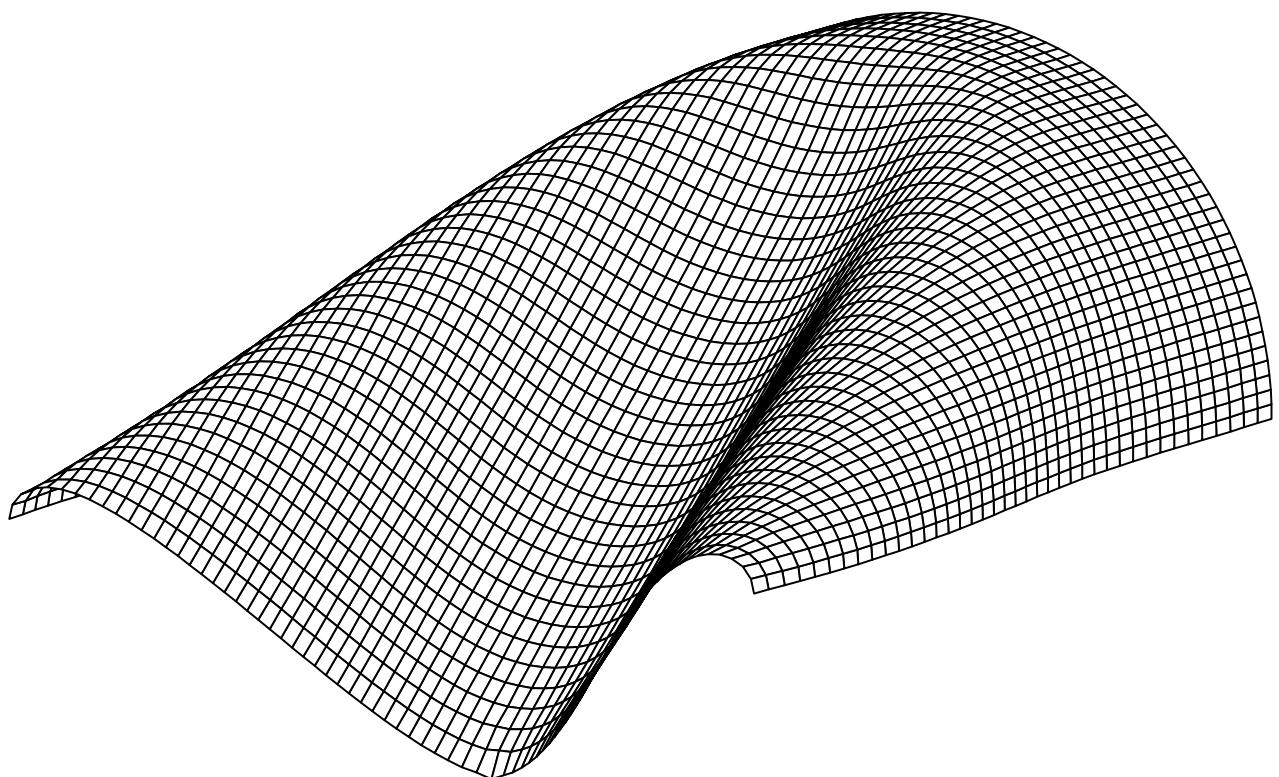


Advanced Shell Finite Elements

Formulation, Implementation
& Enhancing Technologies

Master's Thesis by Peter Wilson



Advanced Shell Finite Elements

Formulation, Implementation & Enhancing Technologies

Submitted to the Department of Civil, Geo and Environmental Engineering
in partial fulfillment of the requirements for the degree of
M.Sc.
at the Technical University of Munich.

Supervised by Dipl.-Ing. (FH) Andreas Winterstein M.Sc.
PD Dr.-Ing. habil. Roland Wüchner
Prof. Dr.-Ing. Kai-Uwe Bletzinger
Chair of Structural Analysis

Submitted by Peter Wilson B.Eng.
Apianstraße 7
85774 Unterföhring

Submitted on Munich, Datum

Abstract - STILL TO DO

In this work, a thin quadrilateral shell element implemented in the multiphysics code KRATOS is presented. The element considered is composed of an Assumed Natural Deviatoric Strains (ANDES) membrane formulation and a Discrete Kirchhoff Quadrilateral (DKQ) bending formulation. Both of these formulations are examined, as well as the general implementation of them in the KRATOS code. Finally, the shell element is tested with the well known shell obstacle course and future work pertaining to the element is identified.

Keywords - STILL TO DO!!!!

FEM; Shell structures; ANDES; DKT; DSG

Table of Contents

1	Introduction - STILL TO DO!!!!	1
2	Shell finite elements - revised 2	2
2.1	Structural modelling with shells.....	2
2.2	Shell models	3
2.3	Locking in shell finite elements	11
2.4	Shell finite element technologies.....	13
2.5	Identification of Kratos shell element formulations	19
3	Composite shells - revised 1	21
3.1	Composite material basics	22
3.2	Orthotropic shell laminates: internal virtual work.....	27
3.3	Laminate strain and stress recovery	31
3.4	Laminate Tsai-Wu failure criterion.....	34
4	Non-linear analysis background - revised 1	36
4.1	Response diagrams	36
4.2	Stability criterion.....	38
4.3	Linear Prebuckling analysis.....	39
4.4	Stability analysis example	40
4.5	Co-rotational transformation fundamentals - STILL TO DO!!!!!!	44
4.6	Co-rotational transformation geometric stiffness - STILL TO DO!!!!!!	44
5	DSG triangle shell element.....	45
5.1	Stiffness matrix formulation.....	45
5.2	ADD IN THEORY OF MEMBRANE DERIVATION HERE !!!!!	46
5.3	ADD IN THEORY OF MEMBRANE DERIVATION HERE !!!!!	46
5.4	Stiffness matrix implementation	48
5.5	Mass matrix formulation and implementation.....	50
5.6	Stress and strain recovery	54
6	ANDES-DKQ quadrilateral shell element.....	57
6.1	Stiffness matrix formulation.....	57
6.2	Stiffness matrix implementation	69
6.3	Mass matrix formulation	72
6.4	Stress and strain recovery	73
7	Extension of shells to composite laminates - revised 1	76
7.1	Composite constitutive matrix formulation	76
7.2	Composite constitutive matrix implementation	77
7.3	Composite stress recovery implementation	79
7.4	Tsai-Wu failure criterion implementation	80

8	Validation of elements.....	82
8.1	Static tests: shell obstacle course	82
8.2	Geometrically non-linear tests	86
8.3	Dynamic tests	87
8.4	Quantity recovery tests.....	89
8.5	Composite tests	94
9	Applications and consequences of element technologies	105
9.1	Euler buckling of CHS column	105
9.2	Shear wrinkling of plate	112
10	Extension of DSG triangle element technology - STILL TO DO!!!!.....	120
10.1	Cell Smoothed DSG approach	120
10.2	DSGc3 approach	126
10.3	Appraisal of alternative DSG technology approaches	132
11	Conclusions and Outlook - STILL TO DO!!!!.....	136
11.1	Future work	136
Appendix A	Analytical stability analysis of Mises truss	137
Appendix B	DSG technology derivation.....	141
Appendix C	Basic-DKQ formulation	146
Appendix D	Basic-T3 formulation	148
Appendix E	Analytical membrane analysis of dome	149
Appendix F	Derivation of Euler buckling load.....	151
Appendix G	DSG element class in Kratos.....	153
G.1	ShellThickElement3D3N methods	153
G.2	ShellThickElement3D3N member variables	156
Appendix H	ANDES-DKQ class in Kratos	159
H.1	ShellThinElement3D4N methods	159
H.2	ShellThinElement3D4N member variables	160

Chapter 1 **Introduction - STILL TO DO!!!!**

KRATOS is a relatively recent multiphysics code emerging from the International Center for Numerical Methods in Engineering (CIMNE) in Barcelona. Primarily aimed at developers, researchers and students, KRATOS's extensibility accommodates the introduction of new functionalities with relative ease.

The additional functionality considered in this paper is the implementation of a thin quadrilateral shell element. This new contribution fills an existing gap in the structural mechanics capabilities of KRATOS, which is currently missing thin quadrilateral and thick triangular shell elements.

Shell elements themselves result from the combination of membrane and bending behaviours into a single element. The thin quadrilateral shell element presented in this paper consists of an Assumed Natural Deviatoric Strains (ANDES) membrane formulation and a complementing Discrete Kirchhoff Quadrilateral (DKQ) bending formulation. Both of these component formulations are discussed in section 1 of the paper. Following this, a high level overview of the element's implementation in KRATOS is presented in section 2, while section 3 covers element benchmarking with the well known shell obstacle course. Future work is discussed in section 4 of this paper.

Chapter 2 Shell finite elements - revised

2

THE employment of shell structures is ubiquitous throughout both nature and the built environment. Eggs, nuts, blood vessels and cell walls are examples of shell designs being the result of structural optimisation via natural evolution over millennia. It is no doubt that man drew inspiration from the optimal natural shell design and realised the efficacy of the shell structure, with the harnessing of these principles culminating in the pre-eminent Roman Pantheon (126). Throughout time, as the understanding of shell structures increased, so did their prevalence, leading to notable structures such as the Hagia Sophia (537), Notre Dame (1345) and St. Peter's Basilica (1626). Indeed, the efficiency of shell structures lies in their high in-plane (membrane) load carrying capacity in slender low-weight constructions. The membrane action serves to stress all fibres approximately equally in the cross section, realising the full mechanical performance of the structure. Contrasting this, shells are incredibly sensitive to a variety of effects such as imperfections, bending, transverse normal forces and support conditions, leading to significant compromise of the membrane structural performance and possibly manifesting in catastrophic failure. To examine one such sensitivity, bending actions result in a non-uniform stressing of material fibres over the cross section, with the outer fibres stressed significantly more than those closer to the neutral axis. Consequently, the limit of the structure in bending is realised when only the outer-most fibre fails instead of the entire cross section of fibres failing under membrane action. This basic example offers a snapshot of the stellar performance of shells juxtaposed against their sensitivity to a multitude of conditions, earning them the title of the *Prima Donna of structures* [41].

2.1. Structural modelling with shells

Although shells present the opportunity of an optimally loaded structure, their delicate position in a sharply varying landscape of performance demands careful consideration of phenomena critical to the analysis undertaken. When this potentially volatile behaviour is held against the scientific ethos of *everything should be made as simple as possible, but no simpler*, the arising tension is immediately recognized, one that can only be curtailed by an in depth knowledge

of the working problem and shells themselves. Within the engineering design context of a particular scenario, there exists as many opportunities to reasonably reduce complexity as there are to incorrectly exclude critical phenomena. Typical structural modelling decisions such as: inclusion or exclusion of inertial and damping effects, non-linear or linear material models, large or small deformation assumptions and dimensional reduction are examples of broad brush strokes limiting the canvas of possibilities resolved. Focussing on the rendering of shells, the assumed underlying structural models effectively scope and frame the domain of potential mechanical expressions for a system.

Inherent in the use of shells in structural models is the concept of dimensional reduction from 3 dimensions to 2 dimensions, relying on the assumption that one dimension (thickness) is significantly smaller than the other two (length and width).

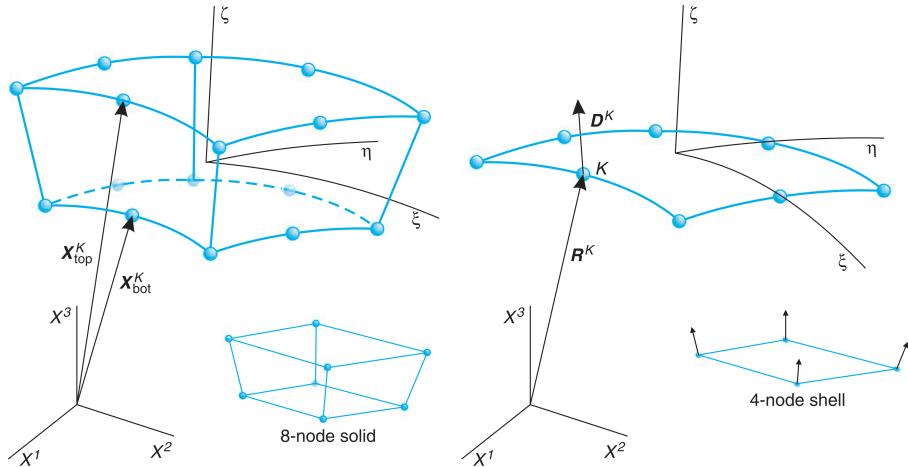


Figure 1 Dimensional reduction of a solid to a shell [8]

Already, it is apparent that the through-thickness response of the shell now must be modelled instead of resolved, with the results now a function of the approximation employed. This apparent simplification promptly begs a key question: what shall the model consider such that it is simple as possible, but not simpler? Can the thickness vary under deformation? Is the shell one uniform material or multi-layered? Is shear deformation of the thickness negligible or not? One may also impose far stricter modelling assumptions by only considering the bending or membrane behaviour of the shell. These common structural modelling decisions, amongst others, have yielded typical shell models.

2.2. Shell models

Commencing in the Renaissance and continuing into the present day, the mathematical development of shell models has facilitated the construction of increasingly elaborate shell structures. The main mathematically-based shell models considered in this work are illustrated

below.

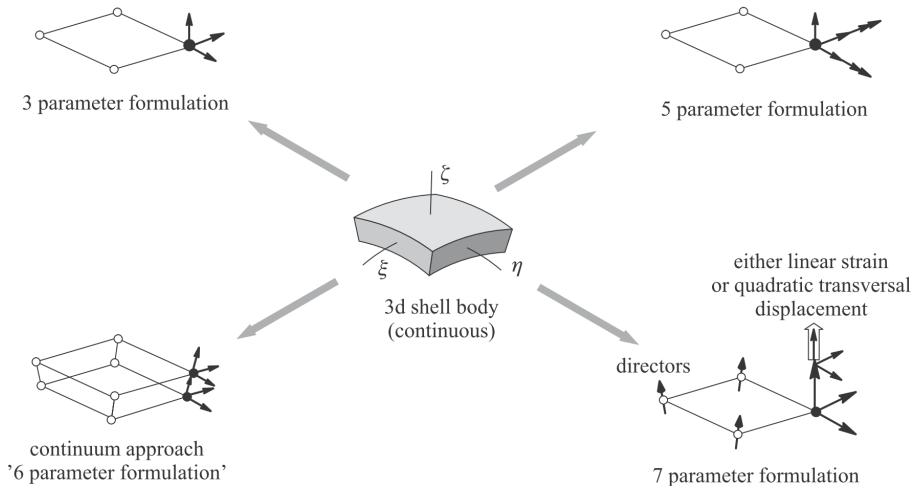


Figure 2 Various shell models [47]

Each of the shell models above are based on different assumptions and physics, which essentially act as a filter of what phenomena the structural model can resolve. These models, as well as the basic membrane model, will be briefly discussed in the following section (further details can be found in References [8] [42] and [16]). To gain further insight, high level formulations of the models are presented with a focus of the mathematical representation of key assumptions. The follow figure illustrates the configurations and notation of the formulations.

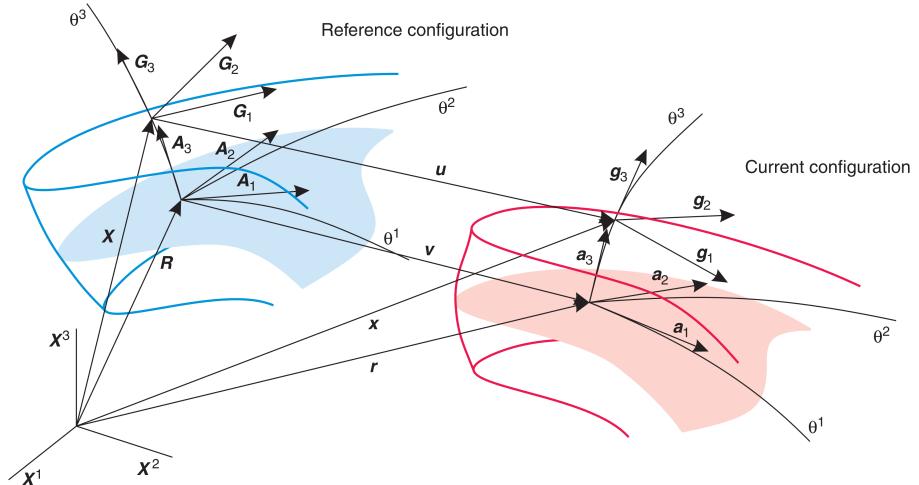


Figure 3 Deformation, reference and current configuration [8]

Quantities in the reference configuration are expressed in upper case while quantities in lower case are in the deformed configuration. The reference shell mid-plane $\theta^3 = 0$ position vector of a point is denoted \mathbf{R} , while an arbitrary point is denoted \mathbf{X} . Correspondingly, base vectors on the mid-plane are denoted \mathbf{A}_i and \mathbf{a}_i while \mathbf{G}_i and \mathbf{g}_i denote arbitrary base vectors. It is noted that Einstein notation is employed here, with Latin characters corresponding to summation

over three dimensions while Greek characters sum over two dimensions. Lastly, \mathbf{v} and \mathbf{u} indicate displacements on the mid-plane and an arbitrary location respectively.

2.2.1. Membrane model

Despite not truly being a shell model, the membrane model is the simplest model available as it completely ignores bending behaviour. Thus, the structural behaviour of the whole element is described by in plane components. Typically it is assumed that all stress and strain components are constant over the thickness. A key model choice is the specification of either plane stress or plane strain behaviour which is implemented in material matrix.

Commencing a high level formulation of the membrane model, the assumption of constant strain and stress components over the thickness allows collapsing the body into an infinitely thin shell. Thus thickness can be ignored in the position vectors.

$$\mathbf{X} = \mathbf{R}, \quad \mathbf{x} = \mathbf{r}, \quad \mathbf{r} = \mathbf{R} + \mathbf{v} \quad (2.1)$$

Using the notation of $(\cdot)_{,\alpha} = \frac{\partial(\cdot)}{\partial \alpha}$ and explicitly writing the base vectors of the coordinate system yields:

$$\mathbf{A}_\alpha = \mathbf{R}_{,\alpha} = \mathbf{X}_{,\alpha}, \quad \mathbf{a}_\alpha = \mathbf{r}_{,\alpha} = \mathbf{A}_\alpha + \mathbf{v}_{,\alpha} \quad (2.2)$$

Considering the metrics of the reference and deformed configuration, the in-plane Green-Lagrange strain components read:

$$\epsilon_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - A_{\alpha\beta}) \quad \text{with} \quad a_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta \quad A_{\alpha\beta} = \mathbf{A}_\alpha \cdot \mathbf{A}_\beta \quad (2.3)$$

Corresponding to the membrane assumptions, all out of plane strain components are 0.

$$\epsilon_{3i} = 0 \quad (2.4)$$

At this point, one notices that all strain terms are completely contained within the two in-plane mid-surface displacements \mathbf{v}_α .

By introducing the elasticity tensor \mathbf{C}_0 (typically plane stress) the stress components can be recovered from the strains.

$$\sigma^{\alpha\beta} = C_0^{\alpha\beta\gamma\delta} \epsilon_{\gamma\delta} \quad (2.5)$$

With stresses and strains determined, the internal and a generalised (where \mathbf{f} is a generalised traction vector and $\delta\mathbf{v}$ are virtual displacements) external virtual work can be expressed:

$$-\delta\Pi_{int} = \int_{\Omega} \boldsymbol{\epsilon} : \mathbf{C}_{mem} : \delta\boldsymbol{\epsilon} \, d\Omega, \quad \delta\Pi_{ext} = \int_{\Omega} \mathbf{f}^T \delta\mathbf{v} \, d\Omega \quad (2.6)$$

It's apparent that the internal work is composed solely of in-plane action, corresponding to the general descriptive assumptions of the membrane model above. By extension, it can be understood that the membrane model provides no resistance to out of plane action. Thus, unless the membrane-modelled structure is pre-stressed, the system will be rendered singular under out of plane loads. This lack of out of plane stiffness can also lead to buckling under compressive stresses. Considering the reduced phenomena that the membrane model can resolve, it is crucial to understand the critical physics of the system before employing it.

2.2.2. 3 parameter model: Kirchhoff-Love shell

The first actual shell model considered is the 3 parameter model, often referred to as the Kirchhoff-Love (KL) shell. This model includes all membrane considerations, but also describes bending behaviour too. The bending behaviour is constrained to a description similar to the Bernoulli beam: shell directors across the thickness remain straight and normal to the mid-surface. Graphically, this is represented in the following figure:

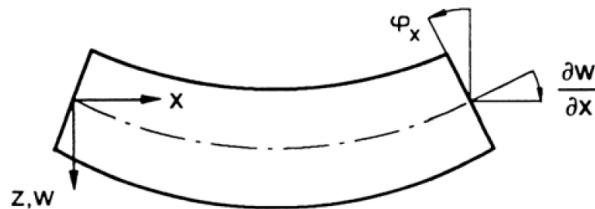


Figure 4 Kirchhoff-Love shell kinematics [11]

A consequence of the above kinematics is that this model ignores transverse shear strains. Thus, the applicability of the 3 parameter model is clearly limited to thin plates in the range of $\frac{1}{5} < \frac{l}{t} < \frac{1}{50}$ where transverse deformations are negligible. Similar to the membrane model, thickness deformation is ignored.

Establishing the geometry of the KL shell requires incorporation of the shell director along θ^3 in the reference \mathbf{D} and deformed configuration \mathbf{d} .

$$\mathbf{X} = \mathbf{R} + \theta^3 \mathbf{D}, \quad \mathbf{x} = \mathbf{r} + \theta^3 \mathbf{d}, \quad \mathbf{r} = \mathbf{R} + \mathbf{v}, \quad \mathbf{d} = \Lambda \mathbf{D} \quad (2.7)$$

The above equation enforces the KL condition of a straight director with the linear description of $\theta^3 \mathbf{d}$. Λ is a rotation tensor composed of two independent rotation parameters β^α relating the reference and deformed directors to each other. In a Cartesian frame the linearised rotation components are: $\beta^1 = \mathbf{v}_{3,2}$ and $\beta^2 = -\mathbf{v}_{3,1}$ [8].

The displacement is thus expressed:

$$\mathbf{u} = \mathbf{x} - \mathbf{X} = \mathbf{v} + \theta^3 (\Lambda - \mathbf{G}) \mathbf{D} = \mathbf{v} + \theta^3 \mathbf{d} \quad (2.8)$$

!!!!!!!!!!!!!! Gotta figure out lamda - $\mathbf{G} = \text{lamda}$!!!!!!!

The KT requirement of the director being normal to the mid surface is expressed via the following dot product:

$$\mathbf{d} \cdot \mathbf{r}_{,\alpha} = (\Lambda \mathbf{D}) \cdot (\mathbf{A}_\alpha + \mathbf{v}_{,\alpha}) = 0 \quad (2.9)$$

Explicitly writing the base vectors of the coordinate system:

$$\mathbf{A}_\alpha = \mathbf{R}_{,\alpha} \quad \mathbf{a}_\alpha = \mathbf{r}_{,\alpha} = \mathbf{A}_\alpha + \mathbf{v}_{,\alpha} \quad (2.10)$$

Eqn (2.9), requiring the director to be normal to the mid-surface, is guaranteed by employing cross products of the base vectors to construct the directors:

$$\mathbf{D} = \frac{\mathbf{A}_1 \times \mathbf{A}_2}{\|\mathbf{A}_1 \times \mathbf{A}_2\|} = \mathbf{A}_3, \quad \mathbf{d} = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\|\mathbf{a}_1 \times \mathbf{a}_2\|} = \mathbf{a}_3, \quad (2.11)$$

As the KT model considers bending, which is related to curvature, the second fundamental form of the system is defined in the reference and deformed configuration:

$$B_{\alpha\beta} = \frac{1}{2} (\mathbf{A}_\alpha \cdot \mathbf{A}_{3,\beta} + \mathbf{A}_\beta \cdot \mathbf{A}_{3,\alpha}) = \mathbf{A}_\alpha \cdot \mathbf{A}_{3,\beta} = \mathbf{A}_\alpha \cdot \mathbf{D}_{,\beta} \quad (2.12)$$

$$b_{\alpha\beta} = \frac{1}{2}(\mathbf{a}_\alpha \cdot \mathbf{a}_{3,\beta} + \mathbf{a}_\beta \cdot \mathbf{a}_{3,\alpha}) = \mathbf{a}_\alpha \cdot \mathbf{a}_{3,\beta} = (\mathbf{A}_\alpha + \mathbf{v}_{,\alpha}) \cdot \left(\frac{(\mathbf{A}_1 + \mathbf{v}_{,1}) \times (\mathbf{A}_2 + \mathbf{v}_{,2})}{\|(\mathbf{A}_1 + \mathbf{v}_{,1}) \times (\mathbf{A}_2 + \mathbf{v}_{,2})\|} \right)_{,\beta} \quad (2.13)$$

Contrasting with the membrane model, the KT strain tensor components now include linearly varying terms corresponding to bending phenomena:

$$E_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - A_{\alpha\beta}) + \theta^3(b_{\alpha\beta} - B_{\alpha\beta}) = \epsilon_{\alpha\beta} + \theta^3\kappa_{\alpha\beta} \quad (2.14)$$

According to the KT assumptions all out of plane strains are zero.

$$E_{3i} = \epsilon_{3i} = \kappa_{3i} = 0 \quad (2.15)$$

Studying the strain components, especially the deformed second fundamental form, reveals that there are now 3 midplane displacements \mathbf{v}_i involved in the description of the KT shell model, hence the name 3 parameter model.

Combining the above developments, and assuming the same general external work as equation (2.6), the internal virtual work can be presented:

$$-\delta\Pi_{int} = \int_{\Omega} \boldsymbol{\epsilon} : \mathbf{C}_{mem} : \delta\boldsymbol{\epsilon} d\Omega + \int_{\Omega} \boldsymbol{\kappa} : \mathbf{C}_{bend} : \delta\boldsymbol{\kappa} d\Omega \quad (2.16)$$

The internal work equation illustrates the 3 parameter model considers in-plane membrane behaviour as well as the additional bending behaviour related to the second integral. Furthermore, under the condition of homogeneous linear material models the membrane and bending behaviour of the model are uncoupled. Due to the kinematics of the 3 parameter model (directors remain straight and normal, no transverse shear strains), it can correctly resolve analyses as shell thicknesses approach zero. This is in contrast to the 5 parameter model, which exhibits significant shear locking. Despite this, a pure rendition of the 3 parameter model is not commonly seen in practical FEM due to the required C_1 continuity at element boundaries (arising from rotations expressed as derivatives of transverse displacement) and the additional complication of effective shear forces on boundaries [8].

2.2.3. 5 parameter model: Reissner-Mindlin shell

By relaxing the assumptions made in the 3 parameter shell model, the Reissner-Mindlin (RM) 5 parameter shell model can be derived. This model includes both membrane and bending

action. While the KL model required that the shell directors remain normal to the mid-surface, the RM model relaxes this, analogous to the relationship between Bernoulli and Timoshenko beam models. Graphically, this is represented in the following figure:

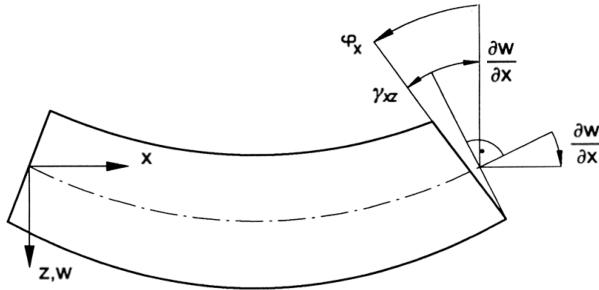


Figure 5 Reissner-Mindlin shell kinematics [11]

Studying the above kinematics confirms this model now considers transverse shear strains, limiting the range of validity of this model to thick plates $\frac{1}{5} < \frac{l}{t} < \frac{1}{10}$ where transverse deformations are a key component of structural behaviour. Similar to the membrane and KL model, thickness deformation is ignored.

The geometry of the RM model is established similar to the KL model:

$$\mathbf{u} = \mathbf{x} - \mathbf{X} = \mathbf{v} + \theta^3(\boldsymbol{\Lambda} - \mathbf{G})\mathbf{D} = \mathbf{v} + \theta^3\mathbf{d} \quad (2.17)$$

However, the strict requirement of maintaining the director remain normal to the mid-surface, as expressed in the KL theory equation (2.9), is no longer enforced. Correspondingly, the rotation tensor $\boldsymbol{\Lambda}$ must now include 2 additional parameters related to these 2 introduced degrees of freedom.

The general strain components are expressed as:

$$E_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - A_{\alpha\beta}) + \theta^3(b_{\alpha\beta} - B_{\alpha\beta}) = \epsilon_{\alpha\beta} + \theta^3\kappa_{\alpha\beta} \quad (2.18)$$

Once again it is noted the assumption of straight directors is enforced by the linear coupling of $\theta^3\kappa_{\alpha\beta}$. Following the assumption of no thickness strain, it is seen:

$$E_{33} = \epsilon_{33} = \kappa_{33} = 0 \quad (2.19)$$

By relaxing the director normality requirements, additional transverse shear strains must be accounted for:

$$E_{\alpha 3} = E_{3\alpha} = \frac{1}{2}(a_{\alpha 3} - A_{\alpha 3}) = \frac{1}{2}\gamma_{3\alpha} \quad (2.20)$$

The internal virtual work is therefore expressed as:

$$-\delta\Pi_{int} = \int_{\Omega} \boldsymbol{\epsilon} : \mathbf{C}_{mem} : \delta\boldsymbol{\epsilon} \, d\Omega + \int_{\Omega} \boldsymbol{\kappa} : \mathbf{C}_{bend} : \delta\boldsymbol{\kappa} \, d\Omega + \int_{\Omega} \boldsymbol{\gamma} : \mathbf{C}_{shear} : \delta\boldsymbol{\gamma} \, d\Omega \quad (2.21)$$

The 3 integrals of the virtual work equation represent the membrane, bending and shear work components, corresponding to the phenomena this model resolves. Furthermore, all these components are decoupled from each other in flat shells with homogeneous linear material models. The consideration of transverse shear deformations in the kinematics render the model applicable to thick shells where these strains are not insignificant. Incorrectly applying this model to thin shells in FEM yields spurious results due to a phenomena called shear locking (discussed in section 2.3.1). Despite this disadvantage, the 5 parameter forms the basis of many shell elements often used in FEM thanks to the lower C_0 continuity required at element boundaries.

2.2.4. 7 parameter model

The previously discussed models all operate under the assumption that the transverse normal strains are zero. The 7 parameter model considers thickness deformation by introducing additional free parameters. Only a brief overview of the 7 parameter model is offered here as shell elements in FEM, the focus of this work, are predominately based off 3 and 5 parameter based formulations. For further details refer Bischoff et al. [8] and Ramm and Wall [42].

Intuitively, one may realise that shell behaviour including thickness change may be described by 6 parameters: 3 mid-surface displacements, 1 thickness change and 2 rotations. However, thickness locking occurs under this regime due to a mismatch of a linearly varying normal thickness stress θ^{33} conjugated with a constant thickness strain ϵ_{33} . Thus the 7th parameter is the enhancement of the through thickness strain ϵ_{33} to a linear field.

It's clear that the additional modelling power of the 7 parameter shell can resolve physics that lower parameter models can't. A prime example of this is the Eigenvalue spectra presented below:

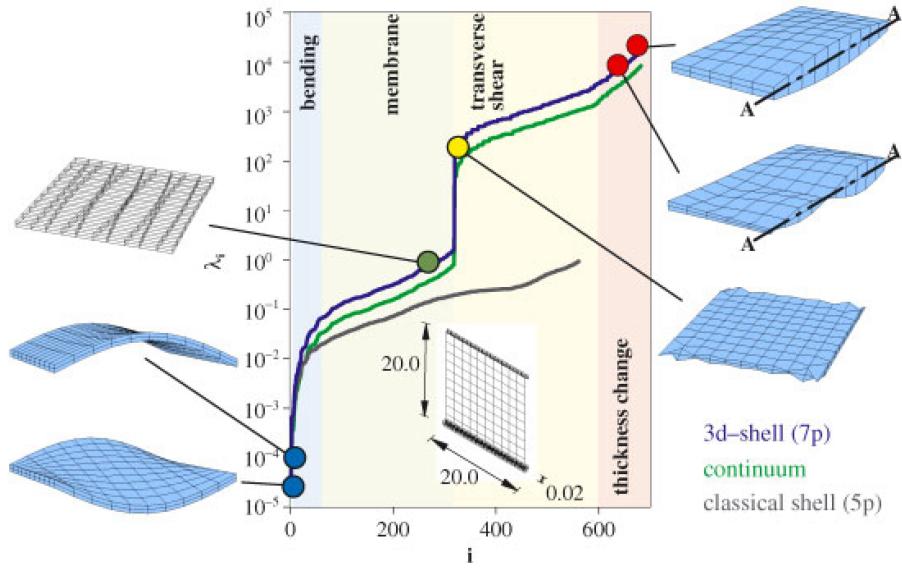


Figure 6 Eigenvalue spectra of various shell models [42]

As expected, the 7 parameter model captures higher Eigen-frequencies associated with thickness modes, while the 5 parameter is unable to resolve these. This is yet another example of model selection limiting the possibility of phenomena resolved.

2.3. Locking in shell finite elements

Surveying a range of shell models has confirmed that not all of them are appropriate for every type of analysis. One must consider the capabilities of the model in conjunction with the supposed critical phenomena of the analysis at hand. Thus, the analysis results are a function of physics the shell model can express. This concept of expression limitation is vital to the correct understanding of shells in the FEM. If the isogeometric approach to the FEM is employed, the field of quantities in the problem are interpolated between discrete nodal values ($\hat{\cdot}$) using shape functions N . In general:

$$\begin{pmatrix} \mathbf{R} \\ \mathbf{v} \\ \epsilon_{ij} \\ \vdots \end{pmatrix}(\xi, \eta) = \sum_{m=1}^{n \text{ nodes}} N(\xi, \eta)_m \begin{pmatrix} \hat{\mathbf{R}}_m \\ \hat{\mathbf{v}}_m \\ \hat{\epsilon}_{ij m} \\ \vdots \end{pmatrix} \quad (2.22)$$

The resolving power of the shape functions undoubtedly restricts what continuous fields can be described from discrete values. They govern not only the description of geometry, but also the deformation modes the element can express. This forms another layer of expression limitation

added to shell models in FEM. Given the propensity to use linear or quadratic shape functions in modern FEM codes, these limitations are often not insignificant. These, together with the physics assumptions and limitations of each shell model, give rise to common numerical inaccuracies, generally termed locking.

2.3.1. Transverse shear locking

Transverse shear locking is perhaps the most recognized and problematic locking phenomena amongst the three considered in this work. As it is related to transverse shear strains, transverse shear locking is possible in the 5 parameter model and impossible for membrane and 3 parameter models. Phenomenologically, transverse shear locking occurs when thin shells incorrectly described by a 5 parameter model are subject to bending situations, with the signature of significantly reduced displacements (ie. 'locked') than expected. By indicating specific material matrices, and removing membrane work for clarity, the internal bending and shear virtual work of the 5 parameter model can be expressed as follows:

$$\bar{\mathbf{C}} = \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \quad \mathbf{C}_{bend} = \frac{Et^3}{12(1-\nu^2)} \bar{\mathbf{C}} \quad \mathbf{C}_{shear} = \alpha Gt \mathbf{I} \quad (2.23)$$

$$-(\delta\Pi_{int} - \delta\Pi_{int\ mem}) = -(\Pi_{bend} + \Pi_{shear}) = \int_{\Omega} \boldsymbol{\kappa} : \frac{Et^3}{12(1-\nu^2)} \bar{\mathbf{C}} : \delta\boldsymbol{\kappa} d\Omega + \int_{\Omega} \boldsymbol{\gamma} : \alpha Gt \mathbf{I} : \delta\boldsymbol{\gamma} d\Omega \quad (2.24)$$

As phenomenologically described, transverse shear locking comes into effect with thin shells. One can see that as $t \rightarrow 0$ the bending internal work ($\Pi_{bend} \propto t^3$) will be far less than the shear internal work ($\Pi_{shear} \propto t$), leading to an incorrect allocation of internal energy. Since the bending internal work is associated with bending deflections, these resulting deflections will be less than they should be and the element will appear locked. The over-representation of shear strains also leads to strong shear force oscillations - another classic symptom of transverse shear locking.

2.3.2. Membrane locking

Membrane locking is the inability to undergo inextensional bending deformations without parasitic membrane contributions. Physically, a primary symptom of this is significantly reduced deformations under pure bending action. Element curvature is a necessary condition for membrane locking, while increasing slenderness exacerbates the problem. Similar to transverse shear locking, as $t \rightarrow 0$ the bending internal work ($\Pi_{bend} \propto t^3$) reduces at a far greater rate than the membrane internal work ($\Pi_{mem} \propto t$) leading to artificial membrane contributions. Thus, membrane locking is possible in 3, 5, and 7 parameter models. The following figure

illustrates the increasing severity of membrane locking as slenderness increases for 3 and 5 parameter NURBS based shell models.

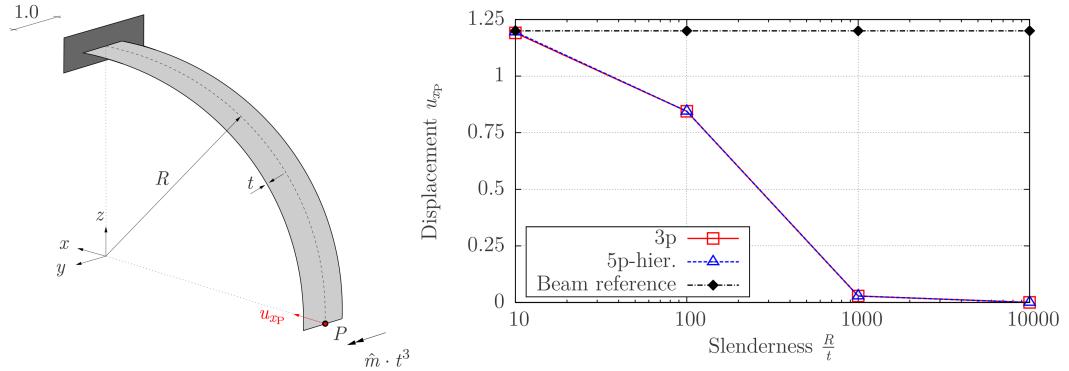


Figure 7 Convergence of cylindrical shell problem demonstrating membrane locking [16]

Despite the bleak results of the above problem, especially in high slenderness range, Bischoff et al. [8] suggest that the adverse effects of membrane locking are mild when using bilinear shape functions, and completely ignored in linear triangle elements (where curvature is always zero). These lower order finite elements form the bulk of what used in commercial FEM codes and are the focus of this work.

2.3.3. Curvature thickness locking

Curvature locking is another locking consideration that only occurs in curved structures with 7 parameter models. The hallmark of curvature thickness locking is artificial through-thickness strains ϵ_{33} under pure bending action. Since the focus of this work is 3 and 5 parameter models that don't include normal strains ϵ_{33} , the reader is referred to Bischoff et al. [8] and Echter [16] for further information.

2.4. Shell finite element technologies

The previous discussion of locking phenomena associated with pure displacement formulations of shell finite elements has given rise to a number of shell finite element technologies to improve element performance. Broadly speaking, these mitigation approaches fall into two categories: reduced integration and B-Bar (\bar{B}) approaches which modify the strain displacement matrix B .

2.4.1. Reduced integration

One of the simplest and oldest methods of curbing locking is reduced integration, which deliberately uses less Gauss points than required to integrate the element stiffness matrix. Typically implemented as selective reduced integration (SRI), where the bending component is fully integrated and only the shear part undergoes reduced integration, the efficacy of the

method relies on how susceptible the reduced integration Gauss point locations are to parasitic strains. Despite this 'chance' aspect, it is often used in crash worthiness simulations with the benefits of reduced locking and reduced computational time. The following graph compares the performance (scaled displacement vs. slenderness) of a fully and reduced integrated 5 parameter shell against the reference solution for a square plate in bending.

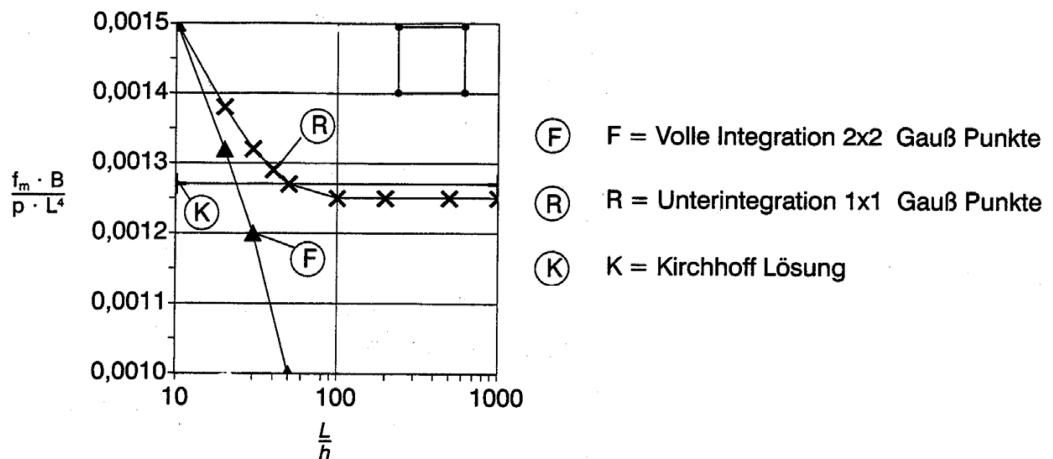


Figure 8 Reduced integration of a 5 parameter Quad 4 shell [11]

F = Full integration, R = Reduced integration, K = Kirchhoff (3 parameter model) solution

It's clear that the normal fully integrated element exhibits severe locking, while the element with reduced integration converges to a value close to the reference solution. Despite this, SRI in general still doesn't guarantee complete removal of shear locking and also introduces spurious zero energy modes. These zero energy modes are often combated by stabilizing matrices ("hourglass" control [49]) which are designed to be activated under the spurious zero-energy regimes and noted as quite complex to formulate [34]. An additional drawback of reduced integration is element performance deterioration as the mesh becomes distorted and warped [38] [48].

2.4.2. Assumed Natural Strains

The Assumed Natural Strain (ANS) approach forms a main umbrella of B-Bar methods, which alters the strain-displacement matrix \mathbf{B} to mitigate locking. The ANS approach [32] works by computing the strain values at particular co-location points less susceptible to parasitic strains in the element (normally chosen as mid-edge and/or centre points) and then interpolating these discrete values through the element to define a new "assumed" shear strain field. As a general approach, many subsequent technologies fall under the ANS umbrella.

2.4.3. Mixed Interpolation of Tensorial Components

Falling within the ANS framework, Dvorkin and Bathe [15] [3] developed the Mixed Interpolation of Tensorial Components (MITC) approach which relies on an assumed shear strain field. A graphical example of this formulation is demonstrated below on a Quad 4 element, with linear interpolation of the shear strain field at mid-side points.

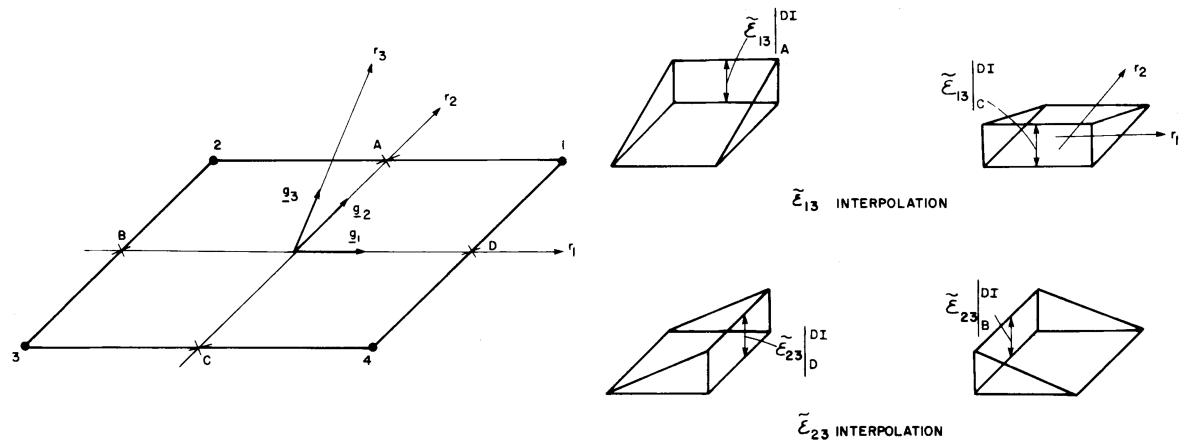


Figure 9 Assumed shear strain field of the MITC4 element [3]

The performance of the MITC formulation clearly depends on the location of the sampling points, and their susceptibility to parasitic shear strains under the case considered. Despite this, the MITC elements have proven resistant to membrane and transverse shear locking [3] and are amongst the most widely used elements throughout FEM codes.

2.4.4. Assumed Natural Deviatoric Strains

The ANS approach was extended into Free Formulation (FF) [7], where the element stiffness matrix is the sum of a basic and higher order stiffness, by Militello and Felippa [21] under the name of the Assumed Natural Deviatoric Strains (ANDES) formulation. An advantage that the ANDES formulation inherits from the FF is that it untethers the derivation of element stiffness from the principle of minimum potential energy, the function continuity requirements of which often result in elements that "tend to be too stiff" [7]. The ANDES basic stiffness ensures consistency of the element and arises from the basic strain field comprising constant strain states and those associated with rigid body motion. Complementing this, the higher order stiffness is responsible for stability and accuracy [23] based on a enhanced strain field where the element enhancements are realised. The FF framework requires this potentially non-conforming higher order field be energy orthogonal to the basic field, which the ANDES formulation fulfils with a deviatoric higher order strain field [22]. The ANDES formulation has proven capable of alleviating membrane and transverse shear locking [35].

2.4.5. Discrete Shear Gap

The Discrete Shear Gap (DSG) approach from Bischoff and Bletzinger [12] [9] is another variant on the ANS approach with the novelty of identifying and manipulating the 'shear gap' field of the element. The shear gap, as illustrated below, is the increase of displacement due to shear, and corresponds to the difference between the actual displacement and that of pure bending (thus the shear gap is always zero in a 3 parameter Kirchhoff-Love model).

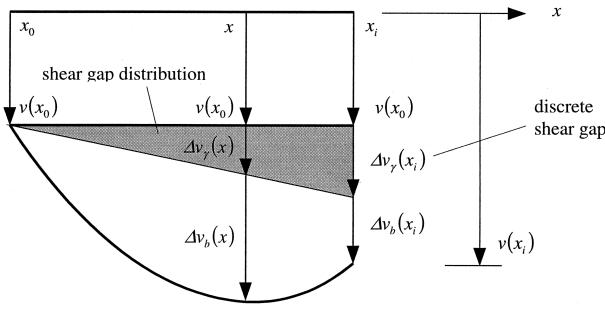


Figure 10 Discrete Shear Gap (DSG) concept [12]

The DSG method aims to set the nodal shear gaps to zero, which, in effect, alters and defines the underlying shear strain field. In bilinear rectangular applications of the DSG method, Bletzinger [12] notes that the MITC4 element is recovered. For a linear triangle element, the shear gap of only two nodes can be set to zero, rendering the element stiffness dependent on node ordering [12]. Despite this drawback, which diminishes with mesh refinement, the DSG method offers an advantage of very fast computational construction of element stiffness matrices and effective mitigation of transverse shear locking.

2.4.6. Discrete Kirchhoff Theory

Elements based on the Discrete Kirchhoff Theory (DKT) are obtained by modifying a basic 5 parameter element and ignoring the transverse shear energy [4]. Since the underlying kinematics of the 5 parameter model are different to Kirchhoff bending theory, the Kirchhoff constraints are enforced via discrete points (typically nodes and mid-edge points) along the element edges relating the rotations to translational displacements. The geometry and tying-points of the Discrete Kirchhoff Quadrilateral (DKQ) element are shown below:

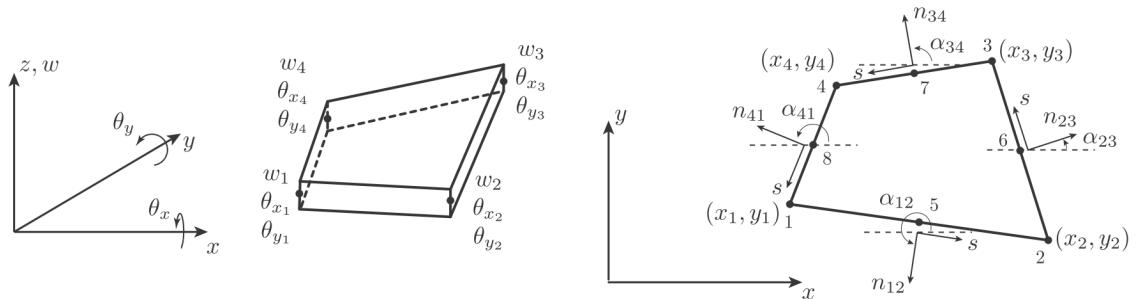


Figure 11 DKQ DOF arrangement and geometry [2]

For example, the Kirchhoff conditions are imposed at corner nodes $i = 1, 2, 3, 4$ and mid-side nodes $k = 5, 6, 7, 8$ [2]:

$$\beta_{xi} + \frac{\partial w}{\partial x}|_i = 0, \quad \beta_{yi} + \frac{\partial w}{\partial y}|_i = 0, \quad \beta_{sk} + \frac{\partial w}{\partial s}|_k = 0 \quad (2.25)$$

Mohan [34] noted that a major drawback of DKT elements is that the transverse displacement isn't explicitly defined within the interior of the element. Despite this, the advantages of DKT formulated elements is that they combine the shear locking free performance of KL models and the lower C_0 continuity requirements of RM models [11].

2.4.7. Enhanced Assumed Strains

The Enhanced Assumed Strain (EAS) approach [45] utilises the three field Hu-Washizu variational principle which allows the simultaneous variation of displacements, stresses and strains. Unlike the other technologies presented which attempt to remove problematic strain terms associated with locking, EAS derived elements feature additional enhanced strain fields designed to balance the parasitic displacement based strain terms. To prevent singular matrices the enhanced strains must be linearly independent from the displacement based strains. Furthermore, orthogonality of the stress functions to the enhanced strains must be ensured such that the associated energy vanish [16]. The application of EAS techniques to elements has been found to improve transverse shear and membrane locking performance [45] [8] [16].

2.4.8. Drilling degrees of freedom

Although drilling degrees of freedom (DOFs) don't counter locking problems, it is a commonly employed finite element technique. The common analysis of structural connections and custom steelwork are instances where shell elements will intersect with each other at arbitrary orientations. The discussion of 3 and 5 parameter shell models confirmed the nodal DOFs to be 3 translation and 2 rotational components:

$$\mathbf{v}_i^T = \begin{pmatrix} v_{xi} & v_{yi} & v_{zi} & \beta_{xi} & \beta_{yi} \end{pmatrix} \quad (2.26)$$

It can be seen that the shell formulations don't require a rotational DOF around the z axis β_{zi} , referred to as the drilling DOF. However, as discussed, shell elements in practical FEA may meet at arbitrary orientations, such as the perpendicular intersection below:

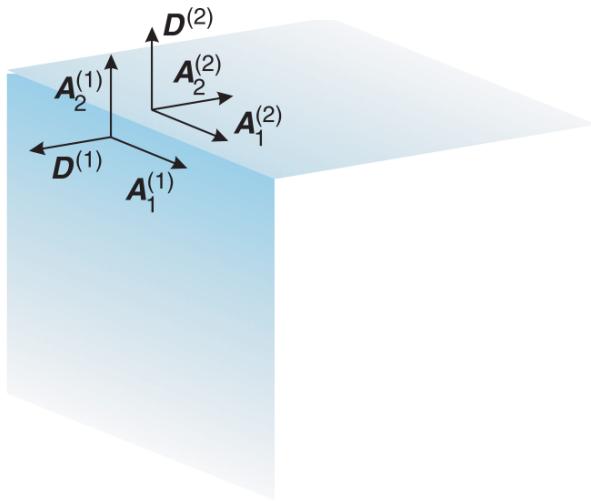


Figure 12 Shell assembly benefitting from drilling DOFs [8]

The figure above illustrates that the real twisting DOF associated with $A_2^{(2)}$ mates with the drilling DOF associated with $D^{(1)}$, which has no theoretically based stiffness value according to the shell formulations. In the current arrangement the connection will clearly be modelled too flexibly. A remedy for this is the addition of an artificial drilling DOF stiffness to the element, however the magnitude of such a fictitious torsional spring has no decisive theoretical foundation. Intuitively, it should be done on an element by element basis and should vary with the characteristic size and stiffness of the element, as opposed to a global constant drilling stiffness. Among others available, one common technique is to introduce a scaling factor (in the strain-displacement matrix or after the element stiffness matrix is constructed) which takes a fraction of the element stiffness and assigns it to the drilling DOFs.

2.4.9. Summary of selected element technologies

Following the discussion of shell models, their associated locking phenomena and element technologies, a summary of the element technologies considered with their relative merits and drawbacks is tabulated below:

Technology	Formulation	Advantages	Disadvantages
ANDES	5 parameter	Reduced membrane and transverse shear locking Relaxed higher order strain field	Locking reduction depends on tying points More complex implementation
ANS	5 parameter	Reduced membrane and transverse shear locking	Locking reduction depends on tying points
DKT	3 parameter	No transverse shear locking	Transverse disp. not explicitly defined
Drilling DOFs	-	Practical assembly of shells	Artificial stiffness
DSG	5 parameter	Reduced transverse shear locking Computationally fast	Node numbering dependency for linear triangle
EAS	5 parameter	Reduced transverse shear and membrane locking	Potentially complex implementation and possibly slower
MITC	5 parameter	Reduced membrane and transverse shear locking	Locking reduction depends on tying points
Reduced integration	-	Lowered computational cost Reduced locking	Zero energy modes Locking reduction depends on integration points

Table 1 Summary of selected element technologies

The table above confirms the "*no free lunch*" theory, with every technology having its own advantages and drawbacks. In the case of a flat shell (naturally, or via projection), where the bending and membrane response are decoupled, a single finite element can easily employ different technologies in each component.

2.5. Identification of Kratos shell element formulations

The shell elements to be implemented in Kratos are the 5 parameter (Reissner-Mindlin theory) triangular shell and the 3 parameter (Kirchhoff Love theory) quadrilateral shell. Obviously the perfect element choices for Kratos would be computationally quick, possess no locking and easy to implement, but it's clear such an element doesn't exist yet. If the requirements are relaxed to computationally quick elements that are relatively free of locking effects the following candidates are selected:

Element	Membrane formulation	Bending formulation
Thick triangular shell	DSG + Drilling DOFs	DSG
Thin quadrilateral shell	ANDES including Drilling DOFs	Discrete Kirchhoff Quadrilateral (DKQ)

Table 2 Selected formulations of implemented shell elements

With the various components of the KRATOS shell elements selected, they shall be implemented in the following sections of this work, commencing with the DSG triangle element.

Chapter 3 Composite shells - revised

1

As is the case with shell structures, anisotropic structures are widely encountered throughout the natural and man-made environment. Truly isotropic materials are but a subset of naturally occurring structures, with the phenotypical material often being anisotropic due to varying environmental pressure in different directions. Intuitively, inspecting the ongoing genetic optimization of constructions forming the natural environment suggests that the best mechanical behaviour for a structure subject to spatially non-uniform requirements will be non-uniform itself. This same philosophy has recently began to dominate the cutting edge of man-made engineering structures where specialized high-performance is demanded, leading to greater adoption of composite materials. The increasing proportion of composite materials in the aerospace industry is but one example of their increasing traction.



Figure 13 Material composition of the Boeing 787 [14]

The reason behind this increased propensity to use composites in high performance engineering applications is their customisability. Material properties such conductivity, density, wear resistance, and directional stiffness can be tailored to suit the exact needs of a design region. Directionally-varying stiffness is effectively leveraged in modern carbon-fiber road bicycles by simultaneously providing high stiffness for efficient power transfer and deliberate compliance for rider comfort and increased traction. This is one example of a composite material fulfilling seemingly mutually exclusive objectives due to finely tuned material properties built upon a knowledge of composite material basics.

3.1. Composite material basics

Composite materials are typically the combination of a high strength/stiffness fibre reinforcement material and a weaker/softer base matrix material. Although a wide range composites exist, they are broadly grouped into fibrous composites (employing fibre reinforcement materials), particulate composites (employing particle reinforcement materials) and laminate composites composed of layers of different materials, including fibrous and particulate composites. Of these, this work focusses on laminates composed of fibrous composites which are commonly used in practical engineering.

3.1.1. Laminae and laminates

Laminae are individual layers or plies of composite materials, which, when stacked together, form a laminate. Each lamina consists of a volume percentage of reinforcement fibres embedded within a matrix material, aligned at a particular orientation to some coordinate system. Common fibre reinforcement materials are various glass fibres (including E-glass and S-glass), carbon/graphite fibres and boron fibres while common matrix materials are thermosetting polymers such as polyester and epoxy resins and metals including aluminium and titanium. The reinforcement fibres may be arranged in the lamina matrix in a variety of patterns and orientations, either continuous or discontinuous [1].

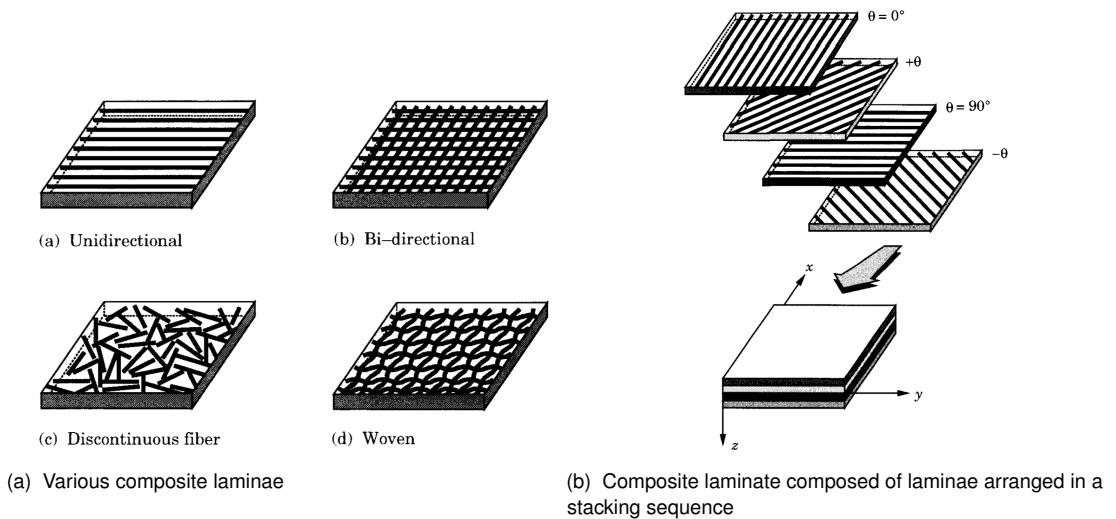


Figure 14 Components of composite laminates [44]

Considering the four laminae in figure 14(a) examples of anisotropy to varying degrees can be intuited. The unidirectional continuous fibre pattern is incredibly stiff and strong in the fibre direction while being relatively compliant and weak perpendicular to this fibre direction. Contrasting this, a practically in-plane isotropic lamina may be achieved with very fine discontinuous fibres randomly oriented within the matrix substrate. Thus, the importance of individual lamina makeup forms a critical determinant of the structural behaviour of the total

laminate as a whole. Another critical determinant of the laminate behaviour is the stacking sequence of the laminate, which prescribes the number of laminae, their vertical order and their orientation as per figure 14(b). The control that these two determinants offer designers facilitate highly optimized structures for well defined structural requirements.

An industrial example of this is spoolable Glass Reinforced Epoxy (GRE) oil piping, which has a central GRE structural lamina sandwiched between two non-structural Polyethylene (PE) laminae intended to provide the structural layer chemical and environmental protection. Furthermore, the fibre alignment of the GRE structural lamina is often orientated at an optimal pressure-capacity angle since the stress field of the pipe operating under internal pressure is well defined.

3.1.2. Constitutive equations of orthotropic laminae

A pre-requisite of modelling complex laminates is the characterisation of each lamina. The general constitutive equations of laminae are thus established under the assumption of perfectly continuous linear elastic materials without fibre breakages or matrix voids. For a generally anisotropic material the stresses can be related to the strains as follows:

$$\sigma^{ij} = C_0^{ijkl} \epsilon_{kl} \quad (3.1)$$

Equivalently, adopting Reddy's [44] notation:

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ & & C_{33} & C_{34} & C_{35} & C_{36} \\ & & & C_{44} & C_{45} & C_{46} \\ & & & & C_{55} & C_{56} \\ & & & & & C_{66} \end{pmatrix} \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{23} \\ 2\epsilon_{13} \\ 2\epsilon_{12} \end{pmatrix} \quad (3.2)$$

The preceding discussion of laminae illustrated there are innumerable design variants for any lamina. In practise however, one of the most common lamina arrangements are orthotropic laminae. Orthotropic laminae have three mutually orthogonal planes of symmetry, which reduces the unique entries of the lamina-orientated constitutive tensor from 21 (anisotropic materials) to 9:

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ & C_{22} & C_{23} & 0 & 0 & 0 \\ & & C_{33} & 0 & 0 & 0 \\ & & & C_{44} & 0 & 0 \\ & & & & C_{55} & 0 \\ & & & & & C_{66} \end{pmatrix} \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{23} \\ 2\epsilon_{13} \\ 2\epsilon_{12} \end{pmatrix} \quad (3.3)$$

sym.

Furthermore, if the lamina is suitably thin, as is typically the case in composite shell structures, a state of plane stress can be assumed. This corresponds to $\sigma_{33} = 0$ as per the following illustration.

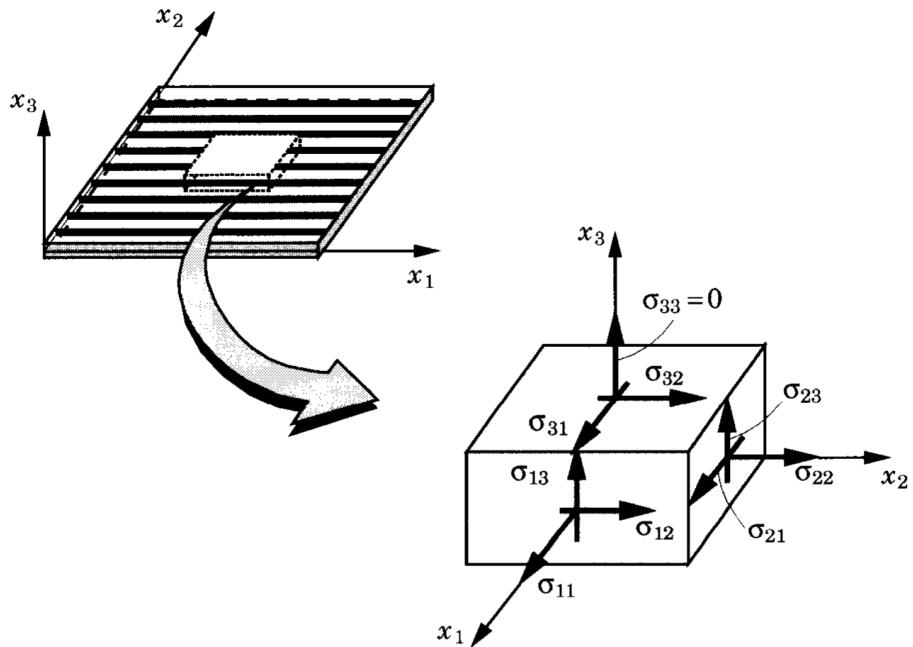


Figure 15 Lamina in a plane stress state [44]

The reduced plane stress orthotropic lamina constitutive tensor is therefore:

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{13} \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} & 0 & 0 & 0 \\ & Q_{22} & 0 & 0 & 0 \\ & & Q_{66} & 0 & 0 \\ & & & Q_{44} & 0 \\ & & & & Q_{55} \end{pmatrix} \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ 2\epsilon_{12} \\ 2\epsilon_{23} \\ 2\epsilon_{13} \end{pmatrix} \quad (3.4)$$

sym.

The entries Q_{ij} are derived from the bulk structural properties of the lamina at the macro-mechanical level. For example, the bulk lamina properties of a continuous fibre unidirectional design depicted in the following figure is considered.

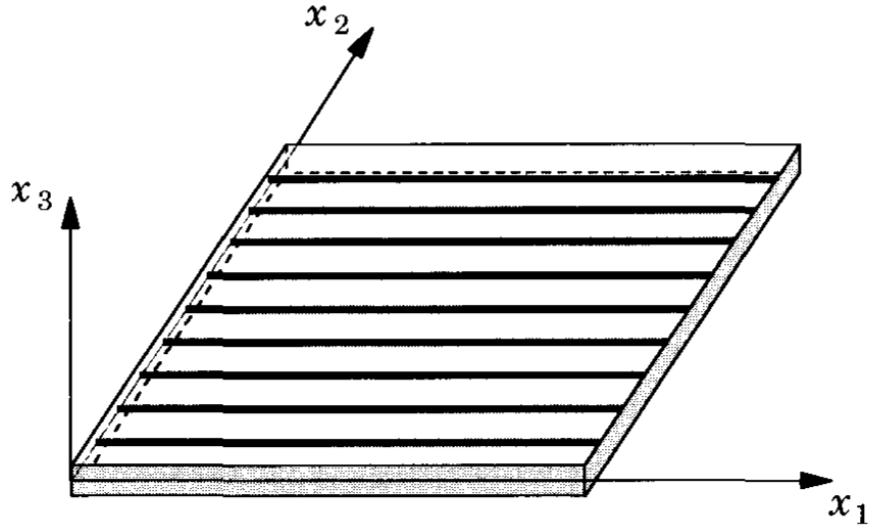


Figure 16 Continuous unidirectional fibre lamina arrangement [44]

If the Young's moduli and the Poisson ratios of both the individual fibre (E_f, ν_f) and matrix (E_m, ν_m) materials are known, along with the volume fraction of each (v_f, v_m), the resulting bulk lamina moduli can be determined.

Consider that the bulk modulus E_1 , along x_1 and parallel to the fibres, is activated by tension σ_{11} along x_1 . The constraint $\epsilon_{11_f} = \epsilon_{11_m} = \epsilon_{11}$ must be satisfied at every cross sectional area A along x_1 . Thus:

$$\sigma_{11} = E_1 \epsilon_{11}, \quad \sigma_{11_f} = E_f \epsilon_{11_f}, \quad \sigma_{11_m} = E_m \epsilon_{11_m} \quad (3.5)$$

Applying force equilibrium:

$$\sigma_{11} A = \sigma_{11_f} A v_f + \sigma_{11_m} A v_m \quad (3.6)$$

Substituting and rearranging yields the bulk longitudinal Young's modulus of the lamina:

$$E_1 = E_f v_f + E_m v_m \quad (3.7)$$

A similar line of analysis can be performed for the other lamina moduli, yielding the following results:

$$E_2 = \frac{E_f E_m}{E_f v_m + E_m v_f}, \quad \nu_{12} = \nu_f v_f + \nu_m v_m, \quad G_{12} = \frac{G_f G_m}{G_f v_m + G_m v_f} \quad (3.8)$$

The example of continuous unidirectional fibre laminae highlight that the material description of laminae can be shifted from a micro-mechanical level to a macro-mechanical level characterised by equivalent parameters. These are related to the entries Q_{ij} of the reduced plane

stress orthotropic lamina constitutive tensor in equation 3.4 as follows:

$$Q_{11} = \frac{E_1}{1 - \nu_{12}\nu_{21}}, \quad Q_{12} = \frac{\nu_{12}E_1}{1 - \nu_{12}\nu_{21}}, \quad Q_{22} = \frac{E_2}{1 - \nu_{12}\nu_{21}} \quad (3.9)$$

$$Q_{66} = G_{12}, \quad Q_{44} = G_{23}, \quad Q_{55} = G_{13} \quad (3.10)$$

The lamina macro-mechanical parameters E_1 , E_2 , ν_{12} , G_{12} , G_{23} and G_{13} can be derived from the micro-mechanical properties of the lamina, as demonstrated, or, as is more common, are obtained experimentally. Regardless of their method of origin, these parameters are always aligned with the lamina local coordinate system, which may not necessarily coincide with the laminate coordinate system, or, more generally, the global reference coordinate system of the analysis.

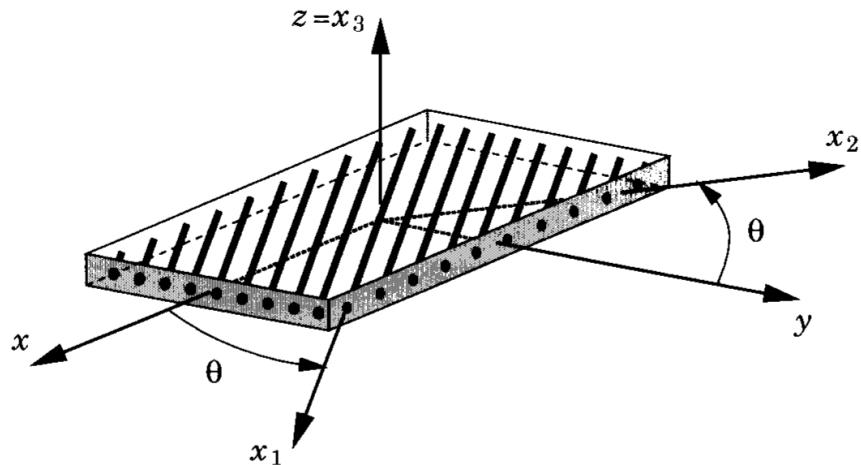


Figure 17 Arbitrary orientation of lamina [44]

The transformation of a locally oriented (x_1, x_2, x_3) lamina constitutive tensor \mathbf{Q} to one aligned with a reference coordinate system (x, y, z) $\bar{\mathbf{Q}}$ through an angle θ about $z = x_3$ is achieved as follows ($c = \cos\theta$, $s = \sin\theta$):

$$\bar{\mathbf{Q}} = \mathbf{T}^T \mathbf{Q} \mathbf{T}, \quad \mathbf{T} = \begin{pmatrix} c^2 & s^2 & -2sc & 0 & 0 \\ s^2 & c^2 & 2sc & 0 & 0 \\ sc & -sc & c^2 - s^2 & 0 & 0 \\ 0 & 0 & 0 & c & s \\ 0 & 0 & 0 & -s & c \end{pmatrix} \quad (3.11)$$

Thus, stresses and strains in the reference coordinate system (x, y, z) are related as such:

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \\ \sigma_{yz} \\ \sigma_{xz} \end{pmatrix} = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} & 0 & 0 \\ & \bar{Q}_{22} & \bar{Q}_{26} & 0 & 0 \\ & & \bar{Q}_{66} & 0 & 0 \\ & & & \text{sym.} & \bar{Q}_{44} & \bar{Q}_{45} \\ & & & & & \bar{Q}_{55} \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix} \quad (3.12)$$

This transformation is required correctly assembly laminae in laminates that have non-zero stacking angles, which, by virtue of their customizability, is almost all laminates. With the individual lamina constitutive behaviour detailed, orthotropic shell laminates can now be built up and modelled.

3.2. Orthotropic shell laminates: internal virtual work

Orthotropic shell laminates are composites made from a stacking sequence of orthotropic laminae. As per their application in shells, the assumption of plane stress is continued.

The 5 parameter shell theory internal virtual work is recalled as:

$$-\delta\Pi_{int} = \int_{\Omega} \boldsymbol{\epsilon} : \mathbf{C}_{mem} : \delta\boldsymbol{\epsilon} \, d\Omega + \int_{\Omega} \boldsymbol{\kappa} : \mathbf{C}_{bend} : \delta\boldsymbol{\kappa} \, d\Omega + \int_{\Omega} \boldsymbol{\gamma} : \mathbf{C}_{shear} : \delta\boldsymbol{\gamma} \, d\Omega \quad (3.13)$$

The integral over the volume can split into area and laminate thickness integrals:

$$-\delta\Pi_{int} = \int_h \int_A \boldsymbol{\sigma}_{mem} : \delta\boldsymbol{\epsilon} \, dAdh + \int_h \int_A \boldsymbol{\sigma}_{bend} : \delta\boldsymbol{\kappa} \, dAdh + \int_h \int_A \boldsymbol{\tau} : \delta\boldsymbol{\gamma} \, dAdh \quad (3.14)$$

By pre-integrating the stress quantities, and restricting the scope of the equations to 2D plane stress conditions, the following equivalent form can be presented in vector notation:

$$-\delta\Pi_{int} = \int_A \mathbf{N}^T \delta\boldsymbol{\epsilon} \, dA + \int_A \mathbf{M}^T \delta\boldsymbol{\kappa} \, dA + \int_A \mathbf{Q}^T \delta\boldsymbol{\gamma} \, dA \quad (3.15)$$

The introduced quantities \mathbf{N} , \mathbf{M} and \mathbf{Q} are force and moment resultants over the entire laminate as per the following figure:

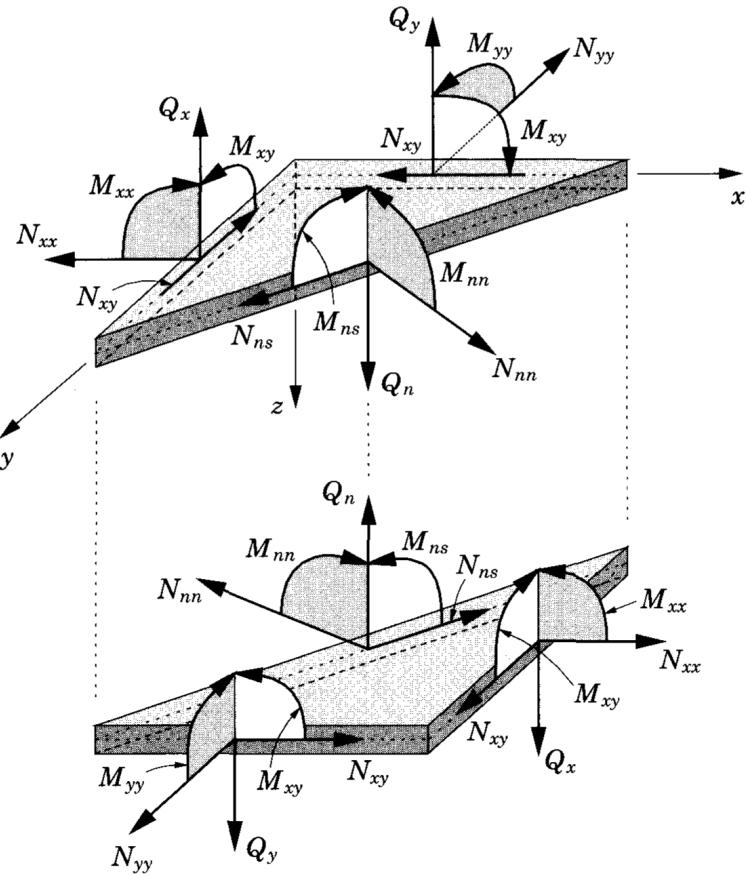


Figure 18 Force and moment resultants of a plate [44]

The force and moment resultants are defined as follows:

$$\mathbf{N} = \begin{pmatrix} N_{xx} \\ N_{yy} \\ N_{xy} \end{pmatrix} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix} dz, \quad \mathbf{M} = \begin{pmatrix} M_{xx} \\ M_{yy} \\ M_{xy} \end{pmatrix} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix} z dz \quad (3.16)$$

The shear resultants only applicable for the 5 parameter model are similarly defined, including a shear energy correction factor of $\alpha = \frac{5}{6}$:

$$\mathbf{Q} = \begin{pmatrix} Q_x \\ Q_y \end{pmatrix} = \alpha \int_{-\frac{h}{2}}^{\frac{h}{2}} \begin{pmatrix} \sigma_{xz} \\ \sigma_{yz} \end{pmatrix} dz \quad (3.17)$$

3.2.1. Laminate constitutive equations

At this point, the laminate force resultants, by way of the stress integrals, must be related back to strains across all laminae via laminate constitutive equations. A laminate of total thickness

h with n laminae is considered below:

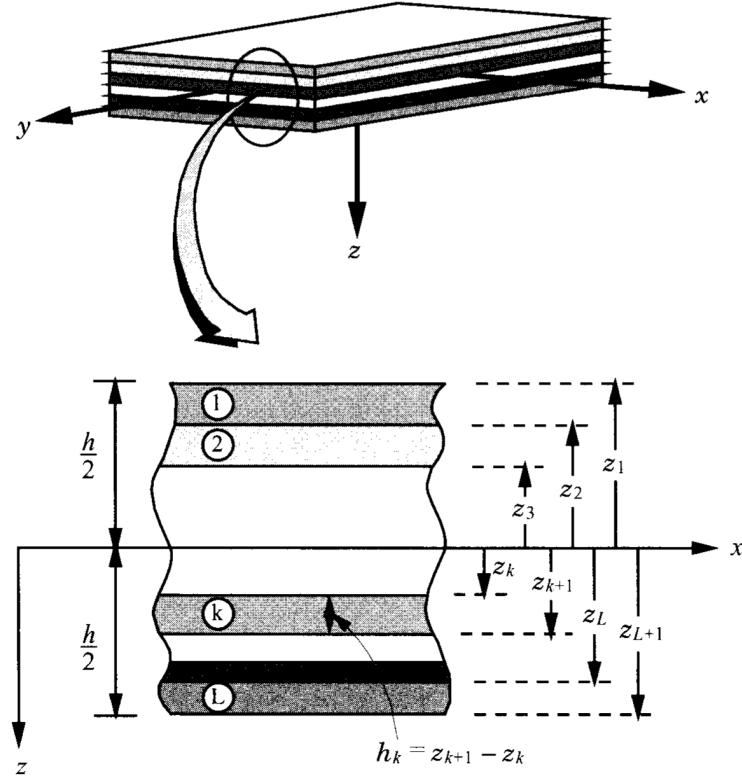


Figure 19 Coordinate system and lamina numbering in a laminate [44]

The force resultants can be determined as:

$$\mathbf{N} = \begin{pmatrix} N_{xx} \\ N_{yy} \\ N_{xy} \end{pmatrix} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix} dz = \sum_{k=1}^n \int_{z_k}^{z_{k+1}} \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix} dz \quad (3.18)$$

Invoking the laminae constitutive properties previously established in equation 3.12 leads to the 'mathematical assemblage' of the laminae into the laminate via the following integral:

$$\begin{aligned} \begin{pmatrix} N_{xx} \\ N_{yy} \\ N_{xy} \end{pmatrix} &= \sum_{k=1}^n \int_{z_k}^{z_{k+1}} \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} \\ \bar{Q}_{12} & \bar{Q}_{22} & \bar{Q}_{26} \\ \bar{Q}_{16} & \bar{Q}_{26} & \bar{Q}_{66} \end{pmatrix}^{(k)} \begin{pmatrix} \epsilon_{xx} + z\kappa_{xx} \\ \epsilon_{yy} + z\kappa_{xx} \\ 2\epsilon_{xy} + 2z\kappa_{xy} \end{pmatrix} dz \\ &= \begin{pmatrix} A_{11} & A_{12} & A_{16} \\ A_{12} & A_{22} & A_{26} \\ A_{16} & A_{26} & A_{66} \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \end{pmatrix} + \begin{pmatrix} B_{11} & B_{12} & B_{16} \\ B_{12} & B_{22} & B_{26} \\ B_{16} & B_{26} & B_{66} \end{pmatrix} \begin{pmatrix} \kappa_{xx} \\ \kappa_{xx} \\ 2\kappa_{xy} \end{pmatrix} \end{aligned} \quad (3.19)$$

Similarly, the moment and shear force resultants can be related to strains:

$$\begin{aligned} \begin{pmatrix} M_{xx} \\ M_{yy} \\ M_{xy} \end{pmatrix} &= \sum_{k=1}^n \int_{z_k}^{z_{k+1}} \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} \\ \bar{Q}_{12} & \bar{Q}_{22} & \bar{Q}_{26} \\ \bar{Q}_{16} & \bar{Q}_{26} & \bar{Q}_{66} \end{pmatrix}^{(k)} \begin{pmatrix} \epsilon_{xx} + z\kappa_{xx} \\ \epsilon_{yy} + z\kappa_{xx} \\ 2\epsilon_{xy} + 2z\kappa_{xy} \end{pmatrix} z \, dz \\ &= \begin{pmatrix} B_{11} & B_{12} & B_{16} \\ B_{12} & B_{22} & B_{26} \\ B_{16} & B_{26} & B_{66} \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \end{pmatrix} + \begin{pmatrix} D_{11} & D_{12} & D_{16} \\ D_{12} & D_{22} & D_{26} \\ D_{16} & D_{26} & D_{66} \end{pmatrix} \begin{pmatrix} \kappa_{xx} \\ \kappa_{yy} \\ 2\kappa_{xy} \end{pmatrix} \end{aligned} \quad (3.20)$$

$$\begin{pmatrix} Q_x \\ Q_y \end{pmatrix} = \alpha \sum_{k=1}^n \int_{z_k}^{z_{k+1}} \begin{pmatrix} \bar{Q}_{44} & \bar{Q}_{45} \\ \bar{Q}_{45} & \bar{Q}_{55} \end{pmatrix}^{(k)} \begin{pmatrix} 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix} dz = \alpha \begin{pmatrix} A_{44} & A_{45} \\ A_{45} & A_{55} \end{pmatrix} \begin{pmatrix} 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix} \quad (3.21)$$

The three introduced matrices represent the extensional stiffnesses A_{ij} , the bending stiffnesses D_{ij} and the bending-extensional coupling stiffnesses B_{ij} , and are determined from the lamina stiffnesses $\bar{Q}_{ij}^{(k)}$:

$$A_{ij} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \bar{Q}_{ij} \, dz, \quad B_{ij} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \bar{Q}_{ij} z \, dz, \quad D_{ij} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \bar{Q}_{ij} z^2 \, dz \quad (3.22)$$

By organising the resultants into a generalised resultant vector $\bar{\mathbf{N}}$ and the strains into a generalized strain vector $\bar{\boldsymbol{\epsilon}}$, the following summary is produced:

$$\bar{\mathbf{N}} = \bar{\mathbf{C}} \bar{\boldsymbol{\epsilon}} = \begin{pmatrix} N_{xx} \\ N_{yy} \\ N_{xy} \\ M_{xx} \\ M_{yy} \\ M_{xy} \\ Q_x \\ Q_y \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{0} \\ \mathbf{B} & \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha \begin{pmatrix} A_{44} & A_{45} \\ A_{45} & A_{55} \end{pmatrix} \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \\ \kappa_{xx} \\ \kappa_{yy} \\ 2\kappa_{xy} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix} \quad (3.23)$$

The internal virtual work of the 5 parameter shell model presented in equation 3.15 can therefore be reduced down to:

$$-\delta\Pi_{int} = \int_A \bar{\mathbf{N}}^T \delta\bar{\boldsymbol{\epsilon}} dA \quad (3.24)$$

3.3. Laminate strain and stress recovery

Critical to the practical design of laminates is the evaluation of stresses and strains not only on the laminate mid-plane, but throughout the thickness across laminae. The following describes the procedure of recovering these quantities at arbitrary positions within the laminate thickness.

3.3.1. Laminate strain recovery

Owing to the dimensional reduction from 3D to 2D, the generalized shell strains $\bar{\boldsymbol{\epsilon}}$ of equation 3.23 are referred to the mid-plane ($z = 0$) of the laminate. Thus, within a local convective laminate coordinate system describing the z-axis orientation as per figure 19:

$$\bar{\boldsymbol{\epsilon}}(x, y) = \boldsymbol{\epsilon}(x, y, 0) \quad (3.25)$$

A consequence of the straight director assumption common to both the 3 and 5 parameter shell models is the membrane strains vary proportionally to the distance from the mid-plane. This can be observed in the following diagram of a plate in pure bending highlighting the linear variation of in-plane stresses (which, in simple cases, are merely scaled strains):

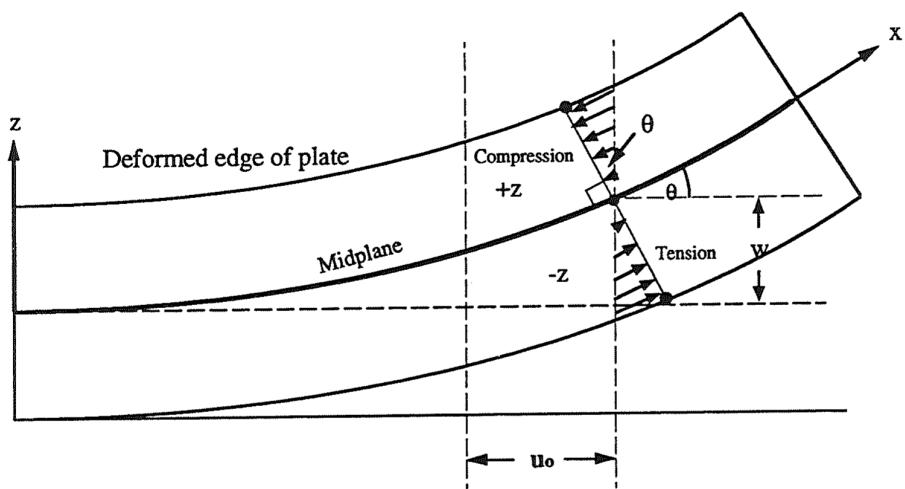


Figure 20 Deformation of 3 parameter plate [36]

Thus the in-plane strains at any position z in the laminate are determined from:

$$\begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \end{pmatrix}_{(z)} = \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \end{pmatrix}_{(z=0)} + z \begin{pmatrix} \kappa_{xx} \\ \kappa_{yy} \\ 2\kappa_{xy} \end{pmatrix}_{(z=0)} \quad (3.26)$$

The in-plane strains of a surface in any lamina within the laminate can be determined by appropriately setting the corresponding z value.

Applicable only to 5 parameter shell models is the recovery of transverse shear strains. A perusal of Kreja's literature review on composite models [29] indicates the amount of academic effort dedicated to accurately modelling transverse shear strains in first order shear deformation theories. The contradiction driving this effort is that 5 parameter models are limited to expressing constant transverse shear strain over the shell thickness, while it is known that for an isotropic material they are actually distributed parabolically. This is the reason behind the shear correction factor of $\alpha = \frac{5}{6}$ which is actually the ratio of internal transverse shear strain energy in a 5 parameter plate to that of 3D elasticity. Without delving into exotic laminate shell theories, two options for recovering the transverse shear strain avail themselves: (1) accept the limitations of the 5 parameter model and consider the transverse shear strain constant across the section, or; (2) reconstruct a parabolic profile from the mid-plane values, as is often done for isotropic materials.

The first approach of accepting constant transverse shear stress is expressed as follows:

$$\begin{pmatrix} 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix}_{(z)} = \begin{pmatrix} 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix}_{(z=0)} \quad (3.27)$$

The second approach of constructing a parabolic strain profile using limiting expressions from [19] is described below, starting with the distribution of the transverse shear stress σ_{xz} in an isotropic plate:

$$\sigma_{xz}(z) = \sigma_{xz}^{max} \left(1 - \frac{4z^2}{h^2}\right) = \frac{3Q_x}{2h} \left(1 - \frac{4z^2}{h^2}\right) = \frac{3}{2} \sigma_{xz}^{av} \left(1 - \frac{4z^2}{h^2}\right) \quad (3.28)$$

Unlike laminate stresses, which are generally discontinuous across the thicknesss, laminate strains are continuous. Therefore, the strain distribution of the isotropic plate will be transferred to the laminate. The strain distribution of an isotropic plate must follow the stress distribution, thus:

$$\gamma_{xz}(z) = \frac{3}{2} \gamma_{xz}^{av} \left(1 - \frac{4z^2}{h^2}\right) = \frac{3}{2} 2\epsilon_{xz}^{(z=0)} \left(1 - \frac{4z^2}{h^2}\right) \quad (3.29)$$

Summarising, the approximated transverse shear distribution for the laminate is:

$$\begin{pmatrix} 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix}_{(z)} = \frac{3}{2}(1 - \frac{4z^2}{h^2}) \begin{pmatrix} 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix}_{(z=0)} \quad (3.30)$$

The in-plane and transverse shear strains determined thus far are referred to the reference coordinate system of the laminate (x, y, z). However, it is often of interest to transform these strains so they are aligned with the individual lamina considered (x_1, y_1, z_1). For a lamina oriented at θ to the laminate coordinate system, the strain components can be transformed as such ($c = \cos\theta, s = \sin\theta$):

$$\begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ 2\epsilon_{12} \\ 2\epsilon_{23} \\ 2\epsilon_{13} \end{pmatrix} = \begin{pmatrix} c^2 & s^2 & sc & 0 & 0 \\ s^2 & c^2 & -sc & 0 & 0 \\ -\sin 2\theta & \sin 2\theta & c^2 - s^2 & 0 & 0 \\ 0 & 0 & 0 & c & -s \\ 0 & 0 & 0 & s & c \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix} \quad (3.31)$$

3.3.2. Laminate stress recovery

With the laminate strains across the thickness described, the stresses can be recovered by simply applying the considered lamina constitutive law as per equation 3.12. Naturally, the material coefficients Q_{ij} should correspond to the k^{th} lamina considered at the height of inspection z :

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \\ \sigma_{yz} \\ \sigma_{xz} \end{pmatrix}_{(z)} = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} & 0 & 0 \\ & \bar{Q}_{22} & \bar{Q}_{26} & 0 & 0 \\ & & \bar{Q}_{66} & 0 & 0 \\ & & & \text{sym.} & \\ & & & & \bar{Q}_{44} & \bar{Q}_{45} \\ & & & & & \bar{Q}_{55} \end{pmatrix}_{(k)} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix}_{(z)} \quad (3.32)$$

As previously implied, although the strain distribution is continuous across laminates, the stress distribution is generally discontinuous due to the varying material properties of the laminae. Indeed, if the laminae of a laminate are the exact same orthotropic composite material, but are stacked at varying angles, the resulting stress distribution will be discontinuous. The following example of a 4 ply laminate with a [0, 45, 45, 0] stacking sequence subject to pure bending highlights the nature of stress and strain distributions through the thickness despite all laminae being the same material.

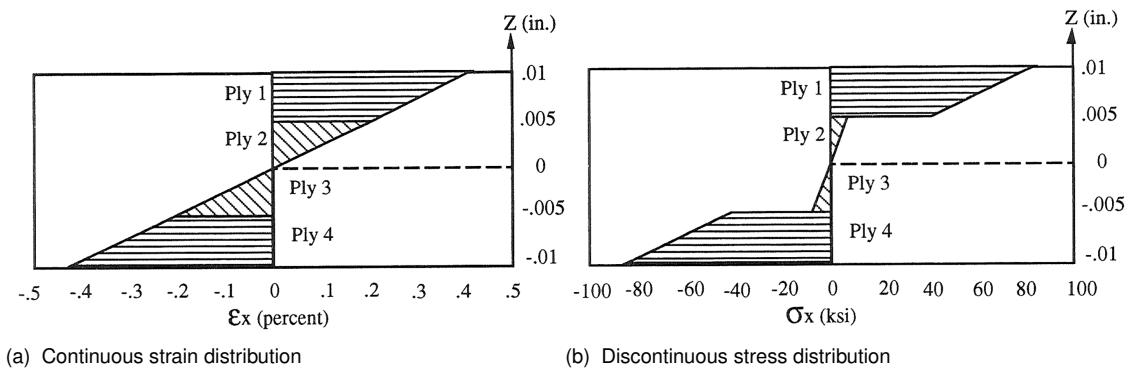


Figure 21 Stress and strain thickness distribution of 4 ply plate subject to pure bending [36]

Similar to the laminate strains, the in-plane and transverse shear stresses determined thus far are referred to the reference coordinate system of the laminate. For a lamina oriented at θ to the laminate coordinate system (x, y, z), the stress components can be transformed to the lamina coordinate system (x_1, y_1, z_1) as such ($c = \cos\theta, s = \sin\theta$):

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ 2\sigma_{12} \\ 2\sigma_{23} \\ 2\sigma_{13} \end{pmatrix} = \begin{pmatrix} c^2 & s^2 & \sin 2\theta & 0 & 0 \\ s^2 & c^2 & -\sin 2\theta & 0 & 0 \\ -sc & sc & c^2 - s^2 & 0 & 0 \\ 0 & 0 & 0 & c & -s \\ 0 & 0 & 0 & s & c \end{pmatrix} \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ 2\sigma_{xy} \\ 2\sigma_{yz} \\ 2\sigma_{xz} \end{pmatrix} \quad (3.33)$$

3.4. Laminate Tsai-Wu failure criterion

The Tsai-Wu failure criterion is a relatively simple method to approximate the safety of composite materials under combined loading. As an extension of the Von Mises distortion energy theory, it provides an invariant scalar indication of how proximate the current stress state of a lamina is to breaching the closed convex failure surface [25]. The closed convex failure surface is defined by material strength parameters F_i and F_{ij} , with failure predicted by fulfilling the following inequality [44]:

$$\sum_{i=1}^6 F_i \sigma_i + \sum_{i=1}^6 \sum_{j=1}^6 F_{ij} \sigma_{ij} \geq 1.0 \quad \text{with} \quad \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} = \begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{pmatrix} \quad (3.34)$$

By considering the tensile (X_T, Y_T, Z_T) and compressive (X_C, Y_C, Z_C) strengths in the lamina principal material directions (1, 2, 3) and the shear strengths (R, S, T) corresponding to (23, 13, 12), the non-zero material strength parameters in a plane stress regime are expressed as [44]:

$$\begin{aligned} F_1 &= \frac{1}{X_T} - \frac{1}{X_C}, & F_2 &= \frac{1}{Y_T} - \frac{1}{Y_C}, & F_{11} &= \frac{1}{X_T X_C}, & F_{22} &= \frac{1}{Y_T Y_C} \\ F_{44} &= \frac{1}{R^2}, & F_{55} &= \frac{1}{S^2}, & F_{66} &= \frac{1}{T^2}, & F_{12} &= \frac{-0.5}{\sqrt{X_T X_C Y_T Y_C}} \end{aligned} \quad (3.35)$$

The strength parameter F_{12} is termed the in-plane interaction factor and is generally difficult to determine experimentally. Thus, it is often approximated from other more readily available strengths, as per equation 3.35 or set to zero entirely. Further information regarding F_{12} can be found in Reference [25]. In the scope of this work F_{12} is determined as per equation 3.35.

Although equation 3.34 describes the Tsai-Wu criterion, it does so in terms of a Failure Index (FI). That is, if the calculated FI is greater than 1.0, failure occurs. Contrasting this, engineers are often interested in safety factors, or more specifically, the ratio with which to increase loads to just achieve failure. This parameter is commonly referred to as either the Reserve Factor (RF) or Strength Ratio (SR), with values less than 1.0 indicating failure while those above 1.0 correspond to safety. The Tsai-Wu RF can be determined as follows [28]:

$$RF = \frac{-b + \sqrt{b^2 + 4a}}{2a} \quad \text{with} \quad a = \sum_{i=1}^6 \sum_{j=1}^6 F_{ij} \sigma_{ij}, \quad b = \sum_{i=1}^6 F_i \sigma_i \quad (3.36)$$

Chapter 4 Non-linear analysis background

- revised 1

THE expedient practical design of engineering structures is often considered fulfilled after the confirmation of allowable deflections and stresses throughout the entire domain. Although the design may in fact be adequate in the vast majority of cases, structural stability is an oft overlooked phenomena in "bread and butter" engineering analysis and is by no means guaranteed by acceptable displacements and stresses. Indeed, some of the most catastrophic structural failures such as the Tacoma Narrows Bridge (1940) and the Twin Towers (2001) have root causes stemming from system instability and reinforce the importance of stability analysis.

Structural stability, expressed crudely, can be thought of as "the power to recover equilibrium", or, alternatively, a structure can be thought stable at an equilibrium position if it returns to that position following a temporary perturbation [20]. In terms more relatable, it's clear a person standing on one foot is more susceptible to falling over in the presence of a strong wind than someone firmly planted with both feet. Although the person with both feet planted may sway and bob, they are more likely to return to their original equilibrium position after the wind has subsided, that is, they are more stable. In an engineering context, perhaps the most well known stability problem is a long beam subject to a compressive end load: the classic Euler buckling case. At low load levels the beam will undergo small deformations and trace back to its original configuration after unloading, however, after the critical Euler buckling load is surpassed structural stability is lost giving way to large arbitrary deformations with non-coincident loading and unloading paths. Although these two simple examples provide a phenomenological picture of structural stability, a more precise engineering description of stability is required for general computational mechanics.

4.1. Response diagrams

Essential to the understanding of structural stability and non-linear FEM is the concept of an equilibrium path, often plotted on load-deflection response diagrams. These diagrams are

commonly two dimensional x-y diagrams that plot a representative force quantity against a characterising displacement thereby ascertaining the behaviour of the structure [17].

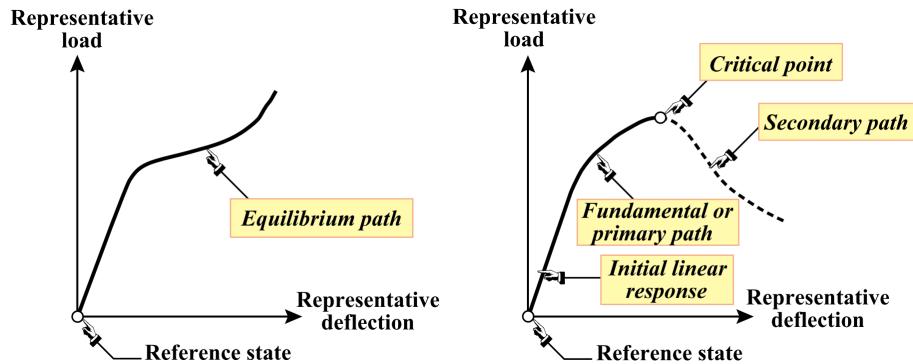


Figure 22 Examples of response diagrams [17]

The equilibrium path of a response diagram indicates the points in load-deflection space where the structure is in equilibrium, that is, the residual vector \mathbf{r} of the system vanishes. If the scope of structures considered is limited to static linear elastic structures subject to conservative loading characterised by a load factor λ linearly scaling the external load vector \mathbf{f} , the residual itself is defined as the gradient of the structure's energy Π [18]:

$$\mathbf{r}(\mathbf{u}, \lambda) = \frac{\partial \Pi(\mathbf{u}, \lambda)}{\partial \mathbf{u}} \quad (4.1)$$

The tangent stiffness matrix of the system is intuitively the slope of the equilibrium path, or alternatively, the Hessian of the structure's energy:

$$\mathbf{K} = \frac{\partial \mathbf{r}(\mathbf{u}, \lambda)}{\partial \mathbf{u}} = \frac{\partial^2 \Pi(\mathbf{u}, \lambda)}{\partial \mathbf{u} \partial \mathbf{u}} \quad (4.2)$$

Recalling that the equilibrium path of a structure is defined by a vanishing residual, it can be presented in a variety of forms:

$$\mathbf{r}(\mathbf{u}, \lambda) = \frac{\partial \Pi(\mathbf{u}, \lambda)}{\partial \mathbf{u}} = \mathbf{K}(\mathbf{u}, \lambda)\mathbf{u} - \mathbf{f}(\mathbf{u}, \lambda) = 0 \quad (4.3)$$

The above expansion offers a variety of perspectives in which to interpret the equilibrium path. A nod to virtual work methods via variation of system energy is present as well as a simple re-arrangement of the well known $\mathbf{K}\mathbf{u} = \mathbf{f}$ introduced in Bachelor FEM courses.

4.2. Stability criterion

If the aforementioned Euler beam buckling example is fully derived, it is apparent that the well known Euler critical load formulae precipitate by setting the determinant of the respective system stiffness matrices to zero. In general, loss of structural stability only occurs at critical points $(\mathbf{u}_c, \lambda_c)$ where the determinant of the tangent stiffness matrix vanishes:

$$\det[\mathbf{K}(\mathbf{u}_c, \lambda_c)] = 0 \quad (4.4)$$

In this Euler case, the instability is manifested by way of bifurcation, commonly referred to as buckling. Bifurcation points designate a critical point in the load-displacement space of the structure where two or more equilibrium paths meet. At these points structure may unpredictably switch equilibrium paths physically manifesting itself as dramatic large deflections. Indeed, in the buckling of a circular-sectioned beam, the direction of buckling deformation is entirely unpredictable (bar coaxing imperfections) with many equilibrium paths coincident at the bifurcation point. Limit, or snap-through, points are the other type of instability that may occur at critical points. In this regime, the critical point coincides with a minimum, maximum or inflection of the load parameter λ . For an everyday example of snap-through, one can consider an umbrella suddenly inverting in a strong storm, demonstrating that once the snapping wind load is reached predictably large deformations are subsequently observed.

In the scope of this work the load magnitude that introduces system instability is of key interest, details regarding how to mathematically distinguish between limit and bifurcation points once a critical point has been identified fall outside the domain surveyed.

The following figure 23(a) illustrates the nature of both instability types on a load-deflection diagram.

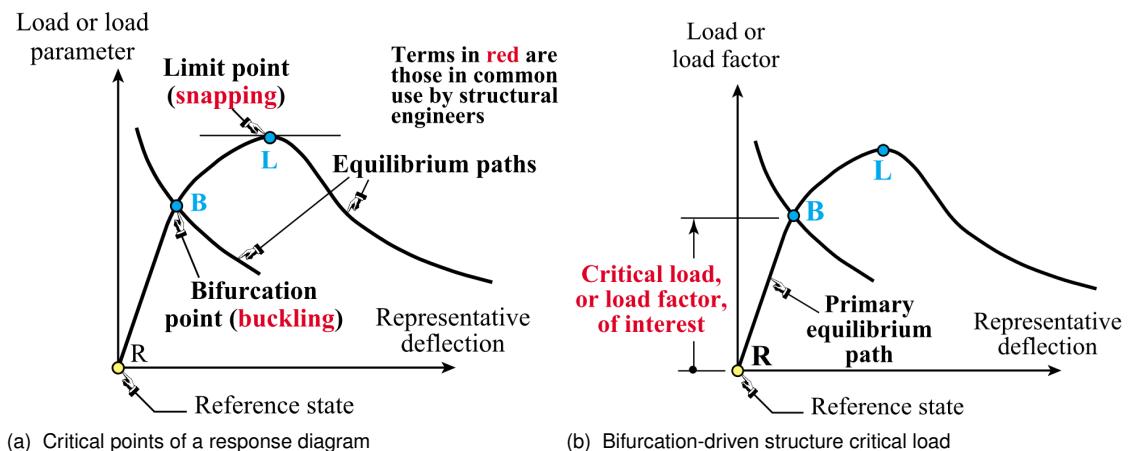


Figure 23 Stability analysis response diagrams [20]

It's clear that after both critical points, as depicted in figure 23(a), the structure loses a significant proportion of its stiffness indicated by the reduced load required for continual increases of deflection. In this post-buckling state the original function, and indeed safety, of the structure (for example, a roof or building) is likely to be significantly compromised. If such a structural critical point exists at a load level less than the design load cases considered for allowable deflections and stresses then the structure cannot be regarded as properly engineered. Figure 23(b) indicates the critical stability load of the structure, which is the load corresponding to the first critical point on the response diagram. Another, more stringent, approach is to determine the lowest load factor throughout the whole equilibrium path, including the post-critical regime, and take this as the critical stability load.

Despite the need for stability analysis to be regularly considered throughout engineering design, the effort and level of analysis required is clearly quite significant for even simple systems. As such, a linearised approach termed Linear Prebuckling analysis, is often employed.

4.3. Linear Prebuckling analysis

Linear Prebuckling (LPB) analysis offers insight, albeit reduced, into the stability characteristics of structures at a fraction of the computational expense required for a full non-linear equilibrium path analysis.

In order to further interrogate the nature of structural stability, and how it may be simplified, the tangent stiffness matrix in equation 4.4 can be decomposed. The total tangent matrix is made up of the material stiffness matrix \mathbf{K}_m (itself composed of the elastic stiffness \mathbf{K}_e and initial displacement stiffness \mathbf{K}_u) and the geometric stiffness matrix \mathbf{K}_g . Thus equation 4.4 can be expanded accordingly:

$$\det[\mathbf{K}_e + \mathbf{K}_u(\mathbf{u}_0) + \mathbf{K}_g(\mathbf{u}_c, \lambda_c)] = \det[\mathbf{K}_m + \mathbf{K}_g(\mathbf{u}_c, \lambda_c)] = 0 \quad (4.5)$$

It's apparent the expression can be characterized as an eigenvalue problem, thus the eigen-vector $\mathbf{z} \neq \mathbf{0}$ corresponding to bifurcation mode shapes can be introduced:

$$\det[\mathbf{K}_m + \mathbf{K}_g(\mathbf{u}_c, \lambda_c)]\mathbf{z} = 0 \quad (4.6)$$

The non-linear eigenvalue problem can be presented in a linearised form:

$$\det[\mathbf{K}_m + \hat{\lambda}\mathbf{K}_g(\mathbf{u}_i, \lambda_i)]\mathbf{z} = 0 \quad (4.7)$$

Equations 4.6 and 4.7 are identical if $\hat{\lambda} = 1$, $\lambda_i = \lambda_c$ and $u_i = u_c$.

If the structure under analysis is considered suitably stiff, with approximately infinitesimal displacements $\mathbf{u} \approx \mathbf{0}$ in the pre-buckling regime and negligible effects of initial displacements ($\mathbf{K}_u(\mathbf{u} \approx \mathbf{0}) \approx \mathbf{0}$), then a simplified stability eigen-problem can be presented:

$$\det[\mathbf{K}_e + \lambda \mathbf{K}_g(\lambda_{ref})] \mathbf{z} = 0 \quad (4.8)$$

The LPB equation above relies on the additional assumptions that the structure remains linearly elastic until buckling and the structure and loading have no imperfections [20]. It's also clear that the geometric stiffness matrix is assumed to scale proportionally with the load factor, which further relies on the assumption of a linearly elastic structure. Along with these assumptions, LPB analysis is subject to some notable limitations. The first of which is the typical over-estimation of the true stability limit, with accuracy deteriorating as the pre-critical structural behaviour exhibits more non-linearity. Secondly, LPB analysis is unable to distinguish between bifurcation and limit points, with all critical points transformed into bifurcation points. A consequence of this is that the recovered eigenvector associated with a limit point reported as a bifurcation point will be erroneous.

Despite the large swathe of assumptions and limitations accompanying LPB, it is no doubt a sensible and computationally-efficient approach when applied to suitably stiff linear elastic structures, which indeed include a considerable amount of industrial engineering structures.

4.4. Stability analysis example

To illustrate the various aspects stability analysis, including response diagrams and the differences between non-linear and linear critical point analysis, an example Mises two truss system is considered. The full analysis can be found in Appendix A, with only key results reproduced below. The system under consideration is depicted in the following figure:

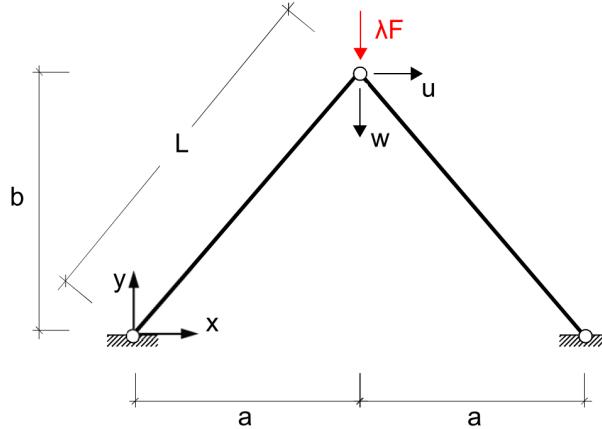


Figure 24 Mises truss geometry

After developing the system's virtual work expression via the 2nd Piola-Kirchhoff stress measure and the conjugate Green-Lagrange strain measure, the residual vector can be expressed as:

$$\mathbf{r} = \begin{pmatrix} \frac{EA}{L^3}[u^3 + uw^2 - 2bwu + 2a^2u] \\ \frac{EA}{L^3}[u^2w + w^3 - 3bw^2 - bu^2 + 2b^2w] - \lambda F \end{pmatrix} = 0 \quad (4.9)$$

With the residual vector in hand, the system's stiffness matrix can be expressed as:

$$\mathbf{K} = \frac{\partial \mathbf{r}}{\partial \mathbf{u}} = \frac{EA}{L^3} \begin{pmatrix} 3u^2 + w^2 - 2bw + 2a^2 & 2uw - 2bu \\ 2uw - 2bu & u^2 + 3w^2 - 6bw + 2b^2 \end{pmatrix} \quad (4.10)$$

Non-linear critical points can be determined by setting the determinant of the stiffness matrix to zero and solving. For the system parameters of ($a = b = EA = F = 1$) and under the assumption of $u = 0$ before instability, the non-linear critical points are calculated to be:

$$\mathbf{P}_{NL1} = \begin{pmatrix} u_{c1} = 0 \\ w_{c1} = 0.4226 \\ \lambda_{c1} = 0.1361 \end{pmatrix}, \quad \mathbf{P}_{NL2} = \begin{pmatrix} u_{c2} = 0 \\ w_{c2} = 1.5774 \\ \lambda_{c2} = -0.1361 \end{pmatrix} \quad (4.11)$$

Although the non-linear critical points have been determined, a LPB analysis of the system will be carried out to compare results between the different approaches. As per equation 4.8, a LPB analysis is an eigenvalue problem which, after assuming small displacements and $u = 0$

in the pre-buckling regime, reduces to the following expression:

$$\det \begin{pmatrix} \frac{EA}{L^3} 2a^2 - \frac{\lambda F}{b} & 0 \\ 0 & \frac{EA}{L^3} 2b^2 - \frac{\lambda F}{b} \end{pmatrix} = 0 \quad (4.12)$$

The eigenvalues of the above expression coalesce into the same value:

$$\lambda_{lpb\ c1} = \lambda_{lpb\ c2} = 0.7071 \quad (4.13)$$

It's clear that the LPB analysis is quite inaccurate for this structure with the estimated onset of instability occurring at over 5 times that of the first non-linear buckling load $\lambda_{c1} = 0.1361$. This inaccuracy is due to the fact that the structure is quite flexible in the pre-critical regime, which violates the small deflections assumption vital to the accurate use of LPB analysis.

The following response diagram plots the equilibrium path λ vs. w (assuming $u = 0$ throughout), determinant of the stiffness matrix and the LPB critical load limit over vertical displacement w .

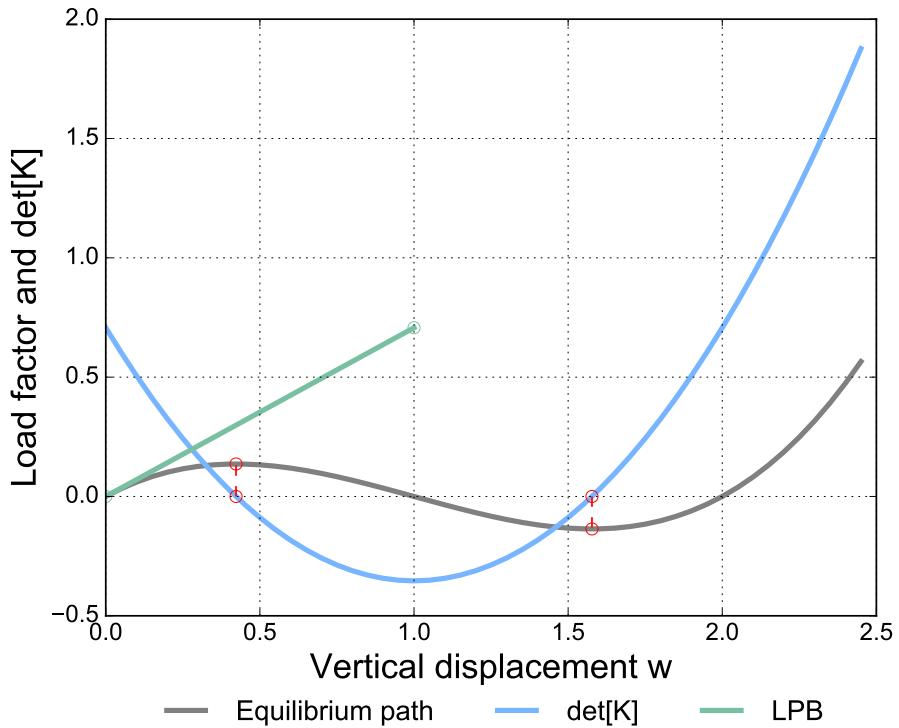


Figure 25 Ratio of Mises truss LPB and NL critical load factors across varying geometry

From the graph it can be seen that the primary equilibrium path traces the two bars snapping

through, with $w = 1.0$ signifying the bars in a perfectly horizontal position and $w = 2.0$ corresponding to a fully snapped through system. Both non-linear critical points are highlighted on the equilibrium path with their connection to the vanishing stiffness determinant at these points explicitly drawn.

As previously discussed, inaccuracies of the LPB analysis are attributable to the consequential pre-critical deflections of the structure. The sensitivity of LPB analysis to this effect can be investigated by gradually increasing the height b of the structure while retaining the same span a . As the height over span ratio b/a increases, the structure will become stiffer and pre-critical vertical displacements will reduce. As the actual structural behaviour gradually aligns with the underpinning LPB assumptions, the LPB results should become more accurate. Across the range of $1 \leq b/a \leq 5$, the ratios of critical LPB load factors to the lowest non-linear critical load factors have been plotted in the following figure. Although it is acknowledged that as b/a increase the instability mode will change from a limit to bifurcation point, the detail of interest here is the load factor associated with the onset of instability, regardless of its nature.

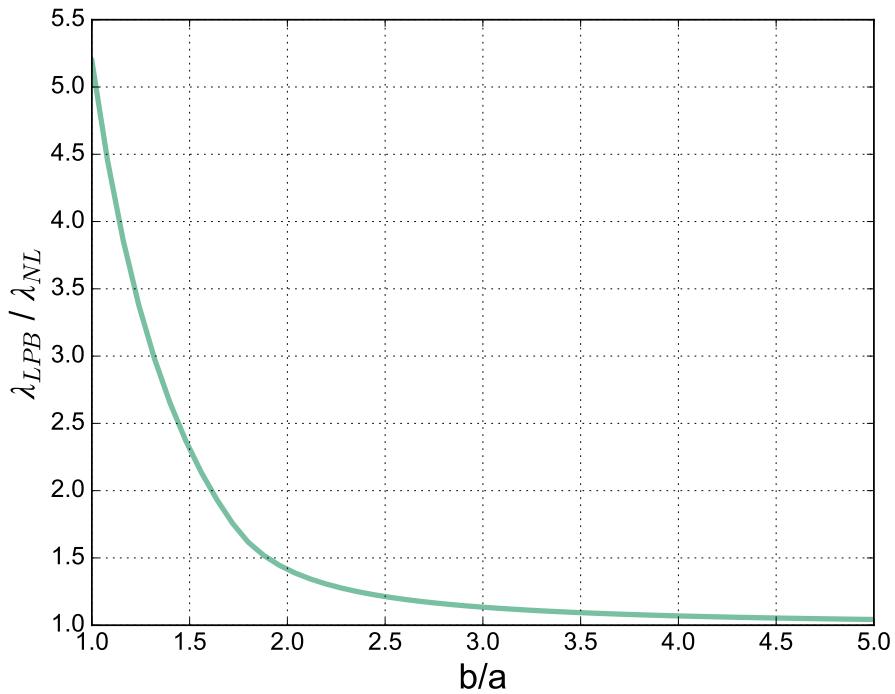


Figure 26 Ratio of Mises truss LPB and NL critical load factors across varying geometry

Consistent with the previous rationale, the LPB analyses tend toward more accurate results as the structure becomes stiffer at greater b/a and pre-critical displacements approach negligible values. Despite this, the figure also confirms that LPB analyses continue to typically overestimate the critical load for a structure. Summarising, it can be seen that when the LPB assumptions are nearly fulfilled it produces quite accurate results with a fraction of computational effort compared to a full non-linear stability analysis, although the tendency to overestimate the onset of instability should also be considered.

4.5. Co-rotational transformation fundamentals - STILL TO DO!!!!

asdfasdf

4.6. Co-rotational transformation geometric stiffness - STILL TO
DO!!!!

asdfasdf

Chapter 5 DSG triangle shell element

The follow sequence elucidates the formulation and implementation of the stiffness matrix, mass matrix and quantity recovery associated with the DSG triangle shell element in Kratos.

5.1. Stiffness matrix formulation

Based on the 5 parameter Reissner-Mindlin shell theory, the thick shell considers internal energy contributions from membrane, bending and shear components. As discussed in Chapter 2, basic finite elements derived from this shell theory face locking problems as the shell slenderness ratio increases. The element implemented is Bletzinger's Discrete Shear Gap (DSG) shell [12] which incorporates an enhanced shear strain formulation to mitigate the aforementioned locking. This triangular element has 18 DOFs ordered as such:

$$\mathbf{u}^T = \begin{pmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 \end{pmatrix} \quad \text{where} \quad \mathbf{u}_i^T = \begin{pmatrix} u_{xi} & u_{yi} & u_{zi} & \beta_{xi} & \beta_{yi} & \beta_{zi} \end{pmatrix} \quad (5.1)$$

The element displacement field is related to the discrete nodal values via shape functions.

$$\mathbf{u}(x, y) = \sum_{i=1}^3 N_i(x, y) \mathbf{u}_i \quad (5.2)$$

N_i are the standard linear triangle shape functions, referred to the Cartesian system, considering the corner points of the triangle x_i, y_i .

$$\begin{aligned} N_1(x, y) &= \frac{1}{2A} [(x_2y_3 - x_3y_2) + x(y_2 - y_3) + y(x_3 - x_2)] \\ N_2(x, y) &= \frac{1}{2A} [(x_3y_1 - x_1y_3) + x(y_3 - y_1) + y(x_1 - x_3)] \\ N_3(x, y) &= \frac{1}{2A} [(x_1y_2 - x_2y_1) + x(y_1 - y_2) + y(x_2 - x_1)] \end{aligned} \quad (5.3)$$

Analogous to internal energy, the element stiffness matrix of the DSG triangle can be decomposed into membrane, bending and shear contributions.

$$\mathbf{K} = \mathbf{K}_{mem} + \mathbf{K}_{bend} + \mathbf{K}_{shear} \quad (5.4)$$

The above expression can be expanded into strain-displacement and material matrices relevant for each component.

$$\mathbf{K} = \int_A (\mathbf{B}_{mem}^T \mathbf{C}_{mem} \mathbf{B}_{mem} + \mathbf{B}_{bend}^T \mathbf{C}_{bend} \mathbf{B}_{bend} + \mathbf{B}_{shear}^T \mathbf{C}_{shear} \mathbf{B}_{shear}) dA \quad (5.5)$$

Rama et al. [41] present the DSG formulation in a similar manner, detailing the strain displacement matrix and material material of each constituent separately.

The membrane strain displacement matrix requires no enhancement since linear triangle elements are impervious to membrane locking as discussed in section 2.3.2. Thus, the standard displacement-based formulation can be confidently utilised:

5.2. ADD IN THEORY OF MEMBRANE DERIVATION HERE !!!!

$$\mathbf{B}_{mem} = \begin{pmatrix} \mathbf{B}_{mem_1} & \mathbf{B}_{mem_2} & \mathbf{B}_{mem_3} \end{pmatrix} \quad (5.6)$$

$$\mathbf{B}_{mem_i} = \begin{pmatrix} N_{i,x} & 0 & 0 & 0 & 0 & 0 \\ 0 & N_{i,y} & 0 & 0 & 0 & 0 \\ N_{i,y} & N_{i,x} & 0 & 0 & 0 & 0 \end{pmatrix} \quad (5.7)$$

The bending strain displacement matrix is also a simple standard displacement-based formulation and can be presented in a similar manner:

5.3. ADD IN THEORY OF MEMBRANE DERIVATION HERE !!!!

$$\mathbf{B}_{bend} = \begin{pmatrix} \mathbf{B}_{bend_1} & \mathbf{B}_{bend_2} & \mathbf{B}_{bend_3} \end{pmatrix} \quad (5.8)$$

$$\mathbf{B}_{bend_i} = \begin{pmatrix} 0 & 0 & 0 & 0 & N_{i,x} & 0 \\ 0 & 0 & 0 & -N_{i,y} & 0 & 0 \\ 0 & 0 & 0 & -N_{i,x} & N_{i,y} & 0 \end{pmatrix} \quad (5.9)$$

Finally, the shear strain displacement matrix, which implements the DSG element enhancement technology fully derived in Appendix B and mitigates shear locking, is expressed as follows:

$$\mathbf{B}_{shear} = \frac{1}{2A} \begin{pmatrix} 0 & 0 & b-c & 0 & A & 0 & 0 & 0 & c & \frac{-bc}{2} & \frac{ac}{2} & 0 & 0 & 0 & -b & \frac{bc}{2} & \frac{bd}{2} & 0 \\ 0 & 0 & d-a & -A & 0 & 0 & 0 & 0 & -d & \frac{bd}{2} & \frac{-ad}{2} & 0 & 0 & 0 & a & \frac{-ac}{2} & \frac{ad}{2} & 0 \end{pmatrix}$$

with : $a = x_2 - x_1$, $b = y_2 - y_1$, $c = y_3 - y_1$, $d = x_3 - x_1$

(5.10)

The material matrices for the membrane and bending parts are presented below:

$$\mathbf{C}_{mem} = \frac{Eh}{(1-\nu^2)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{pmatrix} \quad (5.11)$$

$$\mathbf{C}_{bend} = \frac{Eh^3}{12(1-\nu^2)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{pmatrix} \quad (5.12)$$

To further improve the DSG element performance, Bischoff and Bletzinger [9] [10] applied the enhancement approach that Llyl suggested for MITC-4 elements [31]. This approach modifies the internal shear energy term by scaling the shear constitutive matrix with a correction term τ incorporating the element thickness and an indicator of element size (h_k = longest element side length). The enhanced shear constitutive matrix is thus:

$$\mathbf{C}_{shear} = \tau \kappa G h \begin{pmatrix} 1 & \nu \\ \nu & 1 \end{pmatrix} = \frac{\kappa G h^3}{h^2 + \alpha h_k^2} \begin{pmatrix} 1 & \nu \\ \nu & 1 \end{pmatrix} \quad (5.13)$$

where $\kappa = \frac{5}{6}$ is the shear correction factor and $\alpha = 0.1$ as per [31].

As described in section 2.3.1, transverse shear locking is driven by a mismatch of internal energy allocation between bending ($\Pi_{bend} \propto h^3$) and shear components ($\Pi_{shear} \propto h$) as $h \rightarrow 0$. This modification somewhat alleviates the locking by 'encouraging' the internal shear energy to scale with the cube of the thickness too, thus reducing the artificial energy disparity.

Although all stiffness components are assembled, one notices that lack of entries corresponding to the drilling DOF β_{zi} currently renders the element stiffness matrix singular. The technology of drilling DOFs discussed in 2.4.8 is thus introduced. Nguyen-Thoi et al. [37] proposed to remedy this rotational singularity by setting the drilling DOF entries to one one-thousandth of the maximum diagonal entry in the element stiffness matrix.

$$K_{\beta_{zi}} = \frac{\max(K_{ij}\delta_{ij})}{1000} \quad (5.14)$$

5.4. Stiffness matrix implementation

Despite the relatively simple decoupled stiffness formulation presented, the practical programming of it invariably introduces its own complexities. Furthermore, leveraging the existing functionality that the Kratos code possesses not only prevents re-inventing the wheel, but also makes the code more readable and functionally cohesive.

The new DSG triangle element is implemented in the files `shell_thick_element_3D3N.hpp` and `shell_thick_element_3D3N.cpp`, which are compiled into the 'StructuralMechanicsApplication' module of Kratos. Without extending into extraneous details, the DSG triangle element is derived from the Kratos `element` class and makes extensive use of other existing Kratos utility classes including those offering: coordinate transformations, material properties and pre-defined stiffness matrix and residual vector data types. Correspondingly, it is also subject to the constraints associated with each of these. From a high level view, however, the element stiffness matrix follows the subsequent workflow:

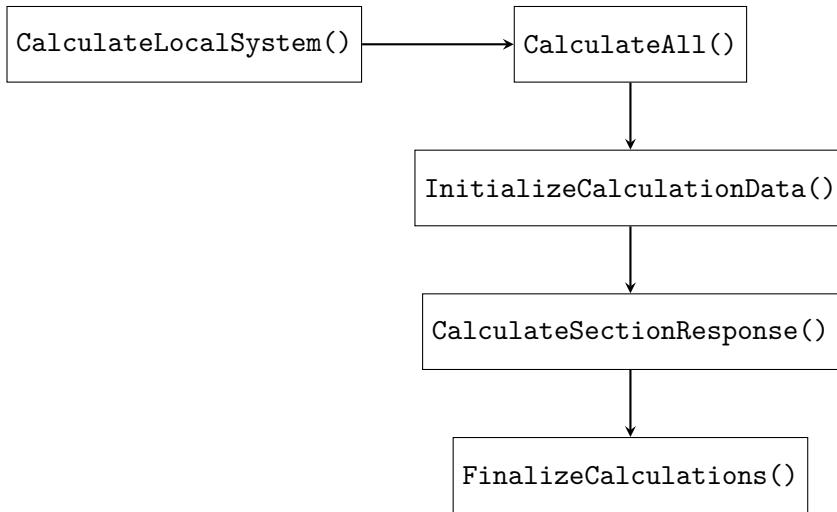


Figure 27 High level overview of DSG element workflow

Initially, the re-implemented virtual method `CalculateLocalSystem()` is called by the Kratos framework automatically for every `ShellThickElement3D3N` in the job definition. This method redirects to `CalculateAll()`, which is the main pipeline of the element stiffness calculation, itself calling three key methods: `InitializeCalculationData()`, `CalculateSectionResponse()` and `FinalizeCalculations()`.

Following the general form of the existing shell elements in Kratos, all the data which remains constant through the Gauss Integration loop is calculated beforehand in the function `InitializeCalculationData()`. The DSG element follows this tradition for consistency, although it isn't strictly necessary because it only requires one Gauss point for the numerical integration. Following `InitializeCalculationData()`, `CalculateSectionResponse()` is called and the material matrix is populated with existing Kratos material classes. It must be noted here that a single 8×8 material matrix \mathbf{C} is returned which is structured as follows (for the setting of 'thick' shell kinematics):

$$\mathbf{C}_{Kratos} = \begin{pmatrix} \mathbf{C}_{mem} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{bend} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_{shear} \end{pmatrix} \quad (5.15)$$

At this stage the shear component of the material matrix is currently unmodified, thus it is subsequently corrected with τ as per equation (5.13). The DOF arrangement of the material matrix also motivates a slight departure from the strain displacement matrices as presented above. Although the element stiffness matrix can certainly be programmed in its constitutive parts, as per equation (5.5), it is more concise to calculate it as follows:

$$\mathbf{K} = \int_A (\mathbf{B}_{comb}^T \mathbf{C}_{Kratos} \mathbf{B}_{comb}) dA = A \mathbf{B}_{comb}^T \mathbf{C}_{Kratos} \mathbf{B}_{comb} \quad (5.16)$$

A consequence of this arrangement is that the combined strain displacement matrix created in `InitializeCalculationData()` must conform to the DOF ordering of the material matrix layout.

The element stiffness matrix is calculated according to equation (5.16) and subsequently modified to include an artificial drilling DOF stiffness as per equation (5.14). Lastly, this is followed by a call to the Kratos function `FinalizeCalculations()` which handles the transformation from the element to the global orientation.

The following pseudocode summarises the key calls and operations involved in calculating the DSG element stiffness matrix.

Algorithm 1 DSG triangle element stiffness matrix pseudocode

Require: Coordinate transformation instance

- 1: **call** `CalculateAll()`
 - 2: Resize *LHS* and *RHS*
 - 3: **call** `InitializeCalculationData(data)`
 - 4: Calculate combined strain-displacement matrix *B*
 - 5: **call** `CalculateSectionResponse(data)`
 - 6: Retrieve material properties *C*
 - 7: Apply shear stabilization to material matrix *C*
 - 8: Calculate *LHS* stiffness matrix
 - 9: Add in artificial drilling stiffness
 - 10: Modify *RHS* residual vector
 - 11: **call** `FinalizeCalculations(data, displacements, LHS, RHS)`
 - 12: **call** `AddBodyForces(data, RHS)`
-

5.5. Mass matrix formulation and implementation

The specification of a mass matrix is necessary to facilitate dynamic analysis with the thick triangular shell element. Both lumped and consistent mass matrices are offered, with the lumped mass matrix used by default.

5.5.1. Lumped mass matrix

The default mass matrix employed is a lumped approach, which results in a diagonal mass matrix that ignores rotary inertia.

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_3 \end{pmatrix} \quad \text{where} \quad \mathbf{M}_i = \begin{pmatrix} \bar{m} & 0 & 0 & 0 & 0 & 0 \\ 0 & \bar{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & \bar{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (5.17)$$

The general lumped mass is determined for a multi-ply material with n plies each of h_i thickness and ρ_i density as follows:

$$\bar{m} = \frac{A}{3} \sum_{i=1}^n \rho_i h_i \quad (5.18)$$

For a single layer material of area A this reduces to:

$$\bar{m} = \frac{A}{3} \rho h \quad (5.19)$$

5.5.2. Consistent mass matrix

If the virtual work expression of a general shell element is expanded to include dynamics, the virtual kinetic energy δK , virtual internal strain energy δU and virtual external potential δV are related over a time interval T as such:

$$\int_0^T \delta K - (\delta U - \delta V) dt = 0 \quad (5.20)$$

Focussing solely on the kinetic energy, pertinent to the consistent mass matrix, clarifying motion in three dimensions yields:

$$\delta K = \int_V \rho(u_1 \delta u_1 + u_2 \delta u_2 + u_3 \delta u_3) dV \quad (5.21)$$

The volume integral can be split into area and thickness integrals and referred to mid-plane

velocity components (\dot{u}_0 , \dot{v}_0 , \dot{w}_0):

$$\delta K = \int_{\Omega} \int_{-\frac{h}{2}}^{\frac{h}{2}} \rho \left[(\dot{u}_0 + z\dot{\beta}_1)(\delta u_0 + z\delta\dot{\beta}_1) + (\dot{v}_0 + z\dot{\beta}_2)(\delta v_0 + z\delta\dot{\beta}_2) + (\dot{w}_0 + z\dot{\beta}_3)(\delta w_0 + z\delta\dot{\beta}_3) \right] dz d\Omega \quad (5.22)$$

The through-thickness integration can be simplified with the help of defining the following translational and rotational inertias under the assumption of constant density throughout the shell thickness:

$$I_i = \int_{-\frac{h}{2}}^{\frac{h}{2}} \rho z^i dz \quad (i = 0, 1, 2) \quad (5.23)$$

$$I_0 = \rho h, \quad I_1 = 0, \quad I_2 = \frac{\rho h^3}{12}$$

Substituting the inertias into equation 5.22 yields:

$$\delta K = \int_{\Omega} \left[I_0(\dot{u}_0 \delta u_0 + \dot{v}_0 \delta v_0 + \dot{w}_0 \delta w_0) + I_2(\dot{\beta}_1 \delta \dot{\beta}_1 + \dot{\beta}_2 \delta \dot{\beta}_2) \right] d\Omega \quad (5.24)$$

Analogous to the method of discretized displacement field, the cartesian velocity and acceleration field can be expressed in terms of shape functions and discrete nodal quantities for both translational and rotational dofs, ordered as per equation 5.1:

$$\hat{\mathbf{u}}(x, y) = \sum_{i=1}^3 N_i(x, y) \hat{\mathbf{u}}_i, \quad \ddot{\mathbf{u}}(x, y) = \sum_{i=1}^3 N_i(x, y) \hat{\ddot{\mathbf{u}}}_i, \quad (5.25)$$

Approached from this discretized perspective, equation 5.24 can be developed into:

$$\delta K = \mathbf{M}_C \hat{\mathbf{u}} \delta \hat{\mathbf{u}} = \int_{\Omega} \left[I_0 \mathbf{N}_T^T \mathbf{N}_T + I_2 \mathbf{N}_R^T \mathbf{N}_R \right] d\Omega \hat{\mathbf{u}} \delta \hat{\mathbf{u}} \quad (5.26)$$

The matrices \mathbf{N}_T and \mathbf{N}_R arrange nodal shape functions N_i to filter the appropriate translational and rotational DOFs respectively, defined as:

$$\mathbf{N}_T = \begin{pmatrix} \mathbf{N}_{T_1} & \mathbf{N}_{T_2} & \mathbf{N}_{T_3} \end{pmatrix}, \quad \mathbf{N}_R = \begin{pmatrix} \mathbf{N}_{R_1} & \mathbf{N}_{R_2} & \mathbf{N}_{R_3} \end{pmatrix} \quad (5.27)$$

$$\mathbf{N}_{T_i} = \begin{pmatrix} N_i & 0 & 0 & 0 & 0 & 0 \\ 0 & N_i & 0 & 0 & 0 & 0 \\ 0 & 0 & N_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \mathbf{N}_{R_i} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_i & 0 & 0 \\ 0 & 0 & 0 & 0 & N_i & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha N_i \end{pmatrix} \quad (5.28)$$

It is noted that the drilling DOF has an artificial mass assigned to it, scaled by α from the other rotational DOFs.

By explicitly stating the translational and rotational inertias, the final expression for the consistent mass matrix \mathbf{M}_C of a shell with constant thickness h is derived:

$$\mathbf{M}_C = h \int_{\Omega} \rho \left[\mathbf{N}_T^T \mathbf{N}_T + \frac{h^2}{12} \mathbf{N}_R^T \mathbf{N}_R \right] d\Omega \quad (5.29)$$

If the average shell density $\bar{\rho}$ is taken at the element centroid, the resulting consistent mass matrix of the DSG element can be explicitly evaluated as:

$$\mathbf{M}_C = \frac{\bar{\rho}hA}{12} \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{h^2}{6} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 \\ 0 & 0 & 0 & 0 & \frac{h^2}{6} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} \\ 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{6} & 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{6} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 \\ 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{6} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{6} & 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{6} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{h^2}{6} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{12} & 0 & 0 & 0 & 0 & 0 & \frac{\alpha^2 h^2}{6} \end{pmatrix} \quad (5.30)$$

5.6. Stress and strain recovery

While the stiffness and mass matrices enable the calculation of nodal displacements, velocities and accelerations, practical engineering analysis is usually more concerned with the strains and stresses of the structure. The non-zero local strains ($\epsilon_{zz} = 0$) of the 5 parameter element can be arranged in a vector form:

$$\boldsymbol{\epsilon}^T = \begin{pmatrix} \epsilon_1 & \epsilon_2 & \epsilon_3 \end{pmatrix} \quad \text{with} \quad \boldsymbol{\epsilon}_i^T = \begin{pmatrix} \epsilon_{xx} & \epsilon_{xx} & 2\epsilon_{xy} & \kappa_{xx} & \kappa_{yy} & 2\kappa_{xy} & 2\epsilon_{xz} & 2\epsilon_{yz} \end{pmatrix} \quad (5.31)$$

The strain field within the element can be recovered from the displacement field by using the strain displacement matrix, which is constant over the element for the DSG triangle.

$$\boldsymbol{\epsilon}(\xi, \eta) = \mathbf{B} \mathbf{u}(\xi, \eta) \quad (5.32)$$

In typical finite element programs the strains and stresses are calculated at the Gauss points of the element, which is also how Kratos operates. Since the DSG has one Gauss point in the centre of the element, the strain is recovered from the discrete nodal displacements $\hat{\mathbf{u}}_i$ as follows:

$$\boldsymbol{\epsilon}_{GP} = \mathbf{B} \sum_{i=1}^{3 \text{ nodes}} N_i(\xi_{GP}, \eta_{GP}) \hat{\mathbf{u}}_i \quad (5.33)$$

With the strains determined, the stresses at the Gauss points can be recovered with the material matrix at the Gauss point.

$$\boldsymbol{\sigma}_{GP} = \mathbf{C}_{GP} \boldsymbol{\epsilon}_{GP} \quad (5.34)$$

The general implementation of the stress and strain recovery described above is illustrated in the following pseudocode.

Algorithm 2 DSG triangle element stress and strain recovery

Require: *requestedQuantity*, calculation of nodal displacements

- 1: **call** InitializeCalculationData(*data*)
- 2: Calculate strain-displacement matrix *B*
- 3: Retrieve element *localDisplacements*
- 4: *generalizedStrains* = product(*B*, *localDisplacements*)
- 5: **if** *requestedQuantity* requires stress **then**
- 6: **call** CalculateSectionResponse(*data*)
- 7: *generalizedStresses* = product (*C*, *generalizedStrains*)
- 8: Decimal correction of *generalizedStresses*
- 9: **end if**
- 10: Decimal correction of *generalizedStrains*
- 11: **if** *requestedQuantity* requires local orientation **then**
- 12: Rotate *requestedQuantity* to local orientation
- 13: **end if**
- 14: Assemble *requestedQuantity* into *outputMatrix*
- 15: **if** *requestedQuantity* requires global orientation **then**
- 16: Rotate *outputMatrix* to global orientation
- 17: **end if**
- 18: Interpolate *outputMatrix* to standard Gauss points for visualisation

5.6.1. Von Mises equivalent stress

The Von Mises equivalent stress is a convenient method to translate a complex 3 dimensional stress state into an equivalent scalar value. As such, it's computation requires element stresses

to be recovered, the tensor of which is:

$$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{pmatrix} \quad (5.35)$$

Limiting the tensor to plane stress states, the deviatoric stress tensor $\boldsymbol{\sigma}'$ can be derived by subtracting the mean hydrostatic stress σ_H :

$$\boldsymbol{\sigma}' = \boldsymbol{\sigma} - \sigma_H \delta_{ij} = \frac{1}{3} \begin{pmatrix} 2\sigma_{xx} - \sigma_{yy} & 3\sigma_{xy} & 3\sigma_{xz} \\ 3\sigma_{xy} & 2\sigma_{yy} - \sigma_{xx} & 3\sigma_{yz} \\ 3\sigma_{xz} & 3\sigma_{yy} & -\sigma_{xx} - \sigma_{yy} \end{pmatrix}, \quad \sigma_H = \frac{\sigma_{xx} + \sigma_{yy} + (\sigma_{zz} = 0)}{3} \quad (5.36)$$

The Von Mises equivalent stress is a scaling of the second deviatoric stress invariant J_2 related by:

$$\sigma_{VM} = \sqrt{3J_2} = \sqrt{\frac{3}{2}\boldsymbol{\sigma}' : \boldsymbol{\sigma}'} \quad (5.37)$$

Expanding the above expression in conjunction with equation 5.36, the Von Mises equivalent stress for the 5 parameter shell model is simplified to:

$$\sigma_{VM} = \sqrt{\sigma_{xx}^2 - \sigma_{xx}\sigma_{yy} + \sigma_{yy}^2 + 3(\sigma_{xy}^2 + \sigma_{xz}^2 + \sigma_{yz}^2)} \quad (5.38)$$

Chapter 6 ANDES-DKQ quadrilateral shell element

The following chapter outlines the formulation and implementation of the stiffness matrix, mass matrix and quantity recovery for the ANDES-DKQ quadrilateral shell element in Kratos.

6.1. Stiffness matrix formulation

As per the element technology discussion in section 2.4, the ANDES-DKQ element uses different finite element technologies for the membrane and bending components, both of which are subsequently developed.

6.1.1. ANDES membrane formulation

The membrane formulation is responsible for providing the membrane stiffness of the element. Unlike the linear triangle DSG element, bilinear quadrilaterals are susceptible to membrane locking, and, as such, the ANDES membrane formulation was prudentially chosen as presented in [26]. A full description and theoretical derivation of the ANDES approach falls outside the scope of this document, however, those interested are directed to Militello's and Felippa's initial paper [33] on the formulation.

Only the membrane portion of the total shell element is considered in this section, in which there are three DOFs per node.

$$\mathbf{u}^T = \begin{pmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 & \mathbf{u}_4 \end{pmatrix} \quad \text{where} \quad \mathbf{u}_i^T = \begin{pmatrix} u_{xi} & u_{yi} & \beta_{zi} \end{pmatrix} \quad (6.1)$$

The ANDES membrane formulation, following the FF framework, is split into the basic stiffness related to the basic constant strain displacement matrix \mathbf{L} and the higher order stiffness related to the higher order deviatoric strain displacement matrix \mathbf{B}_d .

$$\mathbf{K}_{mem} = \mathbf{K}_b + \mathbf{K}_h = \int_A (\mathbf{L} + \mathbf{B}_d \mathbf{H}_K)^T \mathbf{C}_{mem} (\mathbf{L} + \mathbf{B}_d \mathbf{H}_K) dA \quad (6.2)$$

The basic strain displacement matrix \mathbf{L} and the higher order complement \mathbf{B}_d are now developed.

Membrane basic stiffness

The membrane basic stiffness is driven by assuming a constant stress field within the element and lumping this over side edges to consistent nodal forces.

$$\mathbf{f} = \mathbf{L}\boldsymbol{\sigma} \quad \text{where} \quad \boldsymbol{\sigma}^T = \begin{pmatrix} \sigma_{xx} & \sigma_{xx} & \tau_{xy} \end{pmatrix} \quad (6.3)$$

The structure of the above expression is resolved as such:

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_1 & \mathbf{L}_2 & \mathbf{L}_3 & \mathbf{L}_4 \end{pmatrix} \quad \text{and} \quad \mathbf{f} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \\ \mathbf{f}_4 \end{pmatrix} \quad \text{where} \quad \mathbf{f}_i = \begin{pmatrix} f_{xi} \\ f_{yi} \\ m_{zi} \end{pmatrix} \quad (6.4)$$

where each nodal entry ' j ' of the lumping matrix \mathbf{L} is constructed with the following cyclic permutation (i, j, k, l) for the four nodes (1, 2, 3, 4):

$$\mathbf{L}_j = \frac{1}{2A} \begin{pmatrix} y_{ki} & 0 & -x_{ki} \\ 0 & -x_{ki} & y_{ki} \\ \frac{\alpha}{6}(y_{ij}^2 - y_{kj}^2) & \frac{\alpha}{6}(x_{ij}^2 - x_{kj}^2) & \frac{\alpha}{3}(x_{kj}y_{kj} - x_{ij}y_{ij}) \end{pmatrix} \quad (6.5)$$

Throughout this formulation the notation of $x_{ij} = x_i - x_j$ and $y_{ij} = y_i - y_j$ holds. Furthermore, the value of α is taken as 1.5 [33].

Membrane higher order stiffness

The membrane higher order stiffness considers a set of higher order DOFs expressed in terms of the visible DOFs. To improve readability and allow easier code checking, the visible membrane DOFs for the membrane higher order stiffness are arranged as per Haugen's

original formulation [26] (denoted \mathbf{u}_H) in a component-wise fashion:

$$\mathbf{u}_H = \begin{pmatrix} \mathbf{u}_x \\ \mathbf{u}_y \\ \beta_z \end{pmatrix} \quad \text{where} \quad \mathbf{u}_x = \begin{pmatrix} u_{x1} \\ u_{x2} \\ u_{x3} \\ u_{x4} \end{pmatrix}, \quad \mathbf{u}_y = \begin{pmatrix} u_{y1} \\ u_{y2} \\ u_{y3} \\ u_{y4} \end{pmatrix}, \quad \beta_z = \begin{pmatrix} \beta_{z1} \\ \beta_{z2} \\ \beta_{z3} \\ \beta_{z4} \end{pmatrix} \quad (6.6)$$

The higher order rotational DOFs are related to the visible DOFs as described below:

$$\boldsymbol{\theta}_h = \mathbf{H}_{\theta u} \mathbf{u}_H \quad \text{where} \quad \boldsymbol{\theta}_h^T = \left(\theta'_1 \quad \theta'_2 \quad \theta'_3 \quad \theta'_4 \quad \bar{\theta} \right) \quad (6.7)$$

with

$$\mathbf{H}_{\theta u} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{3}{4} & \frac{-1}{4} & \frac{-1}{4} & \frac{-1}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{4} & \frac{3}{4} & \frac{-1}{4} & \frac{-1}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{4} & \frac{-1}{4} & \frac{3}{4} & \frac{-1}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{4} & \frac{-1}{4} & \frac{-1}{4} & \frac{3}{4} \\ \frac{x_{42}}{f} & \frac{x_{13}}{f} & \frac{x_{24}}{f} & \frac{x_{31}}{f} & \frac{y_{42}}{f} & \frac{y_{13}}{f} & \frac{x_{24}}{f} & \frac{y_{31}}{f} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}$$

where $f = 16|\mathbf{J}|$

and $|\mathbf{J}| = \frac{1}{8}[(x_1y_2 - x_2y_1) + (x_2y_3 - x_3y_2) + (x_3y_4 - x_4y_3) + (x_4y_1 - x_1y_4)]$

(6.8)

The higher order translational DOFs are related to the visible DOFs as described below:

$$\tilde{\mathbf{u}}_t = \mathbf{H}_{tu} \mathbf{u}_H \quad \text{where} \quad \tilde{\mathbf{u}}_t^T = \left(\tilde{u}_x \quad \tilde{u}_y \right) \quad (6.9)$$

The translational mapping matrix \mathbf{H}_{tu} was originally prescribed by Haugen to be:

$$\mathbf{H}_{tu} = \begin{pmatrix} s_{\xi_x} & -s_{\xi_x} & s_{\xi_x} & -s_{\xi_x} & s_{\xi_y} & -s_{\xi_y} & s_{\xi_y} & -s_{\xi_y} & 0 & 0 & 0 & 0 \\ s_{\eta_x} & -s_{\eta_x} & s_{\eta_x} & -s_{\eta_x} & s_{\eta_y} & -s_{\eta_y} & s_{\eta_y} & -s_{\eta_y} & 0 & 0 & 0 & 0 \end{pmatrix} \quad (6.10)$$

Where s_ξ and s_η are the Cartesian unit vectors in the ξ and η directions respectively.

However, it is noted after consulting Felippa's work on supernatural quadrilateral elements [24], that the above mapping matrix is limited to strictly structured rectangular elements. To extend element functionality to unstructured meshes the general translational mapping matrix from

[24] is utilised and incorporated into Haugen's original approach in equation 6.10, as outlined below:

$$\mathbf{H}_{tu} = \begin{pmatrix} H_1 s_{\xi_x} & H_2 s_{\xi_x} & H_3 s_{\xi_x} & H_4 s_{\xi_x} & H_1 s_{\xi_y} & H_2 s_{\xi_y} & H_3 s_{\xi_y} & H_4 s_{\xi_y} & 0 & 0 & 0 & 0 \\ H_1 s_{\eta_x} & H_2 s_{\eta_x} & H_3 s_{\eta_x} & H_4 s_{\eta_x} & H_1 s_{\eta_y} & H_2 s_{\eta_y} & H_3 s_{\eta_y} & H_4 s_{\eta_y} & 0 & 0 & 0 & 0 \end{pmatrix} \quad (6.11)$$

The coefficients H_i are determined with the following geometric calculations [24]:

$$\begin{pmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \end{pmatrix} = \begin{pmatrix} \frac{A_0 + A_1 + A_2}{2A_0} \\ \frac{-A_0 + A_1 - A_2}{2A_0} \\ \frac{A_0 - A_1 - A_2}{2A_0} \\ \frac{-A_0 - A_1 + A_2}{2A_0} \end{pmatrix} \quad \text{with} \quad \begin{pmatrix} A_0 = A \\ A_1 = \frac{x_{34}y_{12} - x_{12}y_{34}}{2} \\ A_2 = \frac{x_{23}y_{14} - x_{14}y_{23}}{2} \end{pmatrix} \quad (6.12)$$

Combining both mapping matrices together expresses all higher order DOFs in terms of the visible DOFs:

$$\tilde{\mathbf{u}} = \mathbf{H}\mathbf{u}_H \quad \text{where} \quad \mathbf{H} = \begin{pmatrix} \mathbf{H}_{\theta u} \\ \mathbf{H}_{ut} \end{pmatrix} \quad \text{and} \quad \tilde{\mathbf{u}}^T = \left(\theta'_1 \quad \theta'_2 \quad \theta'_3 \quad \theta'_4 \quad \bar{\theta} \quad \tilde{u}_x \quad \tilde{u}_y \right) \quad (6.13)$$

As discussed prior to equation 6.6 the higher order membrane stiffness thus far is relative to Haugen's DOF ordering \mathbf{u}_H which does not coincide with Kratos ordering $\mathbf{u}_K = \mathbf{u}$. Since the combined mapping matrix \mathbf{H} is the only link between the visible and higher order DOFs, it is possible to "re-route" it via an additional filter operation to the Kratos DOF arrangement. The Kratos and Haugen DOF ordering can be related with a filter matrix \mathbf{Z} :

$$\mathbf{u}_K = \mathbf{Z}\mathbf{u}_H \quad (6.14)$$

$$\begin{pmatrix} u_{x1} \\ u_{y1} \\ \beta_{z1} \\ u_{x2} \\ u_{y2} \\ \beta_{z2} \\ u_{x3} \\ u_{y3} \\ \beta_{z3} \\ u_{x4} \\ u_{y4} \\ \beta_{z4} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_{x1} \\ u_{x2} \\ u_{x3} \\ u_{x4} \\ u_{y1} \\ u_{y2} \\ u_{y3} \\ u_{y4} \\ \beta_{z1} \\ \beta_{z2} \\ \beta_{z3} \\ \beta_{z4} \end{pmatrix} \quad (6.15)$$

Hence, the higher order DOFs can be expressed in terms of the Kratos DOFs, with \mathbf{H}_K denoting the transformed filter matrix:

$$\tilde{\mathbf{u}} = \mathbf{H}\mathbf{Z}\mathbf{u}_H = \mathbf{H}_K\mathbf{u}_K = \mathbf{H}_K\mathbf{u} \quad (6.16)$$

The descriptions for equations from (6.19) to (6.34) are condensed from the original element derivation [26] for brevity. The overarching idea of these equations is to relate the higher order nodal strain gauge readings to Cartesian strain displacement matrices \mathbf{B}_{hi} .

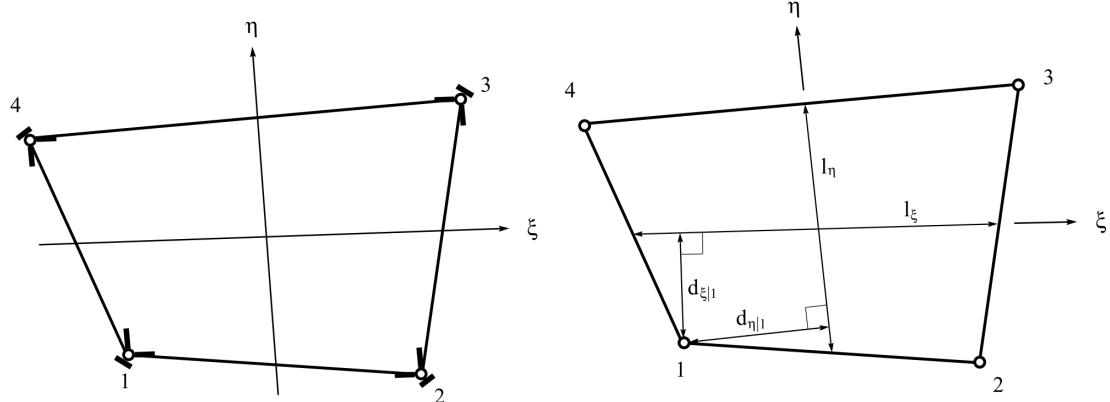


Figure 28 ANDES membrane nodal strain gauges [26]

Figure 29 Geometric dimensions of the quadrilateral element [26]

The strain gauges are placed as per figure 28 and read strains along the ξ and η axes as well as the element diagonals. At each strain gauge, the readings are thus:

$$\boldsymbol{\epsilon}_1 = \boldsymbol{\epsilon}_3 = \begin{pmatrix} \epsilon_\xi \\ \epsilon_\eta \\ \epsilon_{24} \end{pmatrix}, \quad \boldsymbol{\epsilon}_2 = \boldsymbol{\epsilon}_4 = \begin{pmatrix} \epsilon_\xi \\ \epsilon_\eta \\ \epsilon_{13} \end{pmatrix} \quad (6.17)$$

The strain readings are related to the higher order degrees of freedom via the nodal strain templates \mathbf{Q}_i .

$$\boldsymbol{\epsilon}_i = \mathbf{Q}_i \tilde{\mathbf{v}} \quad (6.18)$$

Higher order bending field

As per Haugen's PhD thesis, the primary strain mode the field is trying to match is pure bending of an element in an arbitrary direction. The bending strain field is associated with the higher order DOFs of θ'_i, \tilde{v}_ξ and \tilde{v}_η , and is intuited to be proportional to the distance from the neutral axis of the direction considered. For instance, bending in the ξ direction is assumed proportional to the distance d_ξ (refer figure 29) and the curvature of the element $\frac{\Delta\theta}{l_\xi}$. Thus the general form of the ξ strains are $\frac{d_{\xi|i}}{l_\xi}$, with the η strains derived in a similar manner. The bending strains at each node are:

$$\chi_{\xi|i} = \frac{d_{\xi|i}}{l_\xi}, \quad \chi_{\eta|i} = \frac{d_{\eta|i}}{l_\eta}, \quad (6.19)$$

where the quantities in figure 29 are calculated as follows:

$$d_{\xi|i} = \sqrt{(\mathbf{r}_i \times \mathbf{s}_\xi) \cdot (\mathbf{r}_i \times \mathbf{s}_\xi)}, \quad l_\xi = \sqrt{\mathbf{r}_\xi \cdot \mathbf{r}_\xi}, \quad \mathbf{r}_\xi = \frac{1}{2}(\mathbf{r}_2 + \mathbf{r}_3 - \mathbf{r}_1 - \mathbf{r}_4),$$

$$d_{\eta|i} = \sqrt{(\mathbf{r}_i \times \mathbf{s}_\eta) \cdot (\mathbf{r}_i \times \mathbf{s}_\eta)}, \quad l_\eta = \sqrt{\mathbf{r}_\eta \cdot \mathbf{r}_\eta}, \quad \mathbf{r}_\eta = \frac{1}{2}(\mathbf{r}_2 + \mathbf{r}_3 - \mathbf{r}_1 - \mathbf{r}_4),$$

\mathbf{s}_ξ and \mathbf{s}_η are the normalized parametric base vectors in cartesian coordinates, while \mathbf{r}_i are the nodal position vectors in cartesian coordinates.

The diagonal strain components of the higher order bending field are also assumed as to be proportional to the distance and curvature along the diagonal, thus yielding:

$$\chi_{24} = \frac{d_{24}}{2l_{24}}, \quad \chi_{13} = \frac{d_{13}}{2l_{13}}, \quad (6.20)$$

where

$$d_{24} = d_{13} = \sqrt{(\mathbf{r}_{31} \times \mathbf{e}_{24}) \cdot (\mathbf{r}_{31} \times \mathbf{e}_{24})} \quad l_{24} = \sqrt{\mathbf{r}_{24} \cdot \mathbf{r}_{24}}, \quad \mathbf{r}_{24} = \mathbf{r}_2 - \mathbf{r}_4$$

$$l_{13} = \sqrt{\mathbf{r}_{13} \cdot \mathbf{r}_{13}}, \quad \mathbf{r}_{13} = \mathbf{r}_1 - \mathbf{r}_3 \quad \mathbf{e}_{24} = \frac{\mathbf{r}_{24}}{l_{24}}$$

Higher order torsional field

The higher order torsional field is associated with the $\bar{\theta}$ higher order DOF. Figure 30 illustrates the general torsional displacement mode shape, indicating that no shear strains are produced in this mode.

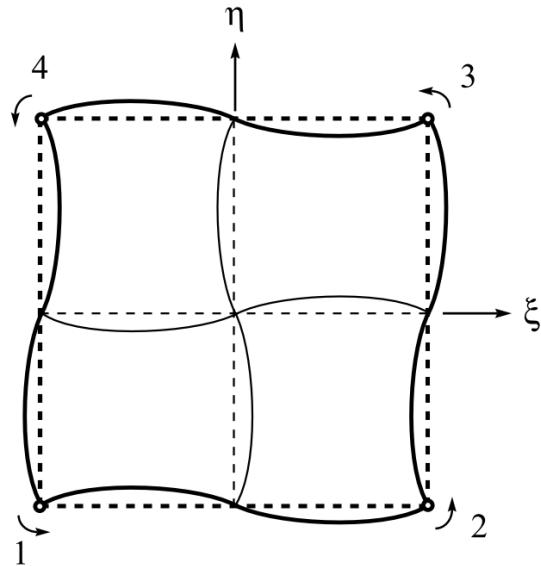


Figure 30 Higher order torsional mode of ANDES membrane formulation [26]

Under this torsional regime, ϵ_ξ is positive in the 1st and 3rd quadrants and negative in the 2nd and 4th quadrants, while ϵ_η has the opposite signs. Furthermore, with a unit rotation at each node the maximum displacement v_ξ will be proportional to l_η . Recalling that ϵ_ξ is the gradient of v_ξ in the ξ direction, one can suppose that ϵ_ξ is proportional to $\frac{1}{l_\xi}$. Considering this approach for ϵ_ξ and ϵ_η , the following torsional strain field terms are determined:

$$\chi_{\xi t} = \frac{l_\eta}{l_\xi}, \quad \chi_{\eta t} = \frac{l_\xi}{l_\eta}, \quad (6.21)$$

Higher order nodal strain templates

Considering the higher order bending and torsional components just outlined, the nodal strain

gauge readings can be described as follows:

$$\mathbf{Q}_1 = \begin{pmatrix} \rho_1 \chi_{\xi|1} & \rho_2 \chi_{\xi|1} & \rho_3 \chi_{\xi|1} & \rho_4 \chi_{\xi|1} & \alpha \chi_{\xi t} & -\beta_1 \frac{\chi_{\xi|1}}{\bar{\chi}_{\xi} l_{\xi}} & 0 \\ -\rho_1 \chi_{\eta|1} & -\rho_4 \chi_{\eta|1} & -\rho_3 \chi_{\eta|1} & -\rho_2 \chi_{\eta|1} & -\alpha \chi_{\eta t} & 0 & -\beta_1 \frac{\chi_{\eta|1}}{\bar{\chi}_{\eta} l_{\eta}} \\ \rho_5 \chi_{24} & \rho_6 \chi_{24} & \rho_7 \chi_{24} & \rho_8 \chi_{24} & 0 & \beta_2 \frac{c_{24\xi}}{l_{24}} & -\beta_2 \frac{c_{24\eta}}{l_{24}} \end{pmatrix} \quad (6.22)$$

$$\mathbf{Q}_2 = \begin{pmatrix} -\rho_2 \chi_{\xi|2} & -\rho_1 \chi_{\xi|2} & -\rho_4 \chi_{\xi|2} & -\rho_3 \chi_{\xi|2} & -\alpha \chi_{\xi t} & -\beta_1 \frac{\chi_{\xi|2}}{\bar{\chi}_{\xi} l_{\xi}} & 0 \\ \rho_4 \chi_{\eta|2} & \rho_1 \chi_{\eta|2} & \rho_2 \chi_{\eta|2} & \rho_3 \chi_{\eta|2} & \alpha \chi_{\eta t} & 0 & \beta_1 \frac{\chi_{\eta|2}}{\bar{\chi}_{\eta} l_{\eta}} \\ \rho_8 \chi_{13} & \rho_5 \chi_{13} & \rho_6 \chi_{13} & \rho_7 \chi_{13} & 0 & -\beta_2 \frac{c_{13\xi}}{l_{13}} & \beta_2 \frac{c_{13\eta}}{l_{13}} \end{pmatrix} \quad (6.23)$$

$$\mathbf{Q}_3 = \begin{pmatrix} \rho_3 \chi_{\xi|3} & \rho_4 \chi_{\xi|3} & \rho_1 \chi_{\xi|3} & \rho_2 \chi_{\xi|3} & \alpha \chi_{\xi t} & \beta_1 \frac{\chi_{\xi|3}}{\bar{\chi}_{\xi} l_{\xi}} & 0 \\ -\rho_3 \chi_{\eta|3} & -\rho_2 \chi_{\eta|3} & -\rho_1 \chi_{\eta|3} & -\rho_4 \chi_{\eta|3} & -\alpha \chi_{\eta t} & 0 & \beta_1 \frac{\chi_{\eta|3}}{\bar{\chi}_{\eta} l_{\eta}} \\ \rho_7 \chi_{13} & \rho_8 \chi_{13} & \rho_5 \chi_{13} & \rho_6 \chi_{213} & 0 & -\beta_2 \frac{c_{13\xi}}{l_{13}} & \beta_2 \frac{c_{13\eta}}{l_{13}} \end{pmatrix} \quad (6.24)$$

$$\mathbf{Q}_4 = \begin{pmatrix} -\rho_4 \chi_{\xi|4} & -\rho_3 \chi_{\xi|4} & -\rho_2 \chi_{\xi|4} & -\rho_1 \chi_{\xi|4} & -\alpha \chi_{\xi t} & \beta_1 \frac{\chi_{\xi|4}}{\bar{\chi}_{\xi} l_{\xi}} & 0 \\ \rho_2 \chi_{\eta|4} & \rho_3 \chi_{\eta|4} & \rho_4 \chi_{\eta|4} & \rho_1 \chi_{\eta|4} & \alpha \chi_{\eta t} & 0 & -\beta_1 \frac{\chi_{\eta|4}}{\bar{\chi}_{\eta} l_{\eta}} \\ \rho_6 \chi_{13} & \rho_7 \chi_{13} & \rho_8 \chi_{13} & \rho_5 \chi_{13} & 0 & \beta_2 \frac{c_{13\xi}}{l_{13}} & -\beta_2 \frac{c_{13\eta}}{l_{13}} \end{pmatrix} \quad (6.25)$$

where

$$c_{13\xi} = \mathbf{s}_{13}^T \mathbf{s}_{\xi}, \quad c_{13\eta} = \mathbf{s}_{13}^T \mathbf{s}_{\eta}, \quad c_{24\xi} = \mathbf{s}_{24}^T \mathbf{s}_{\xi}, \quad c_{24\eta} = \mathbf{s}_{24}^T \mathbf{s}_{\eta}$$

An optimisation of element performance has suggested the following coefficients to be used in the nodal strain gauge templates [26].

$$\begin{aligned} \rho_1 &= 0.1, & \rho_2 &= -0.1, & \rho_3 &= -0.1, & \rho_4 &= 0.1, & \rho_5 &= 0.0, \\ \rho_6 &= 0.5, & \rho_7 &= 0.0, & \rho_8 &= -0.5, & \beta_1 &= 0.6, & \beta_2 &= 0.0 \end{aligned} \quad (6.26)$$

Cartesian higher order strain displacement matrix

The Cartesian strain displacement matrices at the nodes are related to the mapping matrices

\mathbf{Q}_i as described below:

$$\mathbf{B}_{h1} = \mathbf{T}_{13}\mathbf{Q}_1, \quad \mathbf{B}_{h3} = \mathbf{T}_{13}\mathbf{Q}_3$$

where $\mathbf{T}_{13}^{-1} = \begin{pmatrix} s_{\xi x}^2 & s_{\xi y}^2 & s_{\xi x}s_{\xi y} \\ s_{\eta x}^2 & s_{\eta y}^2 & s_{\eta x}s_{\eta y} \\ s_{24x}^2 & s_{24y}^2 & s_{24x}s_{24y} \end{pmatrix}$

(6.27)

$$\mathbf{B}_{h2} = \mathbf{T}_{24}\mathbf{Q}_2, \quad \mathbf{B}_{h4} = \mathbf{T}_{24}\mathbf{Q}_4$$

where $\mathbf{T}_{24}^{-1} = \begin{pmatrix} s_{\xi x}^2 & s_{\xi y}^2 & s_{\xi x}s_{\xi y} \\ s_{\eta x}^2 & s_{\eta y}^2 & s_{\eta x}s_{\eta y} \\ s_{13x}^2 & s_{13y}^2 & s_{13x}s_{13y} \end{pmatrix}$

(6.28)

The total higher order membrane B matrix \mathbf{B}_h is constructed from the interpolation of the nodal \mathbf{B}_{hi} matrices with standard bi-linear shape functions.

$$\mathbf{B}_h(\xi, \eta) = (1 - \xi)(1 - \eta)\mathbf{B}_{h1} + (1 + \xi)(1 - \eta)\mathbf{B}_{h2} + (1 + \xi)(1 + \eta)\mathbf{B}_{h3} + (1 - \xi)(1 + \eta)\mathbf{B}_{h4}$$
(6.29)

A requirement of the underlying FF is energy orthogonality between the basic and higher order strain fields, which is not yet fulfilled. This orthogonality can be achieved by rendering the higher order field deviatoric, as per the formulation name, which involves subtracting the mean integral:

$$\mathbf{B}_d(\xi, \eta) = \mathbf{B}_h(\xi, \eta) - \bar{\mathbf{B}}_h \quad \text{with} \quad \bar{\mathbf{B}}_h = \int_A \mathbf{B}_h(\xi, \eta) dA$$
(6.30)

6.1.2. DKQ bending formulation

The bending formulation is responsible for providing the bending stiffness of the element, with an enhanced formulation selected to pre-empt transverse shear locking. The bending formulation chosen was the Discrete Kirchhoff Quadrilateral (DKQ) formulation originally presented by Batoz [5], presented in a most readable fashion in the PhD dissertation of Barrales [2]. A full description and theoretical derivation of the DKQ approach falls outside the scope of this document, refer [5]. As described in section 2.4.6, the transverse shear strain energy is neglected which prohibits element performance deterioration as the ratio $\frac{l}{t}$ encroaches into thin and very thin plate territories.

Only the bending portion of the total shell element is considered in this section, in which there are three nodal DOFs per node (w_i corresponds to u_{zi} in the figure below).

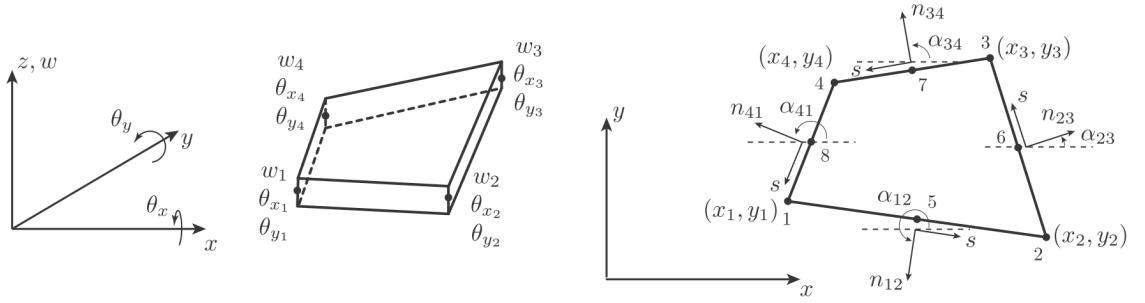


Figure 31 DKQ DOF arrangement and geometry [2]

$$\mathbf{u}^T = \begin{pmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 & \mathbf{u}_4 \end{pmatrix} \quad \text{where} \quad \mathbf{u}_i^T = \begin{pmatrix} u_{zi} & \theta_{xi} & \theta_{yi} \end{pmatrix} \quad (6.31)$$

The nodal rotational interpolation employed is as per the 8 node serendipity quad element:

$$\begin{pmatrix} \beta_x(\xi, \eta) \\ \beta_y(\xi, \eta) \end{pmatrix} = \sum_{i=1}^8 \psi_i(\xi, \eta) \begin{pmatrix} \beta_{xi}(\xi, \eta) \\ \beta_{yi}(\xi, \eta) \end{pmatrix} \quad (6.32)$$

where ψ_i are the standard 8 node serendipity shape functions described by Zienkiewicz [50]:

$$\begin{aligned} \psi_i(\xi, \eta) &= \frac{-1}{4}(1 + \xi_i \xi)(1 + \eta_i \eta)(1 - \xi_i \xi - \eta_i \eta) & i = 1, 2, 3, 4 \\ \psi_i(\xi, \eta) &= \frac{1}{2}(1 - \xi^2)(1 + \eta_i \eta) & i = 5, 7 \\ \psi_i(\xi, \eta) &= \frac{1}{2}(1 + \xi_i \xi)(1 - \eta^2) & i = 6, 8 \end{aligned} \quad (6.33)$$

and ξ_i and η_i are the natural coordinates of the 8 node serendipity element described in figure 31.

The following derivation from equations (6.34) to (6.39) is summarised from that of Barrales [2]. The general idea is the construction of a mapping from the standard 12 DOFs at each node to $\beta_x(\xi, \eta)$ and $\beta_y(\xi, \eta)$ across the element, the derivatives of which are curvatures as expressed in equation (6.39).

The following quantities are required components for the mapping:

$$L_{ij} = \sqrt{x_{ij}^2 + y_{ij}^2}, \quad x_{ij} = x_i - x_j, \quad y_{ij} = y_i - y_j \quad (6.34)$$

$$\begin{aligned} a_k &= \frac{-x_{ij}}{L_{ij}^2}, & b_k &= \frac{3}{4} \frac{x_{ij}y_{ij}}{L_{ij}^2}, \\ c_k &= \frac{\frac{1}{4}x_{ij}^2 - \frac{1}{2}y_{ij}^2}{L_{ij}^2}, & d_k &= \frac{-y_{ij}}{L_{ij}^2}, & e_k &= \frac{\frac{-1}{2}x_{ij}^2 + \frac{1}{4}y_{ij}^2}{L_{ij}^2} \end{aligned} \quad (6.35)$$

The elements of the mapping matrix are arranged as such:

$$\boldsymbol{\Psi}^x = \begin{pmatrix} \Psi_1^x \\ \vdots \\ \Psi_{12}^x \end{pmatrix}, \quad \boldsymbol{\Psi}^y = \begin{pmatrix} \Psi_1^y \\ \vdots \\ \Psi_{12}^y \end{pmatrix} \quad (6.36)$$

where the vectors entries are calculated as per the following scheme:

$$\begin{aligned} \Psi_{3(i-1)+1}^x(\xi, \eta) &= \frac{3}{2}(a_r\psi_r(\xi, \eta) - a_s\psi_s(\xi, \eta)) \\ \Psi_{3(i-1)+2}^x(\xi, \eta) &= b_r\psi_r(\xi, \eta) + b_s\psi_s(\xi, \eta) \\ \Psi_{3(i-1)+3}^x(\xi, \eta) &= \psi_i(\xi, \eta) - c_r\psi_r(\xi, \eta) - c_s\psi_s(\xi, \eta) \end{aligned} \quad (6.37)$$

$$\begin{aligned} \Psi_{3(i-1)+1}^y(\xi, \eta) &= \frac{3}{2}(d_r\psi_r(\xi, \eta) - d_s\psi_s(\xi, \eta)) \\ \Psi_{3(i-1)+2}^y(\xi, \eta) &= -\psi_i(\xi, \eta) + e_r\psi_r(\xi, \eta) + e_s\psi_s(\xi, \eta) \\ \Psi_{3(i-1)+3}^y(\xi, \eta) &= -b_r\psi_r(\xi, \eta) - b_s\psi_s(\xi, \eta) \end{aligned} \quad (6.38)$$

with $i = 1, 2, 3, 4$ and the relationship (i, r, s) as $(1, 5, 8), (2, 6, 5), (3, 7, 6)$ and $(4, 8, 7)$.

Relating curvatures to displacements yield:

$$\boldsymbol{\chi} = \mathbf{B}_{bend}\mathbf{U} \quad (6.39)$$

with \mathbf{B}_{bend} constructed as follows:

$$\mathbf{B}_{bend} = \begin{pmatrix} \frac{\partial \Psi^x}{\partial x} \\ \frac{\partial \Psi^y}{\partial y} \\ \frac{\partial \Psi^x}{\partial y} + \frac{\partial \Psi^y}{\partial x} \end{pmatrix} = \begin{pmatrix} j_{11} \frac{\partial \Psi^x}{\partial \xi} + j_{12} \frac{\partial \Psi^x}{\partial \eta} \\ j_{21} \frac{\partial \Psi^y}{\partial \xi} + j_{22} \frac{\partial \Psi^y}{\partial \eta} \\ j_{11} \frac{\partial \Psi^y}{\partial \xi} + j_{12} \frac{\partial \Psi^y}{\partial \eta} + j_{21} \frac{\partial \Psi^x}{\partial \xi} + j_{22} \frac{\partial \Psi^x}{\partial \eta} \end{pmatrix} \quad (6.40)$$

and the inverse Jacobian entries $j_{\alpha\beta}$:

$$\mathbf{J} = \frac{1}{4} \begin{pmatrix} x_{21} + x_{34} + \eta(x_{12} + x_{34}) & y_{21} + y_{34} + \eta(y_{12} + y_{34}) \\ x_{32} + x_{41} + \xi(x_{12} + x_{34}) & y_{32} + y_{41} + \xi(y_{12} + y_{34}) \end{pmatrix} = \begin{pmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{pmatrix} \quad (6.41)$$

$$j_{11} = \frac{J_{22}}{\det[J]}, \quad j_{12} = \frac{-J_{12}}{\det[J]}, \quad j_{21} = \frac{-J_{21}}{\det[J]}, \quad j_{22} = \frac{J_{11}}{\det[J]}$$

6.1.3. Combined formulation

With the separate membrane and bending B matrices developed, the combined shell B matrix \mathbf{B}_{comb} . can be constructed to form the element stiffness matrix.

$$\mathbf{K}_{el} = \mathbf{B}_{comb}^T \mathbf{C} \mathbf{B}_{comb} \quad (6.42)$$

$$\mathbf{B}_{comb} = (\mathbf{L} + \mathbf{B}_h) + \mathbf{B}_{bend} = \mathbf{B}_{mem} + \mathbf{B}_{bend} = \begin{pmatrix} \mathbf{B}_{comb1} & \mathbf{B}_{comb2} & \mathbf{B}_{comb3} & \mathbf{B}_{comb4} \end{pmatrix} \quad (6.43)$$

The combination of *membrane* and *bending* matrices must consider the DOF ordering of each component and the relation to the total shell DOF ordering, as shown below:

$$\mathbf{u}_i = \begin{pmatrix} \textcolor{blue}{u_{xi}} \\ \textcolor{blue}{u_{yi}} \\ \textcolor{red}{u_{zi}} \\ \textcolor{red}{\beta_{xi}} \\ \textcolor{red}{\beta_{yi}} \\ \textcolor{blue}{\beta_{zi}} \end{pmatrix} \quad (6.44)$$

Considering this, the addition of the membrane (basic and higher order) B matrices and the bending B matrix is conducted for each node i as follows:

$$\mathbf{B}_{comb\ i} = \begin{pmatrix} \mathbf{B}_{mem}[1, 3(i-1)+1] & \mathbf{B}_{mem}[1, 3(i-1)+2] & 0 \\ \mathbf{B}_{mem}[2, 3(i-1)+1] & \mathbf{B}_{mem}[2, 3(i-1)+2] & 0 \\ 0 & 0 & \mathbf{B}_{bend}[1, 3(i-1)+1] \\ 0 & 0 & \mathbf{B}_{bend}[2, 3(i-1)+1] \\ 0 & 0 & \mathbf{B}_{bend}[3, 3(i-1)+1] \\ \mathbf{B}_{mem}[3, 3(i-1)+1] & \mathbf{B}_{mem}[3, 3(i-1)+2] & 0 \\ 0 & 0 & \mathbf{B}_{mem}[1, 3(i-1)+3] \\ 0 & 0 & \mathbf{B}_{mem}[2, 3(i-1)+3] \\ \mathbf{B}_{bend}[1, 3(i-1)+2] & \mathbf{B}_{bend}[1, 3(i-1)+3] & 0 \\ \mathbf{B}_{bend}[2, 3(i-1)+2] & \mathbf{B}_{bend}[2, 3(i-1)+3] & 0 \\ \mathbf{B}_{bend}[3, 3(i-1)+2] & \mathbf{B}_{bend}[3, 3(i-1)+3] & 0 \\ 0 & 0 & \mathbf{B}_{mem}[3, 3(i-1)+3] \end{pmatrix} \quad (6.45)$$

6.2. Stiffness matrix implementation

With the formulation of the ANDES-DKQ shell element established, a high level overview of its implementation in KRATOS is discussed in this section.

The new quad element is implemented in the files `shell_thin_element_3D4N.hpp` and `shell_thin_element_3D4N.cpp`, which are compiled into the 'StructuralMechanicsApplication' module of Kratos. Similar to the DSG triangle element, the new ANDES-DKQ element class `ShellThinElement3D4N` is inherited from the Kratos base class `Element` and also leverages the existing capabilities other Kratos classes offer. The general workflow of calculating the ANDES-DKQ stiffness matrix is as follows:

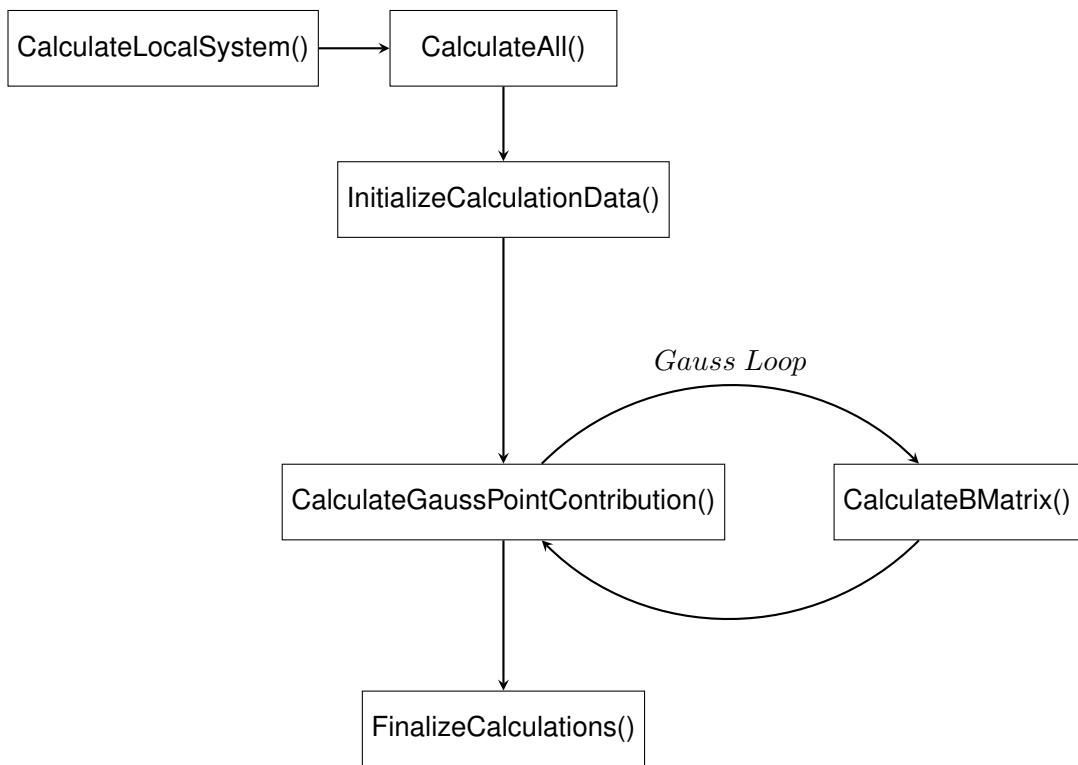


Figure 32 High level overview of ANDES-DKQ element workflow

As per the DSG triangle element, the re-implemented virtual method `CalculateLocalSystem()` is called by the KRATOS framework automatically for every `ShellThinElement3D4N` in the job definition. This method simply calls `CalculateAll()`, which initializes calculating the stiffness matrix by calling `InitializeCalculationData()`, `CalculateGaussPointContribution()` and `FinalizeCalculations()`.

`InitializeCalculationData()` is called first, and pre-calculates quantities so they can be removed from the Gauss loop. These quantities include the ANDES basic lumping matrix \mathbf{L} , the ANDES higher order strain-displacement matrices \mathbf{B}_{hi} and all DKQ coefficients in equation (6.35).

`CalculateAll()` then calls `CalculateGaussPointContribution()` which starts the Gauss integration loop. At each Gauss point `CalculateGaussPointContribution()` performs Gauss integration of the expression $\mathbf{K}_{contribution} = \mathbf{B}_{comb}^T \mathbf{C} \mathbf{B}_{comb} dA$, with the current \mathbf{B}_{comb} determined by calling `CalculateBMatrix()`.

With the Gauss integration complete, `CalculateAll()` lastly calls `FinalizeCalculations()` which transforms the calculated element stiffness from local to global coordinates.

The following pseudocode summarises the key calls and operations involved in calculating the ANDES-DKQ element stiffness matrix.

Algorithm 3 ANDES-DKQ element stiffness matrix pseudocode

Require: Coordinate transformation instance

```
1: call CalculateAll()
2: Resize LHS and RHS
3: call InitializeCalculationData(data)
4: Calculate integration areas  $dA = w_i \cdot \det J(x_i, \eta)$ 
5: Determine basic membrane strain displacement L
6: Construct membrane higher order filter matrix H
7: Arrange higher order natural strain matrices  $Q_i$ 
8: Transform  $Q_i$  into  $B_{hi}$ 
9: Determine  $\bar{B}_h$ 
10: Pre-calculate all DKQ coefficients
11: while  $gaussPoint < 4$  do
12:   call CalculateGaussPointContribution(data)
13:   call CalculateBMatrix(data)
14:     Calculate and combine  $B_{mem}$  and  $B_{bend}$  into B
15:   call CalculateSectionResponse(data)
16:     Calculate material properties C
17:     Add stiffness matrix Gauss point contribution to LHS
18: end while
19: Modify RHS residual vector
20: call FinalizeCalculations(data, displacements, LHS, RHS)
21: call AddBodyForces(data, RHS)
```

6.3. Mass matrix formulation

The mass matrix is necessary to facilitate dynamic analysis with the thin quadrilateral shell element. Both lumped and consistent options have been implemented.

6.3.1. Lumped mass matrix

The lumped mass matrix is the default option used due to its speed of construction without significant loss of accuracy. The general form of the ANDES-DKQ lumped matrix is similar to the DSG lumped matrix.

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{M}_4 \end{pmatrix} \quad \text{where} \quad \mathbf{M}_i = \begin{pmatrix} \bar{m} & 0 & 0 & 0 & 0 & 0 \\ 0 & \bar{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & \bar{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (6.46)$$

The general lumped mass is determined for a multi-ply material with n plies each of t_i thickness and ρ_i density as follows:

$$\bar{m} = \frac{A}{4} \sum_{i=1}^n \rho_i h_i \quad (6.47)$$

For a single layer material of area A this reduces to:

$$\bar{m} = \frac{A}{4} \rho h \quad (6.48)$$

6.3.2. Consistent mass matrix

A consistent mass matrix is also provided, and once again has a similar form to the DSG consistent mass matrix. The continuous form for a shell with constant thickness is re-written below for clarity:

$$\mathbf{M}_C = h \int_{\Omega} \rho \left[\mathbf{N}_T^T \mathbf{N}_T + \frac{h^2}{12} \mathbf{N}_R^T \mathbf{N}_R \right] d\Omega \quad (6.49)$$

For the quadrilateral element, the nodal shape function matrices are expanded from 3 nodes

to 4 nodes:

$$\mathbf{N}_T = \begin{pmatrix} \mathbf{N}_{T_1} & \mathbf{N}_{T_2} & \mathbf{N}_{T_3} & \mathbf{N}_{T_4} \end{pmatrix}, \quad \mathbf{N}_R = \begin{pmatrix} \mathbf{N}_{R_1} & \mathbf{N}_{R_2} & \mathbf{N}_{R_3} & \mathbf{N}_{R_4} \end{pmatrix} \quad (6.50)$$

$$\mathbf{N}_{T_i} = \begin{pmatrix} N_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & N_i & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & N_i & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_i & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \mathbf{N}_{R_i} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & N_i & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & N_i & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & N_i \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha N_i \end{pmatrix} \quad (6.51)$$

Unlike the DSG element which could be evaluated directly, the consistent mass matrix of the ANDES-DKQ is evaluated with numerical 2x2 Gauss integration. Thus, equation 6.49 can be equivalently written in parametric space:

$$\mathbf{M}_C = h \int_{-1}^1 \int_{-1}^1 \rho \left[\mathbf{N}_T^T \mathbf{N}_T + \frac{h^2}{12} \mathbf{N}_R^T \mathbf{N}_R \right] d\eta d\xi \det(\mathbf{J}) \quad (6.52)$$

Shifting over to Gauss integration, the consistent mass matrix is determined numerically from:

$$\mathbf{M}_C = h \sum_{i=1}^{4GP} \rho(\xi_i, \eta_i) \left[\mathbf{N}_T^T(\xi_i, \eta_i) \mathbf{N}_T(\xi_i, \eta_i) + \frac{h^2}{12} \mathbf{N}_R^T(\xi_i, \eta_i) \mathbf{N}_R(\xi_i, \eta_i) \right] w_i \det(\mathbf{J}(\xi_i, \eta_i)) \quad (6.53)$$

6.4. Stress and strain recovery

The stresses and strains of the ANDES-DKT quadrilateral element are recovered in a similar way to the DSG triangle, with the added complication of the multiple Gauss points.

The non-zero local strains ($\epsilon_{zz}, \epsilon_{xz}, \epsilon_{yz} = 0$) of the 4 noded 3 parameter element can be arranged in a vector form:

$$\boldsymbol{\epsilon}^T = \begin{pmatrix} \epsilon_1 & \epsilon_2 & \epsilon_3 & \epsilon_4 \end{pmatrix} \quad \text{with} \quad \boldsymbol{\epsilon}_i^T = \begin{pmatrix} \epsilon_{xx} & \epsilon_{xx} & 2\epsilon_{xy} & \epsilon_{xx} & \kappa_{xx} & \kappa_{yy} & 2\kappa_{xy} \end{pmatrix} \quad (6.54)$$

The nodal strain vector is recovered from the displacement field by applying the strain displacement matrix, which varies over the element.

$$\boldsymbol{\epsilon}(\xi, \eta) = \mathbf{B}(\xi, \eta) \mathbf{u}(\xi, \eta) \quad (6.55)$$

As per the DSG triangle, the strains and stresses are calculated at the Gauss points of the element, with the ANDES-DKT element having four Gauss points ($j = 1, 2, 3, 4$). Thus, the strain vector at each Gauss point j is recovered from the discrete nodal displacements $\hat{\mathbf{u}}_i$ as follows:

$$\boldsymbol{\epsilon}_{GP_j} = \mathbf{B}(\xi_j, \eta_j) \sum_{i=1}^{4 \text{ nodes}} N_i(\xi_j, \eta_j) \hat{\mathbf{u}}_i \quad (6.56)$$

With the strains determined, the stresses at each Gauss point are recovered with the material matrix (which in the general case may vary over the element).

$$\boldsymbol{\sigma}_{GP_j} = \mathbf{C}_{GP_j} \boldsymbol{\epsilon}_{GP_j} \quad (6.57)$$

The general implementation of the stress and strain recovery described above is illustrated in pseudocode algorithm 4.

6.4.1. Von Mises equivalent stress

As per the DSG element, the Von Mises equivalent stress is calculated for the ANDES-DKQ element by double contracting the deviatoric stress tensor. The result to calculate Von Mises equivalent stresses of the DSG element is recalled:

$$\sigma_{VM} = \sqrt{\sigma_{xx}^2 - \sigma_{xx}\sigma_{yy} + \sigma_{yy}^2 + 3(\sigma_{xy}^2 + \sigma_{xz}^2 + \sigma_{yz}^2)} \quad (6.58)$$

Corresponding to the underlying 3 parameter model of the ANDES-DKQ in which $\sigma_{3i} = 0$, the expression above can be simplified, yielding the formula to determine the Von Mises equivalent stress for the ANDES-DKQ element:

$$\sigma_{VM} = \sqrt{\sigma_{xx}^2 - \sigma_{xx}\sigma_{yy} + \sigma_{yy}^2 + 3\sigma_{xy}^2} \quad (6.59)$$

Algorithm 4 ANDES-DKT quadrilateral element stress and strain recovery

Require: *requestedQuantity*, calculation of nodal displacements

```
1: call InitializeCalculationData(data)
2: Calculate constant components of strain-displacement matrix B
3: Retrieve element localDisplacements
4: while gaussPoint < 4 do
5:   call CalculateGaussPointContribution(data)
6:   call CalculateBMatrix(data)
7:     Calculate combined B at current gaussPoint
8:   generalizedStrains = product(B, localDisplacements)
9:   if requestedQuantity requires stress then
10:    call CalculateSectionResponse(data)
11:    generalizedStresses = product (C, generalizedStrains)
12:    Decimal correction of generalizedStresses
13:  end if
14:  Decimal correction of generalizedStrains
15:  if requestedQuantity requires local orientation then
16:    Rotate requestedQuantity to local orientation
17:  end if
18:  Assemble requestedQuantity into outputMatrix
19:  if requestedQuantity requires global orientation then
20:    Rotate outputMatrix to global orientation
21:  end if
22:  Interpolate outputMatrix to standard Gauss points for visualisation
23: end while
```

Chapter 7 Extension of shells to composite laminates - revised 1

WITH the ANDES-DKQ and DSG elements formulated and implemented for isotropic materials in the preceding chapters, the extension into composite laminate materials is now considered. The relevant background is covered in chapter 3 which defines the scope of the formulation and implementation discussed henceforth.

7.1. Composite constitutive matrix formulation

The formulation of orthotropic laminate composite is developed further by recalling the key results presented in chapter 3 and approaching from a programming perspective.

The general laminate shell stress resultants are related to the generalized mid-plane strains via the total combined constitutive matrix $\bar{\mathbf{C}}$ as follows:

$$\bar{\mathbf{N}} = \bar{\mathbf{C}}\bar{\boldsymbol{\epsilon}} = \begin{pmatrix} N_{xx} \\ N_{yy} \\ N_{xy} \\ M_{xx} \\ M_{yy} \\ M_{xy} \\ Q_x \\ Q_y \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{0} \\ \mathbf{B} & \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha \begin{pmatrix} A_{44} & A_{45} \\ A_{45} & A_{55} \end{pmatrix} \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ 2\epsilon_{xy} \\ \kappa_{xx} \\ \kappa_{yy} \\ 2\kappa_{xy} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \end{pmatrix} \quad (7.1)$$

The individual entries of the material sub matrices \mathbf{A} , \mathbf{B} and \mathbf{C} are also recalled:

$$A_{ij} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \bar{Q}_{ij} dz, \quad B_{ij} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \bar{Q}_{ij} z dz, \quad D_{ij} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \bar{Q}_{ij} z^2 dz \quad (7.2)$$

The entries $\bar{Q}_{ij}^{(k)}$ are rotated lamina stiffness's as per equation 3.12, related to the pure lamina-aligned stiffness's through the transformation matrix \mathbf{T} of equation 3.11.

By shifting perspective from abstract formulation to a more programmable approach, the total combined constitutive matrix $\bar{\mathbf{C}}$ of a laminate with n laminae and total thickness h can be decomposed into lamina contributions. Furthermore, the rotation of the lamina stiffness's can be postponed until each lamina constitutive matrix is assembled:

$$\bar{\mathbf{C}} = \sum_{k=1}^n \mathbf{T}^{T(k)} \mathbf{C}^{(k)} \mathbf{T}^{(k)} \quad (7.3)$$

The integral limits of the material sub matrices, which must only span the thickness of each lamina k , are updated accordingly:

$$A_{ij}^{(k)} = \int_{z_k}^{z_{k+1}} Q_{ij}^{(k)} dz, \quad B_{ij}^{(k)} = \int_{z_k}^{z_{k+1}} Q_{ij}^{(k)} z dz, \quad D_{ij}^{(k)} = \int_{z_k}^{z_{k+1}} Q_{ij}^{(k)} z^2 dz \quad (7.4)$$

The above formulation details concern themselves with determining the stiffness matrix of a composite laminate shell. Recovery of lamina stresses also departs from the isotropic formulation, however, the details presented in 3.3 need little further development to be programmable, and are not reproduced here.

7.2. Composite constitutive matrix implementation

The general approach of extending both shell elements to composite laminate materials in a sensible and efficient manner is to abstract the composite specifics from the individual element formulation level as much as possible. This approach reduces duplicate coding and provides a centralized platform to develop and modify the composite capabilities of the elements in the future. Almost all of the composite laminate implementation is achieved by modifying the existing `ShellCrossSection` class, common to all Kratos structural shell elements, and adding a new constitutive law class `LinearElasticOrthotropic2DLaw`. Naturally, access to the new functionality required minor modifications to the code of each shell element. With a view to illustrate the relationship between the individual elements and the `ShellCrossSection` and `LinearElasticOrthotropic2DLaw` classes, a generalized workflow of both elements is outlined in the following flowchart, highlighting only methods relevant to composites:

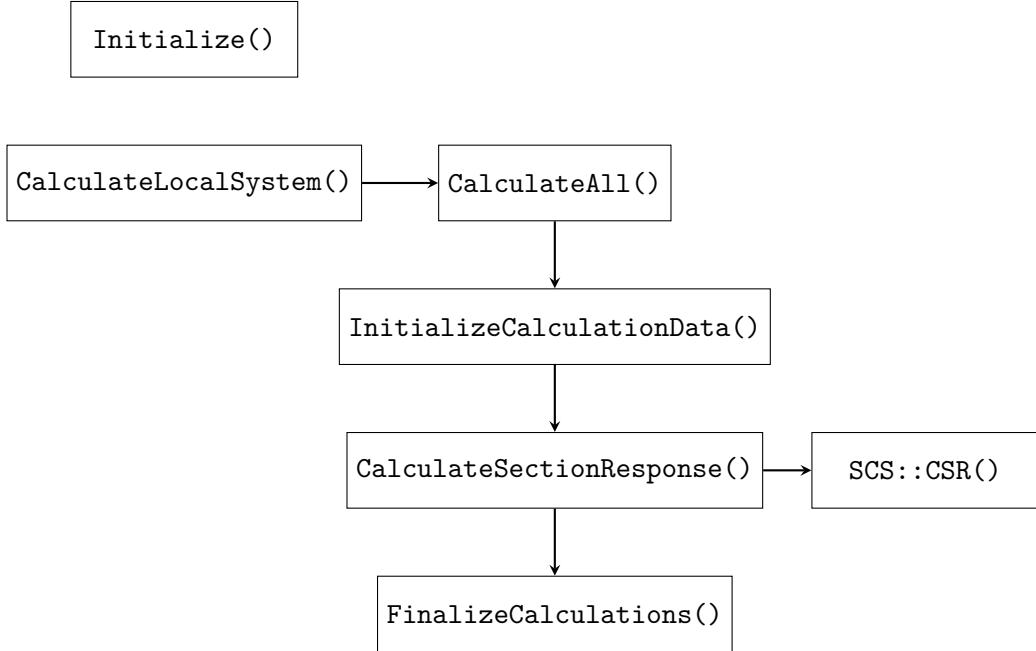


Figure 33 High level overview of composite element workflow

The key differences in the workflow presented above, compared to those previously illustrated, is the explicit inclusion of `Initialize()` and `ShellCrossSection::CalculateSectionResponse()`, noted as `SCS::CSR()`, methods, which implement composite functionality.

`Initialize()` is called upon element creation, completely separate from `CalculateLocalSystem()`, and is responsible for reading in element material properties. Upon reading material data specified with the Kratos variable `SHELL_ORTHOTROPIC_LAYERS`, an instance of `LinearElasticOrthotropic2DLaw` is created and assigned to the `ShellCrossSection`, followed by a call to the `ShellCrossSection` method `ShellCrossSection::ParseOrthotropicPropertyMatrix()` is called which parses and stores the orthotropic material data for all laminae. The checking and parsing of `SHELL_ORTHOTROPIC_LAYERS` material data is performed entirely within the `ShellCrossSection` class, which facilitates un-invasive integration into shell elements and also provides centralization for further composite development should the need arise.

Following the call of `Initialize()`, `CalculateLocalSystem()` is called to compute the element-specific stiffness matrix. The ANDES-DKQ and DSG specifics of `CalculateLocalSystem()` and its subsequent calls are detailed in their respective chapters, the focus here are specifics relating to composite laminates. Logically, the call of `CalculateSectionResponse()` is relevant, itself calling the `ShellCrossSection` function `ShellCrossSection::CalculateSectionResponse()`.

The `ShellCrossSection` method `ShellCrossSection::CalculateSectionResponse()` serves

to calculate the total laminate constitutive matrix $\bar{\mathbf{C}}$ by 'stacking' the results of each lamina together. A loop is established over all laminae k , with each iteration retrieving the raw laminae stiffnesses $\mathbf{Q}^{(k)}$, integrating and assembling them into a lamina constitutive matrix $\mathbf{C}^{(k)}$ and then transforming them according to the stacking sequence of the lamina in the laminate. At the end of this method, the total laminate constitutive matrix $\bar{\mathbf{C}}$ is assembled and suitable for use in element-specific methods to determine the stiffness matrix.

A summary of the above flowchart and commentary is presented in the generalized composite shell element stiffness matrix pseudocode below:

Algorithm 5 Generalized composite shell element stiffness matrix pseudocode

Require: Orthotropic laminate material data specified

```

1: call Initialize()
2:   if materialProperties has orthotropic laminate data then
3:     Create and assign LinearElasticOrthotropic2DLaw
4:     Parse composite material data in ShellCrossSection
5:   end if
6:   call CalculateLocalSystem()
7:   Perform element-specific calculations (may enter Gauss loop)
8:   call CalculateSectionResponse()
9:     call ShellCrossSection::CalculateSectionResponse()
10:    while ( $k <$  number of laminae) do
11:      Retrieve stiffnesses of  $k^{th}$  lamina  $\mathbf{Q}^{(k)}$  from material law
12:      Assemble and integrate unrotated lamina constitutive matrix  $\mathbf{C}^{(k)}$ 
13:      if laminaOrientationAngle  $\neq 0$  then
14:        Transform  $\mathbf{C}^{(k)}$  into  $\bar{\mathbf{C}}^{(k)}$ 
15:      end if
16:      Add lamina  $\bar{\mathbf{C}}^{(k)}$  to laminate  $\bar{\mathbf{C}}$ 
17:    end while
18:   Assemble element-specific stiffness matrix and finalize calculations

```

7.3. Composite stress recovery implementation

The extension of both shell elements to compute composites naturally entails additional functionality to calculate composite stresses. As described in section 3.3, the laminate stresses and strains require the generalized shell mid-plane strains be determined (refer to the stress and strain recovery sections of each element for details). With the mid-surface strains available, the new element method `CalculateLaminaStrains()`, implemented in both elements, calculates and assembles the in-plane strains (as per equation 3.26) for the top and bottom surface of each lamina into a vector of vectors. It should be noted that a constant transverse shear strain is assumed across the laminate thickness, as per equation 3.27. Stresses of the top and bottom surfaces of each lamina are determined in the new element

method `CalculateLaminaStresses()` from the rotated raw lamina stiffnesses and the lamina strains according to equation 3.32. Similar to the isotropic stress recovery implemented, the lamina stresses can be output either locally or globally oriented.

The following pseudocode provides an overview of recovering stresses from the shell elements.

Algorithm 6 Generalized composite shell element stress and strain recovery

Require: *requestedQuantity = laminateStresses*, calculation of nodal displacements

```

1: Perform all preliminary operations necessary to determine mid-plane generalizedStrains
2: while Gauss loop do
3:   Calculate B matrix at current gaussPoint
4:   generalizedStrains = product(B, localDisplacements)
5:   call CalculateLaminaStrains(data)
6:   Determine laminateStrains at top and bottom surface of each lamina
7:   call CalculateLaminaStresses(data)
8:   Retrieve raw laminae stiffnesses
9:   while (k < number of laminae) do
10:    laminateStresses(2k) = product(C(k), laminateStrains(2k))
11:    laminateStresses(2k+1) = product(C(k), laminateStrains(2k+1))
12:   while end
13:   if laminateStresses requires local orientation then
14:     Rotate laminateStresses to local orientation
15:   end if
16:   Assemble laminateStresses into outputMatrix
17:   if laminateStresses requires global orientation then
18:     Rotate outputMatrix to global orientation
19:   end if
20:   Interpolate outputMatrix to standard Gauss points for visualisation
21: end while
```

7.4. Tsai-Wu failure criterion implementation

The Tsai-Wu failure criterion described in section 3.4 is also implemented for both shell elements. Extending functionality to include the Tsai-Wu failure criterion requires that the lamina material strengths outlined in 3.35 be specified in the Kratos input file. Correspondingly, the `ShellCrossSection` parsing method `ShellCrossSection::ParseOrthotropicPropertyMatrix()` has been modified to discern and properly parse the inclusion of material strengths. The actual calculation of the Tsai-Wu reserve factor can be seen as an extension of calculating the lamina stresses and subsequently interacting with the aforementioned lamina material strengths. After calculation of the lamina stresses, as per the previous section 7.3, the stresses of each lamina are rotated to

that lamina's natural orientation with the transformation matrix described in equation 3.33. A loop over all laminate laminae is established in which each iteration determines the minimum Tsai-Wu reserve factor (refer equation 3.36) of the k^{th} lamina, considering top and bottom surfaces, with the method `CalculateTsaiWuPlaneStress()`. The final output is the minimum Tsai-Wu reserve factor across all the whole laminate.

The following pseudocode provides an overview of determining the Tsai-Wu reserve factor of the shell elements.

Algorithm 7 Generalized composite shell element Tsai-Wu reserve factor calculation

Require: $requestedQuantity = tsaiWuReserveFactor$, calculation of nodal displacements

- 1: Recover and store $laminaeStrengths$
- 2: **while** Gauss loop **do**
- 3: Perform all operations necessary to determine $laminateStresses$
- 4: Rotate $laminateStresses$ to respective laminae natural orientations
- 5: **while** ($k <$ number of laminae) **do**
- 6: **call** `CalculateTsaiWuPlaneStress()`
- 7: Return min $tsaiWuReserveFactor$ of top and bottom surfaces for k^{th} ply
- 8: Store minimum $tsaiWuReserveFactor$ across all laminae
- 9: **while end**
- 10: Output $tsaiWuReserveFactor$ to standard Gauss points for visualisation
- 11: **end while**

Chapter 8 Validation of elements

VALIDATION is as important to proper engineering analysis as the calculations performed. The following tests across statics, geometrically non-linear analysis, dynamics, quantity recovery and composites interrogate the correct implementation of the element formulations and also gives an indication of their performance.

8.1. Static tests: shell obstacle course

Considered as the standard benchmark for shell elements, the shell obstacle course proposed by Belytschko [6] runs the elements through 3 problems involving complex loading states. These complex loading states often pose difficulties for un-enhanced elements, which are also tested here.

The *Basic-DKQ* element is a quadrilateral element with an un-enhanced membrane formulation and the DKQ bending formulation. Refer to Appendix C for full details. Any performance differences that arise between this element and the ANDES-DKQ element can be attributed to the ANDES element technology.

The *Basic-T3* element is a triangular element with an un-enhanced shear formulation and no correction to the shear component of the material matrix. Refer to Appendix D for full details. Any performance differences that arise between this element and the DSG element can be attributed to the DSG element technology.

Furthermore, context of element performance is provided by including results from the existing *KRATOS Q4* five parameter quadrilateral element (EAS-MITC formulation) and the *KRATOS T3* three parameter triangle element (ANDES-DKT formulation).

8.1.1. Scordelis-Lo roof

The first problem of the shell obstacle course is the Scordelis-Lo roof, which is part of a cylindrical shell fixed by rigid diaphragms at its axial ends. The loading is a pseudo-gravity distributed load that has a magnitude of 90. Due to symmetry, only a quarter of the shell is modelled. The key result is the vertical displacement of the lateral side at the midpoint, denoted by u in the following diagram. The reference value is $u_{ref} = 0.3024$.

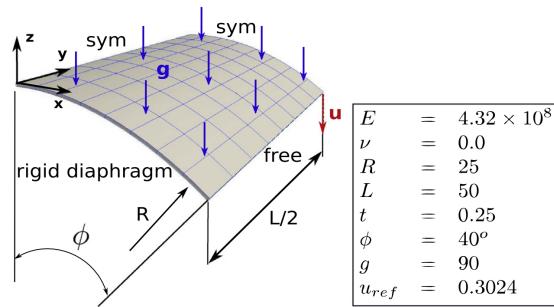
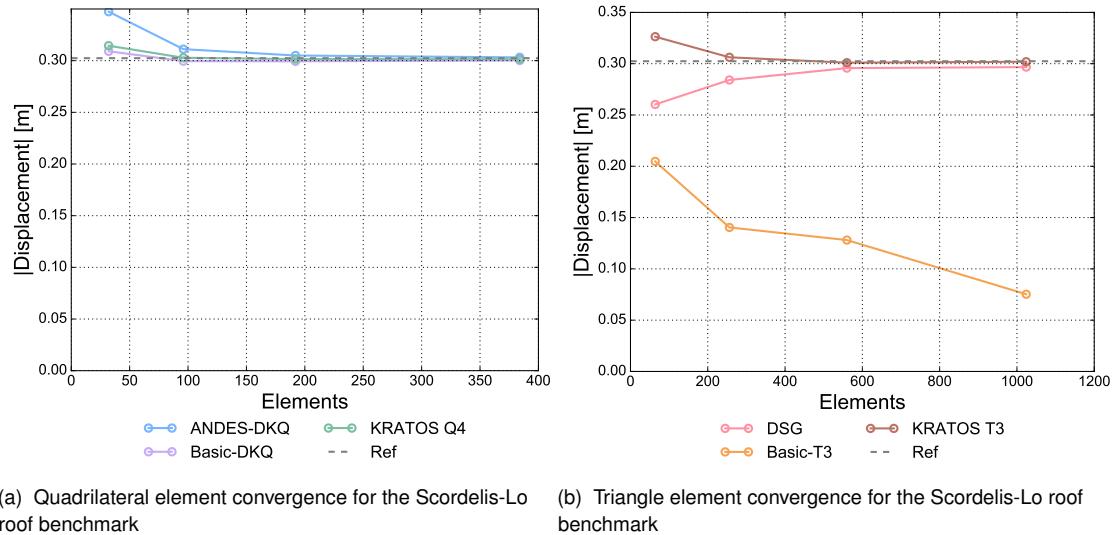


Figure 34 Definition of the Scordelis-Lo roof benchmark[13]



(a) Quadrilateral element convergence for the Scordelis-Lo roof benchmark

(b) Triangle element convergence for the Scordelis-Lo roof benchmark

Figure 35 Scordelis-Lo roof benchmark results

The convergence graphs above indicate the ANDES-DKQ and DSG elements agree to the reference solution. Conversely, the Basic-T3 element shows very poor performance. Given that the Basic-DKQ performs well (which is immune to transverse shear locking), it's suspected that transverse shear locking is crippling the Basic-T3 element, while the DSG element technology effectively mitigates this for the DSG element.

8.1.2. Pinched cylinder

The second problem of the shell obstacle course is the pinched cylinder, which considers a cylindrical shell fixed by rigid diaphragms at its axial ends. The loading consists of two opposing compressive point loads at the centre of the shell. Due to symmetry only an eighth of the shell is modelled. The key result is the vertical displacement under the point load, denoted by u in the following diagram. The reference value is $u_{ref} = 1.8248 \times 10^{-5}$.

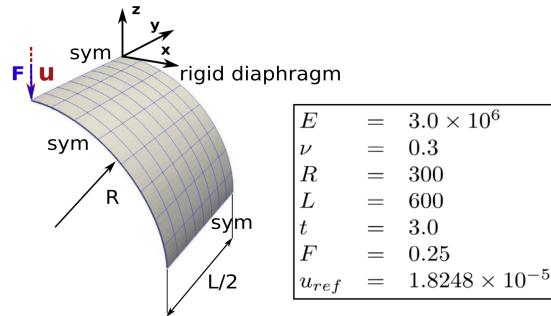
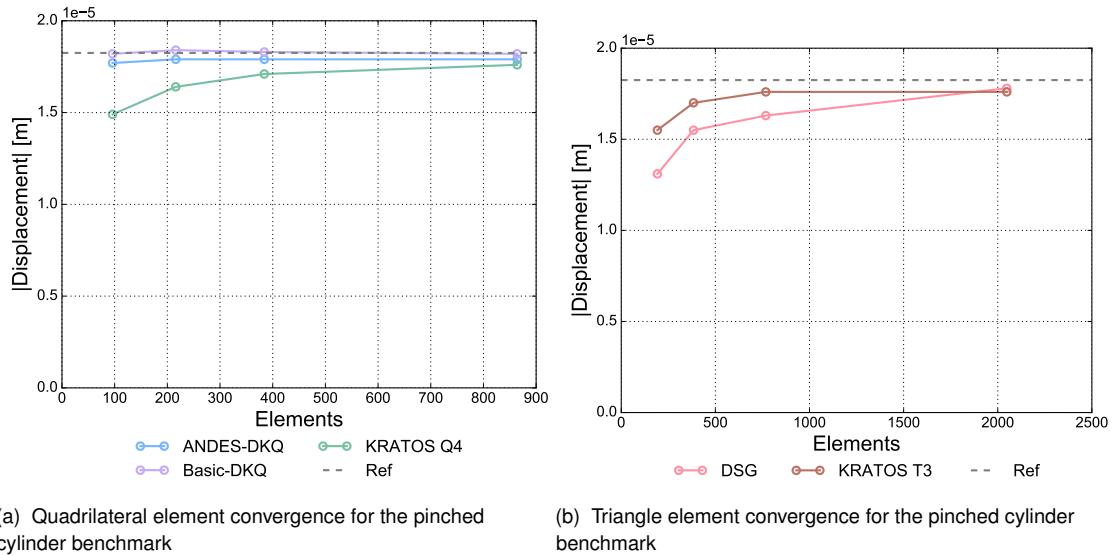


Figure 36 Definition of the pinched cylinder benchmark[13]



(a) Quadrilateral element convergence for the pinched cylinder benchmark

(b) Triangle element convergence for the pinched cylinder benchmark

Figure 37 Pinched cylinder benchmark results

The good performance of both the ANDES-DKQ and DSG elements is demonstrated in the convergence graphs above. The Basic-T3 results were in the order of 1×10^{-3} (roughly 100 times greater than the reference solution) and were omitted from the graph for clarity of scale. Once again, it is clear that the computationally inexpensive DSG element technology drastically improves performance from the un-enhanced Basic-T3 to the DSG element.

8.1.3. Pinched hemisphere

The last test in the shell obstacle course is the pinched hemisphere, which considers a hemispherical shell loaded with opposing point loads along its equator. Due to symmetry only a quarter of the shell is modelled. The key result is the 'x' displacement along one of the point loads, denoted by u in the following diagram. The reference value is $u_{ref} = 0.0924$.

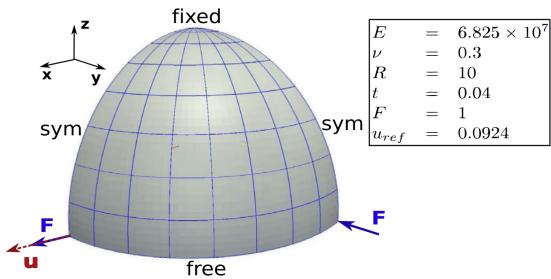


Figure 38 Definition of the pinched hemisphere benchmark [13]

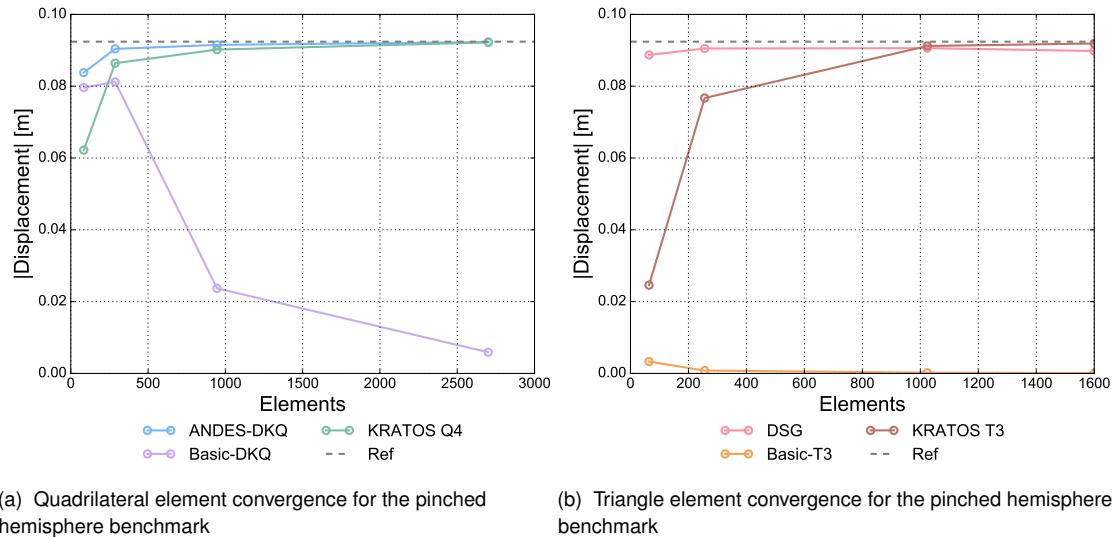


Figure 39 Pinched hemisphere benchmark results

The ANDES-DKQ and DSG elements both perform well in the final statics test, as per the convergence graphs above. It is observed that the Basic-DKQ element appears to exhibit membrane locking corresponding to the high double curvature of the problem ($R_1 = R_2 = 10$) compared to the Scordelis-Lo roof ($R_1 = 10$, $R_2 = \infty$) and the pinched cylinder ($R_1 = 300$, $R_2 = \infty$). The ANDES element technology clearly prevents this deleterious effect. The poor performance of the Basic-T3 element compared to the DSG element once again highlights the effectiveness of the DSG element technology in preventing transverse shear locking.

8.2. Geometrically non-linear tests

Large rotations and displacements mark the departure from geometrically linear to non-linear analyses, however, the assumption of small strains is still maintained. The extension of the elements to geometrically non-linear problems is handled by employing an existing Kratos class which provides co-rotational transformations for shells. At each increment in the non-linear solution, the large displacements and rotations of each element is mapped by rigid body translations and rotations, thus limiting the strains experienced by the element to reasonably small magnitudes. The performance of the element in geometrically non-linear problems is considered with two benchmarks.

8.2.1. Hinged cylindrical roof

The first geometrically non-linear benchmark is the snap-through of a hinged cylindrical roof under a central point load $P_{max} = 3000$ [46]. As per the diagram below, the roof geometry is defined with the parameters: $L = 254$, $R = 2540$, $\theta = 0.1 \text{ rad}$ and $h = 12.7$. The material is defined with a Young's modulus $E = 3102.75$ and Poisson's ratio of $\nu = 0.3$.

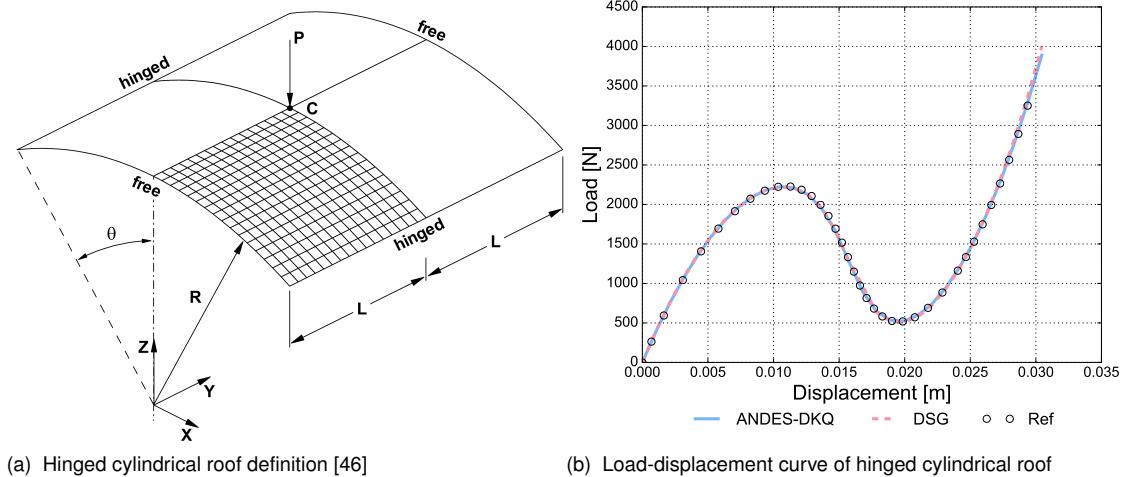


Figure 40 Hinged cylindrical roof benchmark

The load displacement curve plots the equilibrium path for the ANDES-DKQ and DSG elements against the reference path from [46]. The full equilibrium path isn't resolved because Kratos only has a load control non-linear solution method implemented, translating to the restriction of only resolving monotonically increasing paths. Regardless, both elements clearly follow the initial path, and then rejoin the reference solution to correctly resolve the structure in its snapped-through state.

8.2.2. Open cylinder pull-out

The second geometrically non-linear benchmark is the pull-out of an open cylinder with a load $P_{max} = 40\,000$. The geometry of the cylinder is $L = 10.35$, $R = 4.953$ and $h = 0.094$ while the linear elastic material is characterised by $E = 10.5 \times 10^6$ and $\nu = 0.3125$. The measured displacement is the vertical deformation u_z at the point of load application.

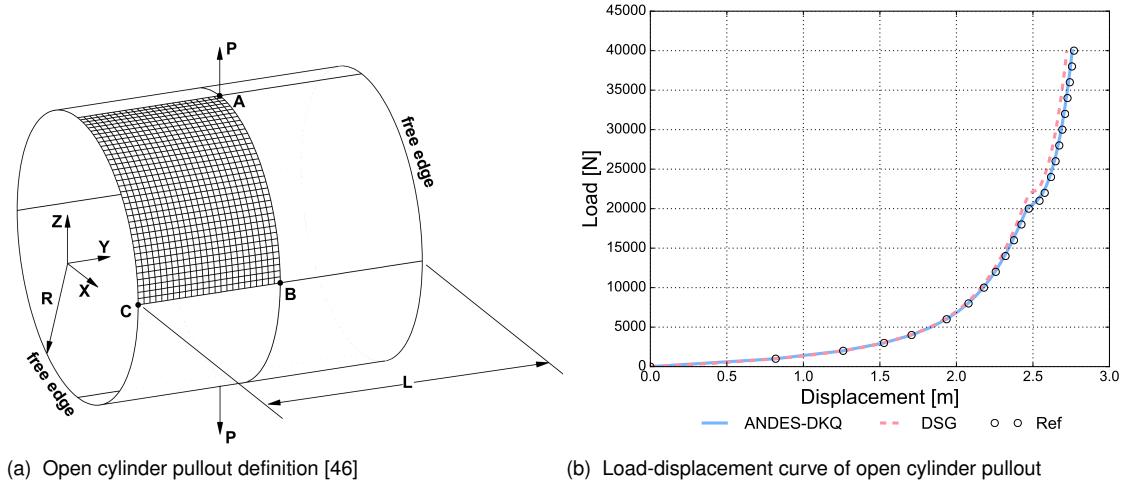


Figure 41 Open cylinder pullout benchmark

The load-displacement curve above plots the equilibrium path for the ANDES-DKQ and DSG elements against the reference solution [46]. Although both elements closely follow the reference path, the ANDES-DKQ performs better in this test than the DSG element. Despite this, the error of the DSG element at maximum load is still only 1.7%.

8.3. Dynamic tests

Dynamic problems introduce inertial effects into the array of phenomena analysed. Combined with the aforementioned co-rotational formulation, it is possible to accurately resolve bodies undergoing large movements over time.

8.3.1. Shell pendulum

The first dynamic benchmark is a simple shell pendulum allowed to freely rotate along one hinged edge. The initial horizontal configuration of the $1m \times 1m \times 0.1m$ thick square plate is subject to gravity $g = 9.8 m/s^2$ acting in the vertical Z direction. The material of the plate is described by $E = 1 \times 10^9 Pa$, $\nu = 0.0$ and $\rho = 7850 kg/m^3$. The key result is the vertical displacement component of the free corner node as drawn below.

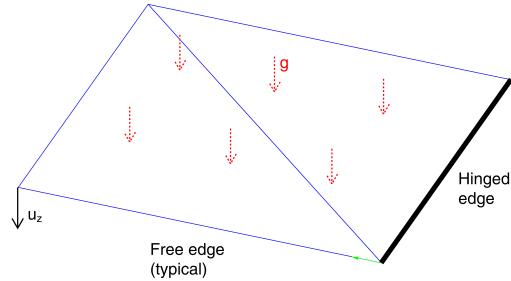


Figure 42 Shell pendulum definition

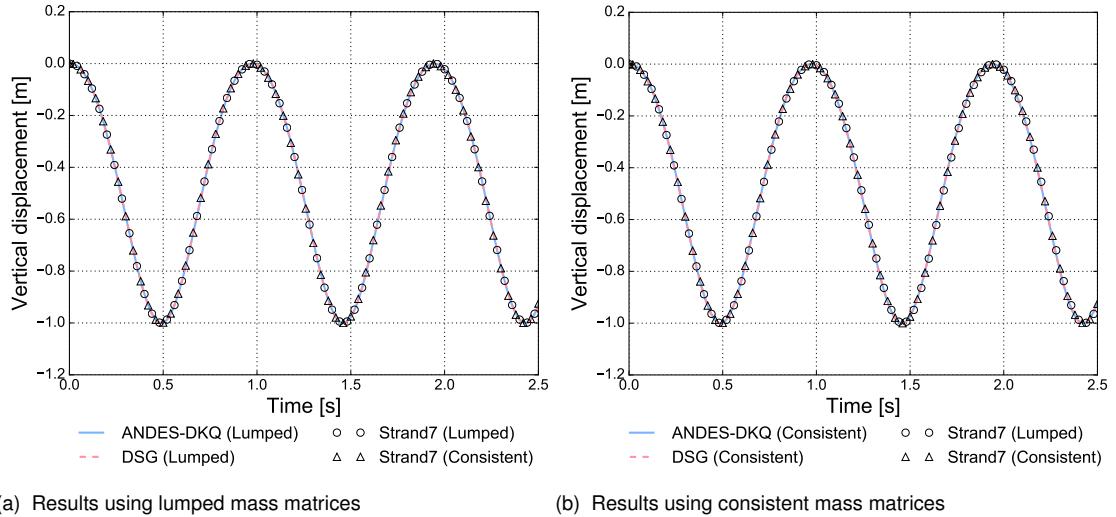


Figure 43 Vertical displacement over time of the shell pendulum analysis

The plot of displacement over time demonstrates the ability of both elements to handle large displacements and rotations, agreeing with the reference solution of the existing Kratos quadrilateral shell element. As expected, the minimum vertical displacement of $u_z = -1m$ corresponds to the position of bottom dead centre of the plate, while the maximum vertical displacement of $u_z = 0m$ corresponds to a fully horizontal plate orientation

8.3.2. Oscillating clamped plate

The oscillating clamped plate benchmark subjects a clamped cantilever square plate $2m \times 2m \times 0.1m$ thick to a uniform globally oriented surface pressure of $P_z = -0.25Pa$. The plate material is linear elastic characterised by $E = 1 \times 10^6 Pa$, $\nu = 0.0$ and $\rho = 7850kg/m^3$. The key result is the vertical displacement component of the free corner node as illustrated below.

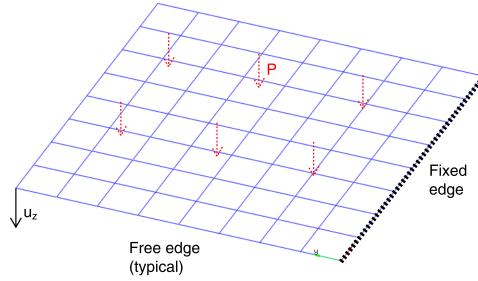
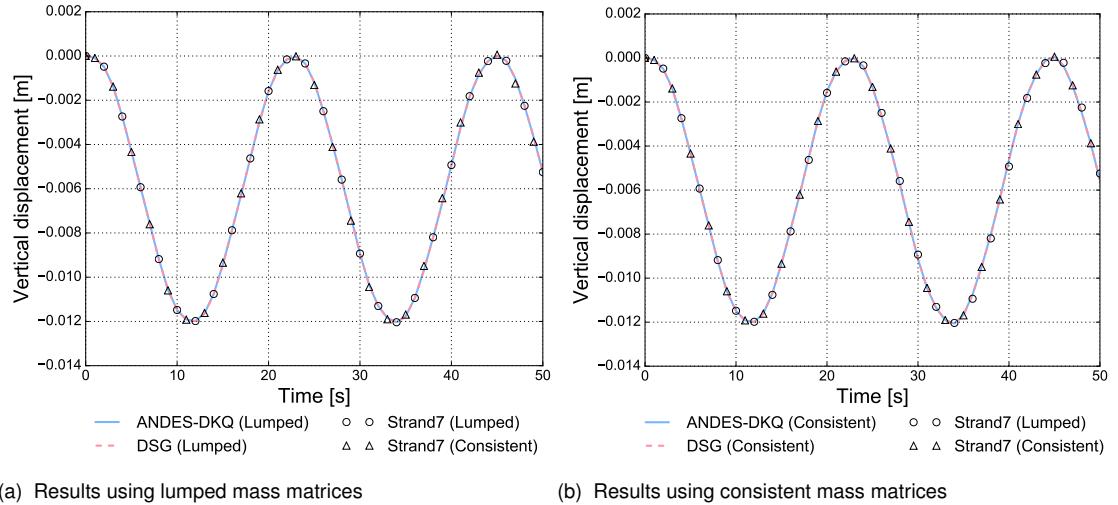


Figure 44 Oscillating clamped plate definition



(a) Results using lumped mass matrices

(b) Results using consistent mass matrices

Figure 45 Vertical displacement over time of the oscillating clamped plate analysis

The plot of vertical displacement over time demonstrates both elements agree with the reference solution, which is the existing Kratos quadrilateral element. The overall results correctly correspond to structural dynamics theory by oscillating with the base natural frequency about the static displacement of $u_z = -0.006m$.

8.4. Quantity recovery tests

Although displacements are the primary solution variables of a finite element analysis, recovered quantities, such as strains, stresses and force resultants are often more determinant to the success or failure of a system. The following tests validate the implemented elements ability to correctly recover these quantities.

8.4.1. Simply supported dome with oculus under self weight

A simply supported dome with an oculus under self weight is considered to evaluate the membrane results of the elements. The hemi-spherical dome is defined by the following parameters: $R = 5m$, $h = 0.01m$, $\rho = 7850kg/m^3$, $g = 9.81m/s^2$, $E = 2 \times 10^{11}Pa$, $\nu = 0.3$. The 20 degree opening has no edge loading. Appendix E derives the analytical formulae which form the reference solution for both force resultants, while an analysis of the problem with ANSYS provides the Von Mises stress reference solution.

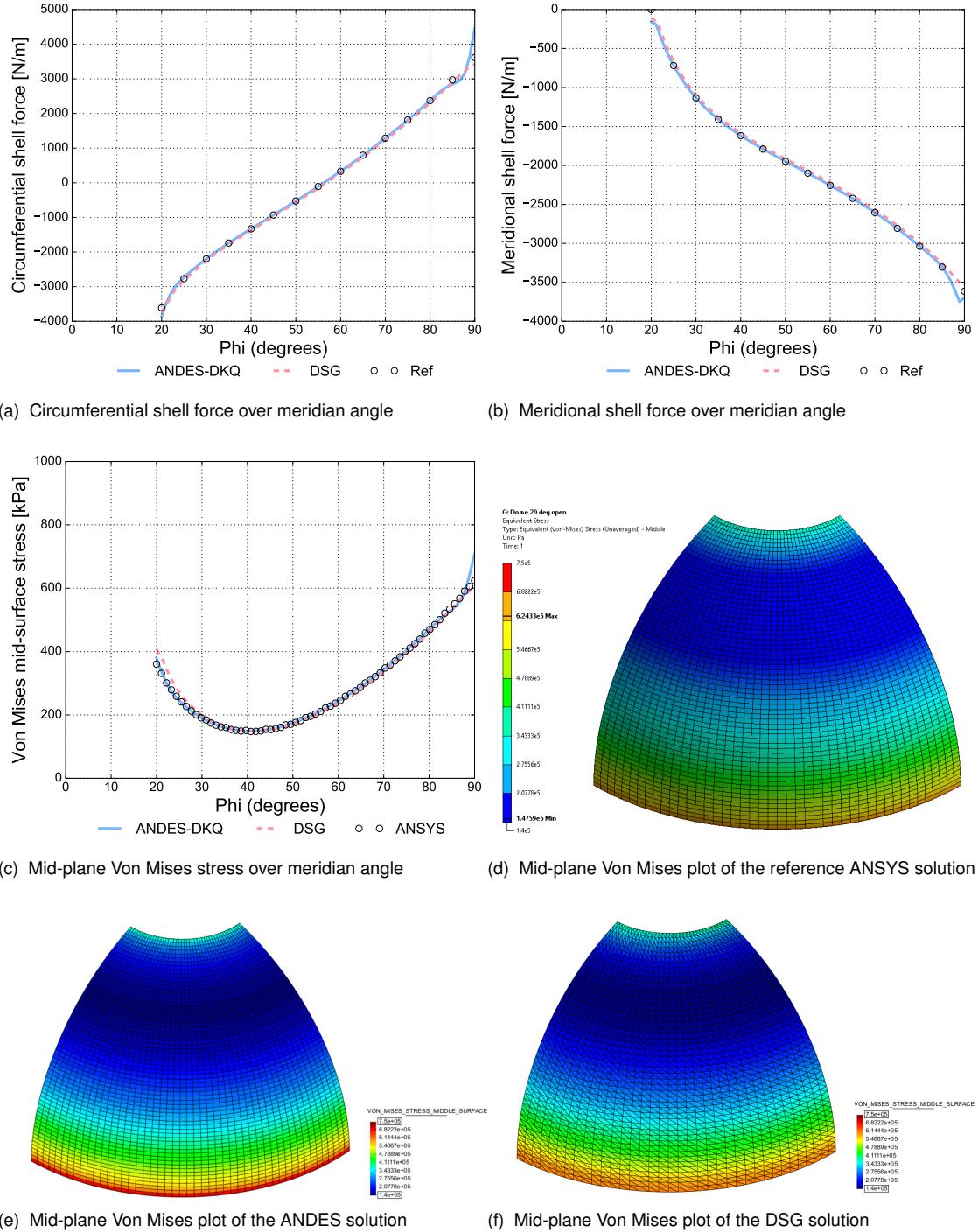


Figure 46 Results of the simply supported dome analysis

The results of both elements for circumferential and meridional shell forces demonstrate excellent agreement with the analytical membrane analysis. Minor deviations occur at both ends of the dome due to the strict assumptions of the analytical membrane theory and mesh effects. The mid-surface Von Mises stress results also agree with the ANSYS solution quite closely, with the ANDES-DKQ slightly out-performing the DSG element. The contour plots (with all limits set to $[1.4 \times 10^5, 7.5 \times 10^5]$) further exhibit the similarities between the three solution methods.

8.4.2. Navier supported plate under sinusoidal load

A Navier supported square plate subject to a sinusoidal load is considered to examine the bending stress results of the ANDES-DKQ element. A 3-parameter analytical solution forms the reference for this problem [44].

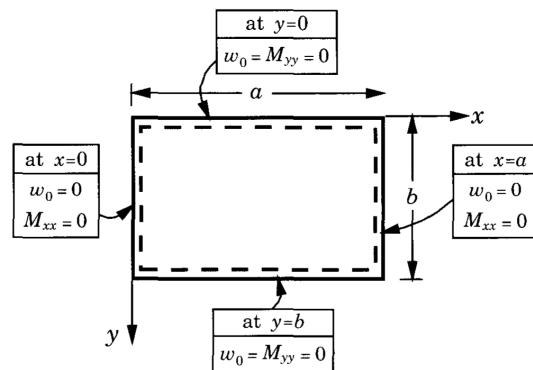


Figure 47 Definition of the Navier supported plate test [44]

Due to symmetry, one quarter of a $a = 200$, $b = 200$, $h = 10$ square plate of linear isotropic material $E = 2 \times 10^{11}$ and $\nu = 0.3$ is modelled and subject to a spatially varying transverse pressure of $q = -1 \times 10^7 \sin(\frac{x\pi}{200}) \sin(\frac{y\pi}{200})$. The following graphs plot quantities along a path from $x = 0$ to $x = 100$ at $y = 100$.

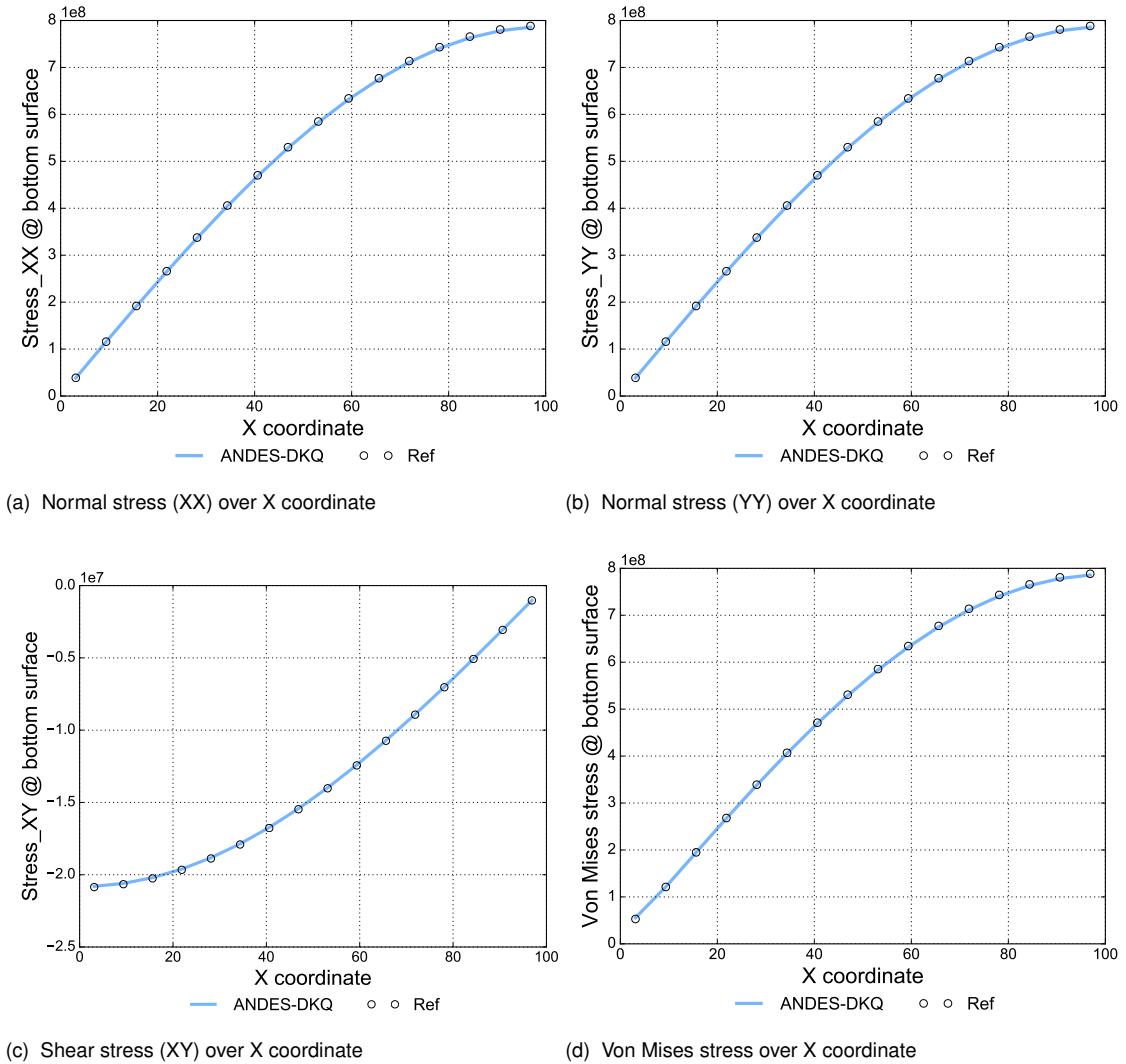


Figure 48 Stresses (bottom surface) of the Navier supported plate under sinusoidal load

The above figures demonstrate exemplary alignment between the analytical solution and the ANDES-DKQ element across all stresses considered.

8.4.3. Navier supported plate under uniformly distributed load

A Navier supported square plate subject to a uniformly distributed load is considered to examine the bending stress results of the DSG element. An ANSYS analysis forms the reference for this problem. The problem setup is identical to the preceding test with the exception of a uniformly distributed transverse pressure $q = -1 \times 10^{11}$ and "hard" supports along the edges ($\phi_x = 0$ on edges parallel to y-axis, $\phi_y = 0$ on edges parallel to x-axis).

Graphs plotted along an X coordinate follow the same path as the preceding tests. Graphs plotted along a Y coordinate run from $y = 0$ to $y = 100$ with $x = 100$, while graphs plotted along a diagonal distance run from $x = y = 0$ to $x = y = 100$.

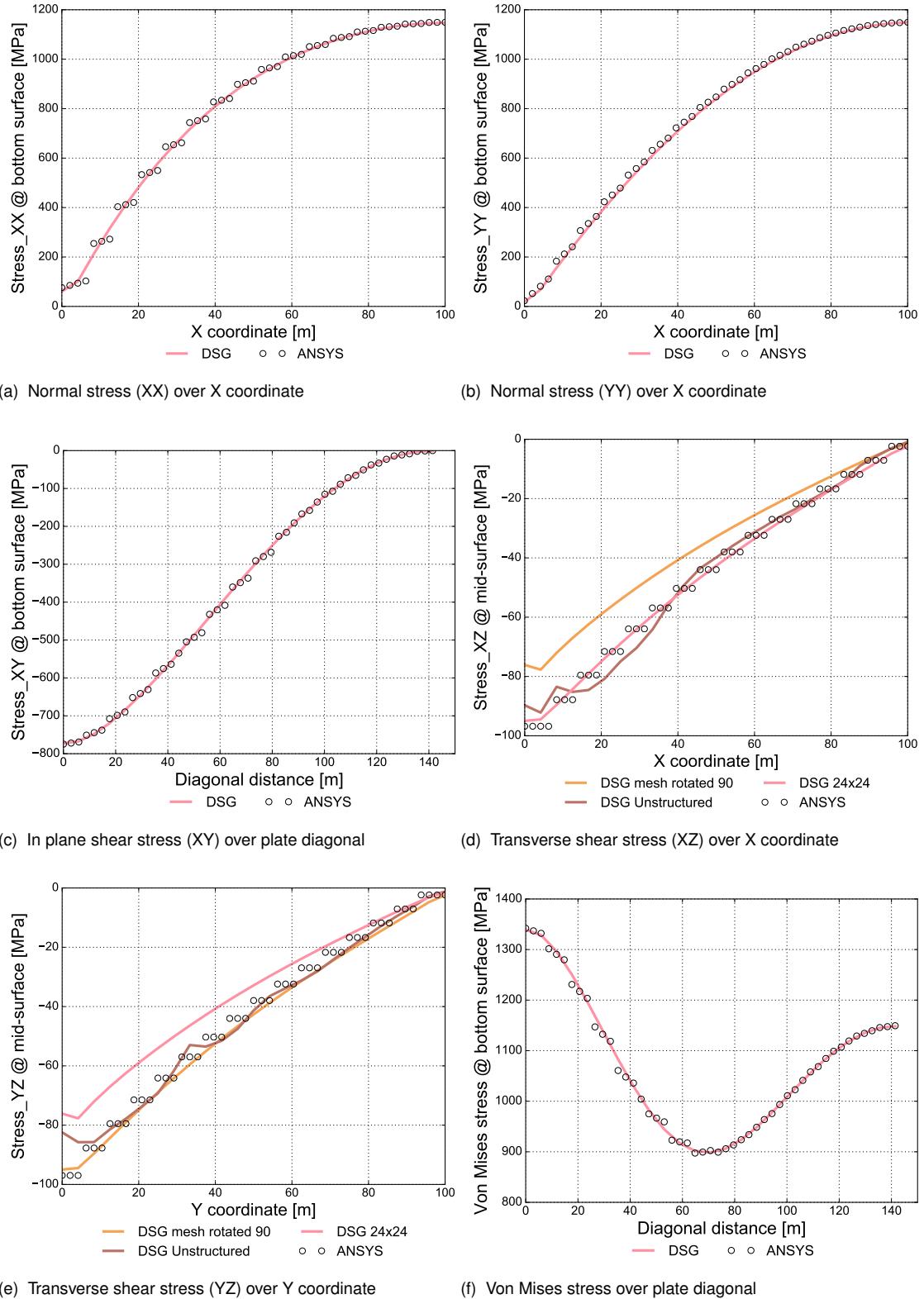


Figure 49 Stresses of the Navier supported plate under uniformly distributed load

The DSG element shows good agreement with the ANSYS results across all stress quantities considered.

8.5. Composite tests

The extension of both shell elements to handle composite laminates requires its own validation across the range of tests considered for isotropic shells. The following tests consider the performance of both elements in cases which all employ composite materials. As per the preceding isotropic element tests, cases across linear/non-linear statics, linear/non-linear dynamics, and quantity recovery are examined.

8.5.1. Composite linear statics: composite barrel vault

The composite barrel vault is an extension of the Scordelis-Lo roof obstacle course test to composite materials [44]. Corresponding to the following figure, the problem is defined with: $\alpha = 40^\circ$, $R = 300$, $a = 600$ and $q = 0.625$. The reference quantity is the vertical displacement at the centre of the roof (point A).

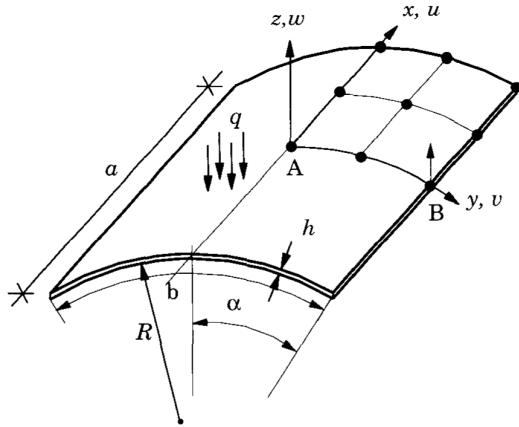


Figure 50 Definition of the composite barrel vault test [44]

The composite material is considered with cross-ply [0/90/0/90...] and angle-ply [-45/45/-45/45/...] lamina stacking arrangements, with each ply having the following material properties: $E_1 = 25E_2$, $E_2 = 1.2 \times 10^5$, $G_{12} = G_{13} = 0.5E_2$, $G_{23} = 0.2E_2$ and $\nu_{12} = 0.25$. Furthermore, the tests are carried out over slenderness ratios of $S = \frac{R}{h} = 20, 50, 100$ which correspond to total laminate thicknesses of $h = 15, 6, 3$.

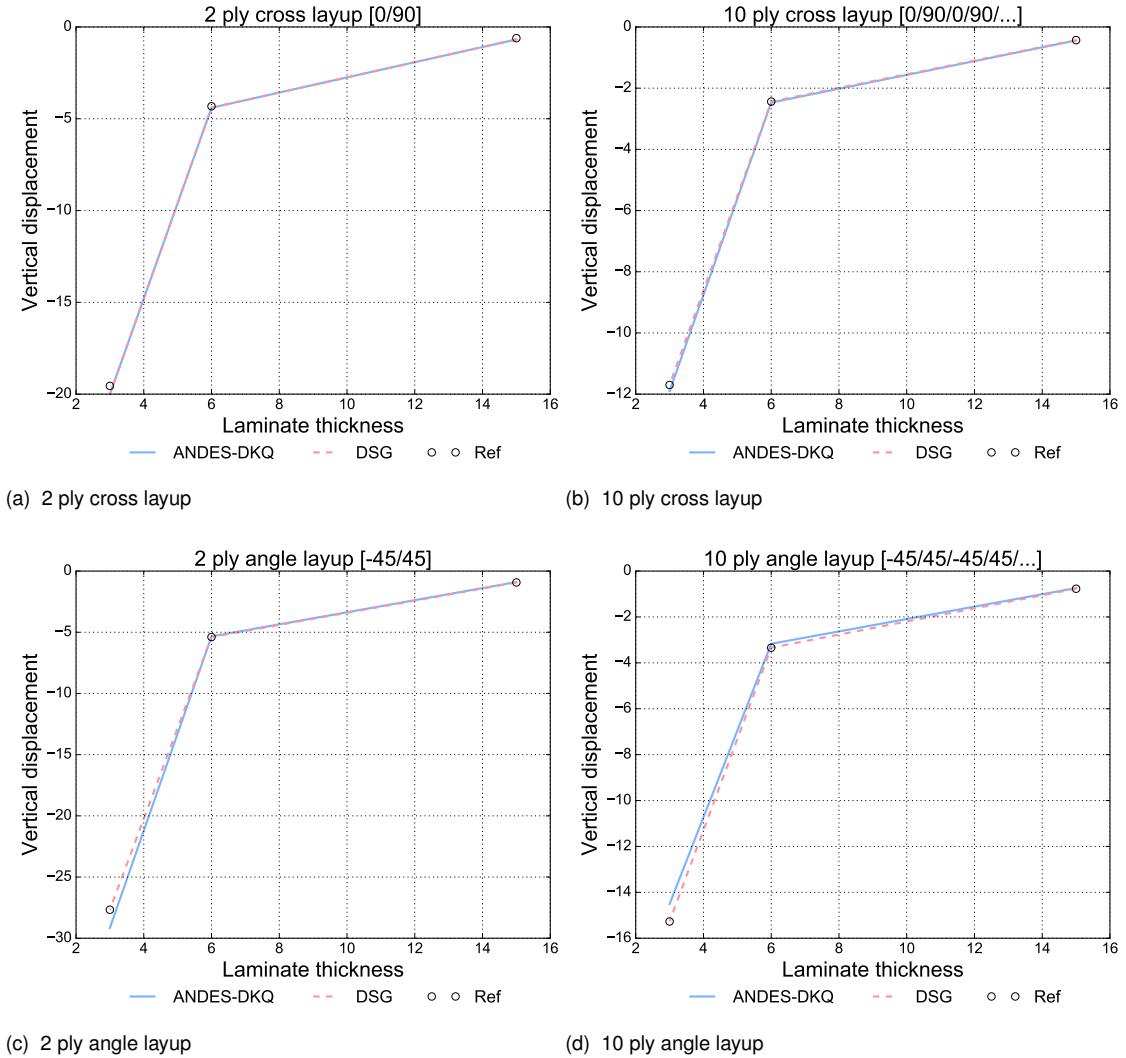


Figure 51 Composite barrel vault test results

Across the wide range of testing considered both elements exhibit excellent agreement with the reference solution. The ANDES-DKQ element performance slightly deteriorates in the thin angle layup scenarios, however, the result is still quite accurate.

8.5.2. Composite linear static: clamped cylinder

Another test for the composite shell elements in the realm of linear statics is a cylinder clamped at both ends subject to internal pressure. A cylinder of length $a = 20$, radius $R = 20$ and total laminate thickness of $h = 1$ is subject to a uniform internal pressure of $p_0 = \frac{6410}{\pi}$ [44]. The key quantity of interest is the maximum radial displacement of the cylinder, with the reference solution taken from Reference [44].

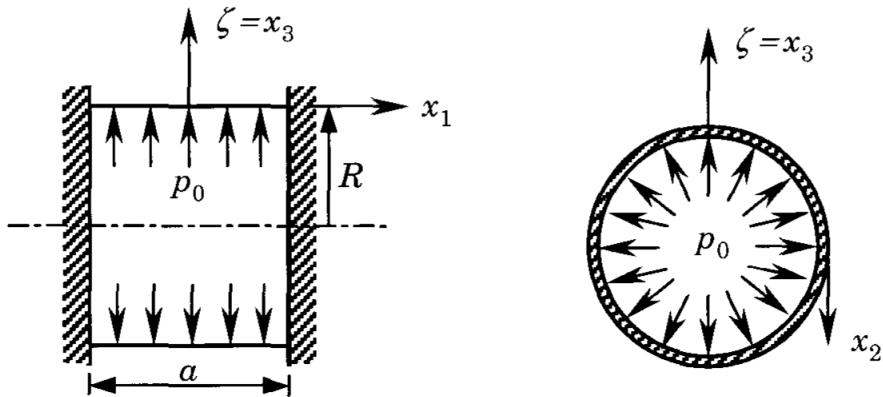


Figure 52 Composite clamped cylinder test definition

The laminate is considered in both single and double layer arrangements, with the lamina properties defined as: $E_1 = 7.5 \times 10^6$, $E_2 = 2 \times 10^6$, $G_{12} = 1.25 \times 10^6$, $G_{13} = G_{23} = 0.625 \times 10^6$ and $\nu_{12} = 0.25$. Due to symmetry only half the cylinder was modelled, while the mesh was refined under the constraint of *circumferential divisions* = $1.5 \times$ *axial divisions*.

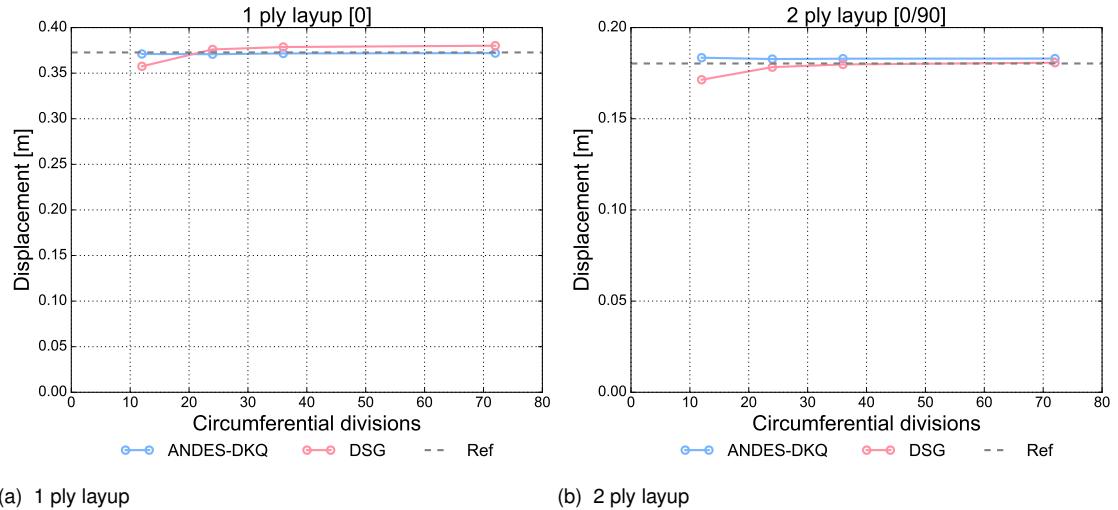


Figure 53 Composite clamped cylinder analysis convergence

Both elements quickly converge to the reference solution and maintain good performance as the mesh is refined further. The reference solution source [44] notes that minor differences are expected between 3-parameter and 5-parameter shell models, however, the vagaries between the ANDES-DKQ and DSG elements are negligible compared to their excellent accuracy.

8.5.3. Composite non-linear static: composite hinged cylindrical roof

To consider the non-linear static performance of composite shells, the hinged cylindrical roof test in section 8.2.1 is identically repeated with a laminate material. The composite laminate

of total thickness $h = 12.7\text{mm}$ is composed of laminae with properties $E_1 = 3.3 \times 10^9$, 1.1×10^9 , $G_{12} = G_{13} = G_{23} = 0.66 \times 10^9$ and $\nu_{12} = 0.25$ arranged in a [90/0/90] stacking sequence [46].

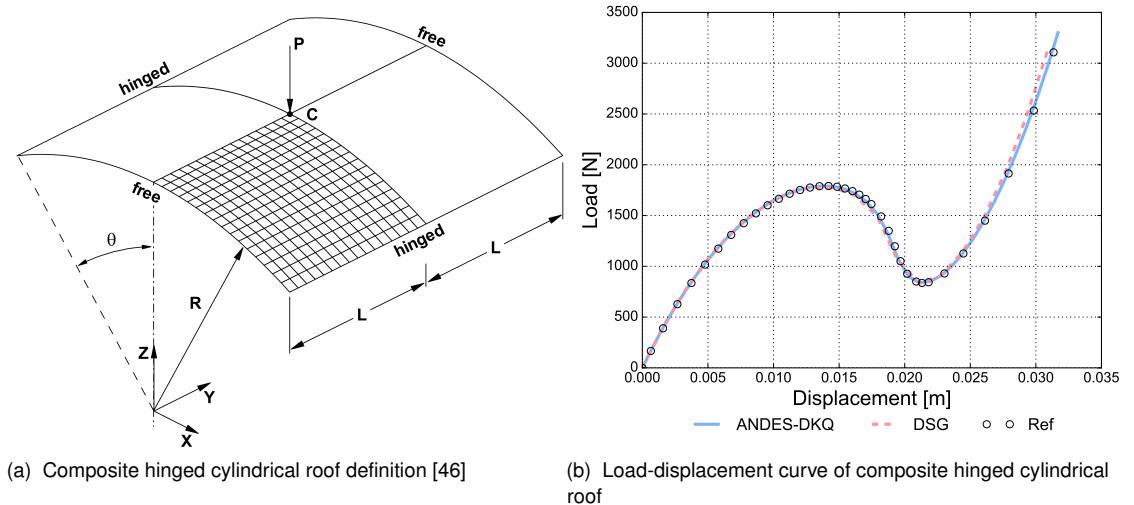


Figure 54 Composite hinged cylindrical roof test

The load displacement curve above clearly shows the ability of both elements to accurately resolve the equilibrium path of the composite roof through the pre and post-critical ranges.

8.5.4. Composite dynamics: shell pendulum and oscillating square plate - FIXUP SECOND GRAPH FONT SIZE

The shell pendulum test originally presented in section 8.3.1 is identically repeated for a composite laminate material. A laminate of total thickness $h = 0.1\text{m}$ is considered with a stacking sequence of [0/90/90/0]. The material properties of each lamina are as per those defined in the composite barrel vault test (section 8.5.1). Since the general lumped and consistent mass matrices were verified in section 8.3.1, only the lumped case is considered in this test to validate the correct integration of composite material in dynamics. The reference solution for this test is that obtained by Strand7 with a lumped mass matrix.

A second composite dynamic test considers the square Navier supported plate described in section 8.4.3 subject to a uniform transverse pressure of $q = 1$. The material is changed from an isotropic material to a $h = 2$ thick laminate of layup [0/90]₄. The lamina properties are as per those defined in section 8.5.1 with the exception of $\rho = 1000$. The reference solution for this test is that obtained by Strand7 with a lumped mass matrix.

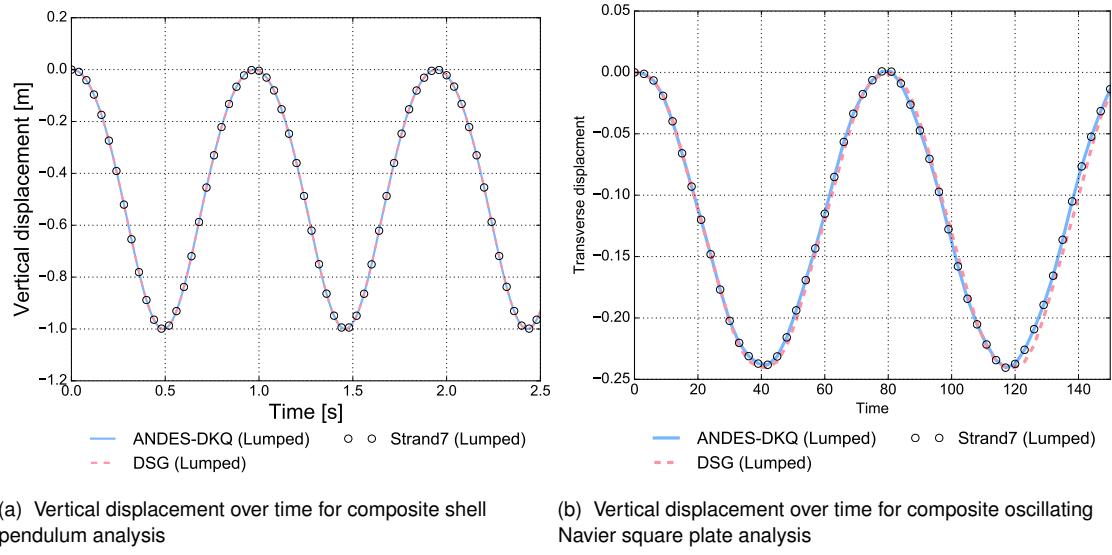


Figure 55 Composite dynamic tests

As per the isotropic dynamic tests, the composite tests agree very well with the reference Strand7 solutions. Figure 55(b) reveals that the DSG element slightly deviates as the simulated time accumulates, however, the error in period is less than 1.3%.

8.5.5. Composite stress recovery: tensile test

The first test to interrogate the stress recovery accuracy of the composite shell elements is a relatively simple tensile test [36]. A horizontal rectangular tensile test token of $10 \times 2 \times 0.02$ total thickness is clamped at one end while the other end is subject to a tensile load of 1000. The token laminate layup is [0/90/90/0] while the material properties of each lamina are defined as: $E_1 = 200.1 \times 10^5$, $E_2 = 130.1 \times 10^4$, $G_{12} = G_{13} = G_{23} = 100.1 \times 10^4$ and $\nu_{12} = 0.3$. Results of the analysis are taken from the mid-length ($x = 5$) of the test token and are averaged across the width. The reference solution for the stresses is from Reference [36] with a Strand7 analysis of the problem providing the Tsai-Wu reference.

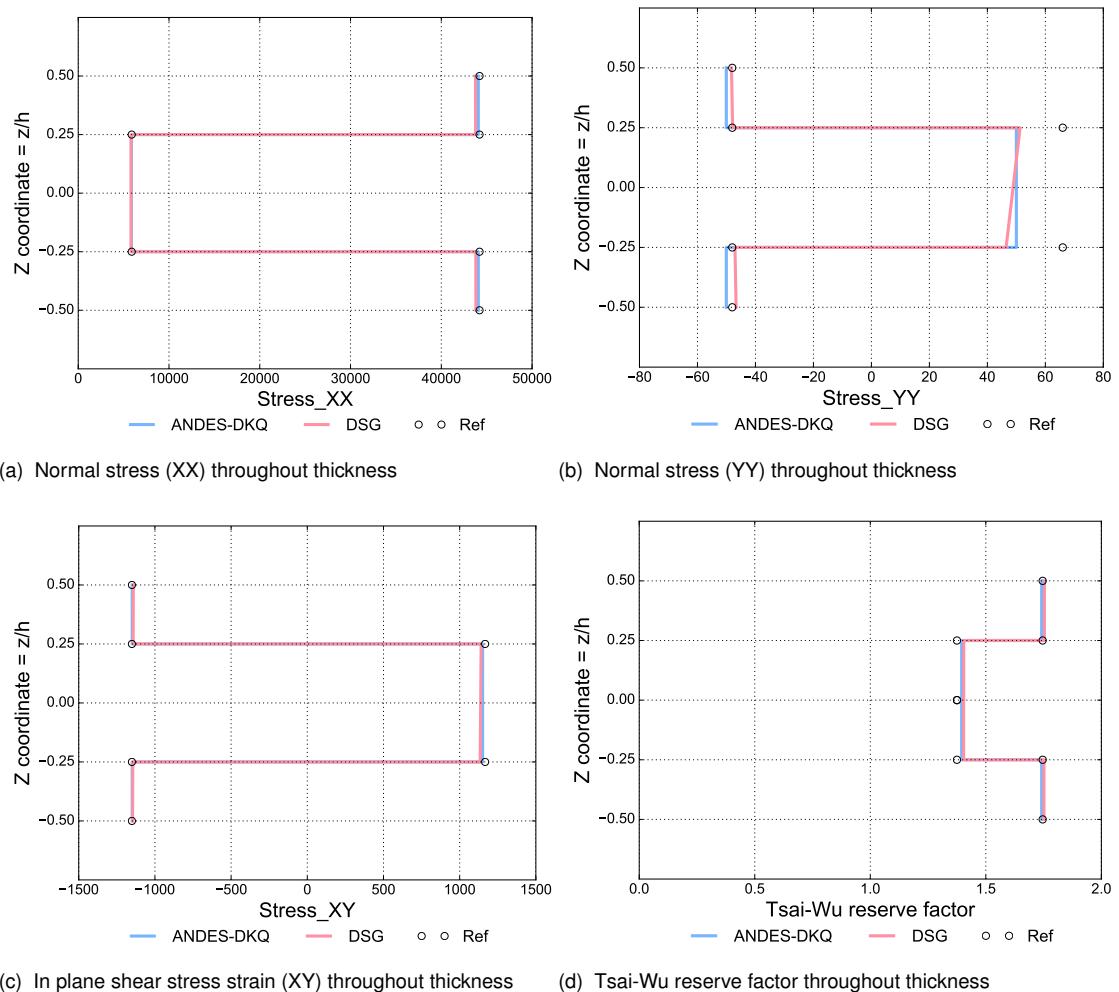


Figure 56 Stresses and Tsai-Wu results of the composite tensile test

The four figures above, illustrating various stresses and Tsai-Wu reserve factor throughout the laminate thickness, confirm the accurate stress recovery of both elements. Although the recovery of Stress_{YY} may initially not seem as accurate as the others, its scale is 3 orders of magnitude less than the primary stress mode Stress_{XX}. It's expected with stress values so low that much of this acceptable deviation is due to rounding errors, both in the implemented elements and the reference literature itself. The Tsai-Wu reserve factor results of both elements are in excellent agreement with the reference values.

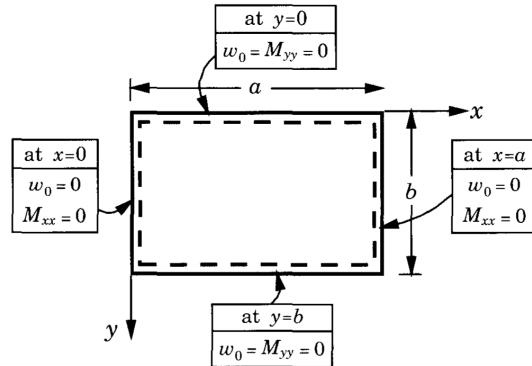
8.5.6. Composite stress recovery: Navier supported laminate under sinusoidal load

The second composite stress recovery test considers the bending dominated scenario of a Navier supported square laminate subject to a sinusoidal load [44]. As per the isotropic stress recovery testing, separate tests were carried out for the two elements: the ANDES-DKQ composite test is based off of section 8.4.2 and the DSG from section 8.4.3. Both analyses have a transverse sinusoidally distributed transverse pressure defined as $q = -100\sin(\frac{x\pi}{200})\sin(\frac{y\pi}{200})$

and consider a $h = 10$ thick [0/90/90/0] laminate with individual lamina properties as per those defined in 8.5.1.

The reference solutions are taken from Reference [44], with the ANDES-DKQ element compared to a classical plate theory solution and the DSG element compared to a Navier plate closed form solution. As per the reference solutions, results for different quantities are sampled at different positions in the square plate. Referring to figure 47 (repeated below for clarity), the stress and strains are recovered throughout the thickness of the laminate at the following positions:

$$\begin{aligned}\sigma_{xx} &= \sigma_{xx}\left(\frac{a}{2}, \frac{b}{2}, z\right) & \epsilon_{xx} &= \epsilon_{xx}\left(\frac{a}{2}, \frac{b}{2}, z\right) \\ \sigma_{yy} &= \sigma_{yy}\left(\frac{a}{2}, \frac{b}{2}, z\right) & \epsilon_{yy} &= \epsilon_{yy}\left(\frac{a}{2}, \frac{b}{2}, z\right) \\ \sigma_{xy} &= \sigma_{xy}(0, 0, z) & \epsilon_{xy} &= \epsilon_{xy}(0, 0, z) \\ \sigma_{xz} &= \sigma_{xz}(0, \frac{b}{2}, z) & \epsilon_{xz} &= \epsilon_{xz}(0, \frac{b}{2}, z) \\ \sigma_{yz} &= \sigma_{yz}\left(\frac{a}{2}, 0, z\right) & \epsilon_{yz} &= \epsilon_{yz}\left(\frac{a}{2}, 0, z\right)\end{aligned}$$



The following set of graphs illustrate the results of the ANDES-DKQ element compared with the reference classical plate theory solution.

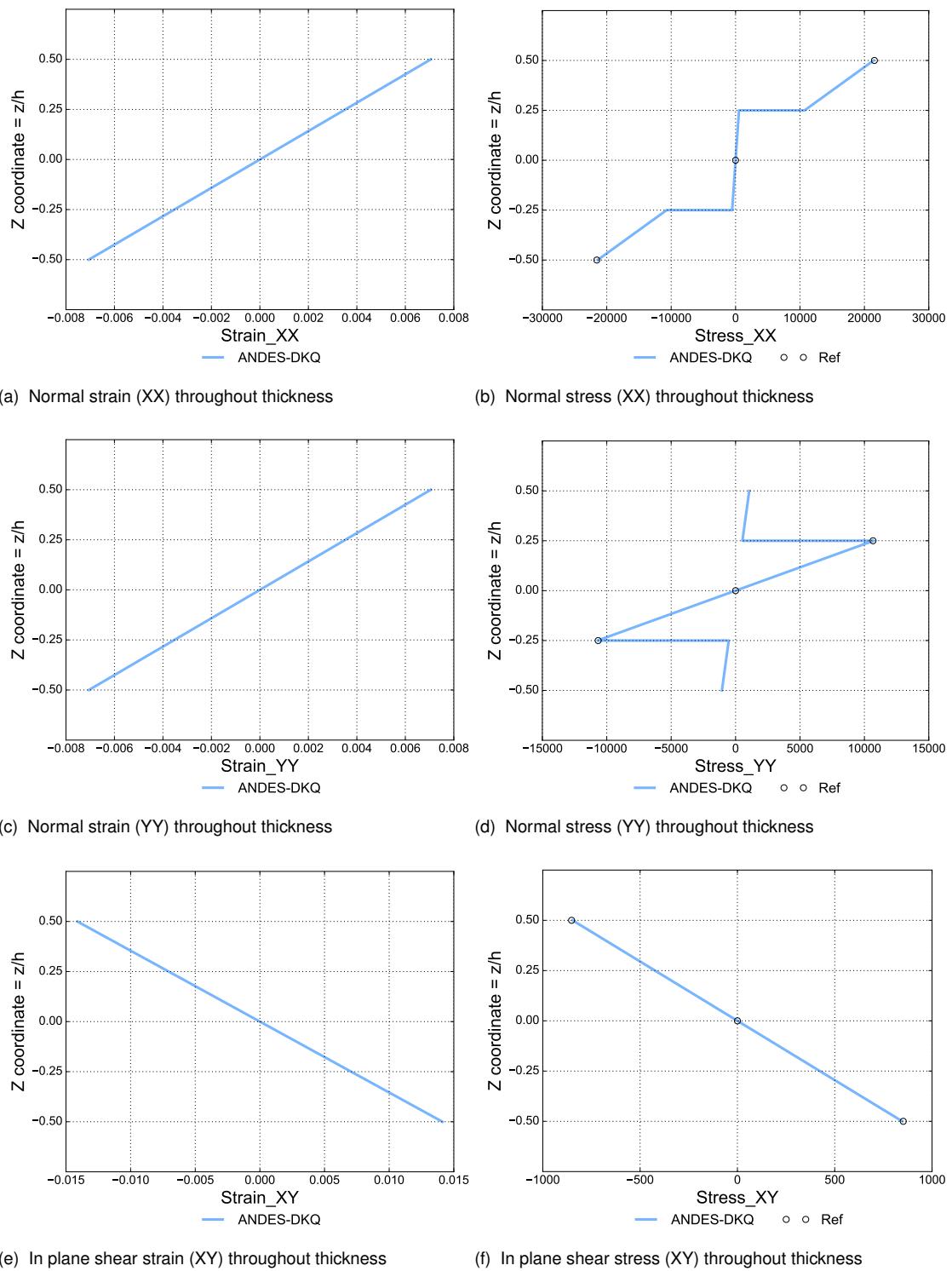


Figure 57 Stresses and strains of the Navier supported ANDES-DKQ laminate under a sinusoidally distributed load

The following set of graphs compare the DSG element results with those of the reference closed form solution.

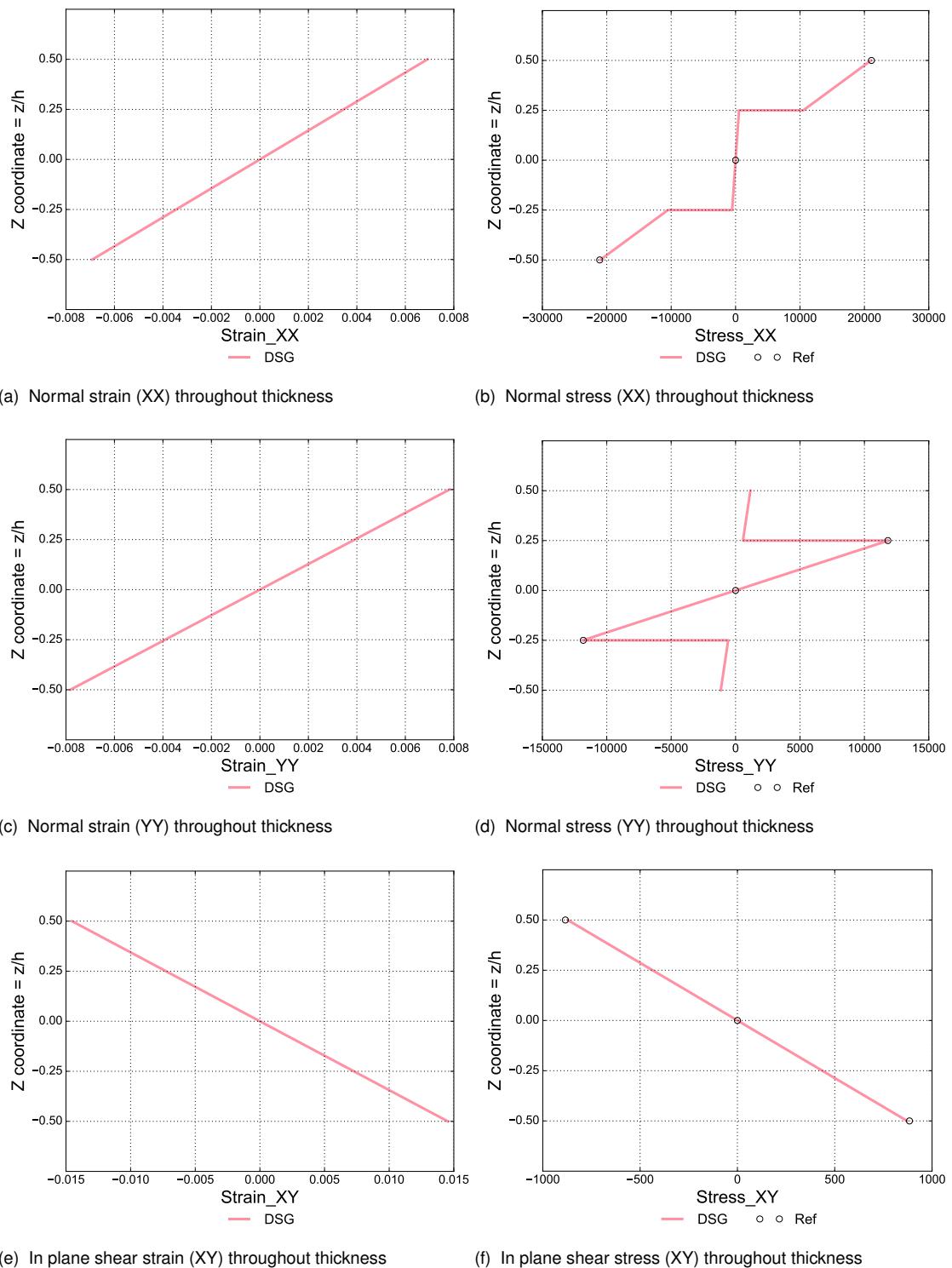


Figure 58 In plane stresses and strains of the Navier supported DSG laminate under a sinusoidally distributed load

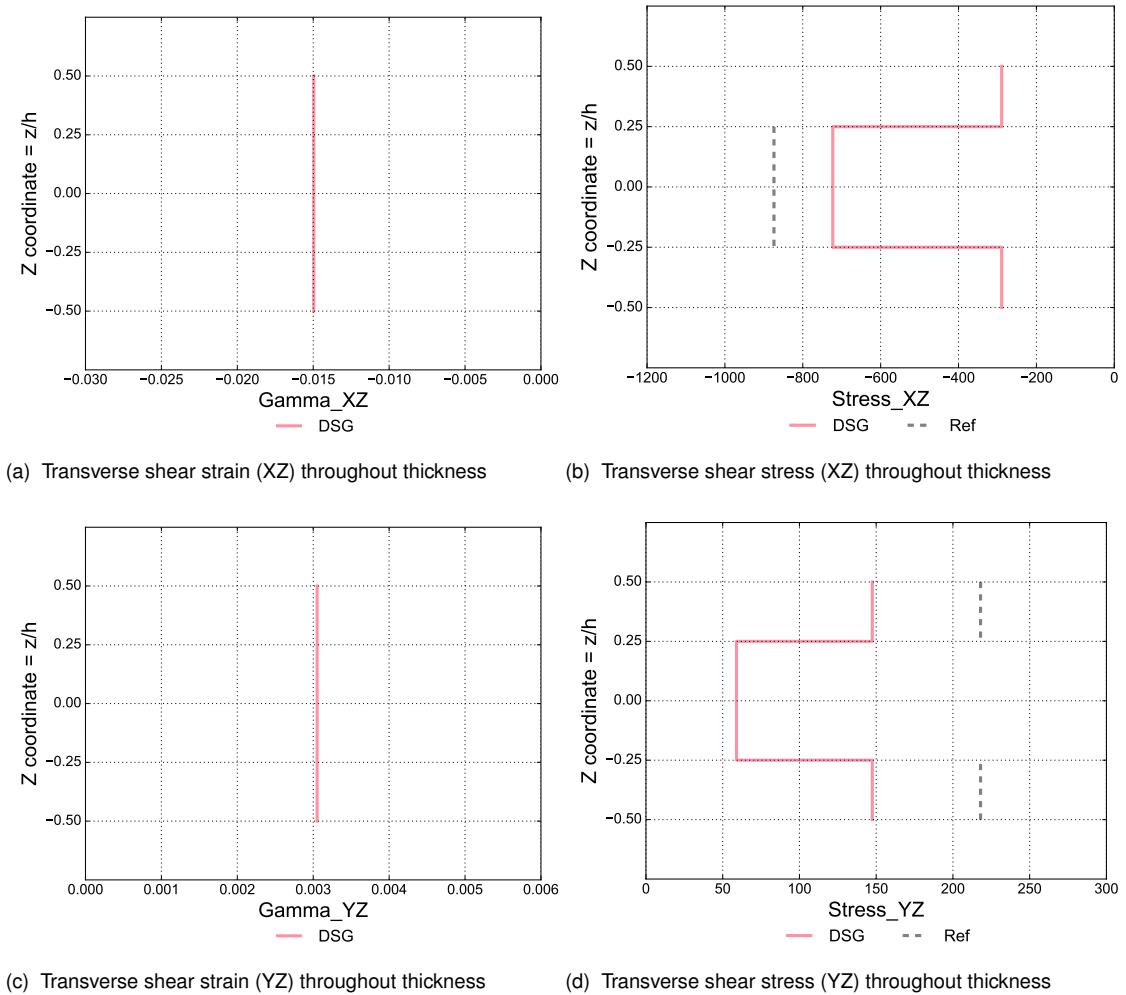


Figure 59 Transverse shear stresses and strains of the Navier supported DSG laminate under a sinusoidally distributed load

The results of the ANDES-DKQ and DSG elements demonstrate the excellent composite stress recovery accuracy across all stresses considered. It also highlights the ability to resolve discontinuous stress distributions from smooth linear strains by considering properly rotated individual lamina material properties. As expected, the in-plane shear stresses (Stress_XY) are smoothly linear because the in-plane shear modulus is invariant under planar rotation. The transverse shear stresses of the DSG are slightly less accurate than other stresses recovered, which is likely due to a combination of mild node-numbering dependency (as per the transverse shear stresses of section 8.4.3) and rounding having a pronounced effect on the comparatively low order-of-magnitude values.

In addition to the stresses and strains considered, the Tsai-Wu reserve factor was also calculated in the centre of the plate ($\frac{a}{2}, \frac{b}{2}, z$) and compared against Tsai-Wu results from a Strand7 analysis of the problem. It must be noted that the Strand7 analysis software approximates the Tsai-Wu in-plane interaction coefficient F_{12} (refer section 3.4) as zero. Thus, to enable a proper comparison, results marked with an asterisk '*' have $F_{12} = 0$, while those

without calculate F_{12} as per equation 3.35.

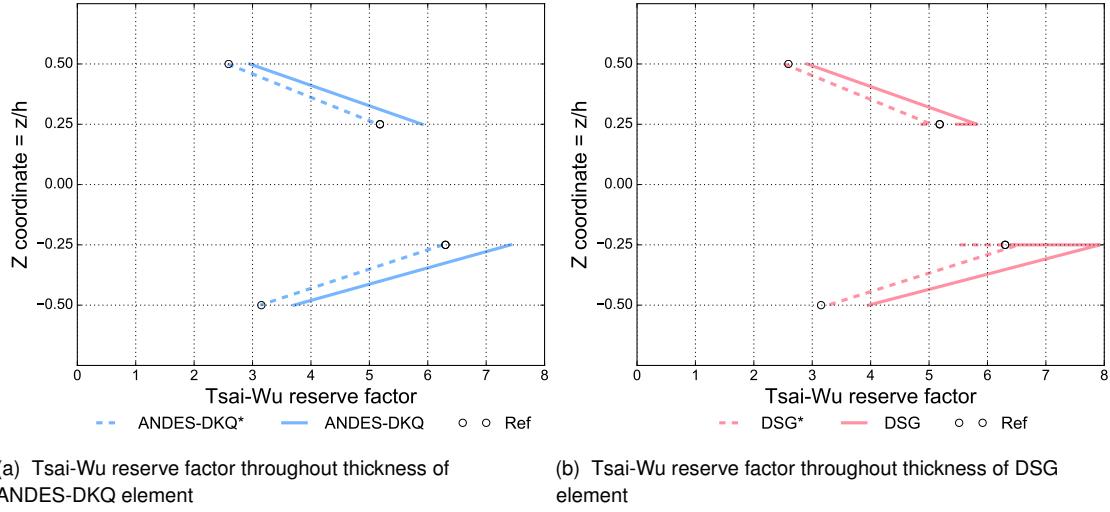


Figure 60 Tsai-Wu reserve factor through the thickness of the Navier supported DSG laminate under a sinusoidally distributed load

The figures above confirm accurate determination of the Tsai-Wu reserve factor in both elements. A central discontinuity is present in the ANDES-DKQ thin shell because it has no stress on the mid-plane under pure bending, yielding an infinite reserve factor. Focussing on the DSG element, the low magnitude of transverse shear stresses on the mid-plane yields a finite reserve factor of roughly 100, which has been omitted for clarity. These transverse shear stresses, which have a greater combined magnitude in the central laminae, are also the reason behind the small 'hooks' of lower reserve factors at the outer/inner ply boundaries.

Chapter 9 Applications and consequences of element technologies

WITH the elements formulation, background and validation complete, some examples of their application are now considered. An emphasis is placed on the relative strengths and weaknesses of each element and the effect of their enhancements by considering six elements for each analysis: ANDES-DKQ, Basic-DKQ, Kratos-Q4, DSG, Basic-T3, Kratos-T3.

9.1. Euler buckling of CHS column

The first application considered is the classic Euler buckling of a slender beam $L = 3m$ with a Circular Hollow Section (CHS) of $D = 300mm, t = 5mm$ subject to an axial compressive load P . Young's modulus and Poisson's ratio are $E = 206.9GPa$ and $\nu = 0.0$ respectively. The following diagram highlights the system setup with the end restraints imposed corresponding to an Euler case 4 buckling regime.

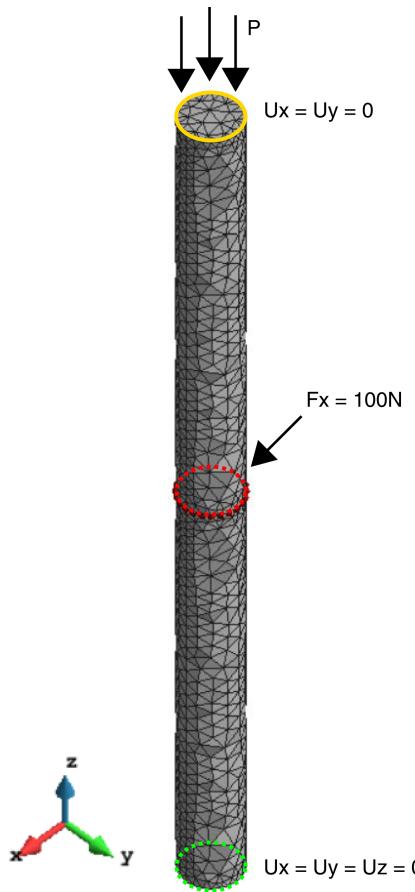


Figure 61 CHS buckling setup

A full derivation of the Euler case 4 buckling load using beam theory is presented in Appendix F in which the critical load corresponding to the first eigenvalue is:

$$P_{crit} = \frac{4\pi^2 EI}{L^2} = 13,222kN \quad (9.1)$$

Due to the unpredictable nature of solving bifurcation problems with FEM a small side load $F_x = 0.1kN$ is added to a horizontal ring of nodes at the beam's mid-span to encourage switching to the secondary equilibrium path defined by buckling in the XZ plane. Without this side load, or some other reliable source of imperfection, it is possible that the FE model would continue along the primary equilibrium path after the first critical point, instead of switching to a secondary path of buckling. Although the unstructured meshes used may provide enough asymmetry to act as a buckling catalyst, the severity of the imbalance would no doubt change from triangular to quadrilateral unstructured meshes, which may affect the results. Conversely, it's supposed that the applied small lateral load provides a consistent source of imperfection of greater magnitude than the underlying mesh imbalances, thereby allowing an apples to apples comparison between triangular and quadrilateral meshes.

The first set of results for the analysis highlighting axial displacement (of the top end) vs axial load P is presented below, with circles indicating the onset of instability. Post-critical behaviour has been omitted for clarity.

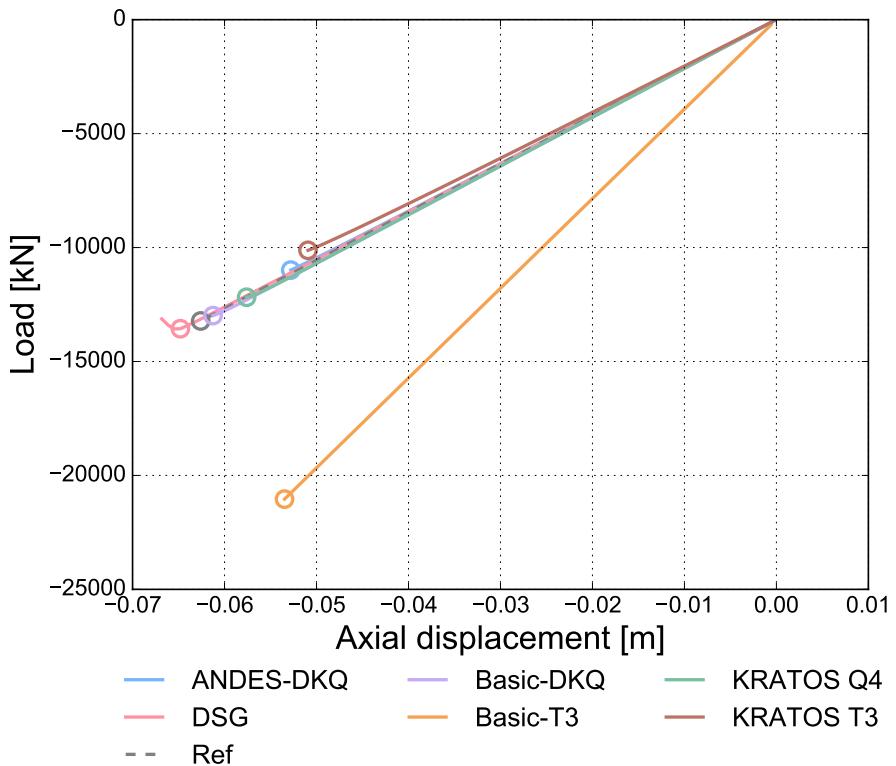


Figure 62 CHS buckling: axial displacement vs axial load

The results above highlight the poor performance of the un-enhanced Basic-T3 element, with its large over-estimation of the critical load a symptom of severe transverse shear locking. Accordingly, this poor performance also demonstrates the efficacy of the DSG element enhancements with the DSG results aligning quite well with the remaining elements and the reference beam theory solution (which has a first order displacement calculated from an axial stiffness of EA/L). The Basic-DKQ element is also quite close to the reference solution most likely because transverse shear locking is mitigated by the DKQ bending formulation while the effect of membrane locking (which this element is susceptible to) is minimized by the fine mesh employed (due to reduced element out of plane warping). Despite this, differences do indeed appear between the Basic-DKQ and ANDES-DKQ elements, the latter of which suggests a lower buckling limit along with the Kratos-T3 thin shell element. To gain more insight into this difference between these two elements and the rest, the axial load is plot against the lateral X-displacement taken at the beam mid-point ($x, y, z) = (D/2, 0, L/2)$.

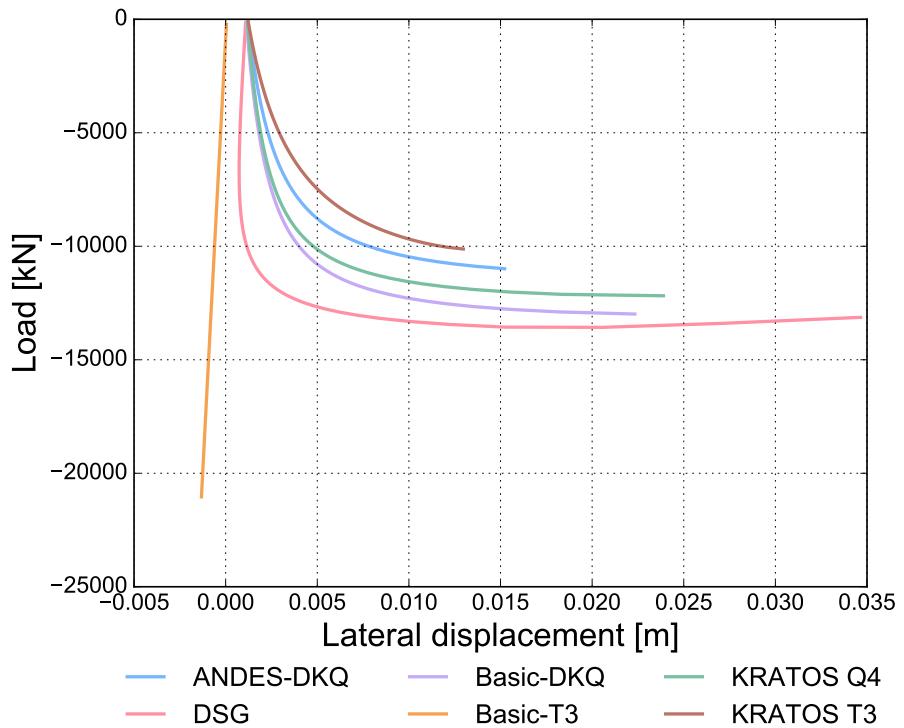


Figure 63 CHS buckling: lateral displacement vs axial load

The alternative perspective presented above confirms the exceedingly poor performance of the Basic-T3 element once again. As previously identified, the ANDES-DKQ and Kratos-T3 elements predict low buckling loads for the system, however the plot above indicates that these are attached to reduced lateral displacements too (compared to the Kratos-Q4, Basic-DKQ and DSG elements). Clearly the various reasonable elements are expressing the system behaviour in different ways with the greatest difference occurring between the Kratos-T3 and DSG elements. To gain further insight into the behaviour of these two envelope cases, the following figures show X and Y displacements of the Kratos-T3 and DSG models at the onset of instability plot at 6.7x deformation with the same contour limits.

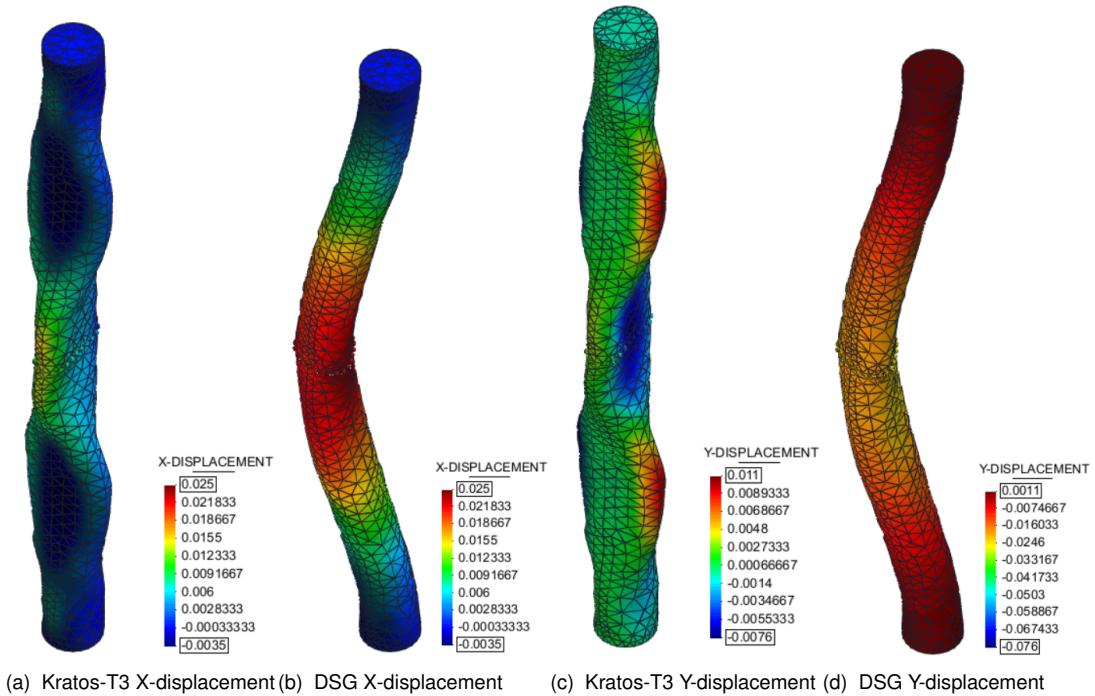


Figure 64 CHS buckling: Kratos-T3 and DSG displacement plots at the onset of instability

The plots above demonstrate the strikingly different structural behaviours modelled by the two elements using exactly the mesh, boundary conditions and loading conditions. Local ovalisation of the CHS appears to drive buckling in the Kratos-T3 case, whereas the DSG element predominantly maintains the circular cross section throughout and exhibits deformation one would expect from a beam element, explaining the close similarity between the DSG and reference beam results. The differences between the Kratos-T3 (3-parameter) and DSG (5-parameter) elements can be explained by their underlying formulation and the resolving power each possesses. Despite the slenderness ratio $R/t = 20$ of the problem being suitable for both thin and thick shell use, it's intuitive that at this point the 5-parameter model would be more reluctant than the 3-parameter model to predict out of plane bending behaviour, leading to ovalisation. Furthermore, the DSG element is computed with a single Gauss Point while the Kratos-T3 element is computed with 3 Gauss Points which confers a relative advantage in resolving complex local displacement fields (such as local ovalisation).

For completeness, the X and Y displacements of the ANDES-DKQ and Basic-DKQ models at the onset of instability are also plotted below with 6.7x deformation and equilibrated contour limits.

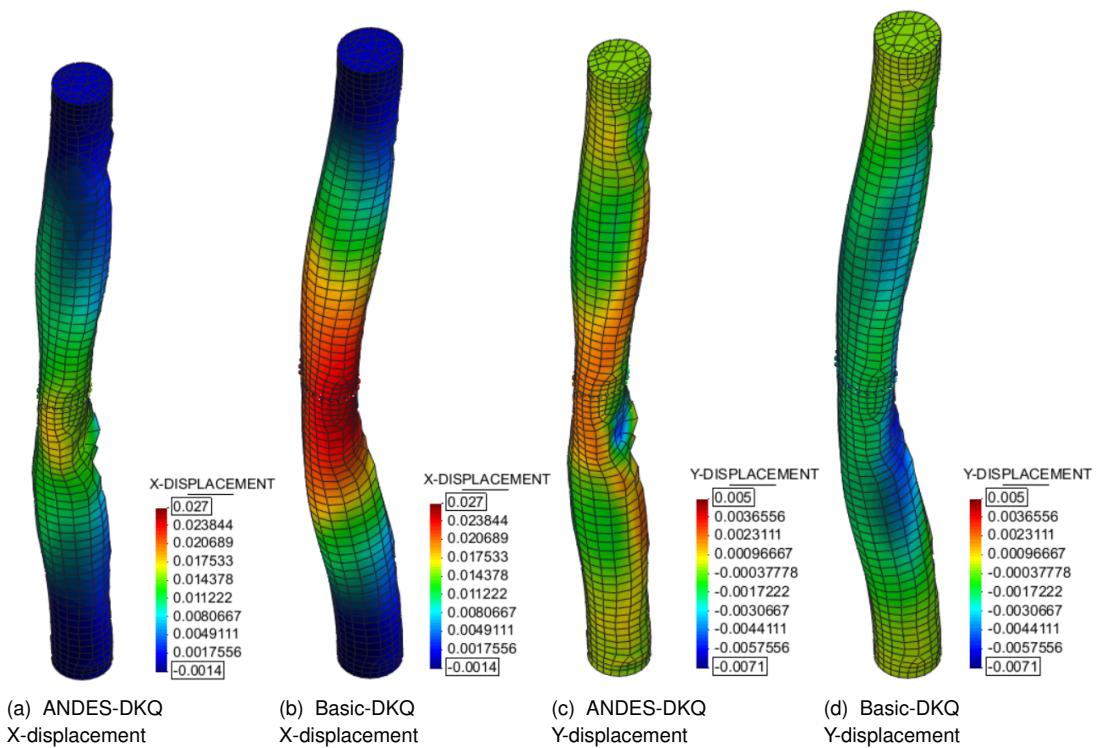
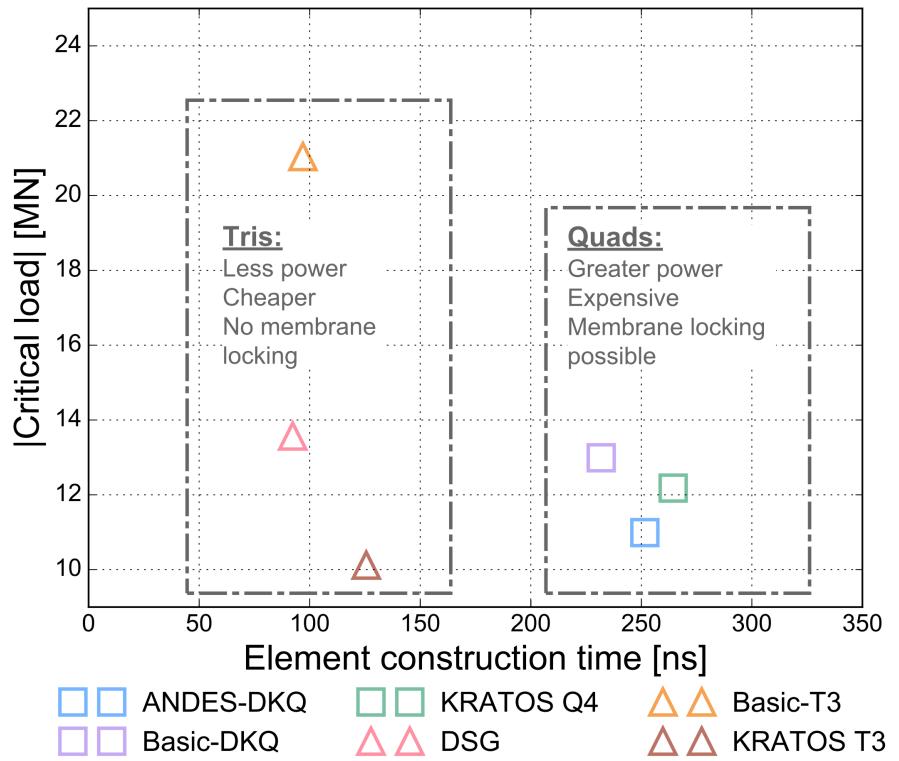
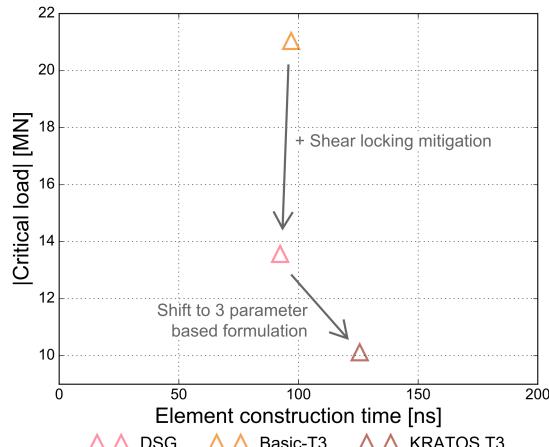


Figure 65 CHS buckling: ANDES-DKQ and Basic-DKQ displacement plots at the onset of instability

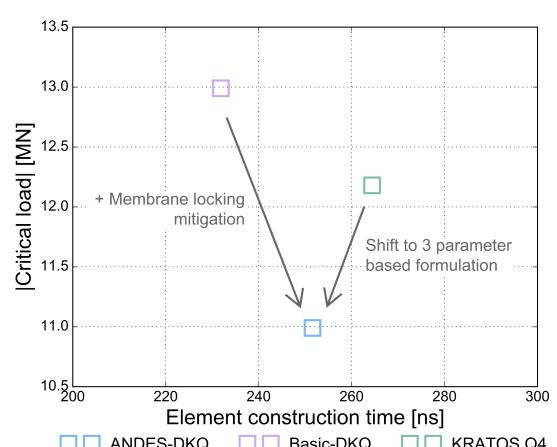
In this comparison, the differences in deformation behaviour is solely due to the ANDES enhancement that mitigates membrane locking. An unstructured quadrilateral mesh of a curved surface, such as above, is a quintessential situation for membrane locking to arise. It can be seen that the ANDES-DKQ case exhibits ovalisation of the cross section as expected by its similarity to the aforementioned Kratos-T3 results. Contrasting this, the Basic-DKQ results display an incredibly minor amount of ovalisation, reduced from its enhanced counterpart solely due to unmitigated membrane locking, with a more classical beam deflection shape maintaining a relatively constant cross section throughout. Although the ANDES-DKQ and the Basic-DKQ have the same local displacement field resolving power of 4 Gauss Points and also have 3-parameter based enhanced bending formulations, the membrane formulation leading to the prevention or inclusion of membrane locking is the decisive factor here and evidently has a significant impact on the results obtained.



(a) ANDES-DKQ @ $\delta = 0.0025m$



(b) ANDES-DKQ @ $\delta = 0.015m$



(c) Basic-DKQ @ $\delta = 0.015m$

Figure 66 Plate wrinkling: ANDES-DKQ and Basic-DKQ Z-displacement plots over equilibrium path

Through the comparison of results obtained with various elements employing different element enhancements it is clear that structural modelling with shells, as discussed in section 2.1, requires careful consideration of simplifications and assumptions made. For the analysed problem of a CHS beam buckling dimensional reduction can be reasonably varied between a 1D and 2D approach, as explored. Undertaking a one dimensional beam approach may yield quick and acceptable 'ballpark' results, such as Euler's buckling formula, but it also immensely filters the space of possible mechanical expressions resolvable, such as local ovalisation

which was seen to form an important driver in some of the results obtained. However, simply adopting a two dimensional shell approach to the problem did not homogenize all results, as was seen. Element base formulations, element enhancements, element order and mesh were all important factors strongly affecting results and correspond to questions that must be answered upon entering the shell regime.

9.2. Shear wrinkling of plate

Although classic Euler buckling is perhaps the most apparent phenomena that comes to mind when considering beam structures, the previous example highlights that local buckling effects can come to dominate the overall structural behaviour. Progressing along this vein, another local instability phenomena that can occur with I-sectioned beams is web buckling wherein the web manifests out of plane bubble-like deformation patterns.

The effect of element selection and enhancement on this phenomena can be considered by analysing a flat plate of $1.2m \times 0.3m \times 0.005m$ thick fully clamped on its lower edge and laterally displaced by $U_x = \delta = 0.015m$ along its top edge, roughly corresponding to an I-beam thin web under shear action. Material properties are taken as $E = 206.9GPa$ and $\nu = 0.0$. The system setup is illustrated below:

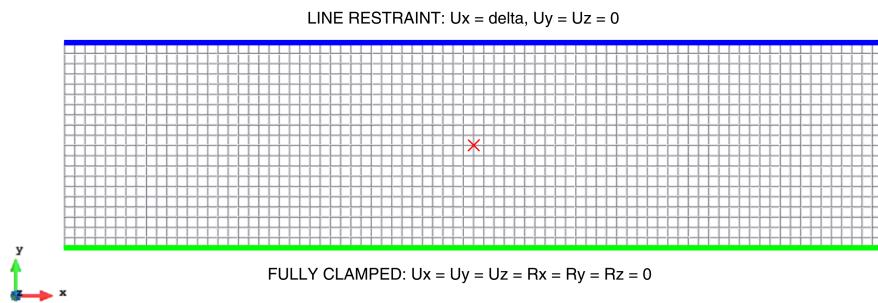


Figure 67 Shear wrinkling of plate setup

As per the previous problem, bifurcation is encouraged with minor out of plane loading, this case being $F_z = -100N$ applied at the centre of the plate denoted by the red cross above.

The first result presented is the imposed lateral displacement δ vs the lateral load, determined as the sum of all nodal lateral reactions R_x along the displaced top edge.

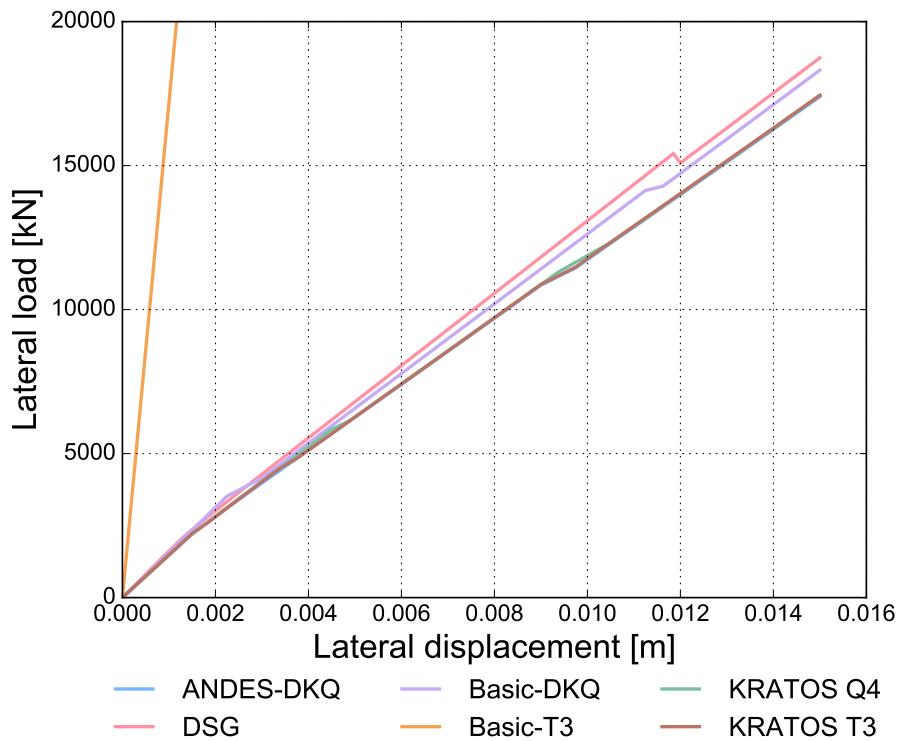


Figure 68 Plate shear stability analysis: lateral displacement vs lateral load

Separated from the other elements, the Basic-T3 element demonstrates spurious behaviour indicative of severe shear locking yet again. This also illustrates the effectiveness of DSG enhancements to inoculate against this deleterious phenomena. Apart from the Basic-T3 element, all remaining elements are grouped relatively close together exhibiting largely linear, albeit slightly softening, behaviour with apparent disturbances at lateral loads of roughly 4000N and 15000N

To avail further insight of the problem, the out of plane displacement of the mid-point (denoted with a red cross in figure 67) is plotted against the lateral load.

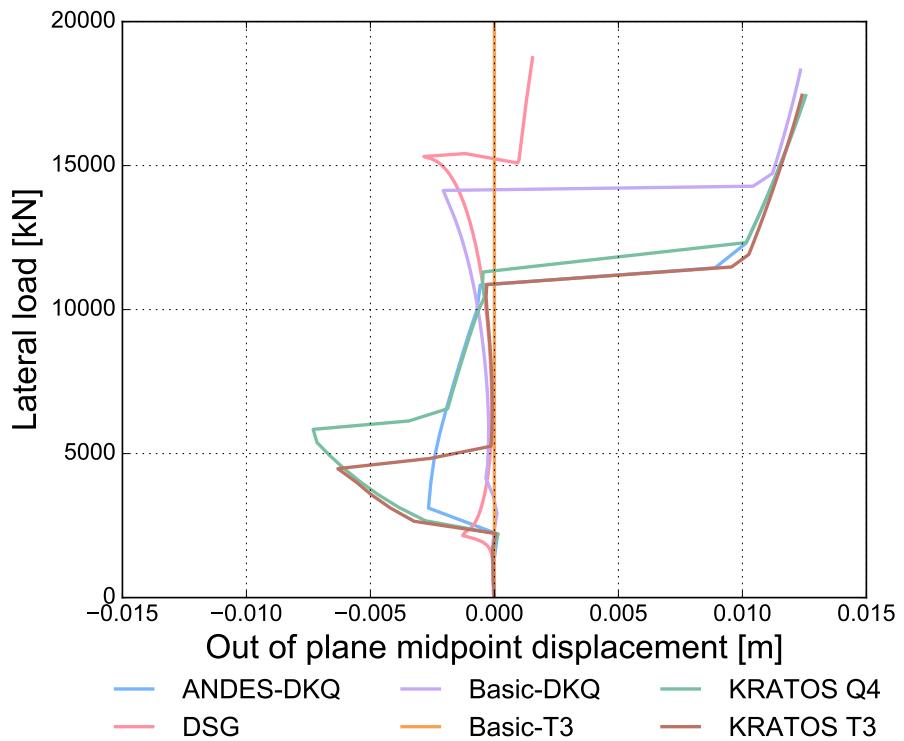
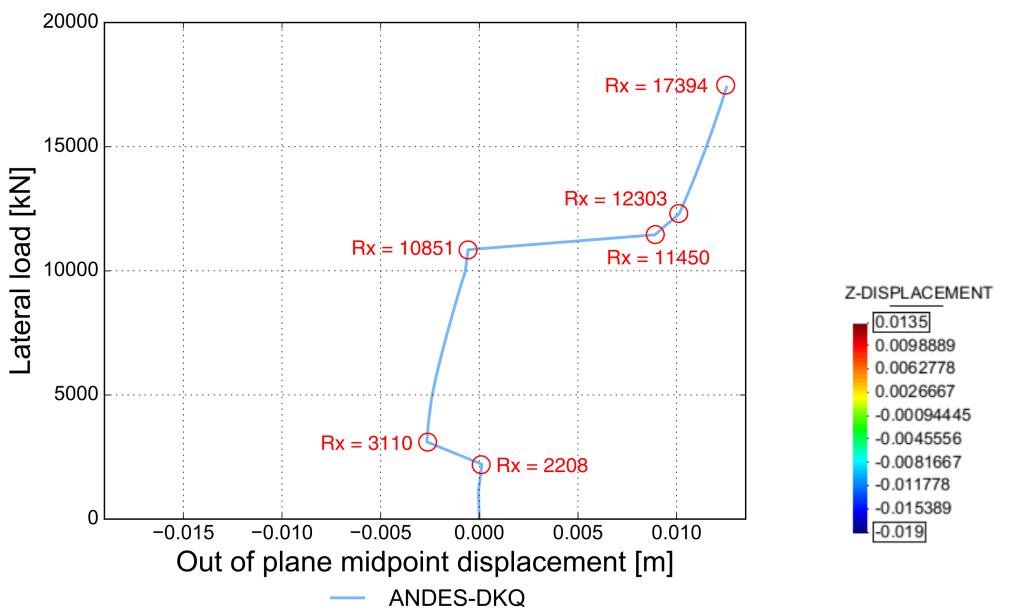


Figure 69 Plate shear stability analysis: out of plane mid-point displacement vs lateral load

This perspective shift affords new clarity illustrating the Basic-T3 element never buckles during the whole analysis thereby explaining its equilibrium path in figure 68. The remaining elements exhibit highly complex non-linear behaviour bifurcating between 3 (DSG) and 5 (Kratos-Q4 and Kratos-T3) times. Despite this, all reasonable elements demonstrate temporary stability in two zones: the secondary branch swaying leftwards in the load range of $(2500 - 6000)N$; and the final branch in the load range of $12000N$ onwards. Despite this rough reconciliation, significant differences of structural behaviour still exist between the elements, which can be explored by examining the displacement contours at key points of interest.

In order to establish a general appreciation of the structural behaviour throughout the lateral reaction load range considered Z-displacement plots of the ANDES-DKQ element, fairly representative of the other elements' behaviour, at various key points on its equilibrium path are presented below:



(a) Response diagram

(b) Contour colour scale

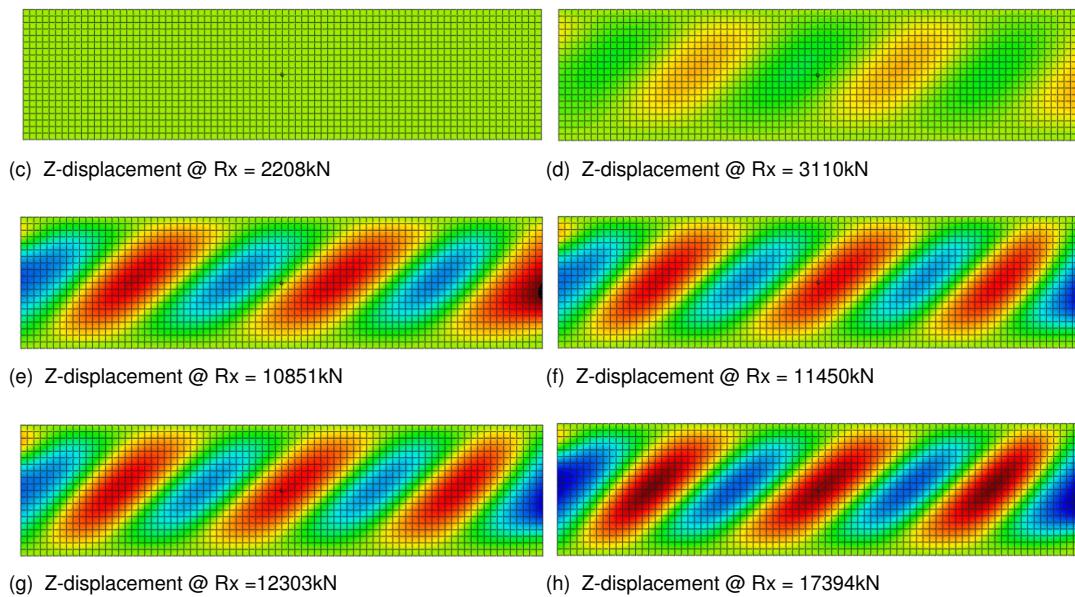


Figure 70 Plate wrinkling: ANDES-DKQ Z-displacement plots over equilibrium path

Plot (c) above shows the stressed unbuckled plate acting in pure membrane action with no developed out of plane displacement (other than that due to the minor point load) just before the onset of instability. Plot (d) highlights the deformation pattern in the secondary equilibrium path, which, for many of the other elements considered, constitutes the first zone of temporary stability as discussed before. A significant loss of membrane stiffness is somewhat tempered by the activation of the plate's bending stiffness via the relatively minor local curvature developed. Progressing to plot (e) illustrates the developing magnitude of the out of plane displacements and gradual re-distribution of the plate's buckling mode into a

pattern of shorter period. The shift from state (e) to state (f) and finally to state (g) marks the transition to the final tightly defined buckling pattern of 3 peaks and 2 troughs, from an initial diffuse pattern of 2 peaks and 2 troughs, which activates more of the bending stiffness due to the higher curvatures involved. The final state (h) highlights the development and utilisation of this buckling mode which forms the second common zone of temporary stability among the other elements considered.

With a greater appreciation of the structural behaviour at hand, the different contours of the DSG and Kratos-T3 element, representing 5-parameter and 3-parameter based formulations respectively, are considered at the two zones of temporary stability.

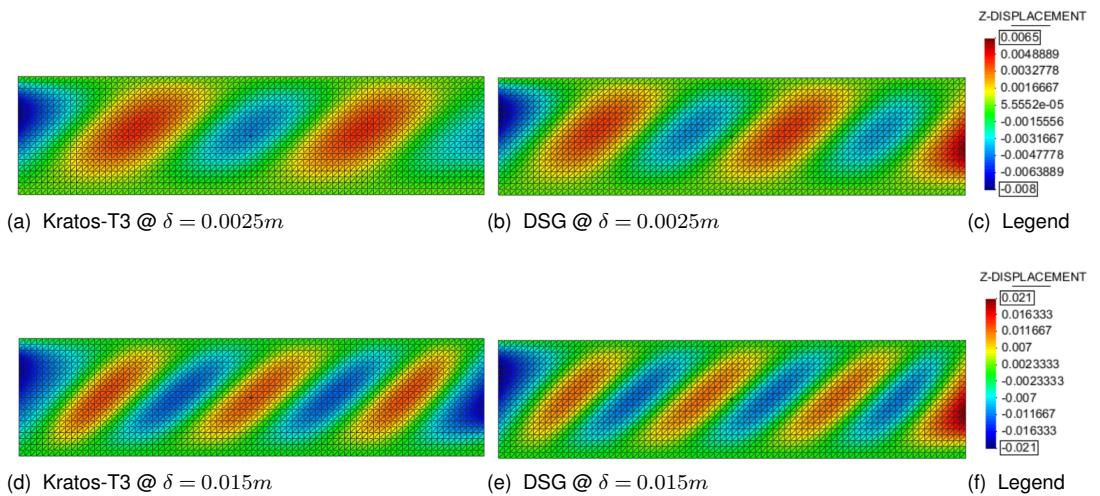


Figure 71 Plate wrinkling: Kratos-T3 and DSG Z-displacement plots over equilibrium path

Although the displacement contours above minimize the differences suggested by figure 69 the DSG element displays a shorter buckling bubble period than the 3-parameter Kratos-T3 element. Intuitively, the phenomenological association between locking (which the DSG element largely mitigates, but remains in its "genes" due to its 5-parameter foundation) and greater stiffness can be superimposed on the general effect of higher stiffness's reducing oscillation periods in general physics, explaining to the reduced spacing between peaks (and troughs) in the DSG element. Despite this difference, it must be noted that compared to the Basic-T3 element, the DSG element performs admirably given its 5-parameter heritage. Furthermore, the smallest slenderness ratio of $L/t = 0.3/0.005 = 60$ places this analysis firmly in the realms of thin plates, rendering 5-parameter based elements as "fish out of water".

As per the above considerations, plots of the ANDES-DKQ and Basic-DKQ elements at the two temporary stability points are presented below:

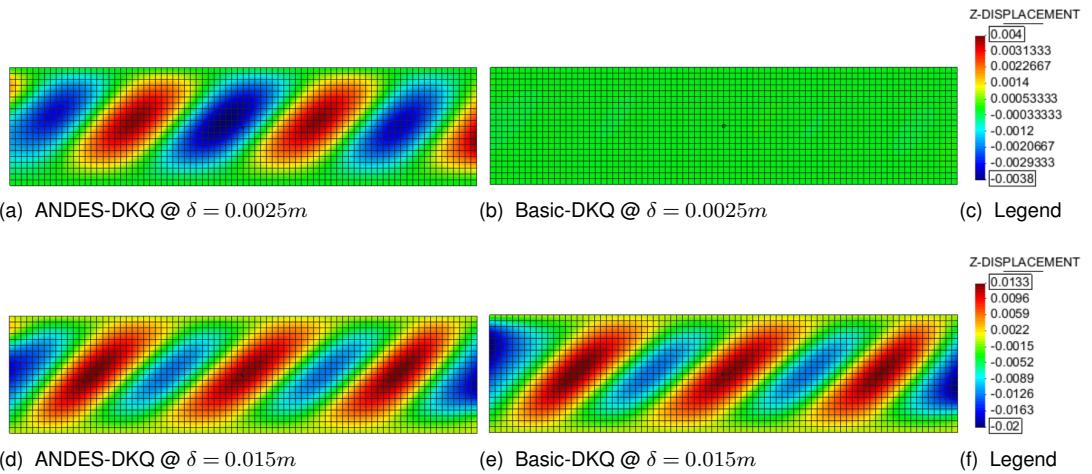


Figure 72 Plate wrinkling: ANDES-DKQ and Basic-DKQ Z-displacement plots over equilibrium path

Given that the problem considered is almost perfectly flat (except due to the destabilising central point load) and relatively free of membrane locking, one would expect no appreciable difference between these two elements (which only differ by the ANDES membrane enhancement), although the plots above suggest otherwise. The Basic membrane formulation clearly delays bifurcation into the secondary path that the ANDES formulation switches to at this load level, along with most other elements considered. Thus, it is likely that the Basic-DKQ element experienced very minor amounts of membrane locking depriving sufficient bending energy allocation and delaying the onset of autocatalytic out of plane displacements, themselves the precursor to entering the secondary branch. Despite this delayed onset of buckling, the final state of the system is demonstrably dominated by the bending formulation, with both elements converging to near identical displacement patterns and magnitudes. Thus, in general, from this comparison, it seems apparent that membrane formulations regulate the onset of instability of the problem while bending formulations determine the buckled mode shapes.

The last perspective considered is the maximum (across the whole domain) absolute out of plane displacement against lateral load, which offers a simplified bulk characterisation of the structural behaviour.

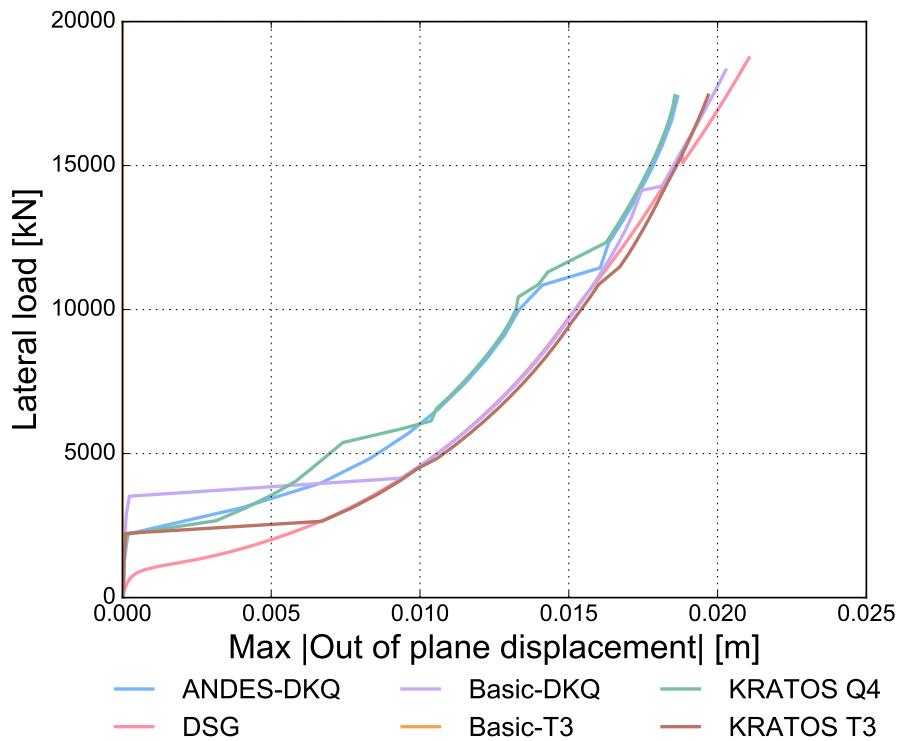


Figure 73 Plate shear stability analysis: maximum absolute out of plane displacement vs lateral load

The figure above summarises the bulk behaviour of the structural and highlights key phenomena of interest in a general sense. The initial vertical paths of the elements (the Basic-T3 path is entirely vertical) represent the deformation under pure membrane action which shortly after break laterally upon bifurcating into the secondary branch. For instance, the late bifurcation of the Basic-DKQ element due to its basic membrane formulation is clearly exposed. After bifurcating, as explored previously, the different elements enter into slightly different buckling modes, explaining the variation of paths between elements, however all buckling modes substitute the lost membrane stiffness with the activated bending stiffness leading to the shared hardening behaviour across all elements.

Complementing the summary of how element enhancements affect the CHS buckling analysis, the plate shear wrinkling also confirms the significant effect of element formulation and enhancement selection in the shell regime. Membrane technologies were found to be key in regulating the onset of buckling, with the un-enhanced membrane formulations either considerably delaying bifurcation (Basic-DKQ) or preventing it completely (Basic-T3). Some interplay with bending technologies is also present in determining initial bifurcation, explaining the difference between the Basic-T3 and DSG element behaviours, both of which have the same membrane formulation. The remaining enhanced membrane formulations of ANDES (ANDES-DKQ, Kratos-T3) and EAS (Kratos-Q4) predict similar points of buckling as per figure 73. This example highlights the importance of selecting appropriate element technologies not only based on the initial state of the system (which, in this case, a basic membrane

formulation would suffice) but also on the predicted deformed state of the system (where a basic membrane formulation is insufficient).

Upon bifurcation, the bending formulation and technologies largely determined the buckling shape exhibited by the system. The proper selection of a 3 or 5 parameter based formulation suitable to the problem considered appears to be of the utmost importance. Although the 5-parameter based DSG element achieved quite similar results to the 3-parameter based elements naturally suitable to slender problem, differences were nonetheless all too apparent, suggesting that element technologies themselves aren't the silver bullet to all problems, but are most effective when applied to base formulations inherently suitable to the problem considered.

Chapter 10 Extension of DSG triangle element technology - STILL TO DO!!!!

ALTHOUGH the DSG element technology drastically enhances the performance of the basic constant strain triangle (Basic-T3) formulation in a very computationally efficient package, the drawback of nodal numbering dependency in coarse linear triangular meshes has tempted academics since its original publication to extend the elegant underlying discrete shear gap concept into a formulation invariant of nodal ordering. This chapter concerns itself with: one such formulation successfully published and gaining traction, another approach under development and illuminating the severity of the DSG nodal numbering sensitivity in order to ascertain whether remedies are actually worthwhile in practical scenarios.

10.1. Cell Smoothed DSG approach

A published and successful approach to remedy the aforementioned nodal numbering dependency is the Cell Smoothed Discrete Shear Gap (CS-DSG) method proposed by Nguyen-Thoi et al. [37] in 2013. The overarching idea is to split each triangular element into 3 sub-triangles, perform the original DSG formulation on each sub-triangle, then assemble these contributions via an area-averaged approach to recover the behaviour of the meta-triangle. In Reference [37], Nguyen-Thoi and his team perform the cell smoothing approach for the membrane, bending and shear B matrices of the original DSG formulation. Since the shear B-matrix is the only component that exhibits nodal ordering dependency, the cell smoothing technique is only applied to the shear B-matrix in this work. The following section illustrates the formulation of the CS-DSG, applied to the shear B-matrix only.

10.1.1. CS-DSG formulation

Crucial to the formulation is the identification of each triangular element's centre point, designated P_0 , which defines the geometry of the 3 sub-triangles, as per the figure below:

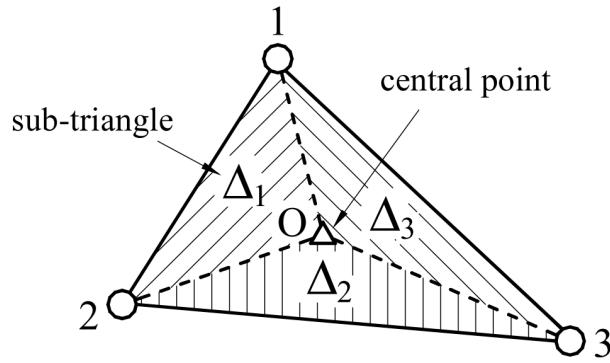


Figure 74 Division of triangle into 3 sub-triangles about centre point [40]

The coordinates of the central point are determined from the 3 exterior points as such:

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \frac{1}{3} \sum_{i=1}^3 \begin{pmatrix} x_i \\ y_i \end{pmatrix} \quad (10.1)$$

As per figure 74, the displacement vectors for each sub-triangle \mathbf{u}^{Δ^i} are:

$$\mathbf{u}^{\Delta^1} = \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \end{pmatrix}, \quad \mathbf{u}^{\Delta^2} = \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{pmatrix}, \quad \mathbf{u}^{\Delta^3} = \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_3 \\ \mathbf{u}_1 \end{pmatrix} \quad (10.2)$$

Critical to the CS-DSG formulation is the relation of \mathbf{u}_0 to the exterior nodal displacement vectors which facilitates the central node to be evaporated from the final formulation. According to the geometrical relation, the central point displacement vector can be expressed as:

$$\mathbf{u}_0 = \frac{1}{3} \sum_{i=1}^3 \mathbf{u}_i \quad (10.3)$$

For each of the sub-triangles Δ_i the original DSG formulation according to equation 5.10 is performed, with x_i , y_i and A updated to match the nodal positions and area of the current sub-triangle respectively. For illustrative purposes, the first sub-triangle shear B-matrix $\mathbf{B}^{\gamma\Delta^1}$ is written explicitly:

$$\mathbf{B}^{\gamma\Delta 1} = \left(\begin{array}{cccc|cccc|cccc} 0 & 0 & b-c & 0 & A_{\Delta 1} & 0 & 0 & 0 & c & \frac{-bc}{2} & \frac{ac}{2} & 0 & 0 & 0 & -b & \frac{bc}{2} & \frac{-bd}{2} & 0 \\ 0 & 0 & d-a & -A_{\Delta 1} & 0 & 0 & 0 & 0 & -d & \frac{bd}{2} & \frac{-ad}{2} & 0 & 0 & 0 & a & \frac{-ac}{2} & \frac{ad}{2} & 0 \end{array} \right) \quad (10.4)$$

with:

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} x_{\Delta n2} - x_{\Delta n1} \\ y_{\Delta n2} - y_{\Delta n1} \\ y_{\Delta n3} - y_{\Delta n1} \\ x_{\Delta n3} - x_{\Delta n1} \end{pmatrix} = \begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \\ y_2 - y_0 \\ x_2 - x_0 \end{pmatrix} \quad (10.5)$$

Furthermore, the shear B-matrix $\mathbf{B}^{\gamma\Delta i}$ for each sub-triangle can be further split into nodal contributions $\mathbf{B}_{\Delta n j}^{\gamma\Delta i}$ corresponding to node j in sub-triangle i , continued in the example of the first sub-triangle below:

$$\mathbf{B}_{\Delta n 1}^{\gamma\Delta 1} = \begin{pmatrix} 0 & 0 & b-c & 0 & A_{\Delta 1} & 0 \\ 0 & 0 & d-a & -A_{\Delta 1} & 0 & 0 \end{pmatrix} \quad (10.6)$$

$$\mathbf{B}_{\Delta n 2}^{\gamma\Delta 1} = \begin{pmatrix} 0 & 0 & c & \frac{-bc}{2} & \frac{ac}{2} & 0 \\ 0 & 0 & -d & \frac{bd}{2} & \frac{-ad}{2} & 0 \end{pmatrix} \quad (10.7)$$

$$\mathbf{B}_{\Delta n 3}^{\gamma\Delta 1} = \begin{pmatrix} 0 & 0 & -b & \frac{bc}{2} & \frac{-bd}{2} & 0 \\ 0 & 0 & a & \frac{-ac}{2} & \frac{ad}{2} & 0 \end{pmatrix} \quad (10.8)$$

Thus, the shear strains for sub-triangle 1 can be summarised as:

$$\boldsymbol{\gamma}^{\Delta 1} = \mathbf{B}^{\gamma\Delta 1} \mathbf{u}^{\Delta 1} = \begin{pmatrix} \mathbf{B}_{\Delta n 1}^{\gamma\Delta 1} & \mathbf{B}_{\Delta n 2}^{\gamma\Delta 1} & \mathbf{B}_{\Delta n 3}^{\gamma\Delta 1} \end{pmatrix} \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \end{pmatrix} \quad (10.9)$$

Recalling equation 10.3, which expresses the central point's displacement vector \mathbf{u}_0 in terms of the 3 external node displacement vectors, the shear B-matrix above can similarly be rearranged

and expressed in terms of the 3 exterior nodes:

$$\gamma^{\Delta 1} = \mathbf{B}_r^{\gamma \Delta 1} \mathbf{u} = \begin{pmatrix} \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 1}}{3} + \mathbf{B}_{\Delta n2}^{\gamma \Delta 1} & \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 1}}{3} + \mathbf{B}_{\Delta n3}^{\gamma \Delta 1} & \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 1}}{3} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{pmatrix} \quad (10.10)$$

The above process can be repeated for sub-triangles 2 and 3, yielding the following results:

$$\gamma^{\Delta 2} = \mathbf{B}_r^{\gamma \Delta 2} \mathbf{u} = \begin{pmatrix} \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 2}}{3} & \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 2}}{3} + \mathbf{B}_{\Delta n2}^{\gamma \Delta 2} & \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 2}}{3} + \mathbf{B}_{\Delta n3}^{\gamma \Delta 2} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{pmatrix} \quad (10.11)$$

$$\gamma^{\Delta 3} = \mathbf{B}_r^{\gamma \Delta 3} \mathbf{u} = \begin{pmatrix} \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 3}}{3} + \mathbf{B}_{\Delta n3}^{\gamma \Delta 3} & \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 3}}{3} & \frac{\mathbf{B}_{\Delta n1}^{\gamma \Delta 3}}{3} + \mathbf{B}_{\Delta n2}^{\gamma \Delta 3} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{pmatrix} \quad (10.12)$$

With the 3 sub-triangle shear B-matrices expressed in terms of the exterior nodes, they can be recombined via an area-averaged approach to recover the total meta-triangle shear B-matrix \mathbf{B}^γ :

$$\mathbf{B}^\gamma = \frac{1}{A_e} \sum_{i=1}^3 \mathbf{B}_r^{\gamma \Delta i} A_{\Delta i} \quad (10.13)$$

10.1.2. Invariance of CS-DSG to nodal ordering

The feature of the CS-DSG formulation is the removal of the original DSG formulation's nodal ordering dependency. The nodal ordering invariance of the CS-DSG formulation is verified with the following example of an equilateral triangle (side lengths $L = 1$, thickness $h = 0.05$) of material properties $E = 1000$, $\nu = 0.29$ fully clamped on its left side (as per the figure below) subject to a surface pressure of $P_z = -0.01$. The displacement of interest is u_z of point 2 as circled below. 3 tests were performed corresponding to 3 different node ordering arrangements of the triangle: the first being [p1, p3, p2], the second [p3, p2, p1] and the last [p2, p1, p3]. The results of the analysis are presented below.

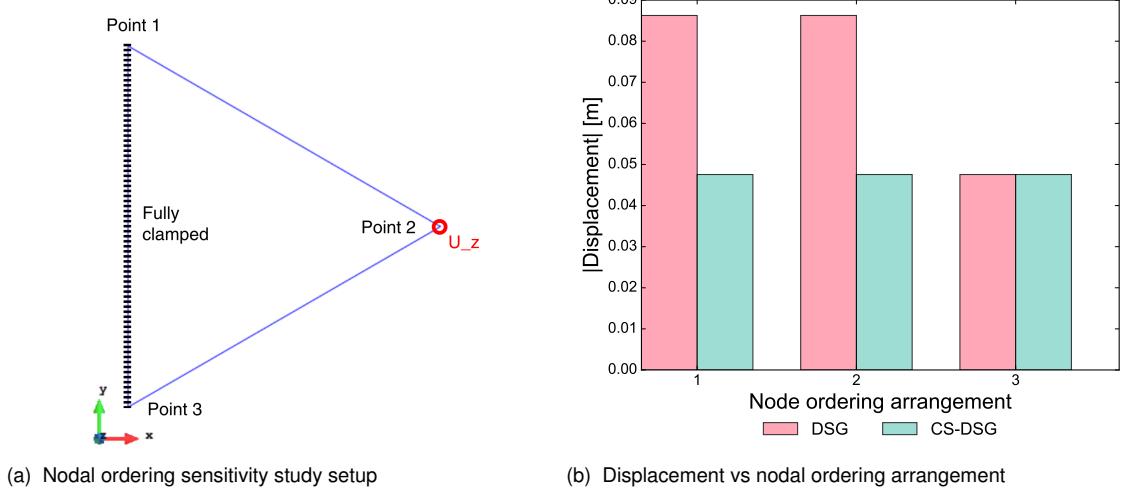


Figure 75 Sensitivity of nodal ordering between DSG and CS-DSG elements

The nodal ordering dependency of the DSG formulation is clear to see, with the third arrangement producing different displacements than the first two. Contrasting this, the CS-DSG formulation produces the exact same result across all nodal numbering arrangements, confirming it is indeed invariant of nodal numbering.

10.1.3. Comparison of DSG and CS-DSG elements in the shell obstacle course

Although the focus of the previous test was nodal numbering dependency, the accuracy of the CS-DSG has not been considered yet. The advantage of nodal numbering invariance is clearly rendered useless if the element is not accurate. Thus, the CS-DSG is run through the shell obstacle course (as per section 8.1) in the current section, with the original DSG formulation results presented for comparison.

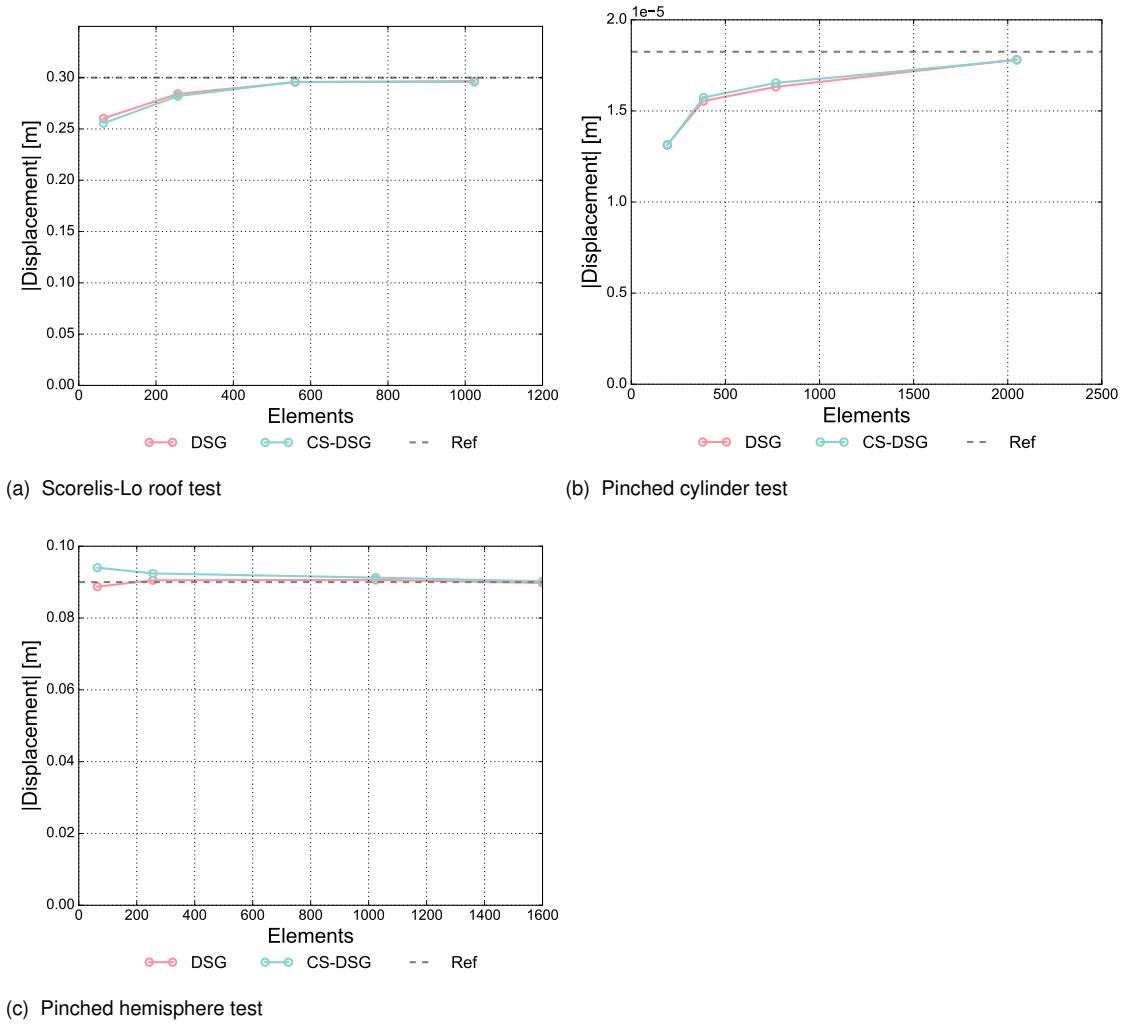


Figure 76 Shell obstacle course: DSG vs. CS-DSG

Results of the shell obstacle course confirm the behaviours of the DSG and CS-DSG elements converge as the mesh is refined, corresponding to the DSG nodal numbering dependency dissipating with finer meshes, as described in the original DSG formulation. The greatest difference occurring between the elements in the first data points (64 element coarse mesh) of the pinched hemisphere test is still relatively minor, highlighting that the DSG nodal dependency of this coarse mesh is already significantly diminished from that of a single triangle (refer figure 75 (b)). Thus, the CS-DSG element maintains the accuracy of the original DSG element while also mitigating the nodal dependency of the original element. However, one may invoke the *no free lunch* theorem, and hold the added computational effort of the CS-DSG element against the limited range of nodal dependency of the DSG element. Two questions arise to address this dichotomy: *"how fast does the nodal dependency of the DSG element dissipate to negligible levels?"*, and, *"how gross is the additional computational effort required for the CS-DSG element?"*. These two questions are addressed in section 10.3, after the discussion of another alternate DSG formulation extension below.

10.2. DSGc3 approach

The DSGc3 approach is a method under development lead by Prof. Bletzinger at TUM's chair of structural engineering aimed at addressing the aforementioned nodal numbering dependency of Bletzinger's original DSG formulation. In it's current stage, the DSGc3 approach offers a successful proof of concept in the form of a parametric-only unit triangle formulation, with the next stage in development being the transformation into arbitrary Cartesian triangles.

10.2.1. DSGc3 unit triangle parametric formulation

Consider the following parametric unit triangle with nodal coordinates $(\xi_i, \eta_i) = (0,0), (1,0), (0,1)$ as per the following figure:

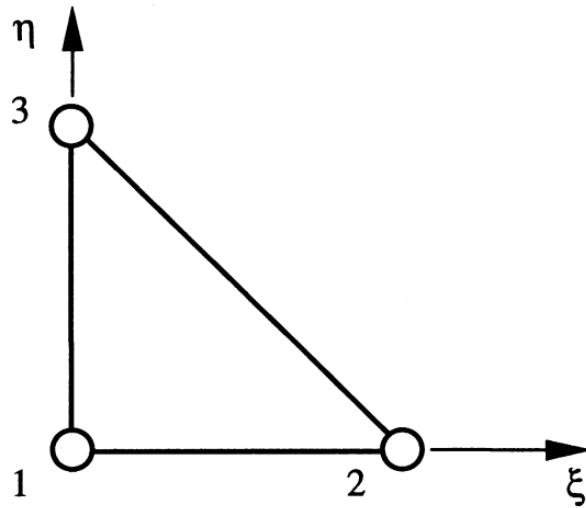


Figure 77 Parametric unit triangle [12]

Three transverse displacement fields are considered across the triangle: a Kirchhoff-Love field w_{KL} , a displacement field due to Reissner-Mindlin shear $w_{RM\gamma}$ and a moderator field w_{MOD} which has the utility of matching both preceding fields thereby rendering the element free of locking effects. The form of the three fields are:

$$w_{KL} = a_1 + \xi a_2 + \eta a_3 + \xi^2 a_4 + 0.5(a_5 + a_6)\xi\eta + \eta^2 a_7 \quad (10.14)$$

$$w_{RM\gamma} = \xi a_8 + \eta a_9 \quad (10.15)$$

$$w_{MOD} = 0.5(a_5 - a_6)\xi\eta \quad (10.16)$$

The coefficients a_1 through a_9 are ansatz coefficients as yet unknown corresponding to one or more triangle DOFs each.

The interaction between the moderator field and the Kirchhoff-Love fields is introduced along parametric axes as

$$w_{KL\xi} = w_{KL} + w_{MOD} = a_1 + \xi a_2 + \eta a_3 + \xi^2 a_4 + a_5 \xi \eta + \eta^2 a_7 \quad (10.17)$$

and

$$w_{KL\eta} = w_{KL} - w_{MOD} = a_1 + \xi a_2 + \eta a_3 + \xi^2 a_4 + a_6 \xi \eta + \eta^2 a_7 . \quad (10.18)$$

According to it's definition, differentiating the modified Kirchhoff-Love displacement field yields a pure rotation field.

$$\beta_{KL\xi} = \frac{\partial w_{KL\xi}}{\partial \xi} = a_2 + 2\xi a_4 + a_5 \eta \quad (10.19)$$

$$\beta_{KL\eta} = \frac{\partial w_{KL\eta}}{\partial \eta} = a_3 + a_6 \xi + 2\eta a_7 \quad (10.20)$$

A displacement gap field can be recovered by identifying the difference between the Reissner-Mindlin shear-displacement and Kirchhoff-Love displacement fields:

$$\Delta_w = w_{RM\gamma} - w_{KL} = -a_1 - a_2 \xi - a_3 \eta - a_4 \xi^2 - a_7 \eta^2 + a_8 \xi + a_9 \eta - 0.5 \eta \xi (a_5 + a_6) . \quad (10.21)$$

Until now the fields have been written in general terms of unknown ansatz coefficients. The fields can be clarified by considering 9 boundary conditions, sufficient to set-up 9 equations and solve the 9 unknowns.

The first three equations constrain the displacement gap field to fulfil discrete nodal displacements:

$$bc1 = \Delta_w(0,0) = w_1 = -a_1 , \quad (10.22)$$

$$bc2 = \Delta_w(1,0) = w_2 = -a_1 - a_2 - a_4 + a_8 , \quad (10.23)$$

and

$$bc3 = \Delta_w(0,1) = w_3 = -a_1 - a_3 - a_7 + a_9 . \quad (10.24)$$

The remaining six impose rotation conditions on the nodal extremities of the modified Kirchhoff-Love fields, shaping both it and the introduced moderator field:

$$bc4 = \beta_{KL\xi}(0,0) = \beta_{\xi 1} = a_2 , \quad (10.25)$$

$$bc5 = \beta_{KL\xi}(1,0) = \beta_{\xi 2} = a_2 + 2a_4 , \quad (10.26)$$

$$bc6 = \beta_{KL\xi}(0,1) = \beta_{\xi 3} = a_2 + a_5 , \quad (10.27)$$

$$bc7 = \beta_{KL\eta}(0,0) = \beta_{\eta 1} = a_3 , \quad (10.28)$$

$$bc8 = \beta_{KL\eta}(1,0) = \beta_{\eta 2} = a_3 + a_6 , \quad (10.29)$$

and

$$bc9 = \beta_{KL\eta}(0,1) = \beta_{\eta 3} = a_3 + 2a_7 . \quad (10.30)$$

Solving the above system of equations yields the following ansatz coefficients:

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \end{pmatrix} = \begin{pmatrix} -w_1 \\ \beta_{\xi 1} \\ \beta_{\eta 1} \\ 0.5(\beta_{\xi 2} - \beta_{\xi 1}) \\ \beta_{\xi 3} - \beta_{\xi 1} \\ \beta_{\eta 2} - \beta_{\eta 1} \\ 0.5(\beta_{\eta 3} - \beta_{\eta 1}) \\ w_2 - w_1 + 0.5(\beta_{\xi 1} + \beta_{\xi 2}) \\ w_3 - w_1 + 0.5(\beta_{\eta 1} + \beta_{\eta 3}) \end{pmatrix} . \quad (10.31)$$

For completeness, the Reissner-Mindlin shear-displacement and moderator fields are re-written with the ansatz results substituted:

$$w_{RM\gamma} = \eta(0.5\beta_{\eta 1} + 0.5\beta_{\eta 3} - w_1 + w_3) + \xi(0.5\beta_{\xi 1} + 0.5\beta_{\xi 2} - w_1 + w_2) , \quad (10.32)$$

and

$$w_{MOD} = \eta\xi(-0.5\beta_{\xi 1} + 0.5\beta_{\xi 3} + 0.5\beta_{\eta 1} - 0.5\beta_{\eta 2}) . \quad (10.33)$$

Indeed, the resulting Reissner-Mindlin shear-displacement field above is similar to the DSG field in the original DSG formulation only without skewed geometry interactions accounted for.

As the moderator field was introduced into the Kirchhoff-Love field previously, so can it be introduced into the Reissner-Mindlin shear-displacement field as

$$w_{RM\gamma\xi} = w_{RM\gamma} + w_{MOD} \quad (10.34)$$

and

$$w_{RM\gamma\eta} = w_{RM\gamma} - w_{MOD}. \quad (10.35)$$

Similar to the original DSG formulation, the shear deformation field can be precipitated from the modified Reissner-Mindlin shear-displacement field via differentiation, yielding

$$\gamma_{RM\gamma\xi} = \frac{\partial w_{RM\gamma\xi}}{\partial \xi} = 0.5\beta_{\xi 1} + 0.5\beta_{\xi 2} - w_1 + w_2 + \eta(-0.5\beta_{\xi 1} + 0.5\beta_{\xi 3} + 0.5\beta_{\eta 1} - 0.5\beta_{\eta 2}) \quad (10.36)$$

and

$$\gamma_{RM\gamma\eta} = \frac{\partial w_{RM\gamma\eta}}{\partial \eta} = 0.5\beta_{\xi 1} + 0.5\beta_{\xi 3} - w_1 + w_3 - \xi(-0.5\beta_{\xi 1} + 0.5\beta_{\xi 3} + 0.5\beta_{\xi 1} - 0.5\beta_{\xi 3}). \quad (10.37)$$

The entries of the above fields can be ordered in a strain-displacement B matrix relating the shear fields to the triangle plate theory DOFs, resulting in:

$$\mathbf{B}_\gamma = \begin{pmatrix} -1 & 1 & 0 & 0.5(1-\eta) & 0.5 & 0.5\eta & 0.5\eta & -0.5\eta & 0 \\ -1 & 0 & 1 & 0.5\xi & 0 & -0.5\xi & 0.5(1-\xi) & 0.5\xi & 0.5 \end{pmatrix}. \quad (10.38)$$

10.2.2. DSGc3 example

In order to illustrate the locking-free performance of the DSGc3 proof of concept formulation, a Python implementation of the element solving prescribed displacements is considered. As per the formulation under development, a unit triangle of thickness $h = 0.005$ is considered with isotropic material properties $E = 1000$, $\nu = 0.0$ subject to no external influence except the following prescribed displacements: $w_1 = w_3 = 0.5$, $w_2 = \beta_{\xi 1} = \beta_{\eta 1} = 1.0$ and $\beta_{\xi 3} = 1.5$. Thus the DOFs $\beta_{\xi 2}$, $\beta_{\eta 2}$ and $\beta_{\eta 3}$ form the problem unknowns. The results of the analysis determine the following unknown values: $\beta_{\xi 2} = -2.0$, $\beta_{\eta 2} = 1.5$ and $\beta_{\eta 3} = -1.0$. Of more interest, however, is the visual representation of the underlying formulation fields, with the various displacement fields fulfilling the solution presented below:

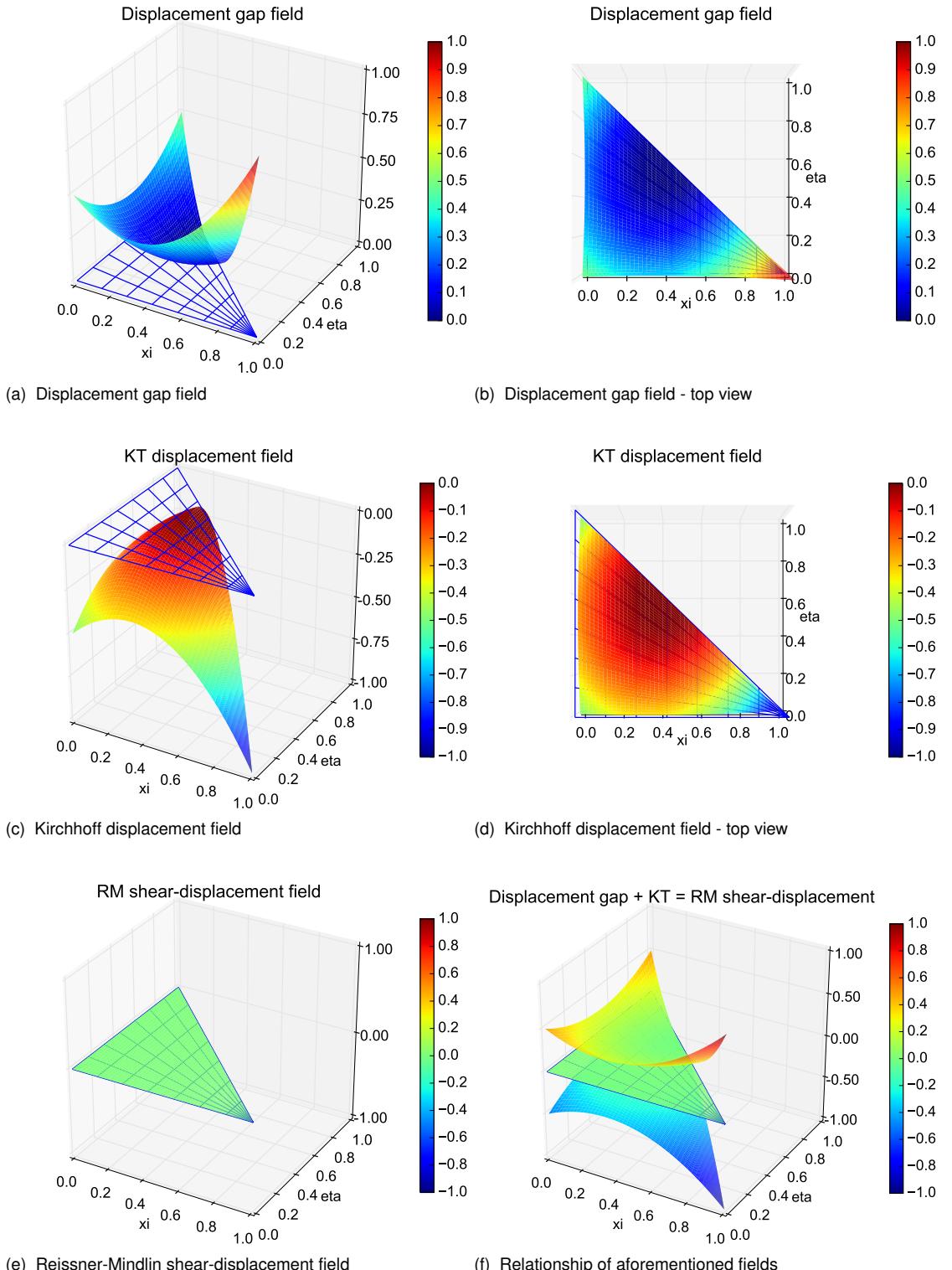


Figure 78 DSGc3 example problem displacement field results

The displacement gap field in sub-figures (a) and (b) illustrate the fulfilment of nodal transverse displacement as per equations 10.22 through 10.24, while the Kirchhoff displacement field in sub-figures (c) and (d) fulfils the nodal rotations of equations 10.25 through 10.30. The

locking-free performance of the element is confirmed in (e) with the RM shear-displacement field having a value of zero throughout. Sub-figure (f) highlights the relationship between these three fields, as per equation 10.21, with the displacement gap and Kirchhoff fields essentially nullifying each other to mitigate locking effects.

The rotation fields of the formulation are presented below:

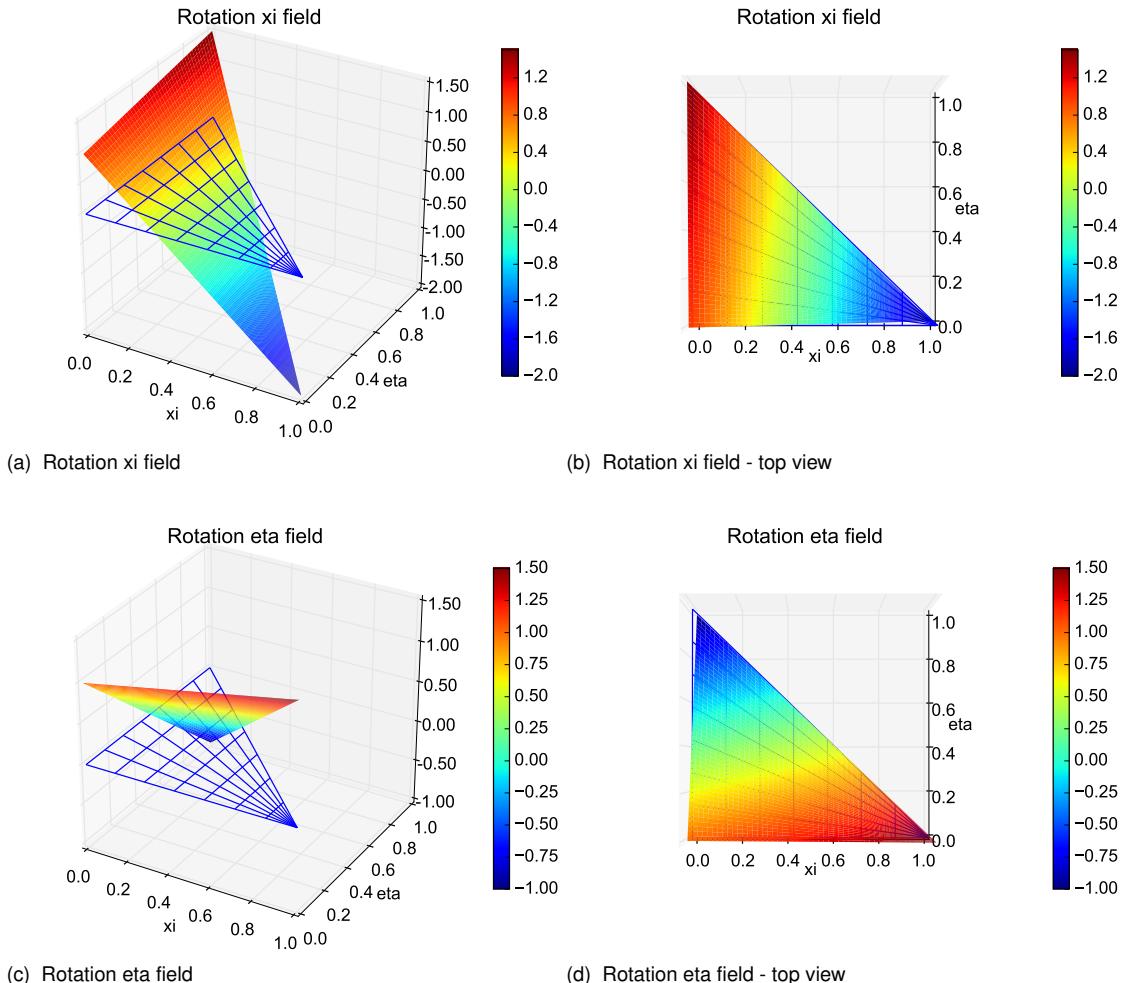


Figure 79 DSGc3 example problem rotation field results

As per equations 10.19 and 10.20, both rotation fields exhibit linear interpolation of nodal values, themselves corresponding to the solution values due to the constraints of equations 10.25 through 10.30.

As an emerging formulation, the DSGc3 offers a promising proof of concept for a locking-free linear triangle element. Further steps to generalise the current unit-triangle parametric formulation to an arbitrarily skewed Cartesian formulation are necessary to effectively use the element in general FEM code, such as Kratos. Thus, given the current state of the

DSGc3 formulation, the remaining discussion of extended DSG formulations is limited to the aforementioned CS-DSG element.

10.3. Appraisal of alternative DSG technology approaches

In an attempt to evaluate whether the DSG or CS-DSG is more suitable for general purpose FEA use, the two questions previously posed are recalled:

- *"how fast does the nodal dependency of the DSG element dissipate to negligible levels?",* and,
- *"how gross is the additional computational effort required for the CS-DSG element?".*

Both of these question are addressed in the subsequent sections.

10.3.1. Dissipation rate of DSG nodal ordering dependency

The dissipation rate of the DSG nodal ordering dependency is studied with the following $20 \times 20 \times 1$ thick square plate of isotropic material $E = 1 \times 10^6$, $\nu = 0.29$ subject to a uniform pressure of $P_z = -1$. Two non-coincident boundary condition cases are designed to extricate the underlying nodal numbering dependency: the first being the bottom edge fully clamped (and all others free) and the second the left edge fully clamped (and all others free).

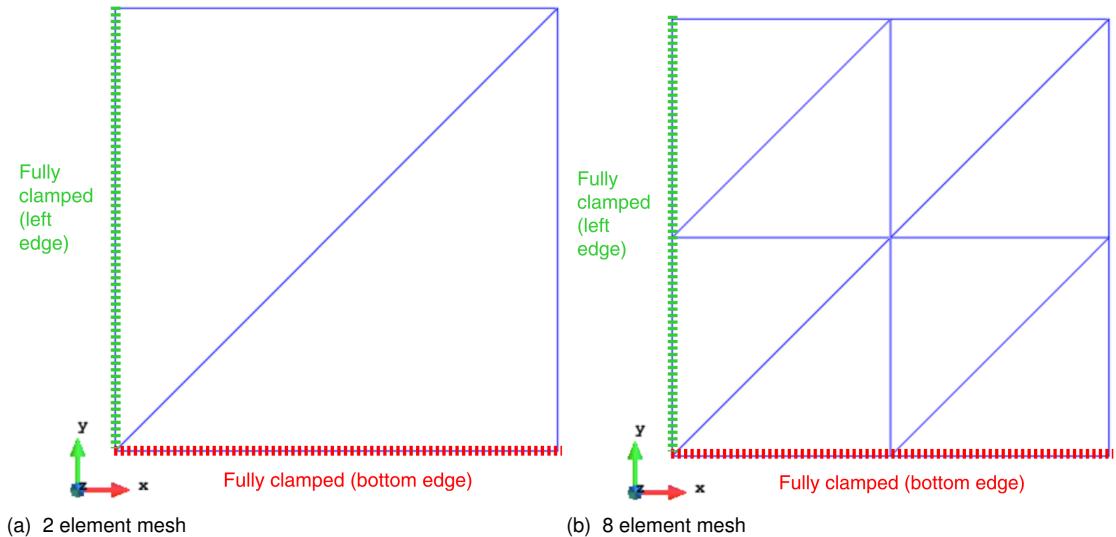


Figure 80 DSG nodal dependency dissipation rate study setup

These two boundary conditions are imposed separately, with the minimum z-displacement across the whole domain taken as the displacement of interest. The results of the study are presented below:

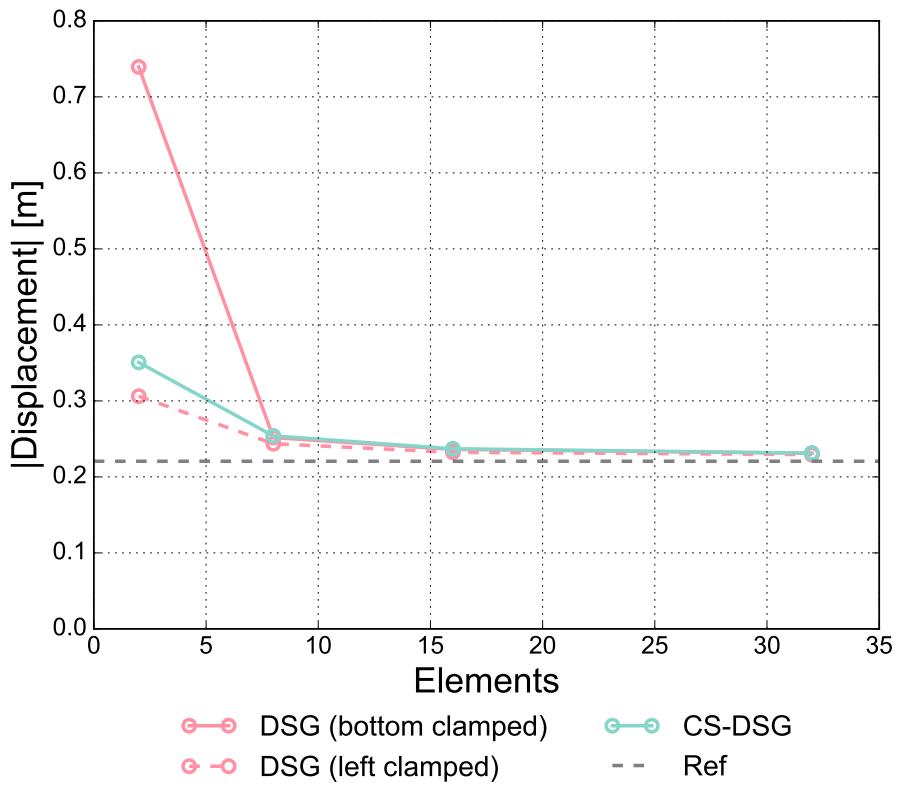


Figure 81 Nodal ordering sensitivity study of DSG formulations

As expected, the coarsest 2 element mesh reveals a high level of nodal numbering dependency in the DSG element, with the 2 boundary conditions producing wildly different displacement results. The CS-DSG element produced identical displacement results for this coarse mesh across both boundary conditions, as well as for all other meshes studied. The 2 element mesh also reveals that although the CS-DSG has no node-numbering dependency, this doesn't necessarily mean it is more accurate than the DSG mesh. Indeed, the left clamped DSG case is closer to the reference solution, however, the bottom clamped DSG case is quite inaccurate. It seems apparent that in this 2 element mesh, and most likely for other very coarse meshes, the CS-DSG element guarantees a result better than the worst possible DSG result but not necessarily better than the best DSG node ordering sensitive result. Despite this caveat, it's clear that the CS-DSG element produces reliably more accurate results than the DSG element for the coarsest mesh.

Progressing with mesh refinement, the 8 element mesh DSG results demonstrate almost complete convergence between the two boundary conditions, indicating negligible nodal numbering dependency. Additionally, the DSG and CS-DSG results essentially coalesce after this single level of refinement. As the mesh is refined further, differences between the 3 result cases continue to evaporate and they converge to the same reference solution of $\min(u_z) = -0.22058$ calculated with an 'overkill' mesh of 11, 250 elements.

10.3.2. Computational cost vs. error for DSG and CS-DSG elements

The previous study of nodal numbering dependency confirms that although the CS-DSG element outperforms the DSG element on the coarsest meshes, the DSG nodal ordering dependency dissipates relatively quickly. Given this performance disparity, the additional computational cost associated with the DS-CSG element must be known to make an informed decision as to what element should be preferred for general use.

Since the shear B-matrix is the only point of difference between the two formulations, the computational cost discrepancy can be reduced to the time taken to construct the shear B-matrix for each element. The previous analysis' 32 element case was re-run with the average wall time for the construction of the shear B-matrix of the DSG element determined to be 12.000 ns while the CS-DSG counterpart was 13.438 ns. As expected, the simpler DSG formulation is quicker than the CS-DSG formulation, with the latter taking 12% longer than the former to construct. If the results of the previous analysis are converted to cumulative element shear B-matrix construction time ($= \text{elements} \times \text{average time to construct shear B matrix}$, ie. the time taken per job run to construct all shear B-matrices) vs. percentage error (against the reference value) the following graph is produced:

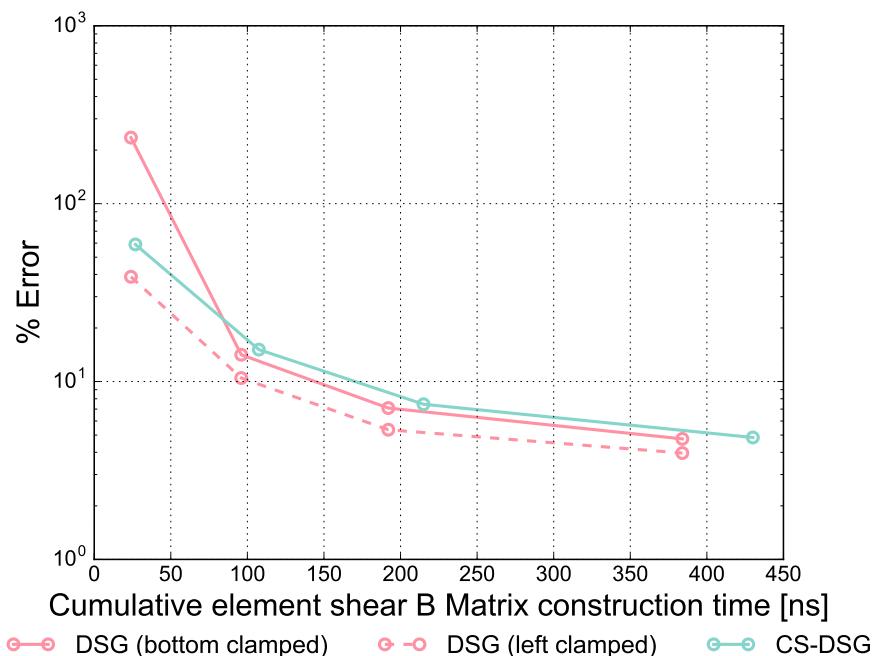


Figure 82 Computational cost vs % error of DSG formulations

Practical FEA is often driven by the tension of obtaining accurate results in a timely manner, with minor differences in element speeds magnified into substantial time delays when a large number of iterations are necessary (such as fine dynamic analysis and highly non-linear analysis). The above graph highlights that the DSG element actually reaches a lower percentage error level 'before' (in the sense of time) the CS-DSG element for all meshes

except the first. That is, for typical practical meshes (indeed 2 element meshes, especially linear triangles, are ill-advised in practise), the DSG element offers comparable accuracy to the CS-DSG element at a lower computational cost. Exceptional cases will no doubt arise where the CS-DSG element is more suitable, a possible example being the analysis of an I-beam where flange meshes are often naturally coarse due to the size difference between the flanges and the web, however this merely reinforces the importance of correct structural modelling of the system at hand and knowledge concerning the interplay of assumptions, element choices and potential deleterious consequences: a central theme of this thesis.

Thus, despite the nodal ordering dependency that demonstrably diminishes quickly, the DSG element appears to be the preferred element for general analysis, while fringe cases involving very coarse meshes would benefit from a shift to the CS-DSG element.

Chapter 11 Conclusions and Outlook - **STILL TO DO!!!!**

This work has considered the implementation of a thin quadrilateral shell element for the multiphysics code KRATOS. Section 1 covered the shell formulation, which is split into membrane and bending components. Following this, the element's implementation in KRATOS was presented, which covered key methods employed and the general workflow to calculate the element stiffness matrix. The element considered was subjected to the well known shell obstacle course in Section 3. Although the element correctly converged to the reference solution for the Scordelis-Lo roof and Pinched Cylinder problem, the Pinched Hemisphere benchmark revealed element deficiencies. These deficiencies were identified, with informed direction suggested for future work associated with improving this element.

11.1. Future work

Extend shell elements model into a variety of directions:

- Consider von karman non-linear strains for very thin shells
- Improve transverse shear stress modelling for composites Look at paper: Improved Transverse Shear Stresses in Composite Finite Elements based on First Order Shear Deformation Theory
Or, develop separate element with higher order resolving theory
- Extend the DSG element into XFEM as per [30]
- Continue development of the DSGc3 formulation

Appendix A Analytical stability analysis of Mises truss

An analytical stability analysis of the Mises truss system considered in chapter 4 is presented, employing the principle of virtual work via the 2nd Piola-Kirchhoff (PK2) stress measure and Green-Lagrange (GL) strain measure.

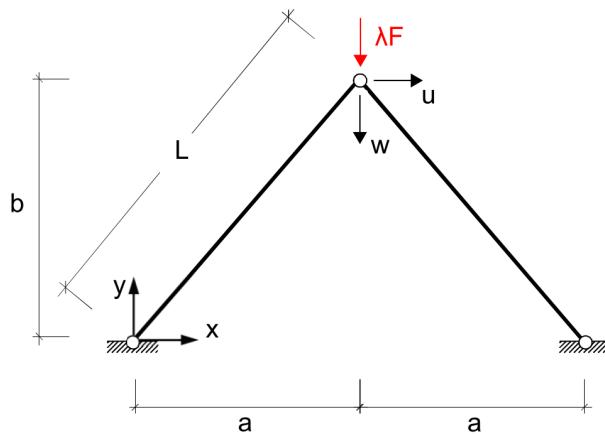


Figure 83 Mises truss geometry

The kinematics of the system are considered first by describing the undeformed truss lengths L (which are the same for both bars), deformed lengths l and the Green-Lagrange strain ϵ_{gl} . Truss 1 is considered first:

$$L^2 = a^2 + b^2 \quad l_1^2 = (a+u)^2 + (b-w)^2 = L^2 + u^2 + 2au + w^2 - 2bw \quad (\text{A.1})$$

The Green-Lagrange strain measure for a truss is recalled and specified, as is its first variation:

$$\epsilon_{gl1} = \frac{1}{2} \left(\frac{l_1^2 - L^2}{L^2} \right) = \left(\frac{u^2 + 2au + w^2 - 2bw}{2L^2} \right) \quad \delta\epsilon_{gl1} = \frac{1}{L^2} [(u+a)\delta u + (w-b)\delta w] \quad (\text{A.2})$$

The kinematics of Truss 2 are presented:

$$l_2^2 = (a - u)^2 + (b - w)^2 = L^2 + u^2 - 2au + w^2 - 2bw \quad (\text{A.3})$$

$$\epsilon_{gl2} = \frac{1}{2} \left(\frac{l_2^2 - L^2}{L^2} \right) = \left(\frac{u^2 - 2au + w^2 - 2bw}{2L^2} \right) \quad \delta\epsilon_{gl1} = \frac{1}{L^2} [(u - a)\delta u + (w - b)\delta w] \quad (\text{A.4})$$

Shifting towards stresses, the 2nd Piola-Kirchhoff stresses are linked to the Green-Lagrange strains via a linearly elastic constitutive law characterised by an axial Young's Modulus E .

$$\sigma_{pk2} = E\epsilon_{gl} \quad (\text{A.5})$$

With the conjugate energy quantities defined, the virtual work expression of the system may be established, in general:

$$-\delta W = \delta W_{int} - \delta W_{ext} = 0 \quad (\text{A.6})$$

Clarifying with respect to the system considered yields:

$$\delta W_{int} - \delta W_{ext} = \int_{V_1} \sigma_{pk2(1)} \epsilon_{gl1} dV_1 + \int_{V_2} \sigma_{pk2(2)} \epsilon_{gl2} dV_2 - \lambda F \delta w = 0 \quad (\text{A.7})$$

Combining the kinematics and constitutive relations yields a crude form of the residual vector \mathbf{r} :

$$\begin{aligned} \mathbf{r} = & EAL \frac{1}{2L^4} (u^2 + 2au + w^2 - 2bw) [(u + a)\delta u + (w - b)\delta w] + \\ & EAL \frac{1}{2L^4} (u^2 - 2au + w^2 - 2bw) [(u - a)\delta u + (w - b)\delta w] - \lambda F \delta w = 0 \end{aligned} \quad (\text{A.8})$$

Development of the above expression, and arranging into vector form yields:

$$\mathbf{r} = \begin{pmatrix} \frac{EA}{L^3} [u^3 + uw^2 - 2bwu + 2a^2u] \\ \frac{EA}{L^3} [u^2w + w^3 - 3bw^2 - bu^2 + 2b^2w] - \lambda F \end{pmatrix} = \mathbf{0} \quad (\text{A.9})$$

Recalling that the system tangent matrix \mathbf{K} is the gradient of the system residual, as per equation 4.2, it can be expressed as follows:

$$\mathbf{K} = \frac{\partial \mathbf{r}}{\partial \mathbf{u}} = \frac{EA}{L^3} \begin{pmatrix} 3u^2 + w^2 - 2bw + 2a^2 & 2uw - 2bu \\ 2uw - 2bu & u^2 + 3w^2 - 6bw + 2b^2 \end{pmatrix} \quad (\text{A.10})$$

Key to stability analysis are the calculation of critical points which indicate the onset of instability. These points occur when the determinant of the tangent matrix vanishes, which reduces to the following expression:

$$(w^2 - 2bw + 2a^2)(3w^2 - 6bw + 2b^2) = 0 \quad (\text{A.11})$$

For the system considered ($a = b = EA = F = 1$), the roots of the above equation are:

$$w_{crit1} = 0.4226 \quad w_{crit2} = 1.5774 \quad (\text{A.12})$$

By substituting these into the system residual the corresponding critical load factors can be determined:

$$\lambda_{crit1} = 0.1361 \quad \lambda_{crit2} = -0.1361 \quad (\text{A.13})$$

With the non-linear portion of the analysis complete, a linearised prebuckling analysis of the system is considered. The general eigenproblem solved in the LPB analysis is recalled from equation 4.8.

$$\det[\mathbf{K}_e + \lambda \mathbf{K}_g(\lambda_{ref})] \mathbf{z} = 0 \quad (\text{A.14})$$

The elastic matrix \mathbf{K}_e is easily determined from $\mathbf{K}(\mathbf{u} = \mathbf{0})$, while the geometric stiffness matrix \mathbf{K}_g for a truss is determined as follows:

$$\mathbf{K}_g = \frac{N}{l} \quad (\text{A.15})$$

N is the normal force in the trusses, derived from vertical static equilibrium, and can also be expressed in terms of the applied load. Furthermore, under the LPB assumption of small

displacements, the deformed and undeformed lengths are considered equivalent $l \approx L$. Thus, \mathbf{K}_g can be reduced to:

$$\mathbf{K}_g = \frac{-\lambda F}{b} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (\text{A.16})$$

The general LPB eigensystem, focussed on critical load factors only, can be assembled:

$$\det \begin{pmatrix} \frac{EA}{L^3} 2a^2 - \frac{\lambda F}{b} & 0 \\ 0 & \frac{EA}{L^3} 2b^2 - \frac{\lambda F}{b} \end{pmatrix} = 0 \quad (\text{A.17})$$

For the system considered ($a = b = EA = F = 1$), the LPB critical load factors coalesce and are calculated to be:

$$\lambda_{lpb\ crit1} = \lambda_{lpb\ crit2} = 0.7071 \quad (\text{A.18})$$

Appendix B DSG technology derivation

The DSG element technology aims to mitigate shear locking in 5-parameter based shell formulations via the concept of discrete shear gaps. This term, coined by Bletzinger et al. [12], refers to the difference in transverse displacements between a pure 3-parameter Kirchhoff formulation and a 5-parameter Reissner-Mindlin formulation. More explicitly, this concept can be illustrated by considering the deformation of a beam (repeated from section 2.4.5):

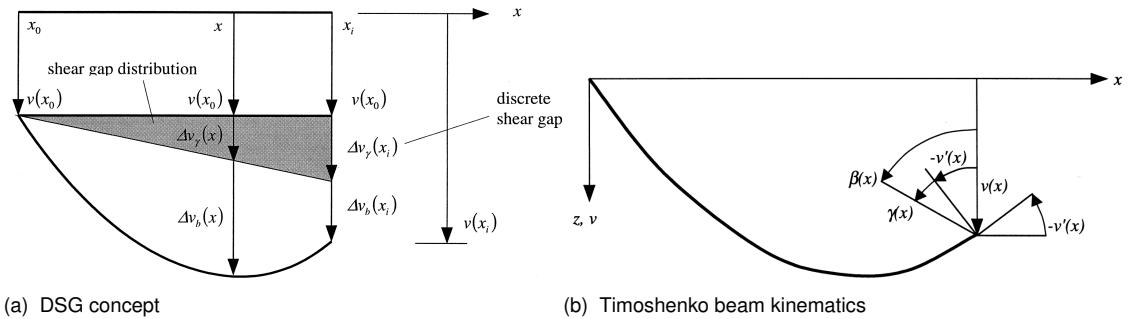


Figure 84 DSG concept and Timoshenko kinematics

The shear deformation $\gamma(x)$ of the beam above is defined as the difference between the section rotation $\beta(x)$ and transverse displacement gradient $v'(x)$:

$$\gamma(x) = v'(x) + \beta(x) \quad (\text{B.1})$$

The Bernoulli beam, being the 2D analogue of the Kirchhoff plate, fulfills the condition of vanishing shear deformation:

$$\gamma(x)_{\text{Bernoulli}} = 0 = v'(x) + \beta(x) \quad (\text{B.2})$$

The shear deformation field (equation B.1) can be discretized into nodal values interpolated with shape functions, typical of the general FEM approach, however, if the same shape

functions are used for rotations and transverse displacements, as discussed in section 2.3, this generally introduces deleterious locking effects. Thus, the novelty of the DSG method, the shear gap, is introduced, which aims to formulate the shear deformation via an integral approach. Considering the 5-parameter formulation analogous Timoshenko beam, the shear gap field can be recovered by integrating the shear distribution over the element:

$$\Delta v_\gamma(x) = \int_{x_o}^x \gamma \, dx = v|_{x_o}^x + \int_{x_o}^x \beta \, dx \quad (\text{B.3})$$

Accordingly, the shear gap field can be approximated with shape functions N_i and discrete shear gaps Δv_γ^i evaluated at each node i :

$$\Delta v_\gamma(x) = \sum_{i=1} N_i \Delta v_\gamma^i, \quad \Delta v_\gamma^i = \int_{x_o}^{x_i} \gamma \, dx = v|_{x_o}^{x_i} + \int_{x_o}^{x_i} \beta \, dx \quad (\text{B.4})$$

The shear deformation field may be recovered by differentiating the shear gap field with the onus on the shape functions:

$$\gamma(x) = \frac{\partial \Delta v_\gamma(x)}{\partial x} = \sum_{i=1} \frac{\partial N}{\partial x} \Delta v_\gamma^i \quad (\text{B.5})$$

With the basic DSG concept established for a simple beam example, the focus can be shifted to the 5-parameter triangular element under development. The general geometry of the element is presented below:

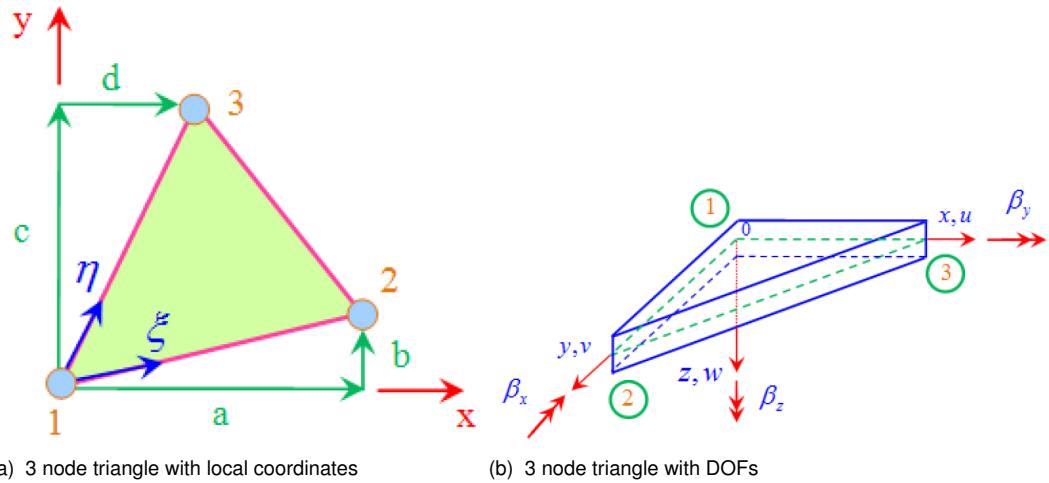


Figure 85 DSG triangle local coordinates and DOFs [37]

At this point it is important to note the current rotation DOF definitions are consistent with plate theory and therefore don't match global Cartesian rotations that follow the right hand rule. The plate theory DOF definitions express shear deformation similar to equation B.1:

$$\begin{pmatrix} \gamma_x \\ \gamma_y \end{pmatrix} = \begin{pmatrix} \frac{\partial w}{\partial x} + \beta_x \\ \frac{\partial w}{\partial y} + \beta_y \end{pmatrix} \quad (\text{B.6})$$

According to the general form of equation B.4, the shear gaps of the triangular element can be determined in parametric space to be:

$$\Delta v_{\gamma 1}^i = w|_{\xi_1}^{\xi_i} + \int_{\xi_1}^{\xi_i} \beta_x a + \beta_y b \, d\xi \quad (\text{B.7})$$

$$\Delta v_{\gamma 2}^i = w|_{\eta_1}^{\eta_i} + \int_{\eta_1}^{\eta_i} \beta_x d + \beta_y c \, d\eta \quad (\text{B.8})$$

The rotation fields β_x and β_y are approximated with nodal values interpolated by the standard linear triangle shape functions:

$$\beta_\alpha(\xi, \eta) = \sum_i^3 N_i(\xi, \eta) \beta_\alpha^i, \quad N_1 = 1 - \xi - \eta, \quad N_2 = \xi, \quad N_3 = \eta \quad (\text{B.9})$$

Evaluation of the discrete shear gaps at each node yields the following results:

$$\Delta v_{\gamma 1}^1 = \Delta v_{\gamma 1}^3 = \Delta v_{\gamma 2}^1 = \Delta v_{\gamma 2}^2 = 0 \quad (\text{B.10})$$

$$\Delta v_{\gamma 1}^2 = w_2 - w_1 + \frac{a(\beta_x^1 + \beta_x^2)}{2} + \frac{b(\beta_y^1 + \beta_y^2)}{2} \quad (\text{B.11})$$

$$\Delta v_{\gamma 2}^3 = w_3 - w_1 + \frac{d(\beta_x^1 + \beta_x^3)}{2} + \frac{c(\beta_y^1 + \beta_y^3)}{2} \quad (\text{B.12})$$

The shear gap field can be constructed with the discrete nodal shear gaps interpolating shape functions (as per equation B.9):

$$\Delta v_{\gamma \alpha}(\xi, \eta) = \sum_i^3 N_i(\xi, \eta) \Delta v_{\gamma \alpha}^i \quad (\text{B.13})$$

Finally, shear deformations are determined by differentiating the shear gap field along Cartesian space:

$$\begin{pmatrix} \gamma_x \\ \gamma_y \end{pmatrix} = \begin{pmatrix} \frac{\partial \Delta v_{\gamma 1}}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \Delta v_{\gamma 2}}{\partial \eta} \frac{\partial \eta}{\partial x} \\ \frac{\partial \Delta v_{\gamma 1}}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \Delta v_{\gamma 2}}{\partial \eta} \frac{\partial \eta}{\partial y} \end{pmatrix} \quad (\text{B.14})$$

The inverse Jacobian is given by:

$$\mathbf{J}^{-1} = \begin{pmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{pmatrix} = \frac{1}{\det J} \begin{pmatrix} c & -b \\ -d & a \end{pmatrix} \quad (\text{B.15})$$

The combination of the above equations leads to Bletzinger's [12] triangle DSG B matrix:

$$\begin{pmatrix} \gamma_x \\ \gamma_y \end{pmatrix} = \frac{1}{\det J} \begin{pmatrix} b-c & \frac{\det J}{2} & 0 & c & \frac{ac}{2} & \frac{bc}{2} & -b & \frac{-bd}{2} & \frac{-bc}{2} \\ d-a & 0 & \frac{\det J}{2} & -d & \frac{-ad}{2} & \frac{-bd}{2} & a & \frac{ad}{2} & \frac{ac}{2} \end{pmatrix} \begin{pmatrix} w_1 \\ \beta_{x1} \\ \beta_{y1} \\ w_2 \\ \beta_{x2} \\ \beta_{y2} \\ w_3 \\ \beta_{x3} \\ \beta_{y3} \end{pmatrix} \quad (\text{B.16})$$

As noted in figure 85, the current B matrix is formulated with respect to rotational DOFs defined as per plate theory, which is not consistent with the Cartesian rotations following the right hand screw rule. To render the DSG formulation in terms of Cartesian rotations suitable for Kratos, the following equivalents can be drawn:

$$\beta_{yi} = -\theta_{xi}, \quad \beta_{xi} = \theta_{yi} \quad (\text{B.17})$$

If the preceding equivalents are substituted into equations B.6 through B.14 the following B matrix, expressed in terms of Cartesian rotations, is obtained, which matches that of Rama et al. [41]:

$$\begin{pmatrix} \gamma_x \\ \gamma_y \end{pmatrix} = \frac{1}{\det J} \begin{pmatrix} b-c & 0 & A & c & \frac{-bc}{2} & \frac{ac}{2} & -b & \frac{bc}{2} & \frac{bd}{2} \\ d-a & -A & 0 & -d & \frac{bd}{2} & \frac{-ad}{2} & a & \frac{-ac}{2} & \frac{ad}{2} \end{pmatrix} \begin{pmatrix} w_1 \\ \theta_{x1} \\ \theta_{y1} \\ w_2 \\ \theta_{x2} \\ \theta_{y2} \\ w_3 \\ \theta_{x3} \\ \theta_{y3} \end{pmatrix} \quad (\text{B.18})$$

Appendix C Basic-DKQ formulation

Introduced in section 8.1, the Basic-DKQ quadrilateral element represents the ANDES-DKQ formulation with the membrane ANDES element technology replaced with an un-enhanced displacement based membrane formulation. The DKQ bending formulation is used as per section 6.1.2.

The membrane stiffness of the Basic-DKQ formulation is purely displacement based and employs the standard bi-linear quadrilateral shape functions:

$$\begin{aligned} N_1(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 - \eta) \\ N_2(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 - \eta) \\ N_3(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 + \eta) \\ N_4(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 + \eta) \end{aligned} \quad (\text{C.1})$$

The membrane strains are related to the discrete in-plane membrane displacements $\hat{\mathbf{v}}$ via the differential operator \mathbf{L} and the aforementioned shape functions \mathbf{N} as such:

$$\boldsymbol{\epsilon} = \mathbf{LN}\hat{\mathbf{v}} = \mathbf{B}\hat{\mathbf{v}} \quad \text{with} \quad \hat{\mathbf{v}} = \begin{pmatrix} \hat{v}_1 \\ \hat{v}_2 \\ \hat{v}_3 \\ \hat{v}_4 \end{pmatrix} \quad \text{and} \quad \hat{\mathbf{v}}_i = \begin{pmatrix} \hat{v}_{xi} \\ \hat{v}_{yi} \end{pmatrix} \quad (\text{C.2})$$

The differential operator is defined as:

$$\mathbf{L} = \begin{pmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{pmatrix} \quad (\text{C.3})$$

The matrix of shape functions is structured as follows:

$$\mathbf{N} = \begin{pmatrix} \mathbf{N}_1 & \mathbf{N}_2 & \mathbf{N}_3 & \mathbf{N}_4 \end{pmatrix} \quad \text{with} \quad \mathbf{N}_i = \begin{pmatrix} N_i & 0 \\ 0 & N_i \end{pmatrix} \quad (\text{C.4})$$

The basic membrane strain displacement matrix \mathbf{B} is thus:

$$\mathbf{B} = \mathbf{LN} = (\mathbf{B}_1 \quad \mathbf{B}_2 \quad \mathbf{B}_3 \quad \mathbf{B}_4) \quad \text{with} \quad \mathbf{B}_i = \begin{pmatrix} N_{i,x} & 0 \\ 0 & N_{i,y} \\ N_{i,y} & N_{i,x} \end{pmatrix} \quad (\text{C.5})$$

The entries of which can be calculated with the help of the Jacobian \mathbf{J} :

$$\mathbf{J} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix} \quad (\text{C.6})$$

$$\mathbf{J}^{-1} \begin{pmatrix} N_{1,\xi} & N_{2,\xi} & N_{3,\xi} & N_{4,\xi} \\ N_{1,\eta} & N_{2,\eta} & N_{3,\eta} & N_{4,\eta} \end{pmatrix} = \begin{pmatrix} N_{1,x} & N_{2,x} & N_{3,x} & N_{4,x} \\ N_{1,y} & N_{2,y} & N_{3,y} & N_{4,y} \end{pmatrix} \quad (\text{C.7})$$

The above strain displacement matrix doesn't cover drilling stiffnesses, so an artificial drilling stiffness as per the DSG element formulation equation 5.14 was added for each node after the construction of the element stiffness matrix:

$$K_{\beta_z} = \frac{\max(K_{ij}\delta_{ij})}{1000} \quad (\text{C.8})$$

Appendix D Basic-T3 formulation

Introduced in section 8.1, the Basic-T3 quadrilateral element corresponds to a basic constant strain triangle element without any DSG enhancements. It shares the same membrane and bending strain displacement matrices as the DSG triangle element (refer equations 5.6 and 5.8), but has a different transverse shear make-up and no correction of the material matrix (as per equation 5.13).

The transverse shear stiffness of the Basic-T3 formulation is purely displacement based and employs the standard linear triangle shape functions as per the DSG element, repeated here:

$$\begin{aligned} N_1(x, y) &= \frac{1}{2A} [(x_2y_3 - x_3y_2) + x(y_2 - y_3) + y(x_3 - x_2)] \\ N_2(x, y) &= \frac{1}{2A} [(x_3y_1 - x_1y_3) + x(y_3 - y_1) + y(x_1 - x_3)] \\ N_3(x, y) &= \frac{1}{2A} [(x_1y_2 - x_2y_1) + x(y_1 - y_2) + y(x_2 - x_1)] \end{aligned} \quad (\text{D.1})$$

The transverse shear strains are related to the derivative of the discrete transverse displacements v_{zi} and the value of nodal rotations β_{xi} and β_{yi} via the following transverse shear strain displacement matrix arrangement:

$$\gamma = (\nabla \mathbf{N}^{v_{zi}} + \mathbf{N}^\beta) \hat{\mathbf{v}} = \mathbf{B} \hat{\mathbf{v}} = (\mathbf{B}_1 \quad \mathbf{B}_2 \quad \mathbf{B}_3) \begin{pmatrix} \hat{\mathbf{v}}_1 \\ \hat{\mathbf{v}}_2 \\ \hat{\mathbf{v}}_3 \end{pmatrix} \quad (\text{D.2})$$

The entries of \mathbf{B}_i and $\hat{\mathbf{v}}_i$ are clarified:

$$\mathbf{B}_i = \begin{pmatrix} N_{i,x} & N_i & 0 \\ N_{i,y} & 0 & N_i \end{pmatrix} \quad \text{and} \quad \hat{\mathbf{v}}_i = \begin{pmatrix} \hat{v}_{zi} \\ \hat{\beta}_{xi} \\ \hat{\beta}_{yi} \end{pmatrix} \quad (\text{D.3})$$

Appendix E Analytical membrane analysis of dome

Forming the reference solution to the isotropic quantity recovery test presented in section 8.4.1, the analytical membrane solution of a simply supported dome with an oculus under self weight is presented.

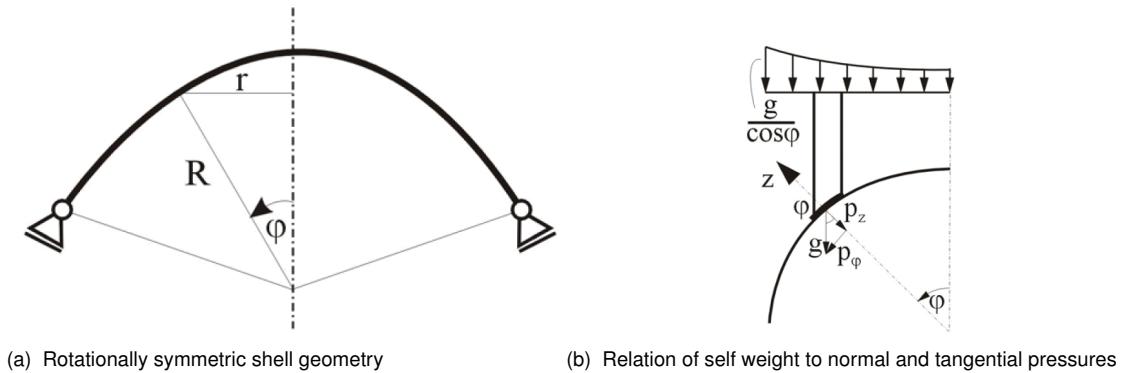


Figure 86 Mid-plane Von Mises stress plots of the simply supported dome analysis

The self weight of a dome with uniform thickness t and density ρ is transferred into normal and tangential pressures as per the following formulae:

$$p_\phi = \gamma t \sin\phi, \quad p_z = -\gamma t \cos\phi \quad \text{with} \quad \gamma = \rho g \quad (\text{E.1})$$

Fulfilling the special case of rotational symmetry, due to uniform thickness and density, the total support reaction P_v of the dome with an oculus corresponding to $\phi_0 = 20^\circ = \frac{\pi}{9} \text{ rad}$ can be determined as:

$$P_v = 2\pi R^2 \int_{\phi_0=\frac{\pi}{9}}^{\phi} \sin^3\phi \gamma t + \sin\phi \cos^2\phi \gamma t \, d\phi + C \quad (\text{E.2})$$

With no edge load along the oculus, $C = 0$. Simplifying and integrating yields:

$$P_v = 2\pi R^2 \gamma t (\cos\phi - \cos\frac{\pi}{9}) \quad (E.3)$$

With the total support reaction known, the meridional n_ϕ and circumferential n_θ force resultants can be determined at any meridional position ϕ with the following simplified expressions:

$$n_\phi = \frac{R\gamma t(\cos\phi - \cos\frac{\pi}{9})}{\sin^2\phi} \quad (E.4)$$

$$n_\theta = -R\gamma t \cos\phi - n_\phi \quad (E.5)$$

Appendix F Derivation of Euler buckling load

As part of the stability analysis of a CHS beam considered in section 9.1 an Euler buckling solution is included for reference, the derivation of which is presented here.

The governing differential for a CHS section beam with Young's Modulus E and a second moment of area I subject solely to an axial compressive load P without any other loading or spring beds imposed is:

$$EIw''' + Pw'' = 0 \quad (\text{F.1})$$

The general solution of the above 4th order differential is a displacement field of the form $w = A\sin(\lambda x) + B\cos(\lambda x) + Cx + D$ where $\lambda^2 = P/EI$. The arbitrary constants A through D can be determined by considering four boundary conditions of the beam which is fixed at both ends. This fixity arrangement prescribes the following displacements and angular deflections at both ends:

$$w(0) = 0 \quad (\text{F.2})$$

$$w(L) = 0 \quad (\text{F.3})$$

$$w'(0) = 0 \quad (\text{F.4})$$

$$w'(L) = 0 \quad (\text{F.5})$$

These boundary conditions can be substituted into the general displacement field and then arranged in a matrix:

$$\begin{pmatrix} 0 & 1 & 0 & 1 \\ \sin(\lambda L) & \cos(\lambda L) & L & 1 \\ \lambda & 0 & 1 & 0 \\ \lambda \cos(\lambda L) & -\lambda \sin(\lambda L) & 1 & 0 \end{pmatrix} \begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = \mathbf{0} \quad (\text{F.6})$$

The critical loads of the system precipitate from the eigenvalues of the matrix above. Thus, the determinant of the above matrix can be simplified and set to zero:

$$\lambda L \sin(\lambda L) + 2[\cos(\lambda L) - 1] = 0 \quad (\text{F.7})$$

The bracketed term can be fulfilled with the following value of λ , which also satisfies the first term too:

$$\lambda = \frac{2n\pi}{L}, n = 1, 2, 3, \dots \quad (\text{F.8})$$

Recalling the original definition of λ , one can combine the eigenvalue and express the critical buckling load of the beam as:

$$P_{crit} = \frac{4\pi^2 n^2 EI}{L^2} \quad (\text{F.9})$$

It's clear that the critical load can be minimized by accepting the first eigenvalue $n = 1$, thus:

$$P_{crit} = \frac{4\pi^2 EI}{L^2} \quad (\text{F.10})$$

Appendix G DSG element class in Kratos

The DSG element is implemented in Kratos under the class `ShellThickElement3D3N` as per the established Kratos element naming convention. This appendix provides a brief description of all methods (arguments excluded for brevity) and variables within class `ShellThickElement3D3N`.

G.1. `ShellThickElement3D3N` methods

G.1.1. Public methods

`ShellThickElement3D3N()`: Overloaded element constructor. Non-linear geometry bool has the default argument of `false()`.

`~ShellThickElement3D3N()`: Element destructor.

`Create()`: Returns pointer to a new instance of this element.

`GetIntegrationMethod()`: Returns currently assigned integration method.

`Initialize()`: Initializes element with material properties, sets up isotropic or orthotropic cross section, initializes coordinate transformation and sets section orientation.

`ResetConstitutiveLaw()`: Resets the cross section by calling `ShellCrossSection::ResetCrossSection()`.

`EquationIdVector()`: Establishes correct relationship between global and local DOF numbering.

`GetDofList()`: Sets up vector of global numbering of local DOFs.

`Check()`: Ensures the validity of element data.

`GetValuesVector()`: Retrieves the element's nodal displacements from a specified historic timestep (default argument is last timestep).

`GetFirstDerivativesVector()`: Retrieves the element's nodal velocities from a specified historic timestep (default argument is last timestep).

`GetSecondDerivativesVector()`: Retrieves the element's nodal accelerations from a specified historic timestep (default argument is last timestep).

`InitializeNonLinearIteration()`: Accessibility function to access
`ShellCrossSection::InitializeNonLinearIteration()`.

`FinalizeNonLinearIteration()`: Accessibility function to access
`ShellCrossSection::FinalizeNonLinearIteration()`.

`InitializeSolutionStep()`: Accessibility function to access
`ShellCrossSection::InitializeSolutionStep()`. Initializes coordinate transformation for the current timestep.

`FinalizeSolutionStep()`: Accessibility function to access
`ShellCrossSection::FinalizeSolutionStep()`. Finalizes coordinate transformation for the current timestep.

`CalculateMassMatrix()`: Calculates element mass matrix. Default is lumped.

`CalculateDampingMatrix()`: Placeholder for future implementation. Currently returns zero matrix.

`CalculateLocalSystem()`: Called by Kratos to add element stiffness contribution to global matrix. Internally re-directs to `CalculateAll()`.

`CalculateRightHandSide()`: Called by Kratos to add element internal force contribution to the global vector. Internally re-directs to `CalculateAll()`.

`GetValueOnIntegrationPoints()`: Overloaded method to calculate requested gauss point results.

`CalculateOnIntegrationPoints()`: Wrapper function callable with Python to access `GetValueOnIntegrationPoints()`.

`Calculate()`: Overloaded method to calculate element specifics based on passed variable. Primarily used to access local element orientation and assign custom section rotations for

composites without knowing the element's derived element type.

`SetCrossSectionsOnIntegrationPoints()`: Allows assigning custom cross sections to the element shell cross section.

G.1.2. Private methods

`CalculateStressesFromForceResultants()`: Calculates stresses from previously determined integrated force resultants as per Reference [19].

`CalculateLaminaStrains()`: Calculates the in-plane strains for the top and bottom surface of each lamina in the laminate as per equation 3.26.

`CalculateLaminaStresses()`: Calculates the in-plane stresses for the top and bottom surface of each lamina in the laminate as per equation 3.32.

`CalculateTsaiWuPlaneStress()`: Called for each lamina in the laminate. Determines the minimum Tsai-Wu reserve factor of the current lamina considering the top and bottom surfaces as per equation 3.36.

`CalculateVonMisesStress()`: Calculates the Von Mises equivalent stress as per equation 5.38. Result can be calculated on the top, middle or bottom shell surface, or the maximum of all 3 surfaces depending on variable passed to method.

`CheckGeneralizedStressOrStrainOutput()`: Preliminary filter method called immediately by the Matrix overloaded `GetValueOnIntegrationPoints()` method to provide control flow.

`DecimalCorrection()`: Rounds very small entries of vectors to zero based on a dynamic numeric tolerance.

`SetupOrientationAngles()`: Allows the element cross section (material) to be rotated in-plane relative to the element geometry.

`CalculateSectionResponse()`: Calculates section response, including stress computation from available strains, according to the constitutive law assigned. Also implements stabilization of shear constitutive entries as per equation 5.13.

`InitializeCalculationData()`: Initializes element data invariant throughout the gauss loop necessary for the computation of the element stiffness matrix.

`CalculateDSGc3Contribution()`: Calculates the shear B-matrix as per the developmental DSGc3 formulation in equation 10.38.

`CalculateSmoothedDSGBMatrix()`: Calculates the shear B-matrix as per the CS-DSG formulation in equation 10.13.

`CalculateDSGShearBMatrix()`: Called three times during CS-DSG element construction to calculate the sub-triangle's shear B-matrix.

`AddBodyForces()`: Adds self-weight forces based off the global acceleration specified in the main analysis.

`CalculateAll()`: The main pipeline to calculate the element stiffness matrix.

`TryGetValueOnIntegrationPoints_MaterialOrientation()`: Provides a contour-representation of the element's material alignment.

`TryGetValueOnIntegrationPoints_GeneralizedStrainsOrStresses()`: The main pipeline to calculate gauss point results such as strains, stresses, moments and integrated forces.

`printMatrix()`: Utility function to neatly print arbitrary matrices to the console for debugging.

G.2. ShellThickElement3D3N member variables

G.2.1. Private

`class CalculationData`: Contains the bulk of element data necessary for stiffness matrix computation. Detailed below.

`mpCoordinateTransformation`: Instance of the element's coordinate transformation. May be linear or non-linear corotational.

`mSections`: Container for the cross section instances associated with each integration point.

`mThisIntegrationMethod`: Current integration method of the element.

`mOrthotropicSectionRotation`: In-plane rotation angle for the element material cross section. Default is 0.0.

G.2.2. Private, class CalculationData member variables

All variables subsequently described are scoped to `CalculationData` unless noted otherwise.

`LCS0`: Reference coordinate system of undeformed configuration.

LCS: Current coordinate system of deformed configuration.

dA: Integration area associated with each gauss point. Equal to element area for a single gauss point.

hMean: Mean thickness of the shell cross section.

TotalArea: Total area of the element.

gpLocations: Vector of gauss point locations, used for results output.

dNxy: Cartesian derivatives of shape functions.

N: Shape function vector at the current integration point.

globalDisplacements: Global displacement vector.

localDisplacements: Local displacement vector.

CalculateRHS: Bool to calculate internal force vector.

CalculateLHS: Bool to calculate stiffness matrix.

parabolic_composite_transverse_shear_strains: Bool to calculate composite transverse shear strains according to a constant (equation 3.27) or parabolic (equation 3.30) profile. Default value is false.

basicTriCST: Bool to use a basic displacement-derived shear formulation as per equation D.3. Default value is false.

ignore_shear_stabilization: Bool to ignore constitutive shear matrix stabilization as per equation 5.13. Default value is false.

smoothedDSG: Bool to use the CS-DSG formulation as per 10.13. Default value is false.

specialDSGc3: Bool to use the DSGc3 formulation as per 10.38. Default value is false.

gpIndex: Integer holding the index of the current gauss point.

B: Element strain-displacement B matrix.

h_e: Length of the element's longest edge, forming part of the constitutive matrix shear

stabilization calculation in equation 5.13.

`alpha`: Constant-value alpha parameter forming part of the constitutive matrix shear stabilization calculation in equation 5.13.

`shearStabilisation`: Condensation of all terms modifying the shear constitutive matrix as per equation 5.13.

`D`: Element material matrix.

`generalizedStrains`: Element strain vector.

`generalizedStresses`: Element stress vector.

`SectionParameters`: Parameters of the shell cross section calculation.

`rLaminateStrains`: Vector of element lamina strain vectors.

`rLaminateStresses`: Vector of element lamina stress vectors.

`CurrentProcessInfo`: Reference to the Kratos job's current process information.

Appendix H ANDES-DKQ class in Kratos

The ANDES-DKQ element is implemented in Kratos under the class `ShellThinElement3D4N` as per the established Kratos element naming convention. This appendix provides a brief description of all methods (arguments excluded for brevity) and variables within class `ShellThinElement3D4N`.

H.1. `ShellThinElement3D4N` methods

H.1.1. Public methods

`ShellThinElement3D4N()`: Overloaded element constructor. Non-linear geometry bool has the default argument of `false()`.

`~ShellThinElement3D4N()`: Element destructor.

Remaining public methods have names and overall descriptions as per `ShellThickElement3D3N`'s public methods.

H.1.2. Private methods

`CalculateStressesFromForceResultants()`: As per identically named `ShellThickElement3D3N` method.

`CalculateLaminaStrains()`: As per identically named `ShellThickElement3D3N` method.

`CalculateLaminaStresses()`: As per identically named `ShellThickElement3D3N` method.

`CalculateTsaiWuPlaneStress()`: As per identically named `ShellThickElement3D3N` method.

`CalculateVonMisesStress()`: As per identically named `ShellThickElement3D3N` method.

`CheckGeneralizedStressOrStrainOutput()`: As per identically named `ShellThickElement3D3N` method.

`DecimalCorrection()`: As per identically named `ShellThickElement3D3N` method.

`SetupOrientationAngles()`: As per identically named `ShellThickElement3D3N` method.

`InitializeCalculationData()`: As per identically named `ShellThickElement3D3N` method.

`CalculateBMatrix()`: Calculates the strain-displacement B matrix at the current gauss point.

`CalculateSectionResponse()`: Calculates section response, including stress computation from available strains, according to the constitutive law assigned.

`CalculateGaussPointContribution()`: Calculates the current gauss point's contribution to the element stiffness matrix.

`AddBodyForces()`: As per identically named `ShellThickElement3D3N` method.

`CalculateAll()`: As per identically named `ShellThickElement3D3N` method.

`TryGetValueOnIntegrationPoints_MaterialOrientation()`: As per identically named `ShellThickElement3D3N` method.

`TryGetValueOnIntegrationPoints_GeneralizedStrainsOrStresses()`: As per identically named `ShellThickElement3D3N` method.

`printMatrix()`: As per identically named `ShellThickElement3D3N` method.

H.2. `ShellThinElement3D4N` member variables

H.2.1. Public

`class JacobianOperator`: Provides commonly used Jacobian functionality. Detailed below.

H.2.2. Private

`class CalculationData`: Contains the bulk of element data necessary for stiffness matrix computation. Detailed below.

`mpCoordinateTransformation`: Instance of the element's coordinate transformation. May be linear or non-linear corotational.

`mSections`: Container for the cross section instances associated with each integration point.

`mThisIntegrationMethod`: Current integration method of the element.

`mOrthotropicSectionRotation`: In-plane rotation angle for the element material cross section. Default is 0.0.

H.2.3. Public, class JacobianOperator methods

All methods described in this sub-section are scoped to `JacobianOperator` unless noted otherwise.

`JacobianOperator()`: Class constructor

`~JacobianOperator()`: Class destructor.

`Jacobian()`: Returns Jacobian matrix at the current gauss point.

`Inverse()`: Returns inverse Jacobian matrix at the current gauss point.

`XYDerivatives()`: Returns a matrix of Cartesian shape function derivatives at the current gauss point.

`Determinant()`: Returns the determinant of the Jacobian matrix at the current gauss point.

H.2.4. Public, class JacobianOperator member variables

All variables described in this sub-section are scoped to `JacobianOperator` unless noted otherwise.

`mJac`: Jacobian matrix

`mInv`: Inverse Jacobian matrix

`mXYDeriv`: Matrix of Cartesian shape function derivatives.

`mDet`: Determinant of the Jacobian matrix.

H.2.5. Private, class CalculationData member variables

All variables described in this sub-section are scoped to `CalculationData` unless noted otherwise.

`LCS0`: Reference coordinate system of undeformed configuration.

`LCS`: Current coordinate system of deformed configuration.

`s_xi`: ξ unit vector in Cartesian coordinates.

s_eta: η unit vector in Cartesian coordinates.

r_cartesian: Array of Cartesian point positions.

dA: Array of integration areas associated with each gauss point. Pre-multiplied with integration weight, equal to $\det(\mathbf{J}_i)w_i$.

globalDisplacements: Global displacement vector.

localDisplacements: Local displacement vector.

CalculateRHS: Bool to calculate internal force vector.

CalculateLHS: Bool to calculate stiffness matrix.

basicQuad: Bool to use a basic displacement-derived membrane formulation as per equation C.5. Default value is false.

N: Shape function vector at the current integration point.

gpIndex: Integer holding the index of the current gauss point.

ANDES membrane formulation data

alpha: Constant parameter alpha in the ANDES basic membrane formulation as per equation 6.5. Value is 1.5.

L_mem: ANDES basic membrane lumping matrix. Refer equation 6.4.

H_mem_mod: ANDES higher order filter matrix as per Haugen's DOF ordering. Refer equation 6.13.

Z: Transformation matrix between Haugen's and Kratos' DOF ordering for the higher order filter matrix as per equation 6.15.

B_h_1: Higher order membrane B-matrix of node 1, as per equation 6.27.

B_h_2: Higher order membrane B-matrix of node 2, as per equation 6.28.

B_h_3: Higher order membrane B-matrix of node 3, as per equation 6.27.

B_h_4: Higher order membrane B-matrix of node 4, as per equation 6.28.

`B_h_bar`: Mean higher order membrane B-matrix, as per equation 6.30.

`T_13`: Transformation matrix as per equation 6.27.

`T_24`: Transformation matrix as per equation 6.28.

DKQ bending formulation data

`DKQ_a`: Vector of DKQ formulation coefficients a_k as per equation 6.35.

`DKQ_b`: Vector of DKQ formulation coefficients b_k as per equation 6.35.

`DKQ_c`: Vector of DKQ formulation coefficients c_k as per equation 6.35.

`DKQ_d`: Vector of DKQ formulation coefficients d_k as per equation 6.35.

`DKQ_e`: Vector of DKQ formulation coefficients e_k as per equation 6.35.

`DKQ_indices`: Matrix of DKQ mapping indices i, r, s ordering as per commentary beneath equation 6.38.

`DKQ_invJac`: Vector of inverse Jacobian matrices for each gauss point as per equation 6.41.

`DKQ_jac`: Vector of Jacobian matrices for each gauss point as per equation 6.41.

`DKQ_jac_det`: Vector of Jacobian determinants for each gauss point as per equation 6.41.

General calculation-variable data

`B`: Element strain-displacement B matrix.

`D`: Element material matrix.

`BTD`: Auxiliary matrix to store the product $\mathbf{B}^T \mathbf{D}$.

`generalizedStrains`: Element strain vector.

`generalizedStresses`: Element stress vector.

`jacOp`: Instance of `JacobianOperator` class.

`rLaminateStrains`: Vector of element lamina strain vectors.

`rLaminateStresses`: Vector of element lamina stress vectors.

`SectionParameters`: Parameters of the shell cross section calculation.

`CurrentProcessInfo`: Reference to the Kratos job's current process information.

List of Figures

1	Dimensional reduction of a solid to a shell	3
2	Various shell models	4
3	Deformation, reference and current configuration	4
4	Kirchhoff-Love shell kinematics	6
5	Reissner-Mindlin shell kinematics	9
6	Eigenvalue spectra of various shell models	11
7	Convergence of cylindrical shell problem demonstrating membrane locking	13
8	Reduced integration of a 5 parameter Quad 4 shell	14
9	Assumed shear strain field of the MITC4 element	15
10	Discrete Shear Gap (DSG) concept	16
11	DKQ DOF arrangement and geometry	16
12	Shell assembly benefitting from drilling DOFs	18
13	Material composition of the Boeing 787	21
14	Components of composite laminates	22
15	Lamina in a plane stress state	24
16	Continuous unidirectional fibre lamina arrangement	25
17	Arbitrary orientation of lamina	26
18	Force and moment resultants of a plate	28
19	Coordinate system and lamina numbering in a laminate	29
20	Deformation of 3 parameter plate	31
21	Stress and strain thickness distribution of 4 ply plate subject to pure bending	34
22	Examples of response diagrams	37
23	Stability analysis response diagrams	38
24	Mises truss geometry	41
25	Ratio of Mises truss LPB and NL critical load factors across varying geometry	42
26	Ratio of Mises truss LPB and NL critical load factors across varying geometry	43
27	High level overview of DSG element workflow	49
28	ANDES membrane nodal strain gauges	61
29	Geometric dimensions of the quadrilateral element	61
30	Higher order torsional mode of ANDES membrane formulation	63
31	DKQ DOF arrangement and geometry	66
32	High level overview of ANDES-DKQ element workflow	70
33	High level overview of composite element workflow	78
34	Definition of the Scordelis-Lo roof benchmark	83
35	Scordelis-Lo roof benchmark results	83

36	Definition of the pinched cylinder benchmark	84
37	Pinched cylinder benchmark results	84
38	Definition of the pinched hemisphere benchmark	85
39	Pinched hemisphere benchmark results	85
40	Hinged cylindrical roof benchmark	86
41	Open cylinder pullout benchmark	87
42	Shell pendulum definition	88
43	Vertical displacement over time of the shell pendulum analysis	88
44	Oscillating clamped plate definition	89
45	Vertical displacement over time of the oscillating clamped plate analysis	89
46	Results of the simply supported dome analysis	90
47	Definition of the Navier supported plate test	91
48	Stresses (bottom surface) of the Navier supported plate under sinusoidal load	92
49	Stresses of the Navier supported plate under uniformly distributed load	93
50	Definition of the composite barrel vault test	94
51	Composite barrel vault test results	95
52	Composite clamped cylinder test definition	96
53	Composite clamped cylinder analysis convergence	96
54	Composite hinged cylindrical roof test	97
55	Composite dynamic tests	98
56	Stresses and Tsai-Wu results of the composite tensile test	99
57	Stresses and strains of the Navier supported ANDES-DKQ laminate under a sinusoidally distributed load	101
58	In plane stresses and strains of the Navier supported DSG laminate under a sinusoidally distributed load	102
59	Transverse shear stresses and strains of the Navier supported DSG laminate under a sinusoidally distributed load	103
60	Tsai-Wu reserve factor through the thickness of the Navier supported DSG laminate under a sinusoidally distributed load	104
61	CHS buckling setup	106
62	CHS buckling: axial displacement vs axial load	107
63	CHS buckling: lateral displacement vs axial load	108
64	CHS buckling: Kratos-T3 and DSG displacement plots at the onset of instability .	109
65	CHS buckling: ANDES-DKQ and Basic-DKQ displacement plots at the onset of instability	110
66	Plate wrinkling: ANDES-DKQ and Basic-DKQ Z-displacement plots over equilibrium path	111
67	Shear wrinkling of plate setup	112
68	Plate shear stability analysis: lateral displacement vs lateral load	113
69	Plate shear stability analysis: out of plane mid-point displacement vs lateral load .	114
70	Plate wrinkling: ANDES-DKQ Z-displacement plots over equilibrium path	115
71	Plate wrinkling: Kratos-T3 and DSG Z-displacement plots over equilibrium path .	116

72	Plate wrinkling: ANDES-DKQ and Basic-DKQ Z-displacement plots over equilibrium path	117
73	Plate shear stability analysis: maximum absolute out of plane displacement vs lateral load	118
74	Division of triangle into 3 sub-triangles about centre point	121
75	Sensitivity of nodal ordering between DSG and CS-DSG elements	124
76	Shell obstacle course: DSG vs. CS-DSG	125
77	Parametric unit triangle	126
78	DSGc3 example problem displacement field results	130
79	DSGc3 example problem rotation field results	131
80	DSG nodal dependency dissipation rate study setup	132
81	Nodal ordering sensitivity study of DSG formulations	133
82	Computational cost vs % error of DSG formulations	134
83	Mises truss geometry	137
84	DSG concept and Timoshenko kinematics	141
85	DSG triangle local coordinates and DOFs	142
86	Mid-plane Von Mises stress plots of the simply supported dome analysis	149

Bibliography

- [1] Bhagwan D Agarwal, Lawrence J Broutman, and K Chandrashekara.
Analysis and performance of fiber composites.
John Wiley & Sons, 2006.
- [2] Fabian Rojas Barrales.
“Development of a nonlinear quadrilateral layered membrane element with drilling degrees of freedom and a nonlinear quadrilateral thin flat layered shell element for the modeling of reinforced concrete walls”.
Dissertation. Los Angeles, California: University of Southern California, 2012.
- [3] Klaus-Jürgen Bathe and Eduardo N. Dvorkin.
“A formulation of general shell elements—the use of mixed interpolation of tensorial components”.
In: *International Journal for Numerical Methods in Engineering* 22.3 (1986), pp. 697–722.
- [4] Jean-Louis Batoz, Klaus-JÜRgen Bathe, and Lee-Wing Ho.
“A study of three-node triangular plate bending elements”.
In: *International Journal for Numerical Methods in Engineering* 15.12 (1980), pp. 1771–1812.
- [5] Jean-Louis Batoz and Mabrouk Ben Tahar.
“Evaluation of a new quadrilateral thin plate bending element”.
In: *International Journal for Numerical Methods in Engineering* 18.11 (1982), pp. 1655–1677.
- [6] Ted Belytschko et al.
“Stress projection for membrane and shear locking in shell finite elements”.
In: *Computer Methods in Applied Mechanics and Engineering* 51.1-3 (1985), pp. 221–258.
- [7] P. G. Bergan and M. K. Nygård.
“Finite elements with increased freedom in choosing shape functions”.
In: *International Journal for Numerical Methods in Engineering* 20.4 (1984), pp. 643–663.

- [8] M. Bischoff et al.
 “Models and Finite Elements for Thin-Walled Structures”.
 In:
Encyclopedia of Computational Mechanics.
 John Wiley and Sons, Ltd, 2004.
- [9] Manfred Bischoff and Kai-Uwe Bletzinger.
 “Improving stability and accuracy of Reissner-Mindlin plate finite elements via algebraic subgrid scale stabilization”.
 In: *Computer Methods in Applied Mechanics and Engineering* 193.15 - 16 (2004), pp. 1517 –1528.
- [10] Manfred Bischoff and Kai-Uwe Bletzinger.
 “Stabilized DSG Plate and Shell Elements”.
 In: *Trends in Computational Structural Mechanics* (2001), pp. 253 –263.
- [11] Kai-Uwe Bletzinger.
Lecture notes in Plates.
 2016.
- [12] Kai-Uwe Bletzinger, Manfred Bischoff, and Ekkehard Ramm.
 “A unified approach for shear-locking-free triangular and rectangular shell finite elements”.
 In: *Computers and Structures* 75.3 (2000), pp. 321 –334.
- [13] Robin Bouclier, Thomas Elguedj, and Alain Combescure.
 “Efficient isogeometric NURBS-based solid-shell elements: Mixed formulation and - method”.
 In: *Computer Methods in Applied Mechanics and Engineering* 267 (2013), pp. 86 –110.
- [14] *Composites vs. Aluminum*.
 American Composites Manufacturers Association.
 2017.
 URL: <http://compositeslab.com/composites-compared/composites-vs-aluminum/>.
- [15] Eduardo N. Dvorkin and Klaus-Jürgen Bathe.
 “A continuum mechanics based four node shell element for general non-linear analysis”.
 In: *Engineering Computations* 1.1 (1984), pp. 77–88.
- [16] Ralph Echter.
 “Isogeometric Analysis of Shells”.
 Dissertation. Stuttgart: Universitaet Stuttgart, 2013.
- [17] C.A. Felippa.
A Tour of Nonlinear Analysis.
 2016.

- [18] C.A. Felippa.
Critical Points.
2016.
- [19] C.A. Felippa.
Kirchhoff Plates: Field Equations.
2017.
- [20] C.A. Felippa.
Stability of Structures: Basic Concepts.
2016.
- [21] C.A. Felippa and C. Militello.
“Variational formulation of high-performance finite elements: Parametrized variational principles”.
In: *Computers and Structures* 36.1 (1990), pp. 1 –11.
- [22] CA Felippa and KC Park.
“Fitting strains and displacements by minimizing dislocation energy”.
In: *Proceedings of the Sixth International Conference on Computational Structures Technology, Prague, Czech Republic.*
2002,
Pp. 49–51.
- [23] Carlos A. Felippa.
“A study of optimal membrane triangles with drilling freedoms”.
In: *Computer Methods in Applied Mechanics and Engineering* 192.16 - 18 (2003),
pp. 2125 –2168.
- [24] Carlos A Felippa.
“Supernatural QUAD4: a template formulation”.
In: *Computer methods in applied mechanics and engineering* 195.41 (2006), pp. 5316–5342.
- [25] William Hansen.
“The The Significance and Measurement of the Tsai-Wu Normal Interaction Parameter F12”.
MA thesis. Oregon State University, 1992.
- [26] Bjørn Haugen.
“Buckling and Stability Problems for Thin Shell Structures Using High Performance Finite Elements”.
Dissertation. Colorado: University of Colorado, 1994.
- [27] Woo-Young Jung and Sung-Cheon Han.
“An 8-Node Shell Element for Nonlinear Analysis of Shells Using the Refined Combination of Membrane and Shear Interpolation Functions,”
In: *Mathematical Problems in Engineering* 2013 (2013).

- [28] Athanasios J Kolios and Stefano Proia.
“Evaluation of the reliability performance of failure criteria for composite structures”.
In: (2012).
- [29] Ireneusz Kreja.
“A literature review on computational models for laminated composite and sandwich panels”.
In: *Open Engineering* 1.1 (2011), pp. 59–80.
- [30] P. Liu et al.
“Buckling failure analysis of cracked functionally graded plates by a stabilized discrete shear gap extended 3-node triangular plate element”.
In: *Composites Part B: Engineering* 77 (2015), pp. 179 –193.
- [31] Mikko Lylly, Rolf Stenberg, and Teemu Vihinen.
“A stable bilinear element for the Reissner-Mindlin plate model”.
In: *Computer Methods in Applied Mechanics and Engineering* 110.3 (1993), pp. 343 –357.
- [32] Richard H. MacNeal.
“Derivation of element stiffness matrices by assumed strain distributions”.
In: *Nuclear Engineering and Design* 70.1 (1982), pp. 3 –12.
- [33] Carmelo Militello and Carlos A. Felippa.
“The First ANDES Elements: 9-DOF Plate Bending Triangles”.
In: *Computer Methods in Applied Mechanics and Engineering* 93 (2 1991), pp. 217–246.
- [34] P. Mohan.
“Development and applications of a flat triangular element for thin laminated shells”.
Dissertation. Virginia: Virginia Polytechnic Institute and State University, 1997.
- [35] Mohammadreza Mostafa.
“A geometric nonlinear solid-shell element based on ANDES, ANS and EAS concepts”.
Dissertation. Colorado: University of Colorado, 2011.
- [36] Alan T Nettles.
“Basic mechanics of laminated composite plates, NASA RP-1351”.
In: (1994).
- [37] T. Nguyen-Thoi et al.
“A cell-based smoothed discrete shear gap method (CS-DSG3) using triangular elements for static and free vibration analyses of shell structures”.
In: *International Journal of Mechanical Sciences* 74 (2013), pp. 32 –45.
- [38] Hieu Nguyen-Van, Nam Mai-Duy, and Thanh. Tran-Cong.
“An improved quadrilateral flat element with drilling degrees of freedom for shell structural analysis”.
In: *CMES: Computer Modeling in Engineering and Sciences* 49.2 (2009), pp. 81 –112.

- [39] Ozden Ochoa and Junuthula Narasimha Reddy.
Finite element analysis of composite laminates.
Dordrecht ; Boston : Kluwer Academic Publishers, 1992.
- [40] Phuc Phung-Van et al.
“Static and free vibration analyses and dynamic control of composite plates integrated with piezoelectric sensors and actuators by the cell-based smoothed discrete shear gap method (CS-FEM-DSG3)”.
In: *Smart Materials and Structures* 22.9 (2013), p. 095026.
- [41] Gil Rama, Dragan Marinkovic, and Manfred Zehn.
“Efficient Co-Rotational 3-Node Shell Element”.
In: *American Journal of Engineering and Applied Sciences* 9 (2 2016), pp. 420–431.
- [42] E. Ramm and W. A. Wall.
“Shell structures - a sensitive interrelation between physics and numerics”.
In: *International Journal for Numerical Methods in Engineering* 60.1 (2004), pp. 381 –427.
- [43] Ekkehard Ramm.
Heinz Isler, Schalen: Katalog zur Ausstellung.
Zurich: vdf Hochschulverlag AG, 2002.
- [44] Junuthula Narasimha Reddy.
Mechanics of laminated composite plates and shells: theory and analysis.
CRC press, 2004.
- [45] J. C. Simo and M. S. Rifai.
“A class of mixed assumed strain methods and the method of incompatible modes”.
In: *International Journal for Numerical Methods in Engineering* 29.8 (1990), pp. 1595–1638.
- [46] K.Y. Sze, X.H. Liu, and S.H. Lo.
“Popular benchmark problems for geometric nonlinear analysis of shells”.
In: *Finite Elements in Analysis and Design* 40.11 (2004), pp. 1551 –1569.
- [47] Wolfgang A Wall, Michael Gee, and Ekkehard Ramm.
“The challenge of a three-dimensional shell formulation-the conditioning problem”.
In: *Proceedings of ECCM*.
Vol. 99.
2000.
- [48] Henry T. Y. Yang et al.
“A survey of recent shell finite elements”.
In: *International Journal for Numerical Methods in Engineering* 47.1-3 (2000).
- [49] O.C. Zienkiewicz and R.L. Taylor.
The Finite Element Method. Volume 2: Solid Mechanics.
Oxford: Butterworth-Heinemann, 2000.

[50] Olgierd Zienkiewicz.
The Finite Element Method.
New York City: McGraw-Hill, 1977.

Declaration

I hereby declare that the thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references. In addition, I declare that I make the present work available to the Chair of Structural Analysis for academic purposes and in this connection also approve of dissemination for academic purposes.

Ort, Datum, Signature