Computer-Aided Design 42 (2010) 808–816

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

Algebraic reduction of beams for CAD-integrated analysis

Kavous Jorabchi, Joshua Danczyk, Krishnan Suresh *

2050 Mechanical Engineering, 1513 University Ave, Madison, WI 53706, University of Wisconsin, Madison, United States

ARTICLE INFO

Article history:
Received 16 October 2007
Accepted 6 May 2010

Keywords:
Beams
Finite element analysis
CAD-integrated analysis
Dimensional reduction

ABSTRACT

Beams are high aspect ratio structural members that are used extensively in civil, automotive, aerospace, and MEMS applications. In all such applications, one must typically analyze and optimize the beams through computer simulations. Standard 3D finite element analysis (FEA) of beams can be used in such simulations; it is however prone to errors, and is computationally expensive for thin structures. Therefore, a common strategy is to carry out a dimensionally reduced 1D beam analysis. Unfortunately, 1D beam analysis is hard to automate and integrate with 3D CAD.

In this paper, we propose an alternate "algebraic reduction" method that combines the generality of 3D FEA, and the computational efficiency of 1D beam analysis. This is achieved via a dual-representation framework where the *geometry* of the beam is captured via a 3D finite element mesh, while the *physics* is captured via a 1D beam model. The proposed method is formally established, and supported through numerical experiments.

1. Introduction

Beams[1] are high aspect ratio structural members. Due to their high strength-to-weight ratio they are used extensively, for example, in civil structures, automotive body panels, aerospace structures, and MEMS applications. Fig. 1, for example, shows a high aspect ratio micro-cantilever beam used as an MEMS vibration sensor [1].

In theory, it is possible to analyze beams via standard 3D finite element analysis [2]. However, the recommended strategy is 1D beam analysis [3] for reasons discussed below (other less-common methods are also reviewed later in this section).

Standard 3D finite element analysis (FEA) (see Fig. 2) has reached a high degree of reliability over the past few decades, making it the *de facto* analysis method today.

However, high aspect ratio beams pose unique challenges to 3D FEA. Specifically, consider the beam problem in Fig. 3(a). If one uses a coarse finite element mesh (element size $\gg$ thickness) as in Fig. 3(b), the presence of poor quality elements leads to *Poisson and shear locking* [4].

On the other hand, if a high quality mesh (element size $\sim$ thickness) is used, the computational cost grows rapidly with the aspect ratio as illustrated in Fig. 4 (aspect ratio is the overall length divided by the thickness of the hollow beam).

Thus, despite the generality and ease of 3D *FEA, it is rarely used today to analyze high aspect ratio structures.*

Indeed, the recommended strategy for thin beam-like geometries is 1D beam analysis [5]. This method entails *explicitly* computing the 1D beam axis and extracting the cross-sectional properties (see Fig. 5); 1D beam analysis does not suffer from locking and ill-conditioning problems, and is highly efficient.

However, 1D beam analysis poses numerous automation challenges. Specifically, as the beam becomes increasingly complex, computing the 1D beam geometry from a 3D CAD model can be cumbersome [6,7]. Further, coupling the 1D geometry to 3D structural elements is non-trivial (see Fig. 5). Finally, post-processing and visualizing the 1D analysis results within the 3D environment defeats the very purpose of 3D modeling.

Besides 3D FEA and 1D analysis, other methods have been proposed for analyzing thin structures [8]. A popular method is based on the concept of solid-shell elements that use an anisotropic space for the finite element basis functions, i.e., relatively low order shape functions are used across the thickness to overcome ill-conditioning, etc. However, solid-shell methods entail *a priori* orientation of the finite element mesh [9], which can pose difficulties for standard finite element mesh generators.

Reduced integration techniques have also been proposed by several researchers to suppress the deficiencies of standard FEA [10,11]; however, under-integration causes generation of *hourglass* modes and needs stabilization. Yet another technique is to use hybrid or mixed variational principles for stresses and displacements [12]. Hybrid elements can be computationally expensive since construction of the element stiffness entails inverting a sizeable matrix [12].

* Corresponding author. Tel.: +1 608 262 3594; fax: +1 608 265 2316.
E-mail address: suresh@engr.wisc.edu (K. Suresh).

[1] In this paper the word "beam" is used in two contexts: "beam" without any prefix implies the actual 3D geometric model, whereas "1D beam" implies 1D (line) idealization of the 3D geometry.

0010-4485/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2010.05.001

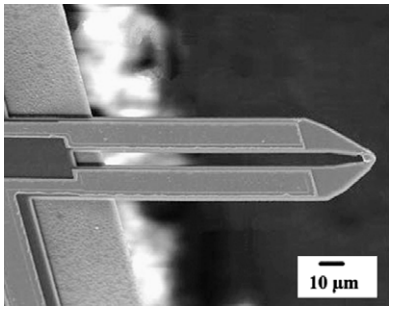


Fig. 1. A high aspect ratio micro-cantilever [1].

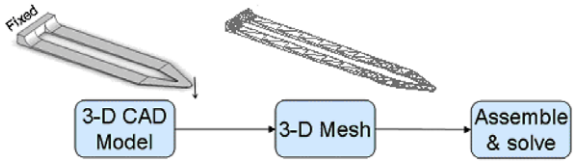


Fig. 2. Standard 3D finite element process.

The authors in [13] developed a CAD-integrated subdivision technique for the analysis of shells and plates. However, this technique assumes that the lower-dimensional manifold (the mid-surface) has already been computed from a 3D CAD model. Finally, medial axis methods have been proposed for an automated analysis of thin structures [7,14,15], but medial axis computation is known to be expensive and difficult [16–19].

Given these limitations, we propose here an *algebraic reduction* method (as opposed to *explicit* reduction methods) which overcomes the challenges through an algebraic manipulation rather than geometric manipulation. This provides a *dual-representation* structural analysis method in that it offers the geometric flexibility and generality of 3D FEA and the computational efficiency and accuracy of 1D beam analysis. In the proposed method, the *geometry* of the structure is captured via an arbitrary 3D finite element mesh, but the *physics* is captured via classic beam theory. Consequently, we show that analysis can be carried out efficiently and accurately within a standard 3D CAD environment.

To motivate the proposed method, we briefly summarize 3D FEA in Section 2.1 and review its limitations. In Section 2.2 we review how 1D beam analysis overcomes these limitations. In Section 2.3 we show how 3D FEA can be suitably modified to mimic 1D analysis; this will lead us to the basic algebraic reduction framework which is generalized in Section 2.4. The framework is then extended in Section 3 to address a variety of scenarios. Numerical examples and a case study are considered in Section 4, followed by conclusions and future work in Section 5.

2. The algebraic reduction strategy

For illustrative purposes, in this section, we shall consider a simple cantilever of width $4H$, height $2H$, and length L, as illustrated in Fig. 6; it is assumed that $L \gg H$. The Young's modulus of the beam is denoted by E and Poisson ratio by ν. The beam is loaded by a shear force at $x = L$, and the objective is to determine the deflection and stress characteristics of the beam accurately and efficiently.

2.1. The 3D finite element analysis

Despite the simplicity of the problem, its 3D FEA is not easy. Recall that in 3D FEA the geometry is first discretized into a 3D finite element mesh. For simplicity we shall use a *single* high aspect ratio 27-noded hexahedral element (see Fig. 7) endowed with 2nd order Lagrangian shape functions $N(\xi, \eta, \chi)$ [2]:

$$N(\xi, \eta, \chi) = l^2(\xi) \odot l^2(\eta) \odot l^2(\chi) \tag{2.1}$$

where $l^2(.)$ is 1D 2nd order Lagrangian shape function:

$$l^2(\xi) = \begin{bmatrix} \xi(\xi-1)/2 & -(\xi+1)(\xi-1) & (\xi+1)\xi/2 \end{bmatrix} \tag{2.2}$$

and $\odot$ is defined as the operation needed to construct 3D FEA shape functions from 1D shape functions. The displacements are therefore approximated via:

$$\begin{aligned} u &= N(\xi, \eta, \chi)\bar{u}; \\ v &= N(\xi, \eta, \chi)\bar{v}; \\ w &= N(\xi, \eta, \chi)\bar{w} \end{aligned} \tag{2.3}$$

where u, v & w are displacements along x, y & z directions respectively, and $\bar{u}$, $\bar{v}$ & $\bar{w}$ are FEA nodal displacements.

To determine the unknown displacements the structural energy minimization principle is exploited [2]:

$$\text{Minimize} \left[\frac{1}{2} \int_{\Omega} \sigma_{ij}\varepsilon_{ij} \mathrm{d}\Omega - \int_{\Gamma_N} w(L, y, z) q_0 \mathrm{d}\Gamma \right] \tag{2.4}$$

where summation over repeated indices is implied. In above equation the strains are determined from the displacements via standard kinematic relations and the stresses are determined via the generalized Hooke's law [2].

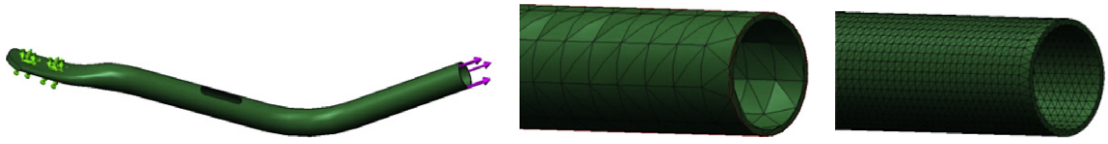
Upon exploiting these well established results Eq. (2.4) leads to a linear algebraic system of equations [2]:

$$\begin{aligned} K^{3D} d^{3D} &= q^{3D} \\ d^{3D}|_{x=0} &= 0 \end{aligned} \tag{2.5}$$

where K^{3D} is the 3D FEA stiffness matrix. Unfortunately, it is well known that:

- The problem in Eq. (2.5) is ill-conditioned as $H/L \to 0$ since 3D FEA fails to incorporate physically observed constraints, as explained in the next section.
- In addition, the minimization principle in Eq. (2.4) involves spurious energy modes (example: $\sigma_{zz}\varepsilon_{zz}$) that dominate over the desired bending energy mode ($\sigma_{xx}\varepsilon_{xx}$) as the thickness goes to zero [4].

For these reasons, a direct 3D analysis of thin structural geometries is impractical. For 3D FEA to be effective, a large number ($\sim O(L/H)$) of high quality elements must be created, making it computationally inefficient, as discussed earlier.



(a) A 3D beam problem. (b) A coarse mesh. (c) A fine mesh.

Fig. 3. Finite element analysis of a 3D beam.

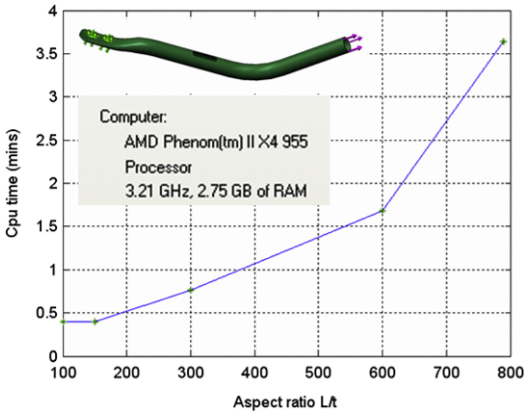


Fig. 4. 3D FEA cpu-time vs. aspect ratio of a beam, for a high quality mesh.

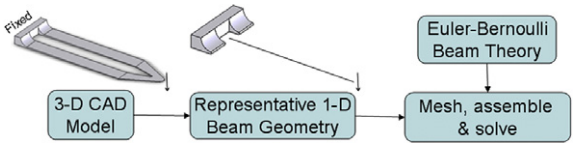


Fig. 5. 1D beam analysis is hard to automate.

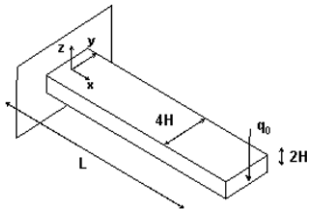


Fig. 6. An illustrative cantilever beam.

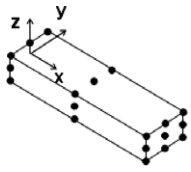


Fig. 7. A 27-noded hexahedral element.

2.2. A 1D beam analysis

In contrast to 3D FEA, a 1D Euler–Bernoulli beam analysis directly addresses the above two limitations. First, the ill-conditioning is addressed by assuming *a priori* that the displacements during bending are tightly coupled via a 1D function $w_0(x)$ as follows [20] (assuming plane-stress in x–z plane):

$$\begin{aligned} u(x,y,z) &= -z\frac{\partial w_0(x)}{\partial x} \\ w(x,y,z) &= w_0(x). \end{aligned} \tag{2.6}$$

Further, $w_0(x)$ is typically approximated via Hermite cubic polynomial $H(t)$; $0 \le t \le 1$ [21]:

$$H = \begin{Bmatrix} 1-3t^2+2t^3 & L(t-2t^2+t^3) & 3t^2-2t^3 & L(-t^2+t^3) \end{Bmatrix} \tag{2.7}$$

as follows:

$$\begin{aligned} w_0(x) &= H(t)d^{1D}; \quad \text{with } x = Lt \\ d^{1D} &= \begin{Bmatrix} w_1 & \theta_1 & w_2 & \theta_2 \end{Bmatrix}^T. \end{aligned} \tag{2.8}$$

Finally, to avoid locking, only the bending energy is considered, and the minimization principle is reduced to [20]:

$$\text{Minimize}\left[\frac{1}{2}\int_\Omega \sigma_{xx}\varepsilon_{xx}\mathrm{d}\Omega - \int_{\Gamma_N} w(L,y,z)q_0\mathrm{d}\Gamma\right] \tag{2.9}$$

where the strain and stress are defined as $\varepsilon_{xx} = u_{,x}$ and $\sigma_{xx} = E\varepsilon_{xx}$ respectively; resulting in a linear system of equations:

$$\begin{aligned} &K^{1D}d^{1D} = q^{1D} \\ &w_1 = 0; \qquad \theta_1 = 0. \end{aligned} \tag{2.10}$$

Eq. (2.10) is now well conditioned and free of locking. However, as stated earlier, 1D beam analysis poses serious automation challenges within a 3D CAD environment. For example, observe that one must explicitly create a 1D beam element endowed with appropriate cross-sectional properties (area, area moment of inertia, principle axes, etc.) over which a 1D function must be defined. This requires significant manual intervention for geometrically complex beams.

2.3. Algebraic reduction strategy

Our objective now is to retain the '*best of both worlds*', i.e., we would like to retain the geometric flexibility of 3D FEA (specifically the ease with which it is integrated into 3D CAD) and the computational efficiency and accuracy of 1D beam analysis. The proposed strategy uses a 3D finite element mesh to discretize the geometry, but incorporates the 1D beam physics as follows:

1. First, instead of constructing the 3D stiffness matrix via Eq. (2.4), we construct a 3D bending stiffness matrix via Eq. (2.9) to prevent locking in 3D FEA; the methodology is discussed below in Section 2.3.1.
2. Next, to ensure well conditioning of the 3D bending stiffness matrix, the 3D finite element degrees of freedom are projected onto a lower-dimensional space spanned by a *virtual* set of Euler–Bernoulli degrees of freedom; the methodology is discussed in Section 2.3.2.

The resulting algebraic equation is shown to be well conditioned and free of locking. Indeed, if certain conditions are satisfied (to be discussed later on), the resulting stiffness matrix and force vector will be exactly equal to the Euler–Bernoulli stiffness matrix and force vector, albeit derived via 3D mesh.

2.3.1. 3D bending stiffness matrix

Recall that the stiffness matrix K^{3D} in Eq. (2.5) is based on the energy principle in Eq. (2.4), and therefore contains spurious energy modes. Our objective is to construct a stiffness matrix $K^{3D:B}$ that contains only the bending energy. We therefore start with the minimization principle in Eq. (2.9) but apply it to the stresses and strains in the 3D mesh.

Note that the relevant strain and stress components in Eq. (2.9) are ε_{xx} and σ_{xx}. In 3D FEA the displacements are given by Eq. (2.3); therefore, the desired strain ε_{xx} is:

$$\varepsilon_{xx} = N_{,x}\bar{u}. \tag{2.11}$$

To ensure consistency with the Euler–Bernoulli constitutive equation, the stress σ_{xx} is assumed to satisfy:

$$\sigma_{xx} = E\varepsilon_{xx} \tag{2.12}$$

resulting in the potential energy:

$$\Pi = \left[\frac{E}{2}\int_\Omega \bar{u}^T N_{,x}^T N_{,x}\bar{u}\mathrm{d}\Omega - \int_{\Gamma_N} \bar{w}^T N^T q_0 \mathrm{d}\Gamma\right]. \tag{2.13}$$

Thus, the bending energy within the 3D mesh of Fig. 7 is encapsulated in the 3D *bending stiffness matrix*:

$$K^{3D:B} \equiv E \int_{\Omega} N_{,x}^{T} N_{,x} \mathrm{d}\Omega \tag{2.14}$$

i.e.,:

$$\Pi = \left[\frac{1}{2} \bar{u}^{T} K^{3D:B} \bar{u} - \bar{w}^{T} q^{3D:B} \right]. \tag{2.15}$$

2.3.2. Beam subspace reduction

Observe that Eq. (2.15) involves the 3D degrees of freedom $\bar{u}$ & $\bar{w}$. Before minimizing Eq. (2.15) kinematic constraints are imposed on these degrees of freedom via Eq. (2.6). In a direct analogy to the 1D beam degrees of freedom d^{1D} in Eq. (2.8), we introduce *virtual* degrees of freedom d^{AR} that constrain the 3D FEA nodal displacements $\bar{u}$ & $\bar{w}$ as follows: for every node (x_i, y_i, z_i) in the 3D region, we require:

$$\begin{aligned} t_i &= x_i / L \\ \bar{u}_i &= -z_i H_{,x}(t_i) d^{AR}; \quad i = 1, \ldots, 27 \\ \bar{w}_i &= H(t_i) d^{AR}; \quad i = 1, \ldots, 27 \end{aligned} \tag{2.16}$$

where $H(t)$ is the Hermite polynomial in Eq. (2.7). In other words, for a 27-noded hexahedral element two matrices of size 27×4 are constructed:

$$\begin{aligned} P_u(i, 1:4) &\equiv -z_i H_{,x}(t_i); \quad i = 1, \ldots, 27 \\ P_w(i, 1:4) &\equiv H(t_i); \quad i = 1, \ldots, 27 \end{aligned} \tag{2.17}$$

with:

$$\begin{aligned} \bar{u} &= P_u d^{AR} \\ \bar{w} &= P_w d^{AR}. \end{aligned} \tag{2.18}$$

Substituting Eq. (2.18) in Eq. (2.15) yields:

$$\Pi = \left[\frac{1}{2} \left(d^{AR} \right)^{T} P_u^{T} K^{3D:B} P_u d^{AR} - \left(d^{AR} \right)^{T} P_w^{T} q^{3D:B} \right] \tag{2.19}$$

whose minimization (with respect to d^{AR}) results in the linear system of equation:

$$K^{AR} d^{AR} = q^{AR} \tag{2.20}$$

where K^{AR} and q^{AR} are the *Algebraic reduction* stiffness matrix and forcing vector defined by:

$$\begin{aligned} K^{AR} &= P_u^{T} K^{3D:B} P_u \\ q^{AR} &= P_w^{T} q^{3D:B}. \end{aligned} \tag{2.21}$$

For the quadratic hexahedral element in Fig. 7, one can easily compute K^{AR} symbolically (of course, in practice, one would use numerical integration) to show that:

$$K^{AR} = \frac{EI}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ & 4L^2 & -6L & 2L^2 \\ & Sym & 12 & -6L \\ & & & 4L^2 \end{bmatrix} \tag{2.22}$$

where I is the y-moment of inertia for the beam in Fig. 6. It can be readily verified that K^{AR} is exactly equal to the Euler–Bernoulli stiffness matrix K^{1D} for the above problem [3]. The underlying reason is that the basis functions of a 2nd order hexahedral element span the following shape functions:

$$\begin{bmatrix} 1 & x & z & zx & x^2 & zx^2 \end{bmatrix}. \tag{2.23}$$

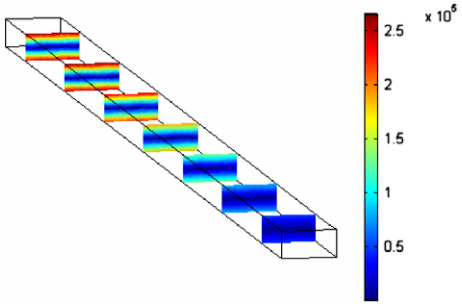


Fig. 8. Algebraic reduction supports 3D post-processing.

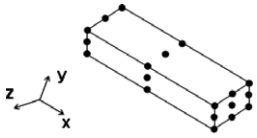


Fig. 9. Local x axis is aligned with the beam direction.

One can show that the above shape functions are sufficient for capturing the strain energy of the 1D Euler–Bernoulli beam model. Indeed, numerical experiments later show that any finite element space that captures the above shape functions will result in the exact Euler–Bernoulli stiffness matrix. Furthermore, in the above scenario, one can show that the force vector q^{AR} in Eq. (2.21) also matches the Euler–Bernoulli force vector q^{1D}. In other words, given identical boundary conditions, it follows that $d^{AR} = d^{1D}$; i.e., we have mimicked the results of the Euler–Bernoulli model within a full 3D FEA environment using a virtual beam element.

Observe that once the virtual degrees of freedom d^{AR} are obtained from Eq. (2.20), the 3D finite element degrees of freedom are easily computed via Eq. (2.18), which are then directly fed into the 3D finite element post-processing module. One can recover, for example, stresses at any cross section directly via the 3D post-processing unit. A typical cross-sectional plot of the von Mises stress is displayed in Fig. 8.

2.4. Generalized algebraic reduction strategy

In the previous section it was shown how to mimic the results of the Euler–Bernoulli model within a full 3D CAD environment using a *virtual* beam element. The kinematic constraints used in 1D beam analysis (and hence in the proposed method), i.e., Eq. (2.6), are based on three implicit assumptions (see Fig. 7), namely:

1. The longitudinal direction of the beam is *a priori* known.
2. The coordinate system is already located on the mid-surface (neutral axis) of the beam.
3. The coordinate system is already aligned with the principle axes of the beam cross section.

These assumptions pose serious limitations on automating the reduction process as discussed before. In this section it is shown that the last two assumptions are easily relaxed in the algebraic reduction framework. However beam longitudinal direction is still required to be known *a priori*; this limitation is currently being addressed.

Since the beam longitudinal direction is known, we align the *local* x axis with this direction; however y & z axes can be arbitrarily oriented with respect to the beam cross section. Further, the origin can be located at any arbitrary point as shown in Fig. 9.

In Section 2.3 we considered bending about only one axis (y axis). However due to linearity of the problem, one can easily append 4 more virtual degrees of freedom to d^{AR} in Eq. (2.18) in order to capture the bending about the other axis (z axis

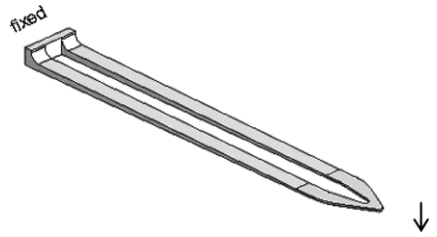


Fig. 10. An illustrative cantilever.

here). Moreover, since the location of neutral axis is not known *a priori*, we introduce new degrees of freedom corresponding to axial deformation. According to these two modifications, the Euler–Bernoulli model in Eq. (2.6) is modified as follows:

$$\begin{aligned} u(x,y,z) &= u_0(x) - z\frac{\partial w_0(x)}{\partial x} - y\frac{\partial v_0(x)}{\partial x} \\ v(x,y,z) &= v_0(x) \\ w(x,y,z) &= w_0(x). \end{aligned} \tag{2.24}$$

The above equations now include an unknown axial displacement $u_0(x)$ to account for the offset of the coordinate system. As before, it is assumed that a cubic Hermite polynomial $H(t);\ 0 \le t \le 1$ approximates $w_0(x)$ and $v_0(x)$, while a 2nd order Lagrangian polynomial $l^2(\xi);\ -1 \le \xi \le 1$ approximates $u_0(x)$. Thus, algebraic reduction degrees of freedom for a single virtual beam element are:

$$d^{\mathrm{DR}} = [\, v_1 \quad \theta_1^y \quad v_2 \quad \theta_2^y \quad w_1 \quad \theta_1^z \quad w_2 \quad \theta_2^z \quad u_1 \quad u_2 \quad u_3 \,]^T \tag{2.25}$$

where θ_i^y and θ_i^z are rotations about z and y axes respectively and v_i and w_i are displacements along y and z axes respectively. Carrying out *beam subspace reduction* (Section 2.3.2) using Eq. (2.24) instead of (2.6) allows y and z coordinates in Eq. (2.24) to be measured from any coordinate system such as the one shown in Fig. 9. In fact, observe that the y & z components of the algebraic reduction stiffness matrix are now coupled, accommodating the skew bending situation. Further, since the degrees of freedom associated with Eq. (2.24) are virtual, one still needs to project them to 3D (using the already built projection matrices) for post-processing purposes. The derivation is omitted here since it is identical to the derivation in Section 2.3 with the above-mentioned modifications.

3. The algebraic reduction method

Consider now the cantilever in Fig. 10 that captures some of the geometric complexities of the one in Fig. 1. The objective of this section is to extend the algebraic reduction approach to address such cantilevers.

Observe that the cantilever in Fig. 10 poses significant challenges compared to the simple rectangular beam treated in the previous section. Specifically:

1. To *capture the geometry*, numerous (tetrahedral and/or hexahedral) finite elements are needed. Thus, *how does one define and compute the* 3D *bending stiffness matrix* $K^{3D:B}$ *and the associated forcing vector*?
2. Next, *to capture the physics*, numerous (virtual) beam elements may be needed across the length of the beam. Thus, *how does one reduce the* 3D *bending stiffness matrix to an appropriate algebraic reduction matrix* K^{AR}?
3. Finally, observe that a portion of the cantilever (including the fillet) must be treated as a 3D structural element, i.e., the physics cannot be reduced to a 1D beam. Thus, *how does one couple* 3D *FEA with the algebraic reduction scheme*?

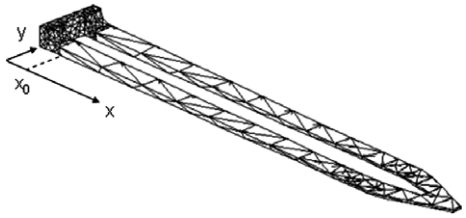


Fig. 11. Thick and thin regions within the cantilever.

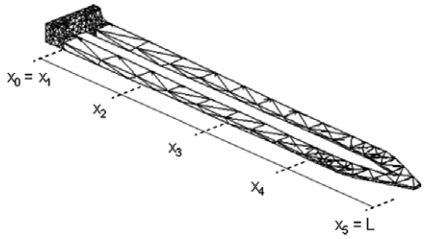


Fig. 12. The virtual beam space.

These questions are addressed below in sequence.

3.1. Computing the 3D bending stiffness matrix

Fig. 11 illustrates a typical finite element tetrahedral mesh of the cantilever. Observe that the elements in the thin region are of poor quality; hence a direct 3D FEA is bound to fail. Therefore, we construct below a 3D bending stiffness matrix for the poor quality elements located to the right of the fillet, i.e., for all elements that lie in the region defined by $x \ge x_0$.

Recall the definition of the 3D bending stiffness matrix for a single 3D element in Eq. (2.14). Since we now have many elements, this definition is generalized to:

$$K^{3D:B} \equiv \sum_{e(x \ge x_0)} \left(E^e \int_{\Omega^e} N_{,x}^T N_{,x} \mathrm{d}\Omega \right) \tag{3.1}$$

where $\sum$ stands for standard FEA assembly process. Inhomogeneous material properties can be handled with ease in this phase. The size of $K^{3D:B}$ is (N, N), where N is the number of nodes in the high aspect ratio region, including the nodes at the interface $x = x_0$.

For the 3D structural block ($0 \le x \le x_0$), we assemble the standard 3D finite element matrix [2]:

$$K^{3D} \equiv \sum_{e(x \le x_0)} \left(\int_{\Omega^e} B^T D B \mathrm{d}\Omega \right). \tag{3.2}$$

The size of K^{3D} is $(3M, 3M)$, where M is the number of nodes in the 'solid' region, again including the interface. Similarly, we assemble the two forcing vectors $q^{3D:B}$ and q^{3D}.

3.2. Beam subspace reduction

Unlike the reduction in Eq. (2.17), where a single (virtual) beam element was used, multiple (virtual) beam elements are now essential to capture the physics (due to tapering). In other words, the high aspect ratio region $x_0 \le x \le L$ is broken into distinct segments:

$$x_0 = X_1 < X_2 < X_3 < \cdots < X_K = L.$$

For example, Fig. 12 illustrates the case when $K = 5$.

For simplicity, it is also assumed that the global coordinate system is aligned with principle axes of the cross section

(see Fig. 11), therefore $v_0(x)$ can be ignored in Eq. (2.24) and thus, algebraic reduction degrees of freedom are:

$$d^{\mathrm{AR}} = \begin{bmatrix} w_1 & \theta_1^z & \cdots & w_K & \theta_K^z & u_1 & u_2 & u_3 & \cdots & u_{2K-1} \end{bmatrix}^T \tag{3.3}$$

and the total number of virtual degrees of freedom is:

$$\left|d^{\mathrm{AR}}\right| = 4K - 1. \tag{3.4}$$

The projection matrices $P_u^{3D:B}$ and $P_w^{3D:B}$ which project the poor quality region to beam subspace are of size $(N, 4K-1)$ and are computed as follows. Given the x coordinate of the ith 3D node in the high aspect ratio region, we first determine the virtual beam element k that the node lies within, i.e., find k such that:

$$X_k < x_i \leq X_{k+1}. \tag{3.5}$$

We then compute the corresponding non-dimensional spatial parameter:

$$t_i = \frac{x_i - X_k}{(X_{k+1} - X_k)}. \tag{3.6}$$

Finally, the non-zero entries in the projection matrices are computed by:

$$\begin{aligned} P_u^{3D:B}(i, 2k-1 : 2k+2) &= -z_i H_{,x}(t_i) \\ P_u^{3D:B}(i, 2K+2k-1 : 2K+2k+1) &= l^2(t_i) \end{aligned} \tag{3.7}$$

and

$$P_w^{3D:B}(i, 2k-1 : 2k+2) = H(t_i). \tag{3.8}$$

These matrices essentially impose the constraints in Eq. (2.24). Moreover, we define an identity projection matrix for the nodes in the 3D structural block region ($0 \leq x < x_0$):

$$P^{3D} = I. \tag{3.9}$$

Nodes at the interface $x = x_0$ are excluded in Eq. (3.9), thereby I in the above equation is a $(3M - 3M_I, 3M - 3M_I)$ identity matrix where M is defined previously and M_I is the number of nodes at the interface. The usefulness of this matrix will be clear in the following section.

3.3. Assembly and solution

In Sections 3.1 and 3.2 we computed an appropriate stiffness matrix, a force vector and a corresponding projection matrix for each of the *thin* and *thick* regions of the cantilever. These are assembled now via standard FEA assembly process:

$$\begin{aligned} K &= \sum (K^{3D}, K^{3D:B}) \\ q &= \sum (q^{3D}, q^{3D:B}) \\ P &= \sum (P^{3D}, P^{3D:B}) \end{aligned} \tag{3.10}$$

which are then reduced via:

$$\begin{aligned} K^{\mathrm{AR}} &= P^T K P \\ q^{\mathrm{AR}} &= P^T q. \end{aligned} \tag{3.11}$$

Eq. (3.11) automatically couples the thin and thick regions due to exclusion of interface nodes in P^{3D} in Eq. (3.9). After enforcing the essential boundary conditions at $x = 0$, the algebraic equations resulting from Eq. (3.11) will be well conditioned and free of locking. Once this system of equations is solved, the 3D degrees of freedom for the entire mesh can be post-processed as stated earlier. This method of coupling reduced and unreduced regions guarantees the continuity of displacement field but not strain (stress) field. For higher compatibility the multipoint coupling technique introduced in [7] can be used.

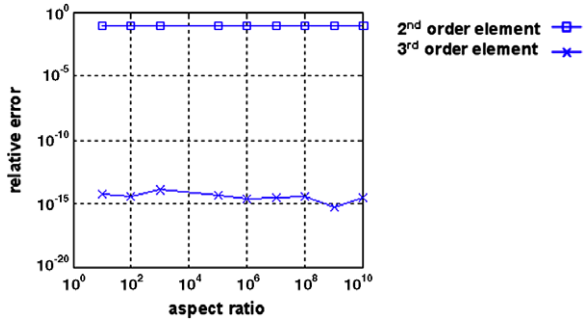


Fig. 13. Tip deflection error for 2nd and 3rd order tetrahedral mesh for the cantilever beam in Fig. 6.

4. Numerical experiments and case study

In the numerical experiments and case study below we assume that $E = 2e^{11}$, $\upsilon = 0.33$ and $q_0 = 1$ unless otherwise stated.

4.1. Numerical experiment 1

Recall from Section 2 that, for the problem posed in Fig. 6, the algebraic reduction method yields the exact Euler–Bernoulli (EB) matrix when a single 2nd order hexahedral element is used to capture the geometry. Here we show that tetrahedral elements can also be used in the algebraic reduction method (tetrahedral elements are often preferred for complex geometries).

In particular, the beam in Fig. 6 is meshed with (a) six 2nd order (Lagrangian) tetrahedrons and (b) six 3rd order tetrahedrons (this amounts to one tetrahedral element across the length of the beam). In both cases, a single virtual beam element will be used in the algebraic reduction scheme.

In case (a), the finite element space does *not* span the required space defined in Eq. (2.23), and therefore the solution will differ from the classic EB solution. On the other hand, for case (b) one can show that the FEA space spans Eq. (2.23), and therefore the solution will be identical to that of the classic EB model. This is confirmed in Fig. 13 where the relative error in tip deflection between algebraic reduction method and 1D Euler–Bernoulli beam model is plotted. As this figure illustrates, relative error of 3rd order elements is on the order of machine precision, while that of 2nd order elements is approximately 7% (but independent of aspect ratio). Increasing the number of tetrahedral elements for case (a) will reduce the error significantly, as demonstrated in the next experiment.

4.2. Numerical experiment 2

The second experiment further highlights the subtle interplay between the finite element space and the virtual beam element subspace.

It was shown in the last experiment that 2nd order tetrahedrons do not span the space defined in Eq. (2.23), thereby resulting in a 7% error. In this experiment, we show the effect of h-refinement of 2nd order Lagrangian tetrahedrons. The problem is identical to experiment 1 except that we fix the aspect ratio to $L/H = 1000$ and study the relative error in tip deflection as the 3D mesh gets refined (with 1 virtual beam element).

Fig. 14 illustrates the results; as the number of 2nd order tetrahedrons increases, the underlying finite element space better approximates the required space in Eq. (2.23). Note that since the size of the final algebraic reduction stiffness matrix (that needs to be 'inverted') is governed by the number of virtual beam elements, the computational cost only increases weakly with increasing 3D mesh size.

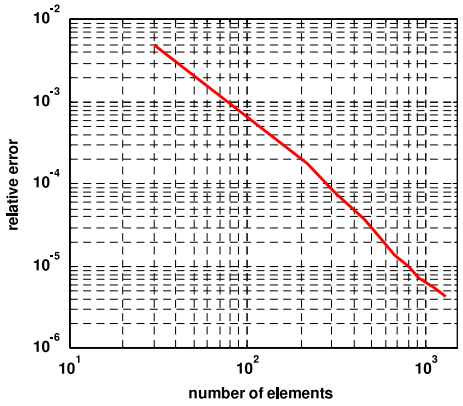


Fig. 14. *h*-refinement of the FEA mesh.

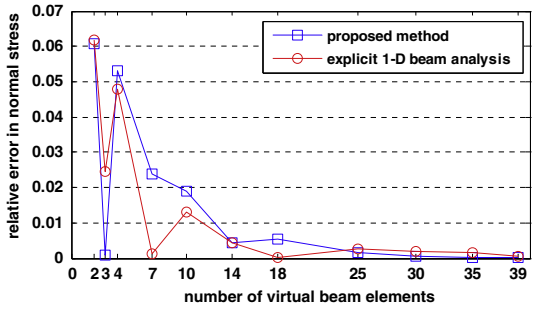


Fig. 15. *h*-refinement of virtual beam elements.

4.3. Numerical experiment 3

The last two examples involved a single virtual beam element. In this example the cantilever beam in Fig. 6 ($L/H = 1000$) is subject to a sinusoidal load at the top surface, i.e., $\sin(4\pi x/L)$ (and tip load $q_0 = 0$). As it is expected, in order to capture the sinusoidal loading, multiple virtual beam elements are required. Also we use 2nd order tetrahedrons to mesh the structure. For a fixed 3D mesh (2nd order tetrahedrons), as the number of virtual beam elements increases, the FEA space within each virtual beam element shrinks. This, as shown in the last experiment, causes inaccuracy. Therefore, the number of 3D finite elements must proportionately increase. In other words, as the number of virtual beam elements increases, we refine the 3D mesh such that each virtual beam element spans at least 2 tetrahedrons in the longitudinal direction. Although the 3D mesh gets refined, it is still of poor quality and suffers from Poisson and shear locking/ ill-conditioning (by conventional measures).

The normal stress in the x direction on the top surface at $x = 0.4L$ is computed as a function of number of virtual beam elements and compared against the analytical solution obtained via stress–moment relationship.

For the sake of comparison, in each case we also compute the stress using explicit 1D beam FEA (real beam elements) with the same number of virtual beam elements used above. The relative errors of both methods are computed against analytical solution as illustrated in Fig. 15. As it can be observed, *h*-refinement of virtual beam elements successfully improves the accuracy, as does the *h*-refinement of classic (real) 1D beam elements. The accuracy is not monotonic in either case, as finite element theory only guarantees that the error will drop monotonically for small mesh size. Neither one is better than the other for the entire range; but in the limit, we expect the two methods to converge identically.

4.4. Numerical experiment 4

In the fourth example we show the ability of the method to handle beams with varying cross section such as the one in Fig. 16.

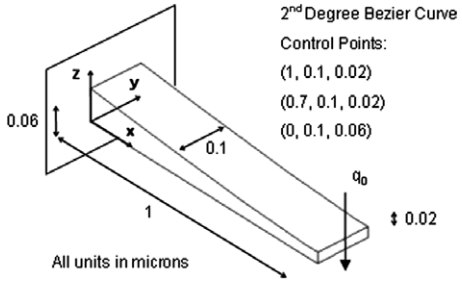


Fig. 16. A beam with non-uniform cross section.

Table 1
Relative error in the tip deflection.

Percentage of relative error	1 virtual element (%)	2 virtual elements (%)	3 virtual elements (%)
	4.7	2.8	1.5

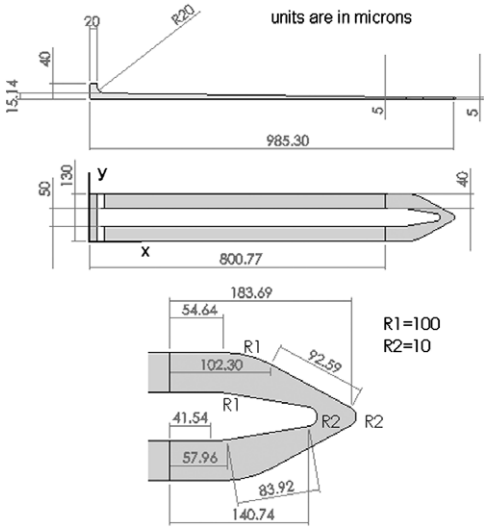


Fig. 17. Varying cross section micro-cantilever.

The cross section of the beam is defined via a 2nd degree Bezier curve with the control points specified in Fig. 16.

The beam is meshed with 2nd order tetrahedral elements, and is then reduced to 1, 2 and 3 virtual beam elements of equal lengths. Here again a tetrahedral mesh is created such that every virtual beam element contains 2 tetrahedral elements across the length. The relative error in tip deflection for each case is compared in Table 1 against a reference solution obtained via 3D FEA with a fine mesh involving 500 2nd order tetrahedral elements. As can be seen, h-refinement of virtual beam elements successfully improves the accuracy when the underlying geometry is complex.

4.5. Case study

In the case study below, we solve the problem shown in Fig. 10 with geometric specifications shown in Fig. 17. Such cantilevers are used as vibration sensors when fabricated as MEMS devices [1]. The micro-cantilever is fixed at the left end and a tip load of 12.056 μN is applied at the right end. A typical 3D 2nd order tetrahedral mesh is shown in Fig. 11.

We shall compute σ_x on the top surface at $x = 425$. An analytical solution is easily obtained via a moment–stress relation. The high aspect ratio region $50 \leq x \leq 985.30$ is reduced to 1, 2, 3 and 4 virtual beam elements of equal length and compared against

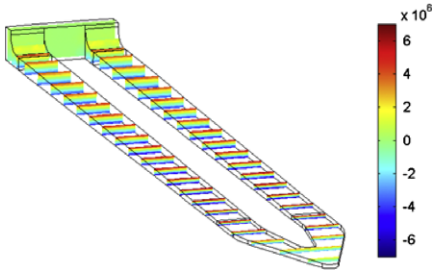


Fig. 18. Bending-stress of micro-cantilever.

Table 2
Error in the normal stress on top surface at $x = 425$.

Percentage of relative error	1 virtual element (%)	2 virtual elements (%)	3 virtual elements (%)	4 virtual elements (%)
	12.4	6.5	5.1	0.63

analytical σ_x. Table 2 shows this comparison. As can be seen from this table, the method is able to predict the quantity of interest with high accuracy despite the fact that the 3D mesh is coarse and quadratic tetrahedral elements are used. As a post-processing step, Fig. 18 illustrates the normal stress computed using 4 virtual beam elements.

5. Conclusion

In this paper we proposed an algebraic reduction method for analyzing beams. This method provides a dual-representation framework which builds upon, and combines the advantages of two well established methods, namely, 3D FEA and 1D beam analysis. Consequently, not only is the proposed method computationally efficient and accurate, but also it can be easily integrated and automated within a modern 3D CAD system. However since the underlying physics is based on 1D beam analysis, some of the limitations associated with explicit 1D beam analysis (such as tendency to ignore stress concentration) are inherited by the proposed method.

While this paper focused primarily on static linear elasticity, the proposed method can be easily extended to dynamic problems. Furthermore, algebraic reduction methods that combine, for example, 3D FEA with classic 2D plate methods are currently being developed.

Appendix. Extracting $K^{3D:B}$ from K^{3D}

Recall the definition of the bending stiffness matrix $K^{3D:B}$ in Eq. (2.14), reproduced below:

$$K^{3D:B} = \int_\Omega N_{,x}^T N_{,x} \mathrm{d}\Omega. \tag{A.1}$$

While one can easily compute $K^{3D:B}$ from 'scratch' (via Gaussian integration), we ask here if it can be extracted from the 3D FEA stiffness matrix K^{3D} in a numerically robust manner. This is of interest since commercial codes exist today that can rapidly compute K^{3D} in a robust manner.

Recall the definition of the 3D FEA stiffness matrix [2]:

$$K^{3D} = \int_\Omega B^T D B \mathrm{d}\Omega \tag{A.2}$$

where

$$B^T = \begin{bmatrix} N_{,x} & 0 & 0 & N_{,y} & 0 & N_{,z} \\ 0 & N_{,y} & 0 & N_{,x} & N_{,z} & 0 \\ 0 & 0 & N_{,z} & 0 & N_{,y} & N_{,x} \end{bmatrix} \tag{A.3}$$

$$D = \bar{E} \begin{bmatrix} \begin{bmatrix} 1-\nu & \nu & \nu \\ \nu & 1-\nu & \nu \\ \nu & \nu & 1-\nu \end{bmatrix} & 0_{(3,3)} \\ 0_{(3,3)} & \left(\frac{1-2\nu}{2}\right) I_{(3,3)} \end{bmatrix} \tag{A.4}$$

where:

$$\bar{E} = \frac{E}{(1+\nu)(1-2\nu)}. \tag{A.5}$$

Upon expanding, we find that:

$$K^{3D} = \bar{E} \int_\Omega \begin{bmatrix} \left\{ \begin{matrix} (1-\nu) N_{,x}^T N_{,x} + \\ \frac{1-2\nu}{2} \left(\begin{matrix} N_{,y}^T N_{,y} + \\ N_{,z}^T N_{,z} \end{matrix} \right) \end{matrix} \right\} & \left\{ \begin{matrix} \nu N_{,x}^T N_{,y} + \\ \frac{1-2\nu}{2} N_{,y}^T N_{,x} \end{matrix} \right\} & \cdots \\ & \left\{ \begin{matrix} (1-\nu) N_{,y}^T N_{,y} + \\ \frac{1-2\nu}{2} \left(\begin{matrix} N_{,x}^T N_{,x} + \\ N_{,z}^T N_{,z} \end{matrix} \right) \end{matrix} \right\} & \cdots \\ \text{Sym} & & \cdots \end{bmatrix} \mathrm{d}\Omega. \tag{A.6}$$

Observe that the desired bending matrix (see Eq. (A.1)) is embedded within Eq. (A.6). However, a major hurdle in extracting $K^{3D:B}$ from Eq. (A.6) is that, if the element is of high aspect ratio, it follows that:

$$\frac{N_{,x}^T N_{,x}}{N_{,z}^T N_{,z}} = O\left((H/L)^2\right). \tag{A.7}$$

Therefore, in all the diagonal entries in Eq. (A.6), $N_{,z}^T N_{,z}$ will dominate over $N_{,x}^T N_{,x}$ as $(H/L) \to 0$, and it will be numerically impossible to extract $N_{,x}^T N_{,x}$ through algebraic manipulation.

However, it is still possible to extract the desired matrix by first creating a pseudo-mesh by scaling the y and z coordinates of the mesh by a factor β, where:

$$\beta \sim O(L/H). \tag{A.8}$$

The precise choice of β is not critical and it will disappear in the formulation below. Once this is done, we set $E = 1$ and $\nu = 0$ in Eq. (A.6). In other words, we construct a *pseudo-stiffness matrix* defined by:

$$K^{3D:P} = \int_\Omega B_\beta^T D^0 B_\beta \mathrm{d}\Omega \tag{A.9}$$

where the B_β matrix is now given by:

$$B_\beta^T = \begin{bmatrix} N_{,x} & 0 & 0 & N_{,y}/\beta & 0 & N_{,z}/\beta \\ 0 & N_{,y}/\beta & 0 & N_{,x} & N_{,z}/\beta & 0 \\ 0 & 0 & N_{,z}/\beta & 0 & N_{,y}/\beta & N_{,x} \end{bmatrix} \tag{A.10}$$

and

$$D^0 = \begin{bmatrix} I_{(3,3)} & 0_{(3,3)} \\ 0_{(3,3)} & \frac{1}{2} I_{(3,3)} \end{bmatrix}. \tag{A.11}$$

Indeed, from Eq. (A.9), we find that the pseudo-stiffness matrix is given by (compare against Eq. (A.6)):

$$K^{3D:P} = \int_\Omega \begin{bmatrix} \left\{ \begin{matrix} N_{,x}^T N_{,x} + \\ \frac{1}{2\beta^2} \left(\begin{matrix} N_{,y}^T N_{,y} \\ + N_{,z}^T N_{,z} \end{matrix} \right) \end{matrix} \right\} & \frac{1}{2\beta} N_{,y}^T N_{,x} & \cdots \\ & \left\{ \begin{matrix} \frac{1}{2} N_{,x}^T N_{,x} + \\ \frac{1}{2\beta^2} \left(\begin{matrix} 2 N_{,y}^T N_{,y} \\ + N_{,z}^T N_{,z} \end{matrix} \right) \end{matrix} \right\} & \cdots \\ \text{Sym} & & \cdots \end{bmatrix} \mathrm{d}\Omega. \tag{A.12}$$

Given the pseudo-stiffness matrix $K^{3D:P}$, a simple check shows that one can now extract the bending stiffness matrix as follows:

$$K^{3D:B} = \frac{3}{2}K_{uu}^{3D:P} - \frac{1}{2}\left(K_{vv}^{3D:P} + K_{ww}^{3D:P}\right). \tag{A.13}$$

The above operation is indeed numerically stable. Further, although we have considered a single element, Eq. (A.13) can be applied to a finite element mesh, in that one can 'scale' an entire mesh, compute the pseudo-stiffness matrix (via finite element assembly), and finally extract the bending stiffness matrix via Eq. (A.13).

References

[1] Lee CS, Nam HJ, Kim YS, Jin WH, Cho SM, Bu JU. Microcantilevers integrated with heaters and piezoelectric detectors for nano data-storage application. Applied Physics Letters 2003;83(23):4839–41.
[2] Shames IH, Dym CL. Energy and finite element methods in structural mechanics. New York: Hemisphere Publishing Corporation; 1985.
[3] Pilkey W. Analysis and design of elastic beams. New York (NY): John Wiley; 2002.
[4] Dow J, Byrd DE. The identification and elimination of artificial stiffening errors in finite elements. International Journal of Numerical Methods in Engineering 1988;26(3):743–62.
[5] Cook RD, Malkus DS, Plesha ME, Witt RJ. Concepts and applications of finite element analysis. John Wiley & Sons. Inc.; 2002. p. 102.
[6] Suresh K. Automating the CAD/CAE dimensional reduction process. In: 8th ACM solid modeling conference. Seattle (USA): ACM; 2003.
[7] Donaghy RJ, McCune W, Bridgett SJ, Armstrong CG, Robinson DJ, McKeag RM. Dimensional reduction of analysis models. In: 5th international meshing roundtable. 1996.
[8] Yang HTY, Saigal S, Masud A, Kapania RK. A survey of recent shell finite element. International Journal of Numerical Methods in Engineering 2000; 47(1):101–27.
[9] Duster A, Broker H, Rank E. The *p*-version of finite element method for three-dimensional curved thin walled structures. International Journal of Numerical Methods in Engineering 2001;52(7):673–703.
[10] Braess D, Kaltenbacher M. Efficient 3d-finite-element-formulation for thin mechanical and piezoelectric structures. International Journal of Numerical Methods in Engineering 2008;73(2):147–61.
[11] Dorfmann A, Nelson RB. Three-dimensional finite element for analysing thin plate/shell structures. International Journal of Numerical Methods in Engineering 1995;38(20):3453–82.
[12] Jog CS. Topology design of three-dimensional structures using hybrid finite elements. In: Bendsøe MP, Olhoff N, Sigmund O, editors. IUTAM symposium on topological design optimization of structures, machines and materials. Springer Netherlands; 2006. p. 33–42.
[13] Cirak F, Scott MJ, Antonsson EK, Ortiz M, Schroder P. Integrated modeling, finite element analysis, and engineering desing for thin-shell structures using subdivision. Computer Aided Design 2002;34(2):137–48.
[14] Armstrong CG, Robinson DJ, McKeag RM, Li TS, Bridgett SJ, Donaghy RJ. et al. Medials for meshing and more. In: Proceedings of 4th international meshing roundtable. Sandia National Lab. 1995.
[15] Suresh K. Skeletal reduction of boundary value problems. International Journal of Numerical Methods in Engineering 2006;66(4):722–39.
[16] Goldak J, Yu X, Knight A, Dong L. Constructing discrete medial axis of 3-D objects. International Journal of Computational Geometry and Applications 1991;1(3):327–39.
[17] Sheehy DJ, Armstrong CJ, Robinson DJ. Computing the medial surface of a solid from a domain delaunay triangulation. IEEE Transactions on Visualization and Computer Graphics 1996;2(1):44–61.
[18] Sherbrooke EC, Patrikalakis NM, Brisson E. An algorithm for the medial axis transform of 3-D polyhedral solids. IEEE Transactions on Visualization and Computer Graphics 1995;2(1):62–72.
[19] Yang Y, Brock O, Moll RN. Efficient and robust computation of an approximated medial axis. In: ACM symposium on solid modeling and applications. Genoa, Italy. 2004.
[20] Wang CM, Reddy JN, Lee KH. Shear deformable beams and plates: relationship to classical solutions. London: Elsevier Science; 2000.
[21] Bhatti MA. Fundamental finite element analysis and applications. 1st ed. New York: John Wiley; 2005.

Kavous Jorabchi received his B.S. from University of Tehran, Iran and joined Engineering Representations and Simulation Laboratory (ERSL) at the University of Wisconsin-Madison to pursue his Ph.D. in Mechanical Engineering. His research interest includes CAD, Finite Element Analysis, Model Reduction and Design Automation.

Joshua Danczyk received his B.S. from the University of Wisconsin-Madison with Honors in Research and Highest Distinction, and has continued his work in the Engineering Representations and Simulation Laboratory (ERSL) at the UW-Madison while pursuing his Ph.D. His research interests include Finite Element Analysis, Model Simplification, Optimization, and CAD.

Krishnan Suresh is an Associate Professor in the Department of Mechanical Engineering, University of Wisconsin-Madison. His research interests are in finite element analysis, shape and topology optimization, high-performance computing, and bio-mechanics modeling.