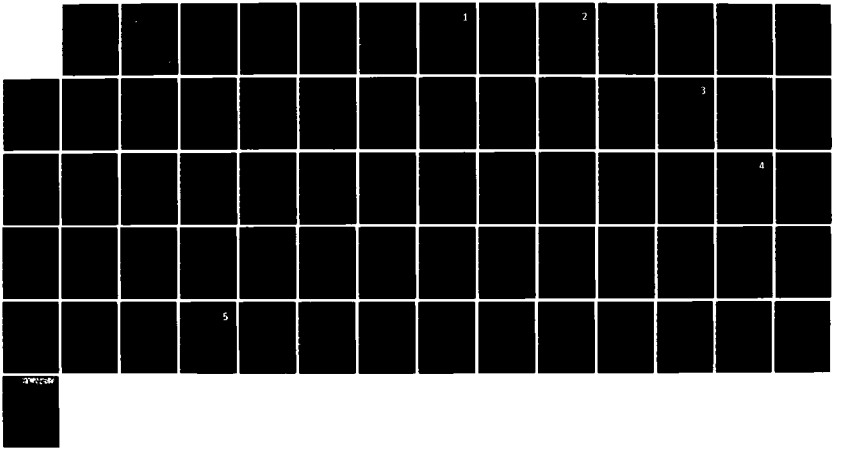
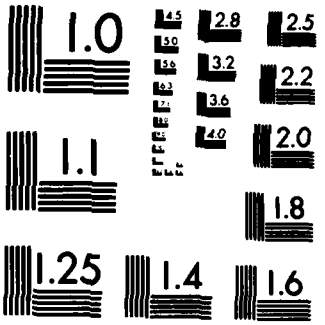
AD-A146 796 FINITE ELEMENT PROCEDURES APPLICABLE TO NONLINEAR ANALYSIS OF REINFORCED..(U) HUGHES (THOMAS J R) STANFORD CA G M STANLEY ET AL. SEP 84 NCEL-CR-84.033 1/1

UNCLASSIFIED N62583-82-M-T122 F/G 13/13 NL





(12)

CR 84.033

NAVAL CIVIL ENGINEERING LABORATORY
Port Hueneme, California

Sponsored by
NAVAL FACILITIES ENGINEERING COMMAND

FINITE ELEMENT PROCEDURES APPLICABLE TO NONLINEAR ANALYSIS OF REINFORCED CONCRETE SHELL STRUCTURES

September 1984

An Investigation Conducted by:
THOMAS J.R. HUGHES, INC.
Stanford, CA 94305

N62583/82 M T122

METRIC CONVERSION FACTORS

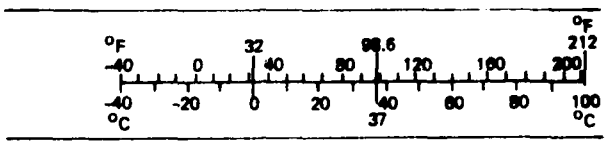
Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2,000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	*grams*	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1,000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	*cubic meters*	35	*cubic feet*	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER CR 84.033	2. GOVT ACCESSION NO. A140796 / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Finite Element Procedures Applicable to Nonlinear Analysis of Reinforced Concrete Shell Structures	5. TYPE OF REPORT & PERIOD COVERED Final Oct 1982 - Apr 1984 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) G.M. Stanley and T.J.R. Hughes	8. CONTRACT OR GRANT NUMBER(s) N62583/82 M T122
9. PERFORMING ORGANIZATION NAME AND ADDRESS Thomas J.R. Hughes, Inc. Stanford, CA 94305	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS YR023.003.01.001
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Civil Engineering Laboratory Port Hueneme, CA 93043	12. REPORT DATE September 1984 13. NUMBER OF PAGES 60
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Facilities Engineering Command 200 Stovall Street Alexandria, VA 22332	15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite element, shell elements, reinforced concrete, nonlinear structural analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The purpose of this report was to adapt earlier work in nonlinear shell-element methodology to the specific kinds of problems encountered in the analysis of critically-loaded reinforced concrete shell structures. The Continuum-Based nonlinear shell-element formulation for thin shells was extended to thicker, layer-model types of shell structures. A

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

resultant-type form of the basic equations was developed that is both appropriate for the treatment of thick, layered shells and leads to an efficient finite-element implementation as well. By eliminating most of the work associated with thickness integration, this approach should be effective when numerous layers and/or thickness integration points are required, as in the case of inelastically responding reinforced concrete. Pointwise constitutive relations for accurate through-thickness stress calculations are still employed though these are ultimately presented in the element computational routines in the form of resultants.

Contents

FINITE ELEMENT PROCEDURES APPLICABLE TO NONLINEAR ANALYSIS OF REINFORCED CONCRETE SHELL STRUCTURES

1

Introduction

The purpose of this report is to adapt our earlier work in nonlinear shell-element methodology [4–6,19] to the specific kinds of problems encountered in the analysis of critically-loaded reinforced concrete shell structures. To this end, we develop a resultant-type form of the basic equations (of motion) that is both appropriate for the treatment of thick, layered shells and leads to an efficient finite-element implementation as well. By eliminating most of the work associated with thickness integration, the approach should be particularly effective when numerous layers and/or thickness integration points are required, as in the case of inelastically responding reinforced concrete. Furthermore, adhering to a Continuum-Based approach, we continue to employ pointwise constitutive relations for accurate through-thickness stress calculations, though these are ultimately presented to the element computational routines in the form of resultants. Finally, the implementation of these new procedures represents a straightforward extension of our current shell-element software package.

The organization of the report is as follows. In Section 2, our previous work is reviewed, with the main intent of introducing all of the pertinent notation. Then in Section 3, the basic equations, used as the starting point for finite-element discretization in [19], are generalized to accomodate thick/layered shells. The impact of these revisions on the finite-element implementation is presented in Section 4, where an outline of the Global and Local solution algorithms is presented as well. Finally, a brief summary and conclusions are provided in Section 5.

2

Review of Previous Work

§2.1 Background

In reference [4], a Continuum-Based (CB) approach to nonlinear finite-element shell analysis is developed. These developments represent a generalization of the "degenerated-solid" element, introduced by Ahmad et al. [1] for linear analysis. The basic advantage of the CB approach is its convenient avoidance of complicated (and often ambiguous) shell equations, while at the same time providing a straightforward vehicle for finite-element discretization — via C^0 displacement interpolation. A second advantage is the natural inclusion of transverse-shear deformability, which increases the thickness regime over which the method is applicable.

The CB approach presented in [4] extends the capabilities to include arbitrarily large rotations, moderately large strains and a large class of (elastic/plastic) constitutive models. However, due to the full three-dimensional character of the equations — the thickness coordinate, $\tilde{z}$, appears explicitly in the element arrays — element formation can be quite expensive. This is especially true in the case of multi-layered inelastic shells, where thickness integration may constitute a substantial portion of the computing time. Thus, in [19], the thickness coordinate is *pre-integrated* out of the arrays by making several "thinness" assumptions, namely:

(A1) **Normality** of transverse fibers is assumed in any fixed configuration. (This does not preclude transverse shear deformation on an incremental basis.)

(A2) **Curvature** effects are approximated by using the average (reference surface) metric when computing kinematic quantities. (This is tantamount to *first approximation* shell theories [2,11] that neglect terms of order h/R with respect to 1.)

(A3) **Taper** of the shell wall is assumed to be gradual enough that: (i) derivatives of $\tilde{z}$ along the surface directions may be neglected (kinematically) and (ii) the 'zero normal stress' hypothesis may be enforced with respect to the *reference-surface* normal. (This does not preclude thickness jumps.)

These assumptions leads to resultant-oriented equations (and element arrays) that involve only surface integration. We therefore refer to the approach as the Continuum-Based Resultant (CBR) formulation — to distinguish it from classical Resultant-Based formulations.

The CBR approach is significantly more economical than the CB approach, but due to the above assumptions the advantages are biased towards "thin" shells, with a degradation of accuracy anticipated as h/R (current thickness to minimum radius of curvature) is increased.

In Section 3, the economic advantages of the CBR approach are extended to the "thick" shell regime. The remainder of the current section reviews the basic CB and CBR shell equations and notational pre-requisites to the sequel. For greater depth and detail, the reader is referred to [19].

§2.2 Continuum-Based (CB) Shell Theory

2.2.1 Continuum Equations

We start with the Linearized Variational (LV) equations governing the incremental motion of a general three-dimensional continuum [16,17]. Note that a Lagrangian (material-based) description of the motion is being employed and, unless otherwise specified, all vector and tensor components are expressed in a fixed (global) Cartesian basis, with indices i, j, k, l ranging from 1–3.

In operator notation, the LV equations may be written

$$\boxed{D\mathcal{M}(\Delta\ddot{\mathbf{u}}) + D\mathcal{F}(\Delta\mathbf{u}) = \mathcal{F}^{ext}(\mathbf{u}) - \mathcal{F}^{int}(\mathbf{u}) - \mathcal{M}(\ddot{\mathbf{u}})} \tag{2.1}$$

where $\mathcal{F}^{ext}$ is the external force operator, $\mathcal{F}^{int}$ the internal force operator, $\mathcal{M}$ the inertial force (or mass) operator, $D\mathcal{F}$ the linearized internal force (or stiffness) operator, and $D\mathcal{M}$ (the linearized mass operator) is identically equal to $\mathcal{M}$. The argument, $\mathbf{u}$, represents the material displacement vector, and $\ddot{\mathbf{u}}$, the corresponding acceleration vector — both considered to be functions of the material coordinates, X_i.

The incremental displacements and accelerations, $\Delta\mathbf{u}$ and $\Delta\ddot{\mathbf{u}}$, emanate from the linearization and are to be interpreted as excursions from the known (current) configuration

associated with u. Note that the entire right-hand-side of (2.1) is thus known, while the left-hand-side involves the *tangent* operators and the unknown Δu. Thus (2.1) constitutes the basis for subsequent iterative algorithms (Section 4).

In the current report, we will concentrate on the internal force and stiffness operators, $\mathcal{F}^{int}$ and $D\mathcal{F}$, since the external force and mass operators are relatively unaffected by the choice of shell formulation (i.e., CB or CBR). Furthermore, it is these operators that engender the bulk of the cost in subsequent finite-element numerical computations. Their continuum definitions are given as follows

$$\mathcal{F}^{int}(\mathrm{u}) = \int_V \delta\underline{\varepsilon}^T \underline{\sigma}\, dV \tag{2.2}$$

and

$$D\mathcal{F}(\Delta\mathrm{u}) = \underbrace{\int_V \delta\underline{\varepsilon}^T \underline{\underline{C}}\, \Delta\underline{\varepsilon}\, dV}_{D\mathcal{F}^{matl}} + \underbrace{\int_V \nabla(\delta u_i)^T \sigma \nabla(\Delta u_i)\, dV}_{D\mathcal{F}^{geom}} \tag{2.3}$$

where the integrals are performed over the current volume V of the body, with associated spatial coordinates, x_i.

In these expressions, $\Delta\underline{\varepsilon}$ is an incremental strain "vector" corresponding to the tensor

$$\Delta\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial \Delta u_i}{\partial x_j} + \frac{\partial \Delta u_j}{\partial x_i}\right) \tag{2.4}$$

and arranged in the order

$$\Delta\underline{\varepsilon} = \lfloor\, \Delta\varepsilon_{11},\ \Delta\varepsilon_{22},\ \Delta\varepsilon_{33},\ 2\Delta\varepsilon_{12},\ 2\Delta\varepsilon_{23},\ 2\Delta\varepsilon_{31}\, \rfloor \tag{2.5}$$

Similarly, $\underline{\sigma}$ is the "vector" corresponding to the Cauchy stress tensor, σ, and arranged in the order

$$\underline{\sigma} = \lfloor\, \sigma_{11},\ \sigma_{22},\ \sigma_{33},\ \sigma_{12},\ \sigma_{23},\ \sigma_{31}\, \rfloor \tag{2.6}$$

The symbol δ in (2.2) and (2.3) indicates a variational (or *virtual*) quantity, so that $\delta\underline{\boldsymbol{\varepsilon}}$ is defined in precisely the same manner as $\Delta\underline{\boldsymbol{\varepsilon}}$ (2.5), except with Δu_i replaced by δu_i, the virtual displacement. (Note: δu may also be viewed as the weighting function used to obtain the variational equations; hence, we endow it with *essentially* the same properties as Δu.)

Finally, $\underline{\mathbf{C}}$ is the "matrix" corresponding to the material response tensor, C. This tensor is associated with the rate constitutive equations presented in [7] (see also §4.2). The corresponding matrix is arranged such that

$$\frac{\partial \delta u_i}{\partial x_j} C_{ijkl} \frac{\partial \Delta u_k}{\partial x_l} \quad = \quad \delta\underline{\boldsymbol{\varepsilon}}^T \underline{\mathbf{C}} \Delta\underline{\boldsymbol{\varepsilon}} \tag{2.7}$$

2.1.2 Introduction of Shell Hypotheses

Next, the continuum is specialized to the form of a *shell*, i.e., a body bounded by two closely-spaced, generally curved surfaces separated by a relatively small, variable *thickness* dimension h.

To obtain the Continuum-Based shell equations, we translate this qualitative description into the following simplifying hypotheses (i.e., mathematical constraints) which are subsequently introduced in the LV equations (2.1)

Kinematic Hypotheses

$$\text{Total:} \quad \begin{cases} \mathbf{x}(\xi,\eta,\tilde{z}) &= \overline{\mathbf{x}}(\xi,\eta) + \tilde{z}\,\widehat{\mathbf{x}}(\xi,\eta) \\ \mathbf{u}(\xi,\eta,\tilde{z}) &= \overline{\mathbf{u}}(\xi,\eta) + \tilde{z}\,\widehat{\mathbf{u}}(\xi,\eta) \end{cases} \tag{2.8}$$

$$\text{Incremental:} \quad \begin{cases} \Delta\mathbf{u} &= \Delta\overline{\mathbf{u}} + \tilde{z}\,\Delta\widehat{\mathbf{u}} \\ \Delta\widehat{\mathbf{u}}\cdot\widehat{\mathbf{x}} &= 0 \end{cases} \tag{2.9}$$

Static Hypotheses

$$\text{Total:} \quad \sigma_{33_\ell}(\xi,\eta,\tilde{z}) = 0 \tag{2.10}$$

$$\text{Incremental:} \quad \sigma^{\langle T\rangle}_{33_\ell}(\xi,\eta,\tilde{z}) = 0 \tag{2.11}$$

In the above expressions, ξ,η are surface (curvilinear) coordinates, and $\tilde{z}$ is the thickness coordinate measured from the reference-surface (see Fig. 1). For the simple case when

the reference-surface is chosen to coincide with the mid-surface, $\tilde{z}$ and h are related by $\tilde{z} = \varsigma h/2$, where ς is a parameter that ranges linearly from -1 at the bottom-surface to +1 at the top-surface.

Regarding the kinematic hypotheses, $\overline{x}_i$ are the coordinates of a point on the reference-surface, $\hat{x}_i$ is the unit *pseudo-normal vector* (i.e, the current orientation of a vector initially normal to the reference-surface), and $\overline{u}_i$ and $\widehat{u}_i$ are the displacements of the reference-surface and psuedo-normal (relative to the reference-surface), respectively (see Fig. 2). Thus, (2.8) expresses the hypothesis that normals remain *straight* via linearity through the thickness. The incremental constraint (2.9) says that the incremental pseudo-normal displacement vector $\Delta\hat{\mathbf{u}}$ has no component parallel to the pseudo-normal itself. The equation is therefore an expression of the *incremental rigidity* of the normal. Note that any two components of $\Delta\hat{\mathbf{u}}$ in a plane *orthogonal* to $\hat{\mathbf{x}}$ may be associated with rotation increments, say $\Delta\theta_{1p}$ and $\Delta\theta_{2p}$, as illustrated in Figure 2.

Regarding the static hypotheses, we have introduced a local Cartesian coordinate basis at each point $(\xi, \eta, \tilde{z})$, called the *lamina* basis (denoted by the subscript "ℓ"). The orthogonal lamina coordinates, $x_{i_\ell} = (x_{1_\ell}, x_{2_\ell}, x_{3_\ell})$ or (x_ℓ, y_ℓ, z_ℓ), are defined such that x_ℓ and y_ℓ lie in the plane tangent to the the surface $\tilde{z}$ = constant, with z_ℓ normal to that surface (Fig. 3). Thus, (2.10) enforces the *zero normal stress* constraint in the current configuration, while (2.11) enforces the same constraint on a rate (or, equivalently, incremental) basis. Note that for this latter purpose, an *objective* rate is chosen, namely $\sigma_{ij}^{\langle T \rangle}$ — the Truesdell rate [17].

Incorporating the above constraints in the continuum variational operators (2.2)–(2.3) leads to the following CB shell counterparts

$$\mathcal{F}^{\text{int}}(\mathbf{u}) = \int_V \delta\tilde{\boldsymbol{\varepsilon}}_\ell^T \tilde{\boldsymbol{\sigma}}_\ell \, dV \tag{2.12}$$

$$D\mathcal{F}^{\text{mat'l}}(\Delta\mathbf{u}) = \int_V \delta\tilde{\boldsymbol{\varepsilon}}_\ell^T \tilde{\underline{\mathbf{C}}}_\ell \, \Delta\tilde{\boldsymbol{\varepsilon}}_\ell \, dV \tag{2.13}$$

$$D\mathcal{F}^{\text{geom}}(\Delta\mathbf{u}) = \int_V \nabla_\ell(\delta u_i)^T \tilde{\sigma}_\ell \nabla_\ell(\Delta u_i) \, dV \tag{2.14}$$

where

$$\Delta\tilde{\boldsymbol{\varepsilon}}_\ell = \lfloor \Delta\varepsilon_{zz_\ell}, \ \Delta\varepsilon_{yy_\ell}, \ 2\Delta\varepsilon_{zy_\ell}, \ 2\Delta\varepsilon_{yz_\ell}, \ 2\Delta\varepsilon_{xz_\ell} \rfloor \tag{2.15}$$

and

$$\tilde{\boldsymbol{\sigma}}_\ell = \lfloor \sigma_{zz_\ell}, \ \sigma_{yy_\ell}, \ \sigma_{zy_\ell}, \ \sigma_{yz_\ell}, \ \sigma_{xz_\ell} \rfloor \tag{2.16}$$

and $\tilde{\underline{\mathbf{C}}}_\ell$ is obtained by static condensation of the '33' component in $\underline{\mathbf{C}}_\ell$ as a result of (2.11).

The lamina-subscripted variables are related to the corresponding global Cartesian quantities via the orthogonal transformation matrix, **L**, defined such that

$$\mathbf{v}_\ell = \mathbf{L}\mathbf{v} \tag{2.17}$$

and

$$\mathbf{t}_\ell = \mathbf{L}\mathbf{t}\mathbf{L}^T \tag{2.18}$$

where **v** is a generic vector and **t** is a generic second-rank tensor.

Finally, note that the total kinematic constraint (2.8), expressing the through-thickness linearity of the displacement field, is implicit in (2.12)–(2.14), while the *incremental* kinematic constraint (2.9) is more effectively imposed later, on a *discrete* (i.e., finite-element nodal) basis.

§2.3 Reduction to Resultant (CBR) Equations

The CB shell equations differ from conventional shell equations in that they involve full volume integration (for a discussion of other differences refer to [19]). In [19] (and similarly in [15]), the thickness coordinate, $\tilde{z}$, is pre-integrated out by making the "thinness" assumptions reviewed in Section 2.1. Without such assumptions, even though the displacements are linear in $\tilde{z}$ (see (2.8)), the strains do not in general inherit this nice feature. The culprit is the Jacobian matrix, $\mathbf{J}_\ell$, relating curvilinear $(\xi, \eta, \tilde{z})$ and Cartesian (x_ℓ, y_ℓ, z_ℓ) derivatives, i.e.,

$$\frac{\partial(\,\cdot\,)}{\partial \mathbf{x}_\ell} \;=\; \mathbf{J}_\ell^{-T} \frac{\partial(\,\cdot\,)}{\partial \boldsymbol{\xi}} \tag{2.19}$$

where

$$\mathbf{J}_\ell \;=\; \frac{\partial \mathbf{x}_\ell}{\partial \boldsymbol{\xi}} \;=\; \begin{bmatrix} \frac{\partial x_\ell}{\partial \xi} & \frac{\partial x_\ell}{\partial \eta} & \frac{\partial x_\ell}{\partial \zeta} \\ \frac{\partial y_\ell}{\partial \xi} & \frac{\partial y_\ell}{\partial \eta} & \frac{\partial y_\ell}{\partial \zeta} \\ \frac{\partial z_\ell}{\partial \xi} & \frac{\partial z_\ell}{\partial \eta} & \frac{\partial z_\ell}{\partial \zeta} \end{bmatrix} \tag{2.20}$$

In general, $\mathbf{J}_\ell$ is a full matrix with at least a linear dependence on $\tilde{z}$; hence, its inverse engenders a *rational* $\tilde{z}$ dependence in all laminar derivatives via (2.19). This distributes the thickness coordinate throughout the variational operators, (2.12)–(2.14), and inhibits any further simplification.

However, under the assumptions (A1)–(A3) given in Section 2.1, $\mathbf{J}_\ell$ is replaced by the $\tilde{z}$-independent matrix

$$\overline{\mathbf{J}}_\ell \;=\; \begin{bmatrix} \bar{\mathbf{j}}_\ell & \mathbf{0} \\ \mathbf{0} & h/2 \end{bmatrix} \tag{2.21}$$

where the bar indicates evaluation at the reference surface. Employing (2.20), the variational equations are reduced to CBR form, wherein the pertinent operators are redefined

as

$$\mathcal{F}^{int}(\mathbf{u}) = \int_S \delta\mathbf{e}^T \mathbf{s}\, dS \tag{2.22}$$

$$D\mathcal{F}^{mat'l}(\Delta\mathbf{u}) = \int_S \delta\mathbf{e}^T \mathbf{D}\, \Delta\mathbf{e}\, dS \tag{2.23}$$

$$D\mathcal{F}^{geom}(\Delta\mathbf{u}) = \int_S \delta\mathbf{g}_i^T \mathbf{S}\, \Delta\mathbf{g}_i\, dS \tag{2.24}$$

Notice that the volume integrals have been replaced by surface integrals and the previous stress and strain measures replaced by corresponding *resultant* quantities. These $\tilde{z}$-independent quantities are partitioned into membrane, bending and transverse-shear components as follows

$$\Delta\mathbf{e} = \begin{Bmatrix} \Delta\mathbf{e}^m \\ \Delta\mathbf{e}^b \\ \Delta\mathbf{e}^s \end{Bmatrix} \quad \text{or} \quad \begin{Bmatrix} \Delta\epsilon \\ \Delta\kappa \\ \Delta\gamma \end{Bmatrix} \tag{2.25}$$

$$\mathbf{s} = \begin{Bmatrix} \mathbf{s}^m \\ \mathbf{s}^b \\ \mathbf{s}^s \end{Bmatrix} \quad \text{or} \quad \begin{Bmatrix} \mathbf{n} \\ \underline{\mathbf{m}} \\ \mathbf{q} \end{Bmatrix} \tag{2.26}$$

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}^m & \mathbf{D}^{mb} & \mathbf{0} \\ & \mathbf{D}^b & \mathbf{0} \\ sym. & & \mathbf{D}^s \end{bmatrix} \tag{2.27}$$

$$\mathbf{S} = \begin{bmatrix} \mathbf{n} & \mathbf{m} & \mathbf{q} \\ & \mathbf{m}' & \mathbf{q}' \\ sym. & & \mathbf{0} \end{bmatrix} \tag{2.28}$$

The resultant quantities $\Delta\mathbf{e}$, $\mathbf{s}$ and $\mathbf{D}$, are related to the pointwise continuum quan-

tities via a $\tilde{z}$-dependent partitioning matrix, $\mathbf{Z}$, i.e.,

$$\Delta\tilde{\underline{\varepsilon}}_\ell = \mathbf{Z}(\tilde{z})\ \Delta\mathbf{e}(\xi,\eta) \tag{2.29}$$

$$\mathbf{s} = \int_{\tilde{z}} \mathbf{Z}^T(\tilde{z})\ \tilde{\underline{q}}_\ell\, d\tilde{z} \tag{2.30}$$

$$\mathbf{D} = \int_{\tilde{z}} \mathbf{Z}^T(\tilde{z})\ \tilde{\underline{\mathbf{C}}}_\ell\, \mathbf{Z}(\tilde{z})\, d\tilde{z} \tag{2.31}$$

where

$$\mathbf{Z} = \begin{bmatrix} \mathbf{I} & \tilde{z}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{2.32}$$

(Note: In [19], the 2×2 identity matrix in the lower right corner of $\mathbf{Z}$ is replaced with a diagonal matrix of shear correction factors.)

In particular, the individual stress resultants appearing in (2.26) and (2.28) are defined as follows

$$\underline{\mathbf{n}} = \int_{\tilde{z}} \lfloor \sigma_{zz_\ell},\ \sigma_{yy_\ell},\ \sigma_{zy_\ell} \rfloor^T\, d\tilde{z} \tag{2.33}$$

$$\underline{\mathbf{m}} = \int_{\tilde{z}} \lfloor \sigma_{zz_\ell},\ \sigma_{yy_\ell},\ \sigma_{zy_\ell} \rfloor^T\, \tilde{z}\, d\tilde{z} \tag{2.34}$$

$$\mathbf{q} = \int_{\tilde{z}} \lfloor \sigma_{zz_\ell},\ \sigma_{zy_\ell} \rfloor^T\, d\tilde{z} \tag{2.35}$$

$$\underline{\mathbf{m}}' = \int_{\tilde{z}} \lfloor \sigma_{zz_\ell},\ \sigma_{yy_\ell},\ \sigma_{zy_\ell} \rfloor^T\, \tilde{z}^2\, d\tilde{z} \tag{2.36}$$

$$\mathbf{q}' = \int_{\tilde{z}} \lfloor \sigma_{zz_\ell},\ \sigma_{zy_\ell} \rfloor^T\, \tilde{z}\, d\tilde{z} \tag{2.37}$$

and $\mathbf{n}$, $\mathbf{m}$ and $\mathbf{m}'$ are expanded matrix counterparts of $\underline{\mathbf{n}}$, $\underline{\mathbf{m}}$ and $\underline{\mathbf{m}}'$, respectively. For example

$$\mathbf{n} = \int_{\tilde{z}} \begin{bmatrix} \sigma_{zz_\ell} & \sigma_{zy_\ell} \\ \sigma_{zy_\ell} & \sigma_{yy_\ell} \end{bmatrix} d\tilde{z}$$

Finally, the individual resultant strain-displacement and gradient-displacement relations (the latter required for $D\mathcal{F}^{ecom}$) may be written

$$\Delta \mathbf{e}^m = \lfloor \frac{\partial \Delta \bar{u}_\ell}{\partial x_\ell}, \quad \frac{\partial \Delta \bar{v}_\ell}{\partial y_\ell}, \quad (\frac{\partial \Delta \bar{u}_\ell}{\partial y_\ell} + \frac{\partial \Delta \bar{v}_\ell}{\partial x_\ell}) \rfloor^T \tag{2.38}$$

$$\Delta \mathbf{e}^b = \lfloor \frac{\partial \Delta \hat{u}_\ell}{\partial x_\ell}, \quad \frac{\partial \Delta \hat{v}_\ell}{\partial y_\ell}, \quad (\frac{\partial \Delta \hat{u}_\ell}{\partial y_\ell} + \frac{\partial \Delta \hat{v}_\ell}{\partial x_\ell}) \rfloor^T \tag{2.39}$$

$$\Delta \mathbf{e}^s = \lfloor (\frac{\partial \Delta \bar{w}_\ell}{\partial y_\ell} + \Delta \hat{v}_\ell), \quad (\frac{\partial \Delta \bar{w}_\ell}{\partial x_\ell} + \Delta \hat{u}_\ell) \rfloor^T \tag{2.40}$$

and

$$\Delta \mathbf{g}_i = \lfloor \nabla_\ell (\Delta \bar{u}_{i_\ell}), \quad \nabla_\ell (\Delta \hat{u}_{i_\ell}), \quad \Delta \hat{u}_{i_\ell} \rfloor^T \tag{2.41}$$

where the laminar derivatives are computed via the reference surface Jacobian, $\bar{\mathbf{j}}_\ell$. It should be noted that through these expressions, the kinematic constraint (2.8) now appears explicitly in the LV equations.

§2.4 Finite-Element Discretization

Finite-element matrix equations of motion are easily obtained from either the CB or CBR shell equations by employing an *isoparametric* [4,20] form of local approximation. Thus, both the displacements and coordinates within an element subdomain are interpolated from corresponding nodal quantities as follows

$$\bar{\mathbf{x}}(\xi,\eta) = \sum_{a=1}^{N_{en}} N_a(\xi,\eta)\,\bar{\mathbf{x}}_a \tag{2.42}$$

$$\hat{\mathbf{x}}(\xi,\eta) = \sum_{a=1}^{N_{en}} N_a(\xi,\eta)\,\hat{\mathbf{x}}_a \tag{2.43}$$

$$\Delta\bar{\mathbf{u}}(\xi,\eta) = \sum_{a=1}^{N_{en}} N_a(\xi,\eta)\,\Delta\bar{\mathbf{u}}_a \tag{2.44}$$

$$\Delta\hat{\mathbf{u}}(\xi,\eta) = \sum_{a=1}^{N_{en}} N_a(\xi,\eta)\,\Delta\hat{\mathbf{u}}_a \tag{2.45}$$

where the subscript "a" ranges over the number of element nodes, and for quadrilateral elements, the N_a are typically chosen as Lagrangian interpolation functions (see Fig. 4). Note that the interpolation is performed in a *fixed* Cartesian basis, usually lamina or global, depending on the context.

Substitution of (2.37)–(2.40) into the LV equations (2.1) yields the corresponding finite-element matrix equations

$$\boxed{\mathbf{M}\Delta\ddot{\mathbf{d}} + (\mathbf{K}^{matl}+\mathbf{K}^{geom})\Delta\mathbf{d} = \mathbf{F}^{ext} - \mathbf{F}^{int} - \mathbf{M}\ddot{\mathbf{d}}} \tag{2.46}$$

where $\mathbf{M}$ is the assembled mass matrix (from $\mathcal{M}$), $\mathbf{F}^{ext}$ is the assembled external force vector (from $\mathcal{F}^{ext}$), $\mathbf{K}^{matl}$ and $\mathbf{K}^{geom}$ are the assembled material and geometric stiffness matrices, respectively (from $D\mathcal{F}^{matl}$ and $D\mathcal{F}^{geom}$) and $\mathbf{F}^{int}$ is the assembled internal force vector (from $\mathcal{F}^{int}$). The incremental displacement vector, $\Delta\mathbf{d}$, contains all of the active nodal degrees-of-freedom. As explained in Section 4, these degrees-of-freedom are usually expressed in a shell-oriented (rather than global-Cartesian) coordinate system.

The above arrays are assembled from the individual element arrays [20], with the linkage established by the *element displacement vectors*, $\Delta\mathbf{d}^e$, that is

$$\Delta\mathbf{d}^e \;=\; \{\,\Delta\mathbf{d}^e_a\,\} \;=\; \{\,\Delta\mathbf{d}_A\,\}$$

where "a" is an *element* node number and "A" is the corresponding *global* node number. At each shell node, we have potentially 6 degrees-of-freedom, defined by

$$\Delta\mathbf{d}^e_a \;=\; \begin{Bmatrix} \Delta\bar{\mathbf{u}}_a \\ \Delta\hat{\mathbf{u}}_a \end{Bmatrix} \tag{2.47}$$

However, during assembly the element arrays are suitably transformed to *discretely* enforce the shell incremental kinematic constraint (2.9) as mentioned above. This reduces the number of degrees-of-freedom per node to 5 — except where junctures occur [19].

In closing this section, we give the specific definitions of the element arrays $\mathbf{F}^{int^e}$, $\mathbf{K}^{matl^e}$ and $\mathbf{K}^{geome}$ which result from substituting (2.37)–(2.40) into the CBR shell operators (2.22)–(2.24), as it will be useful to compare these to the revised arrays presented in Section 4. In terms of nodal contributions, we have

$$\mathbf{F}^{int^e}_a \;=\; \int_S \mathbf{B}^T_a\,\mathbf{s}\,dS \tag{2.48}$$

$$\mathbf{K}^{matl^e}_{ab} \;=\; \int_S \mathbf{B}^T_a\,\mathbf{D}\,\mathbf{B}_b\,dS \tag{2.49}$$

$$\mathbf{K}^{geome}_{ab} \;=\; \int_S \begin{bmatrix} g_{ab}\mathbf{I} & g_{ab}\mathbf{I} \\ g_{ab}\mathbf{I} & g_{ab}\mathbf{I} \end{bmatrix} dS \tag{2.50}$$

where $\mathbf{B}_a$ is a nodal block of the element incremental strain-displacement matrix, defined by the relationship

$$\Delta\mathbf{e}(\xi,\eta) \;=\; \sum_{a=1}^{N_{en}} \mathbf{B}_a(\xi,\eta)\Delta\mathbf{d}_a \tag{2.51}$$

In practice, the B matrix at a given (integration) point, ξ, η, is first defined in the lamina system at that point such that

$$\Delta \mathbf{e}(\xi,\eta) \quad = \quad \sum_{a=1}^{N_{en}} \mathbf{B}_{a_\ell}(\xi,\eta)\Delta \mathbf{d}_{a_\ell} \tag{2.52}$$

where the lamina submatrix, $\mathbf{B}_{a_\ell}$, is related to the *globally attached* version via

$$\mathbf{B}_a \quad = \quad \mathbf{B}_{a_\ell}\lceil \mathbf{L} \rfloor \tag{2.53}$$

The block-diagonal matrix, $\lceil \mathbf{L} \rfloor$, is the expanded lamina transformation for a nodal DOF vector (2.43), i.e.,

$$\lceil \mathbf{L} \rfloor \quad = \quad \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} \end{bmatrix} \tag{2.54}$$

Using the above definitions, the CBR version of $\mathbf{B}_{a_\ell}$ in terms of the element shape functions, N_a, is given as follows

$$\mathbf{B}_{a_\ell} \quad = \quad \left[\begin{array}{ccc|ccc} N_{a,x_\ell} & 0 & 0 & 0 & 0 & 0 \\ 0 & N_{a,y_\ell} & 0 & 0 & 0 & 0 \\ N_{a,y_\ell} & N_{a,x_\ell} & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & N_{a,x_\ell} & 0 & 0 \\ 0 & 0 & 0 & 0 & N_{a,y_\ell} & 0 \\ 0 & 0 & 0 & N_{a,y_\ell} & N_{a,x_\ell} & 0 \\ \hline 0 & 0 & N_{a,x_\ell} & N_a & 0 & 0 \\ 0 & 0 & N_{a,y_\ell} & 0 & N_a & 0 \end{array}\right] \tag{2.55}$$

Finally, the scalar coefficients appearing in the CBR geometric stiffness matrix (2.46) are defined as

$$g_{\bar{a}\bar{b}} = \nabla_\ell(N_a)^t\,\mathbf{n}\nabla_\ell(N_b) \tag{2.56}$$

$$g_{\bar{a}\hat{b}} = \nabla_\ell(N_a)^t\,\mathbf{m}\nabla_\ell(N_b) + \nabla_\ell(N_a)^t\,\mathbf{q}N_b \tag{2.57}$$

$$g_{\hat{a}\bar{b}} = \nabla_\ell(N_a)^t\,\mathbf{m}\nabla_\ell(N_b) + N_a\mathbf{q}^t\nabla_\ell(N_b) \tag{2.58}$$

$$g_{\hat{a}\hat{b}} = \nabla_\ell(N_a)^t\,\mathbf{m}'\nabla_\ell(N_b) + N_a\mathbf{q}'^t\nabla_\ell(N_b) + \nabla_\ell(N_a)^t\,\mathbf{q}'N_b \tag{2.59}$$

3

Extension to Thick/Layered Shells

§3.1 Motivation

While the CBR shell assumptions employed in the previous section greatly reduce finite-element computational costs via the use of thickness pre-integration, at the same time they restrict the maximum thickness-to-radius regime relative to the original CB formulation. Thus, for thick shells with substantial curvature (or curvature change), there is the possibility for a serious degradation of accuracy.

Furthermore, for problems in which transverse-shear effects become significant, e.g., in moderately thick shells with shear-flexible layers, it may be necessary to resort to a more refined description of through-thickness kinematics than provided by the *straight-normal* approximation employed in both CB and CBR formulations.

Hence, our goal in this section is as follows. We would like to relax the maximum-thickness limitations engendered in our previous work, while at the same time retaining the cost-effectiveness of the thickness pre-integration treatment of multi-layered elastic/plastic shells.

§3.2 Revised Kinematics 1: Curvature Effects

The thickness-to-radius limitation imposed by the CBR method may be directly attributed to the Curvature Assumption (A2), wherein the surface metric, or Jacobian, is held constant (at its reference surface value) through the thickness. Recall from §2.3 that the reason for doing this is that it leads to a linear through-thickness variation of the incremental strains, and hence to a simple separation of direct and moment stress-resultants. Once such resultants were computed, the variational (equilibrium) equations were devoid of the thickness coordinate, $\bar{z}$, and the formation of finite-element equilibrium arrays was reduced in scope from volume to surface integration.

Upon closer look at the Jacobian, and how it influences the strain-displacement relations, it becomes apparent that if we agree to retain the other two CBR assumptions, i.e., Normality (A1) and Taper (A3), then the Curvature Assumption may be completely ***abandoned*** without unduly complicating the equations. The derivation follows.

3.2.1 The Surface Jacobian

First, as observed in [19], after invoking CBR assumptions (A1) and (A3), the Jacobian matrix decouples into laminar (in-plane) and transverse partitions

$$\mathbf{J}_\ell \;=\; \frac{\partial \mathbf{x}_\ell}{\partial \xi} \;=\; \begin{bmatrix} \mathbf{j}_\ell & \mathbf{0} \\ \mathbf{0} & h/2 \end{bmatrix} \tag{3.1}$$

where $\mathbf{j}_\ell$, the surface Jacobian, has the linear through-thickness variation

$$\mathbf{j}_\ell \;=\; \bar{\mathbf{j}}_\ell + \tilde{z}\,\hat{\mathbf{j}}_\ell \tag{3.2}$$

with

$$\bar{\mathbf{j}}_\ell \;=\; \begin{bmatrix} \frac{\partial \bar{x}_\ell}{\partial \xi} & \frac{\partial \bar{x}_\ell}{\partial \eta} \\ \frac{\partial \bar{y}_\ell}{\partial \xi} & \frac{\partial \bar{y}_\ell}{\partial \eta} \end{bmatrix} \tag{3.3}$$

and

$$\hat{\mathbf{j}}_\ell \;=\; \begin{bmatrix} \frac{\partial \hat{x}_\ell}{\partial \xi} & \frac{\partial \hat{x}_\ell}{\partial \eta} \\ \frac{\partial \hat{y}_\ell}{\partial \xi} & \frac{\partial \hat{y}_\ell}{\partial \eta} \end{bmatrix} \tag{3.4}$$

The inverse of $\mathbf{j}_\ell$ (used to obtain laminar derivatives from surface-coordinate derivatives) is thus the following matrix of rational polynomials in $\tilde{z}$

$$\boxed{\mathbf{j}_\ell^{-1} \;=\; \left[\bar{\mathbf{j}}_\ell^{cof^T} + \tilde{z}\,\hat{\mathbf{j}}_\ell^{cof^T}\right] / j(\tilde{z})} \tag{3.5}$$

where

$$\bar{\mathbf{j}}_\ell^{cof^T} \;=\; \begin{bmatrix} \frac{\partial \bar{y}_\ell}{\partial \eta} & -\frac{\partial \bar{x}_\ell}{\partial \eta} \\ -\frac{\partial \bar{y}_\ell}{\partial \xi} & \frac{\partial \bar{x}_\ell}{\partial \xi} \end{bmatrix} \tag{3.6}$$

and

$$\hat{\mathbf{j}}_\ell^{cof^T} \;=\; \begin{bmatrix} \frac{\partial \hat{y}_\ell}{\partial \eta} & -\frac{\partial \hat{x}_\ell}{\partial \eta} \\ -\frac{\partial \hat{y}_\ell}{\partial \xi} & \frac{\partial \hat{x}_\ell}{\partial \xi} \end{bmatrix} \tag{3.7}$$

are the transposed cofactor matrices of $\bar{\mathbf{j}}_\ell$ and $\hat{\mathbf{j}}_\ell$, respectively, and the surface Jacobian determinant, $j(\tilde{z})$, appearing in the denominator is the following quadratic polynomial

$$j(\tilde{z}) \;=\; \det(\mathbf{j}_\ell) \;=\; j_0 + j_1\tilde{z} + j_2\tilde{z}^2 \tag{3.8}$$

where

$$j_0 = \left(\frac{\partial \bar{x}_\ell}{\partial \xi}\frac{\partial \bar{y}_\ell}{\partial \eta} - \frac{\partial \bar{x}_\ell}{\partial \eta}\frac{\partial \bar{y}_\ell}{\partial \xi}\right) \tag{3.9}$$

$$j_1 = \left(\frac{\partial \widehat{x}_\ell}{\partial \xi}\frac{\partial \bar{y}_\ell}{\partial \eta} - \frac{\partial \widehat{x}_\ell}{\partial \eta}\frac{\partial \bar{y}_\ell}{\partial \xi}\right) + \left(\frac{\partial \bar{x}_\ell}{\partial \xi}\frac{\partial \widehat{y}_\ell}{\partial \eta} - \frac{\partial \bar{x}_\ell}{\partial \eta}\frac{\partial \widehat{y}_\ell}{\partial \xi}\right) \tag{3.10}$$

$$j_2 = \left(\frac{\partial \widehat{x}_\ell}{\partial \xi}\frac{\partial \widehat{y}_\ell}{\partial \eta} - \frac{\partial \widehat{x}_\ell}{\partial \eta}\frac{\partial \widehat{y}_\ell}{\partial \xi}\right) \tag{3.11}$$

REMARK 3.1

Note that $\bar{\mathbf{j}}_\ell^{cof^T}$, $\widehat{\mathbf{j}}_\ell^{cof^T}$, $j_0, j_1,$, and j_2 involve only the reference-surface quantities, $\bar{\mathbf{x}}$ and $\hat{\mathbf{x}}$ and hence are independent of $\tilde{z}$.

REMARK 3.2

When the CBR Curvature Assumption (A2) is introduced as well, equation (3.1) is replaced by the simpler approximation (2.21) so that

$$\mathbf{j}_\ell^{-1}(\tilde{z}) \leftarrow \bar{\mathbf{j}}_\ell^{-1} = \bar{\mathbf{j}}_\ell^{cof^T}/j_0 \tag{3.12}$$

which is just the first term in a Taylor expansion of (3.5) about $\tilde{z} = 0$, and leads to a *first approximation* shell theory.

Comparing this with (3.5), we see that the price of discarding the Curvature Assumption is an additional $\tilde{z}$-scaled matrix in the Jacobian numerator and a quadratic scalar function of $\tilde{z}$ in the denominator.

3.2.2 Laminar Displacement Derivatives

We now examine the effect of employing the Jacobian matrix (3.5), instead of the more restrictive CBR version, for the computation of laminar derivatives, as required in the CB shell internal force and stiffness operators (2.12)–(2.14).

The in-plane laminar derivatives of displacement are computed from the corresponding surface-coordinate derivatives via

$$\frac{\partial \Delta u_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}} = \mathbf{j}_\ell^{-T} \frac{\partial \Delta u_{i_\ell}}{\partial \boldsymbol{\xi}^{(P)}} \tag{3.13}$$

where the (P) superscript restricts our attention to *planar* components, i.e.,

$$\frac{\partial(\,\cdot\,)}{\partial \mathbf{x}_\ell^{(P)}} = \left\{ \begin{array}{c} \frac{\partial(\,\cdot\,)}{\partial x_\ell} \\ \frac{\partial(\,\cdot\,)}{\partial y_\ell} \end{array} \right\} \quad \text{and} \quad \frac{\partial(\,\cdot\,)}{\partial \boldsymbol{\xi}^{(P)}} = \left\{ \begin{array}{c} \frac{\partial(\,\cdot\,)}{\partial \xi} \\ \frac{\partial(\,\cdot\,)}{\partial \eta} \end{array} \right\} \tag{3.14}$$

while the subscript 'i' in (3.13) may range from 1 – 3.

Now, since

$$\frac{\partial \Delta u_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}} = \frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}} + \tilde{z} \frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}} \tag{3.15}$$

we may apply (3.13) independently to translational and rotational components. Employing the new definition (3.5) leads to

$$\frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}} = \frac{1}{j(\tilde{z})} \left[\bar{\mathbf{j}}_\ell^{cof} \frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \boldsymbol{\xi}} + \tilde{z}\, \hat{\mathbf{j}}_\ell^{cof} \frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \boldsymbol{\xi}} \right] \tag{3.16}$$

and

$$\frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}} = \frac{1}{j(\tilde{z})} \left[\bar{\mathbf{j}}_\ell^{cof} \frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \boldsymbol{\xi}} + \tilde{z}\, \hat{\mathbf{j}}_\ell^{cof} \frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \boldsymbol{\xi}} \right] \tag{3.17}$$

Finally, substituting (3.16) and (3.17) back into (3.15) yields the following useful expression

for computing laminar displacement derivatives

$$\frac{\partial \Delta u_{i_\ell}}{\partial x_\ell^{(P)}} = \frac{1}{\tilde{j}(\tilde{z})}\left[\overline{\frac{\partial \Delta \bar{u}_{i_\ell}}{\partial x_\ell^{(P)}}} + \tilde{z}\left[\overline{\frac{\partial \Delta \hat{u}_{i_\ell}}{\partial x_\ell^{(P)}}} + \frac{\partial \widehat{\Delta \bar{u}}_{i_\ell}}{\partial x_\ell^{(P)}}\right] + \tilde{z}^2 \frac{\partial \widehat{\Delta \hat{u}}_{i_\ell}}{\partial x_\ell^{(P)}}\right] \tag{3.18}$$

where the wide bars and hats above the displacement derivatives indicate association with $\bar{\mathbf{j}}_\ell$ and $\hat{\mathbf{j}}_\ell$, respectively, i.e.

$$\overline{\frac{\partial \Delta \bar{u}_{i_\ell}}{\partial x_\ell^{(P)}}} \stackrel{\text{def}}{=} \bar{\mathbf{j}}_\ell^{-T} \frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \xi^{(P)}} \tag{3.19}$$

$$\overline{\frac{\partial \Delta \hat{u}_{i_\ell}}{\partial x_\ell^{(P)}}} \stackrel{\text{def}}{=} \bar{\mathbf{j}}_\ell^{-T} \frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \xi^{(P)}} \tag{3.20}$$

$$\frac{\partial \widehat{\Delta \bar{u}}_{i_\ell}}{\partial x_\ell^{(P)}} \stackrel{\text{def}}{=} \left(\frac{j_2}{j_0}\right) \tilde{\mathbf{j}}_\ell^{-T} \frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \xi^{(P)}} \tag{3.21}$$

$$\frac{\partial \widehat{\Delta \hat{u}}_{i_\ell}}{\partial x_\ell^{(P)}} \stackrel{\text{def}}{=} \left(\frac{j_2}{j_0}\right) \tilde{\mathbf{j}}_\ell^{-T} \frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \xi^{(P)}} \tag{3.22}$$

and

$$\tilde{j}(\tilde{z}) = j(\tilde{z})/j_0 = 1 + \left(\frac{j_1}{j_0}\right)\tilde{z} + \left(\frac{j_2}{j_0}\right)\tilde{z}^2 \tag{3.23}$$

REMARK 3.3

The CBR (curvature-restricted) case is recovered by neglecting $\hat{\mathbf{j}}_\ell$ and replacing $\tilde{j}$ by 1. The result is

$$\frac{\partial \Delta u_{i_\ell}}{\partial x_\ell^{(P)}} = \left[\overline{\frac{\partial \Delta \bar{u}_{i_\ell}}{\partial x_\ell^{(P)}}} + \tilde{z}\overline{\frac{\partial \Delta \hat{u}_{i_\ell}}{\partial x_\ell^{(P)}}}\right] \tag{3.24}$$

which is the basis for the definitions given in §2.3.

3.2.3 Strain-Displacement Relations

Equation (3.18) may be used to construct similarly partitioned incremental strain-displacement relations.

First define the vector of in-plane incremental strain components as

$$\boxed{\Delta\underline{\boldsymbol{\varepsilon}}^{(P)} = \begin{Bmatrix} \Delta\varepsilon_{xx_\ell} \\ \Delta\varepsilon_{yy_\ell} \\ 2\Delta\varepsilon_{xy_\ell} \end{Bmatrix} = \frac{1}{\tilde{j}(\tilde{z})}\left[\Delta\underline{\boldsymbol{\varepsilon}}_0^{(P)} + \tilde{z}\Delta\underline{\boldsymbol{\varepsilon}}_1^{(P)} + \tilde{z}^2\Delta\underline{\boldsymbol{\varepsilon}}_2^{(P)}\right]} \tag{3.25}$$

where

$$\Delta\underline{\boldsymbol{\varepsilon}}_0^{(P)} = \begin{Bmatrix} \overline{\frac{\partial\Delta\bar{u}_\ell}{\partial x_\ell}} \\ \overline{\frac{\partial\Delta\bar{v}_\ell}{\partial y_\ell}} \\ \overline{\frac{\partial\Delta\bar{v}_\ell}{\partial x_\ell}} + \overline{\frac{\partial\Delta\bar{u}_\ell}{\partial y_\ell}} \end{Bmatrix} \tag{3.26}$$

$$\Delta\underline{\boldsymbol{\varepsilon}}_1^{(P)} = \begin{Bmatrix} \overline{\frac{\partial\Delta\hat{u}_\ell}{\partial x_\ell}} + \widehat{\frac{\partial\Delta\bar{u}_\ell}{\partial x_\ell}} \\ \overline{\frac{\partial\Delta\hat{v}_\ell}{\partial y_\ell}} + \widehat{\frac{\partial\Delta\bar{v}_\ell}{\partial y_\ell}} \\ \left(\overline{\frac{\partial\Delta\hat{v}_\ell}{\partial x_\ell}} + \overline{\frac{\partial\Delta\hat{u}_\ell}{\partial y_\ell}}\right) + \left(\widehat{\frac{\partial\Delta\bar{v}_\ell}{\partial x_\ell}} + \widehat{\frac{\partial\Delta\bar{u}_\ell}{\partial y_\ell}}\right) \end{Bmatrix} \tag{3.27}$$

$$\Delta\underline{\boldsymbol{\varepsilon}}_2^{(P)} = \begin{Bmatrix} \widehat{\frac{\partial\Delta\hat{u}_\ell}{\partial x_\ell}} \\ \widehat{\frac{\partial\Delta\hat{v}_\ell}{\partial y_\ell}} \\ \widehat{\frac{\partial\Delta\hat{v}_\ell}{\partial x_\ell}} + \widehat{\frac{\partial\Delta\hat{u}_\ell}{\partial y_\ell}} \end{Bmatrix} \tag{3.28}$$

Similarly, define a vector of transverse shear strain increments having the same structure as the in-plane components except for an additional contribution from the normal derivatives — $\partial(\,\cdot\,)/\partial z_\ell$

$$\begin{aligned} \Delta\underline{\boldsymbol{\varepsilon}}^{(S)} &= \begin{Bmatrix} 2\Delta\varepsilon_{xz_\ell} \\ 2\Delta\varepsilon_{yz_\ell} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial\Delta w_\ell}{\partial x_\ell} + \frac{\partial\Delta u_\ell}{\partial z_\ell} \\ \frac{\partial\Delta w_\ell}{\partial y_\ell} + \frac{\partial\Delta v_\ell}{\partial z_\ell} \end{Bmatrix} \\ &= \begin{Bmatrix} \overline{\frac{\partial\Delta w_\ell}{\partial x_\ell}} + \tilde{z}\widehat{\frac{\partial\Delta w_\ell}{\partial x_\ell}} \\ \overline{\frac{\partial\Delta w_\ell}{\partial y_\ell}} + \tilde{z}\widehat{\frac{\partial\Delta w_\ell}{\partial y_\ell}} \end{Bmatrix} + \begin{Bmatrix} \Delta\hat{u}_\ell \\ \Delta\hat{v}_\ell \end{Bmatrix} \end{aligned} \tag{3.29}$$

Rewriting (3.29) in the same form as (3.25), we obtain

$$\boxed{\Delta\underline{\varepsilon}^{(S)} = \frac{1}{\tilde{j}(\tilde{z})}\left[\Delta\underline{\varepsilon}_0^{(S)} + \tilde{z}\Delta\underline{\varepsilon}_1^{(S)} + \tilde{z}^2\Delta\underline{\varepsilon}_2^{(S)} + \tilde{j}\Delta\underline{\varepsilon}_3^{(S)}\right]} \tag{3.30}$$

where

$$\Delta\underline{\varepsilon}_0^{(S)} = \left\{\begin{array}{c}\overline{\frac{\partial\Delta\overline{w}_\ell}{\partial x_\ell}} \\ \overline{\frac{\partial\Delta\overline{w}_\ell}{\partial y_\ell}}\end{array}\right\} \tag{3.31}$$

$$\Delta\underline{\varepsilon}_1^{(S)} = \left\{\begin{array}{c}\overline{\frac{\partial\Delta\widehat{w}_\ell}{\partial x_\ell}} + \widehat{\frac{\partial\Delta\overline{w}_\ell}{\partial x_\ell}} \\ \overline{\frac{\partial\Delta\widehat{w}_\ell}{\partial y_\ell}} + \widehat{\frac{\partial\Delta\overline{w}_\ell}{\partial y_\ell}}\end{array}\right\} \tag{3.32}$$

$$\Delta\underline{\varepsilon}_2^{(S)} = \left\{\begin{array}{c}\widehat{\frac{\partial\Delta\widehat{w}_\ell}{\partial x_\ell}} \\ \widehat{\frac{\partial\Delta\widehat{w}_\ell}{\partial y_\ell}}\end{array}\right\} \tag{3.33}$$

$$\Delta\underline{\varepsilon}_3^{(S)} = \left\{\begin{array}{c}\Delta\widehat{u}_\ell \\ \Delta\widehat{v}_\ell\end{array}\right\} \tag{3.34}$$

3.2.4 Resultant-Oriented Strain Measures

To complete the kinematic development and prepare for insertion in the variational equations, we separate out the $\tilde{z}$-dependence in (3.25) and (3.30) via a partitioning matrix and deal with an expanded set of $\tilde{z}$-independent strain measures. This is analogous to the CBR membrane, bending and shear partitions presented in §2.3.

Thus, for the planar (membrane/bending) components, we define

$$\boxed{\Delta\underline{\varepsilon}^{(P)} = \mathbb{Z}^{(P)}\Delta\mathbf{e}^{(P)}} \tag{3.35}$$

where

$$\Delta\mathbf{e}^{(P)} = \left\{\begin{array}{c}\Delta\underline{\varepsilon}_0^{(P)} \\ \Delta\underline{\varepsilon}_1^{(P)} \\ \Delta\underline{\varepsilon}_2^{(P)}\end{array}\right\} \tag{3.36}$$

and

$$\mathbf{Z}^{(P)} = \frac{1}{\tilde{j}(\tilde{z})}[\mathbf{I} \mid \tilde{z}\mathbf{I} \mid \tilde{z}^2\mathbf{I}] \tag{3.37}$$

Similarly, for the transverse (shear) components, we define

$$\boxed{\Delta\underline{\boldsymbol{\varepsilon}}^{(S)} = \mathbf{Z}^{(S)}\Delta\mathbf{e}^{(S)}} \tag{3.38}$$

where

$$\Delta\mathbf{e}^{(S)} = \begin{Bmatrix} \Delta\underline{\boldsymbol{\varepsilon}}_0^{(S)} \\ \Delta\underline{\boldsymbol{\varepsilon}}_1^{(S)} \\ \Delta\underline{\boldsymbol{\varepsilon}}_2^{(S)} \\ \Delta\underline{\boldsymbol{\varepsilon}}_3^{(S)} \end{Bmatrix} \tag{3.39}$$

and

$$\mathbf{Z}^{(S)} = \frac{1}{\tilde{j}(\tilde{z})}[\mathbf{I} \mid \tilde{z}\mathbf{I} \mid \tilde{z}^2\mathbf{I} \mid \tilde{j}(\tilde{z})\mathbf{I}] \tag{3.40}$$

We intend to employ (3.35) and (3.38) to pre-integrate the LV equations through the thickness. However, this is postponed until some additional kinematic effects have been incorporated.

§3.3 Revised Kinematics 2: Transverse-Shear Effects

In this section, we introduce a simple correction to the transverse-shear strain-displacement relations used in the preceding sections. The correction allows for a predefined, e.g., parabolic, distribution of $\Delta\varepsilon_{xz_\ell}$ and $\Delta\varepsilon_{yz_\ell}$ through the thickness as compared with the weighted constant value used in the conventional Reissner-type theory. Hence, more accurate calculation of pointwise transverse-shear stresses is expected, which can be especially important in the case of thick inelastically deforming shells. Furthermore, such an approach may eliminate the need for transverse shear correction factors (in some cases), and hence relieve the user input burden.

Before proceeding, we note that this type of approach is not new. It has been used by others (e.g., [3]) with good experimental correlation, although mainly in the context of thick homogeneous elastic shells. While the advantages of the profile correction are not as obvious for inelastic/layered shells, the justification is that it is easy to evaluate and basically free of charge.

In light of the above explanation, the earlier definitions of the transverse shear strain increments are replaced by

$$\boxed{\left\{\begin{array}{c}\Delta\varepsilon_{xz_\ell}\\ \Delta\varepsilon_{yz_\ell}\end{array}\right\} \leftarrow (1+p(\tilde{z}))\left\{\begin{array}{c}\Delta\varepsilon_{xz_\ell}\\ \Delta\varepsilon_{yz_\ell}\end{array}\right\}} \tag{3.41}$$

where, as suggested in [3], we tentatively employ the parabolic profile

$$p(\tilde{z}) \quad = \quad -\frac{4(\tilde{z}-\tilde{z}^{mid})^2}{h^2} \tag{3.42}$$

in which $\tilde{z}^{mid}$ is the distance from the shell reference surface to the mid-surface, i.e., the *eccentricity.*

The correction given by (3.42) is thus consistent with the important case of shear-stress-free boundary conditions on top and bottom shell surfaces.

Introducing (3.41) in the partitioned strain-displacement relations given in (3.38) sim-

ply changes the definition of $\mathbf{Z}^{(S)}$ (3.39) to

$$\mathbf{Z}^{(S)} = \frac{(1+p(\tilde{z}))}{\tilde{j}(\tilde{z})}\left[\mathbf{I} \mid \tilde{z}\mathbf{I} \mid \tilde{z}^2\mathbf{I} \mid \tilde{j}(\tilde{z})\mathbf{I}\right] \tag{3.43}$$

§3.4 Revised Variational Equations

By direct substitution of the revised kinematic relations given in (3.25)–(3.40) into the CB variational equations (2.1),(2.12)–(2.14) we arrive at what we shall call the CBR2 form of the equations (for "2nd approximation CBR"). The only operators affected are the internal force and stiffness (material and geometric). The revised definitions are presented here.

In the following definitions, all surface integrals apply to the *reference* surface (just like in the CBR versions). The $\tilde{z}$ dependence of dS is not being neglected, but rather absorbed into the integrand via the identity

$$dS(\tilde{z}) \quad = \quad \tilde{j}(\tilde{z})\, d\overline{S} \tag{3.44}$$

where $d\overline{S}$ is the differential surface area at the reference surface and $\tilde{j}$ is the normalized surface Jacobian defined by (3.8) and (3.23). In the sequel, the symbol dS is used instead of $d\overline{S}$ for notational simplicity, that is the bars are omitted.

3.4.1 CBR2 Internal Force Operator

$$\boxed{\mathcal{F}^{int} \quad = \quad \int_S \left(\delta \mathbf{e}^{(P)T} \mathbf{s}^{(P)} + \delta \mathbf{e}^{(S)T} \mathbf{s}^{(S)}\right) dS} \tag{3.45}$$

where

$$\mathbf{s}^{(P)} \quad = \quad \left\{ \begin{array}{c} \mathbf{s}_0^{(P)} \\ \mathbf{s}_1^{(P)} \\ \mathbf{s}_2^{(P)} \end{array} \right\} \quad = \quad \int_{\tilde{z}} \left\{ \begin{array}{c} \underline{\boldsymbol{\sigma}}^{(P)} \\ \underline{\boldsymbol{\sigma}}^{(P)} \tilde{z} \\ \underline{\boldsymbol{\sigma}}^{(P)} \tilde{z}^2 \end{array} \right\} d\tilde{z} \tag{3.46}$$

and

$$\mathbf{s}^{(S)} \quad = \quad \left\{ \begin{array}{c} \mathbf{s}_0^{(S)} \\ \mathbf{s}_1^{(S)} \\ \mathbf{s}_2^{(S)} \\ \mathbf{s}_3^{(S)} \end{array} \right\} \quad = \quad \int_{\tilde{z}} (1 + p(\tilde{z})) \left\{ \begin{array}{c} \underline{\boldsymbol{\sigma}}^{(S)} \\ \underline{\boldsymbol{\sigma}}^{(S)} \tilde{z} \\ \underline{\boldsymbol{\sigma}}^{(S)} \tilde{z}^2 \\ \underline{\boldsymbol{\sigma}}^{(S)} \tilde{j} \end{array} \right\} d\tilde{z} \tag{3.47}$$

The (P) and (S) superscripts on the stress vectors have the same meaning as for the

strain vectors, i.e.,

$$\underline{\boldsymbol{\sigma}}^{(P)} = \begin{Bmatrix} \sigma_{zz_\ell} \\ \sigma_{yy_\ell} \\ \sigma_{zy_\ell} \end{Bmatrix} \tag{3.48}$$

and

$$\underline{\boldsymbol{\sigma}}^{(S)} = \begin{Bmatrix} \sigma_{zz_\ell} \\ \sigma_{zy_\ell} \end{Bmatrix} \tag{3.49}$$

and the vectors, $\mathbf{s}^{(P)}$ and $\mathbf{s}^{(S)}$, constitute membrane/bending and shear stress-resultants conjugate to the strain measures $\delta\mathbf{e}^{(P)}$ and $\delta\mathbf{e}^{(S)}$, respectively.

Note that $\widetilde{j}(\tilde{z})$ from (3.44) has been factored into the stress-resultant definitions (3.46) and (3.47).

3.4.2 CBR2 Material Stiffness Operator

$$\boxed{D\mathcal{F}^{matl} = \int_S (\delta \mathbf{e}^{(P)T} \mathbf{D}^{(P)} \Delta \mathbf{e}^{(P)} + \delta \mathbf{e}^{(S)T} \mathbf{D}^{(S)} \Delta \mathbf{e}^{(S)})\, dS} \tag{3.50}$$

where the in-Plane and transverse-Shear constitutive-resultant matrices are defined by

$$\mathbf{D}^{(P)} = \int_{\tilde{z}} \frac{1}{\tilde{j}} \begin{bmatrix} \underline{\tilde{C}}^{(P)} & \underline{\tilde{C}}^{(P)}\tilde{z} & \underline{\tilde{C}}^{(P)}\tilde{z}^2 \\ & \underline{\tilde{C}}^{(P)}\tilde{z}^2 & \underline{\tilde{C}}^{(P)}\tilde{z}^3 \\ sym. & & \underline{\tilde{C}}^{(P)}\tilde{z}^4 \end{bmatrix} d\tilde{z} \tag{3.51}$$

$$\mathbf{D}^{(S)} = \int_{\tilde{z}} \frac{(1+p(\tilde{z}))}{\tilde{j}} \begin{bmatrix} \underline{\tilde{C}}^{(S)} & \underline{\tilde{C}}^{(S)}\tilde{z} & \underline{\tilde{C}}^{(S)}\tilde{z}^2 & \underline{\tilde{C}}^{(S)}\tilde{j} \\ & \underline{\tilde{C}}^{(S)}\tilde{z}^2 & \underline{\tilde{C}}^{(S)}\tilde{z}^3 & \underline{\tilde{C}}^{(S)}\tilde{z}\tilde{j} \\ sym. & & \underline{\tilde{C}}^{(S)}\tilde{z}^4 & \underline{\tilde{C}}^{(S)}\tilde{z}^2\tilde{j} \\ & & & \underline{\tilde{C}}^{(S)}\tilde{j}^2 \end{bmatrix} d\tilde{z} \tag{3.52}$$

The constitutive tensors, $\underline{\tilde{C}}^{(P)}$ and $\underline{\tilde{C}}^{(S)}$, are the partitions of $\underline{\tilde{C}}_\ell$ corresponding to the planar and transverse-shear stress/strain components defined in (3.25) and (3.29), respectively.

3.4.3 CBR2 Geometric Stiffness Operator

$$\boxed{D\mathcal{F}^{geom} = \sum_{i=1}^{3} \int_S \delta \mathbf{g}_i^T \mathbf{S}\, \Delta \mathbf{g}_i\, dS} \tag{3.53}$$

where

$$\mathbf{S} = \int_{\tilde{z}} \mathbf{Z}^{(G)T} \boldsymbol{\sigma}_\ell \mathbf{Z}^{(G)} \tilde{j}(\tilde{z})\, d\tilde{z} \tag{3.54}$$

and

$$\mathbf{Z}^{(G)} = \frac{1}{\tilde{j}(\tilde{z})} \begin{bmatrix} \mathbf{I} & \tilde{z}\mathbf{I} & \tilde{z}^2\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \tilde{j} \end{bmatrix} \tag{3.55}$$

Expansion of (3.54) yields the following 'initial-stress' resultant matrix

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} & \mathbf{S}_{13} & \mathbf{S}_{14} \\ & \mathbf{S}_{22} & \mathbf{S}_{23} & \mathbf{S}_{24} \\ sym. & & \mathbf{S}_{33} & \mathbf{S}_{34} \\ & & & \mathbf{S}_{44} \end{bmatrix} \tag{3.56}$$

where, for I and $J = 1, 2$ or 3:

$$\mathbf{S}_{IJ} = \int_{\tilde{z}} \frac{1}{\tilde{j}(\tilde{z})} \sigma^{(P)} \tilde{z}^{(I-1)\,(J-1)}\, d\tilde{z} \tag{3.57}$$

and

$$\mathbf{S}_{I4} = \int_{\tilde{z}} \underline{\sigma}^{(S)} \tilde{z}^{I-1}\, d\tilde{z} \tag{3.58}$$

and by the zero-normal-stress condition,

$$\mathbf{S}_{4,4} = \int_{\tilde{z}} \tilde{j}(\tilde{z})\, \sigma_{zz_\ell}\, d\tilde{z} \equiv 0 \tag{3.59}$$

The partitioned gradient vector, $\Delta \mathbf{g}_i$, in (3.53) is defined as follows

$$\Delta \mathbf{g}_i = \begin{Bmatrix} \nabla_0(\Delta u_{i_\ell}) \\ \nabla_1(\Delta u_{i_\ell}) \\ \nabla_2(\Delta u_{i_\ell}) \\ \Delta \hat{u}_{i_\ell} \end{Bmatrix} \tag{3.60}$$

where the subscripted gradient operators are analogous to the subscripted strain measures, (3.26)–(3.28), i.e.,

$$\nabla_0(\Delta u_{i_\ell}) = \overline{\frac{\partial \Delta \bar{u}_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}}} \tag{3.61}$$

$$\nabla_1(\Delta u_{i_\ell}) = \overline{\frac{\partial \Delta \hat{u}_{i_\ell}}{\partial \mathbf{x}_\ell^{(P)}}} + \frac{\partial \widehat{\Delta \bar{u}_{i_\ell}}}{\partial \mathbf{x}_\ell^{(P)}} \tag{3.62}$$

$$\nabla_2(\Delta u_{i_\ell}) = \frac{\partial \widehat{\Delta \hat{u}_{i_\ell}}}{\partial \mathbf{x}_\ell^{(P)}} \tag{3.63}$$

The CBR2 shell operators presented above are now in appropriate form for finite-element discretization.

4

Finite-Element Implementation

§4.1 Overview

In this section, we discuss the numerical implementation of the extended CBR, or CBR2, shell theory using the same finite-element framework as employed in our previous work (e.g., [4],[19]). The discussion includes modifications to the element strain interpolation matrices reflecting the kinematic revisions presented in Section 3, corresponding formulas for computing the element force and stiffness arrays, a summary of specific shell-element types (e.g., 4, 9 and 16-node) and an outline of the solution procedures used to obtain displacements and stresses. Section 2.4 is recommended as background reading.

§4.2 Revised Element Arrays

4.2.1 Strain-Displacement Matrices

Substituting the C^0 element displacement approximations, (2.37)–(2.40) into the CBR2 incremental strain-displacement relations, (3.36) and (3.39), leads to the following element counterparts.

$$\Delta\mathbf{e}^{(P)}(\xi,\eta) \;=\; \sum_{a=1}^{N_{en}} \mathbf{B}_a^{(P)}(\xi,\eta)\Delta\mathbf{d}_a \tag{4.1}$$

and

$$\Delta\mathbf{e}^{(S)}(\xi,\eta) \;=\; \sum_{a=1}^{N_{en}} \mathbf{B}_a^{(S)}(\xi,\eta)\Delta\mathbf{d}_a \tag{4.2}$$

where $\mathbf{B}_a^{(P)}$ and $\mathbf{B}_a^{(S)}$ are element membrane/bending and transverse-shear strain-displacement submatrices at element node "a", respectively; and $\Delta\mathbf{d}_a$ is the element incremental displacement, or DOF, vector at node "a", resolved into *global* components (see (2.43)).

As explained in Section 2.4, the B matrices are obtained first with respect to the local *lamina* basis and then transformed, that is

$$\mathbf{B}_a^{(P)} \;=\; \mathbf{B}_{a_\ell}^{(P)}\lceil\mathbf{L}\rfloor \tag{4.3}$$

and

$$\mathbf{B}_a^{(S)} = \mathbf{B}_{a_\ell}^{(S)} \lceil \mathbf{L} \rfloor \tag{4.4}$$

where $\lceil \mathbf{L} \rfloor$ was defined in (2.5) such that

$$\Delta \mathbf{d}_{a_\ell}^e = \lceil \mathbf{L} \rfloor \Delta \mathbf{d}_a^e \tag{4.5}$$

The definitions of $\mathbf{B}_{a_\ell}^{(P)}$ and $\mathbf{B}_{a_\ell}^{(S)}$ in terms of the element shape functions follow

$$\mathbf{B}_{a_\ell}^{(P)} = \left[\begin{array}{ccc|ccc} \overline{N_{a,x_\ell}} & 0 & 0 & 0 & 0 & 0 \\ 0 & \overline{N_{a,y_\ell}} & 0 & 0 & 0 & 0 \\ \overline{N_{a,y_\ell}} & \overline{N_{a,x_\ell}} & 0 & 0 & 0 & 0 \\ \widehat{N_{a,x_\ell}} & 0 & 0 & \overline{N_{a,x_\ell}} & 0 & 0 \\ 0 & \widehat{N_{a,y_\ell}} & 0 & 0 & \overline{N_{a,y_\ell}} & 0 \\ \widehat{N_{a,y_\ell}} & \widehat{N_{a,x_\ell}} & 0 & \overline{N_{a,y_\ell}} & \overline{N_{a,x_\ell}} & 0 \\ 0 & 0 & 0 & \widehat{N_{a,x_\ell}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \widehat{N_{a,y_\ell}} & 0 \\ 0 & 0 & 0 & \widehat{N_{a,y_\ell}} & \widehat{N_{a,x_\ell}} & 0 \end{array}\right] \tag{4.6}$$

$$\mathbf{B}_{a_\ell}^{(S)} = \left[\begin{array}{ccc|ccc} 0 & 0 & \overline{N_{a,x_\ell}} & 0 & 0 & 0 \\ 0 & 0 & \overline{N_{a,y_\ell}} & 0 & 0 & 0 \\ 0 & 0 & \widehat{N_{a,x_\ell}} & 0 & 0 & \overline{N_{a,x_\ell}} \\ 0 & 0 & \widehat{N_{a,y_\ell}} & 0 & 0 & \overline{N_{a,y_\ell}} \\ 0 & 0 & 0 & 0 & 0 & \overline{N_{a,x_\ell}} \\ 0 & 0 & 0 & 0 & 0 & \overline{N_{a,y_\ell}} \\ 0 & 0 & 0 & N_a & 0 & 0 \\ 0 & 0 & 0 & 0 & N_a & 0 \end{array}\right] \tag{4.7}$$

The wide bars and hats in the above definitions emanate from (3.19)–(3.22) where they were used to define lamina displacement derivatives in terms of the natural (ξ, η) derivatives. Thus

$$\left\{\begin{array}{c} \overline{N_{a,x_\ell}} \\ \overline{N_{a,y_\ell}} \end{array}\right\} = \bar{\mathbf{j}}_\ell^{-T} \left\{\begin{array}{c} N_{a,\xi} \\ N_{a,\eta} \end{array}\right\} \tag{4.8}$$

and

$$\begin{Bmatrix} \widehat{N_{a,x_\ell}} \\ \widehat{N_{a,y_\ell}} \end{Bmatrix} = \widetilde{\mathbf{j}}_\ell^{-T} \left(\frac{j_2}{j_0}\right) \begin{Bmatrix} N_{a,\xi} \\ N_{a,\eta} \end{Bmatrix} \tag{4.9}$$

where the Jacobian matrices and determinant coefficients ($\bar{\mathbf{j}}_\ell, \widehat{\mathbf{j}}_\ell, j_0$ and j_2) were defined in section 3.2.1. They are computed within an element by using the coordinate interpolation formulas (2.37)–(2.38) as shown in section 4.2.5.

4.2.2 CBR2 Element Internal-Force Vector

Substitution of (4.1) and (4.2) into the internal-force operator (3.45) over an individual element subdomain, leads to the corresponding element force vector. A nodal subvector is given by

$$\boxed{\mathbf{F}_a^{int^e} = \int_{S^e} (\mathbf{B}_a^{(P)T} \mathbf{s}^{(P)} + \mathbf{B}_a^{(S)T} \mathbf{s}^{(S)})\, dS^e} \tag{4.10}$$

where all quantities are defined, and it remains only to numerically perform the surface integration.

4.2.3 CBR2 Element Material-Stiffness Matrix

Substitution of (4.1) and (4.2) into (3.50) yields the element material-stiffness matrix. A nodal submatrix is defined as

$$\boxed{\mathbf{K}_{ab}^{matl^e} = \int_{S^e} \mathbf{B}_a^{(P)T} \mathbf{D}^{(P)} \mathbf{B}_b^{(P)} + \mathbf{B}_a^{(S)T} \mathbf{D}^{(S)} \mathbf{B}_b^{(S)}\, dS^e} \tag{4.11}$$

and typically employs the same numerical quadrature scheme as used for the internal-force vector (4.29).

4.2.4 CBR2 Element Geometric-Stiffness Matrix

For the geometric stiffness, we may exploit the simplicity of the C^0 interpolation and obtain (after some manipulation) a much more economical form than suggested by (3.53). The result is the following element nodal submatrix

$$\boxed{\mathbf{K}_{ab}^{geome} = \begin{bmatrix} g_{\bar{a}\bar{b}}\mathbf{I} & g_{\bar{a}\hat{b}}\mathbf{I} \\ g_{\hat{a}\bar{b}}\mathbf{I} & g_{\hat{a}\hat{b}}\mathbf{I} \end{bmatrix}} \tag{4.12}$$

where

$$\begin{aligned} g_{\bar{a}\bar{b}} &= \overline{\nabla N_a}^T(\mathbf{S}_{11}\overline{\nabla N_b} + \mathbf{S}_{12}\widehat{\nabla N_b}) \\ &\quad + \widehat{\nabla N_a}^T(\mathbf{S}_{21}\overline{\nabla N_b} + \mathbf{S}_{22}\widehat{\nabla N_b}) \end{aligned} \tag{4.13}$$

$$\begin{aligned} g_{\bar{a}\hat{b}} &= \overline{\nabla N_a}^T(\mathbf{S}_{12}\overline{\nabla N_b} + \mathbf{S}_{13}\widehat{\nabla N_b} + \mathbf{S}_{14}N_b) \\ &\quad + \widehat{\nabla N_a}^T(\mathbf{S}_{22}\overline{\nabla N_b} + \mathbf{S}_{23}\widehat{\nabla N_b} + \mathbf{S}_{24}N_b) \end{aligned} \tag{4.14}$$

$$\begin{aligned} g_{\hat{a}\bar{b}} &= (\overline{\nabla N_a}^T\mathbf{S}_{21} + \widehat{\nabla N_a}^T\mathbf{S}_{31} + N_a\mathbf{S}_{41})\overline{\nabla N_b} \\ &\quad + (\overline{\nabla N_a}^T\mathbf{S}_{22} + \widehat{\nabla N_a}^T\mathbf{S}_{32} + N_a\mathbf{S}_{42})\widehat{\nabla N_b} \end{aligned} \tag{4.15}$$

$$\begin{aligned} g_{\hat{a}\hat{b}} &= \overline{\nabla N_a}^T(\mathbf{S}_{22}\overline{\nabla N_b} + \mathbf{S}_{23}\widehat{\nabla N_b} + \mathbf{S}_{24}N_b) \\ &\quad + \widehat{\nabla N_a}^T(\mathbf{S}_{32}\overline{\nabla N_b} + \mathbf{S}_{33}\widehat{\nabla N_b} + \mathbf{S}_{34}N_b) \\ &\quad + N_a(\mathbf{S}_{41}\overline{\nabla N_b} + \mathbf{S}_{43}N_b) \end{aligned} \tag{4.16}$$

The shape-function gradient operators in the above definitions correspond to (4.10) and (4.11), i.e.,

$$\overline{\nabla N_a} = \left\{ \begin{array}{c} \overline{N_{a,x_\ell}} \\ \overline{N_{a,y_\ell}} \end{array} \right\} \tag{4.17}$$

and

$$\widehat{\nabla N_a} = \left\{ \begin{array}{c} \widehat{N_{a,x_\ell}} \\ \widehat{N_{a,y_\ell}} \end{array} \right\} \tag{4.18}$$

4.2.5 Computation of Lamina Transformations and Jacobians

The lamina transformation matrix, $\mathbf{L}$, and the Jacobian quantities, $\bar{j}_\ell, \hat{j}_\ell, j_0, j_1$ and j_2, appear (or are required) in most of the revised element arrays. The following is a convenient step-by-step procedure for computing these quantities at an element integration point. As a prerequisite, it is assumed that the *current* element nodal coordinates, $\bar{x}_a$, and the shape functions N_a, $N_{a,\xi}$, and $N_{a,\eta}$ are given.

Step 0 "Normal" Vectors ($\hat{\mathbf{x}}_a$). Before processing any of the integration points, element nodal psuedo-normal vectors, $\hat{\mathbf{x}}_a$, should be evaluated (for $\hat{j}_\ell$, j_1 and j_2). By the Normality Assumption (A1), these may be computed from the reference surface coordinates, i.e.,

$$\hat{x}_a = \bar{e}_{\xi a} \times \bar{e}_{\eta a} \tag{4.19}$$

where

$$\bar{e}_{\xi a} = \frac{\partial \bar{x}}{\partial \xi}(\xi_a) = \sum_{b=1}^{N_{en}} \frac{\partial N_b}{\partial \xi}(\xi_a)\, \bar{x}_b \tag{4.20}$$

$$\bar{e}_{\eta a} = \frac{\partial \bar{x}}{\partial \eta}(\xi_a) = \sum_{b=1}^{N_{en}} \frac{\partial N_b}{\partial \eta}(\xi_a)\, \bar{x}_b \tag{4.21}$$

and ξ_a refers to the (ξ, η) coordinates at node "a". This requires the evaluation of the shape-function natural derivatives at nodes as well as integration points. However, this involves no additional computation, since *natural* derivatives are the same for all elements of a given type and hence may be *pre-computed.*

The following steps are then performed at each element integration point:

Step 1 Tangent and Curvature Vectors ($\partial\bar{\mathbf{x}}/\partial\xi_a$ and $\partial\hat{\mathbf{x}}/\partial\xi_a$).

$$\bar{\mathbf{e}}_\xi(\xi) = \frac{\partial\bar{\mathbf{x}}}{\partial\xi}(\xi) = \sum_{a=1}^{N_{en}} \frac{\partial N_a}{\partial\xi}(\xi)\,\bar{\mathbf{x}}_a \tag{4.22}$$

$$\bar{\mathbf{e}}_\eta(\xi) = \frac{\partial\bar{\mathbf{x}}}{\partial\eta}(\xi) = \sum_{a=1}^{N_{en}} \frac{\partial N_a}{\partial\eta}(\xi)\,\bar{\mathbf{x}}_a \tag{4.23}$$

$$\mathbf{h}_\xi(\xi) = \frac{\partial\hat{\mathbf{x}}}{\partial\xi}(\xi) = \sum_{a=1}^{N_{en}} \frac{\partial N_a}{\partial\xi}(\xi)\,\hat{\mathbf{x}}_a \tag{4.24}$$

$$\mathbf{h}_\eta(\xi) = \frac{\partial\hat{\mathbf{x}}}{\partial\eta}(\xi) = \sum_{a=1}^{N_{en}} \frac{\partial N_a}{\partial\eta}(\xi)\,\hat{\mathbf{x}}_a \tag{4.25}$$

Step 2 Laminar Transformations ($\mathbf{L}$). The lamina-global transformation matrix (see (4.5)) is defined at a given interior point as

$$\mathbf{L} = \begin{bmatrix} \hat{\mathbf{e}}_{x_\ell}^T \\ \hat{\mathbf{e}}_{y_\ell}^T \\ \hat{\mathbf{e}}_{z_\ell}^T \end{bmatrix} \tag{4.26}$$

where $\hat{\mathbf{e}}_{x_\ell}$, $\hat{\mathbf{e}}_{y_\ell}$ and $\hat{\mathbf{e}}_{z_\ell}$ are orthogonal unit vectors parallel to the local x_ℓ, y_ℓ and z_ℓ axes, respectively. To obtain an *unbiased* lamina system — with respect to the natural (surface) coordinates, ξ, η — we construct the unit vectors, $\hat{\mathbf{e}}_{x_\ell}, \hat{\mathbf{e}}_{y_\ell}, \hat{\mathbf{e}}_{z_\ell}$ as follows:

$$(a) \quad \hat{\mathbf{e}}_{z_\ell} = \bar{\mathbf{e}}_\xi \times \bar{\mathbf{e}}_\eta / \| \cdot \| \tag{4.27}$$

$$(b) \quad \hat{\mathbf{e}}_A = (\bar{\mathbf{e}}_\xi + \bar{\mathbf{e}}_\eta)/\| \cdot \| \tag{4.28}$$

$$(c) \quad \hat{\mathbf{e}}_B = \hat{\mathbf{e}}_{z_\ell} \times \hat{\mathbf{e}}_A \tag{4.29}$$

$$(d) \quad \hat{\mathbf{e}}_{x_\ell} = \frac{\sqrt{2}}{2}(\hat{\mathbf{e}}_A - \hat{\mathbf{e}}_B) \tag{4.30}$$

$$(e) \quad \hat{\mathbf{e}}_{y_\ell} = \frac{\sqrt{2}}{2}(\hat{\mathbf{e}}_A + \hat{\mathbf{e}}_B) \tag{4.31}$$

Step 3 Surface Jacobian Matrices ($\bar{\mathbf{j}}_\ell$ and $\hat{\mathbf{j}}_\ell$) .

$$\bar{\mathbf{j}}_\ell = \left[\frac{\partial\bar{\mathbf{x}}_\ell^{(P)}}{\partial\xi} \;\middle|\; \frac{\partial\bar{\mathbf{x}}_\ell^{(P)}}{\partial\eta} \right] = \begin{bmatrix} (\hat{\mathbf{e}}_{x_\ell} \cdot \bar{\mathbf{e}}_\xi) & (\hat{\mathbf{e}}_{x_\ell} \cdot \bar{\mathbf{e}}_\eta) \\ (\hat{\mathbf{e}}_{y_\ell} \cdot \bar{\mathbf{e}}_\xi) & (\hat{\mathbf{e}}_{y_\ell} \cdot \bar{\mathbf{e}}_\eta) \end{bmatrix} \tag{4.32}$$

and

$$\widehat{\mathbf{j}}_\ell = \left[\frac{\partial \widehat{\mathbf{x}}_\ell^{(P)}}{\partial \xi} \mid \frac{\partial \widehat{\mathbf{x}}_\ell^{(P)}}{\partial \eta} \right] = \begin{bmatrix} (\hat{\mathbf{e}}_{x_\ell} \cdot \mathbf{h}_\xi) & | & (\hat{\mathbf{e}}_{x_\ell} \cdot \mathbf{h}_\eta) \\ (\hat{\mathbf{e}}_{y_\ell} \cdot \mathbf{h}_\xi) & | & (\hat{\mathbf{e}}_{y_\ell} \cdot \mathbf{h}_\eta) \end{bmatrix} \tag{4.33}$$

Step 4 Jacobian Determinants and Inverses. The final step is to compute the inverses of $\mathbf{j}_\ell$ and $\widehat{\mathbf{j}}_\ell$ and use formulas (3.9)–(3.11) to obtain the determinant coefficients j_0, j_1 and j_2. Given these primitive quantities, it is then straightforward to compute the shape function laminar derivatives (4.8)–(4.9) and hence all of the kinematic quantities required for the element force and stiffness arrays.

§4.3 Specific Shell-Element Types

The elements to be used in conjunction with the current (CBR2) shell formulation are essentially the same as those used in our earlier work. Several different kinds of 4, 9 and 16-node elements are being adapted for this purpose (Fig. 4). All have in common the use of Lagrange, C^0, interpolation and selective/reduced Gaussian surface quadrature [9].

The $\overline{\mathbf{B}}$ ("B-bar") technique of building the selective (component-by-component) integration into the definition of the B-matrix [8,19] is also used here. However, since the CBR2 B-matrix has a different structure than the CBR version, selective integration has a slightly different connotation. Such differences are explained in the following sample element descriptions.

4.3.1 4-Node Elements

The CBR2 approach was developed primarily for use with *curved* elements, since these are expected to be most effective for large-deformation problems involving thick, inelastic shells (i.e., for reinforced concrete applications). Since the 4-node elements are basically *flat*, they do not benefit from the curvature-corrections introduced in section 3.2 — although they may benefit from the parabolic shear corrections (§3.3). Nevertheless, they will be employed for purposes of comparison with the higher-order elements. In all cases, the current 4-node elements use bilinear shape functions and selective/reduced integration on internal force, stiffness *and* geometric stiffness arrays; refer to [19] for details.

4.3.2 9-Node Elements

The Heterosis element [10], which combines Lagrange (biquadratic) interpolation for rotations and Serendipity (8-node) interpolation for translations, is currently the recommended 9-node element. To enhance convergence without giving rise to communicable mechanisms, we use selective/reduced integration on both transverse-shear and membrane strains. This corresponds to 2×2 quadrature on the entire $\mathbf{B}^{(S)}$ matrix, as well as on partitions of $\mathbf{B}^{(P)}$ that couple to translational DOFs: columns 1–3 of (4.6). On all other terms, we use a *normal* (i.e., nearly exact) rule of 3×3, and, via the B-bar technique, extrapolate the reduced partitions to normal quadrature points before performing the actual integration loop. Furthermore, it has also been found effective to underintegrate the geometric stiffness matrix, as reported in [19].

4.3.3 16-Node Elements

With bi-cubic shape functions, it appears safe to use full 4×4 quadrature on the 16-node elements, as neither locking nor rank deficiency are formally present [18]. However, for thin shells we have found that the convergence rate is significantly improved by using selective 3×3 quadrature in the same manner as described above for the 9-node element. It is likely that similar improvement will be observed for thicker shell problems.

REMARK 4.1

Ideally, we would prefer *uniform* reduced integration for all of the above elements, as this significantly reduces both element formation and stress computation time. However, this also increases rank deficiency, and with it the chances of activating spurious modes. Presently, rigorous techniques for correcting rank deficiency (via stabilization matrices, etc.) are being actively pursued [12–14]; the results will be incorporated in the element library when appropriate.

§4.4 The Global Algorithm

In this section we outline the basic steps involved in obtaining a solution to the finite-element equations of motion for a shell structure. The structural equations (2.41) represent the global assembly of the individual element arrays defined in section 4.2. The constitutive (stress) algorithm, which is embedded within the global (displacement) algorithm, will be discussed subsequently in Section 4.5.

For purposes of illustration, we consider the case of nonlinear dynamics, and employ the generalized Newmark method to integrate the ODE system (i.e., for temporal discretization), and "true-Newton" tangent stiffness updates for nonlinear iteration. The global algorithmic equations to be solved at each nonlinear iteration $(i+1)$ within each time step $(n+1)$ are then [6]:

$$\boxed{\mathbf{K}^{*(i)}_{C_{n+1}} \Delta \mathbf{d}^{(i+1)}_{C_{n+1}} = \mathbf{R}^{*(i)}_{C_{n+1}}} \tag{4.34}$$

for the *iterative* displacement increment, $\Delta \mathbf{d}^{(i+1)}_{C_{n+1}}$, followed by the *corrector* formulas

$$\mathbf{d}^{(i+1)}_{C_{n+1}} = \mathbf{d}^{(i)}_{C_{n+1}} \oplus \Delta \mathbf{d}^{(i+1)}_{C_{n+1}} \tag{4.35}$$

$$\mathbf{v}^{(i+1)}_{C_{n+1}} = \mathbf{v}^{(i)}_{C_{n+1}} + (\frac{1}{\gamma \Delta t}) \Delta \mathbf{d}^{(i+1)}_{C_{n+1}} \tag{4.36}$$

$$\mathbf{a}^{(i+1)}_{C_{n+1}} = \mathbf{a}^{(i)}_{C_{n+1}} + (\frac{1}{\beta \Delta t^2}) \Delta \mathbf{d}^{(i+1)}_{C_{n+1}} \tag{4.37}$$

where $\mathbf{v}$ and $\mathbf{a}$ are approximations of $\dot{\mathbf{d}}$ and $\ddot{\mathbf{d}}$, respectively; β and γ are Newmark algorithmic parameters; Δt is the *time step* increment, and the effective stiffness matrix and residual force vector are defined as

$$\mathbf{K}^* = \frac{1}{\beta \Delta t^2} \mathbf{M} + \mathbf{K} \tag{4.38}$$

and

$$\mathbf{R}^* = \mathbf{F}^{ext} - \mathbf{F}^{int} - \mathbf{M}\mathbf{a} \tag{4.39}$$

The subscript "C" appearing in (4.34)–(4.37) denotes the *computational* basis, that is, the directions used to express the assembled degrees-of-freedom at each node. For shells,

these are usually aligned in some manner with the local surface coordinates, since this is both convenient for boundary conditions and for removal of unnecessary *normal rotation* degrees-of-freedom as well.

Finally, the meaning of the symbolic update operator " $\oplus$ ", which is different from simple addition only for rotations, is explained under Task 4 below.

To complete the specification of the algorithm, the following predictor formulas are used at the beginning of each new time step, $n+1$

$$\mathbf{d}_{n+1}^{(0)} = \mathbf{d}_n \oplus (\Delta t \mathbf{v}_n + \frac{1-2\beta}{2}\Delta t^2 \mathbf{a}_n) \tag{4.40}$$

$$\mathbf{v}_{n+1}^{(0)} = \mathbf{v}_n + (1-\gamma)\Delta t \mathbf{a}_n \tag{4.41}$$

$$\mathbf{a}_{n+1}^{(0)} = \mathbf{0} \tag{4.42}$$

The solve/correct sequence (4.34)–(4.37) is repeated iteratively until convergence at a given time step, i.e., when $\Delta \mathbf{d}$ and $\mathbf{R}^*$ norms become acceptably small. Then the time step is advanced ($n \leftarrow n+1$) by updating the prescribed external force $\mathbf{F}^{ext}$, employing the predictors (4.40)–(4.42), and so on.

The shell-element-related aspects of the above algorithm are now summarized for a single global iteration cycle. Given the current element nodal coordinates $\{\bar{\mathbf{x}}_a\}_{n+1}^{(i)}$, normal vectors $\{\hat{\mathbf{x}}_a\}_{n+1}^{(i)}$, global/computational transformations for translations $[\overline{\mathbf{T}}_a]$, and rotations $[\widehat{\mathbf{T}}_a]_{n+1}^{(i)}$, the stress and constitutive tensor at element integration points, $\{\underline{\sigma}_\ell\}_{n+1}^{(i)}$ and $\{\underline{\mathbf{C}}_\ell\}_{n+1}^{(i)}$, respectively, and corresponding resultant quantities; the following tasks are performed:

Task 1 Element Formation. Form element arrays $\mathbf{K}^e$, $\mathbf{F}^{int^e}$, $\mathbf{F}^{ext^e}$ and $\mathbf{M}^e$, and transform to the nodal computational bases via

$$\mathbf{K}^e_{ab_c} = \mathbf{T}_a^T \mathbf{K}^e_{ab} \mathbf{T}_b \tag{4.43}$$

for matrices, and

$$\mathbf{F}^{int^e}_{a_c} = \mathbf{T}_a^T \mathbf{F}^{int^e}_a \tag{4.44}$$

for vectors, where the nodal block-diagonal transformation matrix, $\mathbf{T}_a$, is defined by

$$\mathbf{T}_a = \begin{bmatrix} \overline{\mathbf{T}}_a & \mathbf{0} \\ \mathbf{0} & \chi_a \widehat{\mathbf{T}}_a \end{bmatrix} \tag{4.45}$$

The orthogonal transformations, $\overline{\mathbf{T}}_a$ and $\widehat{\mathbf{T}}_a$ are assumed to be *user-supplied* at the beginning of the analysis, with the rotational triads, $\widehat{\mathbf{T}}$, updated as explained under Task 4. The defining relations for $\overline{\mathbf{T}}$, $\widehat{\mathbf{T}}$ and χ are as follows:

$$\Delta \bar{\mathbf{u}}_a = \overline{\mathbf{T}}_a \Delta \bar{\mathbf{u}}_{a_c} \tag{4.46}$$

$$\Delta \hat{\mathbf{u}}_a = \chi_a \Delta \hat{\mathbf{u}}_{a_c} = \chi_a \widehat{\mathbf{T}}_a \Delta \boldsymbol{\theta}_{a_c} \tag{4.47}$$

where the skew-symmetric matrix, χ_a, relating rotation increments to increments in relative-displacement of the normal ($\Delta \hat{\mathbf{u}}$), requires only the components of $\hat{\mathbf{x}}_a$, i.e.,

$$\chi_a = \begin{bmatrix} 0 & \hat{x}_{3a} & -\hat{x}_{2a} \\ -\hat{x}_{3a} & 0 & \hat{x}_{1a} \\ \hat{x}_{2a} & -\hat{x}_{1a} & 0 \end{bmatrix} \tag{4.48}$$

Task 2 Assembly. Assemble the element arrays into the global arrays: $\mathbf{K}_C$, $\mathbf{F}^{int}_C$, $\mathbf{F}^{ext}_C$ and $\mathbf{M}_C$; combine to form $\mathbf{K}^*$ and $\mathbf{R}^*$.

Task 3 Incremental Solve. Solve the global matrix equations (4.34) for $\Delta \mathbf{d}^{(i+1)}_{C_{n+1}}$. (If the matrix, $\mathbf{K}^*$, has not been updated, this involves only forward reduction and back substitution; otherwise the matrix is first factored.)

Task 4 Global Displacement Update. Update the nodal configuration according to (4.35)–(4.37). (Here we define the symbol " $\oplus$ ".) For translational components of displacement, (4.35) simply implies

$$\{ \bar{\mathbf{u}}_A \}^{(i+1)}_{n+1} = \{ \bar{\mathbf{u}}_A \}^{(i)}_{n+1} + \{ \Delta \bar{\mathbf{u}}_A \} \tag{4.49}$$

while for rotational components, we instead update the *rotational triads* via

$$[\widehat{\mathbf{T}}_A]^{(i+1)}_{n+1} = [\widehat{\mathbf{T}}_A]^{(i)}_{n+1} \mathbf{Q}(\Delta \boldsymbol{\theta}_{A_c}) \tag{4.50}$$

where $\Delta\boldsymbol{\theta}_{A_C}$ is the rotation increment at node "A" (the global equivalent of element node "a") emanating from the solution vector, $\Delta \mathbf{d}_C$; and the orthogonal matrix operator, $\mathbf{Q}$ (explained thoroughly in [19]) is defined functionally as

$$\mathbf{Q}(\Delta\boldsymbol{\theta}) = \mathbf{I} + \frac{1}{1 + (\Delta\theta^2/4)} \left[\Delta\boldsymbol{\Theta} + \frac{1}{2}\Delta\boldsymbol{\Theta}^2 \right] \tag{4.51}$$

Note that $\Delta\boldsymbol{\Theta}$ is the skew-symmetric matrix corresponding to the vector $\Delta\boldsymbol{\theta}$ (just as $\boldsymbol{\chi}$ corresponds to $\hat{\mathbf{x}}$), and $\Delta\theta$ is its magnitude.

Task 5 Element Displacement Update. Using the total displacements, $\bar{\mathbf{u}}_A$, and the rotational triads, $\hat{\mathbf{T}}_A$, computed globally in Task 4, we then localize to the element level and obtain the new reference surface displacement and pseudo-normal vectors as

$$\{\bar{\mathbf{x}}_a\}_{n+1}^{(i+1)} = \{\bar{\mathbf{x}}_a\}_{n+1}^{(1)} + \{\bar{\mathbf{u}}_a\}_{n+1}^{(i+1)} \tag{4.52}$$

and

$$\{\hat{\mathbf{x}}_a\}_{n+1}^{(i+1)} = [\hat{\mathbf{T}}_a]_{n+1}^{(i+1)} [\hat{\mathbf{T}}_a]_0^T \{\hat{\mathbf{x}}_a\}_0 \tag{4.53}$$

Note that the pseudo-normal vectors, $\hat{\mathbf{x}}_a$, obtained in this way are used only to compute displacement *increments*, $\Delta\hat{\mathbf{u}}_a$, for strain/stress calculations (Task 6). Alternatively, when constructing the CBR2 shell-element kinematic interpolation arrays (e.g., $\mathbf{B}$), we use the reference surface normals as described in Section 3.

Task 6 Constitutive Algorithm. Here, we employ the updated element nodal coordinates (4.52) and pseudo-normal vectors (4.53), and the previous stresses and material-dependent historical quantities to compute the current stresses $\underline{\boldsymbol{\sigma}}_{\ell_{n+1}}^{(i+1)}$ and constitutive matrices $\underline{\mathbf{C}}_{\ell_{n+1}}^{(i+1)}$, resolved in the current lamina basis at each element integration point. The algorithm is sketched in Section 4.5.

Task 7 Stress/Constitutive Resultants — Thickness Integration. This constitutive post-processing step, required in both CBR and CBR2 shell formulations, is also discussed in Section 4.5. The resultant quantities are employed directly in the formation of element arrays for the next iterative cycle.

GO TO Task 1 ($i \leftarrow i + 1$).

---

§4.5 The Constitutive Algorithm

In this section we outline the procedure that has been implemented for computing stresses and the corresponding resultants required as input to the element equilibrium arrays. Specific constitutive models (e.g., elasticity, plasticity, ...) will not be discussed here, but rather a generic algorithm for integrating a class of rate constitutive equations. The algorithm is both implementationally convenient and numerically accurate for problems involving large deformations [7].

4.5.1 Stress Computation

The Truesdell rate constitutive equations (upon which our linearized variational equations are based) may be written as

$$\boxed{\boldsymbol{\sigma}^{\langle T\rangle} = \mathbf{C}\,\dot{\underline{\boldsymbol{\varepsilon}}}} \tag{4.54}$$

where

$$\boldsymbol{\sigma}^{\langle T\rangle} = \dot{\boldsymbol{\sigma}} + \boldsymbol{\sigma}\dot{\boldsymbol{\omega}} - \dot{\boldsymbol{\omega}}\boldsymbol{\sigma} - \hat{\mathbf{C}}\dot{\underline{\boldsymbol{\varepsilon}}} \tag{4.55}$$

and

$$\hat{\mathbf{C}} = \hat{\mathbf{C}}(\boldsymbol{\sigma}) \tag{4.56}$$

and $\dot{\epsilon}_{ij}$ and $\dot{\omega}_{ij}$ are the symmetric and skew symmetric parts of the velocity gradient tensor, $\partial \dot{u}_i/\partial x_j$.

We integrate (4.54) to obtain $\boldsymbol{\sigma}$ via the following incrementally-objective algorithm. The details are discussed in [19].

Given a converged stress state at time (or load) step n, and a displacement increment

between step n and the current global iteration at step $n+1$, we compute

$$\sigma_{\ell_{n+1}} = \overline{\sigma}_{\ell_n} + \Delta\sigma_{\ell_{n+1}} \tag{4.57}$$

where

$$\overline{\sigma}_{\ell_n} = \mathbf{Q}_1 \sigma_{\ell_n} \mathbf{Q}_1^T \tag{4.58}$$

and

$$\Delta\sigma_{\ell_{n+1}} = \mathbf{Q}_2[\overline{\mathbf{C}}_{\ell_n} \Delta\underline{\varepsilon}^{mid}_{\ell_{mid}}]\mathbf{Q}_2^T \tag{4.59}$$

The subscripts ℓ_n, ℓ_{n+1} and ℓ_{mid} refer to the particular lamina basis active at steps n, $n+1$ and $n+1/2$, respectively (i.e., the *time of lamina-resolution*). Except for the incremental quantities, the step number also refers to the *time of evaluation*. Furthermore,

$$\overline{\mathbf{C}} = \mathbf{C} + \hat{\mathbf{C}} \tag{4.60}$$

is the Truesdell-modified material tensor, and

$$\Delta\varepsilon_{ij}^{mid} = \frac{1}{2}\left(\frac{\partial\Delta u_i}{\partial x_j^{mid}} + \frac{\partial\Delta u_j}{\partial x_i^{mid}}\right) \tag{4.61}$$

is the *incremental midpoint strain tensor*, where

$$\Delta\mathbf{u} = \mathbf{u}_{n+1} - \mathbf{u}_n \tag{4.62}$$

and

$$\mathbf{x}^{mid} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1}) \tag{4.63}$$

Finally, $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are orthogonal (rotation) matrices that are functions of the lamina transformations and the skew-symmetric part of $\partial\Delta u_i/\partial x_j^{mid}$. In the presence of small shear deformations (both transverse and in-plane), $\mathbf{Q}_1$ and $\mathbf{Q}_2$ each approach the identity

matrix, I. For the sake of brevity, we will assume this to be the case in the present discussion. The more general case is considered in [19].

To compute $\Delta\underline{\varepsilon}_{\ell_{mid}}^{mid}$ at a typical shell-element integration point, it is merely necessary to replace $\mathbf{x}$ by $\mathbf{x}^{mid}$ in the definition of the $\mathbf{B}$ matrices given in Section 4.2.1. (Note that for shells, (4.63) applies to both $\bar{\mathbf{x}}$ and $\hat{\mathbf{x}}$.) The in-plane and transverse-shear components of the incremental midpoint strain tensor are then evaluated using

$$\Delta\underline{\varepsilon}_{\ell_{mid}}^{(P)mid} = \mathbf{Z}^{(P)} \sum_{a=1}^{N_{en}} \mathbf{B}_a^{(P)mid} \Delta\mathbf{d}_a \tag{4.64}$$

and

$$\Delta\underline{\varepsilon}_{\ell_{mid}}^{(S)mid} = \mathbf{Z}^{(S)} \sum_{a=1}^{N_{en}} \mathbf{B}_a^{(S)mid} \Delta\mathbf{d}_a \tag{4.65}$$

respectively.

This completes the definition of *kinematic* quantities required for the shell constitutive algorithm. Notice, however, that the normal strain component, $(\Delta\underline{\varepsilon}_{\ell_{mid}}^{mid})_{33}$, is not kinematically available from (4.64)–(4.65). For this, the static (zero-normal-stress) constraint is used to extract the unknown strain through the constitutive equations.

In the elastic case, we may solve (4.57) for $(\Delta\underline{\varepsilon}_{\ell_{mid}}^{mid})_{33}$ such that $(\sigma_{\ell_{n+1}})_{33} = 0$. yielding

$$(\Delta\underline{\varepsilon}_{\ell_{mid}}^{mid})_{33} = -\left[(\bar{\sigma}_{\ell_n})_{33} + \sum_{J=1}^{5} (\overline{\mathbf{C}}_{\ell_n})_{3J} (\Delta\underline{\varepsilon}_{\ell_{mid}}^{mid})_J\right] \tag{4.66}$$

where $\overline{\mathbf{C}}_\ell$ and $\Delta\underline{\varepsilon}_\ell^{mid}$ are 6×6 and 6×1 matrix/vector counterparts of the tensor quantities, respectively, arranged so that the 33 lamina component comes last. Substitution of (4.66) in (4.59) then satisfies the static constraint identically.

For the inelastic case, it is suggested that (4.66) be employed in a sub-iterative fashion, so that the updated (inelastic) material tensor, $\overline{\mathbf{C}}$, may be accounted for in the calculation.

4.5.2 Resultants

Given the pointwise stresses and constitutive coefficients, it is then necessary to integrate through the thickness for the corresponding resultant quantities appearing in the element equilibrium arrays. Namely, $\mathbf{s}^{(P)}$ and $\mathbf{s}^{(S)}$ are required for the internal force vector (3.45), $\mathbf{D}^{(P)}$ and $\mathbf{D}^{(S)}$ for the material stiffness matrix (3.50), and $\mathbf{S}$ for the geometric stiffness matrix (3.53).

Numerically, the resultant quantities are generated via one-dimensional Gauss quadrature in a piecewise fashion through an optional sequence of "material" layers. Due to the potential diversity of material properties and layer thicknesses, a different number of quadrature points may be used within each layer. The following thickness integration formula is used for general layered shells

$$\int_{\tilde{z}} f(\tilde{z})\, d\tilde{z} \quad \approx \quad \sum_{l=1}^{N_{layers}} \left[\frac{h_l}{2} \sum_{i=1}^{N_{ip}(l)} w_i f(\tilde{z}_i) \right] \tag{4.67}$$

where h_l is the thickness of layer l, w_i is the Gauss integration weight at point i within layer l and $\tilde{z}_i$ is the corresponding thickness coordinate. The relationship between the Gauss integration coordinate ς_i, which is usually given in the bi-unit interval $(-1,+1)$ — *within each layer* — and the thickness coordinate $\tilde{z}_i$, which is measured from the element reference-surface, is as follows

$$\tilde{z} \quad = \quad \tilde{z}_l + \varsigma \frac{h_l}{2} \tag{4.68}$$

where $\tilde{z}_l$ is the $\tilde{z}$ coordinate at the middle of layer l.

For single-layer shells, at least 2 thickness integration points are required in the elastic case; more for plasticity (e.g., 5–7). For multi-layer shells, it is often possible to make due with less points per layer (i.e., ≥ 1), but this is dependent on the relative layer thicknesses and material properties. It is strongly advised that such problem-dependent decisions be made on the basis of numerical experiment.

5

Summary

matrix, I. For the sake of brevity, we will assume this to be the case in the present discussion. The more general case is considered in [19].

To compute $\Delta\underline{\varepsilon}^{mid}_{\ell_{mid}}$ at a typical shell-element integration point, it is merely necessary to replace x by $\mathbf{x}^{mid}$ in the definition of the B matrices given in Section 4.2.1. (Note that for shells, (4.63) applies to both $\overline{\mathbf{x}}$ and $\widehat{\mathbf{x}}$.) The in-plane and transverse-shear components of the incremental midpoint strain tensor are then evaluated using

$$\Delta\underline{\varepsilon}^{(P)mid}_{\ell_{mid}} = \mathbf{Z}^{(P)} \sum_{a=1}^{N_{en}} \mathbf{B}_a^{(P)mid} \Delta\mathbf{d}_a \tag{4.64}$$

and

$$\Delta\underline{\varepsilon}^{(S)mid}_{\ell_{mid}} = \mathbf{Z}^{(S)} \sum_{a=1}^{N_{en}} \mathbf{B}_a^{(S)mid} \Delta\mathbf{d}_a \tag{4.65}$$

respectively.

This completes the definition of *kinematic* quantities required for the shell constitutive algorithm. Notice, however, that the normal strain component, $(\Delta\underline{\varepsilon}^{mid}_{\ell_{mid}})_{33}$, is not kinematically available from (4.64)–(4.65). For this, the static (zero-normal-stress) constraint is used to extract the unknown strain through the constitutive equations.

In the elastic case, we may solve (4.57) for $(\Delta\underline{\varepsilon}^{mid}_{\ell_{mid}})_{33}$ such that $(\sigma_{\ell_{n+1}})_{33} = 0$. yielding

$$(\Delta\underline{\varepsilon}^{mid}_{\ell_{mid}})_{33} = -\left[(\overline{\sigma}_{\ell_n})_{33} + \sum_{J=1}^{5} (\overline{\underline{C}}_{\ell_n})_{3J} (\Delta\underline{\varepsilon}^{mid}_{\ell_{mid}})_J\right] \tag{4.66}$$

where $\overline{\underline{C}}_\ell$ and $\Delta\underline{\varepsilon}^{mid}_\ell$ are 6×6 and 6×1 matrix/vector counterparts of the tensor quantities, respectively, arranged so that the 33 lamina component comes last. Substitution of (4.66) in (4.59) then satisfies the static constraint identically.

For the inelastic case, it is suggested that (4.66) be employed in a sub-iterative fashion, so that the updated (inelastic) material tensor, $\overline{\mathbf{C}}$, may be accounted for in the calculation.

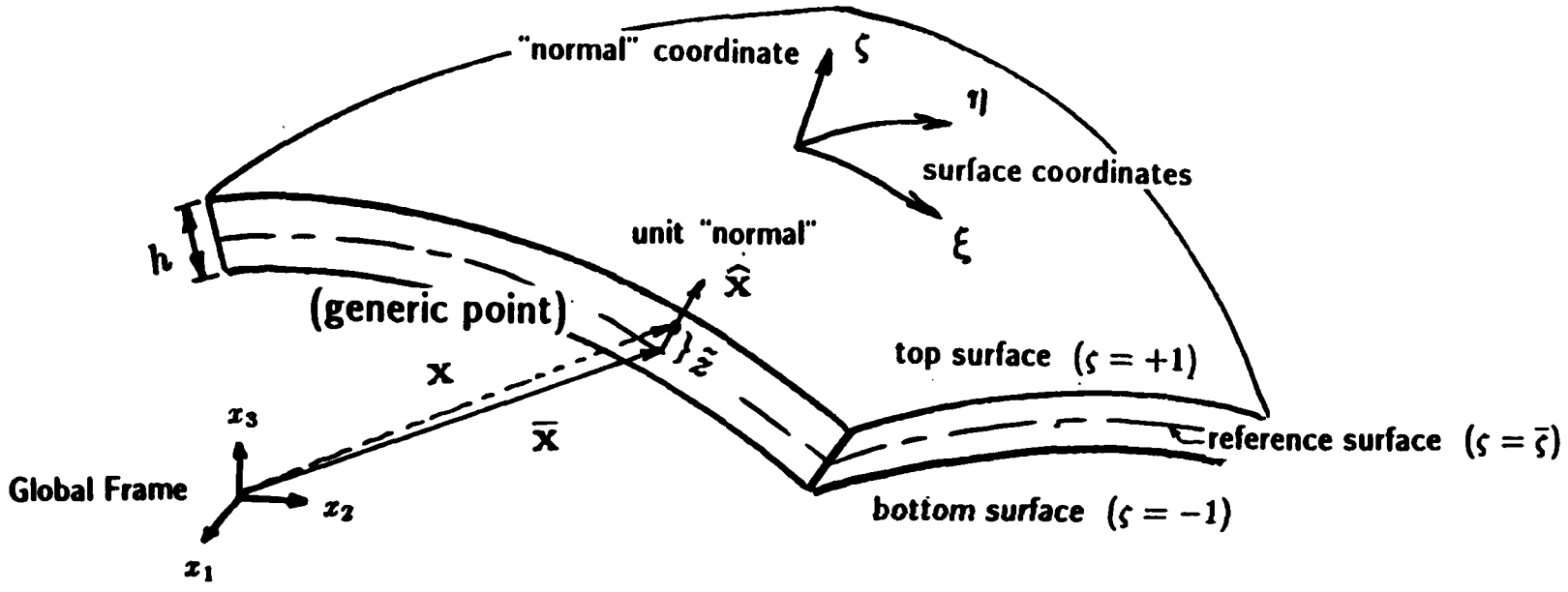
An efficient extension of the nonlinear finite-element procedures presented in [19] for thin shells has been provided herein for the thicker, layer-model type of shell structures encountered in many reinforced concrete applications. The current technique eliminates much of the work associated with through-thickness integration present in the earlier Continuum-Based shell-element formulation [4]. While a similar, resultant-type simplification was introduced in [19] to *first approximation* (i.e., $h/R \ll 1$), the more general case is now handled by including some additional terms that while expanding the element kinematic arrays, still restrict thickness integration to stress and constitutive quantities. This can mean significant element-formation cost savings for multi-layer/elastic-plastic shell problems.

In addition, a higher-order (parabolic) transverse-shear profile has been introduced in an attempt to eliminate the ad-hoc *shear-correction factors* inherited from linear analysis. This is expected to be an improvement for those inelastic problems in which transverse shear plays a significant role. Nevertheless, the representation of transverse shear in such complex cases will require further investigation.

It remains to evaluate the current shell-element procedures using some realistic problems involving reinforced concrete constitutive models. Revisions to the element software implementation are now in progress; these include extension of the element arrays to account for thick-shell behavior (as described in Section 4) and provisions for interfacing with appropriate constitutive modules.

Figures

Typical Configuration



$$\boxed{\mathbf{x}(\xi,\eta,\zeta) \;=\; \bar{\mathbf{x}}(\xi,\eta) \;+\; \tilde{z}\hat{\mathbf{x}}(\xi,\eta)}$$

$$\tilde{z} = (\zeta - \bar{\zeta})\frac{h}{2}$$

Fig. 1. Shell Geometry

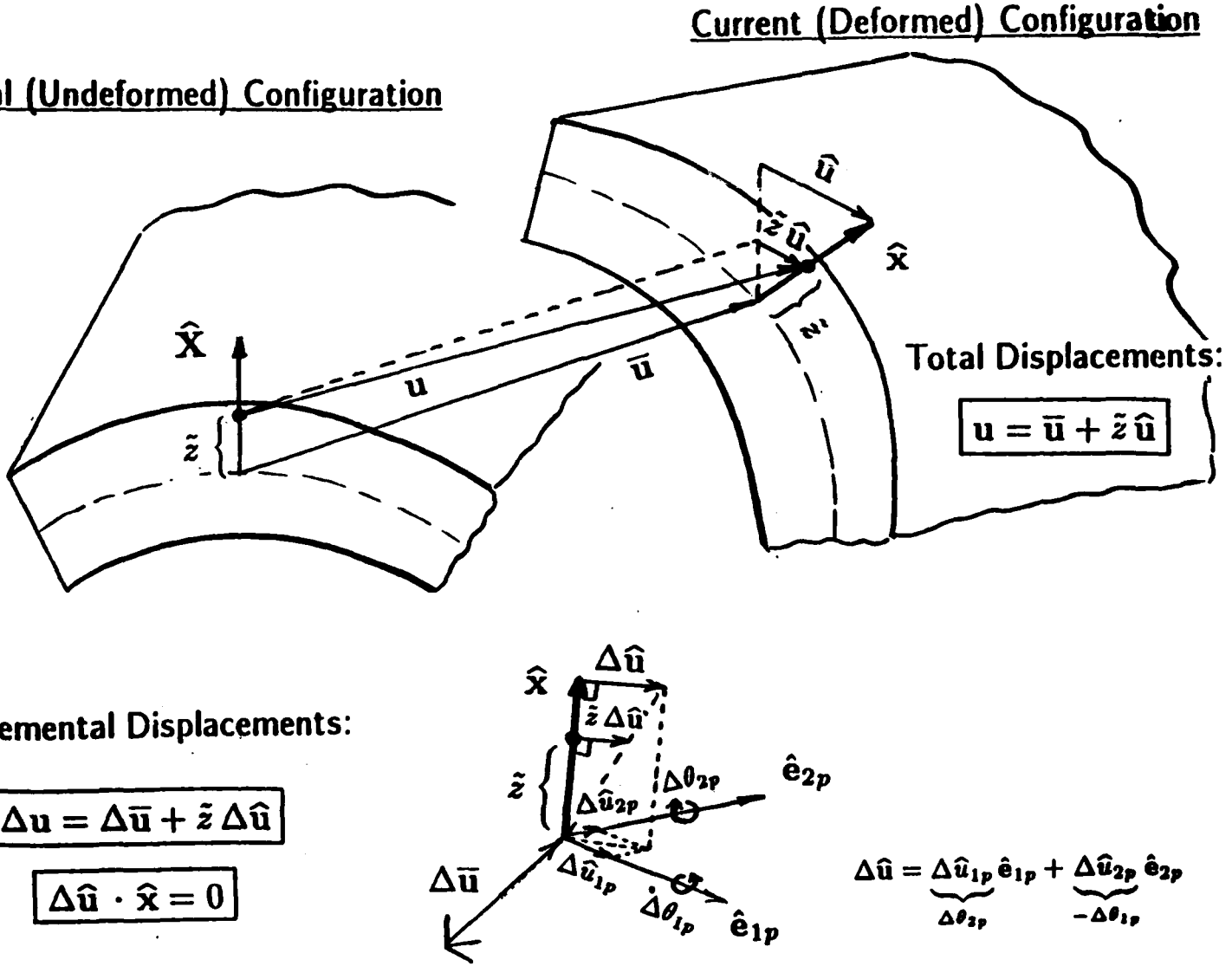


Fig. 2. Shell Kinematics

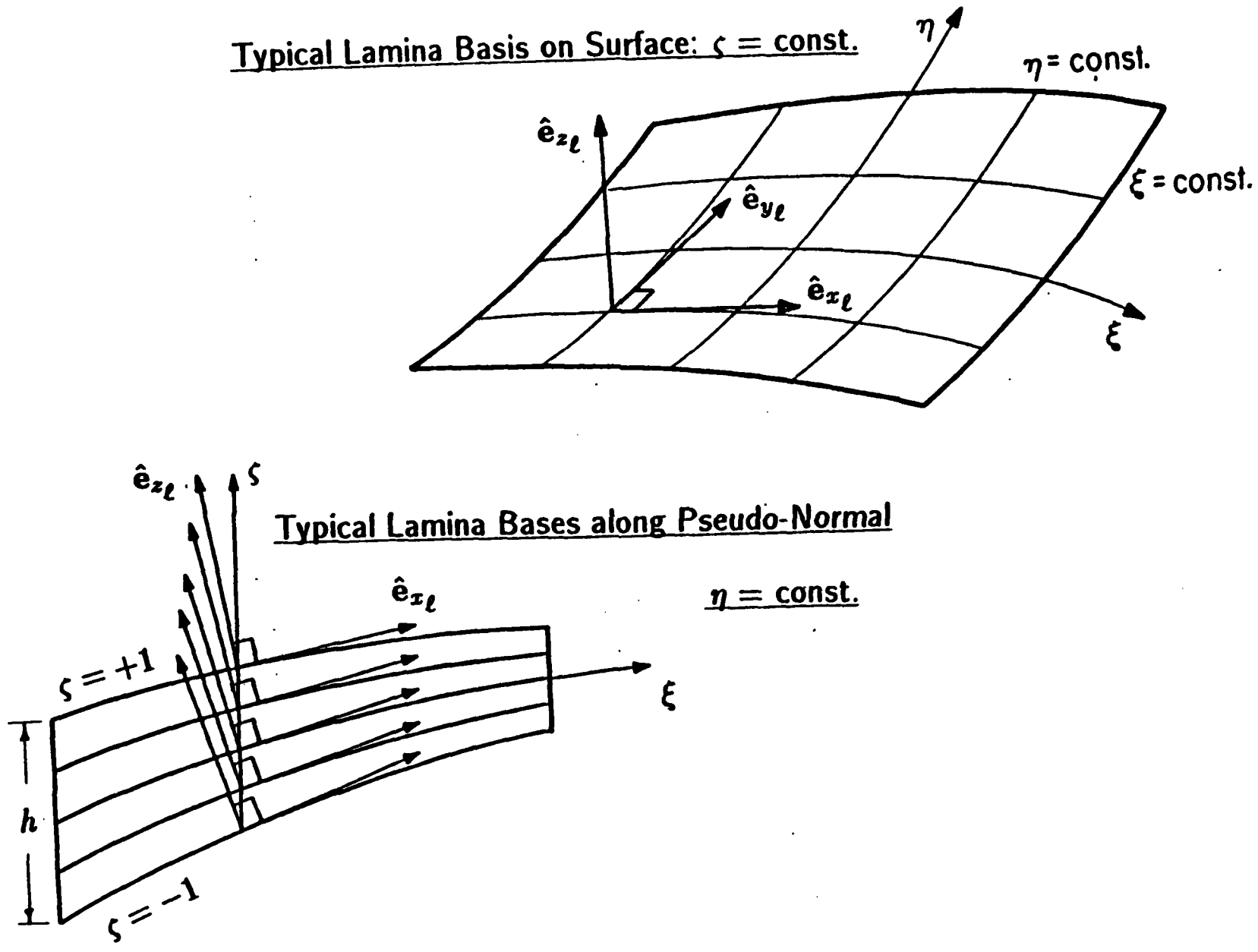
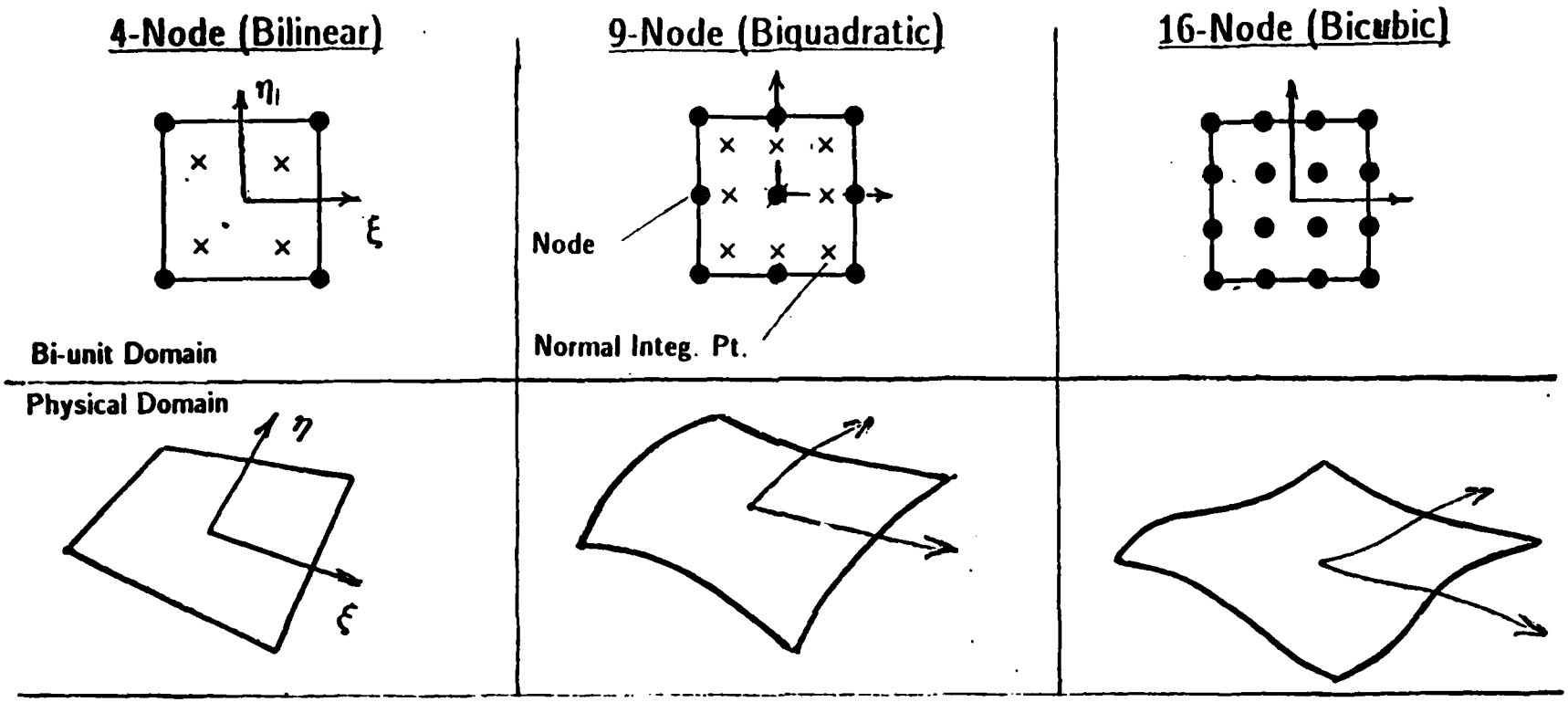


Fig. 3. Lamina Coordinate Basis



Shape Functions

$$N_a = \mathcal{L}_i(\xi)\,\mathcal{L}_j(\eta)$$

$$\bar{x} = \sum_{a=1}^{N_{en}} N_a(\xi,\eta)\,\bar{x}_a$$

(i = nodal row number)
(j = nodal column number)
($n = \sqrt{N_{en}}$)

$$\mathcal{L}_i(\xi) = \frac{(\xi-\xi_1)\cdots(\xi-\xi_{i-1})(\xi-\xi_{i+1})\cdots(\xi-\xi_n)}{(\xi_i-\xi_1)\cdots(\xi_i-\xi_{i-1})(\xi_i-\xi_{i+1})\cdots(\xi_i-\xi_n)}$$

Fig. 4. Basic Lagrange Elements

References

[1] Ahmad, S., Irons, B.M. and Zienkiewiecz, O.C., "Analysis of Thick and Thin Shell Structures by Curved Finite Elements," *Int. J. Num. Meth. Engrg.* 2 (1970) pp. 419–451.

[2] Calladine, C.R., *Theory of Shell Structures*, Cambridge University Press, Cambridge, 1983.

[3] Drysdale, W.H. and Zak, A.R., "Structural Problems in Thick Shells," in *Thin-Shell Structures* (Eds. Y.C. Fung and E.E. Sechler) Prentice-Hall, Englewood Cliffs, New Jersey, 1974, pp 453–464.

[4] Hughes, T.J.R. and Liu, W.K., "Nonlinear Finite-Element Analysis of Shells: Part I," *Int. J. Num. Meth. Engrg.* 26 (1981) pp. 331–362.

[5] Hughes, T.J.R. and Liu, W.K., "Nonlinear Finite-Element Analysis of Shells: Part II," *Int. J. Num. Meth. Engrg.* 27 (1981) pp. 167–181.

[6] Hughes, T.J.R., Liu, W.K. and Levit, I., "Nonlinear Dynamic Finite Element Analysis of Shells", pp. 151–168 in *Nonlinear Finite Element Analysis in Structural Mechanics* (eds. W. Wunderlich et al.), Springer-Verlag, Berlin, 1981.

[7] Hughes, T.J.R. and Winget, J., "Finite Rotation Effects in Numerical Integration of Rate Constitutive Equations Arising in Large-Deformation Analysis", *Int. J. Num. Meth. Engrg.* 15 (1980), pp. 1862–1867.

[8] Hughes, T.J.R., "Generalization of Selective Integration Procedures to Anisotropic and Nonlinear Media," *Int. J. Num. Meth. Engrg.* 15 (1980), pp. 1413–1418.

[9] Hughes, T.J.R., Cohen, M. and Haroun, M., "Reduced and Selective Integration Techniques in the Finite Element Analysis of Plates," *Nucl. Engrg. Design* 46 (1978) pp. 203–222.

[10] Hughes, T.J.R., and Cohen, M., "The 'Heterosis' Family of Plate Finite Elements," Proc. ASCE Electronic Computations Conference, St. Louis, MO, August 6–8, 1979.

[11] Kraus H., *Thin Elastic Shells*, Wiley, New York, 1967.

[12] Park, K.C. and Flaggs, D.L., "An Operational Procedure for the Symbolic Analysis of the Finite Element Method," *Comp. Meth. Appl. Mech. Engrg.* 42 (1984) pp. 37–46.

[13] Park, K.C. and Flaggs, D.L., "A Rank-Sufficient Four-Noded Plate Element with One-Point Integration, Part I: Element Design," to appear in *Comp. Meth. Appl. Mech. Engrg.*

[14] Park, K.C., Stanley, G.M. and Flaggs, D.L., "A Uniformly Reduced, Four-Noded C^0 Shell Element with Consistent Rank Corrections," to appear in *Comp. and Struct.*

[15] Parisch, H., "Nonlinear Analysis of Shells Using Isoparametric Elements," pp. 47–63 in *Nonlinear Finite Element Analysis of Plates and Shells* (eds. T.J.R. Hughes et al.), ASME, New York, 1981.

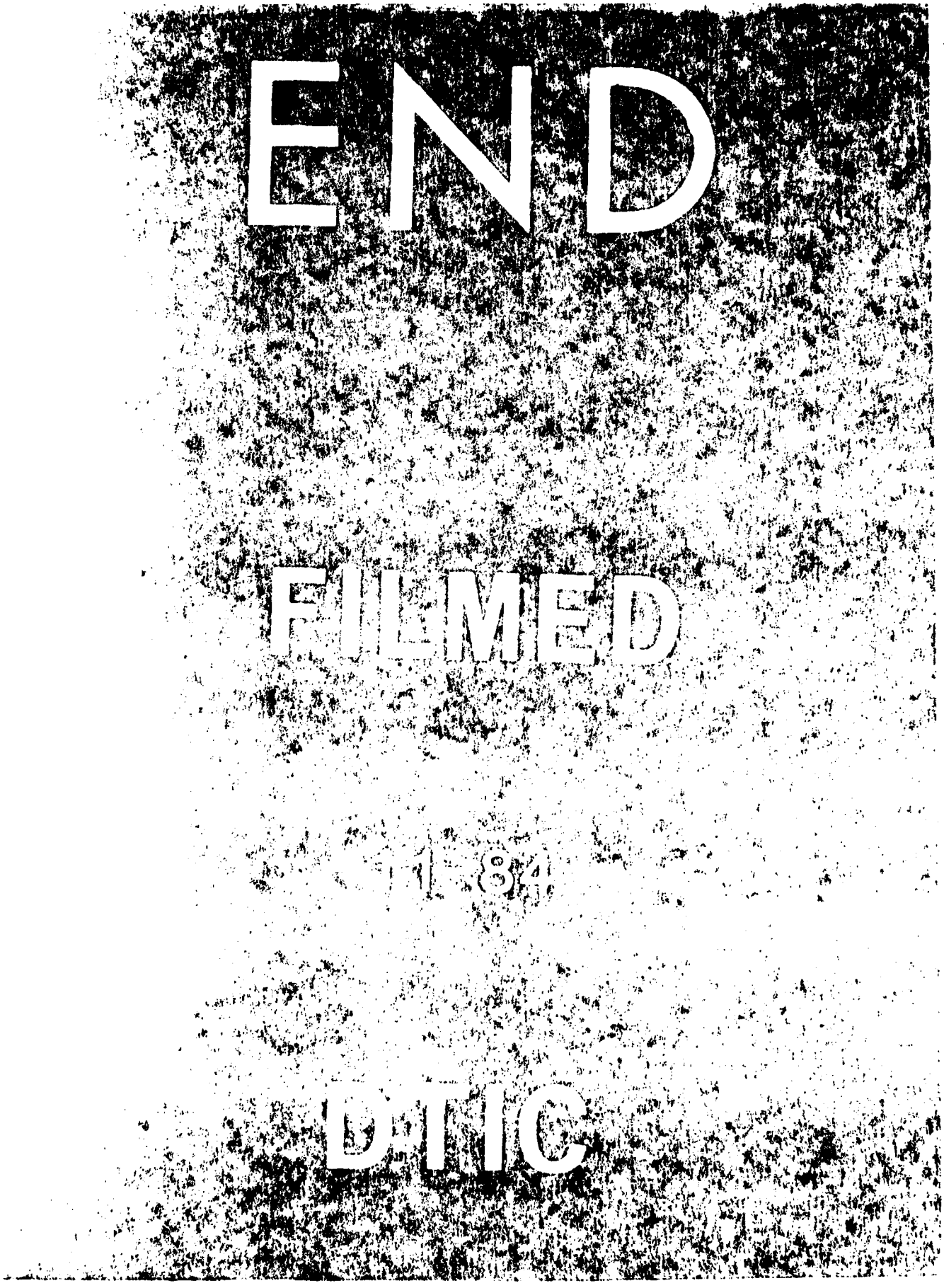
[16] Malvern, L.E., *Introduction to the Mechanics of a Continuous Medium*, Prentice-Hall, Englewood Cliffs, New Jersey, 1969.

[17] Marsden, J.E. and Hughes, T.J.R, *Mathematical Foundations of Elasticity*, Prentice-Hall, Englewood Cliffs, New Jersey, 1983.

[18] Ramm, E. and Sattele, J.M., "Elasto-Plastic Large Deformation Shell Analysis Using Degenerated Elements," pp. 265–282 in *Nonlinear Finite Element Analysis of Plates and Shells* (eds. T.J.R. Hughes et al.), ASME, New York, 1981.

[19] Stanley, G.M., "Continuum-Based Elements for Nonlinear Analysis of Thin Shells," Lockheed Report D879066, in preparation.

[20] Zienkiewicz, O.C., *The Finite Element Method*, McGraw-Hill, London, 1977.


END
FILMED
8-84
DTIC