Available online at www.sciencedirect.com
ScienceDirect

Comput. Methods Appl. Mech. Engrg. 310 (2016) 98–111

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Cut finite element modeling of linear membranes

Mirza Cenanovic[a,*], Peter Hansbo[a], Mats G. Larson[b]

[a] *Department of Mechanical Engineering, Jönköping University, SE-551 11 Jönköping, Sweden*
[b] *Department of Mathematics and Mathematical Statistics, Umeå University, SE-901 87 Umeå, Sweden*

Received 10 November 2015; received in revised form 17 March 2016; accepted 16 May 2016
Available online 6 July 2016

Abstract

We construct a cut finite element method for the membrane elasticity problem on an embedded mesh using tangential differential calculus, i.e., with the equilibrium equations pointwise projected onto the tangent plane of the surface to create a pointwise planar problem in the tangential direction. Both free membranes and membranes coupled to 3D elasticity are considered. The discretization of the membrane comes from a Galerkin method using the restriction of 3D basis functions (linear or trilinear) to the surface representing the membrane. In the case of coupling to 3D elasticity, we view the membrane as giving additional stiffness contributions to the standard stiffness matrix resulting from the discretization of the three-dimensional continuum.

Keywords: Cut finite element method; Membrane shell; Embedded membrane; Tangential derivative

1. Introduction

In this paper we construct finite element methods for linearly elastic membranes embedded in three dimensional space meshed by tetrahedral or hexahedral elements. These meshes do not in general align with the surface of the membrane which instead cuts through the elements. For the modeling of the membrane problems we use tangential differential calculus, introduced for the modeling of surface stresses by Gurtin and Murdoch [1] and for shell models by Delfour and Zolésio [2]. The tangential approach was pioneered for use in finite element methods by Dziuk [3] for discretizing the Laplace–Beltrami operator on meshed surfaces, and has become a standard method of developing discrete schemes on surfaces, cf. Dziuk and Elliott [4] and references therein. The approach was subsequently employed by Hansbo and Larson [5] for meshed membranes, and the aim of this paper is to extend this work following Olshanskii, Reusken, and Grande [6] and construct a Galerkin method by using restrictions of the 3D basis functions defined on the three-dimensional mesh to the surface. This approach can lead to severe ill conditioning, so we adapt a stabilization technique proposed by Burman, Hansbo, and Larson [7] for the Laplace–Beltrami operator to the membrane problem.

[*] Corresponding author.
E-mail addresses: mirza.cenanovic@ju.se (M. Cenanovic), peter.hansbo@ju.se (P. Hansbo), mats.larson@umu.se (M.G. Larson).

http://dx.doi.org/10.1016/j.cma.2016.05.018

The main application that we have in mind is the coupling of membranes to 3D elasticity. This allows for the modeling of reinforcements, such as shear strengthening and adhesive layers. We emphasize, however, that the mechanical modeling herein is restricted in the sense that we simply add membrane stiffness to a continuous 3D approximation. For more accurate mechanical modeling, say of adhesives, the 3D mesh must also be cut to incorporate, e.g., the imperfect bonding approach of Hansbo and Hansbo [8], allowing for relative motion of the continuum on either side of the adhesive. This extension is not explored in this paper but has been considered in a discontinuous Galerkin setting in [9]. The idea of adding stiffness from lower-dimensional structures is a classical approach, cf. Zienkiewicz [10, Chapter 7.9], using element sides or edges as lower dimensional entities. Letting the membranes cut through the elements in an arbitrary fashion considerably increases the practical modeling possibilities.

The paper is organized as follows: in Section 2 we introduce the membrane model problem and the finite element method for membranes and embedded membranes; in Section 3 we describe the implementation details of the method; and in Section 4 we present numerical results.

2. The membrane model and finite element method

2.1. Tangential calculus

In what follows, $\varGamma$ denotes an oriented surface, which is embedded in $\mathbb{R}^3$ and equipped with exterior normal $\boldsymbol{n}_\varGamma$. The boundary of $\varGamma$ consists of two parts, $\partial\varGamma_{\mathrm{N}}$, where zero traction boundary conditions are assumed, and $\partial\varGamma_{\mathrm{D}}$ where zero Dirichlet boundary conditions are assumed.

We let ρ denote the signed distance function fulfilling $\nabla\rho|_\varGamma = \boldsymbol{n}_\varGamma$.

For a given function $u : \varGamma \rightarrow \mathbb{R}$ we assume that there exists an extension $\bar{u}$, in some neighborhood of $\varGamma$, such that $\bar{u}|_\varGamma = u$. Then the tangent gradient $\nabla_\varGamma$ on $\varGamma$ can be defined by

$$\nabla_\varGamma u = \boldsymbol{P}_\varGamma \nabla \overline{u} \tag{1}$$

with ∇ the $\mathbb{R}^3$ gradient and $\boldsymbol{P}_\varGamma = \boldsymbol{P}_\varGamma(\boldsymbol{x})$ the orthogonal projection of $\mathbb{R}^3$ onto the tangent plane of $\varGamma$ at $\boldsymbol{x} \in \varGamma$ given by

$$\boldsymbol{P}_\varGamma = \boldsymbol{I} - \boldsymbol{n}_\varGamma \otimes \boldsymbol{n}_\varGamma \tag{2}$$

where $\boldsymbol{I}$ is the identity matrix. The tangent gradient defined by (1) is easily shown to be independent of the extension $\overline{u}$. In the following, we shall consequently not make the distinction between functions on $\varGamma$ and their extensions when defining differential operators.

The surface gradient has three components, which we shall denote by

$$\nabla_\Sigma u =: \left(\frac{\partial u}{\partial x^\varGamma}, \frac{\partial u}{\partial y^\varGamma}, \frac{\partial u}{\partial z^\varGamma}\right).$$

For a vector valued function $\boldsymbol{v}(\boldsymbol{x})$, we define the tangential Jacobian matrix as the transpose of the outer product of $\nabla_\varGamma$ and $\boldsymbol{v}$,

$$(\nabla_\varGamma \otimes \boldsymbol{v})^{\mathrm{T}} := \begin{bmatrix} \dfrac{\partial v_1}{\partial x^\varGamma} & \dfrac{\partial v_1}{\partial y^\varGamma} & \dfrac{\partial v_1}{\partial z^\varGamma} \\ \dfrac{\partial v_2}{\partial x^\varGamma} & \dfrac{\partial v_2}{\partial y^\varGamma} & \dfrac{\partial v_2}{\partial z^\varGamma} \\ \dfrac{\partial v_3}{\partial x^\varGamma} & \dfrac{\partial v_3}{\partial y^\varGamma} & \dfrac{\partial v_3}{\partial z^\varGamma} \end{bmatrix},$$

the surface divergence $\nabla_\varGamma \cdot \boldsymbol{v} := \operatorname{tr}\nabla_\varGamma \otimes \boldsymbol{v}$, and the in-plane strain tensor

$$\boldsymbol{\varepsilon}_\varGamma(\boldsymbol{u}) := \boldsymbol{P}_\varGamma \boldsymbol{\varepsilon}(\boldsymbol{u}) \boldsymbol{P}_\varGamma, \quad \text{where } \boldsymbol{\varepsilon}(\boldsymbol{u}) := \frac{1}{2}\left(\nabla \otimes \boldsymbol{u} + (\nabla \otimes \boldsymbol{u})^{\mathrm{T}}\right)$$

is the 3D strain tensor.

2.2. The membrane model

We consider, following [5], the problem of finding $\boldsymbol{u} : \Gamma \to \mathbb{R}^3$ such that

$$\begin{aligned} -\nabla_\Gamma \cdot \boldsymbol{\sigma}_\Gamma(\boldsymbol{u}) &= \boldsymbol{f} \quad \text{on } \Gamma, \\ \boldsymbol{\sigma}_\Gamma = 2\mu\boldsymbol{\varepsilon}_\Gamma + \lambda \mathrm{tr}\boldsymbol{\varepsilon}_\Gamma \boldsymbol{P}_\Gamma & \quad \text{on } \Gamma, \end{aligned} \tag{3}$$

where $\boldsymbol{f} : \Gamma \to \mathbb{R}^3$ is a load per unit area and, with Young's modulus E and Poisson's ratio ν,

$$\mu := \frac{E}{2(1+\nu)}, \qquad \lambda := \frac{E\nu}{1-\nu^2}$$

are the Lamé parameters in plane stress. As discussed in [5], these equations can be derived from minimization of the surface potential energy functional

$$\Pi_\Gamma(\boldsymbol{u}) := \frac{1}{2}\int_\Gamma \boldsymbol{\sigma}_\Gamma(\boldsymbol{u}) : \boldsymbol{\varepsilon}_\Gamma(\boldsymbol{u}) d\Gamma - \int_\Gamma \boldsymbol{f} \cdot \boldsymbol{u}\, d\Gamma$$

and assuming the material obeys Hooke's law in plane stress. Splitting the displacement into a normal part $u_\mathrm{N} := \boldsymbol{u} \cdot \boldsymbol{n}_\Gamma$ and a tangential part $\boldsymbol{u}_\mathrm{T} := \boldsymbol{u} - u_\mathrm{N}\boldsymbol{n}_\Gamma$, the corresponding weak statement takes the form: find

$$\boldsymbol{u} \in V := \{\boldsymbol{v} : v_\mathrm{N} \in L_2(\Gamma) \text{ and } \boldsymbol{v}_\mathrm{T} \in [H^1(\Gamma)]^2,\ \boldsymbol{v} = \boldsymbol{0} \text{ on } \Gamma_\mathrm{D}\},$$

such that

$$a(\boldsymbol{u}, \boldsymbol{v}) = l(\boldsymbol{v}), \quad \forall \boldsymbol{v} \in V, \tag{4}$$

where

$$a(\boldsymbol{u}, \boldsymbol{v}) = (2\mu\varepsilon_\Gamma(\boldsymbol{u}), \varepsilon_\Gamma(\boldsymbol{v}))_\Gamma + (\lambda\nabla_\Gamma \cdot \boldsymbol{u}, \nabla_\Gamma \cdot \boldsymbol{v})_\Gamma, \qquad l(\boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v})_\Gamma,$$

and

$$(\boldsymbol{v}, \boldsymbol{w})_\Gamma = \int_\Gamma \boldsymbol{v} \cdot \boldsymbol{w}\, \mathrm{d}\Gamma \quad \text{and} \quad (\varepsilon_\Gamma(\boldsymbol{v}), \varepsilon_\Gamma(\boldsymbol{w}))_\Gamma = \int_\Gamma \varepsilon_\Gamma(\boldsymbol{v}) : \varepsilon_\Gamma(\boldsymbol{w})\, \mathrm{d}\Gamma$$

are the L^2 inner products.

2.3. The cut finite element method

Let $\widetilde{\mathcal{T}}_h$ be a quasi uniform mesh, with mesh parameter $0 < h \le h_0$, into shape regular tetrahedra (hexahedra will be briefly discussed in Section 3) of an open and bounded domain Ω in $\mathbb{R}^3$ completely containing Γ. On $\widetilde{\mathcal{T}}_h$, let ϕ be a continuous, piecewise linear approximation of the distance function ρ and define the discrete surface Γ_h as the zero level set of ϕ; that is

$$\Gamma_h = \{\boldsymbol{x} \in \Omega : \phi(\boldsymbol{x}) = 0\}. \tag{5}$$

Note that it is not necessary to use a distance function to represent the surface; however, ϕ being a distance function ensures that the zero isoline of ϕ is insensitive to small perturbations. It can also be beneficial in cases where the surfaces are moving, cf. [11].

We note that Γ_h is a polygon with flat faces and we let $\boldsymbol{n}_h$ be the piecewise constant exterior unit normal to Γ_h. For the mesh $\widetilde{\mathcal{T}}_h$, we define the active background mesh by

$$\mathcal{T}_h = \{T \in \widetilde{\mathcal{T}}_h : \overline{T} \cap \Gamma_h \neq \emptyset\} \tag{6}$$

cf. Fig. 2, and its set of interior faces by

$$\mathcal{F}_h = \{F = T^+ \cap T^- : T^+, T^- \in \mathcal{T}_h\}. \tag{7}$$

The face normals $\boldsymbol{n}_F^+$ and $\boldsymbol{n}_F^-$ are then given by the unit normal vectors which are perpendicular on F and are pointing exterior to T^+ and T^-, respectively. We observe that the active background mesh $\mathcal{T}_h$ gives rise to a neighborhood of

Γ_h, which we denote by $\Omega_h := \cup_{T \in \mathcal{T}_h} T$ with boundary $\partial\Omega_h$ consisting of those element faces F that are exterior to $\mathcal{T}_h$. Note that for all elements $T \in \mathcal{T}_h$ there is a neighbor $T' \in \mathcal{T}_h$ such that T and T' share a face. We denote by $\partial\Omega_{h,\mathrm{D}}$ the boundary of Ω_h intersected by $\Gamma_{h,\mathrm{D}}$

Finally, let $\widetilde{V}_h$ denote the space of continuous piecewise linear (or trilinear) polynomials defined on $\widetilde{\mathcal{T}}_h$ and

$$V_h = \left\{ \boldsymbol{v} \in [\widetilde{V}_h|_{\Omega_h}]^3 : \ \boldsymbol{v} = \boldsymbol{0} \text{ on } \partial\Omega_{h,\mathrm{D}} \right\} \tag{8}$$

be the space of continuous piecewise linear polynomials defined on $\mathcal{T}_h$. We remark that the zero boundary conditions are taken into account by using the assumption that $\partial\Omega_{h,\mathrm{D}}$ intersects $\partial\Omega_h$, prescribing the displacements in the nodes on the surface mesh in 3D. In a more general case Nitsche's method could be used by adapting the approach proposed by Burman et al. [12]. In that case, the notion of a distance function does not hold with respect to this surface boundary which has to be represented otherwise.

The finite element method on Γ_h takes the form: find $\boldsymbol{u}_h \in V_h$ such that

$$A_h(\boldsymbol{u}_h, \boldsymbol{v}) = l_h(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in V_h. \tag{9}$$

Here the bilinear form $A_h(\cdot,\cdot)$ is defined by

$$A_h(\boldsymbol{v}, \boldsymbol{w}) = a_h(\boldsymbol{v}, \boldsymbol{w}) + j_h(\boldsymbol{v}, \boldsymbol{w}) \quad \forall \boldsymbol{v}, \boldsymbol{w} \in V_h \tag{10}$$

with

$$a_h(\boldsymbol{v}, \boldsymbol{w}) = (2\mu\varepsilon_{\Gamma_h}(\boldsymbol{v}), \varepsilon_{\Gamma_h}(\boldsymbol{w}))_{\Gamma_h} + (\lambda \nabla_{\Gamma_h} \cdot \boldsymbol{v}, \nabla_{\Gamma_h} \cdot \boldsymbol{w})_{\Gamma_h} \tag{11}$$

and

$$j_h(v, w) = \sum_{F \in \mathcal{F}_h} \int_F \tau_0 \left[\boldsymbol{n}_F^+ \cdot \nabla \boldsymbol{v}\right] \cdot \left[\boldsymbol{n}_F^+ \cdot \nabla \boldsymbol{w}\right] ds. \tag{12}$$

Here $[\boldsymbol{v}] = \boldsymbol{v}^+ - \boldsymbol{v}^-$, where $w(\boldsymbol{x})^{\pm} = \lim_{s \to 0^+} w(\boldsymbol{x} \mp s\boldsymbol{n}_F^+)$, denotes the jump of $\boldsymbol{v}$ across the face F, and τ_0 is a constant of $O(1)$. The tangent gradients are defined using the normal to the discrete surface

$$\nabla_{\Gamma_h} v = \boldsymbol{P}_{\Gamma_h} \nabla v = (\boldsymbol{I} - \boldsymbol{n}_h \otimes \boldsymbol{n}_h) \nabla v, \tag{13}$$

and the right hand side is given by

$$l_h(\boldsymbol{v}) = (\boldsymbol{f}^{\mathrm{e}}, \boldsymbol{v})_{\Gamma_h}, \tag{14}$$

where$\boldsymbol{f}^{\mathrm{e}}$ denotes an extension of$\boldsymbol{f}$ from Γ to Γ_h (e.g., using the closest point projection, cf. [7]). The jump term $j_h(\cdot,\cdot)$ serves the purpose of reducing ill-conditioning present in the bilinear form $a_h(\cdot,\cdot)$ resulting from the (difficult to avoid) presence of arbitrarily small cuts through three-dimensional elements. Indeed, depending on the representation of the zero level set, there may even be zero eigenvalues in $a_h(\cdot,\cdot)$, corresponding to an inconsequential rotation of the solution along the isoline as pointed out in the case of the Laplace–Beltrami operator in [7]; this problem is also remedied through the presence of the jump stabilization. We remark however, that the presence of small cuts does not affect convergence, only conditioning, cf. [7,6]. Note also that the thickness t of the membrane, assumed constant, has been omitted here but will, for dimensional correctness, be introduced in the coupling to 3D below.

2.4. The case of embedding in a three-dimensional body

We next consider the case of a membrane embedded in a surrounding elastic matrix. This model could be used for computation of adhesive interfaces or reinforcements using fibers or fiber plates. The setting is very general and allows for both 2D and 1D models to be added to the 3D model. Here we only consider adding membrane stiffness to an elastic 3D matrix, and we then use the triangulation $\widetilde{\mathcal{T}}_h$ for the discretization of three-dimensional elasticity. In $\Omega \backslash \Gamma$ we thus assume there holds

$$-\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{f}_{\Omega}, \quad \boldsymbol{\sigma} = 2\mu\boldsymbol{\varepsilon} + \lambda_{\Omega} \mathrm{tr}\boldsymbol{\varepsilon}\boldsymbol{I}, \tag{15}$$

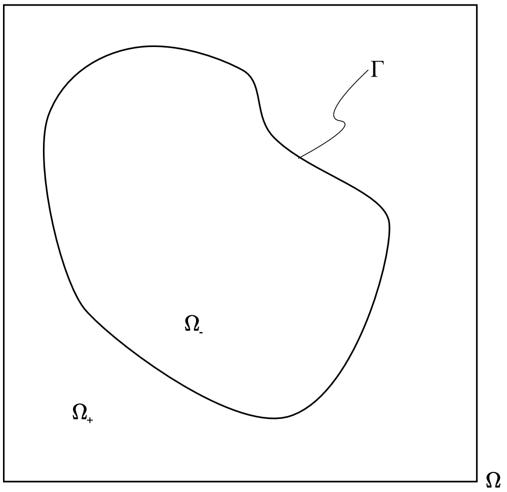


Fig. 1. 2D representation of the problem domain.

for given body force $\boldsymbol{f}_\Omega$, where $\lambda_\Omega := E\nu/((1+\nu)(1-2\nu))$. We assume for simplicity of presentation that $\boldsymbol{u} = \boldsymbol{0}$ on $\partial\Omega_{\mathrm{D}}$, a part of the boundary which is assumed to include $\partial\Omega_{h,\mathrm{D}}$, and that the traction is zero on the rest of the boundary. Our finite element method in the bulk is then based on the finite element space

$$W_h = \left\{ \boldsymbol{v} \in [\widetilde{V}_h]^3 : \ \boldsymbol{v} = \boldsymbol{0} \text{ on } \partial\Omega_{\mathrm{D}} \right\}, \tag{16}$$

and we seek $\boldsymbol{u}_h \in W_h$ such that

$$a_\Omega(\boldsymbol{u}_h, \boldsymbol{v}) + t a_h(\boldsymbol{u}_h, \boldsymbol{v}) = l_\Omega(\boldsymbol{v}) + t l_h(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in W_h \tag{17}$$

where t is the thickness of the membrane,

$$a_\Omega(\boldsymbol{u}_h, \boldsymbol{v}) := (2\mu\varepsilon(\boldsymbol{v}), \varepsilon(\boldsymbol{w}))_\Omega + (\lambda\nabla\cdot\boldsymbol{v}, \nabla\cdot\boldsymbol{w})_\Omega, \qquad \text{and } l_\Omega(\boldsymbol{v}) := (\boldsymbol{f}_\Omega, \boldsymbol{v})_\Omega.$$

The FEM (17) thus takes into account both the stiffness from the bulk and from the membrane. The bulk stiffness matrix is here established independently of the position of the membrane which allows for rapid repositioning of the membrane; this is beneficial for example for the purpose of optimizing the membrane location. We remark that when the membrane is embedded in a three-dimensional mesh used for elasticity in all of Ω, then we can drop the stabilization term (or set $\tau_0 = 0$) since the three-dimensional stiffness matrix gives stability to the embedded membrane.

We remark that the membrane equations in this model are simply added to the bulk equations which in turn are made to hold in Ω, not $\Omega\backslash\Gamma$, so that we are considering two different equilibrium problems involving the same displacement field $\boldsymbol{u}$. The coupling of bulk to membrane thus directly comes from the fact that there is only one displacement field. For more general material modeling, we need to instead consider a discrete bulk problem actually posed on $\Omega\backslash\Gamma$, which suggests the use of cut three dimensional element following [8]. In such a case we also need to reintroduce the stabilization term $j_h(\cdot,\cdot)$. As an example, consider cohesive zone modeling at the membrane where we need to allow for independent relative motion of the bulk on either side of the membrane, with shear resistance. We will return to this problem in future work.

3. Implementation

This Section describes the implementational aspects of the embedded membrane model (exemplified in Fig. 1) and provides an algorithm of the implementation.

3.1. Algorithm

1. Construct a mesh $\tilde{\mathcal{T}}$ in $\mathbb{R}^d$ on the domain Ω in which the implicit surface Γ will be embedded. Let $\mathbf{x}_N$ denote the vector of coordinates in $\tilde{\mathcal{T}}$.

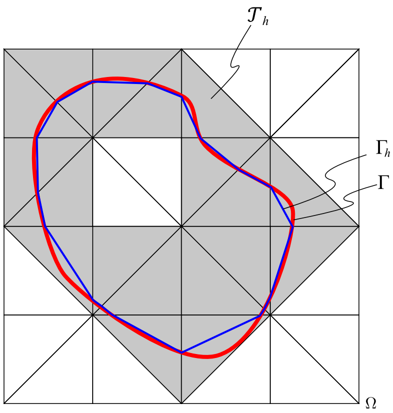
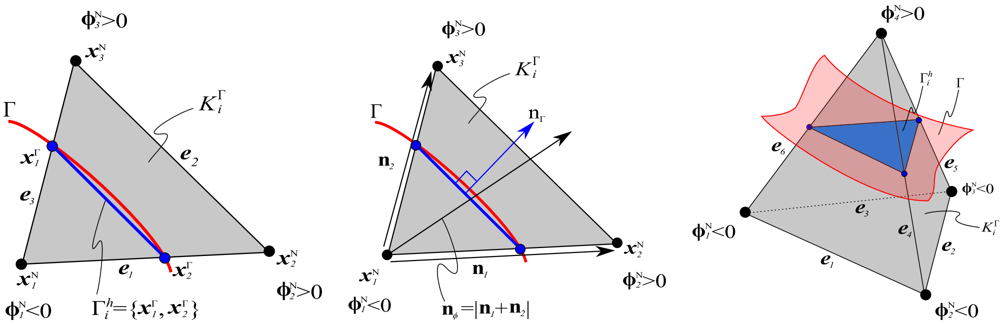


Fig. 2. Surface domain.



(a) 2D case.

(b) Surface element normal.

(c) 3D case.

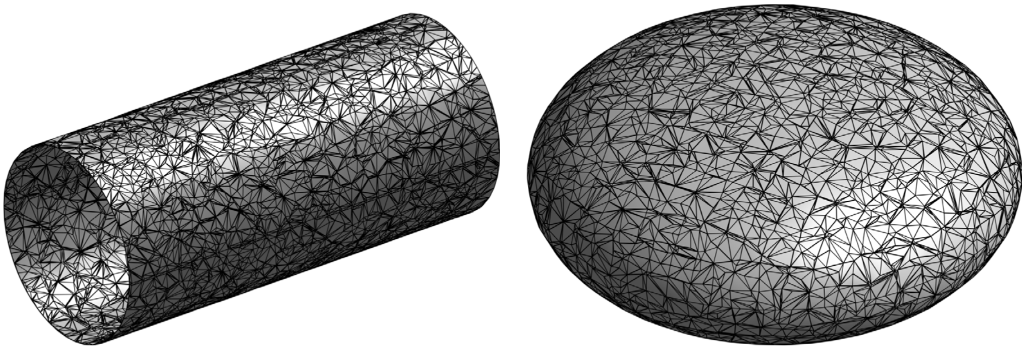
Fig. 3. Surface element Γ_h^i and parent element K_Γ^i in 2D and 3D.

2. Construct the level set function $\rho(\boldsymbol{x})$ either analytically or by the use of surface reconstruction, see Section 3.2.1 for details.
3. Discretize the distance function $\phi = \rho(\mathbf{x}_N)$ by evaluating ρ in the nodes of the complete underlying mesh $\tilde{\mathcal{T}}$.
4. Find the indices to the elements in the background mesh $\mathcal{T}_h$, by using the discrete distance values ($\phi > 0$ and $\phi < 0$) in some nodes of element T_i, see Fig. 3.
5. Extract the following surface quantities. For every element $T_i \in \mathcal{T}_h$
   (a) Find the zero surface points of the polygons Γ_h by looping over all elements $T_i \in \mathcal{T}_h$ and interpolating the discrete signed distance function ϕ linearly for every element edge that has a difference in function value (see Fig. 3 and Section 3.2.2 for details). For simplicity the polygon $\Gamma_{h,i}$ of element T_i is split into triangular elements $\hat{T}$.
   (b) Compute the face normal $\mathbf{n}_f$ of each triangular element, which is used to compute the Jacobian for the basis functions of element T_i. Note that care must be taken when defining the face normal, such that the orientation of the normal field becomes unidirectional. The element normal $\mathbf{n}_\phi$ can be used to orient $\mathbf{n}_f$ in the same direction.
6. Compute the displacement field $\boldsymbol{u}_h$ on $\mathcal{T}_h$ by solving linear system that results from the bilinear equation (10).
7. Interpolate the solution field $\mathbf{u}_h$ to $\mathbf{u}_{\Gamma_h}$ using the basis functions of the elements in $\mathcal{T}_h$.

3.2. *Implementation details*

3.2.1. *Implicit surface construction*

There are a number of ways to generate an implicit surface for analysis. An implicit surface can be approximated from a CAD surface using surface reconstruction techniques, see Belytschko et al. [13]. Another way is to use analytical implicit surface descriptions, see Burman et al. [14] for an overview. In this paper we use the following analytical surface descriptions.



(a) Cylinder.

(b) Oblate spheroid.

Fig. 4. Implicit surfaces.

Cylinder function:

$$\rho(x, y, z) = \sqrt{(x - x_c)^2 + (y - y_c)^2} - r \tag{18}$$

where $[x_c, y_c]$ is the center of the cylinder. See Fig. 4(a).

Oblate spheroid:

$$\rho(x, y, z) = x^2 + y^2 + (2z)^2 - 1. \tag{19}$$

See Fig. 4(b).

3.2.2. *Zero level surface approximation*

The overall procedure is to use linear interpolation on the discrete interface values for each element edge in order to find the zero level surface point, see Fig. 3.

1. For every element $T_i \in \mathcal{T}_h$ loop over all edges e_i.
   (a) For every edge e_i check the sign of the two discrete function values $\phi|_{e_i}$ to determine if the edge is cut.
   (b) Linearly interpolate the cut point $\mathbf{x}_{\Gamma,i}$ along the edge e_i using the two vertex coordinates $\mathbf{x}^m_{e_i}$ and $\mathbf{x}^n_{e_i}$, at nodes m and n (endpoints of e_i) and the function values $\phi|_{e_i} = \{\phi(\mathbf{x}^m_{e_i}), \phi(\mathbf{x}^n_{e_i})\}$.
   (c) Let $\mathbf{x}_{e_i,1} = \mathbf{x}_{e_i}|_{\phi|_{e_i}>0}$ (the coordinate corresponding to the highest value of ϕ) and $\mathbf{x}_{e_i,0} = \mathbf{x}_{e_i}|_{\phi|_{e_i}<0}$ and compute the vector $\mathbf{n}_i = \mathbf{x}_{e_i,1} - \mathbf{x}_{e_i,0}$. See Fig. 3(b).
2. Compute the element vector $\boldsymbol{n}_\phi = \sum \boldsymbol{n}_i$. Note that $\boldsymbol{n}_\phi$ points in the general direction of $\nabla\phi$ and is only used to determine the orientation of the face normals.
3. Depending on the number of nodes in element K^Γ_i and the orientation of the cut, several cut cases must be considered, see Fig. 5 for tetrahedral element and Fig. 6 for hexahedral. A tessellation into triangles is done for all cases.
4. In order to do the tessellation into triangles from an arbitrary polygon, a rotation from $\mathbb{R}^3$ into $\mathbb{R}^2$ was performed and then a 2D convex hull algorithm was applied.

We remark that the approach discussed here is possible to generalize to higher order elements, but at the cost of additional complexity in representing the level set, and the problem of dealing with the possibility of several exits/entries of the level set in one and the same element. If, on the other hand, higher order elements are used without raising the order of the level set, accuracy will be lost due to the poor representation of the geometry of the membrane (planar cases being an obvious but important exception).

3.3. *Membrane embedded in elastic material*

In this Section we demonstrate one particular application of the elastic membrane. Consider a set of membrane surfaces embedded into an elastic material body. We let the embedded membrane stiffen the solution by adding stiffness from the membrane solution to the bulk solution. This is easily done since the membrane shares the same degrees of freedom as the bulk.

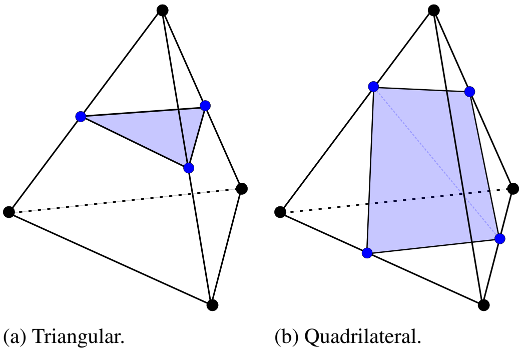

(a) Triangular. (b) Quadrilateral.

Fig. 5. Tetrahedral cut cases.

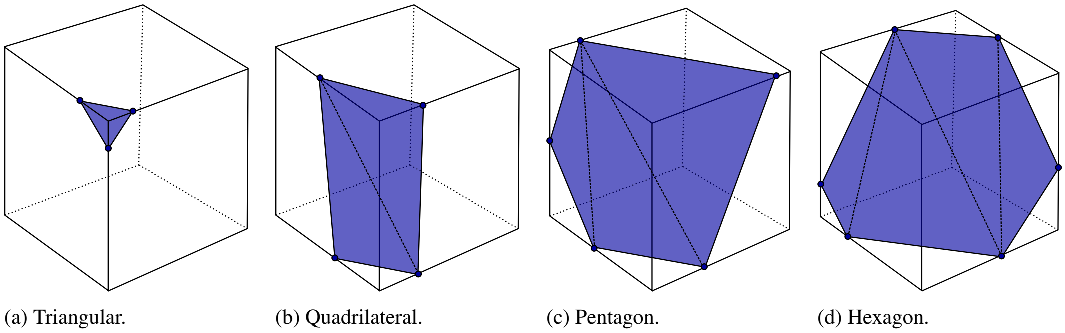

(a) Triangular. (b) Quadrilateral. (c) Pentagon. (d) Hexagon.

Fig. 6. Hexahedral cut cases.

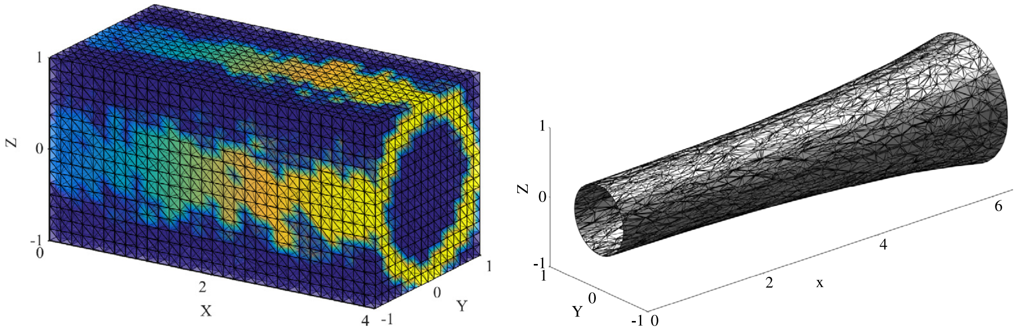
3.3.1. Algorithm

The algorithm described here is similar to the one in Section 3.1. We allow for several membranes (with the possibility of different material properties) to stiffen the bulk.

1. Construct a mesh $\tilde{\mathcal{T}}$ in $\mathbb{R}^d$ on the elastic domain Ω in which the implicit surface Γ will be embedded. Let $\mathbf{x}_N$ denote the vector of coordinates in $\tilde{\mathcal{T}}$.
2. Create a set of surfaces functions $\{\rho(\boldsymbol{x})\}$ in the same way as in the previous algorithm.
3. Discretize the distance functions $\{\phi\} = \{\rho(\mathbf{x}_N)\}$ by evaluating all functions in the set $\{\rho\}$ in the nodes of the complete underlying mesh $\tilde{\mathcal{T}}$.
4. Find the sets $\{\mathcal{T}_h\}$ that are intersected by the surfaces such that for each $\{\phi\}_i$ there exists a corresponding set of cut elements $\{\mathcal{T}_h\}_i$.
5. Follow the same approach as described in the previous algorithm to extract the zero surface information for each $\{\mathcal{T}_h\}_i$.
6. Create a discrete system of equations for the bulk elasticity problem. While assembling the bulk stiffness matrix, for each element that is cut by the membrane surface, compute the membrane element stiffness and add it to the global bulk stiffness matrix. Note that no stabilization is needed in this case since the surrounding elements create a well conditioned stiffness matrix.
7. Solve the linear system.

4. Numerical examples

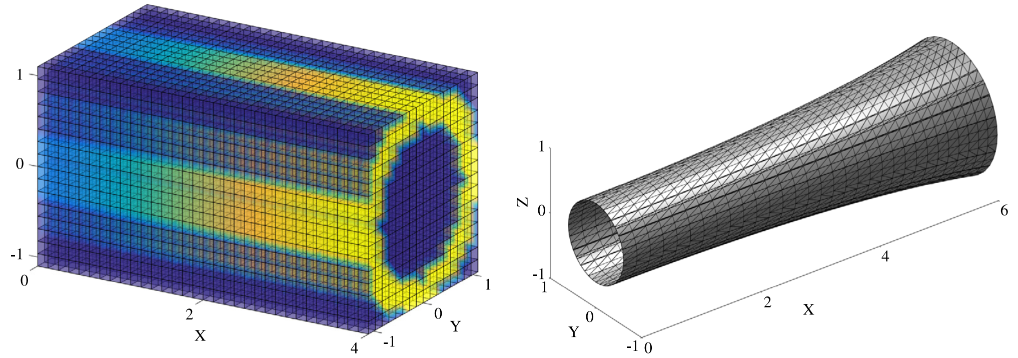
In this Section we show convergence comparison on some geometries for which we can compute the solutions analytically. We compare the convergence rates of this approach with a triangulated surface. Numerical examples are given for both tetrahedral and hexahedral elements. Since all geometrical shapes of the cuts inside the 3D elements are divided into triangles, the numerical integration of the membrane bilinear form $a_h(\cdot, \cdot)$ was performed using one point Gaussian integration on these triangles.



(a) Underlying linear tetrahedral mesh.

(b) Zero level iso-surface.

Fig. 7. Displacement field on a tetrahedral mesh (×10 enlarged).



(a) Underlying linear hexahedral mesh.

(b) Zero level iso-surface.

Fig. 8. Displacement field on a hexahedral mesh (×10 enlarged).

The mesh size is defined by:

$$h := \frac{1}{\sqrt[3]{N}}$$

where N denotes the total number of nodes on the underlying 3D mesh $\widetilde{\mathcal{T}}_h$, which is uniformly refined.

4.1. Pulling a cylinder

Comparing this approach to the approach previously done by Hansbo and Larson [5], we consider a cylindrical shell of radius r and thickness t, with open ends at $x = 0$ and $x = L$ and with fixed axial displacements at $x = 0$ and radial at $x = L$, carrying an axial surface load per unit area

$$f(x, y, z) = \frac{F}{2\pi r}\frac{x}{L^2},$$

where F has the unit of force. The resulting axial stress is

$$\sigma = \frac{F(1-(x/L)^2)}{4\pi r t}.$$

We consider the same example as was used in [5] and choose $r = 1$, $L = 4$, $E = 100$, $\nu = 1/2$, $t = 10^{-2}$ and $F = 1$.

In Fig. 7 we show the solution (enlarged 10 times) on a given tetrahedral mesh and in Fig. 8 the corresponding solution on a hexahedral mesh with the same mesh size. In Fig. 9 we show the $L_2(\Sigma)$ error in stress, $\|\boldsymbol{\sigma} - \boldsymbol{\sigma}_h\|$, where $\boldsymbol{\sigma} := \boldsymbol{\sigma}_\Gamma(\boldsymbol{u})$ and $\boldsymbol{\sigma}_h := \boldsymbol{\sigma}_\Gamma(\boldsymbol{u}_h)$. See Table 1 for convergence rates.

Table 1
Convergence of the cylinder.

Element type	Mesh size	$\|\sigma - \sigma_h\|$	Rate
Tetrahedral unstructured	0.0899	1.7959	–
	0.0481	0.7733	1.3498
	0.0260	0.4017	1.0650
Tetrahedral structured	0.1330	5.7545	–
	0.0721	2.1916	1.5749
	0.0376	0.6846	1.7903
	0.0192	0.2550	1.4722
Hexahedral	0.1644	2.8832	–
	0.0899	1.4008	1.1954
	0.0472	0.6698	1.1438
	0.0242	0.3228	1.0924
Original	0.1565	0.7580	–
	0.0783	0.3793	0.9998
	0.0392	0.1897	1.0015

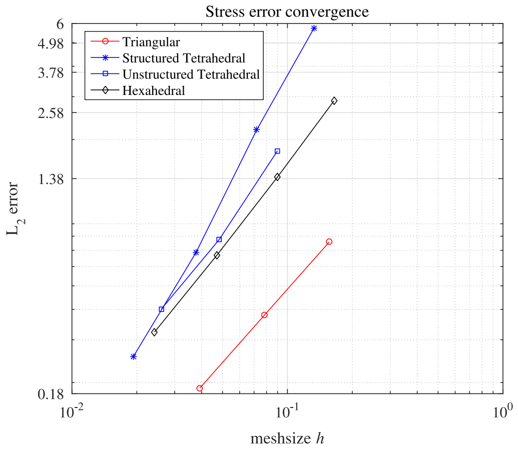
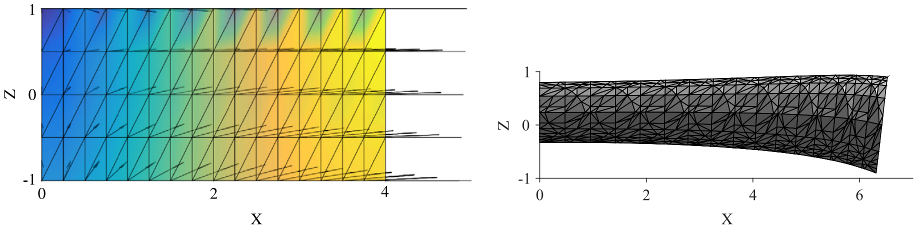


Fig. 9. Stress convergence comparison for the cylinder.

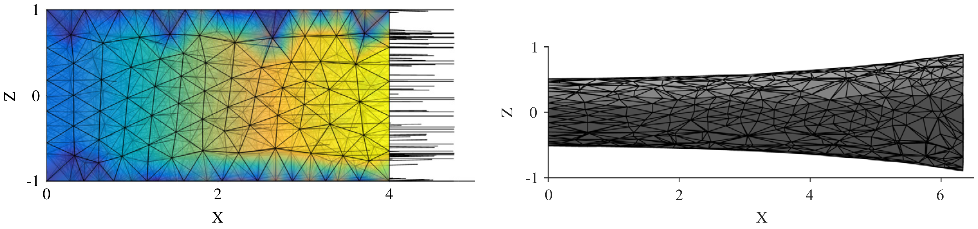


(a) Displacement field on the underlying mesh (Exaggerated lengths).

(b) Displacement field on the extracted linear zero level iso-surface.

Fig. 10. Mesh dependent errors on low resolution linear tetrahedral mesh.

Mesh dependent errors occur in a structured tetrahedral mesh case due to the directionality of the mesh, which induces artificial direction-dependent stiffnesses, see Fig. 10; note that the error becomes less prominent with a finer mesh. This error can be avoided by using an unstructured tetrahedral mesh, see Fig. 11; also, this error does not occur in the hexahedral case, see Fig. 12.



(a) Underlying displacement field.

(b) Surface displacement field.

Fig. 11. Displacement fields on unstructured tetrahedral mesh.

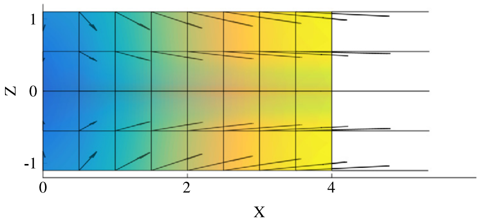


Fig. 12. Displacement field on the underlying tri-linear hexahedral mesh (×10 enlarged).

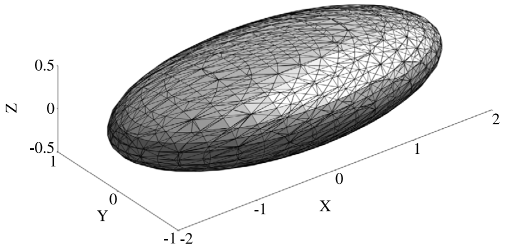


Fig. 13. Displacement field on the extracted linear zero level iso-surface from a linear tetrahedral mesh (×10 enlarged).

Table 2
Convergence of the oblate.

Element type	Mesh size	Number of DOFs	$\|\sigma - \sigma_h\|$	Rate
	0.0790	2414	0.1493	–
Tetrahedral	0.0474	7218	0.0800	1.2222
	0.256	26961	0.0425	1.0261
	0.0198	2562	0.0242	–
Structured, triangular	0.0099	10242	0.0121	1.0004
	0.0049	40962	0.0061	0.9882
	0.0198	2562	0.0288	–
Unstructured, triangular	0.0099	10242	0.0147	0.9707
	0.0049	40962	0.0078	0.9144

4.2. *Pulling an oblate spheroid*

Again we use the same example as in [5] and set the exact solution to be $\boldsymbol{u} = (x, 0, 0)$ and compute the stress and then the corresponding load from (3). We set the parameters $E = 1$, $\nu = 1/2$, and $t = 1$. The computed displacement field can be seen in Fig. 13. Compared to the previous work in [5], the superparametric stabilization method is not needed in this approach since we already stabilize using the term $j_h(\cdot, \cdot)$. A stress convergence comparison can be seen in Fig. 14. The convergence rates can be seen in Table 2.

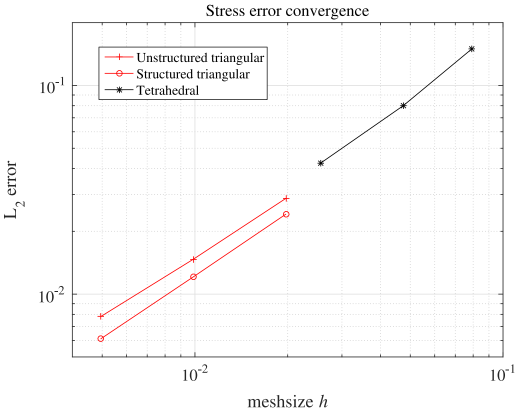


Fig. 14. Convergence of stress.

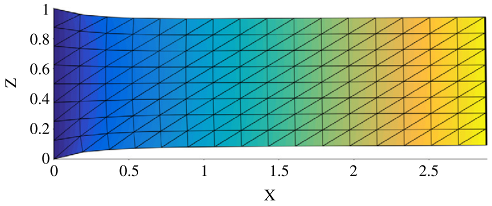


Fig. 15. Elastic beam without embedded elastic membrane.

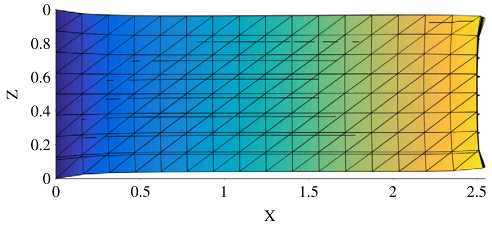


Fig. 16. Elastic beam with embedded elastic membrane.

4.3. *Membrane embedded in elastic material*

A rectangular box ($0 \leq x \leq 2$ and $0 \leq y, z \leq 1$) has a surface load, $f = 1$, applied in the positive x direction at $x = 2$. The bulk material has the following properties, $E = 100$, $\nu = 0.5$. The membrane has $E = 1000$, $\nu = 0.5$ and $t = 0.01$. Fig. 15 shows a displacement plot (40×enlarged) of the elastic bulk material without any embedded membrane. Fig. 16 shows the same displacement plot but with 8 embedded membrane surfaces. The surfaces are visualized in Fig. 17.

Finally, in Figs. 18 and 19 we show the effect on displacements resulting from inserting a circular membrane into a beam in bending. The material properties are the same as in the previous example. We note the marked increase in bending stiffness resulting from the added membrane stiffness.

5. Concluding remarks

In this paper we have introduced an FE model of curved membranes using higher dimensional shape functions that are restricted to (the approximation of) the membrane surface. This allows for rapid insertion of arbitrarily shaped membranes into already existing 3D FE models, to be used for example for optimization purposes. We have shown numerically that the cut element approach to membranes gives errors comparable to triangulated membranes, using the same degree of approximation, and we have proposed a stabilization method which provides stability to the solution

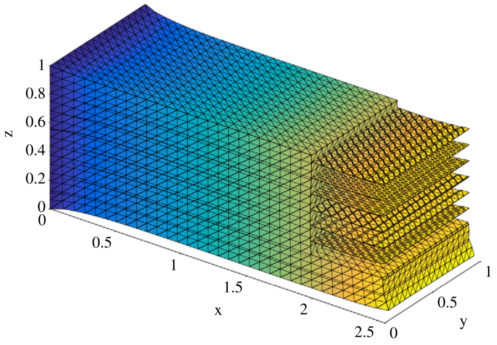


Fig. 17. Elastic beam with embedded elastic membrane.

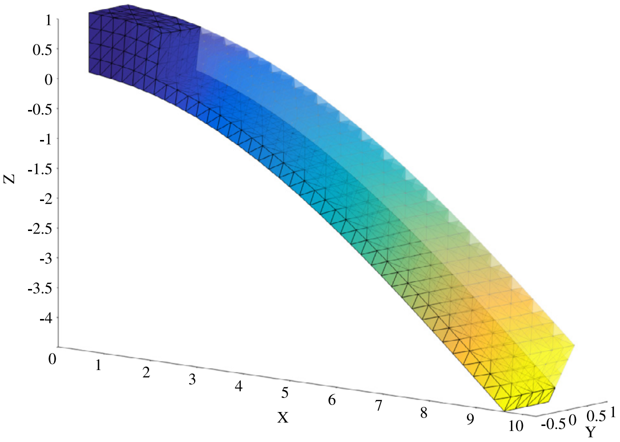


Fig. 18. Elastic beam in bending without embedded elastic membrane.

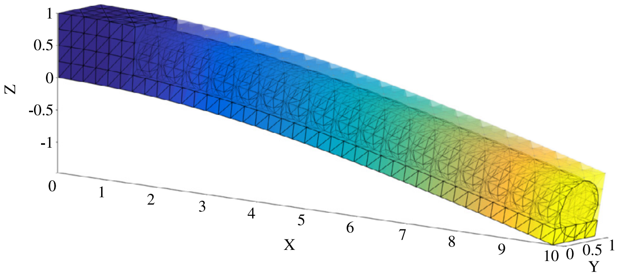


Fig. 19. Elastic beam in bending with embedded elastic membrane.

as well as giving the right conditioning of the discrete system, allowing for arbitrarily small cuts in the 3D mesh. The novelties of our approach are: (1) applying for the first time the cut element approach of [6] to models of thin elastic structures, and (2) modeling membranes embedded in a 3D bulk material with arbitrary orientation of the membrane relative to the meshing of the bulk. In future work, we will consider more realistic models of embedded membranes.

Acknowledgments

This research was supported in part by the Swedish Foundation for Strategic Research Grant No. AM13-0029, the Swedish Research Council Grant Nos. 2011-4992 and 2013-4708, and the Swedish Research Programme Essence.

References

[1] M.E. Gurtin, A.I. Murdoch, A continuum theory of elastic material surfaces, Arch. Ration. Mech. Anal. 57 (1975) 291–323.

[2] M.C. Delfour, J.-P. Zolésio, A boundary differential equation for thin shells, J. Differential Equations 119 (2) (1995) 426–449.

[3] G. Dziuk, Finite elements for the Beltrami operator on arbitrary surfaces, in: Partial Differential Equations and Calculus of Variations, in: Lecture Notes in Math., vol. 1357, Springer, Berlin, 1988, pp. 142–155.

[4] G. Dziuk, C.M. Elliott, Finite element methods for surface PDEs, Acta Numer. 22 (2013) 289–396.

[5] P. Hansbo, M.G. Larson, Finite element modeling of a linear membrane shell problem using tangential differential calculus, Comput. Methods Appl. Mech. Engrg. 270 (2014) 1–14.

[6] M.A. Olshanskii, A. Reusken, J. Grande, A finite element method for elliptic equations on surfaces, SIAM J. Numer. Anal. 47 (5) (2009) 3339–3358.

[7] E. Burman, P. Hansbo, M.G. Larson, A stabilized cut finite element method for partial differential equations on surfaces: the Laplace–Beltrami operator, Comput. Methods Appl. Mech. Engrg. 285 (2015) 188–207.

[8] A. Hansbo, P. Hansbo, A finite element method for the simulation of strong and weak discontinuities in solid mechanics, Comput. Methods Appl. Mech. Engrg. 193 (33–35) (2004) 3523–3540.

[9] P. Hansbo, K. Salomonsson, A discontinuous Galerkin method for cohesive zone modelling, Finite Elem. Anal. Des. 102/103 (2015) 1–6.

[10] O.C. Zienkiewicz, The Finite Element Method, McGraw-Hill, London, 1977. The third expanded and revised, edition of The Finite Element Method in Engineering Science.

[11] M. Cenanovic, P. Hansbo, M.G. Larson, Minimal surface computation using a finite element method on an embedded surface, Internat. J. Numer. Methods Engrg. 104 (7) (2015) 502–512.

[12] E. Burman, P. Hansbo, M.G. Larson, K. Larsson, A. Massing, Finite element approximation of the Laplace–Beltrami operator on a surface with boundary, ArXiv e-prints, September 2015.

[13] T. Belytschko, C. Parimi, N. Moës, N. Sukumar, S. Usui, Structured extended finite element methods for solids defined by implicit surfaces, Internat. J. Numer. Methods Engrg. 56 (4) (2003) 609–635.

[14] E. Burman, S. Claus, P. Hansbo, M.G. Larson, A. Massing, CutFEM: Discretizing geometry and partial differential equations, Internat. J. Numer. Methods Engrg. 104 (7) (2015) 472–501.