Axelsson,O.(1986):Analysis of incomplete matrix factorizations as multigrid smoothers for vector and parallel computers. Applied Mathematics and Computation, 19, 3-22.

[4] Axelsson,O.;Gustafsson,I.(1983):Preconditioning and two-level multigrid methods of arbitrary degree of approximation. Math. of Computation. 40, 219-242.

[5] Babuska,I.;Szabo,B.A.;Katz,I.N.(1981):The p-version of the finite element method. SIAM J. Numer. Anal. 18, 512-545.

[6] Babuska,I.;Stroubulis,S.;Upadhyay,C.S.;Gangaraj,S.K.(1992): A posteriori estimation and adaptive control of the polluting error in the h-version of the finite element method. Technical note BN-1175, Institute for Physical Science and Technology, Univ. of Maryland, College Park, MD.

[7] Babuska,I.;Craig,A.;Mandel,J.;Pitkaranta,J.(1991): Efficient preconditioning for the p-version finite element method in two dimensions. SIAM J. Numer. Anal. 28, 624-661.

[8] Bank,R. E.;Dupont,T. F.;Yserentant,H.(1988):The hierarchical basis multigrid method. Numer. Math. 52, 427-458.

[9] Belytschko,T.;Stolarski,H.;Liu,W.K.;Carpenter,N.;Ong,J.S-J.(1985):Stress projection for membrane and shear locking in shell finite elements. Comp. Meth. in Appl. Mech. and Engg. 51, 221-258.

[10] Belytschko,T.;Fish,J.;Bayliss,A.(1990):The spectral overlay on finite elements for problems with high gradients. *Comp. Meth. Appl. Mech. Eng.*, 81, 71-89.

[11] Bramble,J.;Ewing,R.E.;Pasciak,J.E.;Schatz,A.H.(1988):A preconditioning technique for the efficient solution of problems with local grid refinement.*Comp. Meth. Appl. Mech. Eng.*, 67, 149-159.

[12] Brandt,A.(1977):Multi-level adaptive solutions to boundary-value problems. Mathematics of Computation, 31, 333-390.

[13] Carnevali,P.;Morris,R.B.;Tsuji,Y.;Taylor,G.(1992): New Basis Functions and Computational Procedures for p-type Finite Element Analysis. RJ 8710 (78272) Engineering Technology. April 3, 1992.

[14] Cook, R.D.(1981):Concepts and Applications of Finite Element Analysis. Wiley,New York.

[15] Fish,J.;Markolefas,S.;Guttal,R.;Nayak,P.(1994):On adaptive multilevel superposition of finite element meshes for linear elastostatics. *Appl. Numer. Math.*, 14, 135-164(1994).

[16] Fish,J.(1990):The s-version of the finite element method, SCOREC report #18-1990, Rensselaer Polytechnic Institute Troy, NY-12180; and *Comput. and Struct.*, 43, 539-547.

[17] Fish,J(1992):Hierarchical modeling of discontinuous fields. Comm. Appl. Numer. Methods, 8, 443-453.

[18] Fish,J.;Markolefas, S.(1992):The s-version of the finite element method for multilayer laminates *Int. J. Numer. Methods Eng.*, 33, 1081-1105.

[19] Fish,J.;Guttal,R.(1995):The p-version of Finite Element Method for Shell Analysis. *Comp. Mech. Int. Journal*, 16, 1-13.

[20] Fish,J.;Guttal,R.(1995):The s-version of Finite Element Method for Laminated composites. to be published in *Int. J. Numer. Methods Eng.*

[21] Fish,J.;Markolefas,S.(1994):Adaptive global-local refinement strategy based on interior error estimates of the h-method. *Int. J. Numer. Methods Eng.*, 37, 828-838.

[22] Fish,J.;Belsky,V.(1994):Mutligrid method for periodic heterogeneous media. Part 2: Multiscale modeling and quality control in multidimensional case.accepted in *Comp. Meth. Appl. Mech. Eng.*.

[23] Fish,J.;Belsky,V.;Pandheeradi,M.(1995):Composite grid method for hybrid systems. submitted to *Comp. Meth. Appl. Mech. Eng.*.

[24] Flaherty,J.E.;Moore.P.K.;Ozturan,C.(1989):Adaptive overlapping methods for parabolic systems. in Adaptive Methods for Partial Differential Equations (Eds. J. E. Flaherty, P. J. Paslow, M. S. Shephard and J. D. Vasilakis), SIAM.

[25] George,A.;Liu,J.W-H.(1981):Computer solution of Large Sparse positive definite systems. Prentice-Hall, Inc. Englewood Cliffs, NJ.

[26] Gordon, W.J.;Hall, C.A.(1973):Construction of curvilinear coordinate systems and applications to mesh generation. *Int. j. numer. methods eng.*, 7, 461-477.

[27] Guttal,R.;Fish,J.(1996):A Hierarchic shell element with drilling degrees-of-freedom. *under preparation.*

[28] Hackbusch,W.(1994):Iterative Solution of Large Sparse System of Equations. Springer-Verlag, New York, NY.

[29] Holzer,S.;Rank,E.;Werner,H.;(1990):An implementation of the *hp*-version of the finite element method for reissner-mindlin plate problems. *Int. j. numer. methods eng.*, 30, 459-471.

[30] Hinnant,H.E.(1993):A Fast method of numerical quadrature for *p*-version finite element matrices. A.I.A.A. 1386.

[31] Huberty,C.J.(1994)Applied discriminant analysis. New York : Wiley.

[32] Hughes,T.J.R.(1987):The Finite Element Method. Prentice-Hall.

[33] Kelly, D.W.(1984):The self-equilibration of residuals and complementari a posteriori error estimates in finite element method. Int. j. numer. methods eng. 20, 1491-1506.

[34] Kershaw,D.S.(1968):The incomplete Choleski-conjugate gradient method for the iterative solution of systems of linear equations. J. Comp. Phys. 26, 43-65.

[35] Knight,N.F.;Ransom,J.B.;Griffin,O.H.;Thompson,D.M.(1991):Global/Local methods research using a common structural analysis framework. *Finite Element Analysis and Design,* 9, 91-112.

[36] Lo,K.H.;Christensen,R.M.;Wu,E.M.(1977):A higher order theory of plate deformation. J. Appl. Mech. 44, 669-676.

[37] Mandel,J.(1993):Adaptive Iterative solvers in Finite Elements. Solving Large-scale problems in Mechanics, edited by M. Papadrakakis, John Wiley and Sons Ltd., 65-88. New York, NY.

[38] Mao,K.M.;Sun,C.T.(1991):A refined global-local finite element analysis method. *Int. J. Numer. Methods Eng.*, 32, 29-43.

[39] McCormick,S.F.;Thomas,J.W.(1986):The fast adaptive composite grid (FAC) method for elliptic equations. *Mathematics of Comput.*, 46, 439-456.

[40] Morris,R.B.;Tsuji,Y.;Carnevali,P.(1992):Adaptive solution strategy for solving large systems of *p*-type finite element equations. Int. Jou. of Num. Methods in Engg, 33, 2059-2071.

[41] Noor,A.K.;Burton,W.S.;Peters,J.M.(1990):Predictor-Corrector procedures for stress and free vibration analyses of multilayered composite plates and shells. *Comp. Meth. Appl. Mech. Eng.*, 82, 341-363.

[42] Peano,A.G.(1976):Hierarchies and Conforming Finite Elements for Plane elasticity and Plate Bending. Comp. and Maths. 2, 211-224.

[43] Raju,I.S.;Crews,J.H.(Jr.);Aminpour,M.A.(1987):Convergence of Strain Energy release rate components for Edge-Delaminated Composite Laminates. NASA Technical Memorandum 86135.

[44] Ramm,E.;Stander,N.;Matzenmiller,A.;(1989):An assessment of Assumed strain methods in finite rotation shell analysis. Eng. Comput. 6.

[45] Reddy,J.N.;Robbins,D.H.(1994):Theories and computational models for composite laminates. *Appl. Mech. Rev.*, 47, 147-169.

[46] Robbins,D.H.;Reddy,Y.S.N.;Reddy,J.N.(1991):Analysis of Inter-laminar Stresses and Failures using a Layer-Wise Laminate Theory, Local Mechanics Concepts For Composite Material Systems. Springer-Verlag, Eds. J.N. Reddy and K.L. Reifsnider, 309-340.

[47] Park,K.C.;Stanley,G.M.(1986): A curved C^0 Shell element based on Assumed Natural Coordinate Strains. J. of Appl. Mech. 108, 278-290.

[48] *PROBE* Theoretical Manual. release 1.0 Neotic Technologies Corp 1985.

[49] Shephard,M.S.; Dey, S.(1994):Geometric mapping of finite elements on shell geometry. SCOREC report 15.

[50] Shephard,M.;Niu,Q.;Baehmann,P. L.(1989):Some results using stress projectors for error indication and estimation. In J. E. Flaherty, P. Y. Paslow, M. S. Shephard and Y. D. Vasilakis, eds. Adaptive Methods for Partial Differential Equations, SIAM. 1-14.

[51] Shivakumar,K.N.;Tan,P.W.;Newman,J.C.(1988):A Virtual Crack Closure Technique for Calculating Stress-Intensity factors for Cracked Three-dimensional Bodies. *Int. J. of Fracture,* 36, R43-R50.

[52] Stanley,G.M.;Levitt,I.;Stehlin,B.;Hurlbut,B. (1992): Adaptive Analysis of Composite Shell Structures via Thickness-tailored 3D Finite Elements. Winter Annual Meeting Anaheim, CA, November 11-13. 1992.

[53] Surana,K.S.;Sorem,R.M.;(1991):p-version hierarchical three dimensional curved shell element for elastostatics. Int. j. numer. methods eng. 31, 649-676.

[54] Szabo,B.A.;Sahrmann,G.J.(1988):Hierarchic plate and shell models based on p-extensions. Int. j. numer. methods eng. 26, 1855-1881.

[55] Szabo,B.A.;Babuska,I.;Chayapaty,B.K.(1989):Stress computations for nearly incompressible materials by the p-version of the Finite Element method. Int. j. numer. methods eng. 28, 2175-2190.

[56] Tinney,W.F.(1969):Comments on using sparsity techniques for power system problems. Sparse Matrix Proceedings, IBM Research Rept, RAI 3-12-69.

[57] Wang,S.S.;Choi,I.(1982):Boundary-Layer Effects in Composite Laminates: Part 2-Free-Edge Stress Solutions and Basic Characteristics. *J. of Appl. Mech,* 49, 549-560.

[58] Whitcomb,J.D.(1991):Iterative global-local finite element analysis. *Computers and structures,* 40, 1027-1031.

[59] J. D. Whitcomb, I. S. Raju and J. G. Goree "Reliability of the Finite Element Method for calculating Free edge stresses in composite laminates." *Computers and structures,* 15, 23-27 (1982).

[60] Yserentant,H.(1986):On multilevel splitting of finite element spaces. *Numer. Math.*, 379-412, (1986).

[61] Zeinkiewicz,O.C.(1977):The Finite Element Method. McGraw-Hill,London.

[62] Zeinkiewicz,O.C.; Craig, A.(1986):Adaptive Refinement, Error Estimates, Multigird Solution, and Hierarchic Finite Element Method Concepts. In Babuska,I. Zeinkiewicz, O.C. Gago, J. and Oliveira, E. R. de A. eds. Accuracy Estimates and Adaptive Refinements in Finite Element Computations