



New rotation-free finite element shell triangle accurately using geometrical data [☆]

P.-A. Ubach ^{*}, E. Oñate

International Center for Numerical Methods in Engineering (CIMNE), Universidad Politécnica de Cataluña, Campus Norte UPC, 08034 Barcelona, Spain¹

ARTICLE INFO

Article history:

Received 26 February 2008
 Received in revised form 9 December 2008
 Accepted 8 January 2009
 Available online 21 January 2009

Keywords:

Thin-shells
 Finite elements
 Rotation-free

ABSTRACT

A new triangle shell element is presented. The advantages of this element are threefold: simplicity, generality and geometrical accuracy. The formulation is free from rotation degrees of freedom. The triangle here presented can be used regardless of the mesh topology, thus generality is conserved for any mesh-represented surface.

From an original first order approach we evolve to a third order geometric description. The higher degree geometric description is based on the Bézier triangles concept, a very well known geometry in the domain of CAGD [G. Farin, Curves and Surfaces for CAGD. A Practical Guide, fifth ed., Morgan Kaufmann Publishers, San Francisco, CA, 2002]. Using this concept we show the path to reconstruct a general third order interpolating surface using only the three coordinates at each node.

This work takes as starting point the nodal implementation of a basic triangle shell element [E. Oñate, F. Zárte, Rotation-free triangular plate and shell elements, Int. J. Numer. Methods Engrg. 47 (2000) 557–603]. In order to use an exact formula for the curvature, the normal directions at each node and the way to characterize them are proposed. Then, the geometrical properties and the mechanical behavior of the surface created are introduced. Finally, different examples are presented to depict the versatility and accuracy of the element.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Several attempts have been made to overcome the C^1 condition that computing curvatures for shells demands [3–6]. This work builds on the original proposal for rotation-free shell elements reported in [3,2,7,8], and extends those principles in order to gather richer geometrical information from the patch of all the elements – not only those sharing one side – surrounding a given triangle in the mesh. In doing this we can avoid the need for computing an averaged curvature for the element and instead we can obtain precise values for the curvature at any point in the triangle, in particular at the Gauss points.

The original approach is based on a first order interpolation of the basis functions. Computation of the curvatures is achieved by construction of the gradient of the surface normal at each element. Since this low order approach yields a non-conforming representation of the geometry, a higher order description of the geometry is needed. This is required to enrich the model and avoid local insta-

stabilities. Everything is achieved by using the Bézier triangle concept without introducing new degrees of freedom in the model.

In all cases simplicity is of the utmost importance when dealing with boundary conditions. Boundary conditions are managed very easily, both for Dirichlet and Neumann conditions. The normal vector to the surface at every node accounts for this simplicity without the need to use additional degrees of freedom nor virtual nodes.

The layout of the paper is the following. In the first section the original idea for computing curvatures is introduced leading to a first set of equations. Drawbacks associated to this approach are found, analyzed and explained. In the second section the solution for the drawbacks is introduced. It consists on carefully constructing cubic Bézier triangles. In the third section the first set of equations is modified and developed in order to account for the information contained in the second section. Details for solving the numerical equations are explained as well. Numerical examples are given in the fourth section. Finally, a discussion with conclusions is included in the fifth and last section.

1.1. Developments to find a good solution

The current approach was first introduced by Ubach and Oñate in [9]. This early proposal was followed by the current framework also by Ubach and Oñate in [10]. However, it must be said that, an almost identical work was presented at the same congress by Linhard et al. [11]. This coincidence emphasizes the interest that the

[☆] Invited article submitted as an expansion of the talks given at the 7th World Congress on Computational Mechanics (Los Angeles, July 2006) and at the 9th United States National Congress on Computational Mechanics (San Francisco, July 2007).

^{*} Corresponding author. Tel.: +34 934134181.

E-mail addresses: ubach@cimne.upc.edu (P.-A. Ubach), onate@cimne.upc.edu (E. Oñate).

¹ <http://www.cimne.upc.edu/>.

computation of thin-shells using rotation-free elements enjoys within the computational mechanics community.

1.2. Raw approach and why it does not work

In the following we write the equations on which the formulation is based.

Let [12,13]:

$$\mathbf{X} \in \mathcal{S} \subset \mathbb{R}^3 | \mathbf{X} = (x, y, z)^T = \boldsymbol{\varphi}(u, v) = \boldsymbol{\varphi}, \tag{1}$$

$\nabla \boldsymbol{\varphi}$ defines the tangent plane to \mathcal{S} as follows:

$$(\nabla \boldsymbol{\varphi})_1 \equiv \boldsymbol{\varphi}_{,1} = \frac{\partial \boldsymbol{\varphi}}{\partial u}, \quad (\nabla \boldsymbol{\varphi})_2 \equiv \boldsymbol{\varphi}_{,2} = \frac{\partial \boldsymbol{\varphi}}{\partial v}. \tag{2}$$

We define:

$$\mathbf{n} = \boldsymbol{\varphi}_{,1} \times \boldsymbol{\varphi}_{,2}, \tag{3}$$

then:

$$\mathbf{n} \perp \boldsymbol{\varphi}_{,1}, \quad \mathbf{n} \perp \boldsymbol{\varphi}_{,2} \Rightarrow \nabla(\mathbf{n} \cdot \nabla \boldsymbol{\varphi}) = \mathbf{0}, \tag{4}$$

$$\nabla(\mathbf{n} \cdot \nabla \boldsymbol{\varphi}) = \nabla \mathbf{n} \cdot \nabla \boldsymbol{\varphi} + \mathbf{n} \cdot \nabla \nabla \boldsymbol{\varphi} = \mathbf{0}, \tag{5}$$

$$\nabla \mathbf{n} \cdot \nabla \boldsymbol{\varphi} = \boldsymbol{\kappa} = -\mathbf{n} \cdot \nabla \nabla \boldsymbol{\varphi}. \tag{6}$$

The means to accomplish the objective of computing the curvature at specific locations – using the first equality given by Eq. (6) – are to compute the values of the normal directions at each of the three nodes of the triangle (this process is described in Section 2.3). Using the three values of the normal at the nodes of the triangle we can build the following vector field of normal directions for the triangle and obtain values for its derivatives:

$$\underline{\mathbf{n}} = \frac{\sum_{i=1}^3 N_i(\xi_1, \xi_2) \cdot \mathbf{n}^i}{\|\sum_{i=1}^3 N_i(\xi_1, \xi_2) \cdot \mathbf{n}^i\|}. \tag{7}$$

Using this field of normal vectors we can express the position of any point of the shell as:

$$\mathbf{x} = \sum_{i=1}^3 N_i(\xi_1, \xi_2) \cdot \mathbf{x}^i + \lambda \xi_3 \cdot \mathbf{n}, \tag{8}$$

where the shape functions are represented by N_i and λ stands for the thickness of the shell in the current configuration.

The development of the formulae follows a continuum-based approach (very similar to the one presented in [14]), which yields the following expression for the Cauchy–Green strain tensor:

$$\underline{\underline{\mathbf{C}}} = \underline{\underline{\mathbf{F}}}^T \cdot \underline{\underline{\mathbf{E}}} = \left(\underline{\underline{\mathbf{j}}} \cdot \underline{\underline{\mathbf{J}}}^{-1} \right)^T \cdot \underline{\underline{\mathbf{j}}} \cdot \underline{\underline{\mathbf{J}}}^{-1} = \underline{\underline{\mathbf{G}}}^T \cdot \underline{\underline{\mathbf{g}}} \cdot \underline{\underline{\mathbf{G}}}, \tag{9}$$

where $\underline{\underline{\mathbf{G}}}$ is the matrix of change of coordinates from parametric of the element to global (in the reference configuration) and $\underline{\underline{\mathbf{g}}}$ contains the deformation information. The terms of $\underline{\underline{\mathbf{g}}}$ can be expressed as:

$$g_{ij} = \underline{\underline{\mathbf{L}}}_i^T \cdot \underline{\underline{\mathbf{x}}}^T \cdot \underline{\underline{\mathbf{x}}} \cdot \underline{\underline{\mathbf{L}}}_j + \lambda \xi_3 \underline{\underline{\mathbf{L}}}_i^T \cdot \underline{\underline{\mathbf{x}}}^T \cdot \underline{\underline{\mathbf{U}}} \cdot \underline{\underline{\mathbf{n}}} \cdot \underline{\underline{\mathbf{L}}}_j + \lambda^2 \xi_3^2 \underline{\underline{\mathbf{L}}}_i^T \cdot \underline{\underline{\mathbf{n}}}^T \cdot \underline{\underline{\mathbf{U}}} \cdot \underline{\underline{\mathbf{U}}} \cdot \underline{\underline{\mathbf{n}}} \cdot \underline{\underline{\mathbf{L}}}_j \quad \forall i, j = 1-2, \tag{10}$$

where the matrices $\underline{\underline{\mathbf{L}}}_i$ contain the derivatives of the shape functions, and $\underline{\underline{\mathbf{U}}}$ contains the information regarding the derivatives of the unit normal vectors. The three terms of Eq. (10) are respectively: the membranal deformation, the bending deformation and the so-called second order deformation terms.

Figs. 1 and 2 depict, respectively, the global and parametric coordinates that identify the nodes of the triangles, and the relation between the normals at the nodes and the patch of triangles that surround each node.

The relation between strains and stresses in a linear elastic material is established by the constitutive matrix:

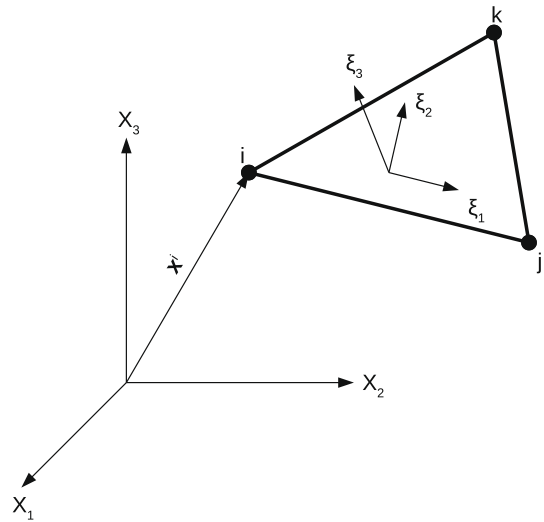


Fig. 1. Representation of the global and parametric coordinates used to define the positions of each of the nodes of the triangle.

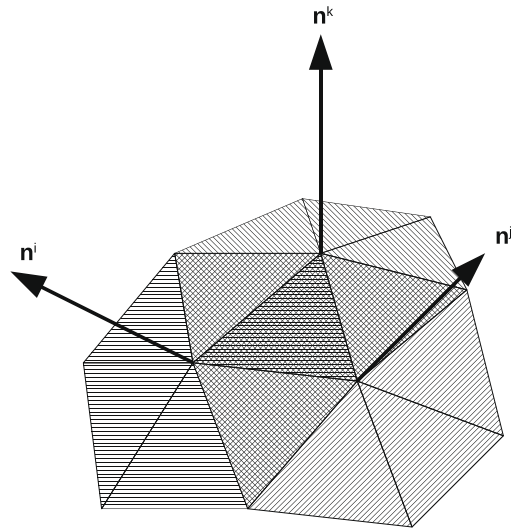


Fig. 2. In order to compute the curvature inside the central triangle, we use the information of the patch of elements that surround each node.

$$\underline{\underline{\mathbf{S}}} = \underline{\underline{\mathbf{D}}} : \underline{\underline{\mathbf{E}}} \Rightarrow \underline{\underline{\mathbf{S}}} = \underline{\underline{\mathbf{D}}} \cdot \underline{\underline{\mathbf{E}}}, \tag{11}$$

where $\underline{\underline{\mathbf{E}}}$ is the Green–Lagrange strain tensor, $\underline{\underline{\mathbf{S}}}$ is the second Piola–Kirchhoff stress tensor, and the ‘s’ indicate that the tensors must be expressed in material local coordinates, since that is the base used by the elasticity matrix $\underline{\underline{\mathbf{D}}}$

$$\underline{\underline{\mathbf{E}}} = \frac{1}{2}(\underline{\underline{\mathbf{C}}} - \underline{\underline{\mathbf{I}}}) = \frac{1}{2}(\underline{\underline{\mathbf{G}}}^T \cdot \underline{\underline{\mathbf{g}}} \cdot \underline{\underline{\mathbf{G}}} - \underline{\underline{\mathbf{I}}}), \tag{12}$$

$$\underline{\underline{\mathbf{E}}} = \underline{\underline{\mathbf{T}}}^T \cdot \underline{\underline{\mathbf{E}}} \cdot \underline{\underline{\mathbf{T}}} = \frac{1}{2}(\underline{\underline{\mathbf{T}}}^T \cdot \underline{\underline{\mathbf{G}}}^T \cdot \underline{\underline{\mathbf{g}}} \cdot \underline{\underline{\mathbf{G}}} \cdot \underline{\underline{\mathbf{T}}} - \underline{\underline{\mathbf{I}}}) = \frac{1}{2}(\underline{\underline{\mathbf{A}}}^T \cdot \underline{\underline{\mathbf{g}}} \cdot \underline{\underline{\mathbf{A}}} - \underline{\underline{\mathbf{I}}}), \tag{13}$$

$$\underline{\underline{\mathbf{A}}} = \underline{\underline{\mathbf{G}}} \cdot \underline{\underline{\mathbf{T}}}, \tag{14}$$

$$\delta \underline{\underline{\mathbf{E}}} = \frac{1}{2} \underline{\underline{\mathbf{A}}}^T \cdot \delta \underline{\underline{\mathbf{g}}} \cdot \underline{\underline{\mathbf{A}}}, \tag{15}$$

where $\underline{\underline{\mathbf{T}}}$ is the matrix of change of coordinates from local to global (in the reference configuration). Thus, $\underline{\underline{\mathbf{A}}}$ results in the matrix of change of coordinates from local material to parametric of the element (in any configuration).

The expression for the internal virtual work can be expressed as:

$$\pi^{int} = \iiint_V \delta E'_{ij} \cdot S'_{ij} \cdot dV = \iiint_V \delta \underline{\underline{E}}' : \underline{\underline{S}}' \cdot dV. \tag{16}$$

Taking advantage of Voigt's notation we can write the following expressions:

$$\begin{aligned} \delta E'_{ij} \cdot S'_{ij} &= \frac{1}{2} A_{ij}^T \cdot \delta g_{ij} \cdot A_{ij} \cdot S'_{ij} = \frac{1}{2} \delta g_{ij} \cdot A_{ij} \cdot A_{ij} \cdot S'_{ij} = \frac{1}{2} \delta g_{ij} \cdot s_{ij} \\ &= \frac{1}{2} \delta \underline{\underline{g}}^T \cdot \underline{\underline{s}}, \end{aligned} \tag{17}$$

$$\underline{\underline{g}} = \begin{bmatrix} g_{11} \\ g_{22} \\ g_{12} + g_{21} \end{bmatrix}, \tag{18}$$

$$\underline{\underline{s}} = \begin{bmatrix} s_{11} \\ s_{22} \\ s_{12} \end{bmatrix} = \underbrace{\begin{bmatrix} A_{11}^2 & A_{12}^2 & 2A_{11}A_{12} \\ A_{21}^2 & A_{22}^2 & 2A_{21}A_{22} \\ A_{11}A_{21} & A_{12}A_{22} & A_{11}A_{22} + A_{12}A_{21} \end{bmatrix}}_{\underline{\underline{Q}}^T} \cdot \begin{bmatrix} S'_{11} \\ S'_{22} \\ S'_{12} \end{bmatrix}, \tag{19}$$

$$\underline{\underline{s}} = \underline{\underline{Q}}^T \cdot \underline{\underline{S}}', \tag{20}$$

$$\delta E'_{ij} \cdot S'_{ij} = \frac{1}{2} \delta g_{ij} \cdot s_{ij} = \frac{1}{2} \delta \underline{\underline{g}}^T \cdot \underline{\underline{Q}}^T \cdot \underline{\underline{S}}' = \frac{1}{2} \delta \underline{\underline{g}}^T \cdot \underline{\underline{Q}}^T \cdot \underline{\underline{D}} \cdot \underline{\underline{E}}', \tag{21}$$

$$\underline{\underline{E}}' = \frac{1}{2} (\underline{\underline{Q}} \cdot \underline{\underline{g}} - \underline{\underline{I}}), \tag{22}$$

$$\underline{\underline{I}} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \tag{23}$$

$$\delta \underline{\underline{g}} = \frac{\partial \underline{\underline{g}}}{\partial \underline{\underline{x}}} \cdot \delta \underline{\underline{x}} = \underline{\underline{B}} \cdot \delta \underline{\underline{x}}, \tag{24}$$

$$\pi^{int} = \frac{1}{4} \iint_S \int_{-\frac{h}{2}}^{\frac{h}{2}} \delta \underline{\underline{x}}^T \cdot \underline{\underline{B}}^T \cdot \underline{\underline{Q}}^T \cdot \underline{\underline{D}} \cdot [\underline{\underline{Q}} \cdot \underline{\underline{g}} - \underline{\underline{I}}] \cdot d\xi_3 \cdot dA, \tag{25}$$

where $\underline{\underline{Q}}$ is defined in Eq. (19).

The development of this first and raw approach does not lead to satisfactory results. The reason behind it is that zero energy modes affect the solution. The first thing that raises our suspicion is that the element as described above is non-conforming. That is, the nor-

mal at each point of the element is not necessarily perpendicular to the surface of the element (see Eq. (8)). We have kept the geometric description of the element intentionally as simple as possible. As a consequence, it is possible for the mesh of triangles to fold like an accordion and not develop significant deformation energy. Note that because of the symmetry of the folds (Fig. 3) the normals barely change direction making their gradient very close to zero. Thus the computed curvature using the first equality in Eq. (6) is also negligible, even if the plane of the triangle differs a lot from the direction of the normals at the nodes.

1.3. Evolving to a higher order description of the triangle

In order to neutralize the instabilities caused by the lack of conformity in the description of the element, we have applied a strategy that consists on increasing the order of the geometrical description so that all the modes of deformation can be represented and thus avoiding the problems associated with the non-conforming formulation. The triangle of lowest order that can interpolate at the same time the positions and the normals at the nodes is a cubic triangle. However, in order to determine unambiguously the ten parameters of a cubic triangle there is not enough data with the three nodes and the three normals. Therefore, choices must be made.

2. Using Bézier triangles

A way to precisely define the geometrical parameters for the cubic triangle without increasing the number of degrees of freedom is adopting the concept of Bézier triangles first defined by de Casteljau [15,16,1]. See Fig. 4.

2.1. Geometric construction of the Bézier triangle

Bézier triangles provide a framework for constructing higher order triangular surfaces. Next we explain the process we have followed and which yields correct results. Other arbitrary choices are not effective in general.

The position for the control points of the net can be evaluated for each point computing the intersection of three planes. For the

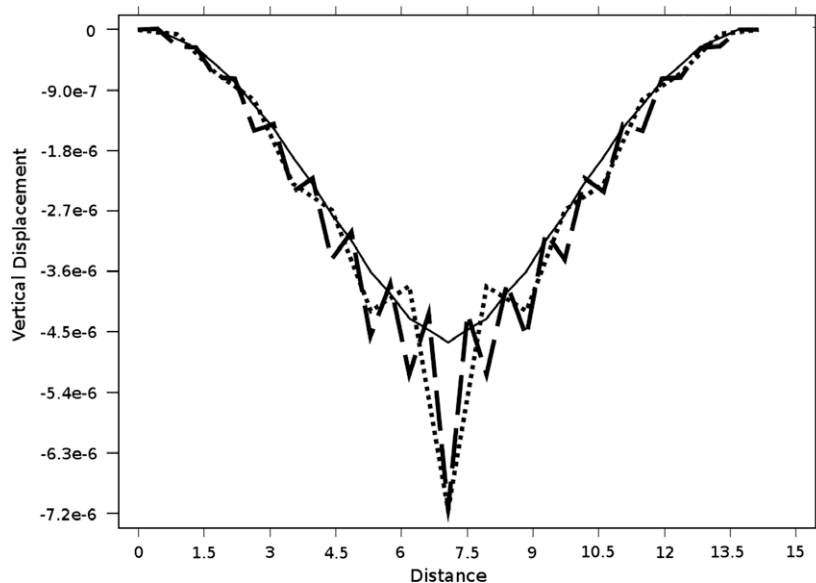


Fig. 3. Numerical example showing the displacements along the diagonals (discontinuous line and dots) of a simply supported square with central point load. The results are compared to the corresponding solution given by the DKT element (solid line).

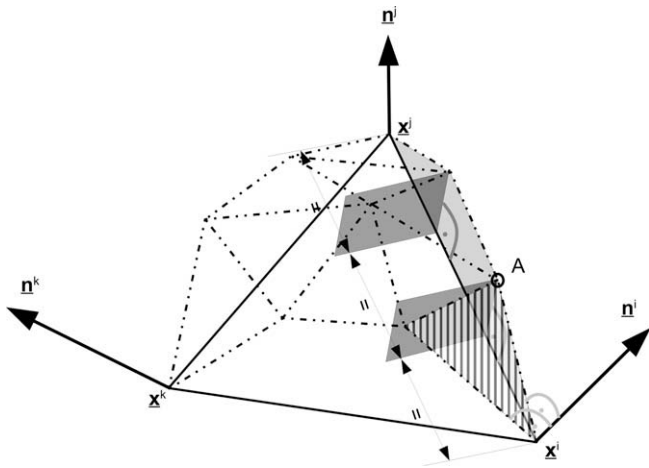


Fig. 4. Representation of the net of control points (dashed) for a cubic Bézier triangle constructed using the nodal positions and normals. Three planes define the position of a control point A: shaded, light grey and dark grey.

contour points, the planes are the ones displayed shaded or darkened in Fig. 4:

- (1) The plane perpendicular to the normal at the vertex – this is a necessary condition to interpolate the normals.
- (2) The plane that contains the curve of the triangle's contour – this is an arbitrary solution to maintain C^0 continuity across the edges of the triangle. The selection of the plane is so that one of the directors is the edge of the flat triangle and the other director is the average of the 2 normals at the nodes. In this way the plane is the same for the two triangles sharing the same edge.
- (3) And a plane perpendicular to the edge of the flat triangle – it can be proven that the plane has to be placed at 1/3 of the edge length. This is a necessary condition so that the element can reproduce a constant curvature along the edge when the size of the element tends to zero.

The central point of the net is obtained as an average of the three candidate points for this position. Each of the three candidate points is computed solving again the intersection of three planes analogously to the control points of the contour:

- (1) A plane perpendicular to the one described in (2) on the left column of this page – this is a symmetrical condition for two triangles sharing the same edge that, if prevails, is sufficient to maintain C^1 continuity across that edge. Cubic Bézier triangles are not general C^1 interpolants. C^1 continuity is only satisfied at the nodes of the triangle. A complete discussion on the conditions to satisfy C^1 continuity in cubic Bézier triangles can be found in [1].
- (2) A plane perpendicular to one of the edges of the flat triangle and passing through the barycenter of the triangle.
- (3) And a plane perpendicular to another edge of the flat triangle and passing through the barycenter of the triangle – this and the previous plane result in an arbitrary condition for maintaining the central control point elevated perpendicularly to the barycenter of the flat triangle.

2.2. Shape functions

Fig. 5 shows a barycentric representation of the cubic Bézier triangle along with the corresponding isoparametric representation. The respective shape functions in barycentric and isoparametric coordinates are as well written in the figure.

In the following we write explicitly the 10 shape functions for the element:

$$\begin{aligned}
 N_1 &= (1 - \xi_1 - \xi_2)^3, & N_2 &= \xi_1^3, & N_3 &= \xi_2^3, \\
 N_4 &= 3 \cdot \xi_1 \cdot (1 - \xi_1 - \xi_2)^2, & N_5 &= 3 \cdot \xi_2 \cdot (1 - \xi_1 - \xi_2)^2, \\
 N_6 &= 3 \cdot \xi_1^2 \cdot \xi_2, & N_7 &= 3 \cdot \xi_1 \cdot (1 - \xi_1 - \xi_2), \\
 N_8 &= 3 \cdot \xi_2^2 \cdot (1 - \xi_1 - \xi_2), & N_9 &= 3 \cdot \xi_1 \cdot \xi_2^2, \\
 N_{10} &= 6 \cdot \xi_1 \cdot \xi_2 \cdot (1 - \xi_1 - \xi_2).
 \end{aligned} \tag{26}$$

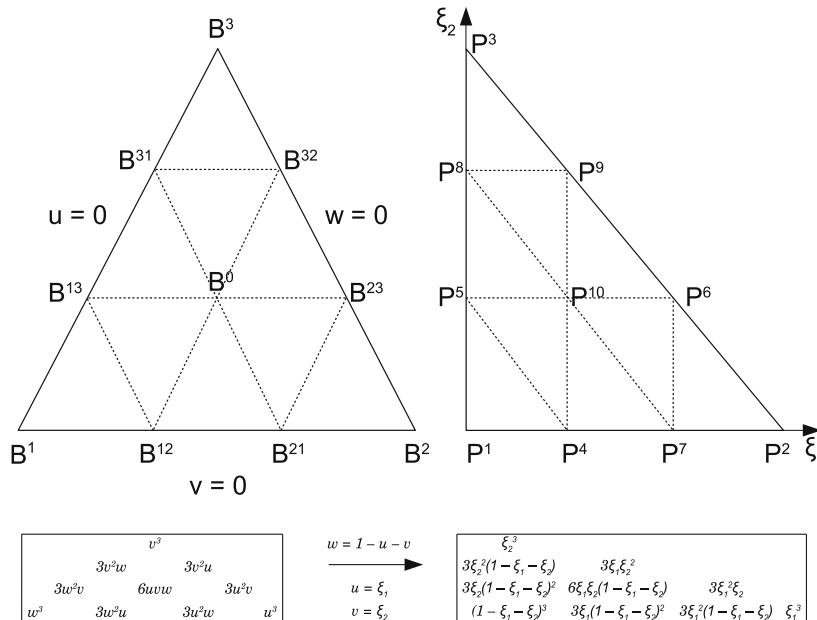


Fig. 5. Left: barycentric representation of the Bézier triangle along with its shape functions corresponding to each of the control points of the net. Right: equivalent representation using isoparametric coordinates. The arrow in the center specifies the transformation used from barycentric to isoparametric coordinates.

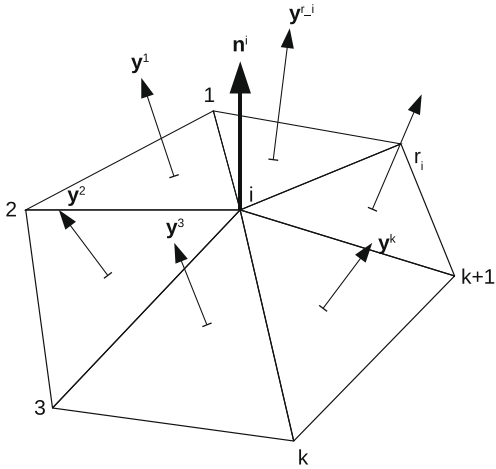


Fig. 6. The normal at node i is estimated using the normals at each of the surrounding triangles. The weights used are the inverse of the areas of each triangle k .

2.3. Selecting the normals

The computation of the normals at the nodes is an arbitrary choice since this is unprescribed data and we must estimate it. We choose to compute the normal at a node as a weighted average of the normals of all the triangles that surround the node. The weights chosen are the inverse of the areas of the corresponding triangles (Fig. 6).

3. Solution strategy

Using the Bézier triangles described in Section 2 we can now develop the formulation presented in Section 1.2.

According to what has been presented, (8) shall be changed by:

$$\underline{\mathbf{x}} = \sum_{i=1}^{10} N_i(\zeta_1, \zeta_2) \cdot \underline{\mathbf{p}}^i + \lambda \zeta_3 \cdot \underline{\mathbf{n}}, \tag{27}$$

where $\underline{\mathbf{p}}^i$ represent the coordinates of each of the control points of the Bézier net, which are found following the procedure described in Section 2.1. Likewise, (7) shall now be changed by:

$$\underline{\mathbf{n}} = \frac{(\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_1) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_2)}{\|(\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_1) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_2)\|}, \tag{28}$$

$$\underline{\mathbf{p}} = [\underline{\mathbf{p}}^1 \ \underline{\mathbf{p}}^2 \ \underline{\mathbf{p}}^3 \ \underline{\mathbf{p}}^4 \ \underline{\mathbf{p}}^5 \ \underline{\mathbf{p}}^6 \ \underline{\mathbf{p}}^7 \ \underline{\mathbf{p}}^8 \ \underline{\mathbf{p}}^9 \ \underline{\mathbf{p}}^{10}]. \tag{29}$$

As a result, now (10) becomes:

$$g_{ij} = \underline{\mathbf{L}}_i^T \cdot \underline{\mathbf{p}}^T \cdot \underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_j + \lambda \zeta_3 \cdot (\underline{\mathbf{L}}_i^T \cdot \underline{\mathbf{p}}^T \cdot \underline{\mathbf{U}}_j + \underline{\mathbf{L}}_j^T \cdot \underline{\mathbf{p}}^T \cdot \underline{\mathbf{U}}_i) + \lambda^2 \zeta_3^2 \cdot \underline{\mathbf{U}}_i^T \cdot \underline{\mathbf{U}}_j \quad \forall i, j = 1-2, \tag{30}$$

where

$$\underline{\mathbf{U}}_1 = (\underline{\mathbf{I}} - \underline{\mathbf{n}} \cdot \underline{\mathbf{n}}^T) \cdot \frac{(\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_{1,1}) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_2) + (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_1) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_{2,1})}{\|(\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_1) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_2)\|}, \tag{31}$$

$$\underline{\mathbf{U}}_2 = (\underline{\mathbf{I}} - \underline{\mathbf{n}} \cdot \underline{\mathbf{n}}^T) \cdot \frac{(\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_{1,2}) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_2) + (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_1) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_{2,2})}{\|(\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_1) \times (\underline{\mathbf{p}} \cdot \underline{\mathbf{L}}_2)\|}. \tag{32}$$

It can be observed that by using the current approach, all the terms of the deformation tensor $\underline{\mathbf{g}}$ of Eq. (30) can be taken into account (including the second order terms) without increasing the complexity of computations significantly.

3.1. Numerical integration

Since the interpolation functions are cubic, we will use four integration points for the surface of the triangles. Taking a look at equations. (30) and (34) we see that the thickness coordinate ζ_3 is elevated to the power of four in Eq. (33), and the same holds for Eq. (44). However, the energy associated with the highest order is relatively small in comparison with the rest of energy modes. Thus, we will use initially only a cubic integration scheme also thickness-wise: two points across the thickness. Nevertheless, for laminated shells we use an appropriate number of integration points across the thickness. For homogeneous material, the isoparametric coordinates of the eight points and the respective weights for the Gauss quadrature are specified in Table 1. The position of these points is displayed in Fig. 7.

3.2. Tangent stiffness matrix

Using Eq. (25) we can write the Principle of Virtual Work as:

$$\begin{aligned} \iiint_V \delta \underline{\mathbf{E}} : \underline{\mathbf{S}} \cdot dV &= \frac{1}{4} \iint_S \int_{-\frac{h}{2}}^{\frac{h}{2}} \delta \underline{\mathbf{x}}^T \cdot \underline{\mathbf{B}}^T \cdot \underline{\mathbf{Q}}^T \cdot \underline{\mathbf{D}} \cdot \underline{\mathbf{Q}} \cdot \underline{\mathbf{g}} - \underline{\mathbf{I}} \cdot d\zeta_3 \cdot dA \\ &= \iint_V \delta \underline{\mathbf{x}}^T \cdot \underline{\mathbf{b}} \cdot dV + \iint_S \delta \underline{\mathbf{x}}^T \cdot \underline{\mathbf{q}} \cdot dA \\ &\quad + \int_{\partial S} \delta \underline{\mathbf{x}}^T \cdot \underline{\mathbf{t}} \cdot d\Gamma + \sum \delta \underline{\mathbf{x}}^T \cdot \underline{\mathbf{F}}, \end{aligned} \tag{33}$$

where $\underline{\mathbf{b}}$, $\underline{\mathbf{q}}$, $\underline{\mathbf{t}}$, and $\underline{\mathbf{F}}$ represent, respectively, forces applied on the volume of the elements, the surface of the elements, the edges of

Table 1

Weights and coordinates for cubic integration using Gauss quadrature in a triangular domain.

Gauss point	Weight	Parametric coordinates
1	$-\frac{27}{96}$	$(\frac{1}{3}, \frac{1}{3}, -\frac{1}{\sqrt{3}})$
2	$-\frac{27}{96}$	$(\frac{1}{3}, \frac{1}{3}, +\frac{1}{\sqrt{3}})$
3	$+\frac{25}{96}$	$(\frac{1}{5}, \frac{1}{5}, -\frac{1}{\sqrt{3}})$
4	$+\frac{25}{96}$	$(\frac{1}{5}, \frac{1}{5}, +\frac{1}{\sqrt{3}})$
5	$+\frac{25}{96}$	$(\frac{1}{5}, \frac{3}{5}, -\frac{1}{\sqrt{3}})$
6	$+\frac{25}{96}$	$(\frac{1}{5}, \frac{3}{5}, +\frac{1}{\sqrt{3}})$
7	$+\frac{25}{96}$	$(\frac{3}{5}, \frac{1}{5}, -\frac{1}{\sqrt{3}})$
8	$+\frac{25}{96}$	$(\frac{3}{5}, \frac{1}{5}, +\frac{1}{\sqrt{3}})$

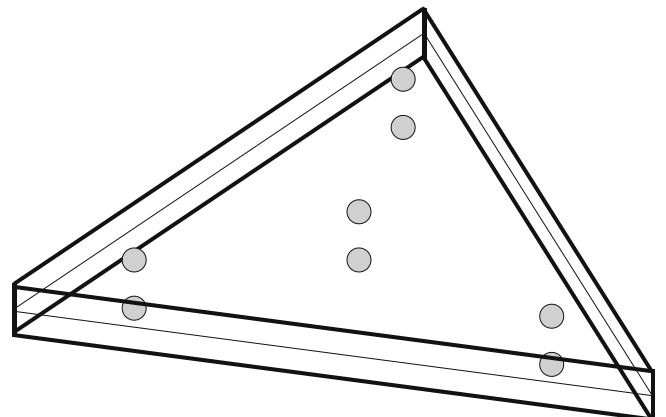


Fig. 7. Cubic integration in the triangular domain requires four points over the surface times two points across the thickness, totaling eight points.

the elements, or the nodes of the elements, respectively. Matrix $\underline{\underline{B}}$, first introduced in Eq. (24), can be computed as follows:

$$\underline{\underline{B}} = \frac{\partial \underline{\underline{g}}}{\partial \underline{\underline{x}}} = \frac{\partial \underline{\underline{g}}}{\partial \underline{\underline{p}}} : \frac{\partial \underline{\underline{p}}}{\partial \underline{\underline{x}}} = \left(\frac{\partial \underline{\underline{g}}_m}{\partial \underline{\underline{p}}} + 2\lambda \xi_3 \cdot \frac{\partial \underline{\underline{g}}_f}{\partial \underline{\underline{p}}} + \lambda^2 \xi_3^2 \cdot \frac{\partial \underline{\underline{g}}_n}{\partial \underline{\underline{p}}} \right) : \frac{\partial \underline{\underline{p}}}{\partial \underline{\underline{x}}} \quad (34)$$

Here we are not taking into account the derivatives of λ because its value changes very little. But in *strictu sensu* we can not interpret that λ remains constant with respect to the nodal displacements. Having said that, we carry on developing further the different terms in Eq. (34)

$$\frac{\partial \underline{\underline{g}}_{m(1)}}{\partial p_{(ij)}} = 2 \cdot p_{(ix)} \cdot L_{1(x)} \cdot L_{1(j)}, \quad (35)$$

$$\frac{\partial \underline{\underline{g}}_{m(2)}}{\partial p_{(ij)}} = 2 \cdot p_{(ix)} \cdot L_{2(x)} \cdot L_{2(j)}, \quad (36)$$

$$\frac{\partial \underline{\underline{g}}_{m(3)}}{\partial p_{(ij)}} = 2 \cdot p_{(ix)} \cdot (L_{1(x)} \cdot L_{2(j)} + L_{2(x)} \cdot L_{1(j)}), \quad (37)$$

$$\frac{\partial \underline{\underline{g}}_{f(1)}}{\partial p_{(ij)}} = U_{1(i)} \cdot L_{1(j)} + U_{1,p(xij)} \cdot p_{(x\beta)} \cdot L_{1(\beta)}, \quad (38)$$

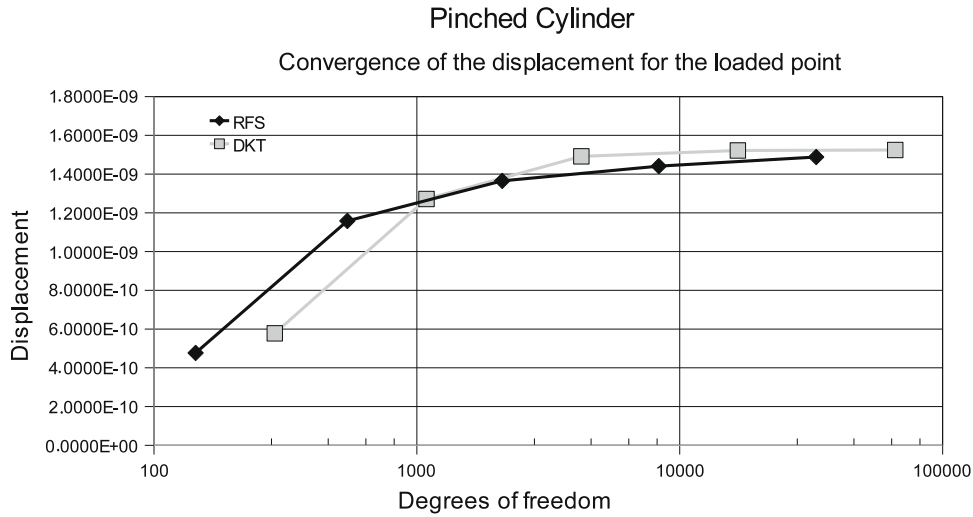


Fig. 8. Pinched cylinder: convergence of the vertical displacement for the loaded node with the number of degrees of freedom. Results are shown for the RFS and the DKT elements.

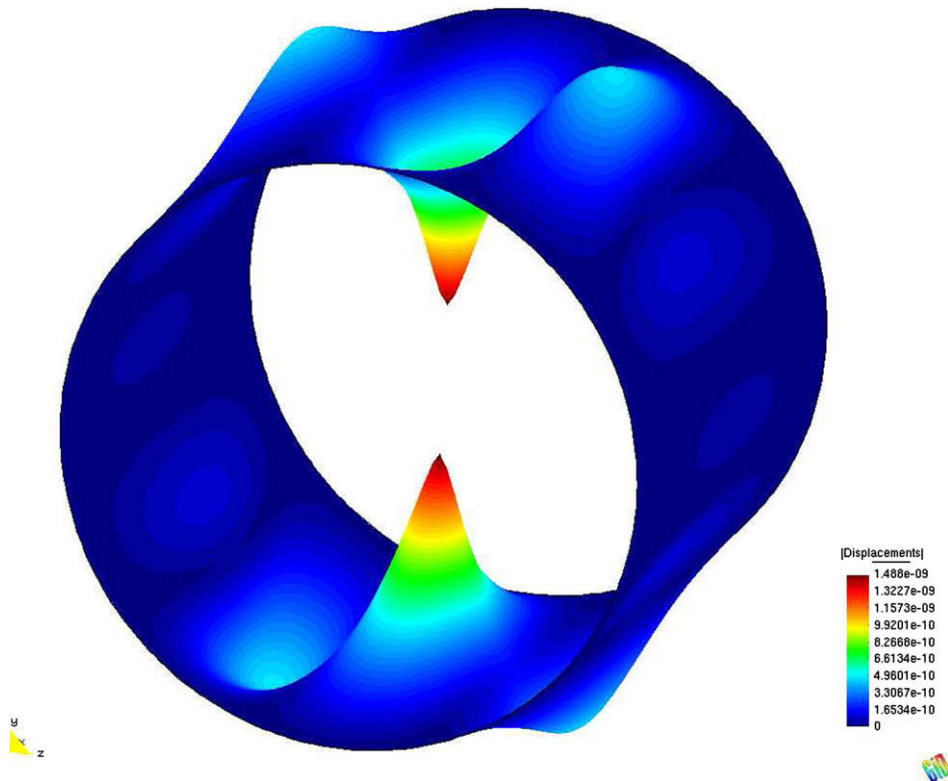


Fig. 9. Contour plot of the displacements of the cylinder. The deformations have been scaled for better viewing.

$$\frac{\partial g_{f(2)}}{\partial p_{(ij)}} = U_{2(i)} \cdot L_{2(j)} + U_{2,p(\alpha ij)} \cdot p_{(\alpha\beta)} \cdot L_{2(\beta)}, \quad (39)$$

$$\begin{aligned} \frac{\partial g_{f(3)}}{\partial p_{(ij)}} &= U_{1(i)} \cdot L_{2(j)} + U_{1,p(\alpha ij)} \cdot p_{(\alpha\beta)} \cdot L_{2(\beta)} \\ &+ U_{2(i)} \cdot L_{1(j)} + U_{2,p(\alpha ij)} \cdot p_{(\alpha\beta)} \cdot L_{1(\beta)}, \end{aligned} \quad (40)$$

$$\frac{\partial g_{n(1)}}{\partial p_{(ij)}} = 2 \cdot U_{1(\alpha)} \cdot U_{1,p(\alpha ij)}, \quad (41)$$

$$\frac{\partial g_{n(2)}}{\partial p_{(ij)}} = 2 \cdot U_{2(\alpha)} \cdot U_{2,p(\alpha ij)}, \quad (42)$$

$$\frac{\partial g_{n(3)}}{\partial p_{(ij)}} = 2 \cdot (U_{1(\alpha)} \cdot U_{2,p(\alpha ij)} + U_{2(\alpha)} \cdot U_{1,p(\alpha ij)}). \quad (43)$$

Finally, we present the equation for the element stiffness matrix, composed of the material stiffness matrix $\underline{\underline{K}}_M$ and the geometric stiffness matrix $\underline{\underline{K}}_G$:

$$\underline{\underline{K}} = \underline{\underline{K}}_M + \underline{\underline{K}}_G, \quad (44)$$

$$\underline{\underline{K}}_M = \frac{1}{4} \iint_A \int_{-\frac{h}{2}}^{\frac{h}{2}} \underline{\underline{B}}^T \cdot \underline{\underline{Q}}^T \cdot \underline{\underline{D}} \cdot \underline{\underline{Q}} \cdot \underline{\underline{B}} \cdot d\xi_3 \cdot dA, \quad (45)$$

$$\underline{\underline{K}}_G = \frac{1}{2} \iint_A \int_{-\frac{h}{2}}^{\frac{h}{2}} \left(\frac{\partial \underline{\underline{B}}}{\partial \underline{\underline{x}}} \right)^T \cdot \underline{\underline{Q}}^T \cdot \underline{\underline{D}} \cdot \underline{\underline{E}}' \cdot d\xi_3 \cdot dA. \quad (46)$$

Matrix $\underline{\underline{K}}_G$ is essential for solving geometrically non-linear problems. This topic will be addressed in a forthcoming publication.

4. Examples

In the following examples we show the solution of the linear problem, therefore carrying out only one step of the classical Newton–Raphson loop for solving non-linear problems. However the formulation presented is fully applicable to a non-linear solution using the Newton–Raphson scheme.

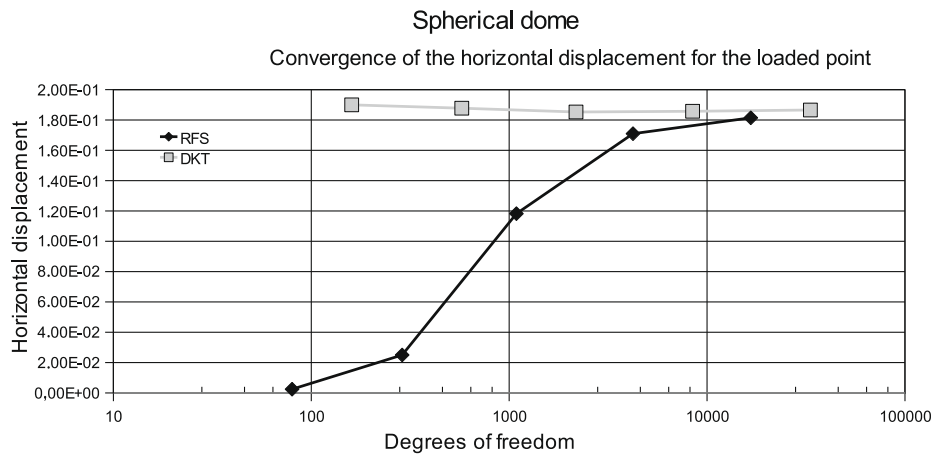


Fig. 10. Spherical dome: convergence of the horizontal displacement for the loaded node with the number of degrees of freedom. Results are shown for the RFS and the DKT elements.

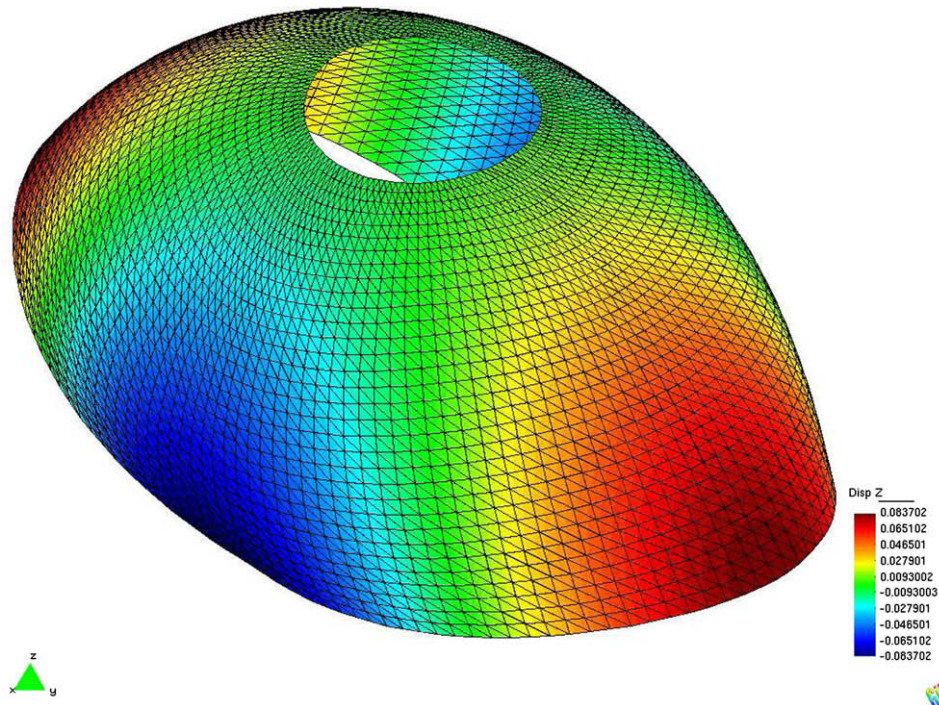


Fig. 11. Contour plot of the vertical displacements of the dome. The deformations have been scaled for better viewing.

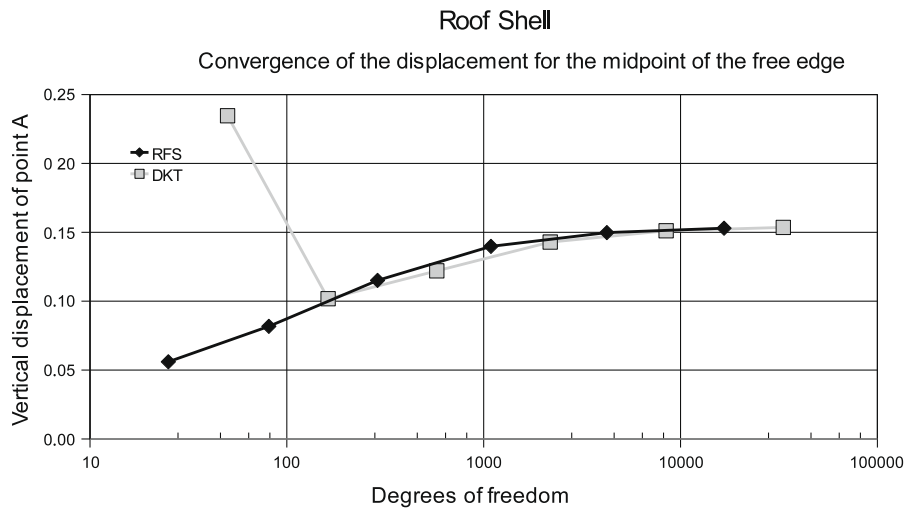


Fig. 12. Cylindrical roof: convergence of the vertical displacement for the free edge's midpoint node with the number of elements. Results are shown for the RFS and the DKT elements.

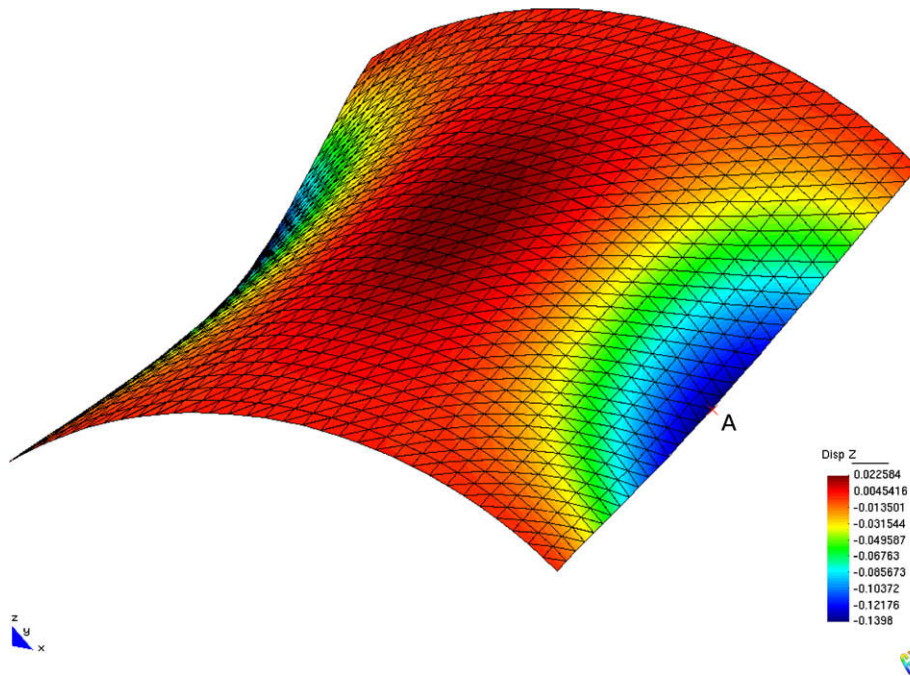


Fig. 13. Contour plot of the vertical displacements of the roof. The deformations have been scaled for better viewing.

In all examples we compare the solutions obtained using the current formulation (denoted RFS for Rotation Free Shell) with those obtained by a classical DKT formulation using the same mesh. We note that in every example, there are half as many degrees of freedom for the presented formulation as for the DKT element.

4.1. Pinched cylinder

We present here the structural response of the classical pinched cylinder with $R = 3.0$ m, $L = 6.0$ m and $h = 0.03$ m. The Young modulus of the material is $E = 300 \times 10^{10}$ Pa and Poisson's ratio is $\nu = 0.3$. The cylinder is simply supported on rigid diaphragms at both ends and is subjected to a pair of point loads diametrically opposed and applied at the center of the cylinder $P = 1.0$ N. Fig. 8

shows the convergence of the displacement of the loaded node with the number of degrees of freedom. A contour plot of the displacement field amplified is shown in Fig. 9.

4.2. Spherical Dome

In this example we present the deformation of a semi-spherical dome ($R = 10.0$ m, $h = 0.04$ m, $E = 6.825 \times 10^7$ Pa, $\nu = 0.3$) that has an 18° hole in its center. The shell is subjected to two pairs of forces ($P = 4.0$ N) diametrically opposed at the base of the dome: two for traction and two for compression. To avoid a rigid body motion in the vertical direction we fix four points equidistant from the loaded points in the base. Fig. 10 shows the convergence of the horizontal displacement of the loaded node with the number of degrees of freedom. In this example where the dominant curva-

ture is spherical, we observe that the RFS element lacks performance. A contour plot of the vertical displacement field is shown in Fig. 11.

4.3. Roof shell

In this final example we test the RFS element to solve a cylindrical roof. This problem is very similar to the one proposed by Scordelis–Lo, with the difference on the boundary conditions applied. Thus, in the present example the diaphragms at both ends are fully fixed whereas in the original problem the diaphragms are free to move in the longitudinal direction. The geometry is defined by an 80° cylindrical shell of $R = 25.0$ m, $L = 50.0$ m and $h = 0.25$ m. The Young modulus is $E = 4.32 \times 10^8$ Pa and Poisson's ratio $\nu = 0.0$. We model the full geometry without taking into account the symmetries. The roof is loaded with a uniform distributed load $q = 90.0$ Pa. Fig. 12 shows the convergence of the vertical displacement of the free edge's midpoint with the number of degrees of freedom. In this example where the dominant curvature is cylindrical, we observe that the RFS element shows a very good convergence pattern. A contour plot of the vertical displacement field is shown in Fig. 13.

5. Conclusions

This paper presents the development of a new rotation-free shell element. The element is free from rotational degrees of freedom and the solution is found using just the displacements of the mesh nodes as nodal variables. The approach is general and can be applied to any mesh topology. The results are very satisfactory.

On the numerical aspects of the present work, we note that by using only displacement degrees of freedom, we reduce considerably the size of the system matrix for any given mesh. The tradeoff is an increase of the system matrix's bandwidth. The resulting bandwidth depends on the mesh topology, that is, the amount of elements surrounding any given node. For a regular triangular mesh (each node surrounded by six elements), the element's system matrix has 36 degrees of freedom.

Numerical examples are presented showing a monotonous convergence of the solution. However, the convergence rate should be improved to make this element competitive with other existing formulations. We pinpoint possible areas for improvement:

- Further work is needed to study the computation of the normals at the nodes. We acknowledge that the selection made is arbitrary and an optimum solution has not been found. A preliminary study on geometry hints that the areas to be taken into account when averaging the normals should not be those of the triangles but rather those of the circumscribed circles of the triangles. However this is yet to be confirmed.
- We need to improve the capability of the element to represent constant curvatures not only in the limit of the element's size

becoming zero, but for the more general case. This has also been acknowledged in the dissertation and our future work will try to solve this issue.

- Also the real need to use cubic integration over the triangular domain of the element should be studied. It would be to the advantage of the presented formulation that no significant accuracy was lost when using reduced integration.
- Finally, there is also interest in developing a technique that allows computing shells with folds or branches. The goal would be to take into account the different normals that a node belonging to the fold line would have depending on the side of the line considered.

Acknowledgement

This research was partially supported by project SEDUREC of the CONSOLIDER Programme of the Spanish Ministry of Science and Innovation.

References

- [1] G. Farin, Curves and Surfaces for CAGD. A Practical Guide, fifth ed., Morgan Kaufmann Publishers, San Francisco, CA, 2002.
- [2] E. Oñate, F. Zárate, Rotation-free triangular plate and shell elements, *Int. J. Numer. Methods Engrg.* 47 (2000) 557–603.
- [3] E. Oñate, Structural Analysis with the Finite Element Method, Plates and Shells, vol. 2, CIMNE-Springer, 2009.
- [4] O. Zienkiewicz, R. Taylor, Finite Element Method, fifth ed., Butterworth-Heinemann, Oxford, UK, 2000.
- [5] F. Cirak, M. Ortiz, P. Schröder, Subdivision surfaces: a new paradigm for thin shell finite-element analysis, *Int. J. Numer. Methods Engrg.* 47 (12) (2000) 2039–2072.
- [6] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comput. Methods Appl. Mech. Engrg.* 194 (39–41) (2005) 4135–4195.
- [7] F.G. Flores, E. Oñate, Improvements in the membrane behaviour of the three node rotation-free BST shell triangle using an assumed strain approach, *Comput. Methods Appl. Mech. Engrg.* 194 (6–8) (2005) 907–932.
- [8] E. Oñate, F. Flores, Advances in the formulation of the rotation-free basic shell triangle, *Comput. Methods Appl. Mech. Engrg.* 194 (21–24) (2005) 2406–2443.
- [9] P.-A. Ubach, E. Oñate, New rotation-free composite shell triangle using accurate geometrical data, in: WCCM VII. 7th World Congress on Computational Mechanics, vol. CD-ROM, UCLA, IACM, Northwestern University, Los Angeles, California, USA, 2006.
- [10] P.-A. Ubach, E. Oñate, Advances of the new rotation-free finite element shell triangle using accurate geometrical data, in: 9th US National Congress on Computational Mechanics, vol. CD-ROM, University of California at Berkeley, USACM, San Francisco, California, USA, 2007.
- [11] J. Linhard, K.-U. Bletzinger, M. Firl, Upgrading membranes to shells – the CEG rotation free shell element and its applications, in: 9th US National Congress on Computational Mechanics, vol. CD-ROM, University of California at Berkeley, USACM, San Francisco, California, USA, 2007.
- [12] E. Kreyszig, Differential Geometry, Dover, 1991.
- [13] D. Struik, Lectures on Classical Differential Geometry, second ed., Dover, 1988.
- [14] R. Taylor, E. Oñate, P.-A. Ubach, Textile composites and inflatable structures, Computational Methods in Applied Sciences, vol. 3, Springer, The Netherlands, 2005, Chapter Finite Element Analysis of Membrane Structures, pp. 47–68.
- [15] P. de Casteljaou, Outillages méthodes calcul, Tech. rep., A. Citroën, Paris, 1959.
- [16] P. de Casteljaou, Courbes et surfaces à pôles, Tech. rep., A. Citroën, Paris, 1963.