Structural Analysis and Optimization with a Locally-Cartesian Hybrid Shell Model

by

William Cooper Thalheimer

B.S., Massachusetts Institute of Technology (2014)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2016

Signature redacted

Author .....
Department of Aeronautics and Astronautics
August 18, 2016

Certified by..... Signature redacted .....
Mark Drela
Professor of Aeronautics and Astronautics
Thesis Supervisor

Certified by..... Signature redacted .....
Robert Haimes
Principal Research Engineer in Aeronautics and Astronautics
Thesis Supervisor

Accepted by..... Signature redacted .....
Paulo C. Lozano
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Structural Analysis and Optimization with a Locally-Cartesian Hybrid Shell Model

by

William Cooper Thalheimer

Submitted to the Department of Aeronautics and Astronautics
on August 18, 2016, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

The Hybrid Shell Model (HSM) is presented as an intermediate-fidelity structural model well suited for conceptual design of aerospace vehicles. Although significantly simpler and more economical than full 3D elasticity models, it can still capture full 3D geometries, large deformations, and anisotropic materials. HSM is formulated from the full 3D equilibrium and compatibility equations all projected onto local bases defined on the 2D shell manifold. General anisotropic constitutive equations are also formulated in the local 2D shell manifold bases. The resulting continuous HSM formulation is discretized in weak form with a Galerkin finite element method (FEM), with spherical interpolation used for the local basis vectors. Displacements, basis rotations, and stress resultants are the primary unknowns.

A fully adjoint-consistent plane-stress HSM version (HSM2D) is developed for the purpose of model verification and demonstration of order-of-accuracy convergence. The Method of Exact Solutions (MES) is applied to the case of a uniform plate hanging under its own weight. The effectiveness of the adjoint model for structural optimization is also demonstrated for a simplified rotor blade in a centrifugal force field, featuring non-uniform forcing, non-zero Poisson ratio, large deflection, and optimization of multiple parameters. The suitability of HSM as an intermediate fidelity conceptual aircraft design tool is thus demonstrated.

Thesis Supervisor: Mark Drela
Title: Professor of Aeronautics and Astronautics

Thesis Supervisor: Robert Haimes
Title: Principal Research Engineer in Aeronautics and Astronautics

Acknowledgments

I owe large amounts of gratitude to Mark Drela and Bob Haimes who provided the oversight and large-scale guidance of my project, and to Marshall Galbraith who managed and assisted me on a daily basis. Working hand-in-hand with Mark has polished my research methodology and has instilled in me a drive to constantly improve both my engineering practices and communication. I will never again take a well formatted equation array or concise yet detailed diagram for granted. Bob has been the driving force behind ensuring my project has direction and clarity, and I can thank him first and foremost for giving me the tools to complete it on schedule.

Marshall has played the largest day-to-day role in ensuring my success. He is the kind of supervisor everyone wishes they could have: approachable, attentive, patient, quick-to-respond, thoughtful, and genuinely interested in the work of his advisees. He also possesses the uncanny ability to explain complicated mathematics and code hierarchy to neophytes of both subjects, which has played a large factor in the completion of my graduate studies. I cannot show him enough gratitude for debugging and testing code until midnight in the weeks leading up to my submission deadline so I could make it on time.

I would also like to thank Steve Allmaras for his help both in coding and in derivation of the boundary conditions which proved to be the final missing puzzle piece that made completion of this project possible. Thank you to David Darmofal for his guidance both in this, and in preparatory projects, and for helping me navigate, and in some cases, translate/mediate as the situation demanded, project development in a room full of highly-accomplished, intelligent, and senior colleagues.

Finally, this thesis would be incomplete without acknowledging the contributions of David Darmofal and his research group of Arthur Huang, Savithru Jayasinghe, Hugh Carson, Ben Couchman, Shun Zhang, and Philip Caplan for their contributions to the development of the Solution Adaptive Numerical Solver (SANS) architecture. SANS provided the finite element analysis, parameter sensitivity, error calculation, and optimization functionality to the HSM C++ implementation presented throughout the thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

Nomenclature

Ref. Frames

$\vec{()}^{\mathbf{e}}, \bar{()}^{\mathbf{e}}$	Vector, tensor in local coordinates
$\vec{()}^{\mathbf{w}}, \bar{()}^{\mathbf{w}}$	Vector, tensor in superelement local coordinates
$\mathbf{x}$, $\bar{\bar{\mathbf{x}}}$	Vector, tensor in global cartesian xyz coordinates
x	Scalar quantity

Coordinates

x, y, z	Global cartesian coordinates
ξ, η	Canonical finite-element coordinates
ξ^1, ξ^2	Canonical finite-element coordinates (alternative)
u, v	Generic surface coordinates (e.g. of the CAD system)

Subscripts

$()_0$	Undeformed quantity

Derivatives

$\partial_x()$	Partial derivative w.r.t. x
$\widetilde{\nabla}()$	Surface gradient (excludes normal component)

Indices

α, β	Greek indices take values 1,2
i, j, k, p, q, r	Latin indices take values 1,2,3
$()_n$	Normal component of a locally-defined vector
$()_j$	For finite elements, quantity of the j^{th} corner node of an element
$()_i$	For finite elements, quantity of the i^{th} center node of a superelement

Chapter 1

Introduction

1.1 Motivations from Aircraft Conceptual Design

Conceptual design is an essential first stage in the aircraft design process. The objective of conceptual design is to arrive at a general aerodynamic and structural layout for the aircraft that will satisfactorily meet all the design requirements. At this early stage, every design parameter is considered to be variable. Important design specifications like the aircraft topology, outer mold line (OML) geometry, and structural configuration, must all be selected and optimized over the design process to best satisfy the project objectives at the specified operation condition. Because there are so many design parameters, conceptual design heavily relies on a process of iteration and convergence. This narrows the scope of the design to a point that will allow feasible detail-testing of interesting point-cases in the preliminary and detail design phases to follow.

1.2 Applied Conceptual Design: Modeling and Computation

The large, multi-dimensional design space that exists in the conceptual design phase necessitates a strategy other than prototyping and testing be used to gain insight and narrow the scope of design parameters. Today, computational simulation is by far the dominant approach. In the context of aero-structural design, simulation of an aircraft involves modeling two systems, the airflow surrounding the aircraft, and the physical deflections of the aircraft structure that the airflow imposes. Furthermore, these two systems are two-way coupled. The aerodynaimc forces deform the elastic airframe, and, in turn, the deflected OML geometry modifies the aerodynamic forces. This interaction between aerodynamic and structural formulations means it is a far better approach to model them together, as a coupled system. This coupling is illustrated in Figure 1-1.

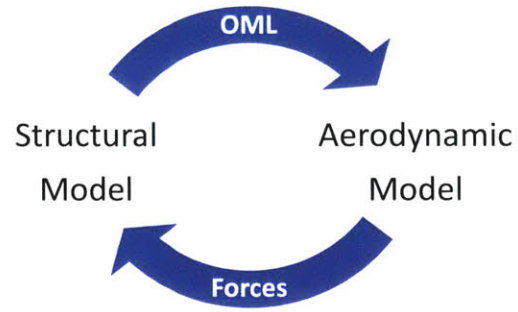


Figure 1-1: The inputs and outputs of the coupled aero-structural model. The aerodynamic model takes the input of an aircraft OML and outputs the pressure forces developed by the airflow. The structural model outputs how the structure will deflect when subjected to these forces.

Aero-structural modeling software necessarily relies on a structural model to provide the stresses, forces, and deformations of loaded solids, and such comprises the bulk of this thesis. Because of the intrinsic reliance on test iteration in the conceptual design phase, the computational implementations of the physical models used to simulate the aircraft's structural response must be able to run quickly. Indeed, it is this premium on computational speed that has historically driven the selection of structural modeling software. To achieve these speed requirements, the historical tendency has been to drastically reduce the fidelity of the model at the conceptual phase to span the full extent of the design space, and then test a handful of interesting cases with high fidelity models to verify the low-fidelity results. Unfortunately, this practice can result in the over-simplified models being used to drive early design methodologies, which can cause designers to make ill-informed decisions in their later detailed designs as they lack the insights of higher-order physical effects. Important examples of this are the couplings of extension, bending, and twist, which may affect the OML such that the aerodynamic forces (in a coupled aero-structural model) are substantially altered from an over-simplified beam-bending case.

Clearly, there is a need for a computational approach to structural modeling that balances calculation speed and high-fidelity real-world approximation. To have practicality in conceptual design and optimization applications, an intermediate level of fidelity is required, where major geometric deformation is captured (beyond simple, linear theories) and therefore available to be coupled to aerodynamic models, but computation time is low.

1.3 Structural Model Requirements

The creation of a new structural model for conceptual design is not purely motivated by the need to address the "fidelity gap" that exists in current structural simulation tools. There are several factors, unavailable as an inclusive set in commercial off-the-shelf (COTS) software, that must be included in this model to facilitate coupling with an aerodynamic model and effective conceptual design.

Fast, "medium-fidelity" model: As previously discussed, the model must capture higher order effects because the aerodynamic response may be disproportionately larger to small OML changes.

Solution sensitivities to variables *and* parameters: Conceptual design frequently involves identifying trends and performing rough optimizations based on the sensitivities of the solution to the input variables and parameters. While many COTS structural modeling suites calculate solution *variable* sensitivities, i.e. the sensitivities to applied stresses and displacements (the primary variables of the partial differential equations used to describe the structural response), a true design tool must also calculate solution *parameter* sensitivities, i.e. the sensitivities to shell thickness, material line direction, and undeformed geometry. Also, this sensitivity calculation must be more efficient than simple, brute-force finite differencing.

Large deformations: A structural deformation model for high aspect ratio configurations must be able to capture geometric nonlinearity from large deflections. However, design-load cases typically assume absence of yield, so simple Hookean constitutive models generally suffice.

Anisotropic material properties: Composite materials are the current state-of-the-art in aerospace structural engineering. Current shell models assume isotropic materials to simplify the equations, and current full-3D models take too long to simulate anisotropic materials. To enable composite ply angle tailoring and optimization, the structural model must include anisotropic material modeling functionality.

Local or global specification of loading and material parameters: Composite aircraft employ the use of fabric layers of varying number and fiber orientations over different parts of the structure; therefore an aerospace structural model must be able to accommodate an arbitrarily complicated material property parametrization.

1.4 The Hybrid Shell Model

This thesis presents the Hybrid Shell Model (HSM), which is well-suited for the conceptual design of aerospace vehicles. HSM is an intermediate fidelity method which idealizes the aircraft structure as a shell located on the aircraft's OML. Structural properties through the volume of 3D components such as spars and ribs are "lumped" so their stiffnesses are smeared across this shell. The shell is characterized in three-dimensional space by having a position and a surface-normal orientation at every point. The surface normal defines a local coordinate system comprised of two in-plane, and one out-of-plane (the surface normal), vectors. In a new approach, HSM uses a Cartesian coordinate frame to define these local coordinates, instead of relying on curvilinear coordinates which only serve to complicate governing equations and computational implementation. The model also includes both displacements and stresses as primary unknowns in the residual state vector, eliminating the problem of shear locking and allowing direct assignment of loading boundary conditions with a Dirichlet formulation.

Chapter 2

Review of Structural Models

HSM has been created to be an "intermediate-fidelity" structural model. To place it on the modeling complexity hierarchy, we briefly review existing structural models, proceeding from lowest to highest fidelity, and therefore most simple to most complex.

2.1 Beam Modeling

A beam is a structures with one dimension much (typically quoted as greater than 10 times) larger than the other two, and is idealized as a line.

Euler-Bernoulli (Simple) Beam Theory (SBT) is the simplest model of elastic deformation that considers the static deflections of beams. The two assumptions in beam theory are that the cross sections of the beam do not deform under the application of transverse or axial loads (deformation can be expressed as two rigid-body translations and one rotation), and that the cross sections of the beam remain planar and normal to the deformed axis of the beam (the rotation occurs along the bending axis so as to remain perpendicular to the normal plane). These are valid when the transverse deflections are small compared to the depth of the beam so that the small angle approximation may be used in the displacement equations. Consequences of these assumptions are that there are no shear strains and no strains in the cross-sectional planes when a beam is exposed to a single axis of flexure. Beam loading can occur simultaneously in multiple transverse directions, which is handled in beam theory by translating the loading orientations into components into the beam's principal centroidal axes of bending and using the superposition of each component as a separate bending load [13], [19].

Rayleigh Beam Theory (RBT) adds the effect of rotary inertia to SBT, allowing it to be used for dynamic beam modeling and eigenmode analysis [13].

Timoshenko Beam Theory (TBT) adds the effect of shear strains to RBT, meaning it eliminates the assumption that a cross section must remain planar and normal to the deformed axis of the beam [13]. This allows the model to be used for slightly larger deflections and for materials that exhibit strong shear coupling to extension or bending.

2.2 Plate Modeling

Plates are planar structural members which have a thickness much smaller (typically quoted as less than one tenth) than the other two dimensions [31].

Kirchoff-Love Plate Theory (KLPT) is an extension of Euler-Bernoulli simple beam theory applied to thin plates [12]. Similar to SBT, KLPT has three assumptions: that straight lines normal to the mid-surface of the plate remain straight after deformation, straight lines normal to the mid-surface remain normal to the mid-surface after deformation, and the the thickness of the plate does not change during a deformation. These assumptions are valid if rotations of the normals of the mid-surface are 10° or less. The only non-zero strains are in the in-plane directions, and the in-plane strains are linear with respect to the in-plane displacement transverse derivatives.

von Karman Extension to KLPT Plate Theory (VKPT) continues to assume the same three assumptions as KLPT, but adds a nonlinear (quadratic) term to the in-plane strain-displacement relation [32]. This helps relate axial forces to transverse displacement [18]. The out-of-plane strains are still considered infinitesimal. The amount of rotation is still assumed small, but the in-plane strains can now be reasonably approximated to 15°.

Mindlin Plate Theory (MPT) is an extension of KLPT that eliminates the assumption that normals of the mid-surface remain normal to the mid-surface after deformation, much as TBT grows from SBT [16]. Without this restriction, MPT incorporates shear deformations through the thickness of a plate, and can be used on thicker plates than KLPT, with the thickness on the order of one tenth that of the other two dimensions (whereas KLPT is best for thinner plates). The theory that is most commonly associated with the term "Mindlin-Reissner plate theory" is actually Mindlin's theory, which also assumes a linear variation of displacement across the plate thickness but zero change in plate thickness during deformation. It also invokes the plane stress condition, which prescribes that the normal stress through the plate is negligible. MPT is a "first-order" shear deformation model because of the linear displacement variation through the thickness [18].

Reissner Plate Theory (RPT) considers deformations caused by transverse shear forces [31] by discarding the assumption of plane stress from MPT [20]. Instead it assumes that the

bending stress is linear and the shear stress is quadratic through the thickness of the plate. Therefore the displacement through the thickness is not necessarily linear and the plate thickness is allowed to change during deformation. It is less commonly used than MPT.

2.3 Shell Modeling

Shells are a generalization of plates. They are structures which have a thickness much smaller (typically quoted as less than one tenth) than the overall dimensions, but they are allowed to exist in three dimensions (i.e. they are not restricted to be flat). The reference-surface of a shell is a 2D manifold existing in the higher-dimensional 3D space.

Directed Surface Theory (DST) also collapses the general three dimensional equations of classical elasticity to a two dimensional manifold comprised of a surface displacement field and surface normal orientation vectors, or "directors" [6]. These directors endow every particle of the continuum with independent rotational degrees of freedom. Strains can thus be measured both by the deformation gradient and by the change in this rotation field [23]. Simo [24] created a singularity-free parametrization of this director field by constraining the function space used to represent the director to be a differentiable manifold modeled on the unit sphere. Simo also made use of bilinear interpolation and spherical interpolation of the directors in his discretization [25, 26, 27]. He makes use of "mixed interpolation", or adding lower order polynomials in his interpolation scheme to overcome the "shear locking" that dominates the performance of displacement finite element formulations (meaning that only displacements compose the primary unknown state vector) in the thin shell limit. Simo's shell theory in particular employs an "assumed strain method" so the strain interpolated differently than the stresses and displacements. Simo and Talamini (whose shell model is based on Simo's [30]) both use an energy functional to define the governing equations and a disconinuous Galerkin (DG) approach to solve their finite element schemes. As a result of the former, the equations lose all sense of directionality so nonisotropic cases cannot be handled. As a result of the latter, the piecewise polynomial representation of the manifold is only C^0 so the director is discontinuous across element edges. If left untreated, this difference in normals would lead to a strain, and therefore deformation energy in the governing energy functional. Talamini address this issue by arbitrarily selecting one of the two nodal state values to formulate the energy functional.

2.4 Solid Modeling

General solid modeling incorporates the full 3D Cauchy stress tensor and is used for structures of arbitrary 3D geometry. No assumptions of relative dimension length or absence of stresses along certain coordinates are made. Loading and deflections in any dimensions are

permitted, but the equations are more complicated than those governing shell models. This general version is what is typically applied in commercial finite element modeling (FEM) software, where the geometry is broken into smaller volumetric elements and the full 3D equations are solved locally in each. This takes considerably more computational power than the geometrical simplifications mentioned earlier so full 3D FEM takes considerably longer to run.

2.5 Method Overview

Classical shell elasticity theory with complex geometry has traditionally been formulated in curvilinear coordinate systems on the shell [22, 4]. The resulting elasticity equations then involve coordinate Christoffel symbols which account for the curvatures of the coordinate lines. This formalism is not only complex, but creates unwarranted demands on geometry smoothness in computational implementations.

Formulations which treat the shell elements as degenerate 3D solids circumvent the problems with curvilinear coordinates by formulating the problem in 3D cartesian space. However, they have their own complications in their need for C^1 or even C^2 continuity of assumed element solution modes, and also have other problems such as shear locking. They also do not capture solid-body rotations exactly without special treatment.

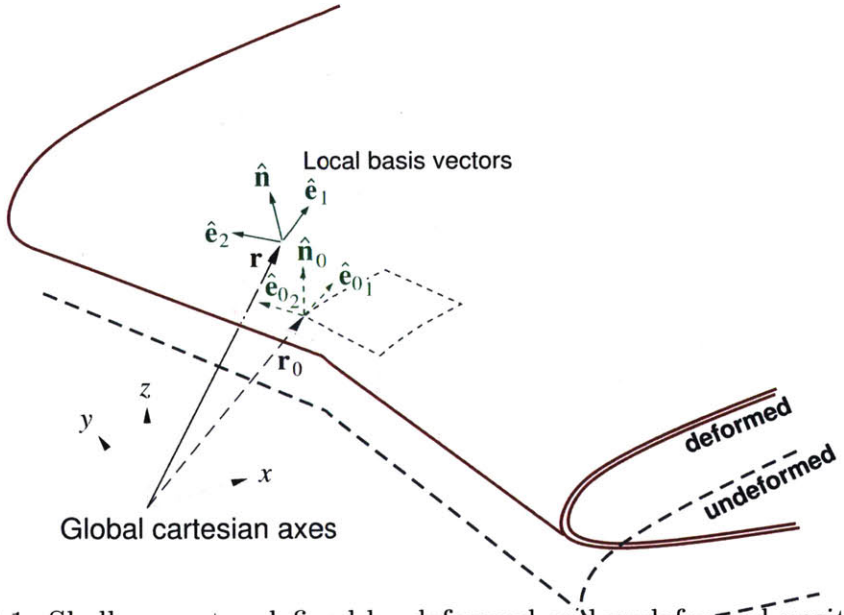


Figure 2-1: Shell geometry defined by deformed and undeformed position vectors $\mathbf{r}, \mathbf{r}_0$. Local in-plane and normal vectors $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}$ are used to define local anisotropic shell properties. All vectors are defined via components along global xyz axes.

The Hybrid Shell Model (HSM) was originally conceived by Mark Drela [9], and developed

into a form that more closely aligns with a classical finite element PDE implementation by William Thalheimer with guidance from Mark Drela, Marshall Galbraith, Steven Allmaras, Robert Haimes, and David Darmofal. The present HSM is formulated in the global 3D cartesian coordinate system, parameterized using local (element) coordinates ξ, η which also define a local basis for forming tangential and normal derivatives, and material strains. This closely follows the analysis of Simo et al [24, 25, 26], except that here the equations are obtained directly from stress equilibrium rather than an energy functional and the continuous Galerkin (CG) finite element method is employed. The present method also defines a complete local basis, in the form of a normal vector and two in-plane vectors, in which general anisotropic materials can be specified independently of the discretization. Simo's method defines only the normal vector (or director), and thus requires that any anisotric properties are referred to the finite element coordinates which can be inconvenient in practice.

Another key feature of HSM is its mixed formulation, where the stress and stress-moment resultants are treated as primary unknowns, which completely eliminates the shear locking problem. The transverse shear strains can then be dropped if they are physically negligible as is typical, which gives a smaller numerical problem. This formulation is called HSM/KL, since the simplification is the same as used in classical Kirchoff-Love [12] plate theory, except here it is applied to a general curved shell geometry. Pure displacement (non-mixed) methods for general geometries cannot drop the transverse shear strains even if they are physically negligible, and thus correspond to the more complex Reissner-Mindlin [16, 20] plate theory. The present HSM formulation which retains the transverse shear strains is called HSM/M. Either the HSM/M or the simpler HSM/KL formulation can be used for any given problem, as appropriate.

HSM can be considered a direct extension of the general nonlinear beam model of ASWING [8] to shell structures, in that it uses the same unknown state vector of position, rotation, and stress and stress-moment resultants. The compatibility, constitutive, and stress-equilibrium equations are all formulated in the global 3D coordinates and then projected onto the shell reference surface. All quantities are parameterized in the local reference-element coordinates with C^0 continuity being sufficient, which allows a relatively simple bi-linear finite element formulation. The same cartesian/projection approach has been successfully used to formulate and solve 3D boundary layer equations on the surface of bodies of complex shapes by Drela [7].

Note on HSM *Flavors*

There currently exist two "flavors" of HSM computational implementation, differing in their choice of coordinate frame for interpolation and governing equation construction.

This flavor of HSM is denoted HSMs (for $\bar{\bar{\mathbf{s}}}$-*matrix*). This version was developed in depth by

William Thalheimer to be more closely aligned with conventional finite element computational methods than the original HSM. The "$\bar{\bar{s}}$-matrix" in question appears in (3.44) and 3.45 and is used to convert from the element-local coordinate system to the "super-element"-local coordinate system before interpolation of orientation vectors so the results are well defined even in situations with degenerate geometric parametrization, as discussed in Section 3.1.2. Since the HSMs stresses and strains are formulated and interpolated in non-degenerate local-basis axes, its residual equations are formulated in terms of their components in the same axes. This is why the discretized governing equations of (4.61), (4.62), (4.63), (4.68), and (4.69) are dotted into their non-degenerate coordinate system components in the residual integrands.

HSMc (for HSM *Cartesian*) has been primarily developed and documented by Mark Drela [9]. HSMc formulates the Cartesian strain and strain-curvature tensors of equations (3.50) and (3.51) directly, just as the stress and stress-moment tensors are calculated here in Section 3.3.3. While this strategy more easily facilitates implementation by avoiding the tensor coordinate transformation of equation (3.49), it violates traditional finite element method construction by introducing a nonlinear function of primary (state vector) variables prior to interpolation. The strategy of converting all entities to the non-degenerate basis in HSMs ensures that only rotations (and therefore linear functions of the primary variables) are performed prior to interpolation. Furthermore, because the HSMc stresses and strains are all formulated and interpolated in Cartesian axes, its residual equations are formulated in terms of their Cartesian components.

Note on Boundary Conditions in Implementation

The implementation of HSM which is presented in Section 7.4 and Chapter 8 has been simplified to a fully adjoint-consistent plane-stress formulation. This version of HSM2D is presented in Chapters 5 and 6. It is important to note that the boundary conditions (BCs) implemented in HSM2D follow from the adjoint-consistent weighted residual formulation of Section 6.4. The method of direct row substitution in the residual Jacobian matrix presented in Section 4.5 is left as an alternate form of BC imposition to be tested in future research.

Chapter 3

The Hybrid Shell Model

3.1 Geometry, Displacements and Rotations

3.1.1 Local Basis

The equations of HSM are formulated on a manifold, a two-dimensional surface existing in three-dimensional space. A point on a manifold is defined at each point in 3-space by a location, (x, y, z), and a surface normal vector, (n_x, n_y, n_z). An arbitrary shell geometry is collapsed to this manifold by means of selecting a "reference surface". The reference surface (e.g. inner, middle, or outer surface) of the deformed shell geometry is defined by the global position vector $\mathbf{r}_{(u,v)}$, while the specified undeformed geometry is defined by $\mathbf{r}_{0(u,v)}$. Here u, v are any convenient surface material coordinates. Locally these will be replaced by the canonical finite-element coordinates ξ, η. The third material coordinate ζ is defined normal to the undeformed reference surface, but will tilt off-normal in the presence of shear strains. As shown in Figures 2-1 and 3-1, each surface point also has a local orthogonal unit-vector basis

$$\bar{\bar{\mathbf{e}}}_{(u,v)} = \bar{\bar{\mathbf{e}}}_{(\mathbf{\Lambda}_{(u,v)})} \equiv \begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} \tag{3.1}$$

$$\bar{\bar{\mathbf{e}}}_{0(u,v)} = \bar{\bar{\mathbf{e}}}_{0(\mathbf{\Lambda}_{0(u,v)})} \equiv \begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_{0_1} & \hat{\mathbf{e}}_{0_2} & \hat{\mathbf{n}}_0 \\ | & | & | \end{bmatrix} \tag{3.2}$$

where $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$ are tangent and $\hat{\mathbf{n}}$ is normal to the reference surface.

Selection of log-quaternion definition of local basis: Because a manifold must be able to represent any arbitrary geometry, the local basis vectors must be carefully defined such that ambiguity is eliminated in all surface configurations. This means that describing the surface orientation with Euler angles is problematic, because Euler angles can exhibit gimbal

lock and therefore a loss of a rotational degree-of-freedom. For this reason, quaternions have been selected to describe the manifold surface basis vector orientations. While less intuitive than Euler angles, they do not suffer from gimbal lock and therefore are more suited to describe arbitrary surfaces.

The $\bar{\bar{\mathbf{e}}}$ and $\bar{\bar{\mathbf{e}}}_0$ basis vector sets are defined not in terms of quaternions themselves however, but rather in terms of the unit-quaternion logarithms $\mathbf{\Lambda}$ and $\mathbf{\Lambda}_0$. The logarithms are chosen as primary variables because they may be bilinearly interpolated directly (as in Section 4.2.2), whereas the quaternions themselves must use SLERP for interpolation. In practice, the specified $\mathbf{r}_0$ will also be used to define $\mathbf{\Lambda}_0$, and both $\mathbf{r}$ and $\mathbf{\Lambda}$ will be treated as a primary unknown variables.

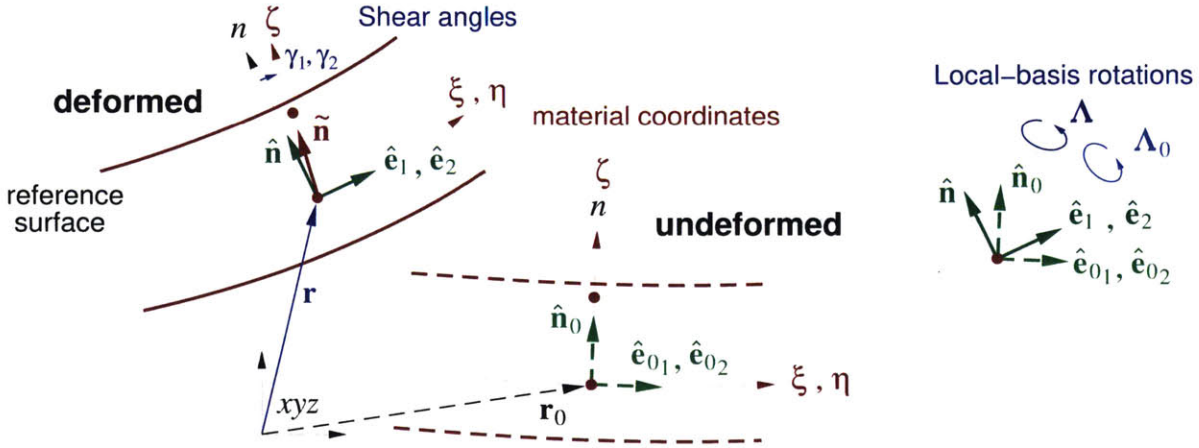


Figure 3-1: View through thickness of shell showing position $\mathbf{r}$ and local basis vectors $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}$, in local material coordinates ξ, η, ζ.

The explicit definition of $\bar{\bar{\mathbf{e}}}_{(\mathbf{\Lambda})}$ is via the actual quaternion $(p_0, p_1, p_2, p_3) = \exp(\mathbf{\Lambda})$, as follows.

$$p_0 = \cos|\mathbf{\Lambda}| \quad \simeq \quad 1 - \tfrac{1}{2}|\mathbf{\Lambda}|^2 \tag{3.3}$$

$$\begin{Bmatrix} p_1 \\ p_2 \\ p_3 \end{Bmatrix} = \frac{\mathbf{\Lambda}}{|\mathbf{\Lambda}|} \sin|\mathbf{\Lambda}| \simeq \mathbf{\Lambda}\left(1 - \tfrac{1}{6}|\mathbf{\Lambda}|^2\right) \tag{3.4}$$

$$\bar{\bar{\mathbf{e}}}_{(\mathbf{\Lambda})} = \begin{bmatrix} 1-2(p_2^2+p_3^2) & 2(p_1p_2-p_0p_3) & 2(p_1p_3+p_0p_2) \\ 2(p_1p_2+p_0p_3) & 1-2(p_1^2+p_3^2) & 2(p_2p_3-p_0p_1) \\ 2(p_1p_3-p_0p_2) & 2(p_2p_3+p_0p_1) & 1-2(p_1^2+p_2^2) \end{bmatrix} \tag{3.5}$$

The approximate expressions in equations (3.3) and (3.4) are used whenever $|\mathbf{\Lambda}|^2 < \epsilon_{\text{machine}}$, which prevents roundoff errors or a divide by zero in this limiting case. Definitions (3.3)–(3.5) are also used to relate $\bar{\bar{\mathbf{e}}}_0$ and $\mathbf{\Lambda}_0$.

The quaternion elements $p_0 \ldots p_3$ are equivalent to Euler-Rodrigues rotation parameters,

and (3.5) is equivalent to the Rodrigues rotation matrix. We note that if $p_1, p_2, p_3 \ll 1$, the above definition reduces to the small-rotation matrix,

$$\bar{\bar{\mathbf{e}}} \simeq \begin{bmatrix} 1 & -2p_3 & 2p_2 \\ 2p_3 & 1 & -2p_1 \\ -2p_2 & 2p_1 & 1 \end{bmatrix} \tag{3.6}$$

which corresponds to linear elasticity if the components from $\mathbf{\Lambda}_0$ are also small. In contrast, the full nonlinear form (3.5) exactly represents solid-body rotations, and hence is usable for arbitrarily large overall deformations.

We also define a material quasi-normal vector

$$\widetilde{\mathbf{n}}_{(u,v)} = \hat{\mathbf{n}} + \Delta\mathbf{n} \tag{3.7}$$

$$\Delta\mathbf{n} \equiv \gamma_1 \hat{\mathbf{e}}_1 + \gamma_2 \hat{\mathbf{e}}_2 \quad ; \qquad \gamma_1 = 2\varepsilon_{1n} \quad , \qquad \gamma_2 = 2\varepsilon_{2n} \tag{3.8}$$

where $\gamma_{1(u,v)}, \gamma_{2(u,v)}$ are tilt angles, shown in Figure 3-1, corresponding to the shell's out-of-plane shear deformations $\varepsilon_{1n}, \varepsilon_{2n}$. These are assumed to be small, so that $\widetilde{\mathbf{n}}$ is still a unit vector to first order. Note also that $\widetilde{\mathbf{n}}_0 = \hat{\mathbf{n}}_0$, since $\varepsilon_{1n}, \varepsilon_{2n}$ are by definition zero for the undeformed geometry. In HSM/M, $\varepsilon_{1n}, \varepsilon_{2n}$ are directly related via shear stiffnesses to the out-of-plane shear stress resultants f_{1n}, f_{2n} which are primary unknowns. In HSM/KL, $\varepsilon_{1n}, \varepsilon_{2n}$ are assumed to be negligible and are dropped, which is adequate and appropriate for shells which are very thin relative to their overall dimensions. This also has the benefit of producing a smaller numerical problem since f_{1n}, f_{2n} can now be omitted from the list of unknown variables.

3.1.2 Non-degenerate Coordinate Basis

One possible complication of using the $\bar{\bar{\mathbf{e}}}_{(u,v)}$ basis for the entire shell surface is that a natural definition such as $\hat{\mathbf{e}}_1 = \partial_u \mathbf{r} / |\partial_u \mathbf{r}|$ will produce numerical problems at coordinate degeneracy points where $\partial_u(\,)$ is singular.

Coordinate degeneracy points, such as the one shown in Figure 3-3 on the left, are unavoidable on a simply-connected closed surface such as a sphere. This is related to the so-called "Hairy Ball Theorem" of algebraic topology which states that any tangent vector field such as $\hat{\mathbf{e}}_{1(u,v)}$ (however defined), cannot be differentiable or even continuous everywhere on the closed surface. To solve this issue, we define alternative tangent vectors $\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2$ in the neighborhood (e.g. finite element) of a reference point, which are the $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$ vectors at a single reference point distributed over the element via spherical interpolation, as sketched in Figure 3-3.

Figure 3-2: A "hairy ball" attempted to be combed flat but instead creating tufts at the poles. [33]

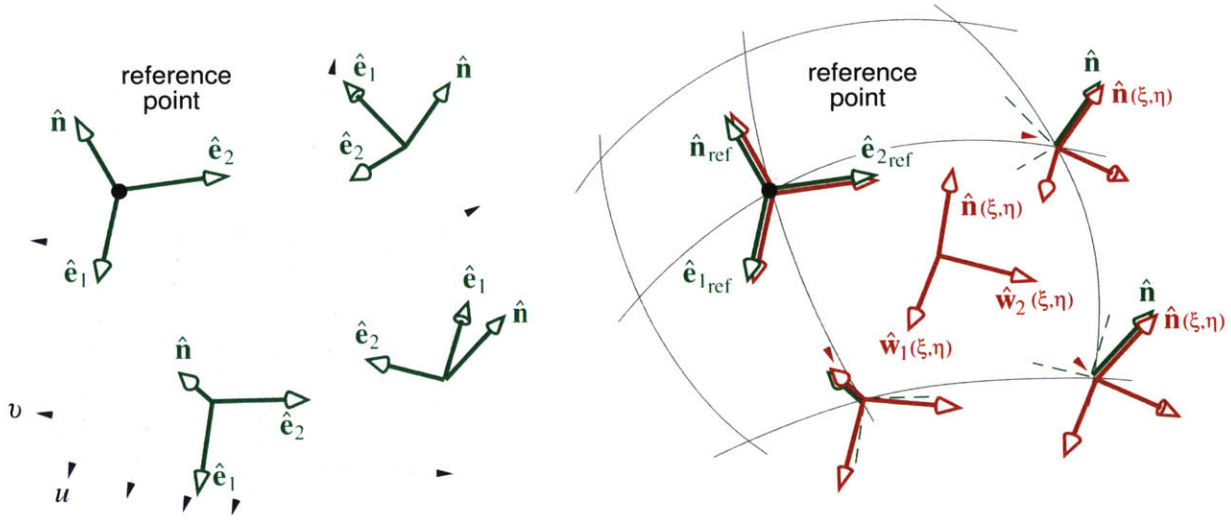


Figure 3-3: The $\hat{\mathbf{e}}_{1_{\text{ref}}}, \hat{\mathbf{e}}_{2_{\text{ref}}}, \hat{\mathbf{n}}_{\text{ref}}$ basis vectors at a reference point are distributed locally over the neighborhood of the reference point, thus defining the $\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, \hat{\mathbf{n}}_{(\xi,\eta)}$ basis vector fields which are continuous and differentiable. In contrast, the $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_{2(\xi,\eta)}$ vector fields may be discontinous, such as at a coordinate fold point shown on the left. The normal vector $\hat{\mathbf{n}}_{(\xi,\eta)}$ is common to both basis systems.

Physically, the orientation of $\hat{\mathbf{n}}$ basis vectors is primarily determined by the combination of the constitutive and stress-equilibrium equations (presented later). The $\hat{\mathbf{n}}_{(\xi,\eta)}$ distribution then determines the deformed geometry via the equations

$$\boxed{\begin{aligned} \hat{\mathbf{n}} \times (\partial_\xi \mathbf{r} \times \partial_\eta \mathbf{r}) \cdot \hat{\mathbf{w}}_1 \; - \; \hat{\mathbf{n}}_0 \times (\partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0) \cdot \hat{\mathbf{w}}_{0_1} \; &= \; 0 \\ \hat{\mathbf{n}} \times (\partial_\xi \mathbf{r} \times \partial_\eta \mathbf{r}) \cdot \hat{\mathbf{w}}_2 \; - \; \hat{\mathbf{n}}_0 \times (\partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0) \cdot \hat{\mathbf{w}}_{0_2} \; &= \; 0 \end{aligned}} \tag{3.9}$$

which in effect tilt the $\mathbf{r}_{(\xi,\eta)}$ surface to make its tangential derivatives $\partial_\xi \mathbf{r}$ and $\partial_\eta \mathbf{r}$ normal to $\hat{\mathbf{n}}$. In each equation the second term involving the undeformed geometry is exactly zero analytically, but not necessarily numerically. It is therefore subtracted off so that we get $\mathbf{r} = \mathbf{r}_0$ exactly for the unloaded case, for any discretization.

A third equation is needed to constrain the rotation of $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$ about $\hat{\mathbf{n}}$ and hence within the surface at each point, with no effect on the geometry. Since the properties of a possibly non-isotropic shell material will be specified in the 1,2 axes along $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$, the physical requirement on their orientation is that they remain fixed to the material, with appropriate allowance for shear angle changes. We therefore first define material line vectors $\mathbf{s}_0$ and $\tilde{\mathbf{s}}$ for

the undeformed and deformed geometries, as sketched in Figure 3-4.

$$\mathbf{s}_0 = \partial_\xi \mathbf{r}_0 \tag{3.10}$$

$$\tilde{\mathbf{s}} = \partial_\xi \mathbf{r} \tag{3.11}$$

This $\tilde{\mathbf{s}}$ is analogous to the $\tilde{\mathbf{n}}$ quasi-normal material vector. The choice to define $\tilde{\mathbf{s}}$ along $\partial_\xi \mathbf{r}$ is arbitrary, and $\partial_\eta \mathbf{r}$ could have been chosen as well.

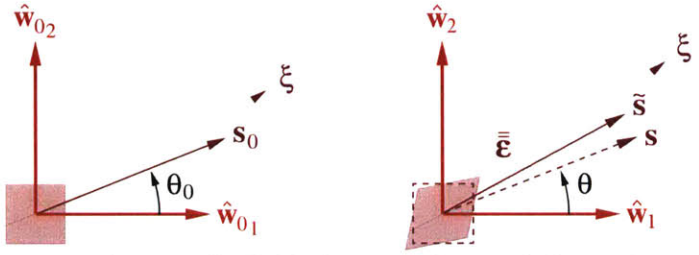


Figure 3-4: Top view of a small shell element for undeformed and deformed geometries, with material vectors $\mathbf{s}_0$ and $\tilde{\mathbf{s}}$. Requiring $\theta = \theta_0$ makes the $\hat{\mathbf{w}}$ basis vectors after deformation to be fixed to the same material as $\hat{\mathbf{w}}_0$, which are locally tied to $\hat{\mathbf{e}}$ and $\hat{\mathbf{e}}_0$, respectively.

We also define a "shear-corrected" material vector $\mathbf{s}$ by removing from $\tilde{\mathbf{s}}$ the rotation due to the in-plane strain tensor $\bar{\bar{\varepsilon}}$,

$$\tilde{\mathbf{s}} = \mathbf{s} + \bar{\bar{\varepsilon}}\,\mathbf{s} = \left[\bar{\bar{\mathbf{I}}} + \bar{\bar{\varepsilon}}\right] \mathbf{s} \tag{3.12}$$

$$\mathbf{s} = \left[\bar{\bar{\mathbf{I}}} + \bar{\bar{\varepsilon}}\right]^{-1} \tilde{\mathbf{s}} \simeq \left[\bar{\bar{\mathbf{I}}} - \bar{\bar{\varepsilon}}\right] \tilde{\mathbf{s}} \tag{3.13}$$

where the convenient approximation in equation (3.13) is valid for the usual small-strain case $\varepsilon \ll 1$. The orientation of the $\hat{\mathbf{w}}_1$ and $\hat{\mathbf{w}}_2$ vectors, and hence the orientation of $\hat{\mathbf{e}}_{1_{\text{ref}}}$ and $\hat{\mathbf{e}}_{2_{\text{ref}}}$ which define them, is then imposed by requiring that $\theta = \theta_0$ in Figure 3-4.

$$\theta = \text{atan2}\left(\, \mathbf{s} \cdot \hat{\mathbf{w}}_2 \;,\; \mathbf{s} \cdot \hat{\mathbf{w}}_1 \,\right) \tag{3.14}$$

$$\theta_0 = \text{atan2}\left(\, \mathbf{s}_0 \cdot \hat{\mathbf{w}}_{0_2} \;,\; \mathbf{s}_0 \cdot \hat{\mathbf{w}}_{0_1} \,\right) \tag{3.15}$$

$$\boxed{\theta - \theta_0 = 0} \tag{3.16}$$

3.2 Equilibrium Equations

The starting point is the following 3D stress-equilibrium differential equation for a material with stress tensor $\bar{\bar{\boldsymbol{\sigma}}}$, density ρ, acceleration $\mathbf{a}$, and gravity $\mathbf{g}$.

$$\nabla \cdot \bar{\bar{\boldsymbol{\sigma}}} \, + \, \rho(\mathbf{g}-\mathbf{a}) \;=\; \mathbf{0} \tag{3.17}$$

It is noted that while the Eulerian gradient ∇ (w.r.t. the deformed is technically correct for imposing force equilibrium on the static state of the deformed geometry, the alternative formulation with Lagrangian gradients ∇_0 (w.r.t the undeformed geometry) also has its advantages, as will be elaborated upon in subsequent sections.

$$\nabla_0 \cdot \bar{\bar{\boldsymbol{\sigma}}} \, + \, \rho(\mathbf{g}-\mathbf{a}) \;=\; \mathbf{0} \tag{3.18}$$

3.2.1 Thickness Resultant Integrals

The integral equilibrium equations for a shell will involve the following mass, stress, and stress-moment resultant integrals over the shell thickness, which are then functions of the surface coordinates ξ, η. Only the in-plane $1,2$ components of the $\bar{\bar{\mathbf{m}}}$ tensor are significant, however.

$$\mu_{(\xi,\eta)} \;\equiv\; \int_{n_{\text{bot}}}^{n_{\text{top}}} \rho \; \mathrm{d}n \quad , \qquad \bar{\bar{\mathbf{f}}}_{(\xi,\eta)} \;\equiv\; \int_{n_{\text{bot}}}^{n_{\text{top}}} \bar{\bar{\boldsymbol{\sigma}}} \; \mathrm{d}n \quad , \qquad \bar{\bar{\mathbf{m}}}_{(\xi,\eta)} \;\equiv\; \int_{n_{\text{bot}}}^{n_{\text{top}}} \bar{\bar{\boldsymbol{\sigma}}} \, n \; \mathrm{d}n \tag{3.19}$$

Note that at a shell element edge with unit normal $\hat{\mathbf{t}}$ as shown in Figure 3-5, $\bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}}$ is the overall edge traction force/length vector, and $\hat{\mathbf{n}} \times \bar{\bar{\mathbf{m}}} \cdot \hat{\mathbf{t}}$ is the edge bending moment/length vector. Also appearing will be the net top–bottom surface traction stress,

$$\mathbf{q}_{(\xi,\eta)} \;\equiv\; \bar{\bar{\boldsymbol{\sigma}}}_{\text{top}} \cdot \hat{\mathbf{n}}_{\text{top}} \, + \, \bar{\bar{\boldsymbol{\sigma}}}_{\text{bot}} \cdot \hat{\mathbf{n}}_{\text{bot}} \;\simeq\; (\bar{\bar{\boldsymbol{\sigma}}}_{\text{top}} - \bar{\bar{\boldsymbol{\sigma}}}_{\text{bot}}) \cdot \hat{\mathbf{n}} \tag{3.20}$$

which is a force/area vector. The last approximate form assumes that the top and bottom surface normal vectors are anti-parallel, which is equivalent to assuming that the shell has negligible thickness variations.

3.2.2 In-Plane Equation Basis Vectors

The in-plane force and moment equations must be resolved along in-plane basis vectors in the numerical solution. Since the $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ vectors defined earlier might possibly have multiple values or undefined gradients at degenerate surface-coordinate points, we instead will use the alternative in-plane basis vectors $\hat{\mathbf{w}}_1$ and $\hat{\mathbf{w}}_2$ which will be defined locally, e.g. over only adjacent finite elements which define an equation residual, rather than over an entire surface.

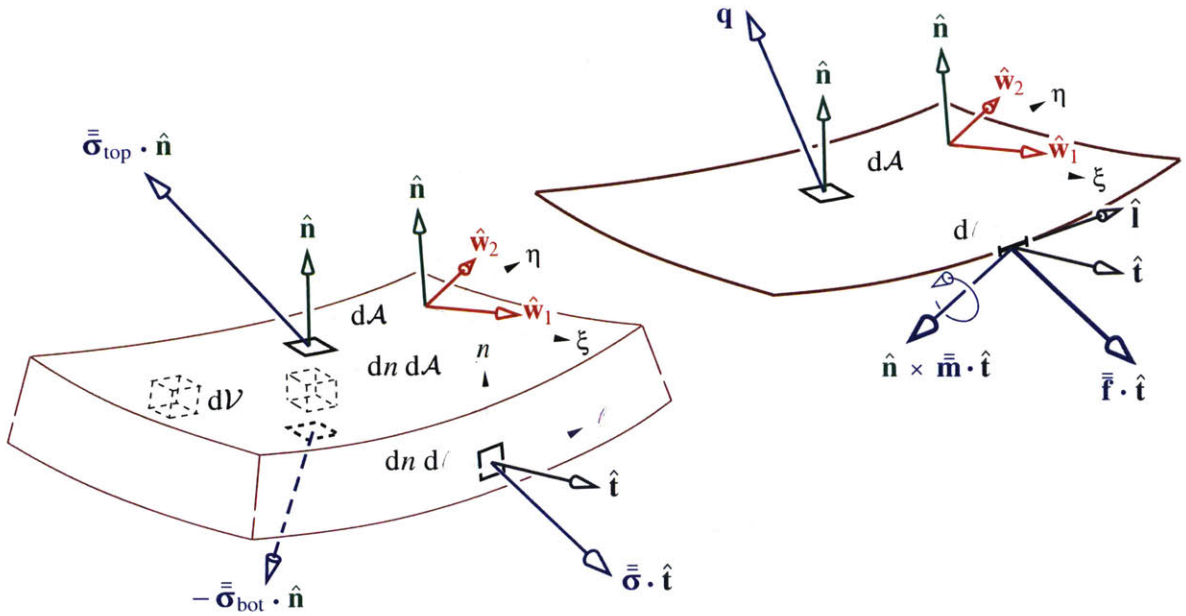


Figure 3-5: Shell element volume used to formulate integral equilibrium equations for the shell's idealized zero-thickness representation on the right. Tractions on volume surfaces become net force loading $\mathbf{q}$ on the shell area, and force loading $\bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}}$ and moment loading $\hat{\mathbf{n}} \times \bar{\bar{\mathbf{m}}} \cdot \hat{\mathbf{t}}$ on the shell edges. In-plane equlibrium equations are resolved along $\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2$ vectors associated with the element.

3.2.3 In-Plane Force Equilibrium

The integral shell in-plane force-equilibrium equations are obtained by forming [eq.(3.17)] $\cdot$ $\hat{\mathbf{w}}_1 W$ and [eq.(3.17)] $\cdot \hat{\mathbf{w}}_2 W$, where $W_{(\xi,\eta)}$ is a weighting function which is nonzero only over a finite element such as the one shown in Figure 3-5. We then expand the divergence term, and integrate over the element volume,

$$\iiint \Big\{ \nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{w}}\, W) \;-\; (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla) \cdot (\hat{\mathbf{w}}\, W) \;+\; \rho\,(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{w}}\, W \Big\}\, \mathrm{d}\mathcal{V} \;=\; 0 \qquad (3.21)$$

where $\hat{\mathbf{w}}$ denotes either in-plane vector $\hat{\mathbf{w}}_1$ or $\hat{\mathbf{w}}_2$.

The first pure divergence volume integral in equation (3.21) is replaced by area integrals over the perimeter surface with normal $\hat{\mathbf{t}}$ and area elements $\mathrm{d}n\,\mathrm{d}\ell$, and over the top/bottom surfaces with normals $\pm\hat{\mathbf{n}}$ and area elements $\mathrm{d}\mathcal{A}$. For the remaining volume integrals the volume element is written as $\mathrm{d}\mathcal{V} = \mathrm{d}n\,\mathrm{d}\mathcal{A}$. Equation (3.21) then becomes

$$\boxed{\begin{aligned} &\oint \hat{\mathbf{w}} \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}}\, W\, \mathrm{d}\ell \\ &\qquad + \iint \Big[-\Big(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\Big) \cdot (\hat{\mathbf{w}}\, W) \;+\; \mathbf{q} \cdot \hat{\mathbf{w}}\, W \;+\; \mu\,(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{w}}\, W \Big]\, \mathrm{d}\mathcal{A} \;=\; 0 \end{aligned}} \qquad (3.22)$$

where the integrations $\int \mathrm{d}n$ across the shell thickness have been carried out in the μ,$\bar{\bar{\mathbf{f}}}$ definitions (3.19), and the $\mathbf{q}$ definition (3.20) has also been invoked. The fact that $W_{(\xi,\eta)}$ is defined to not vary in n allowed it to be put outside the $\int \mathrm{d}n$ thickness integrals. Also, its divergence only involves in-plane contributions, as quantified by the in-plane gradient operator $\widetilde{\nabla}$, which excludes any normal components along $\hat{\mathbf{n}}$.

This weak formulation allows the use of the Lagrangian gradients because the area integrals taken over the whole of the domain should "wash out" any differences between the deformed and undeformed gradients. The only difference to equation (3.22) will be the switching of the $\left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right)$ term to $\left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0\right)$. This has beneficial effects when the domain is discretized, as is discussed in Section (4.4.4).

3.2.4 Normal Force Equilibrium

We first multiply equation (3.17) by the normal-offset distance n,

$$\begin{aligned} \Big[\nabla \cdot \bar{\bar{\boldsymbol{\sigma}}} + \rho(\mathbf{g}-\mathbf{a})\Big] n &= \mathbf{0} \\ \nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}}\, n) - \bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{n}} + \rho n (\mathbf{g}-\mathbf{a}) &= \mathbf{0} \end{aligned} \tag{3.23}$$

where the relation $\nabla n = \hat{\mathbf{n}}$ was used. Combining $\nabla \cdot [\,\text{eq.}(3.23)\,] + [\,\text{eq.}(3.17)\,] \cdot \hat{\mathbf{n}}$ gives

$$\nabla \cdot \Big[\nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}}\, n) + \rho n (\mathbf{g}-\mathbf{a})\Big] - (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla) \cdot \hat{\mathbf{n}} + \rho(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{n}} = 0$$

which is then multiplied by the weighting function W and integrated over the element volume.

$$\begin{aligned} \iiint \Big\{ &\nabla \cdot \Big[\nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}}\, n)\, W\Big] - \Big[\nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}}\, n)\Big] \cdot \nabla W \\ &+ \nabla \cdot \Big[\rho n (\mathbf{g}-\mathbf{a})\Big] W - (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla) \cdot \hat{\mathbf{n}} W + \rho(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{n}} W \Big\} \mathrm{d}\mathcal{V} = 0 \end{aligned} \tag{3.24}$$

$$\begin{aligned} \iiint \Big\{ &\nabla \cdot \Big[\bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{n}} W\Big] - \Big[\nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}}\, n)\Big] \cdot \nabla W \\ &- \rho n (\mathbf{g}-\mathbf{a}) \cdot \nabla W - (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla) \cdot \hat{\mathbf{n}} W + \rho(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{n}} W \Big\} \mathrm{d}\mathcal{V} = 0 \end{aligned} \tag{3.25}$$

To get the second form (3.25), $\nabla \cdot (\bar{\bar{\boldsymbol{\sigma}}}\, n)$ in the first term in (3.24) was replaced by $\bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{n}} - \rho n (\mathbf{g}-\mathbf{a})$ using equation (3.23).

As before, the first pure divergence integral in (3.25) is replaced by area integrals over the perimeter and top/bottom surfaces, and the $\int \mathrm{d}n$ integrations are carried out over the volume and the edges. We will also assume that the moment from gravity and inertial reaction acting

on the shell's mass n-moment couple is negligible.

$$\int_{n_{\rm bot}}^{n_{\rm top}} \rho n \, \mathrm{d}n \simeq 0 \tag{3.26}$$

Equation (3.25) then becomes

$$\boxed{\oint \hat{\mathbf{n}} \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} \, W \, \mathrm{d}\ell + \iint \Big[-\big(\tilde{\nabla} \cdot \bar{\bar{\mathbf{m}}}\big) \cdot \tilde{\nabla} W - \big(\bar{\bar{\mathbf{f}}} \cdot \tilde{\nabla}\big) \cdot \hat{\mathbf{n}} W + \mathbf{q} \cdot \hat{\mathbf{n}} W + \mu (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{n}} W \Big] \, \mathrm{d}\mathcal{A} = 0} \tag{3.27}$$

which is the scalar normal-force equilibrium equation.

3.2.5 Moment Equilibrium

The integral shell moment-equilibrium equation is obtained by forming $[\,\text{eq.}(3.23)\,] \cdot \hat{\mathbf{w}}_1 W$ and $[\,\text{eq.}(3.23)\,] \cdot \hat{\mathbf{w}}_2 W$, and expanding the divergence term, and integrating over the element volume to give

$$\iiint \Big\{ \nabla \cdot (n \bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{w}} W) - (n \bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla) \cdot (\hat{\mathbf{w}} W) - \hat{\mathbf{w}} \cdot \bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{n}} W + \rho n (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{w}} W \Big\} \, \mathrm{d}\mathcal{V} \tag{3.28}$$

where again, $\hat{\mathbf{w}}$ denotes either in-plane weighting vector $\hat{\mathbf{w}}_1$ or $\hat{\mathbf{w}}_2$. Carrying out the $\int \mathrm{d}n$ integration gives the shell moment equilibrium equation,

$$\boxed{\oint \hat{\mathbf{w}} \cdot \bar{\bar{\mathbf{m}}} \cdot \hat{\mathbf{t}} \, W \, \mathrm{d}\ell + \iint \Big[-(\bar{\bar{\mathbf{m}}} \cdot \nabla) \cdot (\hat{\mathbf{w}} W) - \hat{\mathbf{w}} \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{n}} W \Big] \, \mathrm{d}\mathcal{A} = 0} \tag{3.29}$$

in which the mass moment and surface-stress moment terms have been dropped.

The normal force equailibrium equation (3.27) in effect governs $\bar{\bar{\mathbf{m}}}$, while the moment equilibrium equation (3.29) in effect then governs the two transverse shear force components f_{1n}, f_{2n} via the $\bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{n}} = f_{1n}\hat{\mathbf{e}}_1 + f_{2n}\hat{\mathbf{e}}_2$ term. These force components are explicitly needed only in HSM/M where they are used to define the $\varepsilon_{1n}, \varepsilon_{2n}$ transverse shear strains via (3.37), which in turn also define the small normal vector correction $\Delta\mathbf{n}$ via (3.8). In the simpler HSM/KL $\varepsilon_{1n}, \varepsilon_{2n}$ and $\Delta\mathbf{n}$ are all ignored, and hence f_{1n}, f_{2n} are not needed as primary unknowns and equation (3.29) is not used.

3.3 Constitutive Relations

HSM is the first geometrically exact shell model to allow modeling of anisotropic materials without additional complications. The following section derives the constitutive relations of HSM with an emphasis on maintaining anisotropic information throughout.

3.3.1 Stress Tensor

Given the assumption of a Hookean material, the stress tensor components are related to the strain tensor components, both defined in the local $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}$ basis, via the stiffness tensor, C_{ijkl}.

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{Bmatrix} = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1123} & C_{1113} & C_{1112} \\ & C_{2222} & C_{2233} & C_{2223} & C_{2213} & C_{2212} \\ & & C_{3333} & C_{3323} & C_{3313} & C_{3312} \\ & & & C_{2323} & C_{2313} & C_{2312} \\ & symm & & & C_{1313} & C_{1312} \\ & & & & & C_{1212} \end{bmatrix} \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{23} \\ \varepsilon_{13} \\ \varepsilon_{12} \end{Bmatrix} \tag{3.30}$$

Where the shear strains are defined as the *engineering* shear strains. In Voigt notation, pairs of indices are combined as follows: $()_{11} \rightarrow ()_1$, $()_{22} \rightarrow ()_2$, $()_{2n} \rightarrow ()_4$, $()_{1n} \rightarrow ()_5$, $()_{12} \rightarrow ()_6$ For specificity of notation, the stiffness tensor C_{ijkl} will be referred to as c_{ij} when using Voigt notation. The 3 component of the stresses and strains has also changed to n to denote its surface-normal direction.

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{nn} \\ \sigma_{2n} \\ \sigma_{1n} \\ \sigma_{12} \end{Bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} & c_{16} \\ & c_{22} & c_{23} & c_{24} & c_{25} & c_{26} \\ & & c_{33} & c_{34} & c_{35} & c_{36} \\ & & & c_{44} & c_{45} & c_{46} \\ & symm & & & c_{55} & c_{56} \\ & & & & & c_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{nn} \\ \varepsilon_{2n} \\ \varepsilon_{1n} \\ \varepsilon_{12} \end{Bmatrix} \tag{3.31}$$

This general constitutive model is considerably simplified using the shell assumptions that the normal stress (σ_{nn}) and strain (ε_{nn}) are negligible, and that the in-plane stress tensor components $\sigma_{11}, \sigma_{22}, \sigma_{12}$ are decoupled from the out-of-plane stress tensor components

σ_{1n}, σ_{2n}. The stress-strain relations thus reduce to the following.

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{Bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon'_{11} \\ \varepsilon'_{22} \\ \varepsilon'_{12} \end{Bmatrix} \tag{3.32}$$

$$\begin{Bmatrix} \sigma_{1n} \\ \sigma_{2n} \end{Bmatrix} = K \begin{bmatrix} c_{55} & 0 \\ 0 & c_{44} \end{bmatrix} \begin{Bmatrix} \varepsilon'_{1n} \\ \varepsilon'_{2n} \end{Bmatrix} \tag{3.33}$$

in which the remaining normal stress σ_{nn} is assumed to be negligible, and the prime on ε' indicates a strain at a general distance n off the reference surface. Following standard shell-theory approximations we assume that the transverse material lines remain straight, so that we can write

$$\begin{Bmatrix} \varepsilon'_{11} \\ \varepsilon'_{22} \\ \varepsilon'_{12} \end{Bmatrix} = \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \end{Bmatrix} + n \begin{Bmatrix} \kappa_{11} \\ \kappa_{22} \\ \kappa_{12} \end{Bmatrix} \tag{3.34}$$

where κ are components of the curvatures-change tensor. The out-of-plane shear strains $\varepsilon'_{1n}, \varepsilon'_{2n}$ will have some more complicated n-dependence, since they must fall to zero at the top and bottom shell surface. Again following shell theory we will assume that they can be represented in a modulus-weighted average sense by the reference-surface strains $\varepsilon_{1n}, \varepsilon_{2n}$,

$$\begin{Bmatrix} \varepsilon'_{1n} \\ \varepsilon'_{2n} \end{Bmatrix} = K \begin{Bmatrix} \varepsilon_{1n} \\ \varepsilon_{2n} \end{Bmatrix} \tag{3.35}$$

where K is the shear strain energy reduction factor. The commonly-chosen value $K = 5/6$ corresponds to parabolic $\varepsilon'_{1n}(n)$ and $\varepsilon'_{2n}(n)$ across the shell thickness, which is the correct result for a uniform isotropic shell.

3.3.2 Force and Moment Resultants, Shell Lumping

We now substitute the strain components in equation (3.34) into the stress/strain relations (3.32, 3.33), and then insert that into the stress and stress-moment resultant definitions (3.19). This gives the following symmetric linear system which relates the components of the stress and stress-moment resultants, with the components of the strain and curvature-

change tensors of the reference surface.

$$\left\{\begin{array}{c} f_{11} \\ f_{22} \\ f_{12} \\ \hline m_{11} \\ m_{22} \\ m_{12} \end{array}\right\} = \left[\begin{array}{ccc|ccc} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ \cdot & A_{22} & A_{26} & \cdot & B_{22} & B_{26} \\ \cdot & \cdot & A_{66} & \cdot & \cdot & B_{66} \\ \hline B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ \cdot & B_{22} & B_{26} & \cdot & D_{22} & D_{26} \\ \cdot & \cdot & B_{66} & \cdot & \cdot & D_{66} \end{array}\right] \left\{\begin{array}{c} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \\ \hline \kappa_{11} \\ \kappa_{22} \\ \kappa_{12} \end{array}\right\} \tag{3.36}$$

$$\left\{\begin{array}{c} f_{1n} \\ f_{2n} \end{array}\right\} = \left[\begin{array}{cc} A_{55} & 0 \\ 0 & A_{44} \end{array}\right] \left\{\begin{array}{c} \varepsilon_{1n} \\ \varepsilon_{2n} \end{array}\right\} \tag{3.37}$$

The stiffness submatrices above are defined by the following weighted integrals over the shell thickness, which capture the overall lumped properties of the shell cross section.

$$\bar{\bar{A}} = \left[\begin{array}{ccc} A_{11} & A_{12} & A_{16} \\ \cdot & A_{22} & A_{26} \\ \cdot & \cdot & A_{66} \end{array}\right] \equiv \int \left[\begin{array}{ccc} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{array}\right] \mathrm{d}n \tag{3.38}$$

$$\bar{\bar{B}} = \left[\begin{array}{ccc} B_{11} & B_{12} & B_{16} \\ \cdot & B_{22} & B_{26} \\ \cdot & \cdot & B_{66} \end{array}\right] \equiv \int \left[\begin{array}{ccc} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{array}\right] n \ \mathrm{d}n \tag{3.39}$$

$$\bar{\bar{D}} = \left[\begin{array}{ccc} D_{11} & D_{12} & D_{16} \\ \cdot & D_{22} & D_{26} \\ \cdot & \cdot & D_{66} \end{array}\right] \equiv \int \left[\begin{array}{ccc} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{array}\right] n^2 \ \mathrm{d}n \tag{3.40}$$

$$\bar{\bar{A}}' = \left[\begin{array}{cc} A_{55} & 0 \\ 0 & A_{44} \end{array}\right] \equiv \int K \left[\begin{array}{cc} c_{55} & 0 \\ 0 & c_{44} \end{array}\right] \mathrm{d}n \quad , \qquad K \simeq 5/6 \tag{3.41}$$

In shell theory it is traditional to choose the reference surface to lie in the middle of the shell thickness, so that for a homogeneous shell material the $\bar{\bar{B}}$ matrix elements in equation (3.39) are all zero, and the f and m components in the stiffness matrix equation (3.36) then decouple. Here no such assumption is made, to allow complete freedom in the choice of the reference surface location and of the shell composition.

3.3.3 Stress and Moment Basis Conversions

The stress and stress-moment, strain, and curvature-change resultants which are defined in the local 12n axes must be put in the global xyz axes for application in the equilibrium

equations. This is performed using the xyz components of the local $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}$ basis vectors.

$$\begin{bmatrix} f_{xx} & f_{xy} & f_{xz} \\ \cdot & f_{yy} & f_{yz} \\ \cdot & \cdot & f_{zz} \end{bmatrix} = \begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{1n} \\ \cdot & f_{22} & f_{2n} \\ \cdot & \cdot & 0 \end{bmatrix} \begin{bmatrix} - & \hat{\mathbf{e}}_1 & - \\ - & \hat{\mathbf{e}}_2 & - \\ - & \hat{\mathbf{n}} & - \end{bmatrix} \tag{3.42}$$

$$\begin{bmatrix} m_{xx} & m_{xy} & m_{xz} \\ \cdot & m_{yy} & m_{yz} \\ \cdot & \cdot & m_{zz} \end{bmatrix} = \begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & 0 \\ \cdot & m_{22} & 0 \\ \cdot & \cdot & 0 \end{bmatrix} \begin{bmatrix} - & \hat{\mathbf{e}}_1 & - \\ - & \hat{\mathbf{e}}_2 & - \\ - & \hat{\mathbf{n}} & - \end{bmatrix} \tag{3.43}$$

3.3.4 Strain and Curvature Basis Conversions

The strain and curvature-change resultants $\bar{\bar{\varepsilon}}^{\mathsf{e}}$ and $\bar{\bar{\kappa}}^{\mathsf{e}}$ which are defined in the local 12n axes must also be put in the global xyz axes for application in the equilibrium equations. In an effort to limit operations on primary variables used to construct terms in the governing equations to rotations only, the locally defined stress, stress-moment, and stiffness tensors appearing in equations (3.36) and (3.37) must first be rotated to the non-degenerate reference frame. In this way, they are defined everywhere in the local neighborhood of a reference point, and can then be converted back to the cartesian frame for application of the governing equations at any point in the neighborhood. Conversion from the e-frame to the w-frame implicitly involves firstly rotating to the cartesian frame and secondly rotating to the w-frame, since both coordinate bases are defined in terms of their cartesian components. A transformation of this kind takes the form $\bar{\bar{X}}^{\mathsf{w}} = (\bar{\bar{\mathbf{w}}}^T \cdot \bar{\bar{\mathbf{e}}}) \cdot \bar{\bar{X}}^{\mathsf{e}} \cdot (\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{w}}})$. As the product $\bar{\bar{\mathbf{w}}}^T \cdot \bar{\bar{\mathbf{e}}}$ is used frequently, it will be henceforth referred to as $\bar{\bar{s}}$.

$$\bar{\bar{s}} = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} := \begin{bmatrix} - & \hat{\mathbf{w}}_1 & - \\ - & \hat{\mathbf{w}}_2 & - \\ - & \hat{\mathbf{n}} & - \end{bmatrix} \begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} = \left[\begin{array}{c|c} \tilde{\tilde{s}} & 0 \\ \hline 0 & 1 \end{array}\right] \tag{3.44}$$

The off-diagonal blocks in the final $\tilde{\tilde{s}}$ form in (3.44) are zero because $\bar{\bar{\mathbf{e}}}$ and $\bar{\bar{\mathbf{w}}}$ have the same $\hat{\mathbf{n}}$ vector. Explicitly, $\tilde{\tilde{s}}$ is defined as

$$\tilde{\tilde{s}} \equiv \begin{bmatrix} \hat{\mathbf{w}}_1 \cdot \hat{\mathbf{e}}_1 & \hat{\mathbf{w}}_1 \cdot \hat{\mathbf{e}}_2 \\ \hat{\mathbf{w}}_2 \cdot \hat{\mathbf{e}}_1 & \hat{\mathbf{w}}_2 \cdot \hat{\mathbf{e}}_2 \end{bmatrix} \tag{3.45}$$

The strain and strain-curvatures on the right-hand side of equations (3.36) and (3.37) are put into the non-degenerate frame using the $\bar{\bar{s}}$ matrix.

$$\begin{bmatrix} f_{11}^{\mathsf{w}} & f_{12}^{\mathsf{w}} & f_{1n}^{\mathsf{w}} \\ \cdot & f_{22}^{\mathsf{w}} & f_{2n}^{\mathsf{w}} \\ \cdot & \cdot & 0 \end{bmatrix} = \left[\begin{array}{c|c} \approx{s} & 0 \\ \hline 0 & 1 \end{array}\right] \begin{bmatrix} f_{11}^{\mathsf{e}} & f_{12}^{\mathsf{e}} & f_{1n}^{\mathsf{e}} \\ \cdot & f_{22}^{\mathsf{e}} & f_{2n}^{\mathsf{e}} \\ \cdot & \cdot & 0 \end{bmatrix} \left[\begin{array}{c|c} \approx{s}^T & 0 \\ \hline 0 & 1 \end{array}\right] \tag{3.46}$$

$$\begin{bmatrix} m_{11}^{\mathsf{w}} & m_{12}^{\mathsf{w}} \\ \cdot & m_{22}^{\mathsf{w}} \end{bmatrix} = \begin{bmatrix} \\ \approx{s} \\ \\ \end{bmatrix} \begin{bmatrix} m_{11}^{\mathsf{e}} & m_{12}^{\mathsf{e}} \\ \cdot & m_{22}^{\mathsf{e}} \end{bmatrix} \begin{bmatrix} \\ \approx{s}^T \\ \\ \end{bmatrix} \tag{3.47}$$

In some situations, it will be useful to separate the normal shear stress components from the in-plane stress components. In these cases, we adopt the following nomenclature similar to the decomposition of the $\bar{\bar{s}}$ matrix.

$$\bar{\bar{f}} \equiv \begin{bmatrix} f_{11} & f_{12} & f_{1n} \\ \cdot & f_{22} & f_{2n} \\ \cdot & \cdot & 0 \end{bmatrix} = \left[\begin{array}{c|c} \approx{f} & \vec{f}_{\alpha n} \\ \hline \cdot \quad \cdot & 0 \end{array}\right] \tag{3.48}$$

The stiffness tensor C_{ijkl} of equation (3.30) is actually C_{ijkl}^{e}, defined in the local $12n$ coordinates and must also be converted to the w-frame. This is accomplished with the 4th-order tensor basis transformation.

$$C_{ijkl}^{\mathsf{w}} = s_{ip}\, s_{jq}\, s_{kr}\, s_{ls}\, C_{pqrs}^{\mathsf{e}} \tag{3.49}$$

With the C_{ijkl}^{w} now defined, the lumped stiffness submatrices $\bar{\bar{A}}^{\mathsf{w}}$, $\bar{\bar{B}}^{\mathsf{w}}$, $\bar{\bar{D}}^{\mathsf{w}}$ can be calculated using equations (3.38) through (3.41). The procedure for calculating the force-dependent strain and curvature tensors in the local neighborhood of a reference point, $\bar{\bar{\varepsilon}}^{\mathsf{w}(\mathsf{f})}, \bar{\bar{\kappa}}^{\mathsf{w}(\mathsf{f})}$, makes use of the locally defined stress, stress-moment, and stiffness tensors. Using the inverse of the constitutive equations (3.36) and (3.37).

$$\bar{\bar{\varepsilon}}^{\mathsf{w}(\mathsf{f})}, \bar{\bar{\kappa}}^{\mathsf{w}(\mathsf{f})} = \begin{Bmatrix} \varepsilon_{11}^{\mathsf{w}} \\ \varepsilon_{22}^{\mathsf{w}} \\ \varepsilon_{12}^{\mathsf{w}} \\ \hline \kappa_{11}^{\mathsf{w}} \\ \kappa_{22}^{\mathsf{w}} \\ \kappa_{12}^{\mathsf{w}} \end{Bmatrix}_j = \left[\begin{array}{c|c} \bar{\bar{A}}^{\mathsf{w}} & \bar{\bar{B}}^{\mathsf{w}} \\ \hline \bar{\bar{B}}^{\mathsf{w}} & \bar{\bar{D}}^{\mathsf{w}} \end{array}\right]_j^{-1} \begin{Bmatrix} f_{11}^{\mathsf{w}} \\ f_{22}^{\mathsf{w}} \\ f_{12}^{\mathsf{w}} \\ \hline m_{11}^{\mathsf{w}} \\ m_{22}^{\mathsf{w}} \\ m_{12}^{\mathsf{w}} \end{Bmatrix}_j \tag{3.50}$$

$$\begin{Bmatrix} \varepsilon_{1n}^{\mathsf{w}} \\ \varepsilon_{2n}^{\mathsf{w}} \end{Bmatrix}_j = \begin{bmatrix} \\ \bar{\bar{A}}'^{\mathsf{w}} \\ \\ \end{bmatrix}_j^{-1} \begin{Bmatrix} f_{1n}^{\mathsf{w}} \\ f_{2n}^{\mathsf{w}} \end{Bmatrix}_j \tag{3.51}$$

Finally, the w-frame strains and strain-curvatures can be converted back into cartesian co-

ordinates anywhere in the local neighborhood for inclusion in the equilibrium equations.

$$\bar{\bar{\varepsilon}}(\mathbf{f}(\xi^1,\xi^2)) \equiv \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} & \varepsilon_{xz} \\ \cdot & \varepsilon_{yy} & \varepsilon_{yz} \\ \cdot & \cdot & \varepsilon_{zz} \end{bmatrix} = \begin{bmatrix} | & | & | \\ \hat{\mathbf{w}}_1 & \hat{\mathbf{w}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} \begin{bmatrix} \varepsilon^{\mathsf{w}}_{11} & \varepsilon^{\mathsf{w}}_{12} & \varepsilon^{\mathsf{w}}_{1n} \\ \cdot & \varepsilon^{\mathsf{w}}_{22} & \varepsilon^{\mathsf{w}}_{2n} \\ \cdot & \cdot & 0 \end{bmatrix} \begin{bmatrix} - & \hat{\mathbf{w}}_1 & - \\ - & \hat{\mathbf{w}}_2 & - \\ - & \hat{\mathbf{n}} & - \end{bmatrix} \tag{3.52}$$

$$\bar{\bar{\kappa}}(\mathbf{f}(\xi^1,\xi^2)) \equiv \begin{bmatrix} \kappa_{xx} & \kappa_{xy} & \kappa_{xz} \\ \cdot & \kappa_{yy} & \kappa_{yz} \\ \cdot & \cdot & \kappa_{zz} \end{bmatrix} = \begin{bmatrix} | & | & | \\ \hat{\mathbf{w}}_1 & \hat{\mathbf{w}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} \begin{bmatrix} \kappa^{\mathsf{w}}_{11} & \kappa^{\mathsf{w}}_{12} & 0 \\ \cdot & \kappa^{\mathsf{w}}_{22} & 0 \\ \cdot & \cdot & 0 \end{bmatrix} \begin{bmatrix} - & \hat{\mathbf{w}}_1 & - \\ - & \hat{\mathbf{w}}_2 & - \\ - & \hat{\mathbf{n}} & - \end{bmatrix} \tag{3.53}$$

3.4 Compatibility Relations

For points $\mathbf{r}', \mathbf{r}'_0$ which are a distance n from the reference surfaces $\mathbf{r}, \mathbf{r}_0$, along the material-normal vectors $\widetilde{\mathbf{n}}$ and $\widetilde{\mathbf{n}}_0$ we have

$$\mathbf{r}'_{(\xi,\eta,n)} = \mathbf{r} + n\,\widetilde{\mathbf{n}} = \mathbf{r} + n\,(\hat{\mathbf{n}} + \Delta\mathbf{n}) \tag{3.54}$$

$$\mathbf{r}'_{0(\xi,\eta,n)} = \mathbf{r}_0 + n\,\widetilde{\mathbf{n}}_0 = \mathbf{r}_0 + n\,\hat{\mathbf{n}}_0 \tag{3.55}$$

$$\widetilde{\nabla}\mathbf{r}' = \widetilde{\nabla}\mathbf{r} + n\,(\widetilde{\nabla}\hat{\mathbf{n}} + \widetilde{\nabla}\Delta\mathbf{n}) \tag{3.56}$$

$$\widetilde{\nabla}\mathbf{r}'_0 = \widetilde{\nabla}\mathbf{r}_0 + n\,\widetilde{\nabla}\hat{\mathbf{n}}_0 \tag{3.57}$$

where we note that $\Delta\mathbf{n}_0 = \mathbf{0}$ by definition and $\Delta\mathbf{n}$ is defined as follows. In an effort to limit operations on primary variables used in the governing equations to rotations only, the calculation of $\Delta\mathbf{n}$ occurs in the local w-frame. The tilt angles γ_1, γ_2 in the definition of $\Delta\mathbf{n}$ (equation (3.8) and visible in Figure 3-1) come from the shear deformations $\varepsilon^{\mathsf{w}}_{1n}, \varepsilon^{\mathsf{w}}_{2n}$ that were calculated in equation (3.51). Using the $\bar{\bar{\mathbf{w}}}$ basis, we formulate equation (3.8) in the local non-degenerate coordinates.

$$\Delta\mathbf{n} = 2 \begin{bmatrix} | & | & | \\ \hat{\mathbf{w}}_1 & \hat{\mathbf{w}}_2 & \hat{\mathbf{n}} \\ | & | & | \end{bmatrix} \begin{Bmatrix} \varepsilon^{\mathsf{w}}_{1n} \\ \varepsilon^{\mathsf{w}}_{2n} \\ 0 \end{Bmatrix} \tag{3.58}$$

To evaluate $\widetilde{\nabla}\Delta\mathbf{n}$ in equation (3.56), the in-plane parametric derivatives $\partial_\alpha\Delta\mathbf{n}$ are needed, which in turn requires $\partial_\alpha\hat{\mathbf{n}}$, $\partial_\alpha\hat{\mathbf{w}}_i$, and $\partial_\alpha\varepsilon^{\mathsf{w}}_{\beta n}$. Calculation of $\partial_\alpha\hat{\mathbf{n}}$ is covered in Section 4.2.2 and calculation of $\partial_\alpha\hat{\mathbf{w}}_i$ uses equations (4.51)–(4.57). Calculation of the normal strain partial derivatives is accomplished using the explicit expansion of equation (3.51).

$$\varepsilon^{\mathsf{w}}_{1n} = f^{\mathsf{w}}_{1n} \,/\, A_{55} \tag{3.59}$$

$$\varepsilon^{\mathsf{w}}_{2n} = f^{\mathsf{w}}_{2n} \,/\, A_{44} \tag{3.60}$$

The $\partial_\alpha \varepsilon^{\mathbf{w}}_{\beta n}$ derivatives are obtained by application of the chain rule to equations (3.59) and (3.60). The various surface gradients (at fixed n), Lagrangian (w.r.t undeformed geometry) and Eulerian (w.r.t. deformed geometry), e.g.

$$\widetilde{\nabla}_0 \mathbf{r} \equiv \begin{bmatrix} - (\widetilde{\nabla}_0 \mathbf{r})_x - \\ - (\widetilde{\nabla}_0 \mathbf{r})_y - \\ - (\widetilde{\nabla}_0 \mathbf{r})_z - \end{bmatrix} \qquad \text{(Lagrangian)} \quad (3.61)$$

$$\widetilde{\nabla} \mathbf{r} \equiv \begin{bmatrix} - (\widetilde{\nabla} \mathbf{r})_x - \\ - (\widetilde{\nabla} \mathbf{r})_y - \\ - (\widetilde{\nabla} \mathbf{r})_z - \end{bmatrix} = \begin{bmatrix} - \partial_x \mathbf{r}\,|_n - \\ - \partial_y \mathbf{r}\,|_n - \\ - \partial_z \mathbf{r}\,|_n - \end{bmatrix} \qquad \text{(Eulerian)} \quad (3.62)$$

are obtained from $\mathbf{r}_{(\xi,\eta)}$ and $\hat{\mathbf{n}}_{(\xi,\eta)}$ via solutions of the following 3×3 linear systems.

$$\begin{bmatrix} - \partial_\xi \mathbf{r}_0 - \\ - \partial_\eta \mathbf{r}_0 - \\ - \hat{\mathbf{n}}_0 - \end{bmatrix} \begin{bmatrix} \\ \widetilde{\nabla}_0 \mathbf{r} \\ \\ \end{bmatrix} = \begin{bmatrix} - \partial_\xi \mathbf{r} - \\ - \partial_\eta \mathbf{r} - \\ - \mathbf{0} - \end{bmatrix} \qquad \text{(Lagrangian)} \quad (3.63)$$

$$\begin{bmatrix} - \partial_\xi \mathbf{r}_0 - \\ - \partial_\eta \mathbf{r}_0 - \\ - \hat{\mathbf{n}}_0 - \end{bmatrix} \begin{bmatrix} \\ \widetilde{\nabla}_0 \hat{\mathbf{n}} \\ \\ \end{bmatrix} = \begin{bmatrix} - \partial_\xi \hat{\mathbf{n}} - \\ - \partial_\eta \hat{\mathbf{n}} - \\ - \mathbf{0} - \end{bmatrix} \qquad \text{(Lagrangian)} \quad (3.64)$$

$$\begin{bmatrix} - \partial_\xi \mathbf{r} - \\ - \partial_\eta \mathbf{r} - \\ - \hat{\mathbf{n}} - \end{bmatrix} \begin{bmatrix} \\ \widetilde{\nabla} \mathbf{r} \\ \\ \end{bmatrix} = \begin{bmatrix} - \partial_\xi \mathbf{r} - \\ - \partial_\eta \mathbf{r} - \\ - \mathbf{0} - \end{bmatrix} \qquad \text{(Eulerian)} \quad (3.65)$$

$$\begin{bmatrix} - \partial_\xi \mathbf{r} - \\ - \partial_\eta \mathbf{r} - \\ - \mathbf{0} - \end{bmatrix} \begin{bmatrix} \\ \widetilde{\nabla} \hat{\mathbf{n}} \\ \\ \end{bmatrix} = \begin{bmatrix} - \partial_\xi \hat{\mathbf{n}} - \\ - \partial_\eta \hat{\mathbf{n}} - \\ - \mathbf{0} - \end{bmatrix} \qquad \text{(Eulerian)} \quad (3.66)$$

Corresponding systems are used to obtain $\widetilde{\nabla} \Delta \mathbf{n}$, $\widetilde{\nabla} \mathbf{r}_0$, and $\widetilde{\nabla} \hat{\mathbf{n}}_0$ from the parametric derivatives of $\Delta \mathbf{n}_{(\xi,\eta)}$, $\mathbf{r}_{0(\xi,\eta)}$, and $\hat{\mathbf{n}}_{0(\xi,\eta)}$ on the right-hand sides.

The Green strain off the reference surface is

$$\bar{\bar{\boldsymbol{\varepsilon}}}' = \frac{1}{2}\Big[(\widetilde{\nabla} \mathbf{r}')(\widetilde{\nabla} \mathbf{r}')^T - (\widetilde{\nabla} \mathbf{r}'_0)(\widetilde{\nabla} \mathbf{r}'_0)^T\Big] \quad (3.67)$$

$$\simeq \bar{\bar{\boldsymbol{\varepsilon}}} + n \bar{\bar{\boldsymbol{\kappa}}} \quad (3.68)$$

where the strain and curvature-change tensors of the reference surface are defined as follows.

$$\bar{\bar{\boldsymbol{\varepsilon}}}_{(\mathbf{r})} = \frac{1}{2}\Big[(\widetilde{\nabla} \mathbf{r})(\widetilde{\nabla} \mathbf{r})^T - (\widetilde{\nabla} \mathbf{r}_0)(\widetilde{\nabla} \mathbf{r}_0)^T\Big] \quad (3.69)$$

$$\bar{\bar{\boldsymbol{\kappa}}}_{(\mathbf{r})} = \Big[(\widetilde{\nabla} \mathbf{r})(\widetilde{\nabla} \hat{\mathbf{n}} + \widetilde{\nabla} \Delta \mathbf{n})^T - (\widetilde{\nabla} \mathbf{r}_0)(\widetilde{\nabla} \hat{\mathbf{n}}_0)^T\Big] \quad (3.70)$$

Note that this definition of strain uses the Eulerian formulation as the surface gradients are with respect to the deformed geometry. The Almansi strain replaces the Eulerian gradients $\widetilde{\nabla}$ in equations (3.67-3.70) with Lagrangian gradients $\widetilde{\nabla}_0$ (w.r.t. undeformed geometry). In general, the Green strain is more robust for large positive strains, and the Almansi strain is more robust for large negative strains.

In the linearized approximation (3.68) we have dropped the quadratic n^2 term, which is appropriate for the usual case where the shell thickness is much smaller than the shell's radius of curvature. The notation $\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{r}), \bar{\bar{\boldsymbol{\kappa}}}(\mathbf{r})$ indicates that these strains are functions of the shell geometry.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 4

HSM Finite-Element Solution

4.1 Nodal Data

Each element corner node j has the data listed below. Bilinear interpolation to the element interior then makes these quantities functions of the (ξ, η) element coordinates as described earlier. The vector and tensor quantities are defined either in the global xyz axes, or in the node's $12n$ axes along the node's $\bar{\bar{\mathbf{e}}}_j = [\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}]_j$ basis vectors.

4.1.1 Parameters

These input quantities describe the shell geometry, structural properties, mass, and loading. They are all defined at each node j, and used either at the nodes to compute secondary variables, or interpolated to the element interior to construct the equation residuals.

symbol	N	axes	description
$\mathbf{r}_{0_j}$	3	xyz	position vector of undeformed geometry
$\mathbf{\Lambda}_{0_j}$	3	—	rotation log-quaternion of undeformed geometry, from $[\hat{\mathbf{e}}_{0_1}, \hat{\mathbf{e}}_{0_2}, \hat{\mathbf{n}}_0]_j$
$\bar{\bar{A}}^{\mathsf{e}}_j$	6	e	lumped shell stiffness matrix (extension and shear stiffness)
$\bar{\bar{B}}^{\mathsf{e}}_j$	6	e	lumped shell stiffness matrix (extension/bending coupling)
$\bar{\bar{D}}^{\mathsf{e}}_j$	6	e	lumped shell stiffness matrix (bending stiffness)
μ_j	1	—	lumped shell mass (mass/area density)
$\vec{q}^{\mathsf{e}}_j$	3	e	shell-following applied force/area
$\mathbf{q}_{xyz_j}$	3	xyz	fixed-direction applied force/area
$\mathbf{a}_j$	3	xyz	local acceleration
$\mathbf{g}$	3	xyz	gravity

In practice, the undeformed geometry is defined by the parametric surface $\mathbf{r}_{0(u,v)}$, where u, v are the (e.g. B-spline) surface coordinates. Its nodal basis vectors can then be conveniently

computed by evaluating

$$\hat{\mathbf{e}}_{0_1} = \frac{(1-t)\,\partial_u \mathbf{r}_0 + t\,\partial_v \mathbf{r}_0}{|\,(1-t)\,\partial_u \mathbf{r}_0 + t\,\partial_v \mathbf{r}_0\,|} \tag{4.1}$$

$$\hat{\mathbf{n}}_0 = \pm \frac{\partial_u \mathbf{r}_0 \times \partial_v \mathbf{r}_0}{|\partial_u \mathbf{r}_0 \times \partial_v \mathbf{r}_0|} \tag{4.2}$$

$$\hat{\mathbf{e}}_{0_2} = \hat{\mathbf{n}}_0 \times \hat{\mathbf{e}}_{0_1} \tag{4.3}$$

at each node, in which the constant t selects the azimuthal orientation of the $\hat{\mathbf{e}}_{0_1}, \hat{\mathbf{e}}_{0_2}$ vectors within the surface; choosing $t = 0$ aligns $\hat{\mathbf{e}}_{0_1}$ with $\partial_u \mathbf{r}_0$, and choosing $t = 1$ aligns $\hat{\mathbf{e}}_{0_1}$ with $\partial_v \mathbf{r}_0$. The positive-n side of the surface is selected by choosing the $\pm$ sign in equation (4.2). The $\mathbf{\Lambda}_{0_j}$ which corresponds to these nodal basis vectors is obtained by first inverting the definition (3.5). Using the more convenient index notation

$$\begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_{0_1} & \hat{\mathbf{e}}_{0_2} & \hat{\mathbf{n}}_0 \\ | & | & | \end{bmatrix}_j = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix} \tag{4.4}$$

we first compute candidates for the largest quaternion component from the diagonal elements,

$$\tilde{p}_0 = \tfrac{1}{2}\sqrt{\max(\,1+e_{11}+e_{22}+e_{33}\,,\,0\,)} \tag{4.5}$$

$$\tilde{p}_1 = \tfrac{1}{2}\sqrt{\max(\,1+e_{11}-e_{22}-e_{33}\,,\,0\,)} \tag{4.6}$$

$$\tilde{p}_2 = \tfrac{1}{2}\sqrt{\max(\,1-e_{11}+e_{22}-e_{33}\,,\,0\,)} \tag{4.7}$$

$$\tilde{p}_3 = \tfrac{1}{2}\sqrt{\max(\,1-e_{11}-e_{22}+e_{33}\,,\,0\,)} \tag{4.8}$$

and then the remaining components are computed so as to minimize roundoff error.

$$\text{if } \tilde{p}_0 \geq \max(\tilde{p}_1, \tilde{p}_2, \tilde{p}_3) \;:\; \begin{cases} p_0 = \tilde{p}_0 \\ p_1 = \frac{1}{4}(e_{32} - e_{23})/\tilde{p}_0 \\ p_2 = \frac{1}{4}(e_{13} - e_{31})/\tilde{p}_0 \\ p_3 = \frac{1}{4}(e_{21} - e_{12})/\tilde{p}_0 \end{cases} \tag{4.9}$$

$$\text{if } \tilde{p}_1 \geq \max(\tilde{p}_0, \tilde{p}_2, \tilde{p}_3) \;:\; \begin{cases} p_0 = \frac{1}{4}(e_{32} - e_{23})/\tilde{p}_1 \\ p_1 = \tilde{p}_1 \\ p_2 = \frac{1}{4}(e_{21} + e_{12})/\tilde{p}_1 \\ p_3 = \frac{1}{4}(e_{13} + e_{31})/\tilde{p}_1 \end{cases} \tag{4.10}$$

$$\text{if } \tilde{p}_2 \geq \max(\tilde{p}_0, \tilde{p}_1, \tilde{p}_3) \;:\; \begin{cases} p_0 = \frac{1}{4}(e_{13} - e_{31})/\tilde{p}_2 \\ p_1 = \frac{1}{4}(e_{21} + e_{12})/\tilde{p}_2 \\ p_2 = \tilde{p}_2 \\ p_3 = \frac{1}{4}(e_{32} + e_{23})/\tilde{p}_2 \end{cases} \tag{4.11}$$

$$\text{if } \tilde{p}_3 \geq \max(\tilde{p}_0, \tilde{p}_1, \tilde{p}_2) \;:\; \begin{cases} p_0 = \frac{1}{4}(e_{21} - e_{12})/\tilde{p}_3 \\ p_1 = \frac{1}{4}(e_{13} + e_{31})/\tilde{p}_3 \\ p_2 = \frac{1}{4}(e_{32} + e_{23})/\tilde{p}_3 \\ p_3 = \tilde{p}_3 \end{cases} \tag{4.12}$$

Finally, the log-quaternion is obtained by inverting relations (3.3)–(3.4).

$$\mathbf{\Lambda}_{0_j} = \begin{Bmatrix} 0 \\ p_1 \\ p_2 \\ p_3 \end{Bmatrix} \frac{1}{\sqrt{p_1^2 + p_2^2 + p_3^2}} \operatorname{atan2}\left(\sqrt{p_1^2 + p_2^2 + p_3^2} \;,\; p_0 \right) \tag{4.13}$$

4.1.2 Unknowns (Primary Variables)

These are the primary variables which are to be determined at each node j. In HSM/M $\bar{\bar{f}}_j$ has 5 variables, while for HSM/KL its f_{1n}, f_{2n} components are dropped leaving only 3 variables.

symbol	N	axes	description
$\mathbf{r}_j$	3	xyz	position vector of deformed geometry
$\mathbf{\Lambda}_j$	3	—	rotation log-quaternion of deformed geometry
$\bar{\bar{f}}_j^{\mathsf{e}}$	3 or 5	e	stress resultant tensor components
$\bar{\bar{m}}_j^{\mathsf{e}}$	3	e	stress-moment resultant tensor components

4.1.3 Nodal Dependents (Secondary Variables)

These are the secondary variables that are calculated at the nodes. They are simply rotations or summations of primary variables and the parameters.

symbol	N	var. dep.	par. dep.	axes	description
$\bar{\bar{\mathbf{e}}}_j$	9	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	basis vectors $[\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}]_j$ of deformed geom.
$\bar{\bar{\mathbf{w}}}_j$	9	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	basis vectors $[\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, \hat{\mathbf{n}}]_j$ of deformed geom.
$\bar{\bar{\mathbf{f}}}_j$	6	$\bar{\bar{f}}^{\mathsf{e}}_j$, $\mathbf{\Lambda}_j$		xyz	stress resultants
$\bar{\bar{\mathbf{m}}}_j$	6	$\bar{\bar{m}}^{\mathsf{e}}_j$, $\mathbf{\Lambda}_j$		xyz	stress-moment resultants
$\bar{\bar{f}}^{\mathsf{w}}_j$	6	$\bar{\bar{f}}^{\mathsf{e}}_j$, $\mathbf{\Lambda}_j$		w	stress resultants
$\bar{\bar{m}}^{\mathsf{w}}_j$	6	$\bar{\bar{m}}^{\mathsf{e}}_j$, $\mathbf{\Lambda}_j$		w	stress-moment resultants
$\mathbf{q}_j$	3	$\mathbf{\Lambda}_j$	$\mathbf{q}_{xyz_j}, \vec{q}_j$	xyz	total applied load

The cartesian xyz components of the nodal $\bar{\bar{\mathbf{f}}}_j, \bar{\bar{\mathbf{m}}}_j$ tensors are computed directly from $\bar{\bar{f}}_j, \bar{\bar{m}}_j$ using equations (3.42) and (3.43) applied at each node. Similarly, the w-frame $12n$ components of the nodal $\bar{\bar{\mathbf{f}}}_j, \bar{\bar{\mathbf{m}}}_j$ tensors are computed directly from $\bar{\bar{f}}_j, \bar{\bar{m}}_j$ using equations (3.46) and (3.47) applied at each node. This introduces a $\mathbf{\Lambda}_j$ dependence via the $\bar{\bar{\mathbf{e}}}_j$ and $\bar{\bar{s}}_j$ matrices, respectively, which perform these basis changes.

The overall applied force $\mathbf{q}_j$ is the sum of $\mathbf{q}_{xyz_j}$ and $\vec{q}_j$ in the xyz axes

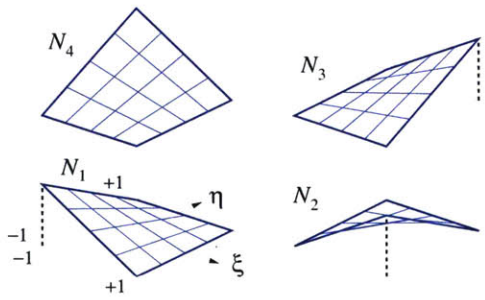
$$\mathbf{q}_j \doteq \mathbf{q}_{xyz_j} + \bar{\bar{\mathbf{e}}}_j \, \vec{q}_j \tag{4.14}$$

so that $\mathbf{q}_{xyz_j}$ is a *fixed-direction load*, while $\vec{q}_j$ is a *shell-following load*.

4.2 Element Interpolation

4.2.1 Bilinear Interpolation

Per standard practice, the nodal quantities are interpolated over the element via four bilinear basis functions $N_{j(\xi,\eta)}$, using the canonical coordinates $-1 \le \xi \le +1$, $-1 \le \eta \le +1$ spanning the cell.



$$\begin{aligned} N_{1(\xi,\eta)} &\equiv \tfrac{1}{4}(1-\xi)(1-\eta) \\ N_{2(\xi,\eta)} &\equiv \tfrac{1}{4}(1+\xi)(1-\eta) \\ N_{3(\xi,\eta)} &\equiv \tfrac{1}{4}(1+\xi)(1+\eta) \\ N_{4(\xi,\eta)} &\equiv \tfrac{1}{4}(1-\xi)(1+\eta) \end{aligned} \tag{4.15}$$

The various scalar and cartesian vector quantities defined on the element are then obtained

as weighted sums over the nodal values, e.g.

$$\mathbf{r}_{(\xi,\eta)} = \sum_{j=1}^{4} \mathbf{r}_j \, N_j \tag{4.16}$$

$$\bar{\bar{\mathbf{f}}}_{(\xi,\eta)} = \sum_{j=1}^{4} \bar{\bar{\mathbf{f}}}_j \, N_j \qquad \text{etc.} \tag{4.17}$$

The parametric derivatives of any interpolated quantity are similarly constructed by a sum over the nodal values weighted by the derivatives of the basis functions.

$$\partial_\xi \mathbf{r}_{(\xi,\eta)} = \sum_{j=1}^{4} \mathbf{r}_j \, \partial_\xi N_j \tag{4.18}$$

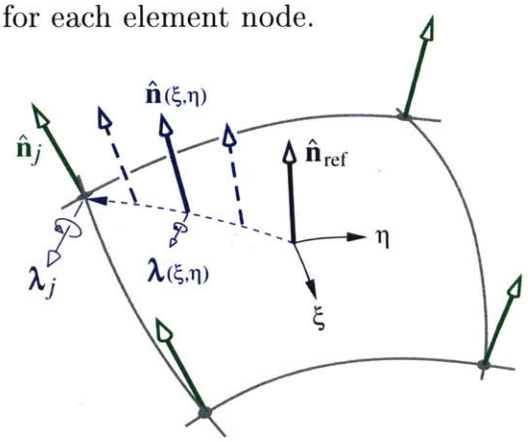
$$\partial_\eta \mathbf{r}_{(\xi,\eta)} = \sum_{j=1}^{4} \mathbf{r}_j \, \partial_\eta N_j \qquad \text{etc.} \tag{4.19}$$

4.2.2 Spherical Interpolation

Application of the simple interpolation as in (4.16) to the nodal normal vectors $\hat{\mathbf{n}}_j$ would not in general produce a unit vector, so we instead perform the following spherical interpolation procedure. An element reference unit vector $\hat{\mathbf{n}}_{\text{ref}}$ is first defined normal to the $\mathbf{r}_{(\xi,\eta)}$ interpolated surface at the element centroid $(\xi, \eta) = (0, 0)$.

$$\hat{\mathbf{n}}_{\text{ref}} \equiv \left. \frac{\partial_\xi \mathbf{r} \times \partial_\eta \mathbf{r}}{|\partial_\xi \mathbf{r} \times \partial_\eta \mathbf{r}|} \right|_{0,0} \tag{4.20}$$

This is then used to define the log-quaternions $\boldsymbol{\lambda}_j$ which would be needed to rotate $\hat{\mathbf{n}}_{\text{ref}}$ into $\hat{\mathbf{n}}_j$ for each element node.



$$\mathbf{v}_j = \hat{\mathbf{n}}_{\text{ref}} \times \hat{\mathbf{n}}_j \tag{4.21}$$

$$\theta_j = \arctan \frac{|\mathbf{v}_j|}{\hat{\mathbf{n}}_{\text{ref}} \cdot \hat{\mathbf{n}}_j} \tag{4.22}$$

$$\boldsymbol{\lambda}_j = \frac{\mathbf{v}_j}{|\mathbf{v}_j|} \frac{\theta_j}{2} \tag{4.23}$$

The nodal $\boldsymbol{\lambda}_j$ values are then interpolated using the following modified basis functions.

$$\begin{aligned} \check{N}_{1(\xi,\eta)} &\equiv [1 - (1-\xi^2)(1-\eta^2)] \, N_1 \\ \check{N}_{2(\xi,\eta)} &\equiv [1 - (1-\xi^2)(1-\eta^2)] \, N_2 \\ \check{N}_{3(\xi,\eta)} &\equiv [1 - (1-\xi^2)(1-\eta^2)] \, N_3 \\ \check{N}_{4(\xi,\eta)} &\equiv [1 - (1-\xi^2)(1-\eta^2)] \, N_4 \end{aligned} \tag{4.24}$$

$$\boldsymbol{\lambda}_{(\xi,\eta)} \;=\; \textstyle\sum_{j=1}^{4} \boldsymbol{\lambda}_j \, \check{N}_j \tag{4.25}$$

This interpolation implicitly includes a fifth term $\boldsymbol{\lambda}_{\text{ref}}(1-\xi^2)(1-\eta^2)$, where $\boldsymbol{\lambda}_{\text{ref}}$ would be defined from $\hat{\mathbf{n}}_{\text{ref}}$ via equations (4.21)–(4.23). But since $\boldsymbol{\lambda}_{\text{ref}} \sim \hat{\mathbf{n}}_{\text{ref}} \times \hat{\mathbf{n}}_{\text{ref}} = \mathbf{0}$ identically, this fifth term is omitted from (4.25). This also forces the interpolated $\hat{\mathbf{n}}_{(\xi,\eta)}$ to be equal to $\hat{\mathbf{n}}_{\text{ref}}$ at the element centroid. The necessity of this modified interpolation scheme comes from the disconnect between the nodal normal vectors $\hat{\mathbf{n}}_j$ and the geometry, as shown in Figure 4-1. The nodal normals are purely functions of the log-quaternion basis rotations, $\hat{\mathbf{n}}_j = f(\boldsymbol{\Lambda}_j)$ via equation (3.1), so for a shell grid in a sawtooth configuration, unmodified interpolation of the nodal normals yields a constant $\hat{\mathbf{n}}_{\text{ref}}$, and therefore a surface gradient of zero. This removes the second term of the normal-force equilibrium equation (3.27) making the sawtooth mode invisible. The solution of matching the interpolated normal to the reference normal at the element centroid via the modified interpolation scheme couples the interpolated normals to the geometry and allows the normal-force equation to sense the grid sawtooth mode.

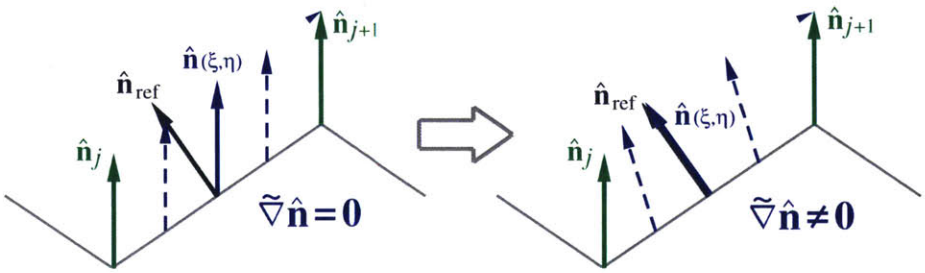


Figure 4-1: Interpolation of the nodal normal vectors $\hat{\mathbf{n}}_j$ across a sawtooth geometry, with (right) and without (left) interpolation via the modified basis functions that match the interpolated normal to the geometrically-defined reference normal $\hat{\mathbf{n}}_{\text{ref}}$ at the element centroid.

The final step is to construct $\hat{\mathbf{n}}_{(\xi,\eta)}$ by rotating $\hat{\mathbf{n}}_{\text{ref}}$ using the interpolated $\boldsymbol{\lambda}_{(\xi,\eta)}$ as follows.

$$w_{(\xi,\eta)} \;=\; \cos|\boldsymbol{\lambda}| \quad \simeq \quad 1 - \tfrac{1}{2}|\boldsymbol{\lambda}|^2 \tag{4.26}$$

$$\mathbf{v}_{(\xi,\eta)} \;=\; \frac{\sin|\boldsymbol{\lambda}|}{|\boldsymbol{\lambda}|}\,\boldsymbol{\lambda} \quad \simeq \quad \left(1 - \tfrac{1}{6}|\boldsymbol{\lambda}|^2\right)\boldsymbol{\lambda} \tag{4.27}$$

$$\hat{\mathbf{n}}_{(\xi,\eta)} \;=\; \hat{\mathbf{n}}_{\text{ref}} \;+\; 2\,\mathbf{v} \times (\mathbf{v} \times \hat{\mathbf{n}}_{\text{ref}} \;+\; w\,\hat{\mathbf{n}}_{\text{ref}}) \tag{4.28}$$

The approximate forms in (4.26) and (4.27) are used whenever $|\boldsymbol{\lambda}|^2 < \epsilon_{\text{machine}}$, thus avoiding roundoff problems or a divide by zero in this limiting case.

The parametric derivatives of $\hat{\mathbf{n}}_{(\xi,\eta)}$ are obtained via the chain rule,

$$\partial_\xi \boldsymbol{\lambda} = \sum_{j=1}^{4} \boldsymbol{\lambda}_j \, \partial_\xi \check{N}_j \tag{4.29}$$

$$\partial_\xi w = -\frac{\sin|\boldsymbol{\lambda}|}{|\boldsymbol{\lambda}|} \boldsymbol{\lambda} \cdot \partial_\xi \boldsymbol{\lambda} \simeq -\boldsymbol{\lambda} \cdot \partial_\xi \boldsymbol{\lambda} \tag{4.30}$$

$$\partial_\xi \mathbf{v} = \frac{\sin|\boldsymbol{\lambda}|}{|\boldsymbol{\lambda}|} \partial_\xi \boldsymbol{\lambda} + \left(\cos|\boldsymbol{\lambda}| - \frac{\sin|\boldsymbol{\lambda}|}{|\boldsymbol{\lambda}|} \right) \frac{\boldsymbol{\lambda}}{|\boldsymbol{\lambda}|^2} \boldsymbol{\lambda} \cdot \partial_\xi \boldsymbol{\lambda} \simeq \partial_\xi \boldsymbol{\lambda} \tag{4.31}$$

$$\partial_\xi \hat{\mathbf{n}} = 2\, \partial_\xi \mathbf{v} \times (\mathbf{v} \times \hat{\mathbf{n}}_{\text{ref}} + w\, \hat{\mathbf{n}}_{\text{ref}}) + 2\, \mathbf{v} \times (\partial_\xi \mathbf{v} \times \hat{\mathbf{n}}_{\text{ref}} + \partial_\xi w\, \hat{\mathbf{n}}_{\text{ref}}) \tag{4.32}$$

and likewise for $\partial_\eta \hat{\mathbf{n}}$. The approximate forms in (4.30) and (4.31) are used whenever $|\boldsymbol{\lambda}|^2 < \epsilon_{\text{machine}}$.

Operations (4.16)–(4.32) are also applied to the undeformed geometry $\mathbf{r}_{0_j}, \hat{\mathbf{n}}_{0_j}$ to obtain the interpolated $\hat{\mathbf{n}}_{0(\xi,\eta)}$ vector and its parametric derivatives.

4.2.3 Tangential Gradients

The tangential gradient components of some interpolated quantities are required for the construction of the equation residuals. From the interpolations of $\mathbf{r}_j = (x_j, y_j, z_j)$ above we have

$$\begin{aligned}
x_{(\xi,\eta)} &= \textstyle\sum_j x_j \, N_{j(\xi,\eta)} \quad ; \quad \partial_\xi x = \sum_j x_j \, \partial_\xi N_j \quad , \quad \partial_\eta x = \sum_j x_j \, \partial_\eta N_j \\
y_{(\xi,\eta)} &= \textstyle\sum_j y_j \, N_{j(\xi,\eta)} \quad ; \quad \partial_\xi y = \sum_j y_j \, \partial_\xi N_j \quad , \quad \partial_\eta y = \sum_j y_j \, \partial_\eta N_j \\
z_{(\xi,\eta)} &= \textstyle\sum_j z_j \, N_{j(\xi,\eta)} \quad ; \quad \partial_\xi z = \sum_j z_j \, \partial_\xi N_j \quad , \quad \partial_\eta z = \sum_j z_j \, \partial_\eta N_j
\end{aligned} \tag{4.33}$$

which are then used to obtain the three Cartesian components of the basis function gradient

$$\widetilde{\nabla} N_{j\,(\xi,\eta)} \equiv (\widetilde{\nabla} N_j)_x \, \hat{\mathbf{x}} + (\widetilde{\nabla} N_j)_y \, \hat{\mathbf{y}} + (\widetilde{\nabla} N_j)_z \, \hat{\mathbf{z}} \tag{4.34}$$

via solution of the following 3×3 linear system.

$$\begin{bmatrix} \partial_\xi x & \partial_\xi y & \partial_\xi z \\ \partial_\eta x & \partial_\eta y & \partial_\eta z \\ n_x & n_y & n_z \end{bmatrix} \begin{Bmatrix} (\widetilde{\nabla} N_j)_x \\ (\widetilde{\nabla} N_j)_y \\ (\widetilde{\nabla} N_j)_z \end{Bmatrix} = \begin{Bmatrix} \partial_\xi N_j \\ \partial_\eta N_j \\ 0 \end{Bmatrix} \tag{4.35}$$

Note that the vector $\{\partial_\xi x,\ \partial_\xi y,\ \partial_\xi z\}$ is also the covariant tangential basis vector, $\mathbf{a}_\xi \equiv \partial_\xi \mathbf{r}$. Here n_x, n_y, n_z are the components of the $\hat{\mathbf{n}}$ vector at that location, so that the last line in the system enforces the requirement that $\widetilde{\nabla} N_j$ has no shell-normal component. Once the system (4.35) is solved for the three components of $\widetilde{\nabla} N_j$, the tangential gradient components

of any interpolated quantity are obtained by summation, for example

$$(\widetilde{\nabla}\Delta\mathbf{n})_x = \sum_{j=1}^{4} \Delta\mathbf{n}_j \, (\widetilde{\nabla}N_j)_x \tag{4.36}$$

$$(\widetilde{\nabla}\Delta\mathbf{n})_y = \sum_{j=1}^{4} \Delta\mathbf{n}_j \, (\widetilde{\nabla}N_j)_y \tag{4.37}$$

$$(\widetilde{\nabla}\Delta\mathbf{n})_z = \sum_{j=1}^{4} \Delta\mathbf{n}_j \, (\widetilde{\nabla}N_j)_z \tag{4.38}$$

System (4.35) is also applied to directly determine the components of $\widetilde{\nabla}\hat{\mathbf{n}}$, specifically

$$\begin{bmatrix} \partial_\xi x & \partial_\xi y & \partial_\xi z \\ \partial_\eta x & \partial_\eta y & \partial_\eta z \\ n_x & n_y & n_z \end{bmatrix} \begin{bmatrix} - & (\widetilde{\nabla}\hat{\mathbf{n}})_x & - \\ - & (\widetilde{\nabla}\hat{\mathbf{n}})_y & - \\ - & (\widetilde{\nabla}\hat{\mathbf{n}})_z & - \end{bmatrix} = \begin{bmatrix} - & \partial_\xi\hat{\mathbf{n}} & - \\ - & \partial_\eta\hat{\mathbf{n}} & - \\ - & \mathbf{0} & - \end{bmatrix} \tag{4.39}$$

where the parametric derivatives $\partial_\xi\hat{\mathbf{n}}, \partial_\eta\hat{\mathbf{n}}$ on the right-hand side are defined by (4.29)–(4.32).

4.2.4 Interpolated Data

The nodal data listed below is interpolated to specified finite-element Gauss points ξ, η in the element interior via the $N_{j(\xi,\eta)}$ interpolation functions. The parametric derivatives of some of the data are also computed at the Gauss points by interpolation using the function derivatives $\partial_\alpha N_{j(\xi,\eta)}$, and the tangential gradients are computed using the function gradient $\widetilde{\nabla}N_{j(\xi,\eta)}$ defined by (4.35). In the following table, abbreviations are: UG (undeformed geometry),

DG (deformed geometry), CF (Cartesian frame), LNDF (local non-degenerate frame).

sym.	N	prim. var. dep.	secondary var. dep.	par. dep.	axes	description
$\partial_\alpha \mathbf{r}_0$	6			$\mathbf{r}_{0_j}$	xyz	UG position derivatives
$\partial_\alpha \mathbf{r}$	6	$\mathbf{r}_j$			xyz	DG position derivatives
$\bar{\bar{\mathbf{f}}}$	6		$\bar{\bar{\mathbf{f}}}_j$		xyz	CF stress resultants
$\bar{\bar{\mathbf{m}}}$	6		$\bar{\bar{\mathbf{m}}}_j$		xyz	CF stress-moment resultants
$\mathbf{q}$	3		$\mathbf{q}_j$		xyz	CF applied loads
$\mathbf{a}$	3			$\mathbf{a}_j$	xyz	CF acceleration
μ	1			μ_j	–	mass/area density
$\bar{\bar{f}}^{\mathsf{w}}$	3 or 5		$\bar{\bar{f}}^{\mathsf{w}}_j$		w	LNDF stress resultants
$\partial_\alpha \vec{\bar{f}}^{\mathsf{w}}_{\beta n}$	4		$\bar{\bar{f}}^{\mathsf{w}}_j$		w	LNDF normal stress resultant derivatives
$\bar{\bar{m}}^{\mathsf{w}}$	3		$\bar{\bar{m}}^{\mathsf{w}}_j$		w	LNDF stress-moment resultants
$\bar{\bar{A}}^{\mathsf{w}}$	6		$\bar{\bar{\mathbf{e}}}_j, \bar{\bar{\mathbf{w}}}_j$	$\bar{\bar{A}}^{\mathsf{e}}_j$	w	LNDF extensional stiffness matrix
$\bar{\bar{B}}^{\mathsf{w}}$	6		$\bar{\bar{\mathbf{e}}}_j, \bar{\bar{\mathbf{w}}}_j$	$\bar{\bar{B}}^{\mathsf{e}}_j$	w	LNDF ext.-bend coupling stiffness matrix
$\bar{\bar{D}}^{\mathsf{w}}$	6		$\bar{\bar{\mathbf{e}}}_j, \bar{\bar{\mathbf{w}}}_j$	$\bar{\bar{D}}^{\mathsf{e}}_j$	w	LNDF bending stiffness matrix
$\bar{\bar{A}}'^{\mathsf{w}}$	2		$\bar{\bar{\mathbf{e}}}_j, \bar{\bar{\mathbf{w}}}_j$	$\bar{\bar{A}}'^{\mathsf{e}}_j$	w	LNDF ext.-shear coupling stiffness matrix
$\partial_\alpha \bar{\bar{A}}'^{\mathsf{w}}$	4		$\bar{\bar{\mathbf{e}}}_j, \bar{\bar{\mathbf{w}}}_j$	$\bar{\bar{A}}'^{\mathsf{e}}_j$	w	LNDF ext.-shear stiffness derivatives

The following unit normal vector data is obtained using the spherical interpolation procedure given by (4.20)–(4.28), and its parametric derivatives are obtained using (4.29)–(4.32).

symbol	N	var. dep.	par. dep.	axes	description
$\hat{\mathbf{n}}_0$	3		$\mathbf{r}_{0_j}, \mathbf{\Lambda}_{0_j}$	xyz	UG normal vector
$\partial_\alpha \hat{\mathbf{n}}_0$	6		$\mathbf{r}_{0_j}, \mathbf{\Lambda}_{0_j}$	xyz	UG normal vector derivatives
$\hat{\mathbf{n}}$	3	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	DG normal vector
$\partial_\alpha \hat{\mathbf{n}}$	6	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	DG normal vector derivatives

The following unit in-plane vector data is obtained using the spherical interpolation procedure given by (4.20)–(4.25) and (4.47)–(4.50), and its parametric derivatives are obtained

using (4.51)–(4.57).

symbol	num.	var. dep.	par. dep.	axes	description
$\hat{\mathbf{w}}_{0\alpha}$	6		$\mathbf{r}_{0_j}, \mathbf{\Lambda}_{0_j}$	xyz	UG in-plane vectors
$\partial_\beta \hat{\mathbf{w}}_{0\alpha}$	12		$\mathbf{r}_{0_j}, \mathbf{\Lambda}_{0_j}$	xyz	UG in-plane vector derivatives
$\hat{\mathbf{w}}_\alpha$	6	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	DG in-plane vectors
$\partial_\beta \hat{\mathbf{w}}_\alpha$	12	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	DG in-plane vector derivatives

The following tangential gradients of the normal vectors are obtained by solving the linear system (3.66), using the interpolated $\partial_\alpha \hat{\mathbf{n}}_0$ and $\partial_\alpha \hat{\mathbf{n}}$ derivatives on the righthand side.

symbol	N	var. dep.	par. dep.	axes	description
$\widetilde{\nabla}\hat{\mathbf{n}}_0$	9		$\mathbf{r}_{0_j}, \mathbf{\Lambda}_{0_j}$	xyz	UG normal vector gradient
$\widetilde{\nabla}\hat{\mathbf{n}}$	9	$\mathbf{r}_j, \mathbf{\Lambda}_j$		xyz	DG normal vector gradient

4.2.5 Interior Secondary Variables

These are the secondary variables that are calculated at the Gauss points using data interpolated to the element interior from the nodes. Nonlinear operations on dependents is allowed at Gauss points (whereas operations were restricted to rotations and summations at the nodes).

symbol	N	interpolated dependence	axes	description
$\bar{\bar{\boldsymbol{\varepsilon}}}^{(\mathbf{f})}$	6	$\hat{\mathbf{w}}_\alpha, \bar{\bar{f}}^{\mathbf{w}}, \bar{\bar{m}}^{\mathbf{w}}, \bar{\bar{A}}^{\mathbf{w}}, \bar{\bar{B}}^{\mathbf{w}}, \bar{\bar{A}}'^{\mathbf{w}}$	xyz	stress-dependent strain resultants
$\bar{\bar{\boldsymbol{\kappa}}}^{(\mathbf{f})}$	6	$\hat{\mathbf{w}}_\alpha, \bar{\bar{f}}^{\mathbf{w}}, \bar{\bar{m}}^{\mathbf{w}}, \bar{\bar{B}}^{\mathbf{w}}, \bar{\bar{D}}^{\mathbf{w}}$	xyz	stress-dep. strain-curvature resultants
$\bar{\bar{\boldsymbol{\varepsilon}}}^{(\mathbf{r})}$	6	$\partial_\alpha \mathbf{r}_0, \partial_\alpha \mathbf{r}$	xyz	geom.-dependent strain resultants
$\bar{\bar{\boldsymbol{\kappa}}}^{(\mathbf{r})}$	6	$\partial_\alpha \mathbf{r}_0, \partial_\alpha \mathbf{r}, \hat{\mathbf{n}}, \partial_\alpha \hat{\mathbf{n}}, \hat{\mathbf{w}}_\alpha, \partial_\alpha \hat{\mathbf{w}}_\beta$ $\partial_\alpha \hat{\mathbf{n}}_0, \bar{\bar{f}}^{\mathbf{w}}, \partial_\alpha \vec{f}^{\mathbf{w}}_{\beta n}, \bar{\bar{A}}'^{\mathbf{w}}, \partial_\alpha \bar{\bar{A}}'^{\mathbf{w}}$	xyz	geom.-dep. strain-curvature resultants

4.3 Edge Interpolation

Quantities along an edge are defined from the nodal values using the linear interpolation functions

$$\begin{aligned} \tilde{N}_{1(\xi)} &\equiv \tfrac{1}{2}(1-\xi) \\ \tilde{N}_{2(\xi)} &\equiv \tfrac{1}{2}(1+\xi) \end{aligned} \tag{4.40}$$

which are the same as N_1 and N_2 in (4.15), with $(\xi, \eta) = (\xi, -1)$. We then have

$$\mathbf{r}_{(\xi)} = \textstyle\sum_{j=1}^{2} \mathbf{r}_j \tilde{N}_j \tag{4.41}$$

$$\bar{\bar{\mathbf{f}}}_{(\xi)} = \textstyle\sum_{j=1}^{2} \bar{\bar{\mathbf{f}}}_j \tilde{N}_j \qquad \text{etc.} \tag{4.42}$$

This interpolation is needed for evaluation of the edge integrals in (3.22), (3.27), and (3.29). The normal vector $\hat{\mathbf{n}}_{(\xi)}$ along an edge is obtained using spherical interpolation, with one of the edge's node normal vectors used as the reference vector. Hence, we have

$$\hat{\mathbf{n}}_{\text{ref}} \;=\; \hat{\mathbf{n}}_1 \tag{4.43}$$

and equations (4.21)–(4.23) are then used to obtain $\boldsymbol{\lambda}_1$ $(= \mathbf{0})$, and $\boldsymbol{\lambda}_2$. These are then linearly interpolated along the edge,

$$\boldsymbol{\lambda}_{(\xi)} \;=\; \textstyle\sum_{j=1}^{2} \boldsymbol{\lambda}_j \, \tilde{N}_j \tag{4.44}$$

and equations (4.26)–(4.28) then give the interpolated $\hat{\mathbf{n}}_{(\xi)}$.

4.4 Equation Weighted Residuals

4.4.1 Residual Weights

HSM uses a Galerkin-type finite-element formulation, where the residual weighting function $W_{i(\xi,\eta)}$ associated with node i is chosen to be the "tent" function formed from the union of the $N_{i(\xi,\eta)}$ interpolants, shown in Figure 4-3. On each element we then have

$$W_{i\,(\xi,\eta)} \;=\; N_{i\,(\xi,\eta)} \tag{4.45}$$

$$\widetilde{\nabla} W_i \;=\; \left(\widetilde{\nabla} W_i\right)_x \hat{\mathbf{x}} + \left(\widetilde{\nabla} W_i\right)_y \hat{\mathbf{y}} + \left(\widetilde{\nabla} W_i\right)_z \hat{\mathbf{z}} \tag{4.46}$$

where the weighting function gradient $\widetilde{\nabla} W_i = \widetilde{\nabla} N_{i\,(\xi,\eta)}$ is defined in the global xyz axes. This makes it operable with the $\bar{\bar{\mathbf{f}}}, \bar{\bar{\mathbf{m}}}, \mathbf{w}$ tensors and vectors which are defined in these same axes.

The $\hat{\mathbf{w}}_{1_i}, \hat{\mathbf{w}}_{2_i(\xi,\eta)}$ basis vectors for forming the various residuals are defined to be the residual node's $\hat{\mathbf{e}}_{1_i}, \hat{\mathbf{e}}_{2_i}$ vectors rotated onto the curved shell surface. The generic operation is shown in Figure 3-3, and here we choose the residual node i as the reference point. The rotation can then use the same log-quaternions which were used to perform the spherical interpolation of the $\hat{\mathbf{n}}_j$ vectors. The only difference is that here $\hat{\mathbf{n}}_i$ plays the role of the reference vector, which is implemented by subtracting $\boldsymbol{\lambda}_i$ from the local $\boldsymbol{\lambda}_{(\xi,\eta)}$ defined earlier by (4.25). The operation is then

$$w_{(\xi,\eta)} \;=\; \cos|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i| \;\simeq\; 1 - \tfrac{1}{2}|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|^2 \tag{4.47}$$

$$\mathbf{v}_{(\xi,\eta)} \;=\; \frac{\sin|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|}{|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|}\,(\boldsymbol{\lambda}-\boldsymbol{\lambda}_i) \;\simeq\; \left(1 - \tfrac{1}{6}|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|^2\right)(\boldsymbol{\lambda}-\boldsymbol{\lambda}_i) \tag{4.48}$$

$$\hat{\mathbf{w}}_{1_i(\xi,\eta)} \;=\; \hat{\mathbf{e}}_{1_i} + 2\,\mathbf{v} \times \left(\mathbf{v} \times \hat{\mathbf{e}}_{1_i} + w\,\hat{\mathbf{e}}_{1_i}\right) \tag{4.49}$$

$$\hat{\mathbf{w}}_{2_i(\xi,\eta)} \;=\; \hat{\mathbf{e}}_{2_i} + 2\,\mathbf{v} \times \left(\mathbf{v} \times \hat{\mathbf{e}}_{2_i} + w\,\hat{\mathbf{e}}_{2_i}\right) \tag{4.50}$$

The approximate forms in (4.47),(4.48) are used whenever $|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|^2 < \epsilon_{\text{machine}}$. With these definitions, at the (ξ,η) location corresponding to the residual node i, the $\hat{\mathbf{w}}_{i(\xi,\eta)}$ vectors coincide with that node's $\hat{\mathbf{e}}_i$ vectors.

The parametric derivatives of $\hat{\mathbf{w}}_{1_i(\xi,\eta)}$ are obtained via the chain rule,

$$\partial_\xi \boldsymbol{\lambda} = \textstyle\sum_{j=1}^{4} \boldsymbol{\lambda}_j \, \partial_\xi \check{N}_j \tag{4.51}$$

$$\partial_\xi w = -\frac{\sin|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|}{|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|} (\boldsymbol{\lambda}-\boldsymbol{\lambda}_i) \cdot \partial_\xi \boldsymbol{\lambda} \tag{4.52}$$

$$\simeq -(\boldsymbol{\lambda}-\boldsymbol{\lambda}_i) \cdot \partial_\xi \boldsymbol{\lambda} \tag{4.53}$$

$$\partial_\xi \mathbf{v} = \frac{\sin|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|}{|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|} \partial_\xi \boldsymbol{\lambda} + \left(\cos|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i| - \frac{\sin|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|}{|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|}\right) \frac{\boldsymbol{\lambda}-\boldsymbol{\lambda}_i}{|\boldsymbol{\lambda}-\boldsymbol{\lambda}_i|^2} (\boldsymbol{\lambda}-\boldsymbol{\lambda}_i) \cdot \partial_\xi \boldsymbol{\lambda} \tag{4.54}$$

$$\simeq \partial_\xi \boldsymbol{\lambda} \tag{4.55}$$

$$\partial_\xi \hat{\mathbf{w}}_{1_i} = 2\,\partial_\xi \mathbf{v} \times \left(\mathbf{v} \times \hat{\mathbf{e}}_{1_i} + w\,\hat{\mathbf{e}}_{1_i}\right) + 2\,\mathbf{v} \times \left(\partial_\xi \mathbf{v} \times \hat{\mathbf{e}}_{1_i} + \partial_\xi w\,\hat{\mathbf{e}}_{1_i}\right) \tag{4.56}$$

$$\partial_\xi \hat{\mathbf{w}}_{2_i} = 2\,\partial_\xi \mathbf{v} \times \left(\mathbf{v} \times \hat{\mathbf{e}}_{2_i} + w\,\hat{\mathbf{e}}_{2_i}\right) + 2\,\mathbf{v} \times \left(\partial_\xi \mathbf{v} \times \hat{\mathbf{e}}_{2_i} + \partial_\xi w\,\hat{\mathbf{e}}_{2_i}\right) \tag{4.57}$$

and likewise for $\partial_\eta \hat{\mathbf{w}}_{1_i}$.

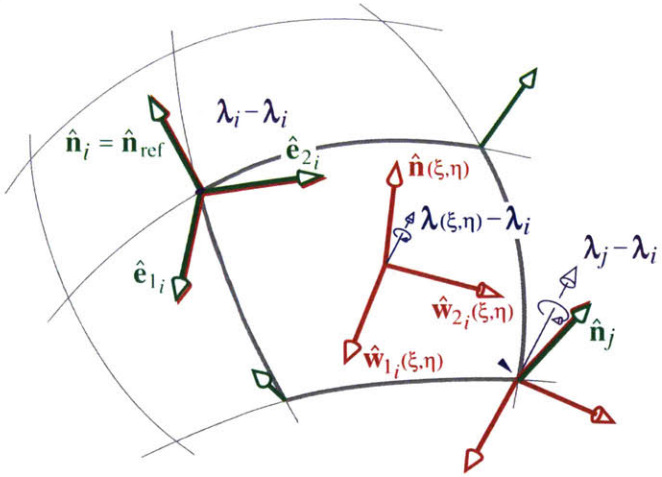


Figure 4-2: Interpolation of the $\bar{\bar{\mathbf{w}}}$ basis (same as in Figure 3-3), using $\hat{\mathbf{e}}_{1_i}, \hat{\mathbf{e}}_{2_i}, \hat{\mathbf{n}}_i$ at the residual node as the reference vectors. The nodal $\boldsymbol{\lambda}_j$ given by (4.23) and the interpolated $\boldsymbol{\lambda}_{(\xi,\eta)}$ given by (4.25) are re-used here.

Equations (4.29)–(4.32) are also applied with the w and $\mathbf{v}$ defined above to obtain the $\partial_\xi \hat{\mathbf{w}}$ and $\partial_\eta \hat{\mathbf{w}}$ parametric derivatives. These are then used in system (4.35) to obtain the tangential gradients $\widetilde{\nabla} \hat{\mathbf{w}}$.

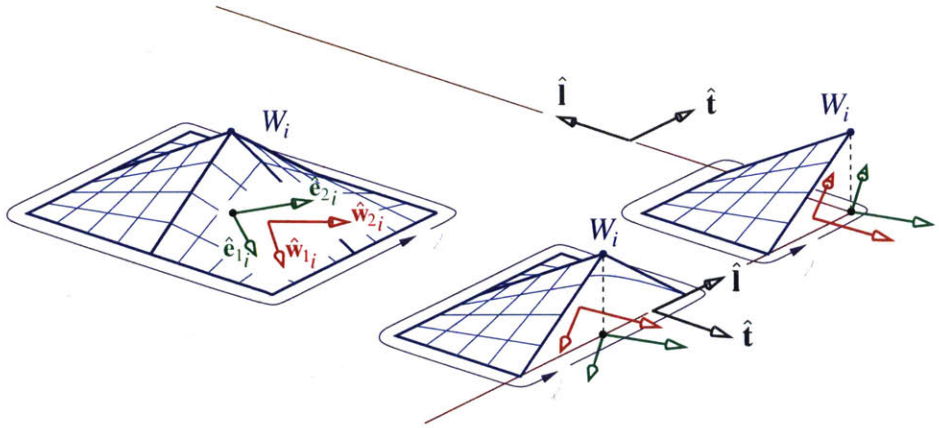


Figure 4-3: Residual weighting functions $W_{i(\xi,\eta)}$ and residual basis vectors $\hat{\mathbf{w}}_{1_i}, \hat{\mathbf{w}}_{2_i(\xi,\eta)}$ associated with interior, edge, and corner nodes. The edge-normal and edge-parallel tangent vectors $\hat{\mathbf{t}}, \hat{\mathbf{l}}$ and edge length coordinate ℓ are also shown.

4.4.2 Gauss Integration

Per standard procedure, all area integrals will be recast in terms of the element coordinates ξ, η and their Jacobian J_0, and then numerically evaluated using 4-point Gaussian quadrature,

$$\iint \mathcal{F}\, \mathrm{d}\mathcal{A}_0 \;=\; \iint \mathcal{F}\, J_0 \;\mathrm{d}\xi\, \mathrm{d}\eta \;\simeq\; \sum_{k=1}^{4} \mathcal{F}_{(\xi_k,\eta_k)}\, J_{0(\xi_k,\eta_k)}\; w_k \tag{4.58}$$

$$J_0 \;=\; |\,\partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0| \tag{4.59}$$

where the index k runs over the Gauss points (ξ_k, η_k) and w_k are the corresponding Gauss weights. The integration is performed over the undeformed geometry, primarily because this simplifies linearization of the resulting residual expressions for Newton solution.

For the edge line integrals appearing in (3.22) and (3.29), 2-point Gaussian integration is used.

$$\int \mathcal{F}\, \mathrm{d}\ell_0 \;=\; \int \mathcal{F}\, J_0 \;\mathrm{d}\xi \;\simeq\; \sum_{k=1}^{2} \mathcal{F}_{(\xi_k)}\, J_0 \;w_k \tag{4.60}$$

Here, $-1 \leq \xi \leq +1$ is the parameter along the edge, and the Jacobian is a constant $J_0 = \Delta\ell_0/2$ where $\Delta\ell_0$ is the edge length of the undeformed element.

4.4.3 Strain-displacement residuals

In the preceding sections, two different expressions have been developed for the strains and curvature changes. The values related to the geometry metric changes, denoted here by $\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{r})$ and $\bar{\bar{\boldsymbol{\kappa}}}(\mathbf{r})$, are given by equations (3.69) and (3.70) applied over the element interior.

The values related to the force and moment resultants, denoted here by $\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{f})$ and $\bar{\bar{\boldsymbol{\kappa}}}(\mathbf{f})$, are given by the matrix equation (3.50) applied at each node, and then interpolated to the element interior. The appropriate governing equations simply state that these two alternative definitions are numerically equal. The corresponding weighted residuals are then defined as

$$\begin{Bmatrix} \mathcal{R}_i^{\varepsilon 11} \\ \mathcal{R}_i^{\varepsilon 22} \\ \mathcal{R}_i^{\varepsilon 12} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} \hat{\mathbf{w}}_{1_i} \cdot \left[\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{r}) - \bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{f}) \right] \cdot \hat{\mathbf{w}}_{1_i} \\ \hat{\mathbf{w}}_{2_i} \cdot \left[\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{r}) - \bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{f}) \right] \cdot \hat{\mathbf{w}}_{2_i} \\ \hat{\mathbf{w}}_{1_i} \cdot \left[\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{r}) - \bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{f}) \right] \cdot \hat{\mathbf{w}}_{2_i} \end{Bmatrix} W_i \, \mathrm{d}\mathcal{A}_0 \tag{4.61}$$

$$\begin{Bmatrix} \mathcal{R}_i^{\kappa 11} \\ \mathcal{R}_i^{\kappa 22} \\ \mathcal{R}_i^{\kappa 12} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} \hat{\mathbf{w}}_{1_i} \cdot \left[\bar{\bar{\boldsymbol{\kappa}}}(\mathbf{r}) - \bar{\bar{\boldsymbol{\kappa}}}(\mathbf{f}) \right] \cdot \hat{\mathbf{w}}_{1_i} \\ \hat{\mathbf{w}}_{2_i} \cdot \left[\bar{\bar{\boldsymbol{\kappa}}}(\mathbf{r}) - \bar{\bar{\boldsymbol{\kappa}}}(\mathbf{f}) \right] \cdot \hat{\mathbf{w}}_{2_i} \\ \hat{\mathbf{w}}_{1_i} \cdot \left[\bar{\bar{\boldsymbol{\kappa}}}(\mathbf{r}) - \bar{\bar{\boldsymbol{\kappa}}}(\mathbf{f}) \right] \cdot \hat{\mathbf{w}}_{2_i} \end{Bmatrix} W_i \, \mathrm{d}\mathcal{A}_0 \tag{4.62}$$

where the sums are taken over all the elements which contain node i, pictured in Figure 4-3. The dot products with the $\hat{\mathbf{w}}_{1_i}, \hat{\mathbf{w}}_{1_i}$ vectors enforce equality only for the in-plane components of the tensors.

4.4.4 Force equilibrium residuals

The area-integral part of the in-plane and normal force equilibrium equations (3.22) and (3.27) for node i is evaluated as follows, again using the $\hat{\mathbf{w}}_{1_i}, \hat{\mathbf{w}}_{1_i}$ vectors.

$$\begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} -\hat{\mathbf{w}}_{1_i} \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \right) \cdot \hat{\mathbf{w}}_{1_i} W_i + \mathbf{q} \cdot \hat{\mathbf{w}}_{1_i} W_i + \mu \, (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{w}}_{1_i} W_i \\ -\hat{\mathbf{w}}_{2_i} \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \right) \cdot \hat{\mathbf{w}}_{2_i} W_i + \mathbf{q} \cdot \hat{\mathbf{w}}_{2_i} W_i + \mu \, (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{w}}_{2_i} W_i \\ -\left(\widetilde{\nabla} \cdot \bar{\bar{\mathbf{m}}} \right) \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \right) \cdot \hat{\mathbf{n}} \, W_i + \mathbf{q} \cdot \hat{\mathbf{n}} \, W_i + \mu \, (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{n}} \, W_i \end{Bmatrix} \mathrm{d}\mathcal{A}_0 \tag{4.63}$$

The various terms in the residuals above are evaluated in the xyz basis as follows.

$$\begin{aligned} \mathbf{w} \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} W_i &= \left[w_x f_{xx} + w_y f_{xy} + w_z f_{xz} \right] (\widetilde{\nabla} W_i)_x \\ &+ \left[w_x f_{xy} + w_y f_{yy} + w_z f_{yz} \right] (\widetilde{\nabla} W_i)_y \\ &+ \left[w_x f_{xz} + w_y f_{yz} + w_z f_{zz} \right] (\widetilde{\nabla} W_i)_z \end{aligned} \tag{4.64}$$

$$\begin{aligned} \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \right) \cdot \hat{\mathbf{w}} &= f_{xx} \, (\widetilde{\nabla} w_x)_x + f_{xy} \, (\widetilde{\nabla} w_x)_y + f_{xz} \, (\widetilde{\nabla} w_x)_z \\ &+ f_{xy} \, (\widetilde{\nabla} w_y)_x + f_{yy} \, (\widetilde{\nabla} w_y)_y + f_{yz} \, (\widetilde{\nabla} w_y)_z \\ &+ f_{xz} \, (\widetilde{\nabla} w_z)_x + f_{yz} \, (\widetilde{\nabla} w_z)_y + f_{zz} \, (\widetilde{\nabla} w_z)_z \end{aligned} \tag{4.65}$$

$$\begin{aligned} \left(\widetilde{\nabla} \cdot \bar{\bar{\mathbf{m}}} \right) \cdot \widetilde{\nabla} W_i &= \left[(\widetilde{\nabla} m_{xx})_x + (\widetilde{\nabla} m_{xy})_y + (\widetilde{\nabla} m_{xz})_z \right] (\widetilde{\nabla} W_i)_x \\ &+ \left[(\widetilde{\nabla} m_{xy})_x + (\widetilde{\nabla} m_{yy})_y + (\widetilde{\nabla} m_{yz})_z \right] (\widetilde{\nabla} W_i)_y \\ &+ \left[(\widetilde{\nabla} m_{xz})_x + (\widetilde{\nabla} m_{yz})_y + (\widetilde{\nabla} m_{zz})_z \right] (\widetilde{\nabla} W_i)_z \end{aligned} \tag{4.66}$$

The force-divergence curvature terms $\left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{n}}$ in the normal-force residual in (4.63) is evaluated in the same manner as (4.65) above.

Alternatively, equation (4.63) can be formulated with Lagrangian gradient definitions (w.r.t. undeformed geometry) resulting in the following force equilibrium residuals.

$$\begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} -\hat{\mathbf{w}}_{1_i} \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0 W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0\right) \cdot \hat{\mathbf{w}}_{1_i} W_i + \mathbf{q} \cdot \hat{\mathbf{w}}_{1_i} W_i + \mu\,(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{w}}_{1_i} W_i \\ -\hat{\mathbf{w}}_{2_i} \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0 W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0\right) \cdot \hat{\mathbf{w}}_{2_i} W_i + \mathbf{q} \cdot \hat{\mathbf{w}}_{2_i} W_i + \mu\,(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{w}}_{2_i} W_i \\ \left(\widetilde{\nabla}_0 \cdot \bar{\bar{\mathbf{m}}}\right) \cdot \widetilde{\nabla}_0 W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0\right) \cdot \hat{\mathbf{n}}\, W_i + \mathbf{q} \cdot \hat{\mathbf{n}}\, W_i + \mu\,(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{n}}\, W_i \end{Bmatrix} \mathrm{d}\mathcal{A}_0 \quad (4.67)$$

This formulation has the advantage of creating a residual Jacobian matrix that is *linear* with respect to the state vector variables since the undeformed geometry is a parameter (not in the state vector). This formulation is more in line with traditional finite element methods because it simplifies and expedites the Jacobian matrix solves and calculation of residual sensitivites to state variables.

4.4.5 Moment equilibrium residuals

The area-integral part of the moment equilibrium equation (3.29) for node i is evaluated in the local xyz axes, and then projected onto the local $\hat{\mathbf{w}}_{1_i}, \hat{\mathbf{w}}_{2_i}$ basis vectors of that node.

$$\begin{Bmatrix} \mathcal{R}_i^{m_1} \\ \mathcal{R}_i^{m_2} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} -\hat{\mathbf{w}}_{1_i} \cdot \bar{\bar{\mathbf{m}}} \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{m}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{w}}_{1_i} W_i - \hat{\mathbf{w}}_{1_i} \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{n}}\, W_i \\ -\hat{\mathbf{w}}_{2_i} \cdot \bar{\bar{\mathbf{m}}} \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{m}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{w}}_{2_i} W_i - \hat{\mathbf{w}}_{2_i} \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{n}}\, W_i \end{Bmatrix} \mathrm{d}\mathcal{A}_0 \quad (4.68)$$

Note that these are not used in the simpler HSM/KL formulation. Also note there is a Lagrangian formulation of this residual as well, similar to (4.67), in which the $\widetilde{\nabla}$ of equation (4.68) become $\widetilde{\nabla}_0$.

4.4.6 Local-basis constraint residuals

The basis vector definition equations (3.9) are integrated and summed over the elements as usual.

$$\begin{Bmatrix} \mathcal{R}_i^{\Lambda_1} \\ \mathcal{R}_i^{\Lambda_2} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} \frac{1}{J_0^2}\left[\hat{\mathbf{n}} \times (\partial_\xi \mathbf{r} \times \partial_\eta \mathbf{r}) \cdot \hat{\mathbf{w}}_{1_i} - \hat{\mathbf{n}}_0 \times (\partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0) \cdot \hat{\mathbf{w}}_{0_{1_i}}\right] \\ \frac{1}{J_0^2}\left[\hat{\mathbf{n}} \times (\partial_\xi \mathbf{r} \times \partial_\eta \mathbf{r}) \cdot \hat{\mathbf{w}}_{2_i} - \hat{\mathbf{n}}_0 \times (\partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0) \cdot \hat{\mathbf{w}}_{0_{2_i}}\right] \end{Bmatrix} W_i \; \mathrm{d}\mathcal{A}_0 \quad (4.69)$$

where again the sums are taken over the elements which contain node i, so that $\hat{\mathbf{n}}$ is tangent to the plane formed by the weighted average of the normals on the elements. The $1/J_0^2$ factor makes each element's contribution scale inversely with the element area, so that the

contributions of smaller elements are appropriately weighted more.

The third basis constraint (3.16) is similarly integrated and summed over the elements, with the $1/J_0^2$ factor again included to increase the weighting for small elements.

$$\tilde{\mathbf{s}} = \partial_\xi \mathbf{r} \tag{4.70}$$

$$\mathbf{s}_0 = \partial_\xi \mathbf{r}_0 \tag{4.71}$$

$$\mathbf{s} = \tilde{\mathbf{s}} - \bar{\bar{\boldsymbol{\varepsilon}}}\tilde{\mathbf{s}} \tag{4.72}$$

$$\theta = \text{atan2}\Big(\mathbf{s}\cdot\hat{\mathbf{w}}_{2_i} \ , \ \mathbf{s}\cdot\hat{\mathbf{w}}_{1_i} \Big) \tag{4.73}$$

$$\theta_0 = \text{atan2}\Big(\mathbf{s}_0\cdot\hat{\mathbf{w}}_{0_{2_i}} \ , \ \mathbf{s}_0\cdot\hat{\mathbf{w}}_{0_{1_i}} \Big) \tag{4.74}$$

$$\mathcal{R}_i^{\Lambda_3} \equiv \sum_{\text{elements}} \iint \frac{1}{J_0^2}\Big(\theta - \theta_0\Big) W_i \ \mathrm{d}\mathcal{A}_0 \tag{4.75}$$

4.5 Edge Boundary Conditions

For a well-posed shell elasticity problem, two boundary conditions (BCs) are required on all the edges of the shell. One BC is either on the position or on the edge force. The other BC is either on the edge angle or on the edge moment. Mixed boundary conditions which represent a compliant attachment can also be formulated, although these will not be considered here.

4.5.1 Boundary condition axes

Boundary conditions are specified in terms of force, moment, position, or angle components either along the global xyz axes, or alternatively along the *edge axes* $t\ell n$ which are defined by the edge basis vectors $\hat{\mathbf{t}}, \hat{\mathbf{l}}, \hat{\mathbf{n}}$. The $\hat{\mathbf{l}}$ vector is first defined either along $\partial_\xi \mathbf{r}$ or $\partial_\eta \mathbf{r}$, depending on which element edge is being considered, and using $\hat{\mathbf{n}}$ then defines $\hat{\mathbf{t}}$.

$$\hat{\mathbf{l}} = \begin{cases} \partial_\xi \mathbf{r}/|\partial_\xi \mathbf{r}| & , \ \text{along constant-}\eta \text{ edge} \\ -\partial_\eta \mathbf{r}/|\partial_\eta \mathbf{r}| & , \ \text{along constant-}\xi \text{ edge} \end{cases} \tag{4.76}$$

$$\hat{\mathbf{t}} = \hat{\mathbf{l}} \times \hat{\mathbf{n}} \tag{4.77}$$

Edge axes at a node are obtained via the undeformed geometry $\mathbf{r}_{o(u,v)}$ which was also used to define the nodal $\bar{\bar{\mathbf{e}}}_0$ basis via equations (4.1)–(4.3).

$$\hat{\mathbf{l}}_{0_i} = \begin{cases} (\ \partial_u \mathbf{r}_0/|\partial_u \mathbf{r}_0|)_i & , \ \text{along constant-}v \text{ edge} \\ (-\partial_v \mathbf{r}_0/|\partial_v \mathbf{r}_0|)_i & , \ \text{along constant-}u \text{ edge} \end{cases} \tag{4.78}$$

$$\hat{\mathbf{l}}_i = \Big(\hat{\mathbf{l}}_{0_i}\cdot\hat{\mathbf{e}}_{0_{1_i}}\Big)\,\hat{\mathbf{e}}_{1_i} + \Big(\hat{\mathbf{l}}_{0_i}\cdot\hat{\mathbf{e}}_{0_{2_i}}\Big)\,\hat{\mathbf{e}}_{2_i} \tag{4.79}$$

$$\hat{\mathbf{t}}_i = \hat{\mathbf{l}}_i \times \hat{\mathbf{n}}_i \tag{4.80}$$

All the boundary condition data which can be imposed in the shell edges is listed in the table below. For generality we specify and superimpose two separate loads: fixed-direction loads in the xyz axes, and shell-following loads in the $t\ell n$ axes, in the same manner as the overall surface load $\mathbf{q}$ was obtained from $\mathbf{q}_{xyz}$ and $\vec{q}$.

The edge position and orientation is specified by $\mathbf{r}_{\text{BC}}$ and $\hat{\mathbf{t}}_{\text{BC}}$, also listed in the table below. These will frequently be the same as $\mathbf{r}_0$ and $\hat{\mathbf{t}}_0$ of the undeformed geometry, but this is not required.

symbol	N	axes	description
$\mathbf{f}_{xyz_{\text{BC}}}$	3	xyz	imposed fixed-direction edge boundary force/length
$f_{t_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary t-force/length
$f_{\ell_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary ℓ-force/length
$f_{n_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary n-force/length
$\mathbf{m}_{xyz_{\text{BC}}}$	3	xyz	imposed fixed-direction edge boundary moment/length
$m_{tt_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary bending moment/length
$m_{\ell\ell_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary torsion moment/length
$m_{t\ell_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary cross-moment/length
$\mathbf{r}_{\text{BC}}$	3	xyz	imposed edge boundary position
$\mathbf{n}_{\text{BC}}$	3	xyz	imposed edge surface-normal direction

4.5.2 Force/displacement boundary conditions

Each shell edge point requires either a specified force/length BC or a specified displacement BC. Either one is imposed on the force-equilibrium residuals $\mathcal{R}_i^f$ defined by (4.63).

Edge force. The specified edge force/length is constructed as

$$\mathbf{f}_{\text{BC}} \;=\; \mathbf{f}_{xyz_{\text{BC}}} \;+\; f_{t_{\text{BC}}}\hat{\mathbf{t}} \;+\; f_{\ell_{\text{BC}}}\hat{\mathbf{l}} \;+\; f_{n_{\text{BC}}}\hat{\mathbf{n}} \tag{4.81}$$

which is then inserted into the edge integrals which appear in equations (3.22), (3.27), and (3.29). After projection to the in-plane and normal directions, this produces the additional residual changes

$$\begin{Bmatrix} \Delta\mathcal{R}_i^{f_1} \\ \Delta\mathcal{R}_i^{f_2} \\ \Delta\mathcal{R}_i^{f_n} \end{Bmatrix} \equiv \sum_{\text{edges}} \oint \begin{bmatrix} - & \hat{\mathbf{w}}_{1_i} & - \\ - & \hat{\mathbf{w}}_{2_i} & - \\ - & \hat{\mathbf{n}} & - \end{bmatrix} \begin{Bmatrix} | \\ \mathbf{f}_{\text{BC}} \\ | \end{Bmatrix} W_i \, \mathrm{d}\ell \tag{4.82}$$

which are evaluated using Gaussian integration (4.60), and $\mathbf{f}_{\text{BC}(\xi)}$ along the edge is defined by linear interpolation from the two nodes using the $\tilde{N}_{j(\xi)}$ functions (4.40). The result is

then added to the force equilibrium residuals $\mathcal{R}_i^f$ defined by (4.63).

$$\begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} \leftarrow \begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} + \begin{Bmatrix} \Delta\mathcal{R}_i^{f_1} \\ \Delta\mathcal{R}_i^{f_2} \\ \Delta\mathcal{R}_i^{f_n} \end{Bmatrix} \tag{4.83}$$

Edge position. The specified position of the edge at node i is imposed by replacing the force equilibrium residuals $\mathcal{R}_i^f$ with the following position-constraint residuals $\mathcal{R}_i^r$.

$$\begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} \leftarrow \begin{Bmatrix} \mathcal{R}_i^{r_1} \\ \mathcal{R}_i^{r_2} \\ \mathcal{R}_i^{r_n} \end{Bmatrix} \equiv \begin{bmatrix} - & \hat{\mathbf{e}}_{1_i} & - \\ - & \hat{\mathbf{e}}_{2_i} & - \\ - & \hat{\mathbf{n}}_i & - \end{bmatrix} \begin{Bmatrix} | \\ \mathbf{r}_i - \mathbf{r}_{\mathrm{BC}_i} \\ | \end{Bmatrix} \tag{4.84}$$

The projection onto the residual-node basis vectors $\bar{\bar{\mathbf{e}}}_i$ in (4.84) is not strictly necessary, although it would likely be convenient if compliant-attachment BC were to be imposed, presumably in terms of these basis vectors.

Symmetry plane force/displacement. To impose force/displacement boundary conditions at a symmetry plane, we must first project the force equilibrium residuals into the edge $t\ell n$ axes,

$$\begin{Bmatrix} \mathcal{R}_i^{f_t} \\ \mathcal{R}_i^{f_\ell} \end{Bmatrix} = \begin{bmatrix} - & \hat{\mathbf{t}}_i & - \\ - & \hat{\mathbf{l}}_i & - \end{bmatrix} \begin{bmatrix} | & | \\ \hat{\mathbf{e}}_{1_i} & \hat{\mathbf{e}}_{2_i} \\ | & | \end{bmatrix} \begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \end{Bmatrix} \tag{4.85}$$

with the $\mathcal{R}_i^{f_n}$ residual left unchanged. The t-force residual is then replaced with the position constraint residual

$$\mathcal{R}_i^{f_t} \leftarrow \mathcal{R}_i^{r_t} \equiv (\mathbf{r}_i - \mathbf{r}_S) \cdot \mathbf{n}_S \tag{4.86}$$

where the specified $\mathbf{r}_S$ is a point on the symmetry plane and the specified plane-normal vector $\mathbf{n}_S$ gives the plane's orientation. The remaining $\mathcal{R}_i^{f_\ell}$ and $\mathcal{R}_i^{f_n}$ residuals are left unchanged, which is equivalent to specifying that the resulting edge force has zero components within the plane.

4.5.3 Moment/normal-vector boundary conditions

Each shell edge point requires either a specified moment/length or a specified shell normal vector (or the equivalent surfaces angles). Either one is imposed on the curvature-constraint residuals $\mathcal{R}_i^\kappa$ defined by (4.62), which are first put into the $t\ell$ edge axes, along with the

edge-node moments $\bar{\bar{m}}$.

$$\begin{Bmatrix} \mathcal{R}_i^{\kappa_{tt}} \\ \mathcal{R}_i^{\kappa_{\ell\ell}} \\ \mathcal{R}_i^{\kappa_{t\ell}} \end{Bmatrix} = \bar{\bar{\mathbf{C}}}_i \begin{Bmatrix} \mathcal{R}_i^{\kappa_{11}} \\ \mathcal{R}_i^{\kappa_{22}} \\ \mathcal{R}_i^{\kappa_{12}} \end{Bmatrix} \tag{4.87}$$

$$\begin{Bmatrix} m_{tt} \\ m_{\ell\ell} \\ m_{t\ell} \end{Bmatrix}_i = \bar{\bar{\mathbf{C}}}_i \begin{Bmatrix} m_{11} \\ m_{22} \\ m_{12} \end{Bmatrix}_i \tag{4.88}$$

$$\bar{\bar{\mathbf{C}}} \equiv \begin{bmatrix} (\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{t}})^2 & (\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{t}})^2 & 2\,(\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{t}})(\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{t}}) \\ (\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{l}})^2 & (\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{l}})^2 & 2\,(\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{l}})(\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{l}}) \\ (\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{t}})(\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{l}}) & (\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{t}})(\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{l}}) & (\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{t}})(\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{l}}) + (\hat{\mathbf{e}}_1 \cdot \hat{\mathbf{l}})(\hat{\mathbf{e}}_2 \cdot \hat{\mathbf{t}}) \end{bmatrix} \tag{4.89}$$

Edge moment. The specified edge moment/length components are imposed as residuals which replace the corresponding $\mathcal{R}_i^{\kappa}$ residuals.

$$\mathcal{R}_i^{\kappa_{tt}} \leftarrow \mathcal{R}_i^{m_{tt}} \equiv m_{tt_i} - (\mathbf{m}_{xyz_{\mathrm{BC}_i}} \cdot \hat{\mathbf{t}}_i + m_{tt_{\mathrm{BC}_i}}) \tag{4.90}$$

$$\mathcal{R}_i^{\kappa_{\ell\ell}} \leftarrow \mathcal{R}_i^{m_{\ell\ell}} \equiv m_{\ell\ell_i} - (\mathbf{m}_{xyz_{\mathrm{BC}_i}} \cdot \hat{\mathbf{l}}_i + m_{\ell\ell_{\mathrm{BC}_i}}) \tag{4.91}$$

$$\mathcal{R}_i^{\kappa_{t\ell}} \leftarrow \mathcal{R}_i^{m_{t\ell}} \equiv m_{t\ell_i} - m_{t\ell_{\mathrm{BC}_i}} \qquad \text{(corner node only)} \tag{4.92}$$

As indicated, (4.92) is imposed only at a free salient corner, typically with $m_{t\ell_{\mathrm{BC}}} = 0$.

If the HSM/RM formulation is being used, the edge-moment integral term in equation (3.29) must also be evaluated and added to the corresponding discrete residuals (4.68). These moment edge integrals have the same form as the edge force integrals (4.82), except that only the 1,2 moment components are imposed.

$$\mathbf{m}_{\mathrm{BC}} = \mathbf{m}_{xyz_{\mathrm{BC}}} + m_{tt_{\mathrm{BC}}}\hat{\mathbf{t}} + m_{\ell\ell_{\mathrm{BC}}}\hat{\mathbf{l}} \tag{4.93}$$

$$\begin{Bmatrix} \Delta\mathcal{R}_i^{m_1} \\ \Delta\mathcal{R}_i^{m_2} \end{Bmatrix} \equiv \sum_{\text{edges}} \oint \begin{bmatrix} - & \hat{\mathbf{w}}_{1_i} & - \\ - & \hat{\mathbf{w}}_{2_i} & - \end{bmatrix} \begin{Bmatrix} | \\ \mathbf{m}_{\mathrm{BC}} \\ | \end{Bmatrix} W_i \, \mathrm{d}\ell \tag{4.94}$$

$$\begin{Bmatrix} \mathcal{R}_i^{m_1} \\ \mathcal{R}_i^{m_2} \end{Bmatrix} \leftarrow \begin{Bmatrix} \mathcal{R}_i^{m_1} \\ \mathcal{R}_i^{m_2} \end{Bmatrix} + \begin{Bmatrix} \Delta\mathcal{R}_i^{m_1} \\ \Delta\mathcal{R}_i^{m_2} \end{Bmatrix} \tag{4.95}$$

Edge normal. The shell surface angles are imposed by requiring the shell normal vector to be parallel to the specified normal vector. The corresponding residual replaces the shell

edge-normal and edge-tangent curvature constraint residuals.

$$\begin{Bmatrix} \mathcal{R}_i^{\kappa_{tt}} \\ \mathcal{R}_i^{\kappa_{t\ell}} \end{Bmatrix} \leftarrow \begin{Bmatrix} \mathcal{R}_i^{\vartheta_t} \\ \mathcal{R}_i^{\vartheta_\ell} \end{Bmatrix} \equiv \begin{Bmatrix} \hat{\mathbf{n}}_i \times \mathbf{n}_{\mathrm{BC}_i} \cdot \hat{\mathbf{l}}_i \\ \hat{\mathbf{n}}_i \times \mathbf{n}_{\mathrm{BC}_i} \cdot \hat{\mathbf{t}}_i \end{Bmatrix} \tag{4.96}$$

Symmetry plane moment/normal-vector. At a symmetry plane, the tangential curvature-constraint residual is replaced with the requirement that the $\hat{\mathbf{n}}_i$ vector is normal to the symmetry-plane normal vector.

$$\mathcal{R}_i^{\kappa_{tt}} \leftarrow \mathcal{R}_i^{\vartheta_{S_t}} \equiv \hat{\mathbf{n}}_i \cdot \mathbf{n}_S \tag{4.97}$$

4.6 Edge Joining Conditions

The present HSM allows a shell to be built up of multiple pieces which are joined at edges. Assuming the edge joint is rigid, we must enforce matching conditions on the primary variables on each pair of adjacent nodes i and i' on the two shell pieces. After contributions from all the elements in a shell piece are summed onto the nodes, the residuals at the i and i' nodes are modified as follows.

4.6.1 Force and position matching

The force matching condition is the requirement that the internal loads cancel. This is performed by adding the force residuals from the two nodes, after rotating the node i' residuals into the $12n$ axes of node i.

$$\begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} \leftarrow \begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \\ \mathcal{R}_i^{f_n} \end{Bmatrix} + \begin{bmatrix} - & \hat{\mathbf{e}}_{1_i} & - \\ - & \hat{\mathbf{e}}_{2_i} & - \\ - & \hat{\mathbf{n}}_i & - \end{bmatrix} \begin{bmatrix} | & | & | \\ \hat{\mathbf{e}}_{1_i'} & \hat{\mathbf{e}}_{2_i'} & \hat{\mathbf{n}}_{i'} \\ | & | & | \end{bmatrix} \begin{Bmatrix} \mathcal{R}_{i'}^{f_1} \\ \mathcal{R}_{i'}^{f_2} \\ \mathcal{R}_{i'}^{f_n} \end{Bmatrix} \tag{4.98}$$

The added residuals replace those for node i as indicated. The force residuals for node i' are then cleared and replaced with position continuity constraints on the node coordinates. These have the same form as the specified-position constraints (4.84).

$$\begin{Bmatrix} \mathcal{R}_{i'}^{f_1} \\ \mathcal{R}_{i'}^{f_2} \\ \mathcal{R}_{i'}^{f_n} \end{Bmatrix} \leftarrow \begin{Bmatrix} \mathcal{R}_{i'}^{\Delta r_1} \\ \mathcal{R}_{i'}^{\Delta r_2} \\ \mathcal{R}_{i'}^{\Delta r_n} \end{Bmatrix} \equiv \begin{bmatrix} - & \hat{\mathbf{e}}_{1_{i'}} & - \\ - & \hat{\mathbf{e}}_{2_{i'}} & - \\ - & \hat{\mathbf{n}}_{i'} & - \end{bmatrix} \begin{Bmatrix} | \\ \mathbf{r}_i - \mathbf{r}_{i'} \\ | \end{Bmatrix} \tag{4.99}$$

4.6.2 Angle and moment matching

Before the curvature-definition residuals for nodes i and i' are combined, they are first put into their respective $t\ell$ axes using (4.87).

$$\begin{Bmatrix} \mathcal{R}_i^{\kappa_{tt}} \\ \mathcal{R}_i^{\kappa_{\ell\ell}} \\ \mathcal{R}_i^{\kappa_{t\ell}} \end{Bmatrix} = \bar{\bar{\mathbf{C}}}_i \begin{Bmatrix} \mathcal{R}_i^{\kappa_{11}} \\ \mathcal{R}_i^{\kappa_{22}} \\ \mathcal{R}_i^{\kappa_{12}} \end{Bmatrix} \tag{4.100}$$

$$\begin{Bmatrix} \mathcal{R}_{i'}^{\kappa_{tt}} \\ \mathcal{R}_{i'}^{\kappa_{\ell\ell}} \\ \mathcal{R}_{i'}^{\kappa_{t\ell}} \end{Bmatrix} = \bar{\bar{\mathbf{C}}}_{i'} \begin{Bmatrix} \mathcal{R}_{i'}^{\kappa_{11}} \\ \mathcal{R}_{i'}^{\kappa_{22}} \\ \mathcal{R}_{i'}^{\kappa_{12}} \end{Bmatrix} \tag{4.101}$$

The κ_{tt} curvature-definition residuals for nodes i and i' are then subtracted to form the new residual which forces bending-moment continuity across the edge.

$$\mathcal{R}_i^{\kappa_{tt}} \leftarrow \mathcal{R}^{\Delta m_{tt}} \equiv \mathcal{R}_i^{\kappa_{tt}} - \mathcal{R}_{i'}^{\kappa_{tt}} \tag{4.102}$$

The corresponding residual for node i' is then replaced by the constraint that the relative angle between $\hat{\mathbf{n}}_i$ and $\hat{\mathbf{n}}_{i'}$ is the same for the deformed and undeformed geometries.

$$\vartheta_0 = \text{atan2}\left(\hat{\mathbf{n}}_{0_i} \times \hat{\mathbf{n}}_{0_{i'}} \cdot \hat{\mathbf{l}}_{0_i} \;,\; \hat{\mathbf{n}}_{0_i} \cdot \hat{\mathbf{n}}_{0_{i'}} \right) \tag{4.103}$$

$$\vartheta = \text{atan2}\left(\hat{\mathbf{n}}_i \times \hat{\mathbf{n}}_{i'} \cdot \hat{\mathbf{l}}_i \;,\; \hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_{i'} \right) \tag{4.104}$$

$$\mathcal{R}_{i'}^{\kappa_{tt}} \leftarrow \mathcal{R}^{\Delta\vartheta} \equiv \vartheta - \vartheta_0 \tag{4.105}$$

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 5

HSM2D: Simplifications in Two-Dimensions

The Hybrid Shell Model of Chapter 3 is a method for modeling the deformations of arbitrarily-shaped two-dimensional manifolds under the influence of arbitrary loading in three dimensions. This version, "HSM2D", has been simplified for the case of a flat plate subject to in-plane loading only. Thus the undeformed geometry $\mathbf{r}_0$, deformed geometry $\mathbf{r}$, and loading $\mathbf{q}$ are only defined in the xy coordinate system. The normal vector remains the same, $\hat{\mathbf{n}}_0 = \hat{\mathbf{n}} = \{0 \;\; 0 \;\; 1\}^T$ before and after deformation. HSM2D further makes the assumption that the parametrization of the undeformed geometry is coaxial with the Cartesian unit vectors, i.e.

$$\hat{\boldsymbol{\xi}}_0 = \frac{\partial_\xi \mathbf{r}_0}{|\partial_\xi \mathbf{r}_0|} = \hat{\mathbf{x}} = \{1 \;\; 0 \;\; 0\}^T \qquad \hat{\boldsymbol{\eta}}_0 = \frac{\partial_\eta \mathbf{r}_0}{|\partial_\eta \mathbf{r}_0|} = \hat{\mathbf{y}} = \{0 \;\; 1 \;\; 0\}^T \tag{5.1}$$

The 2D assumptions eliminate the possibility of deflections normal to the reference plane, so the HSM/KL and HSM/M formulations reduce to a single model in HSM2D.

This chapter will present HSM2D in an abbreviated format compared to the earlier presentation of HSM. The inclusion of HSM2D in this thesis is primarily intended to streamline the computational implementation process by providing explicit equations for the two-dimensional simplifications of HSM.

5.1 Local Basis

This section provides two-dimensional simplifications for the definition of the local basis (Section 3.1.1 for HSM).

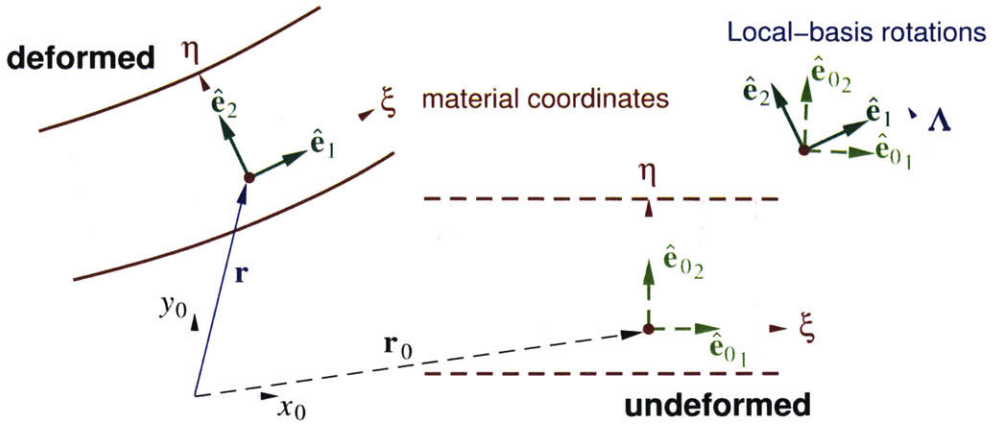


Figure 5-1: Bird's eye view of plate showing position $\mathbf{r}$ and local basis vectors $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$, in local material coordinates ξ, η.

5.1.1 Undeformed Basis

Every point on the plate has associated with it a local orthogonal unit-vector basis. The two-dimensional construct reduces the matrix from a 3×3 matrix in HSM to a 2×2 matrix in HSM2D. As previously mentioned, the undeformed basis $\bar{\bar{\mathbf{e}}}_0$ is defined to follow the cartesian xyz coordinates.

$$\bar{\bar{\mathbf{e}}}_{0(u,v)} = \bar{\bar{\mathbf{e}}}_{0(\mathbf{\Lambda}_{0(u,v)})} \equiv \begin{bmatrix} e_{0_{1x}} & e_{0_{2x}} \\ e_{0_{1y}} & e_{0_{2y}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.2}$$

5.1.2 Deformed Basis

The deformed local basis $\bar{\bar{\mathbf{e}}}$ is defined in terms of the unit-quaternion logarithm $\mathbf{\Lambda}$. Both $\mathbf{r}$ and $\mathbf{\Lambda}$ are treated as primary unknown variables.

$$\bar{\bar{\mathbf{e}}}_{(u,v)} = \bar{\bar{\mathbf{e}}}_{(\mathbf{\Lambda}_{(u,v)})} \equiv \begin{bmatrix} e_{1x} & e_{2x} \\ e_{1y} & e_{2y} \end{bmatrix} \tag{5.3}$$

where $\hat{\mathbf{e}}_\alpha = \{\, e_{\alpha x} \;,\; e_{\alpha y} \,\}^T$, and the vectors $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$ are tangent and $\hat{\mathbf{n}} = \{\, 0 \;\; 0 \;\; 1 \,\}^T$ is normal to the reference surface.

The local basis $\bar{\bar{\mathbf{e}}}$ is defined from a quaternion-based rotation about a reference vector. In this 2D implementation, the reference vector is the surface normal vector

$$\hat{\mathbf{v}}_{\text{ref}} = \hat{\mathbf{n}} = \{\, 0 \;\; 0 \;\; 1 \,\}^T \tag{5.4}$$

For a quaternion $\mathbf{p}$ representing a rotation through angle θ about a unit reference vector $\hat{\mathbf{v}}_{\text{ref}}$, the unit-quaternion logarithm $\mathbf{\Lambda}$ is defined as below, using equation (5.4), to simplify the 2D implementation.

$$\mathbf{\Lambda} = \begin{Bmatrix} \Lambda_0 \\ \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{Bmatrix} \equiv \left\{ \begin{array}{c} 0 \\ \hline | \\ \hat{\mathbf{v}}_{\text{ref}}\ \theta/2 \\ | \end{array} \right\} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \theta/2 \end{Bmatrix} \tag{5.5}$$

The quaternion itself is $\mathbf{p} = \exp(\mathbf{\Lambda})$ and is defined as follows.

$$\mathbf{p} = \begin{Bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{Bmatrix} \equiv \left\{ \begin{array}{c} \cos\theta/2 \\ \hline | \\ \hat{\mathbf{v}}_{\text{ref}}\ \sin\theta/2 \\ | \end{array} \right\} = \begin{Bmatrix} \cos\theta/2 \\ 0 \\ 0 \\ \sin\theta/2 \end{Bmatrix} = \begin{Bmatrix} \cos\Lambda_3 \\ 0 \\ 0 \\ \sin\Lambda_3 \end{Bmatrix} \tag{5.6}$$

The explicit definition of $\bar{\bar{\mathbf{e}}}_{(\mathbf{\Lambda})}$ is via this actual quaternion $(p_0, p_1, p_2, p_3) = \exp(\mathbf{\Lambda})$, as follows.

$$\bar{\bar{\mathbf{e}}}_{(\mathbf{\Lambda})} = \begin{bmatrix} 1-2p_3^2 & -2p_0p_3 & 0 \\ 2p_0p_3 & 1-2p_3^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.7}$$

Note the four (three independent) components of the unit-quaternion logarithm reduce to a single parameter θ which defines the non-zero components (p_0 and p_3) in two-dimensions. This is the in-plane rotation of the local-basis. It does not encounter the gimbal-lock issue of Euler angles in three-dimensions because the rotation axis is the fixed normal vector $\hat{\mathbf{n}}$. In practice, the log-quaternion component Λ_3 is used as the orientation primary unknown. This is because a log-quaternion may be interpolated via the standard bilinear basis functions whereas an angle must be interpolated via spherical interpolation. As seen in equation (5.5), Λ_3 is just twice the rotation angle θ.

Further note there is no need for the non-degenerate local basis of HSM; the $\bar{\bar{\mathbf{e}}}$ basis is sufficient as a definition for the orientation unit-quaternion logarithms. A two-dimensional shape cannot be a closed manifold, and thus the "Hairy Ball" Theorem does not apply. A non-degenerate geometrical parametrization will always exist for a flat plate so there is no need to construct an alternate non-degenerate coordinate system.

5.2 In-Plane Force Equilibrium

This section provides two-dimensional simplifications for construction of the in-plane force equilibrium equations (Section 3.2.3 for HSM). As there is no normal force component and no bending moments in the planar case, the only equilibrium equations to consider describe the in-plane force equilibrium.

The starting point for the two-dimensional force balance is the same as for the 3D case: the 3D stress-equilibrium differential equation (3.17) for a material with stress tensor $\bar{\bar{\boldsymbol{\sigma}}}$, density ρ, acceleration $\mathbf{a}$, and gravity $\mathbf{g}$.

$$\nabla \cdot \bar{\bar{\boldsymbol{\sigma}}} + \rho(\mathbf{g}-\mathbf{a}) = \mathbf{0} \tag{5.8}$$

In HSM2D, we immediately switch to the Lagrangian formulation of the stress equilibrium equations, for the implementation-driven reasons in the explanation of equation (4.67). Thus the force equilibrium equation becomes:

$$\nabla_0 \cdot \bar{\bar{\boldsymbol{\sigma}}} + \rho(\mathbf{g}-\mathbf{a}) = \mathbf{0} \tag{5.9}$$

The integral shell in-plane force-equilibrium equations are obtained by forming $[\,\text{eq.}(5.9)\,]\cdot\hat{\mathbf{e}}_1$ and $[\,\text{eq.}(5.9)\,]\cdot\hat{\mathbf{e}}_2$, and expanding the divergence term to give

$$\nabla_0 \cdot (\bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{e}}) - (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla_0) \cdot \hat{\mathbf{e}} + \rho(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{e}} = 0 \tag{5.10}$$

where $\hat{\mathbf{e}}$ denotes either in-plane vector $\hat{\mathbf{e}}_1$ or $\hat{\mathbf{e}}_2$. We next multiply (5.10) by a weighting function $W_{(\xi,\eta)}$ and integrate over the shell volume. We will define W to be nonzero only over a finite element, so the integration is in effect performed only over this element.

$$\iiint \Big\{ \nabla_0 \cdot (\bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{e}}) - (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla_0) \cdot \hat{\mathbf{e}} + \rho(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{e}} \Big\} W \, \mathrm{d}\mathcal{V} = 0$$

$$\iiint \Big\{ \nabla_0 \cdot (\bar{\bar{\boldsymbol{\sigma}}} \cdot \hat{\mathbf{e}} W) - \hat{\mathbf{e}} \cdot \bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla_0 W - (\bar{\bar{\boldsymbol{\sigma}}} \cdot \nabla_0) \cdot \hat{\mathbf{e}} W + \rho(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{e}} W \Big\} \, \mathrm{d}\mathcal{V} = 0 \tag{5.11}$$

Next, the first pure divergence volume integral is replaced by area integrals over the perimeter surface with normal $\hat{\mathbf{t}}$ and area elements $\mathrm{d}n\mathrm{d}\ell$, and over the top/bottom surfaces with normals $\pm\hat{\mathbf{n}}$ and area elements $\mathrm{d}\mathcal{A}$. For the remaining volume integrals the volume element is written as $\mathrm{d}\mathcal{V} = \mathrm{d}n\,\mathrm{d}\mathcal{A}$. Equation (5.11) then becomes

$$\oint \hat{\mathbf{e}} \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} \, W \, \mathrm{d}\ell + \iint \Big[-\hat{\mathbf{e}} \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0 W - \Big(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}_0\Big) \cdot \hat{\mathbf{e}} W + \mathbf{q} \cdot \hat{\mathbf{e}} W + \mu(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{e}} W \Big] \, \mathrm{d}\mathcal{A} = 0 \tag{5.12}$$

where the integrations $\int \mathrm{d}n$ across the shell thickness have been carried out in the replacement of ρ and $\bar{\bar{\boldsymbol{\sigma}}}$ with μ and $\bar{\bar{\mathbf{f}}}$. The fact that $W_{(\xi,\eta)}$ is defined to not vary in n allows it to be put outside the $\int \mathrm{d}n$ thickness integrals.

5.3 Constitutive Relations

This section provides two-dimensional simplifications of the stress-strain relationship (Section 3.3 for HSM).

5.3.1 Force Resultants, Shell Lumping

The 12 components of the nodal strain tensor $\bar{\bar{\varepsilon}}^{\mathsf{e}}_j$, denoted $\varepsilon^{\mathsf{e}}_{\alpha\beta(\mathbf{f})}$ in (6.20), are computed from $\bar{\bar{f}}_j$ using the compliance matrix equation (3.36) with two simplifications. The first is that the reference surface lies in the middle of the shell thickness, so $\bar{\bar{B}} = 0$ from symmetry. The second is that there is no normal component of stress, strain, or stiffness in 2D, so $\bar{\bar{D}} = 0$. Therefore, $\bar{\bar{A}}^{\mathsf{e}}$ is the only matrix that effects the extensional constitutive relation and equation (3.50) reduces to the following.

$$\begin{Bmatrix} \varepsilon^{\mathsf{e}}_{11} \\ \varepsilon^{\mathsf{e}}_{22} \\ \varepsilon^{\mathsf{e}}_{12} \end{Bmatrix}_j = \begin{bmatrix} \\ \bar{\bar{A}}^{\mathsf{e}} \\ \\ \end{bmatrix}^{-1}_j \begin{Bmatrix} f^{\mathsf{e}}_{11} \\ f^{\mathsf{e}}_{22} \\ f^{\mathsf{e}}_{12} \end{Bmatrix}_j \tag{5.13}$$

The stiffness submatrix, $\bar{\bar{A}}^{\mathsf{e}}$, is defined by the following weighted integral of the stiffness tensor $\bar{\bar{c}}^{\mathsf{e}}$ over the shell thickness, which captures the overall lumped properties of the shell cross section. The compliance matrix is the inverse of the stiffness matrix. Note in the following equations, the superscript $(\,)^{\mathsf{e}}$ has been dropped from the components for brevity, but the tensors are defined in the $12n$ basis.

$$\bar{\bar{A}}^{\mathsf{e}} = \begin{bmatrix} A_{11} & A_{12} & A_{16} \\ \cdot & A_{22} & A_{26} \\ \cdot & \cdot & A_{66} \end{bmatrix} \equiv \int \begin{bmatrix} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{bmatrix} \mathrm{d}n \tag{5.14}$$

For an isotropic shell material, the modulus E and Poisson's ratio ν are sufficient to give all the components of $\bar{\bar{c}}$ as follows.

$$\begin{bmatrix} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{bmatrix} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \cdot & 1 & 0 \\ \cdot & \cdot & 1-\nu \end{bmatrix} \tag{5.15}$$

For an orthotropic material such as a balanced and symmetric composite laminate, the

components of the un-lumped stiffness tensor are

$$\begin{bmatrix} c_{11} & c_{12} & c_{16} \\ \cdot & c_{22} & c_{26} \\ \cdot & \cdot & c_{66} \end{bmatrix} = \frac{1}{1-\nu^2} \begin{bmatrix} E_1 & E_2\,\nu_{12} & 0 \\ \cdot & E_2 & 0 \\ \cdot & \cdot & 2G_{12}(1-\nu^2) \end{bmatrix} \quad , \qquad \nu^2 \equiv \nu_{12}^2 E_2/E_1 \tag{5.16}$$

For general composite layering the overall thickness-averaged properties can be calculated from laminate theory (see Appendix D for more information). Specifically, unbalanced laminates will also have nonzero c_{16} and c_{26} constants which represent extension/shear coupling.

5.3.2 Stress Basis Conversions

The force and moment resultants which are defined in the local 12 axes must be put in the global xy axes for application in the equilibrium equations. This is performed using the xy components of the local $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$ basis vectors.

$$\begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix} = \begin{bmatrix} & \\ & \bar{\bar{\mathbf{e}}} \\ & \end{bmatrix} \begin{bmatrix} f^{\mathbf{e}}_{11} & f^{\mathbf{e}}_{12} \\ f^{\mathbf{e}}_{12} & f^{\mathbf{e}}_{22} \end{bmatrix} \begin{bmatrix} & \\ & \bar{\bar{\mathbf{e}}}^T \\ & \end{bmatrix} \tag{5.17}$$

This can be recast into a more convenient form.

$$\begin{Bmatrix} f_{xx} \\ f_{yy} \\ f_{xy} \end{Bmatrix} = \begin{bmatrix} e_{1x}^2 & e_{2x}^2 & 2\,e_{1x}\,e_{2x} \\ e_{1y}^2 & e_{2y}^2 & 2\,e_{1y}\,e_{2y} \\ e_{1x}\,e_{1y} & e_{2x}\,e_{2y} & e_{1x}\,e_{2y} + e_{1y}\,e_{2x} \end{bmatrix} \begin{Bmatrix} f^{\mathbf{e}}_{11} \\ f^{\mathbf{e}}_{22} \\ f^{\mathbf{e}}_{12} \end{Bmatrix} \tag{5.18}$$

5.4 Compatibility Relations

This section provides two-dimensional simplifications for the definition of the strain-displacement relation (Section 3.4 for HSM).

HSM2D uses the Almansi-Lagrange strain tensor which is formulated in Cartesian coordinates as

$$\bar{\bar{\boldsymbol{\varepsilon}}}^{(\mathbf{r})} = \frac{1}{2}\Big[(\widetilde{\nabla}_0\mathbf{r})(\widetilde{\nabla}_0\mathbf{r})^T - (\widetilde{\nabla}_0\mathbf{r}_0)(\widetilde{\nabla}_0\mathbf{r}_0)^T\Big] \tag{5.19}$$

$$= \frac{1}{2}\Big[(\widetilde{\nabla}_0\mathbf{r})(\widetilde{\nabla}_0\mathbf{r})^T - \bar{\bar{\mathbf{I}}}_2\Big] \tag{5.20}$$

where $\bar{\bar{\mathbf{I}}}_2$ is the 2×2 identity matrix and the surface gradient is defined as

$$\widetilde{\nabla}_0\mathbf{r} \equiv \begin{bmatrix} \partial_{x_0}x & \partial_{x_0}y \\ \partial_{y_0}x & \partial_{y_0}y \end{bmatrix} \tag{5.21}$$

The undeformed geometry Lagrangian surface gradient is the identity matrix

$$\widetilde{\nabla}_0 \mathbf{r}_0 \equiv \begin{bmatrix} \partial_{x_0} x_0 & \partial_{x_0} y_0 \\ \partial_{y_0} x_0 & \partial_{y_0} y_0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.22}$$

The Almansi-Lagrange strain tensor is chosen as to be consistent with the Langrangian formulation of the stress equilibrium equation (5.9) for the sake of deriving analytic solutions to be used in model verification. Such analytic solutions are derived in Section 7.4.

The Cartesian stress tensor is then rotated into the local $12n$ axes with the following transformation.

$$\begin{bmatrix} \varepsilon^{\mathrm{e}}_{11} & \varepsilon^{\mathrm{e}}_{12} \\ \varepsilon^{\mathrm{e}}_{12} & \varepsilon^{\mathrm{e}}_{22} \end{bmatrix} = \begin{bmatrix} \\ \bar{\bar{\mathbf{e}}}^T \\ \\ \end{bmatrix} \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} \\ \varepsilon_{xy} & \varepsilon_{yy} \end{bmatrix} \begin{bmatrix} \\ \bar{\bar{\mathbf{e}}} \\ \\ \end{bmatrix} \tag{5.23}$$

5.5 Edge Boundary Conditions

This section provides two-dimensional simplifications of boundary condition formulation (Section 4.5 for HSM).

For a well-posed shell elasticity problem in 2D, one boundary condition (BC) is required on all the edges of the shell. This specifies either the position or on the edge force. Mixed boundary conditions which represent a compliant attachment can also be formulated, although these will not be considered here.

5.5.1 Boundary condition axes

Boundary conditions are specified in terms of force or position components either along the global xyz axes, or alternatively along the *edge axes* $t\ell n$ which are defined by the edge basis vectors $\hat{\mathbf{t}}, \hat{\mathbf{l}}, \hat{\mathbf{n}}$. The $\hat{\mathbf{l}}$ vector is first defined either along $\partial_\xi \mathbf{r}$ or $\partial_\eta \mathbf{r}$, depending on which element edge is being considered, and using $\hat{\mathbf{n}}$ then defines $\hat{\mathbf{t}}$.

$$\hat{\mathbf{l}} = \begin{cases} \partial_\xi \mathbf{r}/|\partial_\xi \mathbf{r}| & , \text{ along constant-}\eta \text{ edge} \\ -\partial_\eta \mathbf{r}/|\partial_\eta \mathbf{r}| & , \text{ along constant-}\xi \text{ edge} \end{cases} \tag{5.24}$$

$$\hat{\mathbf{t}} = \hat{\mathbf{l}} \times \hat{\mathbf{n}} \tag{5.25}$$

All the boundary condition data which can be imposed in the shell edges is listed in the table below. For generality we specify and superimpose two separate loads: fixed-direction loads in the xyz axes, and shell-following loads in the $t\ell n$ axes, in the same manner as the overall surface load $\mathbf{q}$ was obtained from $\mathbf{q}_{xyz}$ and $\vec{q}$.

The edge position and orientation is specified by $\mathbf{r}_{\mathrm{BC}}$ and $\hat{\mathbf{t}}_{\mathrm{BC}}$, also listed in the table below.

These will frequently be the same as $\mathbf{r}_0$ and $\hat{\mathbf{t}}_0$ of the undeformed geometry, but this is not required.

symbol	num.	axes	description
$\mathbf{f}_{xy_{\text{BC}}}$	2	xy	imposed fixed-direction edge boundary force/length
$f_{t_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary t-force/length
$f_{\ell_{\text{BC}}}$	1	$t\ell n$	imposed shell-following edge boundary ℓ-force/length
$\mathbf{r}_{\text{BC}}$	3	xy	imposed edge boundary position

5.5.2 Displacement BCs

The specified position $\mathbf{r}$ of an edge Γ_{r} is imposed via

$$\mathbf{r} = \mathbf{r}_{\text{BC}} \qquad \text{on } \Gamma_{\text{r}} \tag{5.26}$$

where $\mathbf{r}_{\text{BC}}$ are Cartesian vectors specified as Dirichlet data.

5.5.3 Force BCs

The specified lumped stress $\bar{\bar{\mathbf{f}}}$ of an edge Γ_{f} is imposed via

$$\bar{\bar{\mathbf{e}}}^t \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} = \vec{f}^{\mathbf{e}}_{\text{BC}} \qquad \text{on } \Gamma_{\text{f}} \tag{5.27}$$

where $\vec{f}^{\mathbf{e}}_{\text{BC}}$ are vectors in the local $12n$ frame specified as Dirichlet data. It is noted that using the definition $\bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} = \mathbf{f}_{\text{BC}}$ in which the force BCs are specified as a Cartesian vector $\mathbf{f}_{\text{BC}}$ results in non-sensical BC weights. The force BC *must* be specified in the local frame to produce well defined adjoint BCs.

Chapter 6

HSM2D Finite-Element Solution

6.1 Nodal Data

This section provides the two-dimensional simplifications for the Finite-Element Solution of HSM (Section 4.1).

The following data is defined at each element corner node j. Bilinear interpolation to the element interior then makes these quantities functions of the (ξ, η) element coordinates. The vector and tensor quantities are defined either in the global xy axes (note $z = 0$), or in the node's 12 axes (note $n = 0$) along the node's $\bar{\bar{\mathbf{e}}}_j = [\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2]_j$ basis vectors. For more terse notation, the 12 vectors and tensors will be denoted in plain italic rather than bold, e.g.

$$\vec{q} \equiv \begin{Bmatrix} q_1 \\ q_2 \end{Bmatrix} \quad , \quad \bar{\bar{f}} \equiv \begin{bmatrix} f_{11} & f_{12} \\ \cdot & f_{22} \end{bmatrix} \tag{6.1}$$

6.1.1 Parameters

These input quantities describe the shell geometry, structural properties, mass, and loading.

symbol	num.	axes	description
$\mathbf{r}_{0_j}$	2	xy	position vector of undeformed geometry
$\bar{\bar{A}}_j^{\mathbf{e}}$	6	12	lumped shell stiffness matrix (extension and shear stiffness)
μ_j	1	—	lumped shell mass (mass/area density)
$\vec{q}_j^{\mathbf{e}}$	2	12	shell-following applied force/area
$\mathbf{q}_{xy_j}$	2	xy	fixed-direction applied force/area
$\mathbf{a}_j$	2	xy	local acceleration
$\mathbf{g}$	2	xy	gravity

6.1.2 Unknowns (Primary Variables)

These are the primary variables which are to be determined at each node j.

symbol	num.	axes	description
$\mathbf{r}_j$	2	xy	position vector of deformed geometry
$\mathbf{\Lambda}_j$	1	—	rotation log-quaternion of deformed geometry
$\bar{\bar{f}}^{\mathsf{e}}_j$	3	12	stress resultant tensor components

The state vector will be defined as var

$$\mathsf{var} = [\, r_x,\, r_y,\, \Lambda_3,\, f^{\mathsf{e}}_{11},\, f^{\mathsf{e}}_{22},\, f^{\mathsf{e}}_{12} \,] \tag{6.2}$$

6.2 Element Interpolation

This section provides the two-dimensional simplifications for the Finite-Element Solution of HSM (Section 4.2).

6.2.1 Bilinear Interpolation

Nodal quantities are interpolated over the element via the same four bilinear basis functions $N_{j(\xi,\eta)}$, using the same normalized parameters $-1 \le \xi \le +1$, $-1 \le \eta \le +1$ spanning the cell as in Section 4.2.1.

Note there is no need for the special treatment of the log-quaternion spherical interpolation of Section 4.2.2 because of the fixed normal direction. The log-quaternions may be interpolated like any other nodal quantity.

$$\boldsymbol{\lambda}_{(\xi,\eta)} \;=\; \textstyle\sum_{j=1}^{4} \boldsymbol{\lambda}_j \, N_j \tag{6.3}$$

6.2.2 Tangential Gradients

The covariant tangential basis vectors $\mathbf{a}_\alpha$ are defined as parametric derivatives of the shell geometry function $\mathbf{r}_{(\xi^\alpha)}$.

$$\mathbf{a}_{\alpha\,(\xi^1,\xi^2)} \;\equiv\; \partial_\alpha \mathbf{r} \;:=\; \{\; \partial_\alpha x \quad \partial_\alpha y \;\}^T \tag{6.4}$$

Note that since ξ^α are not necessarily lengths, these $\mathbf{a}_\alpha$ are not unit vectors, and are not even dimensionless. They are also not orthogonal in general.

The tangential gradient components of some interpolated quantities are required for the construction of the equation residuals. From the interpolations of $\mathbf{r}_j = (x_j, y_j)$ above we

have

$$\begin{aligned} x_{(\xi,\eta)} &= \textstyle\sum_j x_j\, N_{j(\xi,\eta)} \quad ; \quad \partial_\xi x = \sum_j x_j\, \partial_\xi N_j \quad , \quad \partial_\eta x = \sum_j x_j\, \partial_\eta N_j \\ y_{(\xi,\eta)} &= \textstyle\sum_j y_j\, N_{j(\xi,\eta)} \quad ; \quad \partial_\xi y = \sum_j y_j\, \partial_\xi N_j \quad , \quad \partial_\eta y = \sum_j y_j\, \partial_\eta N_j \end{aligned} \tag{6.5}$$

which are then used to obtain the two cartesian components of the basis function gradient, which reduces to the cartesian partial derivatives in 2D

$$\widetilde{\nabla} N_{j\,(\xi,\eta)} \equiv (\widetilde{\nabla} N_j)_x\, \hat{\mathbf{x}} + (\widetilde{\nabla} N_j)_y\, \hat{\mathbf{y}} = \partial_x N_j\, \hat{\mathbf{x}} + \partial_y N_j\, \hat{\mathbf{y}} \tag{6.6}$$

via solution of the following 2×2 linear system.

$$\begin{bmatrix} \partial_\xi x & \partial_\xi y \\ \partial_\eta x & \partial_\eta y \end{bmatrix} \begin{Bmatrix} \partial_x N_j \\ \partial_y N_j \end{Bmatrix} = \begin{Bmatrix} \partial_\xi N_j \\ \partial_\eta N_j \end{Bmatrix} \tag{6.7}$$

Note that the vector $\{\partial_\xi x,\, \partial_\xi y\}$ is also the covariant tangential basis vector, $\mathbf{a}_\xi \equiv \partial_\xi \mathbf{r}$, from equation (6.4). The tangential gradients with respect to the undeformed and deformed geometries may be calculated using a similar linear system. The definitions of the non-identical tangential gradient operators are:

$$\widetilde{\nabla}_0 \mathbf{r} \equiv \begin{bmatrix} \partial_{x_0} x & \partial_{x_0} y \\ \partial_{y_0} x & \partial_{y_0} y \end{bmatrix} \qquad \widetilde{\nabla} \mathbf{r}_0 \equiv \begin{bmatrix} \partial_x x_0 & \partial_x y_0 \\ \partial_y x_0 & \partial_y y_0 \end{bmatrix} \tag{6.8}$$

For the deformed position gradients w.r.t. the undeformed geometry $\widetilde{\nabla}_0 \mathbf{r}$ (used in the Lagrangian definition of strain), the 2×2 system to solve is

$$\begin{bmatrix} \partial_\xi x_0 & \partial_\xi y_0 \\ \partial_\eta x_0 & \partial_\eta y_0 \end{bmatrix} \begin{bmatrix} \partial_{x_0} x & \partial_{x_0} y \\ \partial_{y_0} x & \partial_{y_0} y \end{bmatrix} = \begin{bmatrix} \partial_\xi x & \partial_\xi y \\ \partial_\eta x & \partial_\eta y \end{bmatrix} \tag{6.9}$$

For the undeformed position gradients w.r.t. the deformed geometry $\widetilde{\nabla} \mathbf{r}_0$ (used in the Eulerian definition of strain), switching the parametric derivative matrices gives the 2×2 system to solve.

$$\begin{bmatrix} \partial_\xi x & \partial_\xi y \\ \partial_\eta x & \partial_\eta y \end{bmatrix} \begin{bmatrix} \partial_x x_0 & \partial_x y_0 \\ \partial_y x_0 & \partial_y y_0 \end{bmatrix} = \begin{bmatrix} \partial_\xi x_0 & \partial_\xi y_0 \\ \partial_\eta x_0 & \partial_\eta y_0 \end{bmatrix} \tag{6.10}$$

In an alternate formulation (not used in the current HSM2D implementation), once the system (4.35) is solved for the two components of $\widetilde{\nabla} N_j$, the tangential gradient components of any interpolated quantity can be obtained by summation, for example

$$(\widetilde{\nabla} \mathbf{r})_x = \partial_x \mathbf{r} = \textstyle\sum_{j=1}^{4} \mathbf{r}_j\, \partial_x N_j \tag{6.11}$$

$$(\widetilde{\nabla} \mathbf{r})_y = \partial_y \mathbf{r} = \textstyle\sum_{j=1}^{4} \mathbf{r}_j\, \partial_y N_j \tag{6.12}$$

6.2.3 Interpolated Data

The nodal data listed below is interpolated to specified finite-element Gauss points ξ, η in the element interior via the $N_{j(\xi,\eta)}$ interpolation functions. The parametric derivatives of some of the data are also computed at the Gauss points by interpolation using the function derivatives $\partial_\alpha N_{j(\xi,\eta)}$, and the tangential gradients are computed using the function gradient $\widetilde{\nabla} N_{j(\xi,\eta)}$ defined by (4.35).

sym.	N	var. dep.	par. dep.	axes	description
$\mathbf{r}_0$	6		$\mathbf{r}_{0_j}$	xy	undeformed-geometry position
$\mathbf{r}$	6	$\mathbf{r}_j$		xy	deformed-geometry position
$\mathbf{\Lambda}$	1	$\mathbf{\Lambda}_j$		xy	deformed-geometry log-quaternions
$\bar{\bar{f}}^{\mathsf{e}}$	3	$\bar{\bar{f}}^{\mathsf{e}}_j$		e	local non-degen. stress resultants
$\bar{\bar{A}}^{\mathsf{e}}$	6		$\bar{\bar{A}}^{\mathsf{e}}_j$	e	local in-plane stiffness matrix
$\mathbf{q}$	3		$\mathbf{q}_{xy_j}, \vec{q}^{\,\mathsf{e}}_j$	xy	cartesian applied loads
$\mathbf{a}$	3		$\mathbf{a}_j$	xy	cartesian local acceleration
μ	1		μ_j	–	mass/area density

Note $\bar{\bar{\mathbf{e}}}_0$ and $\mathbf{g}$ are not interpolated because they are global quantities that are the same everywhere.

6.2.4 Interior Secondary Variables

These are the secondary variables that are calculated at the Gauss points using data interpolated to the element interior from the nodes.

symbol	N	interpolated dep.	axes	description
$\bar{\bar{\mathbf{e}}}_j$	4	$\mathbf{\Lambda}$	xy	deformed basis vectors $[\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{n}}]_j$
$\bar{\bar{\mathbf{f}}}_j$	6	$\bar{\bar{f}}^{\mathsf{e}}$, $\mathbf{\Lambda}$	xy	cartesian stress resultants
$\mathbf{q}_j$	3	$\mathbf{\Lambda}$, $\mathbf{q}$	xy	cartesian total applied load
$\bar{\bar{\boldsymbol{\varepsilon}}}^{(\mathsf{r})}$	6	$\partial_\alpha \mathbf{r}$, $\partial_\alpha \mathbf{r}_0$	xy	cartesian compatibility strain resultants
$\bar{\bar{\boldsymbol{\varepsilon}}}^{(\mathsf{f})}$	6	$\bar{\bar{f}}^{\mathsf{e}}$, $\bar{\bar{A}}^{\mathsf{e}}$	xy	cartesian constitutive strain resultants

The overall applied force $\mathbf{q}_j$ is the sum of $\mathbf{q}_{xyz_j}$ and $\vec{q}^{\,\mathsf{e}}_j$ in the xyz axes.

$$\mathbf{q}_j \;=\; \mathbf{q}_{xy_j} \;+\; \bar{\bar{\mathbf{e}}}_j \, \vec{q}^{\,\mathsf{e}}_j \tag{6.13}$$

6.2.5 Edge Interpolation

Quantities along an edge are defined from the nodal values using the linear interpolation functions

$$\begin{aligned} \tilde{N}_{1(\xi)} &\equiv \tfrac{1}{2}(1-\xi) \\ \tilde{N}_{2(\xi)} &\equiv \tfrac{1}{2}(1+\xi) \end{aligned} \tag{6.14}$$

which are the same as N_1 and N_2 in (4.15), with $(\xi, \eta) = (\xi, -1)$. We then have

$$\mathbf{r}_{(\xi)} = \textstyle\sum_{j=1}^{2} \mathbf{r}_j \, \tilde{N}_j \tag{6.15}$$

$$\bar{\bar{\mathbf{f}}}_{(\xi)} = \textstyle\sum_{j=1}^{2} \bar{\bar{\mathbf{f}}}_j \, \tilde{N}_j \qquad \text{etc.} \tag{6.16}$$

This interpolation is needed for evaluation of the edge integral in equation (3.22).

6.3 Equation Weighted Residuals

This section provides the two-dimensional simplifications for the Finite-Element Solution of HSM (Section 4.4).

6.3.1 Residual Weights

HSM2D uses the same Galerkin-type finite-element formulation as HSM (see Section 4.4.1). The one difference is that there is no z-component of the weights and there is no need for the construction of the non-degenerate basis.

6.3.2 Gauss Integration

Per standard procedure, all area integrals will be recast in terms of the element coordinates ξ, η and their Jacobian J_0, and then numerically evaluated using 4-point Gaussian quadrature,

$$\iint \mathcal{F} \, \mathrm{d}\mathcal{A}_0 = \iint \mathcal{F} \, J_0 \, \mathrm{d}\xi \, \mathrm{d}\eta \simeq \sum_{k=1}^{4} \mathcal{F}_{(\xi_k, \eta_k)} \, J_{0(\xi_k, \eta_k)} \, w_k \tag{6.17}$$

$$J_0 = |\, \partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0 | \tag{6.18}$$

where the index k runs over the Gauss points (ξ_k, η_k) and w_k are the corresponding Gauss weights. The integration is performed over the undeformed geometry, primarily because this simplifies linearization of the resulting residual expressions for Newton solution.

For the edge line integral appearing in (3.22), 2-point Gaussian integration is used.

$$\int \mathcal{F} \, \mathrm{d}\ell_0 = \int \mathcal{F} \, J_0 \, \mathrm{d}\xi \simeq \sum_{k=1}^{2} \mathcal{F}_{(\xi_k)} \, J_0 \, w_k \tag{6.19}$$

Here, $-1 \leq \xi \leq +1$ is the parameter along the edge, and the Jacobian is a constant $J_0 = \Delta\ell_0/2$ where $\Delta\ell_0$ is the edge length of the undeformed element.

6.3.3 Strain-displacement Residuals

In the preceding sections, two different expressions have been developed for the strains. The expression related to the geometry metrics and basis functions, denoted here by $\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{r})$, is given by (5.20). The values related to the force resultants, denoted here by $\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{f})$, is given by (5.13). Both definitions are calculated at the Gauss points from the necessary interpolated quantities. The appropriate governing equations simply state that these two alternative definitions are numerically equal. The corresponding weighted residuals are then defined as the following.

$$\begin{Bmatrix} \mathcal{R}_i^{\varepsilon 11} \\ \mathcal{R}_i^{\varepsilon 22} \\ \mathcal{R}_i^{\varepsilon 12} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} \varepsilon_{11}^{\mathbf{e}}(\mathbf{r}) - \varepsilon_{11}^{\mathbf{e}}(\mathbf{f}) \\ \varepsilon_{22}^{\mathbf{e}}(\mathbf{r}) - \varepsilon_{22}^{\mathbf{e}}(\mathbf{f}) \\ \varepsilon_{12}^{\mathbf{e}}(\mathbf{r}) - \varepsilon_{12}^{\mathbf{e}}(\mathbf{f}) \end{Bmatrix} W_i \; \mathrm{d}\mathcal{A}_0 \tag{6.20}$$

Note there is no need to dot with the $\hat{\mathbf{w}}_i$ vectors to determine the the non-degenerate basis components as in equation (4.61) because the local basis $\bar{\bar{\mathbf{e}}}$ is assumed to be non-degenerate itself.

6.3.4 Force Equilibrium Residuals

The area-integral part of the in-plane and normal force equilibrium equations for node i is evaluated in the global xy axes. Note these are the same residuals of equation (4.63) except the local basis vectors $\hat{\mathbf{e}}_i$ take the place of the non-degenerate vectors $\hat{\mathbf{w}}_i$, and there is no shell-normal component of the residual.

$$\begin{Bmatrix} \mathcal{R}_i^{f_1} \\ \mathcal{R}_i^{f_2} \end{Bmatrix} \equiv \sum_{\text{elements}} \iint \begin{Bmatrix} -\hat{\mathbf{e}}_1 \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{e}}_1 \, W_i + \mathbf{q} \cdot \hat{\mathbf{e}}_1 \, W_i + \mu \, (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{e}}_1 \, W_i \\ -\hat{\mathbf{e}}_2 \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} W_i - \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{e}}_2 \, W_i + \mathbf{q} \cdot \hat{\mathbf{e}}_2 \, W_i + \mu \, (\mathbf{g} - \mathbf{a}) \cdot \hat{\mathbf{e}}_2 \, W_i \end{Bmatrix} \mathrm{d}\mathcal{A}_0 \tag{6.21}$$

The flux term is evaluated in the xy basis as follows.

$$\begin{aligned} \hat{\mathbf{e}}_1 \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} W_i \;&=\; \left(\hat{\mathbf{e}}_{1x} f_{xx} + \hat{\mathbf{e}}_{1y} f_{xy}\right) (\widetilde{\nabla} W_i)_x \\ &+\; \left(\hat{\mathbf{e}}_{1x} f_{xy} + \hat{\mathbf{e}}_{1y} f_{yy}\right) (\widetilde{\nabla} W_i)_y \end{aligned}$$

The force-curvature term is evaluated in the xy basis as follows.

$$\begin{aligned} \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{e}}_1 \;&=\; f_{xx} \, \partial_x \hat{\mathbf{e}}_{1x} \;+\; f_{xy} \, \partial_y \hat{\mathbf{e}}_{1x} \\ &+\; f_{xy} \, \partial_x \hat{\mathbf{e}}_{1y} \;+\; f_{yy} \, \partial_y \hat{\mathbf{e}}_{1y} \end{aligned}$$

Note $\bar{\bar{\mathbf{f}}}$ refers to the matrix of primary unknowns from Eq.6.1.

6.3.5 Local-basis In-Plane Rotation Constraint Residual

An equation is needed to constrain the rotation of $\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2$ about $\hat{\mathbf{n}}$ and hence within the surface, with no effect on the geometry. Since the properties of a possibly non-isotropic shell material will be specified in the 1,2 axes along $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$, the physical requirement on their orientation is that they remain fixed to the material, with appropriate allowance for shear angle changes. We therefore define material line vectors $\mathbf{s}_0$ and $\tilde{\mathbf{s}}$ for the undeformed and deformed geometries, via their components along $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$.

This basis constraint is similarly integrated and summed over the elements, with the $1/J_0^2$ factor to increase the weighting for small elements. It is also the sixth equation, which closes the system for the state vector var with $N_{\mathrm{DOF}}=6$.

$$\mathcal{R}_i^{e_3} \equiv \sum_{\text{elements}} \iint \frac{1}{J_0^2} \left(\tilde{s}_1 \, \tilde{s}_{0_2} - \tilde{s}_2 \, \tilde{s}_{0_1} \right) W_i \, \mathrm{d}\mathcal{A}_0 \tag{6.22}$$

The Jacobian is defined as

$$J_0 = | \, \partial_\xi \mathbf{r}_0 \times \partial_\eta \mathbf{r}_0 \, | = | \, \partial_\xi x_0 \, \partial_\eta y_0 - \partial_\eta x_0 \, \partial_\xi y_0 \, | \tag{6.23}$$

We define $\tilde{\mathbf{s}}_0$ by rotating $\mathbf{s}_0$ by the $\bar{\bar{\varepsilon}}$ in-plane strain tensor, to account for the fact that $\tilde{\mathbf{s}}$ also is subjected to this same rotation.

$$\tilde{s}_1 = \hat{\mathbf{e}}_1 \cdot \partial_\xi \mathbf{r} \tag{6.24}$$

$$\tilde{s}_2 = \hat{\mathbf{e}}_2 \cdot \partial_\xi \mathbf{r} \tag{6.25}$$

$$s_{0_1} = \hat{\mathbf{e}}_{0_1} \cdot \partial_\xi \mathbf{r}_0 = \partial_\xi x_0 \tag{6.26}$$

$$s_{0_2} = \hat{\mathbf{e}}_{0_2} \cdot \partial_\xi \mathbf{r}_0 = \partial_\xi y_0 \tag{6.27}$$

$$\begin{Bmatrix} \tilde{s}_{0_1} \\ \tilde{s}_{0_2} \end{Bmatrix} = \begin{bmatrix} 1+\varepsilon_{11} & \varepsilon_{12} \\ \varepsilon_{12} & 1+\varepsilon_{22} \end{bmatrix} \begin{Bmatrix} s_{0_1} \\ s_{0_2} \end{Bmatrix} \tag{6.28}$$

The orientation of the $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ vectors in the deformed case is then imposed by requiring that that $\tilde{\mathbf{s}}$ is parallel to $\tilde{\mathbf{s}}_0$. The choice to define $\tilde{\mathbf{s}}$ along $\partial_\xi \mathbf{r}$ is arbitrary, and $\partial_\eta \mathbf{r}$ could have been chosen as well.

Note the strain tensor $\bar{\bar{\varepsilon}}$ used in (6.28) is actually the geometry-dependent strain tensor $\bar{\bar{\varepsilon}}^{\mathbf{e}(\mathbf{r})}$ from equation (5.20) which has been rotated into the cartesian frame $\bar{\bar{\boldsymbol{\varepsilon}}}(\mathbf{r})$ with equation (5.23). The geometric-dependent strain is used because it will give slightly fewer entries in the residual matrix.

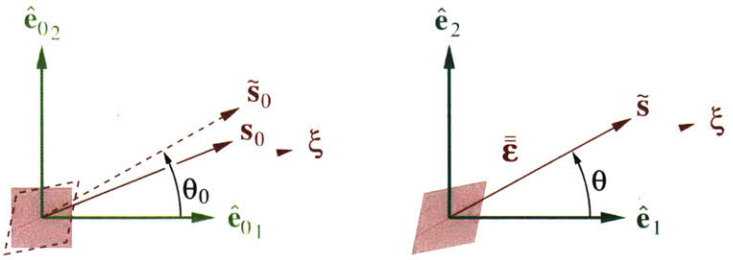


Figure 6-1: Top view of a small shell element for undeformed and deformed geometries. Requiring $\tilde{\mathbf{s}}$ to be parallel to $\tilde{\mathbf{s}}_0$ makes the $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ basis vectors after deformation to be fixed to the same material as $\hat{\mathbf{e}}_{0_1}$ and $\hat{\mathbf{e}}_{0_2}$.

6.4 Boundary Condition Weighted Residuals

This section was developed by Steven Allmaras [2] and edited by William Thalheimer. Here we derive weights for incorporating the boundary conditions of Section 5.5 into the weighted residual formulation of Section 6.3. This involves duality analysis for the analytic/continuous equations to determine appropriate BC weights such that the adjoint has well defined BCs. A well defined adjoint facilitates the the use of HSM2D in optimization, examined in detail in Section 8.

In the discretized geometry, edge axes at a node are obtained via the undeformed geometry $\mathbf{r}_{0(u,v)}$.

$$\hat{\mathbf{l}}_{0_i} = \begin{cases} (\partial_u \mathbf{r}_0/|\partial_u \mathbf{r}_0|)_i & , \text{ along constant-}v \text{ edge} \\ (-\partial_v \mathbf{r}_0/|\partial_v \mathbf{r}_0|)_i & , \text{ along constant-}u \text{ edge} \end{cases} \tag{6.29}$$

$$\hat{\mathbf{l}}_i = \left(\hat{\mathbf{l}}_{0_i} \cdot \hat{\mathbf{e}}_{0_{1_i}}\right) \hat{\mathbf{e}}_{1_i} + \left(\hat{\mathbf{l}}_{0_i} \cdot \hat{\mathbf{e}}_{0_{2_i}}\right) \hat{\mathbf{e}}_{2_i} \tag{6.30}$$

$$\hat{\mathbf{t}}_i = \hat{\mathbf{l}}_i \times \hat{\mathbf{n}}_i \tag{6.31}$$

6.4.1 Dyadic Notation

The following equations make use of dyadic notation to express vector and tensor quantities. A full explanation of dyadics is available in Appendix E. Here we formulate several important quantites in dyadic notation.

The deformed and undeformed local bases are expressed as follows.

$$\bar{\bar{\mathbf{e}}} = \begin{pmatrix} e_{1x} & e_{2x} \\ e_{1y} & e_{2y} \end{pmatrix} = \hat{\mathbf{e}}_1 \hat{\boldsymbol{\imath}} + \hat{\mathbf{e}}_2 \hat{\boldsymbol{\jmath}} \tag{6.32}$$

$$\bar{\bar{\mathbf{e}}}_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \hat{\boldsymbol{\imath}}\hat{\boldsymbol{\imath}} + \hat{\boldsymbol{\jmath}}\hat{\boldsymbol{\jmath}} \tag{6.33}$$

The cartesian stress tensor is

$$\bar{\bar{\mathbf{f}}} = \begin{pmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{pmatrix} = \hat{\imath}\hat{\imath} f_{xx} + (\hat{\imath}\hat{\jmath} + \hat{\jmath}\hat{\imath}) f_{xy} + \hat{\jmath}\hat{\jmath} f_{yy}, \tag{6.34}$$

And the two definitions of strain are

$$\bar{\bar{\boldsymbol{\varepsilon}}}_r \equiv \bar{\bar{\boldsymbol{\varepsilon}}}_{(\mathbf{r})} = \frac{1}{2}\left[(\widetilde{\nabla}_0 \mathbf{r}) \cdot (\widetilde{\nabla}_0 \mathbf{r})^T - \bar{\bar{\mathbf{I}}}_2\right], \qquad \rightarrow \qquad \bar{\bar{\varepsilon}}_r^{\mathbf{e}} = \bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\boldsymbol{\varepsilon}}}_r \cdot \bar{\bar{\mathbf{e}}} \tag{6.35}$$

$$\bar{\bar{\varepsilon}}_f^{\mathbf{e}} \equiv \bar{\bar{\varepsilon}}^{\mathbf{e}}{}_{(\mathbf{f})} = \left[\bar{\bar{\mathbf{A}}}^{\mathbf{e}}\right]^{-1} \cdot \bar{\bar{f}}^{\mathbf{e}} \tag{6.36}$$

with the inner products (expressed by the dots) defined in equations (E.5-E.8).

In contrast to the interior residuals which use the geometry-based strains ($\bar{\bar{\varepsilon}}_r^{\mathbf{e}}$) to calculate the material line vectors (6.28), here we use force-based strains ($\bar{\bar{\varepsilon}}_f^{\mathbf{e}}$) to significantly simplify the linearization and duality analysis. The material line vectors are thus

$$\tilde{\mathbf{s}} = \bar{\bar{\mathbf{e}}}^T \cdot \frac{\partial \mathbf{r}}{\partial \xi}, \qquad \mathbf{s}_0 = \bar{\bar{\mathbf{e}}}_0^T \cdot \frac{\partial \mathbf{r}_0}{\partial \xi}, \qquad \tilde{\mathbf{s}}_0 = \left(\bar{\bar{\mathbf{I}}}_2 + \bar{\bar{\varepsilon}}_f^{\mathbf{e}}\right) \cdot \mathbf{s}_0 \tag{6.37}$$

$$J_0 = \left| \frac{\partial \mathbf{r}_0}{\partial \xi} \times \frac{\partial \mathbf{r}_0}{\partial \eta} \right| \tag{6.38}$$

6.4.2 Primal and Adjoint Solution Vectors

Primal solution variables and associated adjoint solution variables are the following.

$$\underline{\mathbf{u}} = \left\{r_x,\, r_y,\, \Lambda,\, f_{11}^{\mathbf{e}},\, f_{22}^{\mathbf{e}},\, f_{12}^{\mathbf{e}}\right\}^T, \qquad \underline{\mathbf{w}} = \left\{\rho_x,\, \rho_y,\, \Psi,\, \phi_{11}^{\mathbf{e}},\, \phi_{22}^{\mathbf{e}},\, \phi_{12}^{\mathbf{e}}\right\}^T \tag{6.39}$$

Adjoint variables are also used as weights in primal residuals, and linearized primal variables are weights in adjoint residuals. Note that $\underline{\mathbf{w}}$ variables will almost certainly have different units than $\underline{\mathbf{u}}$.

6.4.3 Strain-Displacement Residuals

The adjoint equation to (6.20) imposes equality of the two strain definitions and weights the integral by $\phi_{\alpha\beta}^{\mathbf{e}}$ (the adjoint of $f_{\alpha\beta}^{\mathbf{e}}$),

$$\mathcal{R}^{(\varepsilon_{\alpha\beta})}(\underline{\mathbf{u}}, \phi_{\alpha\beta}^{\mathbf{e}}) = \iint \left(\varepsilon_{ij}^{\mathbf{e}}(\mathbf{r}) - \varepsilon_{\alpha\beta}^{\mathbf{e}}(\mathbf{f})\right) \phi_{\alpha\beta}^{\mathbf{e}} \; \mathrm{d}\mathcal{A}_0, \qquad \alpha,\, \beta \in [1,\, 2] \tag{6.40}$$

Summing over indices gives,

$$\mathcal{R}^{(\varepsilon)}(\underline{\mathbf{u}}, \bar{\bar{\phi}}^{\mathbf{e}}) = \sum_{\alpha=1}^{2}\sum_{\beta=1}^{2} \mathcal{R}^{(\varepsilon_{\alpha\beta})}(\underline{\mathbf{u}}, \phi_{ij}^{\mathbf{e}}) = \iint (\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{r}) - \bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{f})) : \bar{\bar{\phi}}^{\mathbf{e}} \; \mathrm{d}\mathcal{A}_0, \tag{6.41}$$

where the double-dot product for symmetric $\bar{\bar{\mathbf{A}}}$ and $\bar{\bar{\mathbf{B}}}$ is,

$$\bar{\bar{\mathbf{A}}} : \bar{\bar{\mathbf{B}}} \;=\; A_{11}B_{11} + A_{22}B_{22} + 2A_{12}B_{12} \tag{6.42}$$

6.4.4 Force Equilibrium Residuals

The adjoint equation to (6.21) imposes in-plane force equilibrium and weights the integral by ρ_α (adjoint of r_α),

$$\begin{aligned}\mathcal{R}^{(f_\alpha)}(\underline{\mathbf{u}}, \rho_\alpha) = \iint \Big\{ -\hat{\mathbf{e}}_\alpha \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\rho_\alpha \;-\; \rho_\alpha \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{e}}_\alpha \;+\; \mathbf{q} \cdot \hat{\mathbf{e}}_\alpha \, \rho_\alpha \;+\; \mu\,(\mathbf{g}-\mathbf{a}) \cdot \hat{\mathbf{e}}_\alpha \, \rho_\alpha \Big\} \; \mathrm{d}\mathcal{A}_0 \\ + \oint \hat{\mathbf{e}}_\alpha \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} \, \rho_\alpha \; \mathrm{d}\ell, \qquad \alpha \in [1,\, 2]\end{aligned} \tag{6.43}$$

Summing over indices gives,

$$\begin{aligned}\mathcal{R}^{(f)}(\underline{\mathbf{u}}, \boldsymbol{\rho}) &= \sum_{\alpha=1}^{2} \mathcal{R}^{(f_\alpha)}(\underline{\mathbf{u}}, \rho_\alpha) \\ &= \iint \Big\{ -\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\boldsymbol{\rho} : \bar{\bar{\mathbf{I}}}_2 \;-\; \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \cdot \bar{\bar{\mathbf{e}}}^T\right) \cdot \boldsymbol{\rho} \;+\; \mathbf{q} \cdot \bar{\bar{\mathbf{e}}}^T \cdot \boldsymbol{\rho} \;+\; \mu\,(\mathbf{g}-\mathbf{a}) \cdot \bar{\bar{\mathbf{e}}}^t \cdot \boldsymbol{\rho} \Big\} \; \mathrm{d}\mathcal{A}_0 \\ &\quad + \oint (\bar{\bar{\mathbf{f}}} \cdot \bar{\bar{\mathbf{e}}} \cdot \boldsymbol{\rho}) \cdot \hat{\mathbf{t}} \; \mathrm{d}\ell\end{aligned} \tag{6.44}$$

where $\bar{\bar{\mathbf{I}}}_2$ is the identity matrix. The first term could alternately be interpreted as the trace of $\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\boldsymbol{\rho}$. The term, along with the boundary integral, results from the integration-by-parts as follows,

$$\boldsymbol{\rho} \cdot \widetilde{\nabla} \cdot (\bar{\bar{\mathbf{f}}} \cdot \bar{\bar{\mathbf{e}}}) \;=\; \widetilde{\nabla} \cdot \left(\bar{\bar{\mathbf{f}}} \cdot \bar{\bar{\mathbf{e}}} \cdot \boldsymbol{\rho}\right) - \bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot (\widetilde{\nabla}\boldsymbol{\rho}) : \bar{\bar{\mathbf{I}}}_2 \tag{6.45}$$

$$\sum_i \rho_i \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}\right) \cdot \hat{\mathbf{e}}_i \;=\; \left(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \cdot \bar{\bar{\mathbf{e}}}^T\right) \cdot \boldsymbol{\rho} \tag{6.46}$$

$$\sum_i \mathbf{q} \cdot \hat{\mathbf{e}}_i \, \rho_i \;=\; \mathbf{q} \cdot \bar{\bar{\mathbf{e}}}^T \cdot \boldsymbol{\rho} \tag{6.47}$$

6.4.5 In-Plane Basis Rotation Constraint

The adjoint equation to (6.22) enforces in-plane rotations and weights the integral by Ψ (the adjoint of Λ),

$$\mathcal{R}^{(e_3)}(\underline{\mathbf{u}}, \Psi) = \iint \frac{1}{J_0} \hat{\mathbf{n}} \cdot \left(\tilde{\mathbf{s}} \times \tilde{\mathbf{s}}_0\right) \Psi \ \mathrm{d}\mathcal{A}_0 \tag{6.48}$$

6.4.6 Weighted Residuals

The continuous Dirichlet boundary conditions (5.26-5.27) are put in weighted residual form in the following manner.

$$\mathcal{R}^{(\mathbf{r}_{\mathrm{BC}})}(\underline{\mathbf{u}}, \underline{\mathbf{w}}) = \int_{\mathrm{r}} \mathbf{w}_{\mathrm{r}} \cdot \left(\mathbf{r} - \mathbf{r}_{\mathrm{BC}}\right) \ \mathrm{d}\Gamma_{\mathrm{r}} \tag{6.49}$$

$$\mathcal{R}^{(\mathbf{f}_{\mathrm{BC}})}(\underline{\mathbf{u}}, \underline{\mathbf{w}}) = \int_{\mathrm{f}} \mathbf{w}_{\mathrm{f}} \cdot \left(\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} - \bar{\bar{f}}^{\mathbf{e}}_{\mathrm{BC}}\right) \ \mathrm{d}\Gamma_{\mathrm{f}} \tag{6.50}$$

where weights $\mathbf{w}_{\mathrm{r}}$ and $\mathbf{w}_{\mathrm{f}}$ are to-be-determined functions of $\underline{\mathbf{w}}$ and $\underline{\mathbf{u}}$ (linear in $\underline{\mathbf{w}}$ and potentially nonlinear in $\underline{\mathbf{u}}$). Their specific form will be determined from duality analysis. The appropriate weights will produce well defined adjoint BCs. Formation of the primal residual is then posed as follows:

Find $\underline{\mathbf{u}} \in \mathcal{U}$ such that $\mathcal{R}(\underline{\mathbf{u}}, \underline{\mathbf{w}}) = 0$ for all $\underline{\mathbf{w}} \in \mathcal{U}$, where

$$\mathcal{R}(\underline{\mathbf{u}}, \underline{\mathbf{w}}) = \mathcal{R}^{(\epsilon)}(\underline{\mathbf{u}}, \bar{\bar{\phi}}^{\mathbf{e}}) + \mathcal{R}^{(f)}(\underline{\mathbf{u}}, \boldsymbol{\rho}) + \mathcal{R}^{(e_3)}(\underline{\mathbf{u}}, \Psi) + \mathcal{R}^{(\mathbf{r}_{\mathrm{BC}})}(\underline{\mathbf{u}}, \underline{\mathbf{w}}) + \mathcal{R}^{(\mathbf{f}_{\mathrm{BC}})}(\underline{\mathbf{u}}, \underline{\mathbf{w}}) \tag{6.51}$$

6.4.7 Duality Analysis and Adjoint PDE and BCs

Duality is a statement on the equivalence between primal and adjoint output functionals for solutions that satisfy the primal and adjoint PDE and BCs. We will use an somewhat unconventional form of the duality statement that explicitly incorporates weighted primal and adjoint residuals. The resulting analysis provides a means of determining the adjoint PDEs and associated BCs. It should also determine the appropriate primal BC weights $\mathbf{w}_{\mathrm{f}}$ and $\mathbf{w}_{\mathrm{r}}$ such that the adjoint BCs are well defined. For general BC weights the duality analysis leads to multiple and inconsistent constraints on the boundary adjoint, giving a nonsensical formulation. Only a specific choice of BC weights leads to well defined adjoint BCs (i.e. unique and determinate conditions on the boundary adjoint).

The process is as follows:

(i) Linearize the primal residuals and output functional about a state $\underline{\mathbf{u}}$ that satisfies the primal PDE and BCs;

(ii) Linear duality statement to identify adjoint residual and output functional;

(iii) Integration-by-parts (IBP) where needed to remove derivatives from primal weights in adjoint residual, and collect terms by primal weight multipliers;

(iv) Examine boundary integral contributions to determine adjoint BCs and constraints on primal BC weights.

6.4.8 Discrete Residuals

The final (continuous) primal residual is,

$$\begin{aligned}\mathcal{R}(\underline{\mathbf{u}},\underline{\mathbf{w}}) &= \mathcal{R}^{(\varepsilon)}(\underline{\mathbf{u}},\bar{\bar{\phi}}^{\mathbf{e}}) + \mathcal{R}^{(f)}(\underline{\mathbf{u}},\boldsymbol{\rho}) + \mathcal{R}^{(e_3)}(\underline{\mathbf{u}},\Psi) + \mathcal{R}^{(\mathbf{r}_{\mathrm{BC}})}(\underline{\mathbf{u}},\mathbf{w}_{\mathrm{r}}) + \mathcal{R}^{(\mathbf{f}_{\mathrm{BC}})}(\underline{\mathbf{u}},\mathbf{w}_{\mathbf{f}}) \\ &= \tilde{\mathcal{R}}^{(\varepsilon)}(\underline{\mathbf{u}},\bar{\bar{\phi}}^{\mathbf{e}}) + \tilde{\mathcal{R}}^{(f)}(\underline{\mathbf{u}},\boldsymbol{\rho}) + \tilde{\mathcal{R}}^{(e_3)}(\underline{\mathbf{u}},\Psi) \end{aligned} \tag{6.52}$$

where the BC-augmented residuals are,

$$\tilde{\mathcal{R}}^{(\varepsilon)}(\underline{\mathbf{u}},\bar{\bar{\phi}}^{\mathbf{e}}) = \iint_{\mathcal{A}_0} (\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{r}) - \bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{f})) : \bar{\bar{\phi}}^{\mathbf{e}} - \int_{\mathrm{r}} \hat{\mathbf{t}}\cdot\bar{\bar{\mathbf{e}}}\cdot\bar{\bar{\phi}}^{\mathbf{e}}\cdot\bar{\bar{\mathbf{e}}}^{T}\cdot(\widetilde{\nabla}_0\mathbf{r})\cdot(\mathbf{r}-\mathbf{r}_{\mathrm{BC}}) \tag{6.53}$$

$$\begin{aligned}\tilde{\mathcal{R}}^{(f)}(\underline{\mathbf{u}},\boldsymbol{\rho}) = \iint_{\mathcal{A}_0}\Big\{-\bar{\bar{\mathbf{e}}}^{T}\cdot\bar{\bar{\mathbf{f}}}\cdot\widetilde{\nabla}\boldsymbol{\rho} : \bar{\bar{\mathbf{I}}}_2 - \Big(\bar{\bar{\mathbf{f}}}\cdot\widetilde{\nabla}\cdot\bar{\bar{\mathbf{e}}}^{T}\Big)\cdot\boldsymbol{\rho} + \mathbf{q}\cdot\bar{\bar{\mathbf{e}}}^{T}\cdot\boldsymbol{\rho} + \mu(\mathbf{g}-\mathbf{a})\cdot\bar{\bar{\mathbf{e}}}^{T}\cdot\boldsymbol{\rho}\Big\} \\ + \oint_{\ell}(\boldsymbol{\rho}\cdot\bar{\bar{\mathbf{e}}}^{T}\cdot\bar{\bar{\mathbf{f}}})\cdot\hat{\mathbf{t}} - \int_{\mathrm{f}}\boldsymbol{\rho}\cdot(\bar{\bar{\mathbf{e}}}^{T}\cdot\bar{\bar{\mathbf{f}}}\cdot\hat{\mathbf{t}} - \bar{\bar{f}}^{\mathbf{e}}_{\mathrm{BC}})\end{aligned} \tag{6.54}$$

$$\tilde{\mathcal{R}}^{(e_3)}(\underline{\mathbf{u}},\Psi) = \iint_{\mathcal{A}_0}\frac{1}{J_0}\hat{\mathbf{n}}\cdot(\tilde{\mathbf{s}}\times\tilde{\mathbf{s}}_0)\Psi + \int_{\mathrm{r}}\frac{1}{J_0}(\hat{\boldsymbol{\xi}}\cdot\hat{\mathbf{t}})\Psi\,\hat{\mathbf{n}}\cdot\Big(\tilde{\mathbf{s}}_0\times\bar{\bar{\mathbf{e}}}^{T}\Big)\cdot(\mathbf{r}-\mathbf{r}_{\mathrm{BC}}) \tag{6.55}$$

We now expand the primal and adjoint solutions into the discretized geometry DOFs times basis/shape functions,

$$\underline{\mathbf{u}}(x,y) = \sum_j \underline{\mathbf{u}}_j\, N_{j(x,y)}, \qquad \underline{\mathbf{w}}(x,y) = \sum_j \underline{\mathbf{w}}_j\, N_{j(x,y)}, \tag{6.56}$$

where $N_{j(x,y)}$ are the basis/shape functions. Substitute $\underline{\mathbf{w}}$ into the primal residuals,

$$\mathcal{R}(\underline{\mathbf{u}},\underline{\mathbf{w}}) = \mathcal{R}(\underline{\mathbf{u}},\sum_j \underline{\mathbf{w}}_j\, N_j) = \sum_j \underline{\mathbf{w}}_j\cdot\boldsymbol{R}(\underline{\mathbf{u}},N_j) = 0 \tag{6.57}$$

where $\boldsymbol{R}(\underline{\mathbf{u}},N_j)$ is the vector of residuals associated with a given node/DOF,

$$\boldsymbol{R}(\underline{\mathbf{u}},N_j) = \Big\{\tilde{R}_j^{(\varepsilon 11)}(\underline{\mathbf{u}}),\ \tilde{R}_j^{(\varepsilon 22)}(\underline{\mathbf{u}}),\ \tilde{R}_j^{(\varepsilon 12)}(\underline{\mathbf{u}}),\ \tilde{R}_j^{(f_1)}(\underline{\mathbf{u}}),\ \tilde{R}_j^{(f_2)}(\underline{\mathbf{u}}),\ \tilde{R}_j^{(e_3)}(\underline{\mathbf{u}})\Big\}^{T} \tag{6.58}$$

With arbitrary weight DOFs, each nodal residual must be zero. The individual nodal residuals are then,

$$\tilde{R}_j^{(\varepsilon 11)}(\underline{\mathbf{u}}) = \iint_{\mathcal{A}_0} (\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{r}) - \bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{f})) : (\hat{\imath}\hat{\imath} N_j) - \int_{\mathrm{r}} \hat{\mathbf{t}} \cdot \bar{\bar{\mathbf{e}}} \cdot (\hat{\imath}\hat{\imath} N_j) \cdot \bar{\bar{\mathbf{e}}}^T \cdot (\widetilde{\nabla}_0 \mathbf{r}) \cdot (\mathbf{r} - \mathbf{r}_{\mathrm{BC}}) \tag{6.59}$$

$$\tilde{R}_j^{(\varepsilon 22)}(\underline{\mathbf{u}}) = \iint_{\mathcal{A}_0} (\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{r}) - \bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{f})) : (\hat{\jmath}\hat{\jmath} N_j) - \int_{\mathrm{r}} \hat{\mathbf{t}} \cdot \bar{\bar{\mathbf{e}}} \cdot (\hat{\jmath}\hat{\jmath} N_j) \cdot \bar{\bar{\mathbf{e}}}^T \cdot (\widetilde{\nabla}_0 \mathbf{r}) \cdot (\mathbf{r} - \mathbf{r}_{\mathrm{BC}}) \tag{6.60}$$

$$\tilde{R}_j^{(\varepsilon 12)}(\underline{\mathbf{u}}) = \iint_{\mathcal{A}_0} (\bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{r}) - \bar{\bar{\varepsilon}}^{\mathbf{e}}(\mathbf{f})) : \big((\hat{\imath}\hat{\jmath} + \hat{\jmath}\hat{\imath}) N_j\big) - \int_{\mathrm{r}} \hat{\mathbf{t}} \cdot \bar{\bar{\mathbf{e}}} \cdot \big((\hat{\imath}\hat{\jmath} + \hat{\jmath}\hat{\imath}) N_j\big) \cdot \bar{\bar{\mathbf{e}}}^T \cdot (\widetilde{\nabla}_0 \mathbf{r}) \cdot (\mathbf{r} - \mathbf{r}_{\mathrm{BC}}) \tag{6.61}$$

$$\tilde{R}_j^{(f_1)}(\underline{\mathbf{u}}) = \iint_{\mathcal{A}_0} \Big\{ -\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}(\hat{\imath} N_j) : \bar{\bar{\mathbf{I}}}_2 - \Big(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \cdot \bar{\bar{\mathbf{e}}}^T\Big) \cdot (\hat{\imath} N_j) + \big(\mathbf{q} + \mu(\mathbf{g} - \mathbf{a})\big) \cdot \bar{\bar{\mathbf{e}}}^T \cdot (\hat{\imath} N_j) \Big\} + \oint_{\ell} (\hat{\imath} N_j) \cdot \bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} - \int_{\mathrm{f}} (\hat{\imath} N_j) \cdot \big(\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} - \bar{\bar{f}}^{\mathbf{e}}_{\mathrm{BC}}\big) \tag{6.62}$$

$$\tilde{R}_j^{(f_2)}(\underline{\mathbf{u}}) = \iint_{\mathcal{A}_0} \Big\{ -\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla}(\hat{\jmath} N_j) : \bar{\bar{\mathbf{I}}}_2 - \Big(\bar{\bar{\mathbf{f}}} \cdot \widetilde{\nabla} \cdot \bar{\bar{\mathbf{e}}}^T\Big) \cdot (\hat{\jmath} N_j) + \big(\mathbf{q} + \mu(\mathbf{g} - \mathbf{a})\big) \cdot \bar{\bar{\mathbf{e}}}^T \cdot (\hat{\jmath} N_j) \Big\} + \oint_{\ell} (\hat{\jmath} N_j) \cdot \bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} - \int_{\mathrm{f}} (\hat{\jmath} N_j) \cdot \big(\bar{\bar{\mathbf{e}}}^T \cdot \bar{\bar{\mathbf{f}}} \cdot \hat{\mathbf{t}} - \bar{\bar{f}}^{\mathbf{e}}_{\mathrm{BC}}\big) \tag{6.63}$$

$$\tilde{R}_j^{(e_3)}(\underline{\mathbf{u}}) = \iint_{\mathcal{A}_0} \frac{1}{J_0} \hat{\mathbf{n}} \cdot (\tilde{\mathbf{s}} \times \tilde{\mathbf{s}}_0) N_j + \int_{\mathrm{r}} \frac{1}{J_0} (\hat{\boldsymbol{\xi}} \cdot \hat{\mathbf{t}}) N_j \, \hat{\mathbf{n}} \cdot \Big(\tilde{\mathbf{s}}_0 \times \bar{\bar{\mathbf{e}}}^T\Big) \cdot (\mathbf{r} - \mathbf{r}_{\mathrm{BC}}) \tag{6.64}$$

where the dyadics can be interpreted in terms of vectors and matrices,

$$\hat{\imath} = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}, \qquad \hat{\jmath} = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}, \qquad \hat{\imath}\hat{\imath} = \begin{Bmatrix} 1 & 0 \\ 0 & 0 \end{Bmatrix}, \qquad \hat{\jmath}\hat{\jmath} = \begin{Bmatrix} 0 & 0 \\ 0 & 1 \end{Bmatrix}, \qquad \hat{\imath}\hat{\jmath} + \hat{\jmath}\hat{\imath} = \begin{Bmatrix} 0 & 1 \\ 1 & 0 \end{Bmatrix} \tag{6.65}$$

and the $\hat{\mathbf{n}} \cdot \Big(\tilde{\mathbf{s}}_0 \times \bar{\bar{\mathbf{e}}}^T\Big)$ term in equation (6.64) is calculated via equation (E.28) with $\hat{\boldsymbol{k}} = \hat{\mathbf{n}}$ so explicitly the term becomes

$$\hat{\mathbf{n}} \cdot \Big(\tilde{\mathbf{s}}_0 \times \bar{\bar{\mathbf{e}}}^T\Big) = \hat{\imath}(e_{1_y} \tilde{s}_{0_x} - e_{1_x} \tilde{s}_{0_y}) + \hat{\jmath}(e_{2_y} \tilde{s}_{0_x} - e_{2_x} \tilde{s}_{0_y}) \tag{6.66}$$

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 7

Verification

7.1 Definition

A physical model may be "verified" if, in solving the continuum partial differential equations and continuum boundary conditions via discretization, the observed discretization error decreases to zero as the mesh increments decrease to zero [21]. This means the equations are "solved correctly", or that the code is solving the equations it was intended to. To be mathematically explicit, the verification process proves that the governing equations, as implemented in the code, are solved *consistently.* Code verification does not seek to answer if the equations of the HSM method are correctly determining the deformation of solids, i.e. that we are solving the physically correct equations. This is code *validation*, which is left as a topic for future investigation of HSM.

7.2 Methods

Several methods have been used in the HSM verification process.

The Method of Exact Solutions (MES) involves using an exact, analytical solution for plate deformation as a benchmark for comparison against the HSM results. In comparing the HSM to an exact solution, the same loading and boundary conditions are imposed on a given geometry, and the resulting HSM deformations and stresses are compared directly to the analytical deformations and stresses. MES is limited in scope and complexity however, since few analytical solutions exist for plate deformations. Furthermore, *all* exact solutions only apply to small deformations (and are typically based on small-deformation approximations of plate behavior, such as Kirchoff-Love or Mindlin-Reissner theories), which doesn't help test the intended purpose of HSM which is to accurately resolve *large* surface deformations by only applying these approximations locally.

The Method of Manufactured Solutions (MMS) addresses the restrictions of MES. A manufactured solution is an exact solution to the set of governing PDEs of HSM that has been

constructed by solving the problem *in reverse*. This means the solution procedure begins with specifying the deformed geometry and boundary conditions, and the applied loads which will result in the specified geometry are then computed. When HSM is to be compared to a manufactured solution, the loads calculated from the manufactured solution are imposed on the same geometry and boundary conditions in HSM, and the output deformations from HSM are compared back to the originally specified (input) deformations of the manufactured solution. Because these displacements are arbitrary, MMS does not have the geometry and loading restrictions that MES does; any displacement may be imposed on the shell and the associated forces will be calculated. However, the complication of MMS is that *source terms* must be added to the governing PDEs. These are additional terms that allow the freedom in initial condition specification by "filling in the gaps" in the mapping between loading and displacement. For MMS to be used, these source terms must be specified in the case setup of the computational implementation of HSM at the same time as the output forces from the MMS calculation are. Currently, these terms are not included in the implementation. Therefore it is noted that although this study proposes several test cases for HSM, their actual use in code verification is left for future study.

The Method of Comparison to an Industry Standard Analog (MCA) involves comparing the HSM results to a structural modeling code that has already gained widespread acceptance. The commercially-available finite element code used for comparison in this study is Solidworks, which features shell modeling capable of large deflections. The advantage of comparing HSM to another code (as opposed to MES) is that Solidworks can model geometries and loadings closer to what is expected for HSM to encounter in practice (i.e. wings bending under aerodynamic and inertial loads). This study also combines MCA with MES to reinforce claims of verification by simultaneously verifying the industry standard analog. In doing so, the shell geometries and loading configurations have been limited to cases with analytical solutions. More complicated configurations are left as further research topics.

7.3 3D Verification

7.3.1 Initial Verification with MES and MCA

A simplified model of HSM/KL was used in the early stages of HSM development to verify the computational implementation via MES and MCA. The differences between this early version and the current HSM formulation are:

- Definition of local basis, $\bar{\bar{\mathbf{e}}}$, from parametric derivatives $\partial_\xi \mathbf{r}$, $\partial_\eta \mathbf{r}$, instead of via log-quaternions
- Definition of strains via the metric tensor method of Simo [24] instead of the Cartesian method of Section 3.4.

This version was implemented in Fortran by Prof. Mark Drela and verified using MES and MCA. The error metric is global L2 error via Gauss quadrature. This way both HSM and the industry-standard analog (Solidworks) are verified with an analytical method, giving more confidence to future MCA verification.

The configuration of the first HSM test was a simply-supported square plate loaded in the normal direction by a uniform pressure field. Several sources have performed the Kirchoff-Love analysis for such loading of a pinned rectangular plate of side lengths a, and b, where $0 \leq x \leq a$ and $0 \leq y \leq b$. eFunda [10] reports the transverse deflection, w, of a plate of flexural rigidity $D = E\,t^3 / \,12\,(1-\nu^2)$ subjected to a uniform traverse load p as

$$w_{(x,y)} = \sum_{m=1,3,\ldots}^{\infty} \frac{16p}{\pi^6 D} \sum_{n=1,3,\ldots}^{\infty} \frac{\sin\left(\pi\, x\, \frac{m}{a}\right) \sin\left(\pi\, y\, \frac{n}{b}\right)}{m\, n \left(\left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2\right)} \tag{7.1}$$

A square simply supported plate was chosen as the first test case due to its simplicity and symmetry. The problem setup is shown in Figure 7-1, with the plate parameters in Table 7.1.

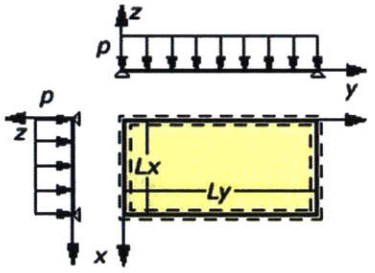


Figure 7-1: Loading of a simply-supported, uniform-transversely-loaded rectangular plate

param.	value
L_x, L_y	100 m
t	0.5 m
p	1 Pa
E	1 GPa
ν	0.3

Table 7.1: SS square plate parameters

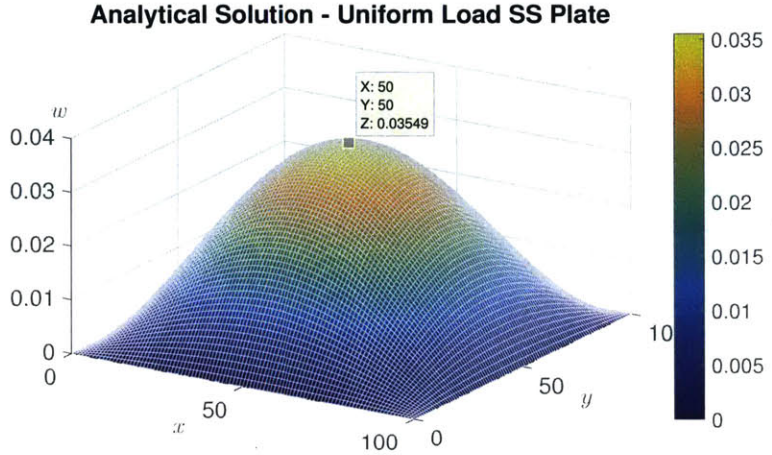


Figure 7-2: Analytic solution using KLPT for the transverse deflection of a simply-supported uniform-tranversely-loaded square plate

The analytical solution was implemented in MATLAB and sampled on the nodes of a very fine (101x101) rectilinear grid, as shown in Figure 7-2. The same plate was then modeled on a coarser grid (21x21) with HSM as shown in Figure 7-3. Note HSM uses a rectilinear mesh.

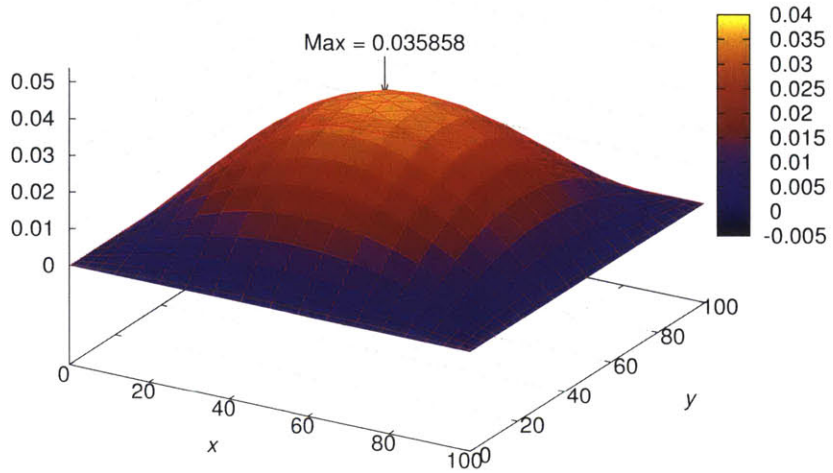


Figure 7-3: HSM calculation of the transverse deflection of a simply-supported uniform-tranversely-loaded square plate with maximum deflection location and value noted

The plate was then modeled as a thin shell in Solidworks and the solution was determined using the "Solidworks Simulation" finite element tool, with the small displacement option turned on (Figure 7-4). Note Solidworks uses a triangular mesh. A discussion on the actual algorithms implemented by the Solidworks solver is given in Appendix B.

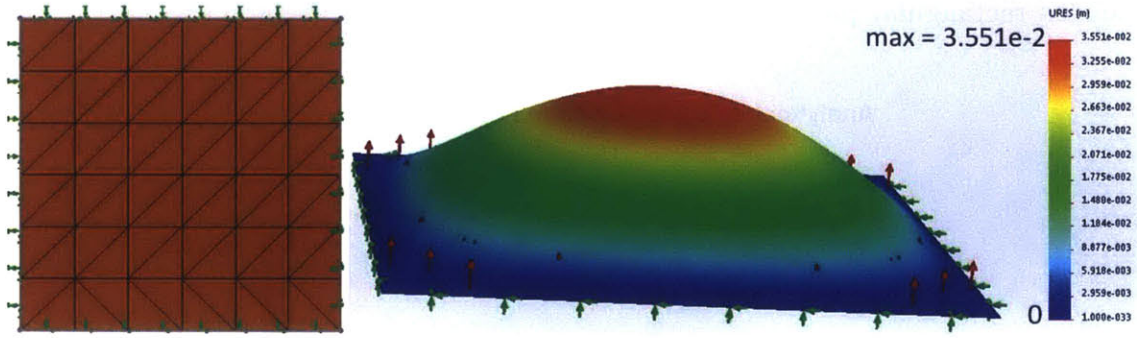


Figure 7-4: Solidworks mesh (left) and calculation of transverse deflection (right) of a simply-supported uniform-tranversely-loaded square plate

The HSM and Solidworks results were then compared to the analytical Kirchoff-Love solution for several grid sizes. The results were compared using different error integral approximation algorithms. The Solidworks integral error was computed via 4-point Gauss integration on each triangular element (more detail in Appendix A.2). The HSM integral error was computed with three different algorithms. The green line in Figure 7-5 is a nodal analysis, where

the squared difference from the analytical solution was taken at each node, and weighted by the total area divided by the number of nodes (i.e. the size of the average element if they are equal and set on the nodes). The red line is the same algorithm as the one used to process the Solidworks data. Each quadrilateral element in HSM was split into two triangles, and then a 4-point Gauss quadrature was done on each triangular element. The blue line is a 4-point Gauss quadrature done on each quadrilateral element. The mathematical implementation for quadrilateral integration is slightly different than the triangular set up, and is summarized in Appendix A.1. The l^2 error is plotted against the characteristic element length, $L_{char} = 1/\sqrt{N_{elems}}$, for five mesh sizes for both HSM and Solidworks and is shown in Figure 7-5. The expected error convergence was L_{char}^{-2}. Solidworks and HSM (with all error integral approximations) very closely matched this.

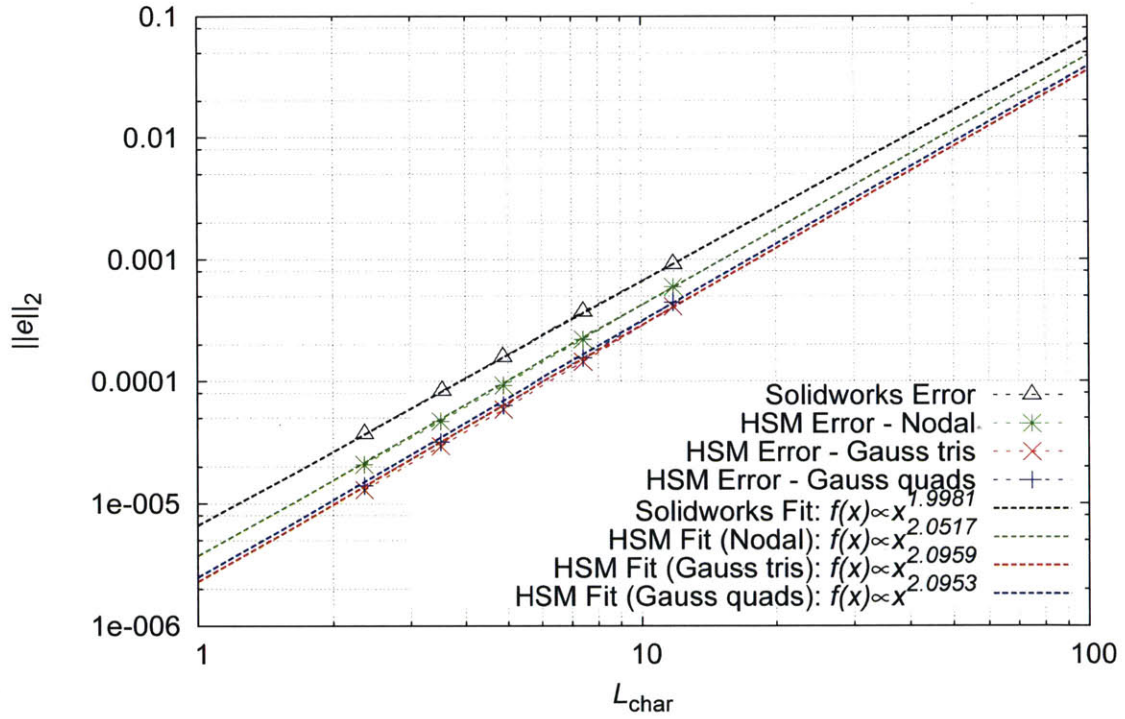


Figure 7-5: Error convergence for HSM and Solidworks models of simply-supported uniform-transversely-loaded square plate with L2 error metrics using (1) nodal summation, (2) Gauss quadrature on triangular elements, and (3) Gauss quadrature on quadrilateral elements.

7.3.2 Future Verification with MMS

More exotic loadings and geometries may be verified in the future via the method of manufactured solutions. Several recommendations of test cases are proposed in this section. They have been modeled in Mathematica with the intent of being compared to the HSM results

via Gauss quadrature in a similar procedure to Sections 7.3.1 and 7.4.

Manufactured solutions are parametrized by a ξ, η rectangular grid where $\xi, \eta \in [0, 1]$. This computational grid is mapped to an x, y, z grid where $x = f(\xi,\eta)$, $y = g(\xi,\eta)$, $z = h(\xi,\eta)$ defines an arbitrary geometry in Cartesian space. The undeformed geometry is represented with x_0, y_0, z_0 and the deformed geometry is represented with x, y, z.

Note that only the geometric deformations and stresses are shown in this paper. A Mathematica script was written to also output the nodal source term data necessary to recreate these manufactured solutions with the HSM code. The ability to take source term input has not been implemented in the HSM code yet, and is left as an avenue for future development.

Parabolically Bent Square Plate in 1D

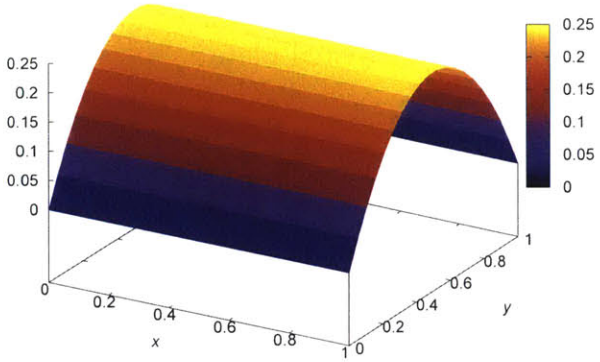
A simple test case is square plate, discretized with a 20x20 grid, parabolically bent in the y-direction. The deformed and undeformed geometries of this manufactured solution are summarized in Table 7.2. The material properties and loading conditions of the plate are summarized in Table 7.3. Results for this manufactured solution are shown in Figure 7-6. Note the expected result that there is no x-direction or shear-stress since the only curvature is in the y direction, also that the stress in the y-direction assumes the expected parabolic profile.

Table 7.2: Parametrization of 1D parabolic bent plate

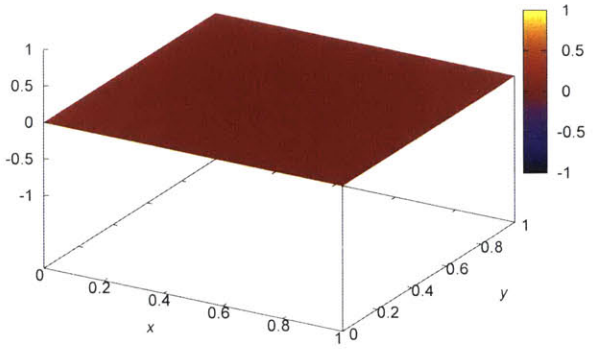
	Undef.	Deformed
$f(\xi,\eta)$	$x_0 = \xi$	$x = \xi$
$g(\xi,\eta)$	$y_0 = \eta$	$y = \eta$
$h(\xi,\eta)$	$z_0 = 0$	$z = \eta(1-\eta)$

Table 7.3: Material properties and loading conditions of 1D bent plate

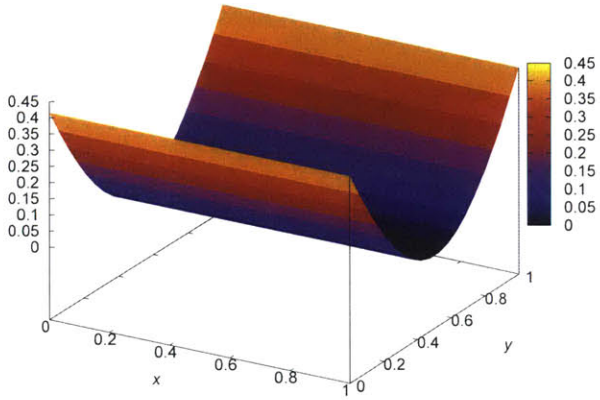
Parameter	Sym.	Value
x-length	L_x	1 m
y-length	L_y	1 m
Shell (z) thickness	L_z	0.01 m
Young's Modulus	E	100 Pa
Poisson's Ratio	ν	0
Applied force/area	$\vec{q}$	0
Local gravity	$\vec{g}$	0
Local acceleration	$\vec{a}$	0
Material Type		Isotropic



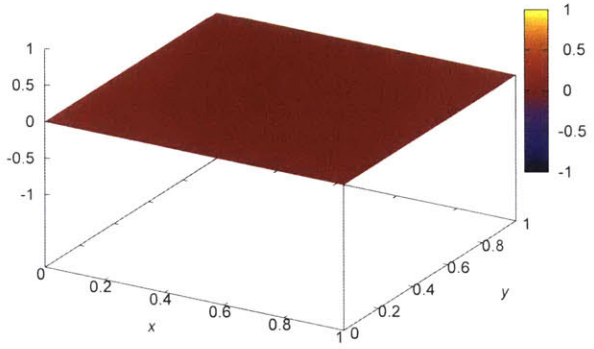
(a) Deformation



(b) x-stress (σ_{11})



(c) y-stress (σ_{22})



(d) Shear-stress (σ_{12})

Figure 7-6: Manufactured 1D Parabolic Plate

Parabolically Bent Square Plate in 2D

A more complicated test case involves parabolically bending the same square plate from Section 7.3.2 (see Table 7.3 for material/loading parameters) bent in *both* the x- and y-directions. The deformed and undeformed geometries are summarized in Table 7.4.

Results for this manufactured solution are shown in Figure 7-7. Note the x and y extensional stresses are greatest at the extrema of the associated dimension and along the centerline of the other dimension (aka x-direction extensional stresses are greatest at $(x, y) = (\pm x_{max}, y_{CL})$, and the y-direction extensional stresses are greatest at $(x, y) = (x_{CL}, \pm y_{max})$). This is as expected from the perspective of membrane loading. Also note that the undeformed middle

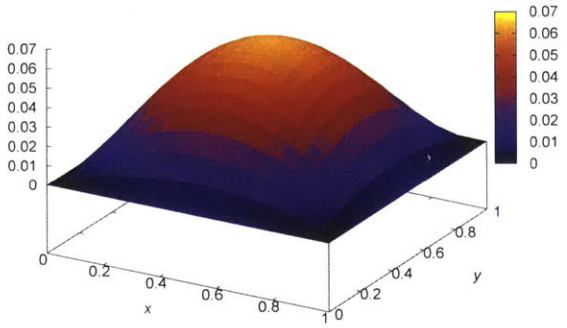
region has little to no stress (extensional or shear) as expected.

Table 7.4: Parametrization of 2D parabolic bent plate

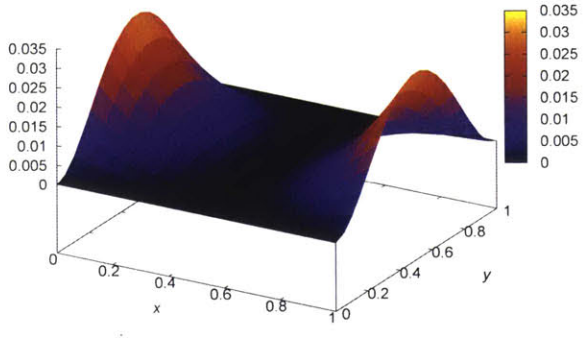
	Undef.	Deformed
$f(\xi,\eta)$	$x_0 = \xi$	$x = \xi$
$g(\xi,\eta)$	$y_0 = \eta$	$y = \eta$
$h(\xi,\eta)$	$z_0 = 0$	$z = \xi\eta(1-\xi)(1-\eta)$

Table 7.5: Parametrization of rectangular twisted plate

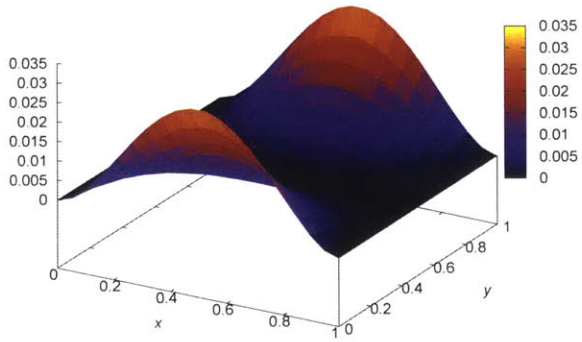
	Undef.	Deformed
$f(\xi,\eta)$	$x_0 = \xi$	$x = \xi$
$g(\xi,\eta)$	$y_0 = \frac{1}{2}(\eta - \frac{1}{2})$	$y = \frac{1}{4}(\cos(\pi\xi)+1)(\eta - \frac{1}{2})$
$h(\xi,\eta)$	$z_0 = 0$	$y = \frac{1}{4}(\cos(\pi\xi)-1)(\eta - \frac{1}{2})$



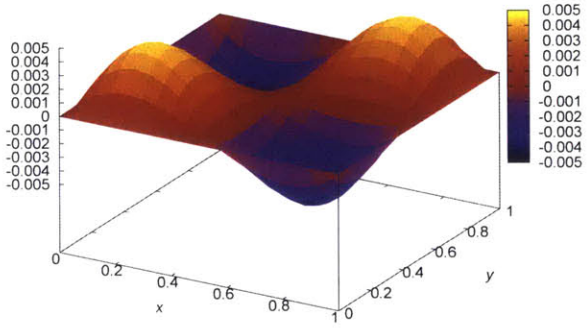
(a) Deformation



(b) x-stress (σ_{11})



(c) y-stress (σ_{22})

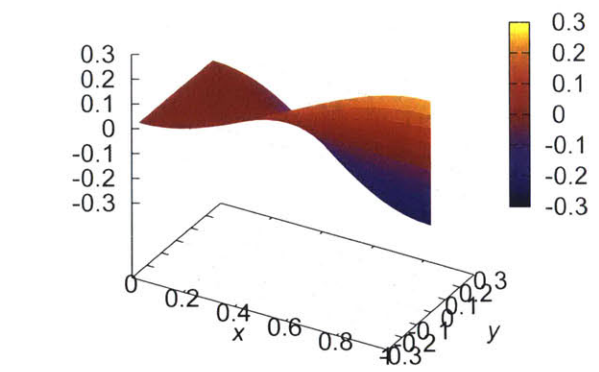


(d) Shear-stress (σ_{12})

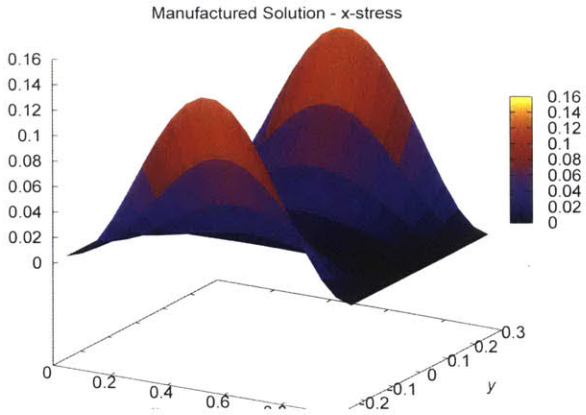
Figure 7-7: Manufactured 1D Parabolic Plate

Twisted Rectangular Plate

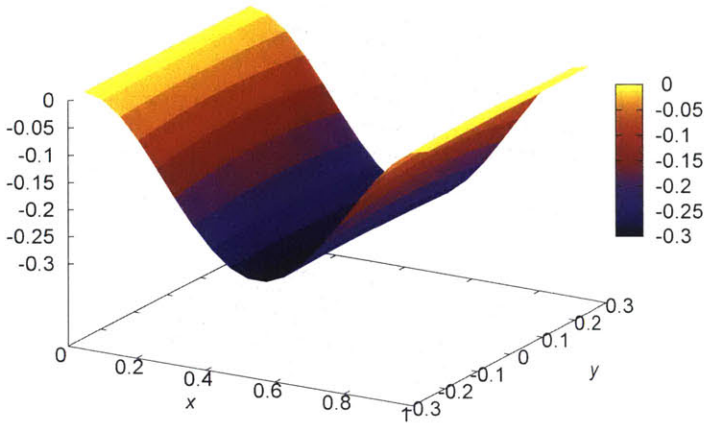
Another test case is that of a rectangular plate with a 90 degree twist about the centerline x-axis. The plate has the same properties as the square plate from Section 7.3.2 (see Table 7.3 for material/loading parameters), except the y-length is halved ($L_y = 0.5$m). This model is selected to demonstrate large deformations and show some applicability towards typical wing skin deformations. The deformed and undeformed geometries are summarized in Table 7.5.



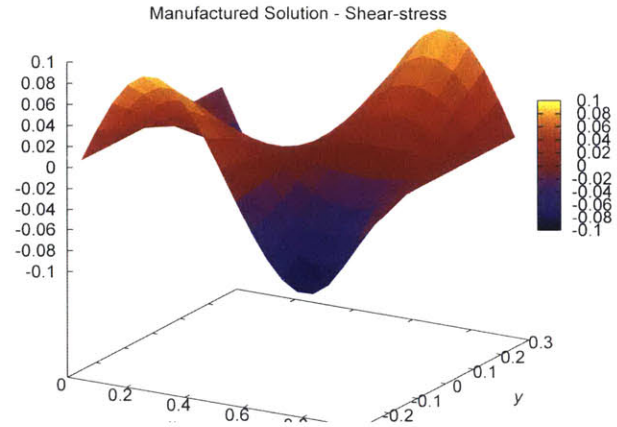
(a) Deformation



(b) x-stress (σ_{11})



(c) y-stress (σ_{22})



(d) Shear-stress (σ_{12})

Figure 7-8: Manufactured Twisted Rectangular Plate

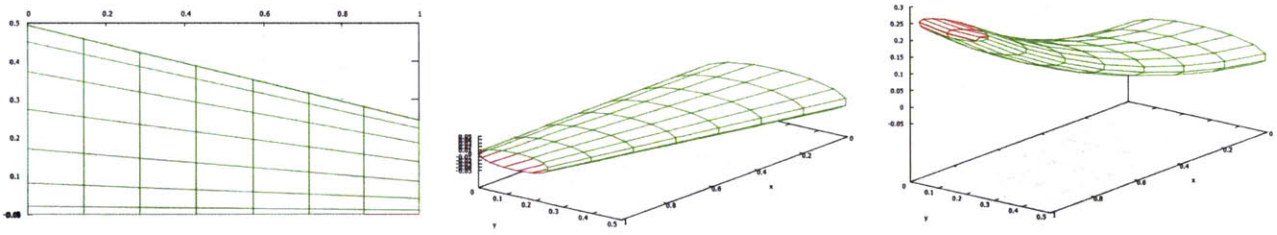
Results for this manufactured solution are shown in Figure 7-8. Note the x-stresses are greatest at the tightest curvature (i.e. at the middle x-location and the y-extrema), while the y-stresses are essentially constant along the y-direction since the twist is happening at a constant rate along the x-direction.

Bent Wing with Elliptical Cross-Section

Ultimately, the intended application of the HSM code is to model the deflection of wings under aerodynamic loading. To approximate such a geometry, a manufactured solution consisting of a wing (with the same structural properties and loading, but not dimensions, as Table 7.3) is suggested as a reasonably complex test case. The wing modeled in this example has an elliptical cross-section with a thickness-to-chord ratio, $^t/_c$, of 10%, and is bent upwards along an arc with radius of twice the wingspan. The deformed and undeformed geometries are summarized in Table 7.6 and Figure 7-9.

Table 7.6: Parametrization of bent elliptic cross-section wing

	Undef.	Deformed
$f_{(\xi,\eta)}$	$x_0 = \xi$	$x = \sin(\xi/2)\left(2 - \frac{1}{20}\sin(\xi/2)(1 - (\xi/2))\right)$
$g_{(\xi,\eta)}$	$y_0 = \frac{1}{4}(1 - \cos(2\pi\eta))(1 - (\xi/2))$	$y = \frac{1}{4}(1 - \cos(2\pi\eta))(1 - (\xi/2))$
$h_{(\xi,\eta)}$	$z_0 = \frac{1}{20}\sin(2\pi\eta)(1 - (\xi/2))$	$z = 2(1 - \cos(\xi/2)) + \frac{1}{20}\cos(\xi/2)\sin(2\pi\eta)(1 - (\xi/2))$

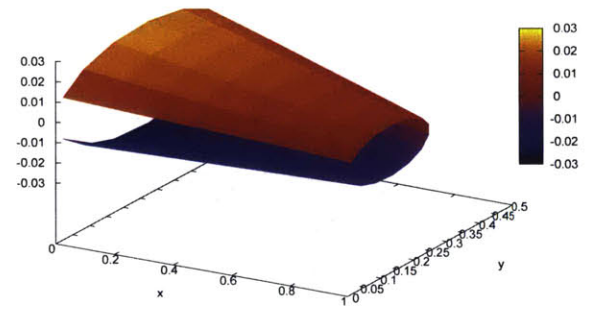


(a) Wing planform (b) Undeformed 3-view (c) Deformed 3-view

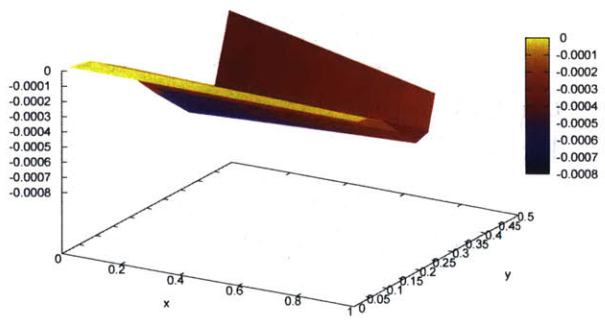
Figure 7-9: Wing Geometry

Table 7.7: Wing geometry parameters

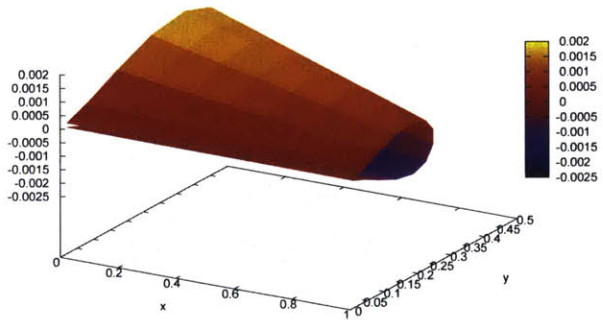
Parameter	Sym.	Value
Span	b	1
Root chord	c_r	0.5
Tip chord	c_t	0.25
Sweep		TE straight
Airfoil		Ellipse
Thickness/chord	t/c	10%
Radius of bend	κ	2



(a) x-stress (σ_{11})



(b) y-stress (σ_{22})



(c) Shear-stress (σ_{12})

Figure 7-10: Manufactured Bent Wing

7.4 2D Verification

HSM2D (Section 6) was implemented in C++ for a simple test case for verification with MES. It was created as a module in the Solution Adaptive Numerical Solver (SANS) framework developed by Darmofal, Allmaras, Galbraith, et al. at the MIT Aerospace Computational Design Laboratory (ACDL).

The first test case consists of a rectangular plate hanging under its own weight in a uniform gravitational field, as seen in Figure 7-11. The plate is isotropic and uniform in its stiffness, thickness, and density distributions.

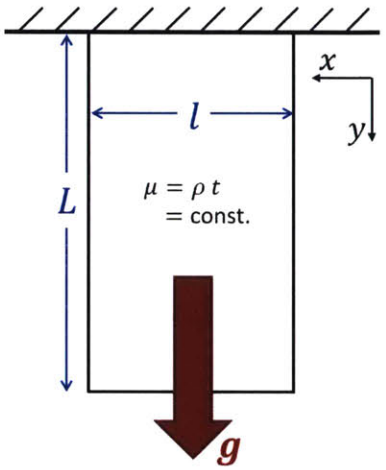


Figure 7-11: Hanging plate diagram

Table 7.8: Hanging plate parameters

param.	value
L	2
l	1
ρ	1,000
t	0.01
E	500,000
ν	0
g	10

We choose the case with Poisson ratio $\nu = 0$ because it decouples the extensional responses of the height and width of the bar. This allows gravity to stretch the bar in the y-dimension without changing shape in the x-direction so the width of the bar will remain uniform before and after deflection.

We begin with the force equilibrium equation (5.9), repeated here for convenience.

$$\nabla_0 \cdot \bar{\bar{\boldsymbol{\sigma}}} + \rho(\mathbf{g}-\mathbf{a}) = \mathbf{0} \tag{7.2}$$

We then use the shell lumping definitions (3.19) to convert from $\bar{\bar{\boldsymbol{\sigma}}}$ and ρ to $\bar{\bar{\mathbf{f}}}$ and μ. We also eliminate the acceleration field.

$$\nabla_0 \cdot \bar{\bar{\mathbf{f}}} + \mu\,\mathbf{g} = \mathbf{0} \tag{7.3}$$

We note that $\mathbf{g}$ only has a y_0-component, and thus $\nabla_0 \cdot \bar{\bar{\mathbf{f}}}$ will only have a y_0-component. The divergence term becomes a y_0-derivative in the single dimension.

$$\partial_{y_0} f_{22} + \mu\, g = 0 \tag{7.4}$$

We then integrate both sides in y_0 to get

$$f_{22}(y_0) - f_{22}(0) + \mu g y_0 = 0 \tag{7.5}$$

From our free-body diagram, we note that the base of the bar carries the entire weight of the bar. Remembering that the lumped shell stress f is a force per length, we calculate the y_0-direction lumped shell stress at the base of the block as

$$f_{22}(0) = \frac{F_{\text{grav}}}{\ell} = \frac{\mu g L \ell}{\ell} = \mu g L \tag{7.6}$$

The analytic solution for the lumped stress is then

$$f_{22}(y) = \mu g (L - y_0) \tag{7.7}$$

We then use the constitutive relations to map this stress to a strain. We begin with equation (5.13) repeated here for convenience.

$$\begin{Bmatrix} \varepsilon_{11(\mathbf{f})} \\ \varepsilon_{22(\mathbf{f})} \\ \varepsilon_{12(\mathbf{f})} \end{Bmatrix}_j = \begin{bmatrix} \\ \bar{\bar{A}} \\ \\ \end{bmatrix}_j^{-1} \begin{Bmatrix} f_{11} \\ f_{22} \\ f_{12} \end{Bmatrix}_j \tag{7.8}$$

Where we have dropped the $()^{\mathbf{e}}$ superscripts because we have specified extensional loading aligned with the y-direction and we expect no deformation in the x-direction, so we expect the local material directions $\hat{\mathbf{e}}_\alpha$ to remain fixed and co-linear with the x and y directions after deformation. Using (5.15) to define $\bar{\bar{A}}^{\mathbf{e}}$ in terms of E and ν, we note that the $\bar{\bar{\mathbf{c}}}$ matrix reduces to an identity when $\nu = 0$. We assume the Young's Modulus is constant through the thickness resulting in the following

$$\bar{\bar{A}} = E t \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7.9}$$

Since f_{22} is the only nonzero component of the stress tensor, equation (5.13) reduces to a simple Hookean 1D stress-strain relation. Introducing the stress function (7.7) and assuming a constant density ρ so $\mu = \rho t$ yields

$$\varepsilon_{22(f_{22})} = (E t)^{-1} f_{22} \tag{7.10}$$

$$\varepsilon_{22(f_{22(y_0)})} = \frac{\rho t g}{E t} (L - y_0) \tag{7.11}$$

$$= \frac{\rho g}{E} (L - y_0) \tag{7.12}$$

We then introduce the Almansi-Lagrange strain-displacement relation (5.20) repeated here for convenience.

$$\bar{\bar{\varepsilon}}_{(\mathbf{r})} = \frac{1}{2}\left[(\widetilde{\nabla}_0\mathbf{r})(\widetilde{\nabla}_0\mathbf{r})^T - \boldsymbol{I}_2\right] \tag{7.13}$$

From the definition of $\widetilde{\nabla}_0\mathbf{r}$ (5.19), the only nonzero term is $\partial_{y_0} y$, so the strain-displacement relation becomes

$$\varepsilon_{22(y)} = \frac{1}{2}\left[(\partial_{y_0} y)^2 - 1\right] \tag{7.14}$$

We solve for y by integrating in y_0. We incorporate the strain consistency equation (6.20) to set the constitutive strain (7.12) equal to the compatibility strain (7.14)

$$\partial_{y_0} y = \sqrt{2\varepsilon_{22} + 1} \tag{7.15}$$

$$= \sqrt{\frac{2\,\rho\, g}{E}\,(L - y_0) + 1} \tag{7.16}$$

$$= \sqrt{k\,(L - y_0) + 1} \tag{7.17}$$

With $k = 2\,\rho g/E$. We integrate both sides of (7.17) in y_0 from 0 to some y_0 with the boundary condition $y_0(0) = 0$ to get.

$$y_{(y_0)} = \frac{2}{3k}\left\{\sqrt{1 + k\,L} - \sqrt{1 + k\,(L - y_0)}\right.$$
$$\left. + k\left(y_0\sqrt{1 + k\,(L - y_0)} + L\left(\sqrt{1 + k\,L} - \sqrt{1 + k\,(L - y_0)}\right)\right)\right\} \tag{7.18}$$

This analytic result was compared against triangular and quadrilateral grids at varying resolution to test the error convergence rates. The test case rotated the domain by $\pi/8$ radians so that all terms in the state vector would be exercised in an asymmetric manner (e.g./ keeping the orientation as pictured in Figure 7-11 would only exercise the y and f_{22} terms, rotating by $\pi/2$ would only exercise the x and f_{11} terms, and rotating by $\pi/2$ would exercise the x, y and f_{11}, f_{22} pairs symmetrically). Results are shown in Figure 7-12. Note the similar response of x and y for both tri and quad element types. Also note the stresses are approximately 9-10 orders more accurate for rectangular grids than triangular grids. This is because the analytic solution (7.7) is linear in y_0 so the behavior can be captured exactly with a single quadrilateral element using hierarchical basis functions. It is noted that the expected $P+1$ order of accuracy is not achieved for even polynomial basis functions ($P2$ and $P4$). Additional stabilization is needed to stabilize the discretization and achieve the expected order of accuracy.

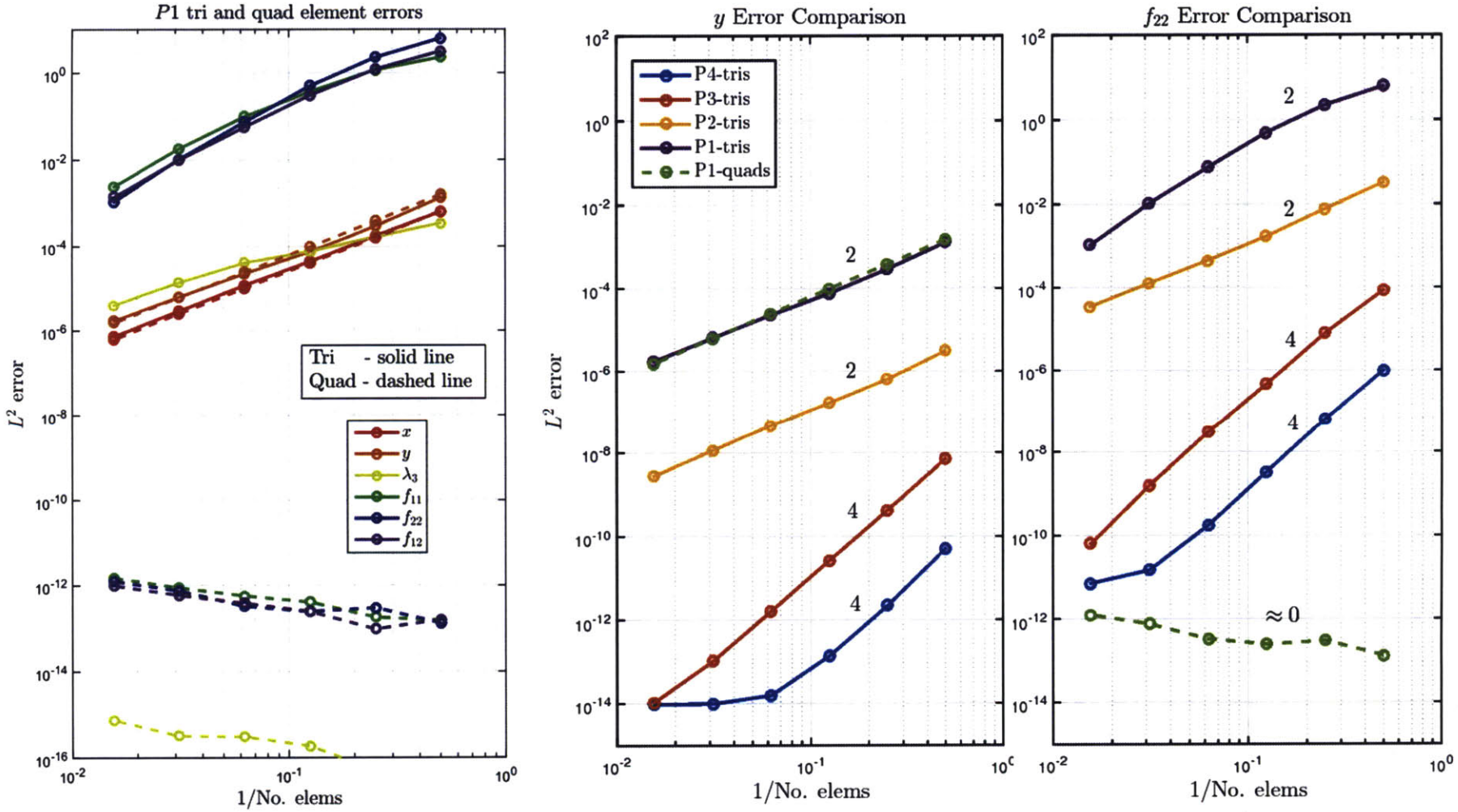


Figure 7-12: Error convergence for all state vector quantities (left) for P1 tri (solid line) and quad (dashed line) elements. Error convergence with increases in grid resolution for multiple polynomial orders and element shapes (right), for a displacement (y) and a stress (f_{22}). These two variables were chosen to highlight both "clusters" of variables seen in the left-hand plot.

Chapter 8

Optimization in 2D

8.1 Verification

We seek to verify optimization of the thickness distribution of the hanging plate via MES. We use the same plate and constant gravitational loading from the previous section, but now we seek to minimize the mass of the plate for a given tip deflection constraint. We simplify the problem by specifying a constant density, so the mass minimization becomes a volume minimization. To further simplify the optimization, we pose this constrained optimization problem as an unconstrained optimization problem by prescribing a penalty function for tip displacements other than the constraint. The objective function J is thus

$$J = \iint t \, \mathrm{d}\mathcal{A}_0 + K \left(L \, \ell \, t_L\right) \left(\frac{\delta_{tip}}{\delta_{spec}} - 1\right)^2 \tag{8.1}$$

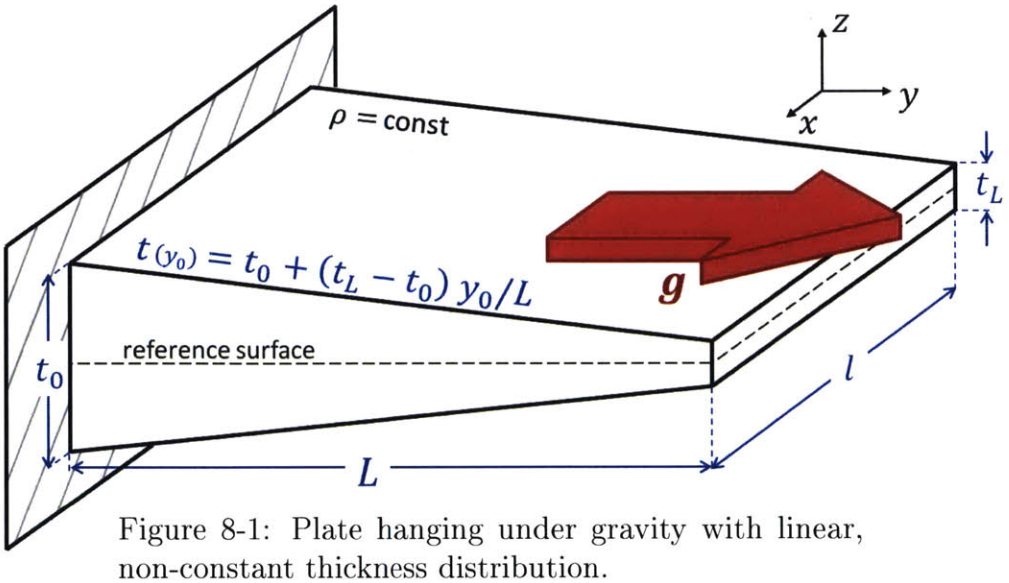


Figure 8-1: Plate hanging under gravity with linear, non-constant thickness distribution.

With plate of dimensions $L \times \ell \times t_{(y)}$, specified tip thickness t_L, and tip displacement δ_{tip}. The constant $(L\,\ell\,t_L)$ is included to convert the second term to units of volume to match the first term (the area integral of thickness). The tip deflection penalty K is thus a non-dimensional gain set to an arbitrarily large number ≈ 100.

We now assign a thickness distribution to the plate. We begin with a linear distribution.

$$t_{(y_0)} \;=\; t_0 + (t_L - t_0)\,\frac{y_0}{L} \tag{8.2}$$

Where the tip thickness t_L and plate length L are parameters, and the root thickness t_0 is the variable to be optimized.

The plate geometry is visible in Figure (8-1).

Table 8.1: Hanging plate opt. parameters

param.	value
L	1
l	1
ρ	1,000
t_L	0.01
E	500,000
ν	0
g	10
δ_{spec}	0.008
K	100

In the SANS (C++) implementation, the centerline single-point tip displacement is actually defined as the integral of the displacements along the tip edge weighted by an arbitrarily thin Gaussian, so that the centerline displacement values are weighted most heavily. However, for this test case with Poisson ratio $\nu = 0$, the cross-sections of the bar remain straight through the deformation, so every point along the tip edge are displaced the same amount. We therefore ignore the weighting complication in this analysis and proceed to plug in the definition of displacement $\delta_{tip} = y_{(L)} - L$. We may also eliminate the plate width ℓ as the area integral will be linear in ℓ.

$$J \;=\; \frac{1}{2}\,(t_0 + t_L) + K\,t_L \left(\frac{(y_{(L)} - L)}{\delta_{spec}} - 1 \right)^2 \tag{8.3}$$

With the new assumption that the thickness is nonuniform, the force equilibrium analysis form the previous section must be repeated. Equation (7.4) now becomes

$$\partial_{y_0} f_{22} \;+\; \rho\, t\, g \;=\; 0 \tag{8.4}$$

with $\mu = \rho t$. We solve this equation with the linear thickness distribution (8.2) and the boundary condition, similar to (7.6), that specifies the base of the plate supports the weight of the full plate.

$$f_{22}(0) \;=\; \rho\, t_{avg}\, g\, L \;=\; \frac{1}{2}\,\rho\, g\, L (t_0 + t_L) \tag{8.5}$$

We plug the solution of the differential equation and the linear thickness distribution into

the constitutive strain definition (7.10).

$$\varepsilon_{22(f_{22})} = (E\,t)^{-1}\,f_{22} \tag{8.6}$$

$$= \left\{E\left(t_0 + (t_L - t_0)\,\frac{y_0}{L}\right)\right\}^{-1} f_{22} \tag{8.7}$$

$$= \frac{\rho\,g}{2\,E}\left(\frac{(L - y_0)\left(L\,(t_0 + t_L) + y_0\,(t_L - t_0)\right)}{L\,t_0 + y_0\,(t_L - t_0)}\right) \tag{8.8}$$

We again set the two definitions of strain equal (6.20) so $\varepsilon_{22(\mathbf{r})} = \varepsilon_{22(\mathbf{f})}$ by solving the differential equation

$$2\,\varepsilon_{22(f_{22(y_0)})} = (\partial_{y_0} y)^2 - 1 \tag{8.9}$$

We take this opportunity to select values for the parameters, visible in Table (8.1).
This substitution yields:

$$\varepsilon_{22(f_{22(y_0)})} = -2\,\frac{(100\,t_0\,(y_0 - 1) - y_0 - 1)(y_0 - 1)}{100\,(100\,t_0\,(y_0 - 1) - y_0)} = (\partial_{y_0} y)^2 - 1 \tag{8.10}$$

The solution of (8.10), evaluated at $y_0 = L$ to yield the tip deflection, is too long to be included here. The plot of tip deflection vs root thickness is included in Figure 8-2.

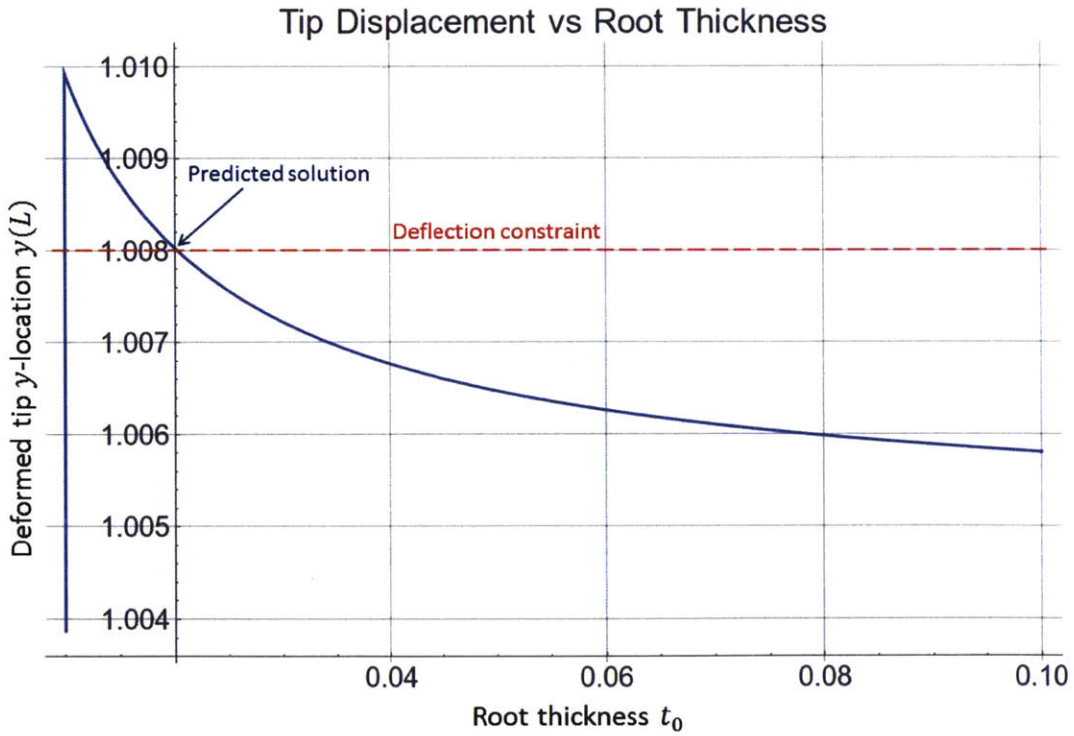


Figure 8-2: Tip displacement for gravitationally loaded plate as a function of root thickness for fixed tip thickness. Problem construction assumes $t_0 > t_L$ which is cause of discontinuity at $t_0 = t_L = 0.01$.

The motivation for setting the tip deflection constraint at $\delta_{spec} = 0.008$ also comes from this plot, as it conveniently occurs at the intersection of major gridlines on the plot. We then seek to minimize the objective function (8.3). We plug the displacement equation $y(y_0; t_0)$ evaluated at $y_0 = L$ into the objective function (8.3) with $K = 100$, $t_L = 0.01$, and $L = 1$. We also impose the "constraint" on tip deflection by setting $\delta_{spec} = 0.008$. We minimize $J_{(t_0)}$ by solving

$$\partial_{t_0} J = \partial_{t_0} \left\{ \frac{1}{2}(t_0 + 0.01) + \left(\frac{y(y_0; t_0) - 1}{0.008} - 1 \right)^2 \right\} = 0 \tag{8.11}$$

The formula for $\partial_{t_0} J$ is again too long to be included in this report, but a plot has again been included in Figure 8-3.

Solving (8.11) yields an optimum value for the root thickness of $t_0 = 0.0191013$.

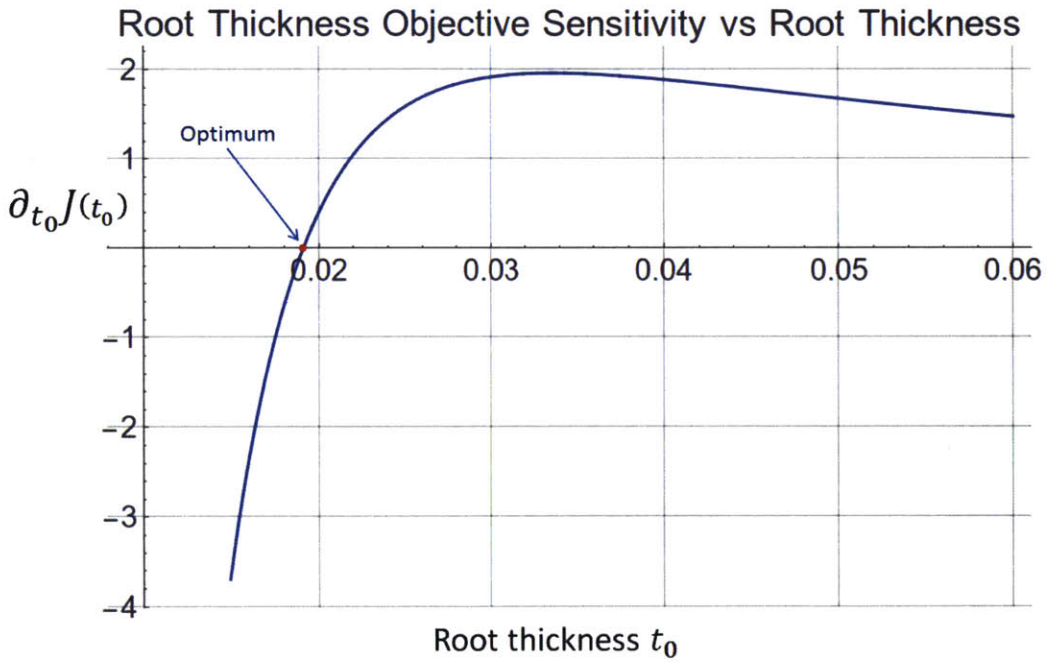


Figure 8-3: Objective function sensitivity to root thickness as a function of root thickness. Objective function is minimized at $\partial_{t_0} J = 0$.

HSM2D was implemented in C++ as described in Section 6. Sensitivities of the state variable outputs to the input parameters were made available as inputs to an external optimization library. Optimization was performed using the nlopt library [11]. The specific optimization algorithm employed was the method of moving asymptotes [28] which is a globally-convergent [1] method for gradient-based local optimization.

[1] guaranteed to converge to some local minimum of the objective function from any feasible starting point in the domain

A grid refinement study was performed to test that the optimum root thickness converged to the exact solution at the expected order of accuracy. Figure 8-4 shows that the optimization is indeed 2nd order accurate (for $P1$ quads).

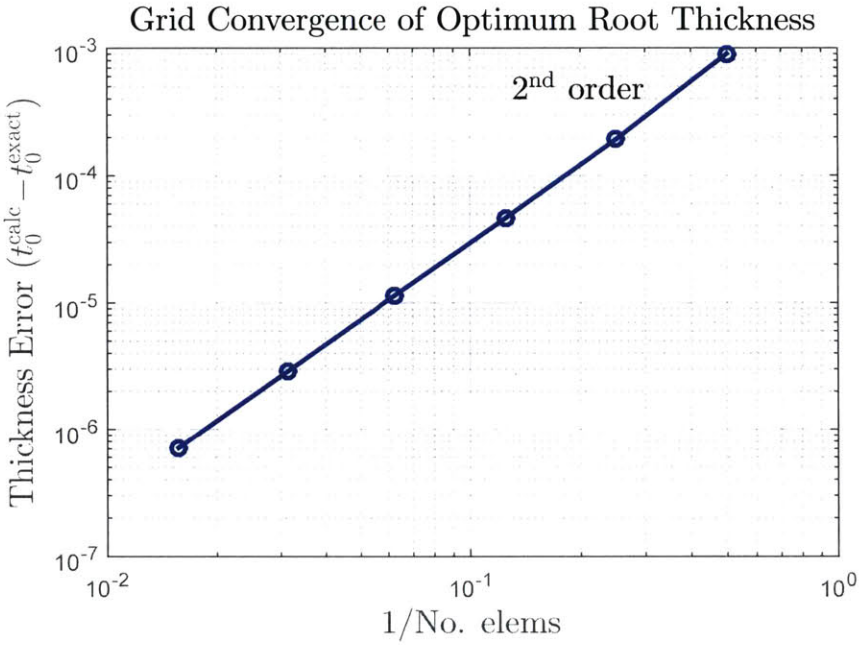


Figure 8-4: Optimum root thickness error for several grid resolutions using $P1$ quad elements.

8.2 Results

With the optimization validated in the linear-thickness hanging plate case, a more complicated configuration was optimized to exemplify the full feature set of HSM2D, including:

- Non-linear thickness distribution
- Non-constant acceleration
- Non-zero Poisson ratio
- Large deformation

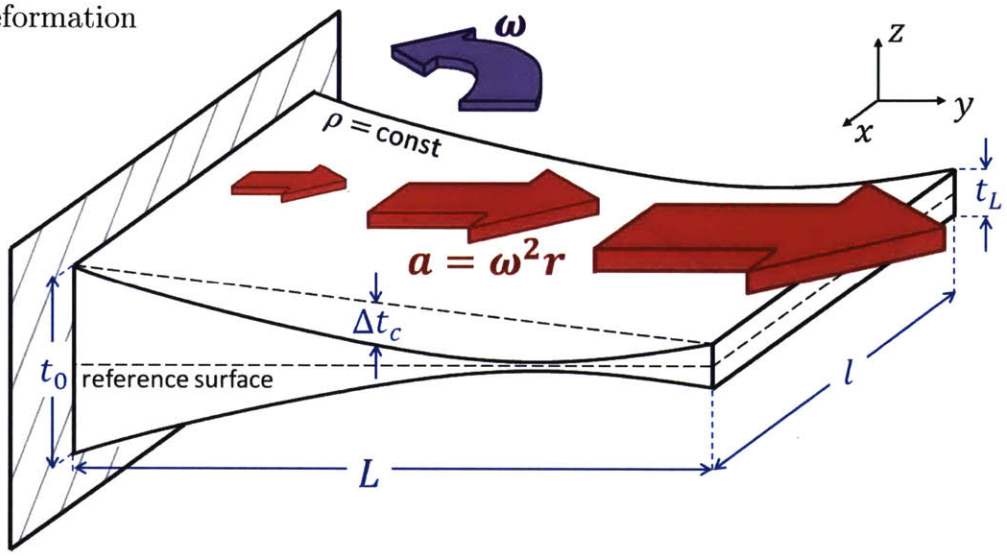


Figure 8-5: Spinning rotor blade with quadratic thickness distribution

The problem is that of a rotor blade subject to a large centrifugal force, as shown in Figure 8-5. The blade is clamped at the rotor hub and deforms according to the centrifugal force field proportional to the radial (spanwise) coordinate. The blade still has a constant density but now has a thickness distribution quadratic in y_0. This is implemented by using the hierarchical basis functions of Figure 8-6 to represent the blade thickness.

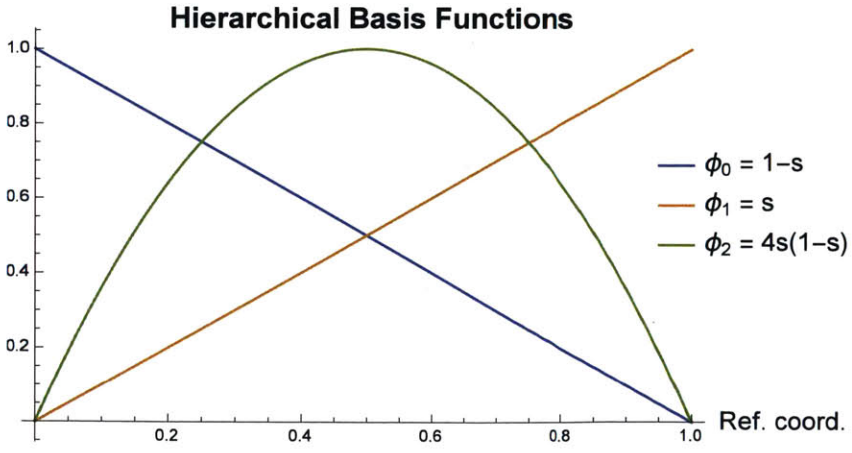


Figure 8-6: Hierarchical basis functions used to represent quadratic blade thickness. Note that, as implemented, the reference coordinate $s = 0$ corresponds to the tip of the blade $y_0 = 1$, and $s = 1$ corresponds to $y_0 = 0$

The geometry/loading of the blade, and the parameters for its thickness optimization, are summarized in Tables 8.2 and 8.3, respectively. The blade is rotating with frequency $f = 2\pi\omega$ of 3Hz, chosen so the average acceleration felt by the blade, $a_{avg} = \frac{1}{2}(a_0 + a_L) = a_L/2$ was approximately double that of the gravity in the optimization verification case (Section 8.1).

The constraint in this optimization is the tip deflection, increased to 10% of the span to prove the large deflection modeling ability of HSM. However, in the SANS (C++) implementation, the tip displacement parameter sensitivities are only available as weighted integrals of the displacements along the entire length of the tip edge. To isolate the centerline tip-edge displacement values, the a Gaussian weighting distribution was chosen such that the the middle 10% of the edge lies within three standard deviations of the mean. The Gaussian weighting function is

$$W_{(x_0)} = \left(2\sigma^2\pi\right)^{-\frac{1}{2}} \exp\left(-(x_0-\mu)^2/\left(2\sigma^2\right)\right) \tag{8.12}$$

where $\mu = \frac{1}{2}x_0^{\max}$ and $\mu = \frac{1}{60}x_0^{\max}$, and is visible in Figure 8-7.

Table 8.2: Rotor blade geometry/loading

param.	value
L	1
l	0.5
ρ	1,000
t_L	0.01
E	500,000
ν	0.3
g	0
f	3
a	$\omega^2 r$

To enable convergence of the non-linear solver, minimum values of the optimization variables were selected. The minimum quadratic basis function weight was selected as the value that would give the plate a minimum thickness of 5% of t_L when t_0 was also at a minimum.

$$\frac{\Delta t_c^{\min}}{t_L} = 0.95 \cdot \frac{-1}{4}\left(1 + 2\sqrt{\frac{t_0^{\min}}{t_L}} + \frac{t_0^{\min}}{t_L}\right) \tag{8.13}$$

The minimum t_0 was set at eight times larger than t_L, chosen purely heuristically.

The optimization took 30 iterations to converge at an optimum plate thickness of $t_0 = 0.0929$ and quadratic weight of $\Delta t_c = -0.0341$. The results can be seen in Figure 8-8. The large deformation to the desired tip deflection is clearly visible, as is the necking expected of a plate with non-zero Poisson ratio in uniaxial extension. As expected, the optimizer thickened the root and allowed a quadratic taper to the thin tip. The outwards flare of the sides and buldging of the tip edge is correctly captured by the in-plane rotation log-quaternion Λ_3. The x-direction tensile stress along the base and the concentrated shear stress at the base corners due to the clamped boundary condition are also captured correctly.

Table 8.3: Rotor blade opt. parameters

param.	value
K	100
N_x	16
N_y	32
δ_{spec}	0.1
$t_0^{\min}$	0.085
$\Delta t_c^{\min}$	-3.64×10^{-2}
σ	8.33×10^{-3}
μ	0.25

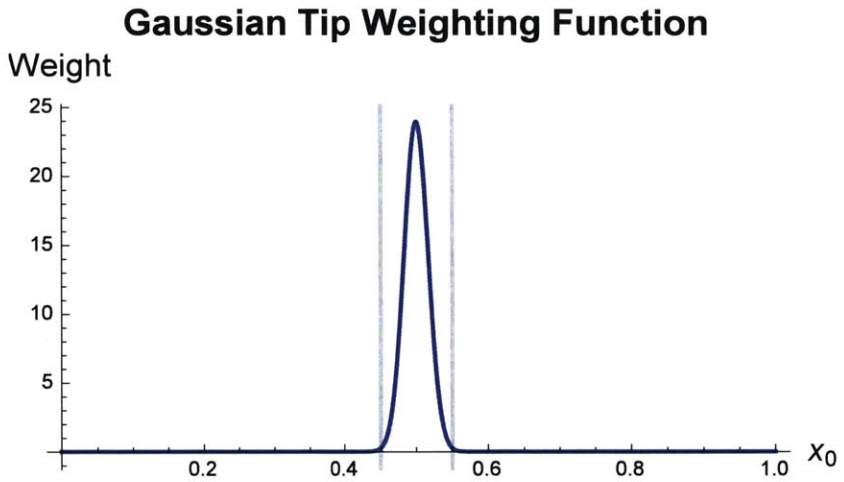


Figure 8-7: Gaussian weighting distribution to isolate tip-edge centerline deflection values. The third standard deviation aligns with the middle 10% of the edge, so that 99.7% of the weighting comes from that span.

8.3 Summary and Conclusions

The development of HSM is motivated by the need for an intermediate-fidelity structural model capable of capturing effects due to complex 3D geometries, large deformations, and anisotropic materials.

HSM is formulated on two-dimensional manifolds in three-dimensional space employing locally-Cartesian coordinate bases on which the governing equations are constructed. The governing equations include the in-plane and out-of-plane force and moment equilibrium equations, the equations enforcing the duality of the constitutive and compatibility strain definitions, and the equations bounding rotations to within the manifold. Both displacements *and* the stresses themselves are counted as primary unknowns in these equations which gives the "Hybrid" Shell Method its name. Structural parameters of three-dimensional objects are integrated, or *lumped*, through the thickness to idealize them as shells and reduce the complexity of the model.

The continuous HSM formulation is discretized with a Galerkin finite element method (FEM). The governing equations are cast in the weak form as weighted area integrals taken over the manifold surface. Spherical interpolation of the local Cartesian bases keeps vector-defined quantities consistent across a changing surface orientation without resorting to curvilinear coordinate systems.

HSM is simplified to HSM2D by restricting the geometry and loading to a plane for the purpose of model verification. The HSM2D boundary conditions (BCs) are cast in the weak

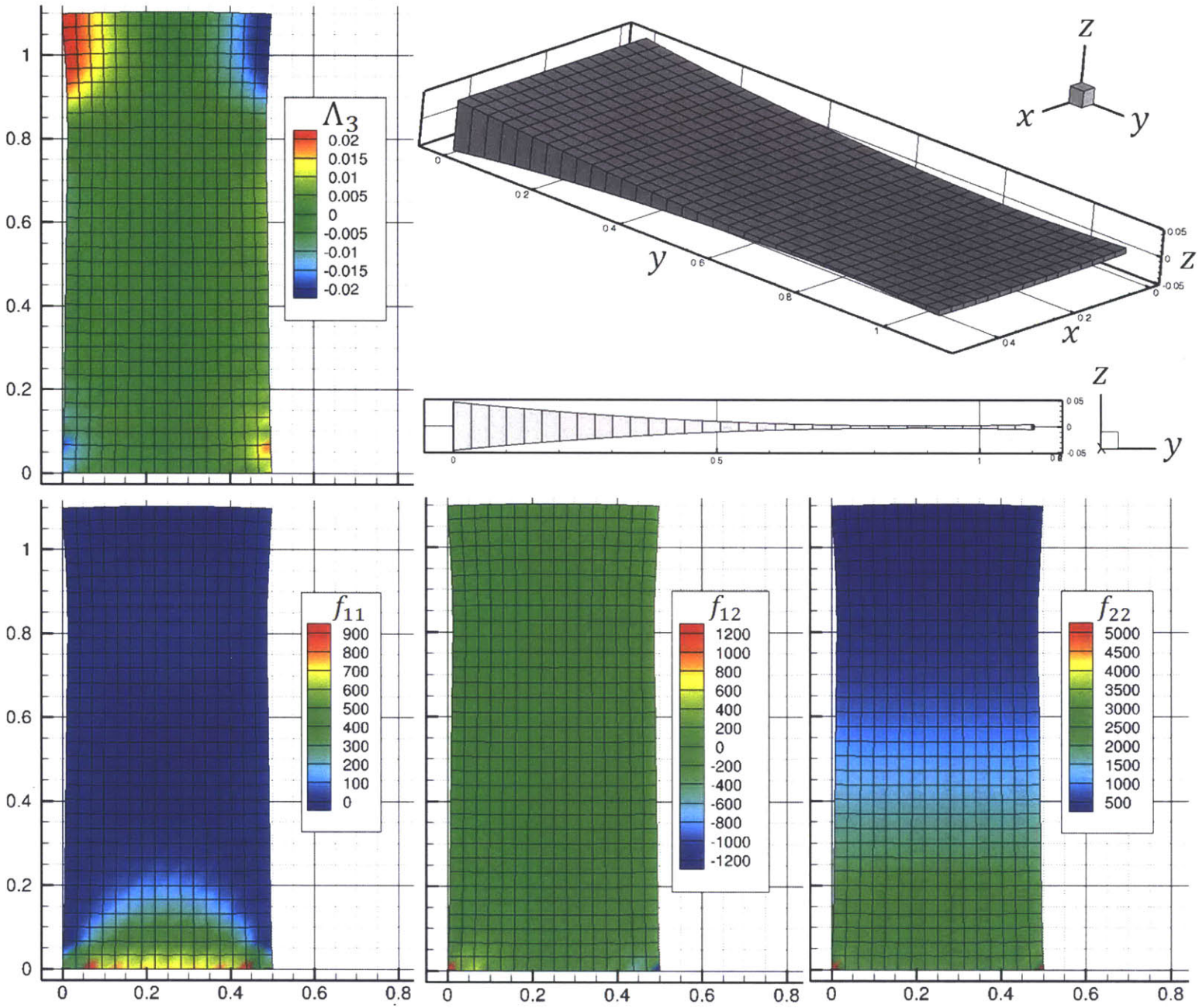


Figure 8-8: Optimized rotor blade results with views of deformed geometry and thickness distribution, in-plane rotation log quaternion (Λ_3), x-stress (f_{11}), y-stress (f_{22}), and shear stress (f_{12}).

form and imposed as additional weights to the governing equation residuals yielding an adjoint-consistent equation set.

The order-of-accuracy convergence for HSM2D, the adjoint-consistent plane-stress version of HSM, is verified by the Method of Exact Solutions (MES). $P1$ elements are found to be second order accurate for a a uniform plate hanging under its own weight with zero Poisson ratio. The optimization features of HSM2D are exercised on plate thickness distributions. The verification test case adds a linear thickness distribution to the hanging plate problem. The objective function is defined to minimize plate weight subject to a tip deflection constraint, imposed by adding a tip deflection penalty term to the objective function so the optimization is unconstrained. The root thickness (for a specified tip thickness) is optimized and is found to be second order accurate with $P1$ elements via verification with MES.

A final test case, a rotor blade with quadratic thickness distribution in a centrifugal force field, exercises the full feature set of HSM2D: non-uniform forcing, non-zero Poisson ratio, large deflection, and optimization of multiple parameters. The rapid calculation of this complex optimization provides proof of concept for HSM as an intermediate fidelity conceptual aircraft design tool.

Appendix A

Gauss Quadrature in Two-Dimensions

Two-dimensional Gauss quadrature, the method of calculating surface integrals by using a weighted sum of function values evaluated at specific locations within the the surface integral bounds, was used to define the L^2 norms used as the error metric in the grid convergence studies of Sections 7.3.1, 7.4, and 8.1. Both triangular and quadrilateral elements were considered.

A.1 Quadrilateral Elements

Gauss quadrature for an arbitrary quadrilateral element is a three step procedure [5]. The first step is to map the arbitrary quadrilateral to a square as shown in Figure A-1. Per

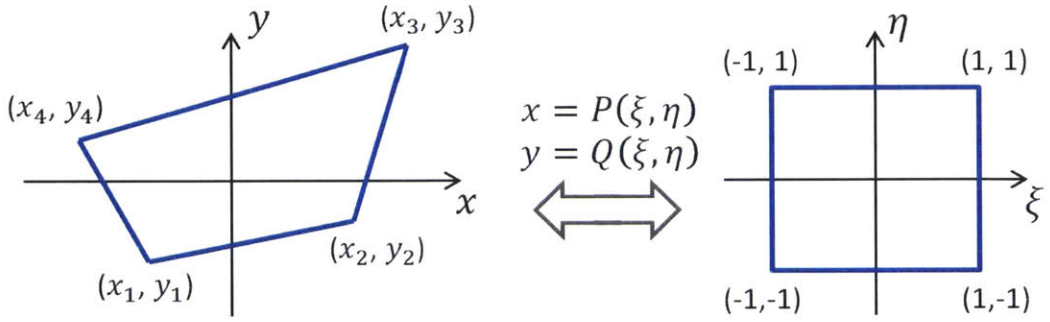


Figure A-1: Mapping of arbitrary quadrilateral element to reference square element

standard practice, this is accomplished with the nodal (bilinear) shape functions, $N_{j(\xi,\eta)}$, of Equations 4.15. The mapping is then:

$$x = P_{(\xi,\eta)} = \sum_{i=1}^{4} x_i \, N_{i(\xi,\eta)} \tag{A.1}$$

$$y = Q_{(\xi,\eta)} = \sum_{i=1}^{4} y_i \, N_{i(\xi,\eta)} \tag{A.2}$$

With the Jacobian of the transformation defined by

$$J_{(\xi,\eta)} = \frac{\partial(x,y)}{\partial(\xi,\eta)} = \begin{bmatrix} \partial_\xi x & \partial_\xi y \\ \partial_\eta x & \partial_\eta y \end{bmatrix} \tag{A.3}$$

The second step is to calculate the quadrature weights and nodes[1]. Finally, the squared error from the analytical solution at each node is weighted accordingly and summed. Since the values of the modeled solution are only available at the element nodes, they must be interpolated to find the values at each quadrature point [2].

The equation for the N order Gauss quadrature of a quadrilateral element K is

$$\iint_K F_{(x,y)}\,\mathrm{d}x\,\mathrm{d}y \approx \sum_{i=1}^{N}\sum_{j=1}^{N} w_i\, w_j F\left(P\left(\xi_i,\xi_j\right), Q\left(\xi_i,\xi_j\right)\right)\left|J\left(\xi_i,\xi_j\right)\right| \tag{A.4}$$

A.2 Triangular Elements

Gauss quadrature for triangular elements is similar to that for quadrilateral elements. In this case, the mapping is from arbitrary triangle to standard, right triangle, as shown in Figure A-2.

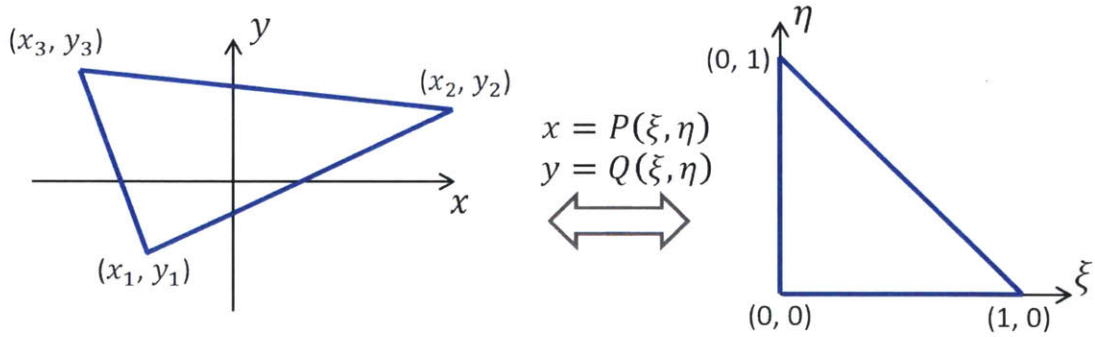


Figure A-2: Mapping of arbitrary triangular element to reference right-triangle element

This is again accomplished with nodal functions. For the triangle, the nodal functions, $M_{j(\xi,\eta)}$, are

$$\begin{aligned} M_{1(\xi,\eta)} &\equiv 1-\xi-\eta \\ M_{2(\xi,\eta)} &\equiv \xi \\ M_{3(\xi,\eta)} &\equiv \eta \end{aligned} \tag{A.5}$$

[1] Accomplished in Section 7.3.1 via the `lgwt.m` MATLAB function from Greg von Winckel and in Sections 7.4 and 8.1 with native methods in SANS

[2] Accomplished in Section 7.3.1 via the `griddata.m` MATLAB function

The mapping is then

$$x = P_{(\xi,\eta)} = \sum_{i=1}^{3} x_i \, M_{i(\xi,\eta)} \tag{A.6}$$

$$y = Q_{(\xi,\eta)} = \sum_{i=1}^{3} y_i \, M_{i(\xi,\eta)} \tag{A.7}$$

The Jacobian of this mapping can be found to be $J_{(\xi,\eta)} = 2A_K$ where A_K is the area of element K, and is calculated via the shoelace formula (shown below for an n-point polygon)

$$A = \frac{1}{2}\left|\sum_{i=1}^{n} \det\begin{pmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{pmatrix}\right| \tag{A.8}$$

The quadrature weights and points are different for the reference triangle than for the reference quadrilateral element [3].

The equation for the N^{th} order Gauss quadrature of a triangular element K is

$$\iint_K F_{(x,y)} \, \mathrm{d}x \, \mathrm{d}y \approx A_k \sum_{i=1}^{N} w_i \, F\big(P\left(\xi_i, \eta_i\right), Q\left(\xi_i, \eta_i\right)\big) \tag{A.9}$$

[3] Accomplished in Section 7.3.1 via the `TriGaussPoints.m` function from Shaozhong Deng [5] and in Sections 7.4 and 8.1 with native methods in SANS

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix B

Discussion of Solidworks Solver Algorithms

The information concerning which algorithms Solidworks actually uses for their shell models is proprietary to Dassault Systemes. The publically-available theory manual [29], while giving comprehensive background on FEA in general, does not identify which software options employ which equations. To gain insight, a brief study was performed with the same plate geometry, material properties, and loading as Table 7.1 to test the effect of each different solver option on the maximum deformation of the square plate (with a constant mesh size). The options include thick vs thin shell, small displacement vs large displacement, linear vs nonlinear modeling, and constant direction vs norm following loads. The results are shown in Figures B-1 and B-2.

It is noted from Figure B-1 that there is little difference between the thick and thin shell options. As discussed in Section 2.2, the difference in application between the two dominant plate deformation models is that KLPT is for thin plates and MPT is for thick plates. This test case was specifically selected so the thin-plate approximation of KLPT would apply (in order for the KLPT analytic solution to be compared against the Solidworks solution in Section 7.3.1). Therefore, we don't expect to see a difference in thick and thin modeling options.

It appears then, that the "Thin shell - Small Displacement" option employs KLPT and the "Thick shell - Small Displacement" option employs MPT. Further investigation of the "Large Displacement" options is shown in Figure B-2.

It can be assumed the large displacement options use the full non-linear shear deformation equations and are solved in multiple time-steps to allow large deformations. The non-linear option, as expected, results in smaller max-deflections than the linear option because it most-likely takes the increased shear stiffness at large strains into account. Furthermore, the norm-following results in smaller max-deflections than constant-direction loading, because

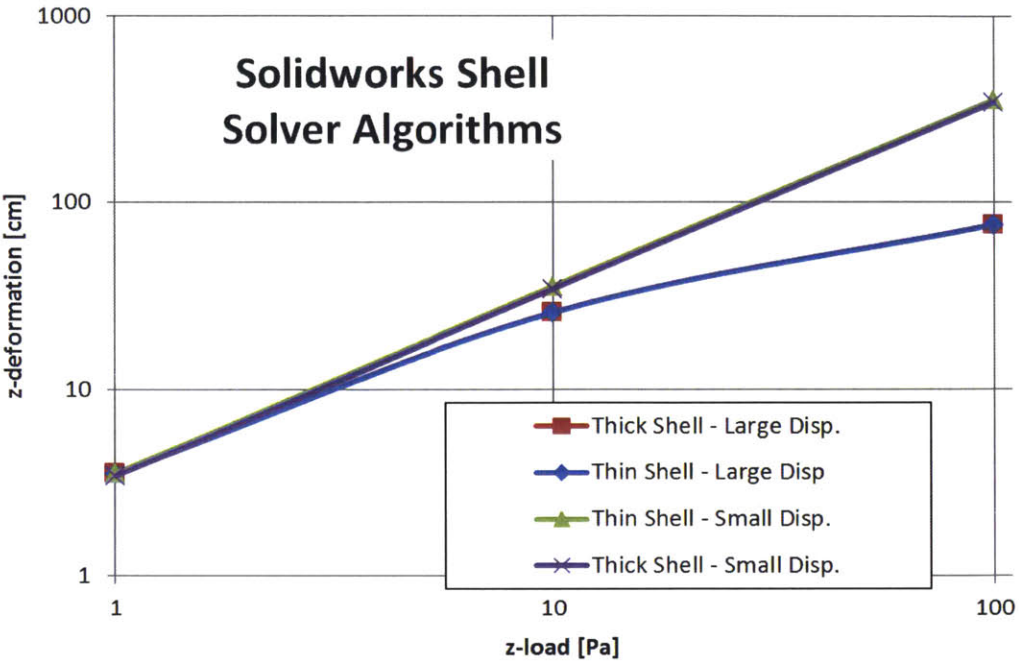


Figure B-1: Comparison of thick vs thin shell and large vs small displacement options for a transversely-loaded SS square plate in Solidworks

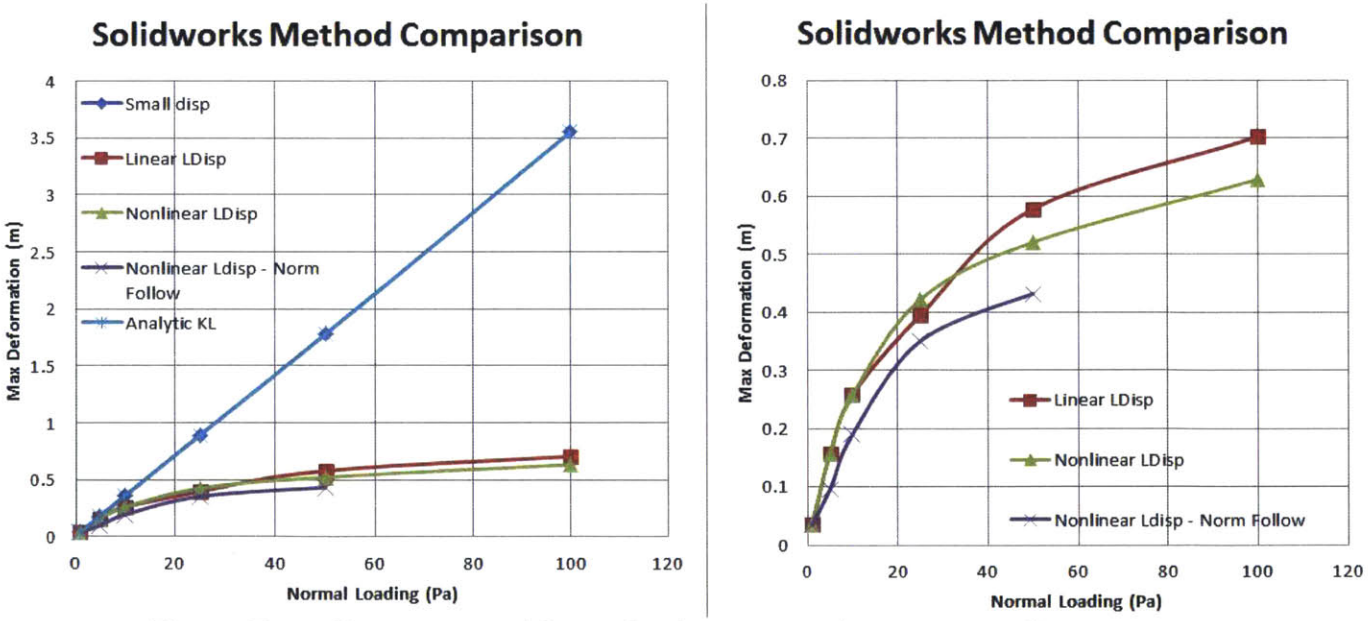


Figure B-2: Comparison of large displacement solver options, linear vs non-linear and fixed-direction vs norm-following load, for a transversely-loaded SS square plate in Solidworks

at large deflections, a non-trivial fraction of the initially transverse force no longer points in the transverse direction.

Appendix C

Analytical Solutions for Plate Deformations

Several sources have performed the Kirchoff-Love analysis for a rectangular plate with dimensions $0 \leq x \leq a$ and $0 \leq y \leq b$, pinned on all edges, subjected to a uniform transverse pressure load. eFunda [10] gives the bending deflection of a plate of flexural rigidity $D = (E\,t^3)/\left(12\,(1-\nu^2)\right)$ subjected to a uniform traverse load p as

$$w_{(x,y)} = \sum_{m=1,3,\ldots}^{\infty} \frac{16p}{\pi^6 D} \sum_{n=1,3,\ldots}^{\infty} \frac{\sin\left(\pi\, x\, \frac{m}{a}\right) \sin\left(\pi\, y\, \frac{n}{b}\right)}{m\, n \left(\left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2\right)} \tag{C.1}$$

Liu and Li [14] studied thin, isotropic plates based on classical thin plate theory. They determined the bending deflection of a rectangular plate subjected to a uniform transverse load with two adjacent edges simply supported and the others slidingly-supported to be

$$\begin{aligned} w_{(x,y)} = \sum_{n=1,3,\ldots}^{\infty} & 32\, p\, a^3 \sin\left(\tfrac{n\pi}{2}\right) \cos\left(\tfrac{\alpha_n x}{b}\right) \left[D n^5 \pi^5 \left(1+\cosh\left(2\alpha_n\right)\right)\right]^{-1} \\ & \left[2\,a\left(1+\cosh\left(2\alpha_n\right)\right) + n\,\pi\, y \cosh\left(\alpha_n\right) \sinh\left(\tfrac{\alpha_n y}{b}\right)\right. \\ & \left. - \cosh\left(\tfrac{\alpha_n y}{b}\right)\left(4\,a\cosh\left(\alpha_n\right) + 2\,a\,\alpha_n \sinh\left(\alpha_n\right)\right)\right] \end{aligned} \tag{C.2}$$

where $\alpha_n = b\,n\pi/(2a)$

eFunda [10] also gives the analytical solution for a circular plate of radius R subject to a uniform transverse pressure load p.

$$w(r) = \frac{p\,R^4}{64\,D\,(1+\nu)} \left[2\,(3+\nu)\left(1-\left(\frac{r}{R}\right)^2\right) - (1+\nu)\left(1-\left(\frac{r}{R}\right)^4\right)\right] \tag{C.3}$$

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix D

Composite Laminate Theory

D.1 Overview

Lamination parameters describe the structural properties (in-plane, bending, and extension-bending coupling stiffnesses) of a composite laminate without having to maintain the detail of the material, thickness, and fiber-angle of each layer of the composite stack. The classical stiffness tensor, consisting of the $\bar{\bar{\mathbf{A}}}, \bar{\bar{\mathbf{B}}}, \bar{\bar{\mathbf{D}}}$ sub-matrices, governs the material deformation via the stress-strain and constitutive relations. For a composite stack, these relations can be decoupled into "material dependent" and "geometry dependent" effects. The material dependent effects are summarized by the five *Tsai-Pagano parameters*, which are functions purely of the elastic and shear moduli and Poisson's ratios of the materials. The geometry dependent effects are then summarized by the 15 *lamination parameters*, which are functions of the number, thicknesses, and fiber orientations of the composite plies. This number of lamination parameters is sufficient regardless of the complexity of the composite layup. Furthermore, three of the lamination parameters are either constant or functions of the total thickness alone, so the geometric dependent effects of a constant-thickness, arbitrarily complex, composite laminate is summarized by only 12 parameters.

Use of the Tsai-Pagano and lamination parameters considerably simplifies a lamination composite optimization problem because the feasible regions of lamination parameters (i.e. lamination parameter combinations that correspond to physically realizable layup sequences) is convex. The feasibility region for 0, 90, ± 30, ± 45, and ± 60 degree plies has been derived, validated, and confirmed in previous work[15]. This method creates hyperplane-bounded convex hulls and then groups the three different feasibility regions created from the in-plane (A-matrix), coupling(B-matrix), and out-of-plane (D-matrix) stiffness elements.

Finding the optimum laminate stacking sequence for a given set of constraints is a two-step optimization process. Once the feasibility region has been created, the first step is to search the convex feasible space for the optimum lamination parameters. Then a second optimiza-

tion must be done to realize a physical stacking sequence from the optimum lamination parameters (since multiple stacking sequences can yield the same set of lamination parameters). The lamination parameter optimization can utilize inverse-gradient based methods but the stacking sequence optimization is necessarily discrete because it involves an integer number of layers. Previous implementations have shown success in these discrete optimizations with particle swarm[3] and genetic[1] algorithms. It is also possible to approximate the discretized problem with a continuous domain. The optimized number of layers will involve fractional layers which are rounded to the nearest integer post optimization. Previous studies have shown this method to nearly approximate the optimum stacking sequence[34], but the post-process rounding implies the true optimum is not being found.

D.2 Nomenclature

A composite stacking sequence can be very complex, with dozens of layers each at a unique fiber angle, so the industry has developed a compact nomenclature[34]. Composites are described beginning with the top (outermost) layer of a stack. The simplifying assumption in this nomenclature (and indeed the analysis and design methods described in later sections) is that all layers are equally thick and composed of the same material. A composite with N layers is represented as

$$[\theta_1/\theta_2/\cdots/\theta_N]$$

Where the top layer has fiber angle θ_1 and the bottom layer has angle θ_N (both with respect to some reference orientation). *On-axis* layers have $\theta = 0°$ or $\theta = 90°$. *Off-axis* layers can have any other angle of fiber orientation.

Repeated layers, or groups of layers, are represented by subscripts denoting the number of repetitions. Repeating layer sequences are grouped with parentheses, as in the following examples:

$$\begin{aligned} [0_2/45/60_3] &= [0/0/45/60/60/60] \\ [0/(45/60_2)_2] &= [0/45/60/60/45/60/60] \\ [0/45/60]_2 &= [0/45/60/0/45/60] \end{aligned}$$

Many composites are symmetric about the centerline. Such composites are abbreviated with a subscript s after the brackets. Note that any symmetric composite has an even number of

layers. Symmetric stacks can also have repeated subgroups as in the following examples:

$$\begin{aligned}
[0/45/60]_s &= [0/45/60/60/45/0] \\
[0/(45/60)_2]_s &= [0/45/60/45/60/60/45/60/45/0] \\
[0/45/60]_{2s} &= [0/45/60/0/45/60/60/45/0/60/45/0]
\end{aligned}$$

Another category of laminates are *balanced laminates* which have a ply oriented at angle θ for each occurrence of a $-\theta$ layer. These layers opposite layers need not be stacked adjacently, but in practice they often are. Such laminates are abbreviated with $\pm$ or $\mp$ as follows:

$$\begin{aligned}
[\pm 15/\mp 30] &= [15/\text{-}15/\text{-}30/30] \\
[\pm 15/\mp 30]_s &= [15/\text{-}15/\text{-}30/30/30/\text{-}30/\text{-}15/15]
\end{aligned}$$

D.3 Derivation and Relation to Classical Stiffness Matrix

The following derivation is an abbreviated form of the derivation in [34] with some edits.

D.3.1 Stress-strain Relations for Rotated Orthotropic Material

The stress-strain relation for an orthotropic material in plane stress (thin section) is:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & 0 \\ Q_{12} & Q_{22} & 0 \\ 0 & 0 & Q_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix} = \bar{\bar{\mathbf{Q}}} \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix} \tag{D.1}$$

where:

$$Q_{11} = \frac{E_{11}^2}{E_{11} - E_{22}\,\nu_{12}^2} \tag{D.2}$$

$$Q_{22} = \frac{E_{11} E_{22}}{E_{11} - E_{22}\,\nu_{12}^2} \tag{D.3}$$

$$Q_{12} = \nu_{12}\, Q_{22} \tag{D.4}$$

$$Q_{66} = G_{12} \tag{D.5}$$

The transformation matrix for a rotation through an angle θ (from axes x, y to axes 1, 2) is:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \begin{bmatrix} \cos^2(\theta) & \sin^2(\theta) & 2\cos(\theta)\sin(\theta) \\ \sin^2(\theta) & \cos^2(\theta) & -2\cos(\theta) \\ -\cos(\theta)\sin(\theta) & \cos(\theta)\sin(\theta) & \cos^2(\theta)-\sin^2(\theta) \end{bmatrix} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \bar{\bar{\mathbf{T}}} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} \tag{D.6}$$

and similarly

$$\begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_{12} \end{Bmatrix} = \bar{\bar{\mathbf{T}}} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_{xy} \end{Bmatrix} \tag{D.7}$$

The conversion between tensor shear strains, ε, and engineering shear strains, γ, is:

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_{xy} \end{Bmatrix} = \bar{\bar{\mathbf{R}}} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_{xy} \end{Bmatrix} \tag{D.8}$$

All together, for an orthotropic material rotated by an angle θ, the transformed stress-strain relation is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \bar{\bar{\mathbf{T}}}^{-1} \bar{\bar{\mathbf{Q}}} \bar{\bar{\mathbf{R}}} \bar{\bar{\mathbf{T}}} \bar{\bar{\mathbf{R}}}^{-1} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \bar{\mathbf{Q}} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} \\ \bar{Q}_{12} & \bar{Q}_{22} & \bar{Q}_{26} \\ \bar{Q}_{16} & \bar{Q}_{26} & \bar{Q}_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} \tag{D.9}$$

Where $\bar{Q}_{ij} = f(E, \nu, G, \theta)$

The combination of the stress-strain relations and the constitutive relations yield the forces and moment resultants through the classical $\bar{\bar{\mathbf{A}}}, \bar{\bar{\mathbf{B}}}, \bar{\bar{\mathbf{D}}}$ stiffness tensor (which makes the Kirchoff-Love assumption of zero in-plane deformations of the mid-plane in bending):

$$\left\{\begin{array}{c} f_x \\ f_y \\ f_{xy} \\ \hline m_x \\ m_y \\ m_{xy} \end{array}\right\} = \begin{bmatrix} \bar{\bar{\mathbf{A}}} & \bar{\bar{\mathbf{B}}} \\ \bar{\bar{\mathbf{B}}} & \bar{\bar{\mathbf{D}}} \end{bmatrix} \left\{\begin{array}{c} \varepsilon_x \\ \varepsilon_y \\ \gamma_{12} \\ \hline \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{array}\right\} = \left[\begin{array}{ccc|ccc} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ \cdot & A_{22} & A_{26} & \cdot & B_{22} & B_{26} \\ \cdot & \cdot & A_{66} & \cdot & \cdot & B_{66} \\ \hline B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ \cdot & B_{22} & B_{26} & \cdot & D_{22} & D_{26} \\ \cdot & \cdot & B_{66} & \cdot & \cdot & D_{66} \end{array}\right] \left\{\begin{array}{c} \varepsilon_x \\ \varepsilon_y \\ \gamma_{12} \\ \hline \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{array}\right\} \tag{D.10}$$

D.3.2 Lamination Parameters and the Stiffness Matrix

For a laminate of orthotropic plies, the material properties are found by integrating the transformed stiffnesses $\bar{Q}_{ij}$ through the thickness of the stack. Thus the $\bar{\bar{\mathbf{A}}}, \bar{\bar{\mathbf{B}}}, \bar{\bar{\mathbf{D}}}$ stiffness matrices can be expressed as a summation of the contributions of each layer. The integrals

can be reduced to summations because the fiber orientation angle θ is constant through each layer. For a laminate of N layers and total height h, where the k^{th} layer has thickness $t_k = z_k - z_{k-1}$, top-plane z-coordinate z_k, and mid-plane z-coordinate $\bar{z_k} = \frac{1}{2}(z_k + z_{k+1})$, the stiffness terms are:

$$A_{ij} = \sum_{k=1}^{N} \left(\bar{Q}_{ij}\right)_{(k)} (z_k - z_{k-1}) \tag{D.11}$$

$$B_{ij} = \frac{1}{2} \sum_{k=1}^{N} \left(\bar{Q}_{ij}\right)_{(k)} \left(z_k^2 - z_{k-1}^2\right) \tag{D.12}$$

$$D_{ij} = \frac{1}{3} \sum_{k=1}^{N} \left(\bar{Q}_{ij}\right)_{(k)} \left(z_k^3 - z_{k-1}^3\right) \tag{D.13}$$

The lamination parameter formulation is used to decouple the geometry dependency from the material property dependency. This allows optimization over a smaller design space since geometric properties can be lumped through the thickness. The first step is to define the Tsai-Pagano parameters, also known as the *material invariants*, which are functions of the material properties Q_{ij} alone.

$$U_1 = \frac{1}{8}(3\,Q_{11} + 3\,Q_{22} + 2\,Q_{12} + 4\,Q_{66}) \tag{D.14}$$

$$U_2 = \frac{1}{2}(Q_{11} - Q_{22}) \tag{D.15}$$

$$U_3 = \frac{1}{8}(Q_{11} + Q_{22} - 2\,Q_{12} - 4\,Q_{66}) \tag{D.16}$$

$$U_4 = \frac{1}{8}(Q_{11} + Q_{22} + 6\,Q_{12} - 4\,Q_{66}) \tag{D.17}$$

$$U_5 = \frac{1}{8}(Q_{11} + Q_{22} - 2\,Q_{12} + 4\,Q_{66}) \tag{D.18}$$

The geometric, or ply-dependent, parameters are then:

$$V_{0\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \left\{h, 0, \frac{h^3}{12}\right\} \tag{D.19}$$

$$V_{1\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} t_k \cos(2\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.20}$$

$$V_{2\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} t_k \sin(2\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.21}$$

$$V_{3\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} t_k \cos(4\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.22}$$

$$V_{4\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} t_k \sin(4\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.23}$$

where

$$d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \left\{1,\ \bar{z_k},\ \left(z_k^2 - 2\,z_k\,z_{k-1} + z_{k-1}^2\right)\right\} \tag{D.24}$$

represents the dimensionality of the $\bar{\bar{\mathbf{A}}}, \bar{\bar{\mathbf{B}}}, \bar{\bar{\mathbf{D}}}$ tensors after discretization of the integral $\int_{z_{k-1}}^{z_k}\{1,\ z,\ ,z^2\}dz$ through constant fiber orientation layers.

If each layer of the laminate has the same thickness ($t_k = t$), the layer thicknesses can be moved outside the summation. The geometric terms can then be further simplified by normalizing by the total thickness, $h = Nt$ [17]. This formulation yields the lamination parameters, where $\xi_n = \frac{V_n}{h}$:

$$\xi_{0\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \left\{1, 0, \frac{h^2}{12}\right\} \tag{D.25}$$

$$\xi_{1\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} \cos(2\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.26}$$

$$\xi_{2\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} \sin(2\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.27}$$

$$\xi_{3\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} \cos(4\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.28}$$

$$\xi_{4\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} = \sum_{k=1}^{N} \sin(4\,\theta_{(k)})\, d_{k\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \tag{D.29}$$

The $\bar{\bar{\mathbf{A}}}, \bar{\bar{\mathbf{B}}}, \bar{\bar{\mathbf{D}}}$ stiffness matrices may then be reconstructed as the combination of the material properties and geometric configuration:

$$\begin{bmatrix} \{A_{11}, B_{11}, D_{11}\} \\ \{A_{22}, B_{22}, D_{22}\} \\ \{A_{12}, B_{12}, D_{12}\} \\ \{A_{66}, B_{66}, D_{66}\} \\ \{A_{16}, B_{16}, D_{16}\} \\ \{A_{26}, B_{26}, D_{26}\} \end{bmatrix} = h \begin{bmatrix} U_1 & U_2 & 0 & U_3 & 0 \\ U_1 & -U_2 & 0 & U_3 & 0 \\ U_4 & 0 & 0 & -U_3 & 0 \\ U_5 & 0 & 0 & -U_3 & 0 \\ 0 & 0 & U_2 & 0 & 0 \\ 0 & 0 & U_2 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \xi_{0\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \\ \xi_{1\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \\ \xi_{2\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \\ \xi_{3\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \\ \xi_{4\{\bar{\bar{\mathbf{A}}},\bar{\bar{\mathbf{B}}},\bar{\bar{\mathbf{D}}}\}} \end{Bmatrix} \tag{D.30}$$

D.4 Properties

There are several rules-of-thumb with which a composite designer should be familiar:

- A composite in uniaxial tension will have stiffness properties that are the unweighted-average of all the layers. In bending, due to the large z-dependence of section modulus, the stiffness properties of the outer layers have a far greater effect than those of the inner layers.
- There is no coupling ($\bar{\bar{\mathbf{B}}} = 0$) between in-plane and bending deformations of a symmetric laminate.
- Extension-twist coupling (B_{16} and B_{26}) are caused by off-axis ($\theta \neq 0$ or 90) layers that are not symmetric with respect to the laminate mid-plane. If the laminate is made of isotropic layers, then $B_{16} = B_{26} = 0$ even for asymmetric laminates.
- Bending-twist coupling (D_{16} and D_{26}) exists for all laminates with off-axis layers. Balancing the laminate (with $\pm$ layers) can make this coupling smaller. The closer together the balancing layers are, the smaller the coupling will be.
- Balanced laminates will not have shear-extension coupling (A_{16} and A_{26}) regardless of how far apart the balancing layers are.

D.5 Takeaways

- 15 parameters can describe the geometry of any composite laminate of a single material.
- The of the geometric parameters ($\xi_{0\{\bar{\mathbf{A}},\bar{\mathbf{B}},\bar{\mathbf{D}}\}}$) are constants or functions of the total stack thickness alone, leaving **12 geometry dependent parameters** for composite of a given thickness.
- The design space for realizable lamination parameters is **always convex**.
- There is **no direct mapping** from lamination parameters to physical ply lay-up, optimization routines must be used.
- Lamination parameters may use **continuous** optimization strategies, while the lay-up optimization is necessarily **discrete**.
- PSO and GA methods have been successfully used to generate optimized laminate stacks from given lamination parameters.

Appendix E

Dyadic Notation

This section was developed by Steven Allmaras [2] and shared with the author via offline communication.

Let $\hat{\imath}$ and $\hat{\jmath}$ be unit vectors in x and y, respectively. Operations are dot or inner products (e.g. $\hat{\imath} \cdot \hat{\jmath} = 0$), and outer products (e.g. $\hat{\imath}\hat{\jmath}$). Outer products of unit vectors represent higher order tensors.

Matrix representation of 2-rank tensors,

$$\hat{\imath}\hat{\imath} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad \hat{\imath}\hat{\jmath} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \qquad \hat{\jmath}\hat{\imath} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \qquad \hat{\jmath}\hat{\jmath} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \tag{E.1}$$

Identity matrix,

$$\bar{\bar{\mathbf{I}}}_2 = \hat{\imath}\hat{\imath} + \hat{\jmath}\hat{\jmath} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{E.2}$$

Consider generic vector (1-rank tensor) and matrices (2-rank tensors),

$$\mathbf{r} = \hat{\imath}a + \hat{\jmath}b, \qquad \bar{\bar{\mathbf{A}}} = \hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D, \qquad \bar{\bar{\mathbf{B}}} = \hat{\imath}\hat{\imath}E + \hat{\imath}\hat{\jmath}F + \hat{\jmath}\hat{\imath}G + \hat{\jmath}\hat{\jmath}H \tag{E.3}$$

where a, b, A, B, C, D, E, F, G and H are scalars. Matrix transpose,

$$\bar{\bar{\mathbf{A}}}^T = \hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}C + \hat{\jmath}\hat{\imath}B + \hat{\jmath}\hat{\jmath}D \tag{E.4}$$

Matrix-vector inner products:

$$\bar{\bar{\mathbf{A}}} \cdot \mathbf{r} = (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D) \cdot (\hat{\imath}a + \hat{\jmath}b) = \hat{\imath}(Aa + Bb) + \hat{\jmath}(Ca + Db) \tag{E.5}$$

$$\bar{\bar{\mathbf{A}}}^T \cdot \mathbf{r} = (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}C + \hat{\jmath}\hat{\imath}B + \hat{\jmath}\hat{\jmath}D) \cdot (\hat{\imath}a + \hat{\jmath}b) = \hat{\imath}(Aa + Cb) + \hat{\jmath}(Ba + Db) \tag{E.6}$$

$$\mathbf{r} \cdot \bar{\bar{\mathbf{A}}} = (\hat{\imath}a + \hat{\jmath}b) \cdot (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D) = \hat{\imath}(Aa + Cb) + \hat{\jmath}(Ba + Db) \tag{E.7}$$

$$\bar{\bar{\mathbf{A}}} \cdot \mathbf{r} = \mathbf{r} \cdot \bar{\bar{\mathbf{A}}}^T, \qquad \bar{\bar{\mathbf{A}}}^T \cdot \mathbf{r} = \mathbf{r} \cdot \bar{\bar{\mathbf{A}}} \tag{E.8}$$

Matrix-vector outer products (producing 3-rank tensors):

$$\begin{aligned} \bar{\bar{\mathbf{A}}}\mathbf{r} &= (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D)\,(\hat{\imath}a + \hat{\jmath}b) \\ &= \hat{\imath}\hat{\imath}\hat{\imath}Aa + \hat{\imath}\hat{\imath}\hat{\jmath}Ab + \hat{\imath}\hat{\jmath}\hat{\imath}Ba + \hat{\imath}\hat{\jmath}\hat{\jmath}Bb + \hat{\jmath}\hat{\imath}\hat{\imath}Ca + \hat{\jmath}\hat{\imath}\hat{\jmath}Cb + \hat{\jmath}\hat{\jmath}\hat{\imath}Da + \hat{\jmath}\hat{\jmath}\hat{\jmath}Db \end{aligned} \tag{E.9}$$

$$\begin{aligned} \bar{\bar{\mathbf{A}}}^T\mathbf{r} &= (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}C + \hat{\jmath}\hat{\imath}B + \hat{\jmath}\hat{\jmath}D)\,(\hat{\imath}a + \hat{\jmath}b) \\ &= \hat{\imath}\hat{\imath}\hat{\imath}Aa + \hat{\imath}\hat{\imath}\hat{\jmath}Ab + \hat{\imath}\hat{\jmath}\hat{\imath}Ca + \hat{\imath}\hat{\jmath}\hat{\jmath}Cb + \hat{\jmath}\hat{\imath}\hat{\imath}Ba + \hat{\jmath}\hat{\imath}\hat{\jmath}Bb + \hat{\jmath}\hat{\jmath}\hat{\imath}Da + \hat{\jmath}\hat{\jmath}\hat{\jmath}Db \end{aligned} \tag{E.10}$$

$$\begin{aligned} \mathbf{r}\bar{\bar{\mathbf{A}}} &= (\hat{\imath}a + \hat{\jmath}b)\,(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D) \\ &= \hat{\imath}\hat{\imath}\hat{\imath}Aa + \hat{\imath}\hat{\imath}\hat{\jmath}Ba + \hat{\imath}\hat{\jmath}\hat{\imath}Ca + \hat{\imath}\hat{\jmath}\hat{\jmath}Da + \hat{\jmath}\hat{\imath}\hat{\imath}Ab + \hat{\jmath}\hat{\imath}\hat{\jmath}Bb + \hat{\jmath}\hat{\jmath}\hat{\imath}Cb + \hat{\jmath}\hat{\jmath}\hat{\jmath}Db \end{aligned} \tag{E.11}$$

$$\left(\mathbf{r}\bar{\bar{\mathbf{A}}}\right)^T = \bar{\bar{\mathbf{A}}}^T\mathbf{r} \tag{E.12}$$

Gradient operator:

$$\nabla = \hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y} \tag{E.13}$$

Gradient and vector operations,

$$\nabla \cdot \mathbf{r} = \left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right) \cdot \left(\hat{\imath}a + \hat{\jmath}b\right) = \frac{\partial a}{\partial x} + \frac{\partial b}{\partial y} \tag{E.14}$$

$$\mathbf{r} \cdot \nabla = \left(\hat{\imath}a + \hat{\jmath}b\right) \cdot \left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right) = a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y} \tag{E.15}$$

$$\nabla\mathbf{r} = \left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right)\left(\hat{\imath}a + \hat{\jmath}b\right) = \hat{\imath}\hat{\imath}\frac{\partial a}{\partial x} + \hat{\imath}\hat{\jmath}\frac{\partial b}{\partial x} + \hat{\jmath}\hat{\imath}\frac{\partial a}{\partial y} + \hat{\jmath}\hat{\jmath}\frac{\partial b}{\partial y} \tag{E.16}$$

$$\mathbf{r}\nabla = \left(\hat{\imath}a + \hat{\jmath}b\right)\left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right) = \hat{\imath}\hat{\imath}a\frac{\partial}{\partial x} + \hat{\imath}\hat{\jmath}a\frac{\partial}{\partial y} + \hat{\jmath}\hat{\imath}b\frac{\partial}{\partial x} + \hat{\jmath}\hat{\jmath}b\frac{\partial}{\partial y} \tag{E.17}$$

Gradient and matrix operations,

$$\nabla \cdot \bar{\bar{\mathbf{A}}} = \left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right) \cdot \left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right) = \hat{\imath}\left(\frac{\partial A}{\partial x} + \frac{\partial C}{\partial y}\right) + \hat{\jmath}\left(\frac{\partial B}{\partial x} + \frac{\partial D}{\partial y}\right) \tag{E.18}$$

$$\bar{\bar{\mathbf{A}}} \cdot \nabla = \left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right) \cdot \left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right) = \hat{\imath}\left(A\frac{\partial}{\partial x} + B\frac{\partial}{\partial y}\right) + \hat{\jmath}\left(C\frac{\partial}{\partial x} + D\frac{\partial}{\partial y}\right) \tag{E.19}$$

$$\begin{aligned}\nabla\bar{\bar{\mathbf{A}}} &= \left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right)\left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right) \\ &= \hat{\imath}\hat{\imath}\hat{\imath}\frac{\partial A}{\partial x} + \hat{\imath}\hat{\imath}\hat{\jmath}\frac{\partial B}{\partial x} + \hat{\imath}\hat{\jmath}\hat{\imath}\frac{\partial C}{\partial x} + \hat{\imath}\hat{\jmath}\hat{\jmath}\frac{\partial D}{\partial x} + \hat{\jmath}\hat{\imath}\hat{\imath}\frac{\partial A}{\partial y} + \hat{\jmath}\hat{\imath}\hat{\jmath}\frac{\partial B}{\partial y} + \hat{\jmath}\hat{\jmath}\hat{\imath}\frac{\partial C}{\partial y} + \hat{\jmath}\hat{\jmath}\hat{\jmath}\frac{\partial D}{\partial y}\end{aligned} \tag{E.20}$$

$$\begin{aligned}\bar{\bar{\mathbf{A}}}\nabla &= \left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right)\left(\hat{\imath}\frac{\partial}{\partial x} + \hat{\jmath}\frac{\partial}{\partial y}\right) \\ &= \hat{\imath}\hat{\imath}\hat{\imath}A\frac{\partial}{\partial x} + \hat{\imath}\hat{\imath}\hat{\jmath}A\frac{\partial}{\partial y} + \hat{\imath}\hat{\jmath}\hat{\imath}B\frac{\partial}{\partial x} + \hat{\imath}\hat{\jmath}\hat{\jmath}B\frac{\partial}{\partial y} + \hat{\jmath}\hat{\imath}\hat{\imath}C\frac{\partial}{\partial x} + \hat{\jmath}\hat{\imath}\hat{\jmath}C\frac{\partial}{\partial y} + \hat{\jmath}\hat{\jmath}\hat{\imath}D\frac{\partial}{\partial x} + \hat{\jmath}\hat{\jmath}\hat{\jmath}D\frac{\partial}{\partial y}\end{aligned} \tag{E.21}$$

Cross Products

$\hat{\boldsymbol{k}}$ is normal to plane,

$$\hat{\imath} \times \hat{\jmath} = \hat{\boldsymbol{k}}, \qquad \hat{\jmath} \times \hat{\boldsymbol{k}} = \hat{\imath}, \qquad \hat{\boldsymbol{k}} \times \hat{\imath} = \hat{\jmath}, \qquad \hat{\jmath} \times \hat{\imath} = -\hat{\boldsymbol{k}}, \qquad \ldots \tag{E.22}$$

$$\hat{\imath} \times \hat{\boldsymbol{k}} = \begin{vmatrix} \hat{\imath} & \hat{\jmath} & \hat{\boldsymbol{k}} \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{vmatrix} = -\hat{\jmath} \tag{E.23}$$

$$\mathbf{s} = \hat{\imath}c + \hat{\jmath}d \tag{E.24}$$

$$\mathbf{r} \times \mathbf{s} = (\hat{\imath}a + \hat{\jmath}b) \times (\hat{\imath}c + \hat{\jmath}d) = \hat{\boldsymbol{k}}(ad - bc) \tag{E.25}$$

$$\mathbf{s} \times \mathbf{r} = (\hat{\imath}c + \hat{\jmath}d) \times (\hat{\imath}a + \hat{\jmath}b) = \hat{\boldsymbol{k}}(cb - da) = -\mathbf{r} \times \mathbf{s} \tag{E.26}$$

$$\bar{\bar{\mathbf{A}}} \times \mathbf{r} = (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D) \times (\hat{\imath}a + \hat{\jmath}b) = \hat{\imath}\hat{\boldsymbol{k}}(Ab - Ba) + \hat{\jmath}\hat{\boldsymbol{k}}(Cb - Da) \tag{E.27}$$

$$\mathbf{r} \times \bar{\bar{\mathbf{A}}} = (\hat{\imath}a + \hat{\jmath}b) \times (\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D) = \hat{\boldsymbol{k}}\hat{\imath}(Ca - Ab) + \hat{\boldsymbol{k}}\hat{\jmath}(Da - Bb) \tag{E.28}$$

$$\left(\mathbf{r} \times \bar{\bar{\mathbf{A}}}\right)^T = -\bar{\bar{\mathbf{A}}}^T \times \mathbf{r} \tag{E.29}$$

$$\begin{aligned}
\bar{\bar{\mathbf{A}}} \times \bar{\bar{\mathbf{B}}} &= \left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right) \times \left(\hat{\imath}\hat{\imath}E + \hat{\imath}\hat{\jmath}F + \hat{\jmath}\hat{\imath}G + \hat{\jmath}\hat{\jmath}H\right) \\
&= \hat{\imath}\hat{\boldsymbol{k}}A\left(\hat{\imath}G + \hat{\jmath}H\right) - \hat{\imath}\hat{\boldsymbol{k}}B\left(\hat{\imath}E + \hat{\jmath}F\right) + \hat{\jmath}\hat{\boldsymbol{k}}C\left(\hat{\imath}G + \hat{\jmath}H\right) - \hat{\jmath}\hat{\boldsymbol{k}}D\left(\hat{\imath}E + \hat{\jmath}F\right) \\
&= \hat{\imath}\hat{\boldsymbol{k}}\hat{\imath}(AG - BE) + \hat{\imath}\hat{\boldsymbol{k}}\hat{\jmath}(AH - BF) + \hat{\jmath}\hat{\boldsymbol{k}}\hat{\imath}(CG - DE) + \hat{\jmath}\hat{\boldsymbol{k}}\hat{\jmath}(CH - DF) \qquad \text{(E.30)} \\
\bar{\bar{\mathbf{B}}} \times \bar{\bar{\mathbf{A}}} &= \left(\hat{\imath}\hat{\imath}E + \hat{\imath}\hat{\jmath}F + \hat{\jmath}\hat{\imath}G + \hat{\jmath}\hat{\jmath}H\right) \times \left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right) \\
&= \hat{\imath}\hat{\boldsymbol{k}}E\left(\hat{\imath}C + \hat{\jmath}D\right) - \hat{\imath}\hat{\boldsymbol{k}}F\left(\hat{\imath}A + \hat{\jmath}B\right) + \hat{\jmath}\hat{\boldsymbol{k}}G\left(\hat{\imath}C + \hat{\jmath}D\right) - \hat{\jmath}\hat{\boldsymbol{k}}H\left(\hat{\imath}A + \hat{\jmath}B\right) \\
&= \hat{\imath}\hat{\boldsymbol{k}}\hat{\imath}(EC - FA) + \hat{\imath}\hat{\boldsymbol{k}}\hat{\jmath}(ED - FB) + \hat{\jmath}\hat{\boldsymbol{k}}\hat{\imath}(GC - HA) + \hat{\jmath}\hat{\boldsymbol{k}}\hat{\jmath}(GD - HB) \qquad \text{(E.31)} \\
\left(\bar{\bar{\mathbf{B}}} \times \bar{\bar{\mathbf{A}}}\right)^T &= -\bar{\bar{\mathbf{A}}}^T \times \bar{\bar{\mathbf{B}}}^T \qquad \text{(E.32)}
\end{aligned}$$

For material line boundary-integral term, transfer $\mathbf{r}$ to final position,

$$\begin{aligned}
\left(\bar{\bar{\mathbf{A}}} \cdot \mathbf{r}\right) \times \mathbf{s} &= -\mathbf{s} \times \left(\bar{\bar{\mathbf{A}}} \cdot \mathbf{r}\right) \\
&= -\left(\hat{\imath}c + \hat{\jmath}d\right) \times \left[\left(\hat{\imath}\hat{\imath}A + \hat{\imath}\hat{\jmath}B + \hat{\jmath}\hat{\imath}C + \hat{\jmath}\hat{\jmath}D\right) \cdot \left(\hat{\imath}a + \hat{\jmath}b\right)\right] \\
&= -\left(\hat{\imath}c + \hat{\jmath}d\right) \times \Big(\hat{\imath}(Aa + Bb) + \hat{\jmath}(Ca + Db)\Big) \\
&= -\hat{\boldsymbol{k}}\Big(c(Ca + Db) - d(Aa + Bb)\Big) = \hat{\boldsymbol{k}}\Big(a(Ad - Cc) + b(Bd - Dc)\Big) \\
&= \left(\hat{\imath}a + \hat{\jmath}b\right) \cdot \Big(\hat{\imath}\hat{\boldsymbol{k}}(Ad - Cc) + \hat{\jmath}\hat{\boldsymbol{k}}(Bd - Dc)\Big) \\
&= \mathbf{r} \cdot \left(\bar{\bar{\mathbf{A}}}^T \times \mathbf{s}\right) \\
&= \Big(\hat{\boldsymbol{k}}\hat{\imath}(Ad - Cc) + \hat{\boldsymbol{k}}\hat{\jmath}(Bd - Dc)\Big) \cdot \left(\hat{\imath}a + \hat{\jmath}b\right) \\
&= -\left(\mathbf{s} \times \bar{\bar{\mathbf{A}}}\right) \cdot \mathbf{r} \qquad \text{(E.33)}
\end{aligned}$$

Bibliography

[1] N. Taranu A. Axinte, L. Bejan and P. Ciobanu. Modern approaches on the optimization of composite structures. Technical report, Polytechnic Institute of Iasi, Construction and Architecture Section, Nov 2013.

[2] Steven Allmaras. *2D Hybrid Shell Model: Duality Analysis and BCs Weights*. Dept. of Aeronautics and Astronautics, Massachusetts Institute of Technology, July 2016. Offline communication.

[3] M.W. Bloomfield and J.E. Herencia. Optimization of anisotropic laminated composite plates incorporating non-conventional ply orientations. In *49th AIAA/ASMA/ASCE/ASC Structures, Structural Dynamics, and Materials Conver-ence*, Schaumburg, IL, 2008.

[4] F. Cirak, M. Ortiz, and P. Schröder. Subdivision surfaces: A new paradigm for thin-shell finite-element analysis. *Int. J. for Numerical Methods in Engineering*, 47(12):2039–2072, Apr 2000.

[5] S. Deng. Quadrature formulas in two dimensions. Class notes Math 5172, Department of Mathematics and Statistics, University of North Carolina Charlotte, 2010.

[6] C.N. DeSilva and P.J. Tsai. A general theory of directed surfaces. *Acta Mechanica*, 18:89–101, 1973.

[7] M. Drela. Three-dimensional integral boundary layer formulation for general configurations. AIAA-2013-2437, June 2013.

[8] Mark Drela. Integrated simulation model for preliminary aerodynamic, structural, and control-law design of aircraft. AIAA 40'th SDM Conference, Paper 1999-1394, April 1999. Also at `http://web.mit.edu/drela/Public/web/aswing`.

[9] Mark Drela. *Hybrid Shell Model*. Dept. of Aeronautics and Astronautics, Massachusetts Institute of Technology, 2014–2016. Offline communication.

[10] Inc. eFunda. efunda: Plate calculator: – simply supported rectangular plate with uniformly distributed loading, Accessed 11/14/14. Available at `http://www.efunda.com/formulae/solid_mechanics/plates/calculators/SSSS_PUniform.cfm`.

[11] Steven G. Johnson. The NLopt nonlinear-optimization package. Available at `http://ab-initio.mit.edu/nlopt`.

[12] G.R. Kirchoff. Über das gleichgewicht und die bewegung einer elastischen scheibe. *J. Füer die Reine und Angewandte Mathematik*, 49:51–88, 1850.

[13] A. Labuschagne, N.F.J. van Rensburg, and A.J. van der Merwe. Comparison of linear beam theories. *Mathematical and Computer Modelling*, 49:20–30, 2009.

[14] Y. Liu and R. Li. Accurate bending analysis of rectangular plates with two adjacent edges free and the others clamped or simply supported based on new symplectic approach. *Applied Mathematical Modeling*, 34:856–865, 2010.

[15] C.G. Diaconu M. W. Bloomfield and P.M. Weaver. On feasible regions of lamination parameters for lay-up optimization of laminated composites. *Proc. R. Soc. A*, 465:1123–1143, 2009.

[16] J.N. Mindlin. Influence of rotatory inertia and shear on flexural motions of isotropic, elastic plates. *Journal of Applied Mechanics*, 18:31–38, 1951.

[17] J. Morlier and D. Bettebghor. Notes on laminates optimization. Workshop 2.3 MAAXIMUS, Jan 2010.

[18] V.V. Nishawala. A study of large deflection of beams and plates. Master's thesis, Rutgers University, Deparment of Mechanical and Aerospace Engineering, January 2011.

[19] Raul Radovitzky. *Simple Beam Theory*. Massachusetts Institute of Technology, Department of Aeronautics and Astronautics.

[20] E. Reissner and M. Stein. Torsion and transverse bending of cantilever plates. Technical Note 2369, NACA, Feb 1951.

[21] K. Salari and P. Knupp. Code verification by the method of manufactured solutions. Technical Report Sandia Report SAND2000-1444, Sandia National Laboratories, June 2000.

[22] Jr. Sanders, J.L. Nonlinear theories for thin shells. Technical Report 10, Division of Engineering and Applied Physics, Harvard University, Feb 1961.

[23] Carlo Sansour and Herbert Bednarczyk. The cosserat surface as a shell model, theory and finite-element forumlation. *Computational Methods in Applied Mechanics and Engineering*, 120:1–32, 1995.

[24] J.C. Simo and D.D. Fox. On a stress resultant geometrically exact shell model. part i: Formulation and optimal parametrization. *Computer Meth. in Appl. Mechanics and Eng.*, 72:267–304, 1989.

[25] J.C. Simo, D.D. Fox, and M.S. Rifai. On a stress resultant geometrically exact shell model. part ii: The linear theory; computational aspects. *Computer Meth. in Appl. Mechanics and Eng.*, 73:53–92, 1989.

[26] J.C. Simo, D.D. Fox, and M.S. Rifai. On a stress resultant geometrically exact shell model. part iii: Computational aspects of the nonlinear theory. *Computer Meth. in Appl. Mechanics and Eng.*, 79:21–70, 1990.

[27] J.C. Simo, D.D. Fox, and M.S. Rifai. On a stress resultant geometrically exact shell model. part iv: Variable thickness shells with through-the-thickness stretching. *Computer Meth. in Appl. Mechanics and Eng.*, 81:91–126, 1990.

[28] Krister Svanberg. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal of Optimization*, 12(2):555–573, 2002.

[29] Dassault Systemes. Solidworks simulation 2014. Theoretical manual, 2014.

[30] Brandon Louis Talamini. *Simulation of Deformation and Fracture in Very Large Shell Structures.* PhD dissertation, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, September 2015.

[31] E. Ventsel and T. Krauthammer. *Thin Plates and Shells.* Marcel Dekker, Inc., 2001.

[32] T. von Karman. Fesigkeitsprobleme in maschinenbau. *Encycl de Math Wiss*, 4:348–351, 1910.

[33] Wikipedia. Hairy ball theorem figure. Available at `https://en.wikipedia.org/wiki/Hairy_ball_theorem#/media/File:Hairy_ball.png`.

[34] R. T. Haftka Z. Gurdal and P. Hajela. *Design and Optimization of Laminated Composite Materials.* John Wiley & Sons, Inc., 1999.