



Topology optimization using polytopes

Arun L. Gain^a, Glaucio H. Paulino^{a,b,*}, Leonardo S. Duarte^c, Ivan F.M. Menezes^c

^a *Department of Civil & Environmental Engineering, University of Illinois at Urbana-Champaign, USA*

^b *School of Civil and Environmental Engineering, Georgia Institute of Technology, USA*

^c *Tecgraf, Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Brazil*

Received 28 August 2014; received in revised form 4 May 2015; accepted 6 May 2015

Available online 14 May 2015

Abstract

Meshing complex engineering domains is a challenging task. Arbitrary polyhedral meshes can provide the much needed flexibility in automated discretization of such domains. The geometric property of polyhedral meshes such as its unstructured nature and the connectivity of faces between elements makes them specially attractive for topology optimization applications. Numerical anomalies in designs such as the single node connections and checkerboard pattern can be naturally circumvented with polyhedrons. In the current work, we solve the governing three-dimensional elasticity state equation using the Virtual Element Method (VEM) approach. The main characteristic difference between VEM and standard finite element methods (FEM) is that in VEM the canonical basis functions are not constructed explicitly. Rather the stiffness matrix is computed directly utilizing a projection map which extracts the linear component of the deformation. Such a construction guarantees the satisfaction of the patch test (used by engineers as an indicator of optimal convergence of numerical solutions under mesh refinement). Finally, the computations reduce to the evaluation of matrices which contain purely geometric surface facet quantities. The present work focuses on the first-order VEM in which the degrees of freedom are associated with the vertices. The features of the current optimization approach are demonstrated using numerical examples for compliance minimization and compliant mechanism problems.

© 2015 Elsevier B.V. All rights reserved.

Keywords: Density-based method; Virtual Element Method; Polyhedrons; Voronoi tessellation

1. Introduction

Numerical simulation of a typical engineering problem often begins with meshing a complicated domain, which is a challenging task. In general, unstructured meshes are preferred as they have shown to produce reliable and more accurate numerical solutions [1]. Thus, the polyhedral elements have become increasingly popular due to the flexibility they impart in automatic discretization of complicated design domains. Also, physical attributes such as facial connectivity between neighboring polyhedrons and unstructured nature of the mesh, make them specially attractive in topology optimization. Due to their characteristic geometry, numerical anomalies such as single node connections and checkerboarding are naturally alleviated. Orthogonally intersecting tension and compression members, distinctive

* Corresponding author.

E-mail address: paulino@gatech.edu (G.H. Paulino).

Nomenclature

a, a_h	Continuous and discrete bilinear form
f	Continuous load linear form
\mathbf{t}	Surface tractions
\mathbf{C}	Elasticity tensor
Ω	Topology optimization working domain
$\mathcal{V}, \mathcal{V}_h$	Continuous and conforming discrete solution space
J	Objective function
ρ, ρ_e	Continuous density function and discrete element density
V_f	Prescribed volume fraction
$ \cdot $	Measure (area or volume) of a set. Also, Euclidean norm of vector
\mathcal{W}	Space of finite dimensional smooth functions over element e
a^e, a_h^e	Restriction of a, a_h to \mathcal{W} (element contribution)
s^e	Approximation of bilinear form corresponding to higher-order deformation modes
φ_i	Generic barycentric coordinates
$\boldsymbol{\varphi}_i$	Canonical basis function
$\bar{\mathbf{v}}$	Mean of values of \mathbf{v} sampled over the vertices of element e
$\langle \mathbf{v} \rangle$	Volume average of \mathbf{v}
\mathcal{P}	Space of linear deformation mode
\mathbf{p}_i	Bases spanning space \mathcal{P}
$\pi_{\mathcal{P}}$	Projection map to extract \mathcal{P} , $\pi_{\mathcal{P}} : \mathcal{W} \rightarrow \mathcal{P}$
$\mathbf{P}_{\mathcal{P}}$	Matrix representation of $\pi_{\mathcal{P}}$
$\mathbf{W}_{\mathcal{P}}$	Matrices containing surface integration quantities
$\mathbf{N}_{\mathcal{P}}$	Matrices containing rearranged nodal coordinates of vertices
\mathbf{q}_i	Vector of facial surface integration of barycentric coordinates
\mathbf{K}_e	Element stiffness matrix
\mathbf{D}	Material matrix which is function of elasticity tensor \mathbf{C}
\mathbf{I}	Identity matrix
α^e	Positive scaling coefficient for stability term s^e
\mathcal{S}	Set of distinct seeds to generate Voronoi cells
E	Young's modulus
ν	Poisson's ratio
\mathbf{K}	Global stiffness matrix
\mathbf{F}	Global force vector
\mathbf{U}	Global displacement vector
r_{min}	Filter radius

features of an optimum structural layout [2,3], are conveniently captured by polyhedral meshes. Thus, the goal of the current work is to develop a numerically efficient and accurate topology optimization approach for arbitrary polyhedral elements.

In the past, special interpolants have been proposed for arbitrary shaped elements such as Wachspress [4,5], Sibson coordinates [6,7], non-Sibson coordinates or Laplace shape functions [8–10,7,11], mean value coordinates [12,13], metric coordinate method [14], and maximum entropy shape function [15–17]. A detailed overview of the main developments in the field of conforming polygonal interpolants is provided in [18]. A summary of the polygonal/polyhedral interpolants is shown in Table 1.

For two-dimensional topology optimization, polygonal shape functions have been explored in the past [23–25]. Utilizing the iso-parametric mapping scheme for numerical integration, polygonal shape functions can be efficiently implemented [11]. Polygonal shape functions and their derivatives are computed once for different reference n-gons and stored. Subsequently, these quantities can be retrieved as and when required. In three-dimensions, however, such a mapping scheme does not exist. Thus, numerical integration for each element are performed in physical coordinates

Table 1
Summary of the polygonal/polyhedral interpolants.

Interpolants	References	Concave elements	Remarks
Wachspress	[4,5,19]	No	Earliest interpolant based on rational polynomials
Sibson	[6,7]	No	Utilizes Voronoi tessellations to construct interpolant which reduces to the ratios of areas of Voronoi cells
Non-Sibson	[8–10,7]	No	Also based on Voronoi tessellations. The interpolant is a function of Lebesgue measure of Voronoi edge and L^2 distance norm
Mean value coordinates	[12,13]	Yes	Interpolant is a function of geometric quantities — L^2 distance norm and area
Metric coordinate method	[14]	Yes	Uses measures such as edge length, signed area of triangle, and trigonometric functions of sine and cosine to construct the shape functions
Maximum entropy	[15–17]	Yes	Shape functions and their derivatives are obtained by maximizing the Shannon's entropy function under prescribed boundary conditions
Harmonic coordinates	[20–22]	Yes	Shape functions and their derivatives are obtained by solving the Laplace equation hierarchically

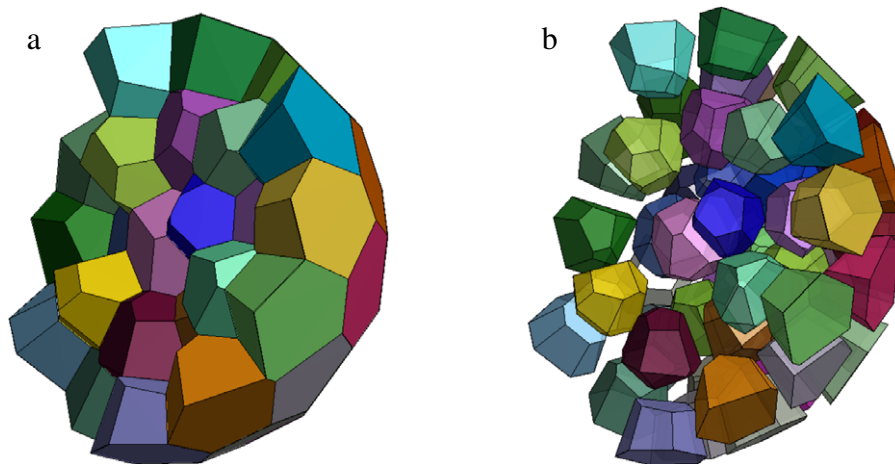


Fig. 1. Illustration of the complexity of elements in an unstructured polyhedral mesh. (a) A section of the mesh for spherical domain. (b) Split view of the mesh.

which is computationally expensive. Moreover, to achieve accurate results, a high order numerical quadrature is required. This difficulty arises due to the non-polynomial nature of these shape functions. Fig. 1 illustrates the complexity of elements in a typical polyhedral mesh.

Recently developed Virtual Element Method addresses some of the challenges facing the polyhedral elements for three-dimensional applications. The VEM originated from Mimetic Finite Difference (MFD) methods which have been successfully applied to diffusion [26], fluid flow [27] and elasticity problems [28]. The MFDs differ from the standard finite element approaches in the sense that in MFDs there are no explicitly defined shape functions associated with the discrete degrees of freedom. Thus, the continuous differential operators such as the div, grad, curl and trace, are approximated or *mimicked* by their discrete counterparts which utilize the discrete quantities defined only at the degrees of freedom. This provides greater flexibility in the geometric shapes of the admissible elements. In fact, high quality, skewed, degenerate and even non-convex polyhedra are all admissible. The MFD methods have been evolving from a finite difference/finite volume framework towards a Galerkin finite element-type framework called the Virtual Element Method [29,30]. The feature which distinguishes VEM from classical finite element methods is that VEM does not require explicit computation of the approximation space. The construction of the discrete bilinear form for the elasticity problem begins with the kinematic decomposition of the element deformation space into constant

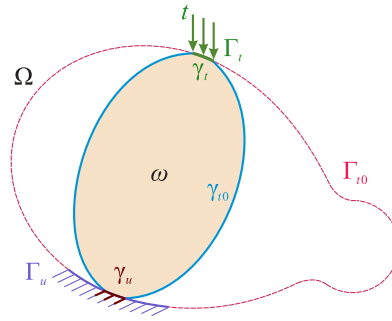


Fig. 2. Domain description for the topology optimization problem. The boundary, $\partial\Omega$, of the working domain, Ω , consists of Γ_u (displacement boundary), Γ_{t0} (homogeneous traction boundary) and Γ_t (non-homogeneous traction boundary). The design ω , with boundary $\partial\omega = \gamma_u \cup \gamma_{t0} \cup \gamma_t$, is constrained to satisfy $\gamma_u \subseteq \Gamma_u$ and $\gamma_t = \Gamma_t$. Boundaries γ_u , γ_{t0} , and γ_t correspond to displacement, homogeneous traction, and non-homogeneous traction boundary conditions on $\partial\omega$, respectively.

strain and higher-order modes. The linear deformations are captured exactly, thereby passing the engineering patch test. A projection map is defined which enables such a decomposition. Using the discrete version of the projection map, the element stiffness matrix is constructed which requires computation of surface integral of the canonical basis functions over the element boundary faces. In this work, we explore the effectiveness of first-order VEM¹ [30] for three-dimensional topology optimization considering linear elastic material behavior.

The remainder of this paper is organized as follows. In Section 2, we define the governing elasticity problem and discuss the topology optimization problem formulation. Section 3 provides a brief overview of the Virtual Element Method for linear elasticity. A centroidal Voronoi tessellation (CVT) based meshing approach used to generated the polyhedral meshes is discussed in Section 4. In Section 5, we explore several numerical examples to evaluate our current approach. Finally, we provide some concluding remarks in Section 6.

2. Topology optimization problem formulation

In this work, we concentrate on topology optimization of linearized elastic system under small deformations subjected to surface tractions \mathbf{t} . The elasticity problem, for a smooth bounded domain $\Omega \subseteq \mathbb{R}^3$, is defined as follows. Find \mathbf{u} :

$$a(\mathbf{u}, \mathbf{v}) = f(\mathbf{v}), \quad \forall \mathbf{u}, \mathbf{v} \in \mathcal{V} \tag{1}$$

where

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \mathbf{C}\boldsymbol{\epsilon}(\mathbf{u}) : \boldsymbol{\epsilon}(\mathbf{v}) \, dx, \quad f(\mathbf{v}) = \int_{\Gamma_t} \mathbf{t} \cdot \mathbf{v} \, ds \tag{2}$$

$$\mathcal{V} = \left\{ \mathbf{v} \in H^1(\Omega)^3 : \mathbf{v}|_{\Gamma_u} = \mathbf{0} \right\}.$$

Here, $\boldsymbol{\epsilon}(\mathbf{u}) = 1/2(\nabla\mathbf{u} + \nabla\mathbf{u}^T)$ is the second-order linearized strain tensor and \mathbf{C} is the elasticity tensor. In topology optimization, the working domain, Ω , contains all the admissible shapes ω , i.e., $\omega \subseteq \Omega$. Its boundary $\partial\Omega$ consists of three disjoint segments, $\partial\Omega = \Gamma_u \cup \Gamma_{t0} \cup \Gamma_t$, where Γ_u , Γ_{t0} , and Γ_t represent displacement, homogeneous traction, and non-homogeneous traction boundary conditions ($\mathbf{t} \neq \mathbf{0}$), respectively. Also, the design ω , with boundary $\partial\omega = \gamma_u \cup \gamma_{t0} \cup \gamma_t$, is constrained to satisfy $\gamma_u \subseteq \Gamma_u$ and $\gamma_t = \Gamma_t$. Here, γ_u , γ_{t0} , and γ_t correspond to the boundaries of ω with displacement, homogeneous traction, and non-homogeneous traction boundary conditions, respectively (c.f. Fig. 2). Body forces are ignored. The VEM is used to solve the state Eq. (1).

We shall focus on two categories of topology optimization problems, compliance minimization and linear compliant mechanism. The topology optimization problem of compliance minimization refers to finding the stiffest configuration under applied loads and boundary conditions. Compliance, the work done by the loads, is defined as:

$$J_1(\rho) = \int_{\Gamma_t} \mathbf{t} \cdot \mathbf{u} \, ds = \int_{\Omega} \mathbf{C}(\rho) \boldsymbol{\epsilon}(\mathbf{u}) : \boldsymbol{\epsilon}(\mathbf{u}) \, dx. \tag{3}$$

¹ Polyhedral elements are considered to be linear with three degrees of freedom associated with each vertex.

The effective elasticity tensor \mathbf{C} for the domain Ω is a function of density $\rho(\mathbf{x})$ and, as per the Solid Isotropic Material with Penalization (SIMP) model [31–33], is expressed as:

$$\mathbf{C}(\rho) = [\varepsilon + (1 - \varepsilon) \rho^p] \mathbf{C}^0. \tag{4}$$

The solid and void regions are filled with material of elasticity tensor \mathbf{C}^0 and $\varepsilon \mathbf{C}^0$, respectively, where ε is chosen as 10^{-4} . The penalization parameter, p , is set to 3.

The second category of problems we study are the linear compliant mechanisms, specifically the displacement inverter and the gripper problem. The objective is to maximize the displacement in a predefined direction, u_{out} , in response to the force, f_{in} , exerted by the actuator, modeled by a spring of stiffness k_{in} . So, the quantity we aim to minimize is:

$$J_2(\rho) = -u_{out}. \tag{5}$$

Thus, the combined optimization problem, (3), (5), can be expressed as:

$$\begin{aligned} & \inf_{\rho} J_i(\rho) \quad \text{for } i = 1, 2 \\ \text{subject to: } & a(\mathbf{u}, \mathbf{v}) = f(\mathbf{v}), \quad \int_{\Omega} \rho(\mathbf{x}) \, d\mathbf{x} \leq V_f |\Omega| \end{aligned} \tag{6}$$

where V_f is the prescribed maximum volume fraction and $|\cdot|$ denotes the measure (area or volume) of a set as well as the Euclidean norm of vector. In this work, we shall denote the components of vectors, matrices and tensors in the canonical Euclidean basis with subscripts inside parentheses (e.g. $\mathbf{u}_{(i)}$ or $\boldsymbol{\epsilon}_{(ij)}$) in order to make a distinction with indexed quantities.

For our simulations, we use gradient-based Optimality Criteria (OC) [34] as the optimization algorithm, the sensitivity analysis for which are shown in Appendix A. Other mathematical programming algorithms such as, Method of Moving Asymptotes (MMA) [35], Sequential Linear Programming (SLP), Sequential Quadratic Programming (SQP), and CONVex LINearization approximations (CONLIN) [36] have also been used in density-based methods.

3. Virtual Element Method for linear elasticity

In this section, we present the Virtual Element Method for linear elasticity [30,29] which is the governing state equation for the current topology optimization implementation. We discuss the mathematical framework of the method and also provide implementation details.

3.1. Theoretical background

Consider a partition of Ω into disjoint non-overlapping polyhedrons, e , of maximum diameter h . The Galerkin approximation \mathbf{u}_h of \mathbf{u} , obtained by solving the discrete counterpart of (1), belongs to the conforming discrete space $\mathcal{V}_h \subseteq \mathcal{V}$ which consists of continuous displacement fields whose restriction to polyhedron e belongs to the finite-dimensional space of smooth functions \mathcal{W} . Space \mathcal{W} contains all the deformation states represented by the element e — linear deformations and higher-order modes. Due to the conformity of \mathcal{V}_h , the continuous (a) and discrete bilinear form (a_h) can be expressed as the corresponding sums of element contributions:

$$a(\mathbf{u}, \mathbf{v}) = \sum_e a^e(\mathbf{u}, \mathbf{v}), \quad a_h(\mathbf{u}, \mathbf{v}) = \sum_e a_h^e(\mathbf{u}, \mathbf{v}). \tag{7}$$

In the space \mathcal{W} , three degrees of freedom are associated with each vertex of a polyhedron. For this purpose, we consider the canonical basis $\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_{3n}$ of the form

$$\boldsymbol{\varphi}_{3i-2} = [\varphi_i, 0, 0]^T, \quad \boldsymbol{\varphi}_{3i-1} = [0, \varphi_i, 0]^T, \quad \boldsymbol{\varphi}_{3i} = [0, 0, \varphi_i]^T, \quad i = 1, \dots, n \tag{8}$$

where $\varphi_1, \dots, \varphi_n$ are a set of *barycentric coordinates*, for instance [4,7,12,15,16], which satisfy all the desired properties of a conforming interpolants such as partition of unity, Kronecker-delta, linear completeness and piecewise linear (C^0 function) along the edges of e . We shall see later that VEM concerns only with the behavior of \mathcal{W} along the boundaries of the element e .

The VEM construction of the stiffness matrix begins with the kinematic decomposition of deformation states of \mathcal{W} . To facilitate the derivations, the mean of the values of a function v sampled at the vertices of e and volume average of v are represented by \bar{v} and $\langle v \rangle$, respectively. We define the bases that span the space of linear deformations, \mathcal{P} , over element e as:

$$\begin{aligned} p_1(\mathbf{x}) &= \mathbf{e}_1, & p_2(\mathbf{x}) &= \mathbf{e}_2, & p_3(\mathbf{x}) &= \mathbf{e}_3, \\ p_4(\mathbf{x}) &= (\mathbf{e}_1 \otimes \mathbf{e}_1)(\mathbf{x} - \bar{\mathbf{x}}), & p_5(\mathbf{x}) &= (\mathbf{e}_1 \otimes \mathbf{e}_2)(\mathbf{x} - \bar{\mathbf{x}}), & p_6(\mathbf{x}) &= (\mathbf{e}_1 \otimes \mathbf{e}_3)(\mathbf{x} - \bar{\mathbf{x}}), \\ p_7(\mathbf{x}) &= (\mathbf{e}_2 \otimes \mathbf{e}_1)(\mathbf{x} - \bar{\mathbf{x}}), & p_8(\mathbf{x}) &= (\mathbf{e}_2 \otimes \mathbf{e}_2)(\mathbf{x} - \bar{\mathbf{x}}), & p_9(\mathbf{x}) &= (\mathbf{e}_2 \otimes \mathbf{e}_3)(\mathbf{x} - \bar{\mathbf{x}}), \\ p_{10}(\mathbf{x}) &= (\mathbf{e}_3 \otimes \mathbf{e}_1)(\mathbf{x} - \bar{\mathbf{x}}), & p_{11}(\mathbf{x}) &= (\mathbf{e}_3 \otimes \mathbf{e}_2)(\mathbf{x} - \bar{\mathbf{x}}), & p_{12}(\mathbf{x}) &= (\mathbf{e}_3 \otimes \mathbf{e}_3)(\mathbf{x} - \bar{\mathbf{x}}). \end{aligned} \quad (9)$$

Next, we define projection map $\pi_{\mathcal{P}} : \mathcal{W} \rightarrow \mathcal{P}$ to extract linear deformations as:

$$\pi_{\mathcal{P}}v = \bar{v} + \langle \nabla v \rangle (\mathbf{x} - \bar{\mathbf{x}}). \quad (10)$$

We observe that the volumetric integral $\langle \nabla v \rangle$ can be converted as:

$$\langle \nabla v \rangle = \frac{1}{|e|} \int_e \nabla v \, d\mathbf{x} = \frac{1}{|e|} \int_{\partial e} v \otimes \mathbf{n} \, ds. \quad (11)$$

An important property of $\pi_{\mathcal{P}}$ which ensures consistency of the VEM approach is that $(v - \pi_{\mathcal{P}}v)$ is *energetically* orthogonal to \mathcal{P} , $\forall v \in \mathcal{W}$, i.e.

$$a^e(\mathbf{p}, v - \pi_{\mathcal{P}}v) = 0, \quad \forall \mathbf{p} \in \mathcal{P}, v \in \mathcal{W}. \quad (12)$$

To verify this identity, we can use the fact that the stress $\boldsymbol{\sigma}(\mathbf{p})$ is constant and $\langle \boldsymbol{\epsilon}(v) \rangle = \boldsymbol{\epsilon}(\pi_{\mathcal{P}}v)$, i.e. the volume average of the strain of a field v is the same as the strain of its linear deformation component, $\pi_{\mathcal{P}}v$. It will be useful to express the linear projection map $\pi_{\mathcal{P}}$ in terms of the bases of \mathcal{P} as:

$$\begin{aligned} \pi_{\mathcal{P}}v &= (\bar{v})_{(1)}p_1 + (\bar{v})_{(2)}p_2 + (\bar{v})_{(3)}p_3 + \langle \nabla v \rangle_{(11)}p_4 + \langle \nabla v \rangle_{(12)}p_5 + \langle \nabla v \rangle_{(13)}p_6 \\ &\quad + \langle \nabla v \rangle_{(21)}p_7 + \langle \nabla v \rangle_{(22)}p_8 + \langle \nabla v \rangle_{(23)}p_9 + \langle \nabla v \rangle_{(31)}p_{10} + \langle \nabla v \rangle_{(32)}p_{11} + \langle \nabla v \rangle_{(33)}p_{12}. \end{aligned} \quad (13)$$

Finally, using $\pi_{\mathcal{P}}$, any deformation state $v \in \mathcal{W}$ can be kinematically decomposed as:

$$v = \pi_{\mathcal{P}}v + (v - \pi_{\mathcal{P}}v). \quad (14)$$

The higher-order component $(v - \pi_{\mathcal{P}}v)$ belongs to a $(3n - 12)$ dimensional subspace of \mathcal{W} .

Based on the kinematic decomposition of deformation state (14) and the energy orthogonality property (12), the continuous bilinear form can be written as:

$$a^e(\mathbf{u}, v) = a^e(\pi_{\mathcal{P}}\mathbf{u}, \pi_{\mathcal{P}}v) + a^e(\mathbf{u} - \pi_{\mathcal{P}}\mathbf{u}, v - \pi_{\mathcal{P}}v). \quad (15)$$

The first term corresponds to the constant strain modes and can be computed exactly by knowing the volume of the element e (note that the arguments of the bilinear term are linear). Thus, ensuring that the engineering patch test is passed. The second term, corresponding to higher-order deformation modes, is difficult to compute. As observed in [37], the exact computation of the second term can be done away with and it can be replaced by a crude approximation s^e (which can be easily computed) without affecting the energy associated with linear deformation modes. Thus, the discrete bilinear form is defined as:

$$a^e(\mathbf{u}, v) \doteq a^e(\pi_{\mathcal{P}}\mathbf{u}, \pi_{\mathcal{P}}v) + s^e(\mathbf{u} - \pi_{\mathcal{P}}\mathbf{u}, v - \pi_{\mathcal{P}}v). \quad (16)$$

Here, s^e must be symmetric positive definite bilinear form on the space of higher-order deformations and should be chosen such that strain energy associated with higher-order deformation modes scales with the exact strain energy. This ensures stability of the method. Provided the aforementioned conditions are satisfied, there is quite a bit of freedom in the choice of s^e [37,30]. Here, we make a computationally inexpensive choice as [37]:

$$s^e(\mathbf{u}, v) = \sum_{i=1}^n \alpha^e \mathbf{u}(x_i) \cdot v(x_i) \quad (17)$$

where α^e is a positive coefficient that ensures correct scaling of the energies of higher-order mode. In the following section we provide explicit expressions for the element stiffness matrix.

3.2. Implementation details

Here we show the implementation details of the VEM formulation discussed in the previous section. We provide the expression for discrete linear projection map which is later used to construct the stiffness matrix terms. Let, $\mathbf{P}_{\mathcal{P}}$ be the discrete representation of the projection $\pi_{\mathcal{P}}$, i.e.,

$$\pi_{\mathcal{P}}\varphi_j = \sum_{k=1}^{3n} (\mathbf{P}_{\mathcal{P}})_{(kj)} \varphi_k. \tag{18}$$

To obtain $\mathbf{P}_{\mathcal{P}}$, we use (13) to express $\pi_{\mathcal{P}}$ in terms of its corresponding bases as:

$$\pi_{\mathcal{P}}\varphi_j = \sum_{\ell=1}^{12} (\mathbf{W}_{\mathcal{P}})_{(j\ell)} \mathbf{p}_{\ell} \tag{19}$$

where $\mathbf{W}_{\mathcal{P}}$ is an $3n \times 12$ matrix whose j th row is given by,

$$[(\bar{\varphi}_j)_{(1)}, (\bar{\varphi}_j)_{(2)}, (\bar{\varphi}_j)_{(3)}, \langle \nabla \varphi_j \rangle_{(11)}, \langle \nabla \varphi_j \rangle_{(12)}, \langle \nabla \varphi_j \rangle_{(13)}, \langle \nabla \varphi_j \rangle_{(21)}, \langle \nabla \varphi_j \rangle_{(22)}, \langle \nabla \varphi_j \rangle_{(23)}, \langle \nabla \varphi_j \rangle_{(31)}, \langle \nabla \varphi_j \rangle_{(32)}, \langle \nabla \varphi_j \rangle_{(33)}]. \tag{20}$$

Using the linear precision property of the canonical basis functions (8), we express \mathbf{p}_{ℓ} in (19) in terms of its discrete counterpart (bases sampled at the vertices) and upon simplification, we obtain:

$$\pi_{\mathcal{P}}\varphi_j = \sum_{k=1}^{3n} (\mathbf{N}_{\mathcal{P}} \mathbf{W}_{\mathcal{P}}^T)_{(kj)} \varphi_k. \tag{21}$$

Subsequent comparison with (18), we get $\mathbf{P}_{\mathcal{P}} = \mathbf{N}_{\mathcal{P}} \mathbf{W}_{\mathcal{P}}^T$, where $\mathbf{N}_{\mathcal{P}}, \mathbf{W}_{\mathcal{P}} \in \mathbb{R}^{3n \times 12}$. A block of three rows of $\mathbf{N}_{\mathcal{P}}$, corresponding to i th vertex, is explicitly expressed as:

$$(\mathbf{N}_{\mathcal{P}})_{(3i-2:3i,:)} = \begin{bmatrix} 1 & 0 & 0 & (\mathbf{x}_i - \bar{\mathbf{x}})_{(1)} & (\mathbf{x}_i - \bar{\mathbf{x}})_{(2)} & (\mathbf{x}_i - \bar{\mathbf{x}})_{(3)} \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ (\mathbf{x}_i - \bar{\mathbf{x}})_{(1)} & (\mathbf{x}_i - \bar{\mathbf{x}})_{(2)} & (\mathbf{x}_i - \bar{\mathbf{x}})_{(3)} & 0 & 0 & 0 \\ 0 & 0 & 0 & (\mathbf{x}_i - \bar{\mathbf{x}})_{(1)} & (\mathbf{x}_i - \bar{\mathbf{x}})_{(2)} & (\mathbf{x}_i - \bar{\mathbf{x}})_{(3)} \end{bmatrix} \tag{22}$$

and a block of three rows of $\mathbf{W}_{\mathcal{P}}$ is simplified as:

$$(\mathbf{W}_{\mathcal{P}})_{(3i-2:3i,:)} = \begin{bmatrix} 1/n & 0 & 0 & (\mathbf{q}_i)_{(1)} & (\mathbf{q}_i)_{(2)} & (\mathbf{q}_i)_{(3)} \\ 0 & 1/n & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/n & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ (\mathbf{q}_i)_{(1)} & (\mathbf{q}_i)_{(2)} & (\mathbf{q}_i)_{(3)} & 0 & 0 & 0 \\ 0 & 0 & 0 & (\mathbf{q}_i)_{(1)} & (\mathbf{q}_i)_{(2)} & (\mathbf{q}_i)_{(3)} \end{bmatrix} \tag{23}$$

where the surface integral vector \mathbf{q}_i is given by:

$$\mathbf{q}_i = \frac{1}{|e|} \int_{\partial e} \varphi_i \mathbf{n} ds. \tag{24}$$

To compute the surface integral, such as the one encountered in (24), we use first-order accurate nodal quadrature scheme which does not require the computation of canonical basis functions inside the elements. The approximation of the surface integration of a generic function g over a face F is defined as:

$$\int_F g ds \doteq \sum_{j \in F} w_j^F g(\mathbf{x}_j) \tag{25}$$

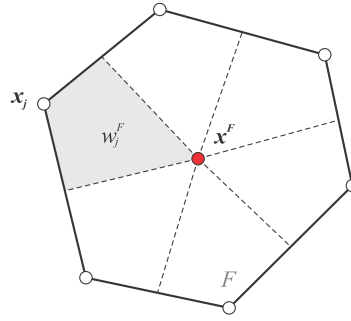


Fig. 3. Nodal quadrature scheme for surface integration. The variables \mathbf{x}^F and w_j^F represent the centroid of the polygon and the nodal weight associated with the vertex \mathbf{x}_j on any face F , respectively.

where $\{\mathbf{x}_j : j \in F\}$ represent the vertices on the face F . The nodal weight w_j^F for vertex \mathbf{x}_j is taken as the area of the quadrilateral formed by \mathbf{x}_j , the centroid of the face F , and mid-points of the edges incident on \mathbf{x}_j (Fig. 3).

The element stiffness matrix \mathbf{K}_e can now be obtained using the surface integral matrix $\mathbf{W}_{\mathcal{P}}$ and discrete projection $\mathbf{P}_{\mathcal{P}}$. From (16) we have,

$$(\mathbf{K}_e)_{(jk)} = a_h^e(\boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k) = a^e(\pi_{\mathcal{P}}\boldsymbol{\varphi}_j, \pi_{\mathcal{P}}\boldsymbol{\varphi}_k) + s^e(\boldsymbol{\varphi}_j - \pi_{\mathcal{P}}\boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k - \pi_{\mathcal{P}}\boldsymbol{\varphi}_k). \tag{26}$$

Using (19), the first term of \mathbf{K}_e can be simplified as:

$$a^e(\pi_{\mathcal{P}}\boldsymbol{\varphi}_j, \pi_{\mathcal{P}}\boldsymbol{\varphi}_k) = |e| \left(\mathbf{W}_{\mathcal{P}} \mathbf{D} \mathbf{W}_{\mathcal{P}}^T \right)_{(jk)} \tag{27}$$

where the matrix \mathbf{D} is a function of elasticity tensor \mathbf{C} and is defined as:

$$(\mathbf{D})_{(\ell m)} = \frac{1}{|e|} a^e(\mathbf{p}_{\ell}, \mathbf{p}_m) = \mathbf{C} \boldsymbol{\epsilon}(\mathbf{p}_{\ell}) : \boldsymbol{\epsilon}(\mathbf{p}_m), \quad \ell, m = 1, \dots, 12. \tag{28}$$

Now, adopting the inexpensive choice of s^e (17), the second term in the stiffness matrix is written as:

$$s^e(\boldsymbol{\varphi}_j - \pi_{\mathcal{P}}\boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k - \pi_{\mathcal{P}}\boldsymbol{\varphi}_k) = \left[(\mathbf{I} - \mathbf{P}_{\mathcal{P}})^T \mathbf{S}^e (\mathbf{I} - \mathbf{P}_{\mathcal{P}}) \right]_{(jk)} \tag{29}$$

where $(\mathbf{S}^e)_{(jk)} = s^e(\boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k)$ and corresponds to $\mathbf{S}^e = \alpha^e \mathbf{I}_{3n}$. We need to ensure that $s^e(\cdot, \cdot)$ is of the same order of magnitude as $a^e(\cdot, \cdot)$, so an appropriate value of α^e is

$$\alpha^e = \bar{\alpha}^e \text{trace}(|e| \mathbf{W}_{\mathcal{P}} \mathbf{D} \mathbf{W}_{\mathcal{P}}^T). \tag{30}$$

Based on the studies conducted in reference [38], $\bar{\alpha}^e$ is chosen as 0.05. Finally, the element stiffness matrix is given by

$$\mathbf{K}_e = |e| \mathbf{W}_{\mathcal{P}} \mathbf{D} \mathbf{W}_{\mathcal{P}}^T + \alpha^e (\mathbf{I} - \mathbf{P}_{\mathcal{P}})^T (\mathbf{I} - \mathbf{P}_{\mathcal{P}}). \tag{31}$$

To compute the force vectors corresponding to a body force, a first-order accurate scheme, similar to the one discussed in (25), can be used. Nodal quadrature for body forces involves computing nodal volume weights for volume integration. The weight for vertex \mathbf{x}_i is the volumes of the polyhedron formed by \mathbf{x}_i , the centroid of element e , the centroid of faces and the centroids of the edges incident on \mathbf{x}_i . In case of surface traction, (25) can be used. The global stiffness matrix and global force vector are obtain by standard assembly process. For a detailed study on the accuracy and convergence of VEM for three-dimensional elasticity refer to [30].

4. On centroidal Voronoi tessellation meshing

The design domain is discretized using centroidal Voronoi tessellation based method [39–41]. The algorithm uses the concept of signed distance function to define the boundaries, which are later approximated by collection of

intersecting planes. The boundary approximation is obtained by including a set of reflections of the seeds during the meshing process. In this section, we explain the meshing in detail.

Given a set of n_s distinct seeds $\mathcal{S} = \{s_i\}_{i=1}^{n_s}$, the Voronoi tessellation of the domain $\Omega \subseteq \mathbb{R}^3$ is defined as:

$$\mathcal{D}(\mathcal{S}) = \{e \cap \Omega : s_i \in \mathcal{S}\} \tag{32}$$

where e is the Voronoi cell corresponding to seed s_i :

$$e = \left\{ \mathbf{x} \in \mathbb{R}^3 : |\mathbf{x} - s_i| < |\mathbf{x} - s_j|, \quad \forall j \neq i \right\}. \tag{33}$$

The above definition of e represents a domain consisting of all points that are closer to seed s_i than any other seed $s_j \in \mathcal{S}$. Note that the Voronoi cells are necessarily convex polyhedrons since they are formed by the finite intersection of convex half-planes.

Following the guidelines discussed in [40,41], a polyhedral discretization is obtained from the Voronoi diagram of a given set of seeds and their reflections about the closest boundary of Ω , i.e. Γ . As indicated earlier, our meshing algorithm is implemented for general domains using signed distance functions. A signed distance function $d_\Omega(\mathbf{x})$ is defined as:

$$d_\Omega(\mathbf{x}) = z(\mathbf{x}) \min(|\mathbf{x} - \mathbf{y}|), \quad \forall \mathbf{y} \in \Gamma \tag{34}$$

where $z(\mathbf{x})$ is the sign function defined as:

$$z(\mathbf{x}) = \begin{cases} -1, & \mathbf{x} \in \Omega, \\ +1, & \mathbf{x} \in \mathbb{R}^3 \setminus \Omega. \end{cases} \tag{35}$$

Thus, $d_\Omega(\mathbf{x}) = 0$ if $\mathbf{x} \in \Gamma$ and $d_\Omega(\mathbf{x}) < 0$ if $\mathbf{x} \in \Omega \setminus \Gamma$. Using the signed distance function and its gradient, the reflection, s_i^r , of the seed s_i can be calculated as:

$$s_i^r = s_i^r - 2d_\Omega(s_i)\nabla d_\Omega(s_i). \tag{36}$$

First, to construct a polyhedron discretization of the domain Ω , each point in \mathcal{S} is reflected about the closest boundary Γ . The resulting set of points are denoted by $\mathcal{R}_\Omega(\mathcal{S})$. Subsequently, we construct the Voronoi diagram of the space using the original point set and its reflection, $\mathcal{T}(\mathcal{S} \cup \mathcal{R}_\Omega(\mathcal{S}); \mathbb{R}^3)$. Thus, the discretization of Ω is the collection of the Voronoi cells associated with seeds \mathcal{S} . For a given point set \mathcal{S} , the discretization of the domain Ω is uniquely defined and denoted by:

$$\mathcal{M}_\Omega(\mathcal{S}) = \left\{ e \in \mathcal{T}(\mathcal{S} \cup \mathcal{R}_\Omega(\mathcal{S}); \mathbb{R}^3) : s_i \in \mathcal{S} \right\}. \tag{37}$$

If the Voronoi cell of a seed s_i and its reflection have a common edge, then this edge forms an approximation to the domain boundary and a reasonable discretization of Ω is obtained. In order to mesh complicated geometries, a signed distance function along with set operations such as union, difference, and intersection are used [42,41]. Note that for a convex domain Ω , $\mathcal{M}_\Omega(\mathcal{S})$ circumscribes Ω .

In our meshing algorithm, a set of signed distance functions corresponding to basic geometric shapes such as three-dimensional plane, sphere, cylinder and rectangular box is defined. We also construct a bounding box B , that contains the domain Ω , to generate the random seeds in \mathbb{R}^3 . A random seed is accepted only if it lies inside the domain Ω , determined by evaluating the sign of the resultant distance function, d_Ω , associated with Ω . Algorithm 1 shows the basic steps for obtaining a random point set of size n_s .

We handle *convex* and *non-convex* features of Ω by carefully choosing a set of seeds to be reflected w.r.t. the boundary. Reflection of a seed far from the boundary may land inside the domain or interfere with the reflection of another seed. Since the reflection of most of the seeds in the interior of the domain has no effect on the approximation of the boundary, we reflect only the seeds that are in a band near the boundary. A seed $s_i \in \mathcal{S}$ is reflected about boundary segment Γ provide that:

$$|d_\Omega(s_i)| < c \left(\frac{|\Omega|}{n_s} \right)^{1/3} \tag{38}$$

Algorithm 1 Initial random seed placement

```

input:  $B, n_s$       %%  $B \supset \Omega \in \mathbb{R}^3$  and  $n_s$  is the number of seeds
  set  $\mathcal{S} = \emptyset$ 
  while  $|\mathcal{S}| < n_s$  do
    generate random point  $\mathbf{y} \in B$ 
    if  $d_\Omega(\mathbf{y}) < 0$  then
       $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathbf{y}\}$ 
    end if
  end while
output:  $\mathcal{S}$ 

```

where c is the proportionality constant, chosen to be greater than 1 to make the band size near the boundary larger than the average element volume.

Clearly, Voronoi meshes generated from random/quasi-random seeds may cause inconsistencies at the boundaries resulting in a poor approximation of the boundaries of the design domain. To introduce some regularity in the Voronoi meshes, we construct Centroidal Voronoi tessellations (CVT) using a modified Lloyd's algorithm [43]. For a large number of iterations, CVT cells tend to be uniform in size [44]. To generate CVT meshes, we replace seeds \mathcal{S} with centroids \mathcal{S}_c of the Voronoi cells. We compute the polyhedron centroid by partitioning it into tetrahedrons and evaluating the weighted mean of the centroids of the resulting tetrahedrons. The weights are the volumes of the tetrahedrons.

To construct the element stiffness matrix using the approach discussed in Section 3, along with vertices locations and element connectivity information, we need to know the vertices incident on each face of polyhedrons. The pseudo-code listed in Algorithm 2 summarizes our approach to obtain facial information in MATLAB. The inputs to the algorithm are the element connectivity and the coordinates of the vertices and seeds. For each seed of the mesh, the algorithm computes the convex hull of the set \mathbf{V} (vertices in an element). The convex hull of \mathbf{V} is a matrix \mathbf{H} with as many rows as the number of triangular subdivision of faces present in the convex hull, and three columns containing the indexes of the vertices of the corresponding triangular subdivisions. Then, we iterate over all triangles and unite those that are co-planar to obtain the polygonal faces of the polyhedron. The resulting array, called *Elm*, contains the faces and vertices of each polyhedral element. Fig. 4 shows some of the sample meshes and their statistical information, obtained from the current algorithm.

Algorithm 2 Construction of final mesh consisting of vertices, elements and faces in MATLAB.

```

input:  $e, N, \mathcal{S}, n_s$       %% Voronoi cells  $e$ , vertices coords  $N$ , seeds coords  $\mathcal{S}$ 
   $Elm \leftarrow 0$           %% initialize an array with size of  $n_s$ 
  for  $i = 1$  to  $n_s$  do
    let  $\mathbf{V} = N(e(i))$ 
    construct convex hull  $\mathbf{H} \leftarrow \mathcal{H}(\mathbf{V}; \mathbb{R}^3)$ 
     $m \leftarrow |\mathbf{H}|$       %% number of triangular subdivisions  $m$  obtained from convex hull
     $ElementFaces \leftarrow 0$ 
    for  $j = 1$  to  $m$  do
      let  $\mathbf{T} = e(\mathbf{H}(j))$       %% vertices of a triangle
       $\mathbf{T} = OrderVertices(N, \mathbf{T}, \mathcal{S}(i))$ 
       $ElementFaces \leftarrow \mathbf{T}$   %% create/unite faces
    end for
     $Elm(i) \leftarrow ElementFaces$ 
  end for
output:  $Elm$ 

```

Note that even with CVT approach, the meshing process generates a few elements with large aspect ratios or elements with a few small faces/edges in the interior as well as on the boundary. As shown by Gain et al. [30], the VEM approach for three-dimensional linear elasticity is fairly robust with respect to the random element geometries

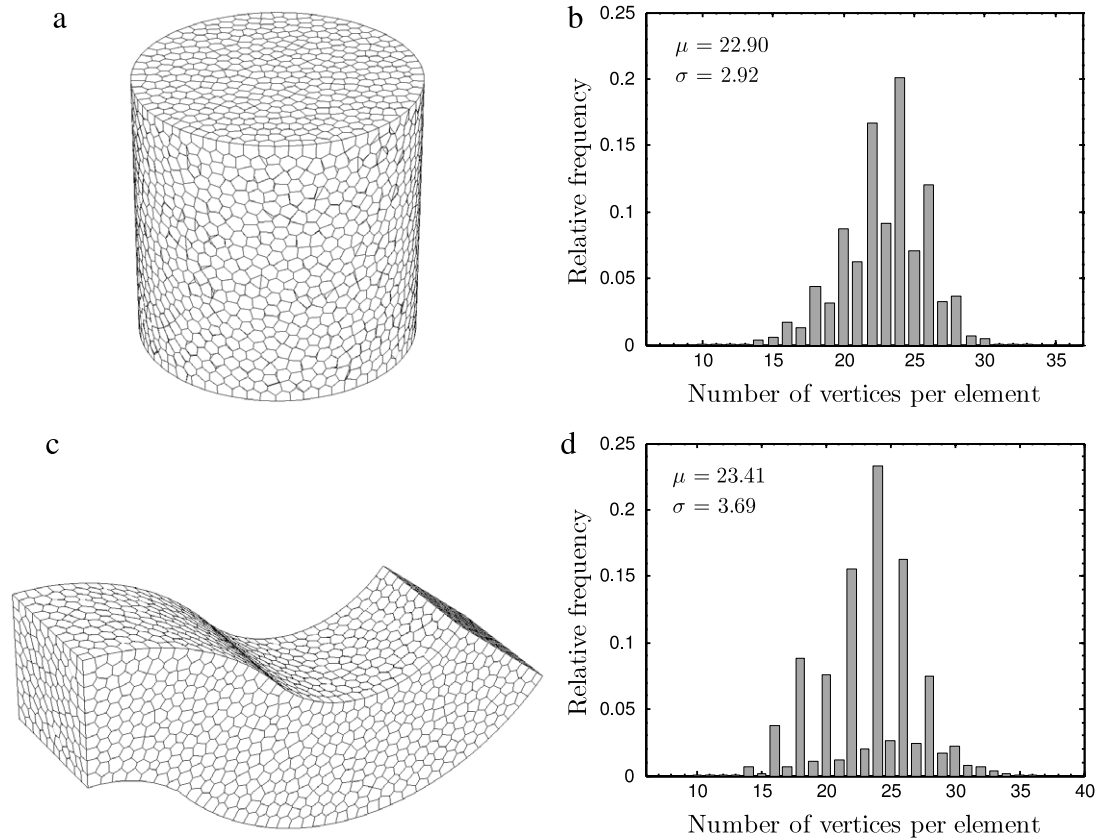


Fig. 4. Sample meshes. (a) Cylinder. (b) Cylinder mesh statistics. On average, elements in the mesh have approximately 23 vertices with a standard deviation of 2.92. (c) Curved cantilever beam. (d) Curved cantilever mesh statistics. On average, the elements in the mesh also have approximately 23 vertices with a standard deviation of 3.69.

as indicated by the optimal rates of convergence for the relative displacement and stress errors. Hence, we did not incorporate any special schemes to remove or modify irregular elements or element with undesirable attributes (small faces/edges) in our meshing algorithm.

5. Numerical examples

In this section, numerical examples are shown to demonstrate the effectiveness of the current approach. We start with the benchmark cantilever beam problem solved on a box domain using different mesh discretizations, followed by problems on non-Cartesian design domains. For our simulations, Young's modulus, E , and Poisson's ratio, ν , are taken as 10,000 and 0.3, respectively. As mentioned earlier, Optimality Criteria (OC) is the optimizer of choice. The optimization is terminated when either the maximum of the change in element densities is less than 0.01 or the number of iterations exceed 300. Similar to traditional finite elements, polyhedral elements based topology optimization results too suffer from mesh-dependency (i.e. refinement of mesh results in solution with more intricate design features). Application of filters is one of the remedies to the problem of non-existence of solutions. Filters on densities and sensitivities have been used widely in the literature. In this work we use linear density filters (discussed in briefly in [Appendix B](#)).

5.1. Cantilever beam problem on box domain

We begin with the benchmark cantilever beam problem for a design domain of dimension $2 \times 1 \times 1$. The left face of the box is fixed and a point load is applied in the middle of the right face (refer [Fig. 5](#)). The problem is solved on both hexahedral and polyhedral element meshes. Taking advantage of symmetry, only half of the domain is optimized,

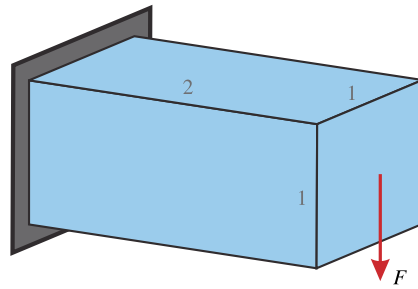


Fig. 5. Cantilever beam problem.

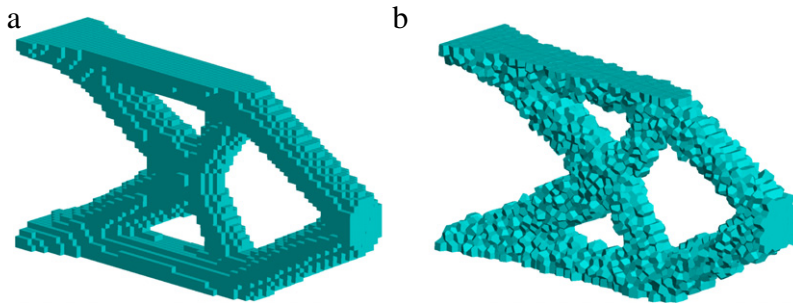


Fig. 6. Converged topologies for the cantilever beam problem using the present method. (a) Hexahedral mesh of 54,872 elements, 60,060 nodes ($J = 0.1098$). (b) Polyhedral mesh of 10,000 elements, 58,601 nodes. The average number of vertices per polyhedron is, $\mu = 22.85$, with standard deviation, $\sigma = 3.80$, ($J = 0.1082$).

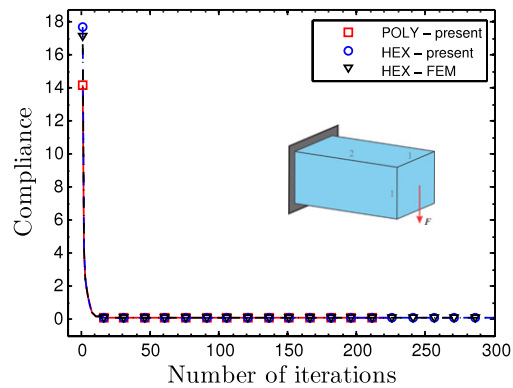


Fig. 7. Convergence history for the cantilever beam problem.

which is discretized using 54,872 hexahedrons (60,060 nodes) and 10,000 polyhedrons (58,601 nodes). A linear filter of radius 5% of the maximum domain dimension is used and a volume fraction of 0.1 is prescribed.

With the present approach, both mesh discretizations produce similar optimization results (Fig. 6). Note that the optimized topologies shown in Fig. 6 (also all subsequent results) show only the elements whose density exceeds 0.5. The final compliance values are 0.1098 and 0.1082 for the topologies on hexahedral and polyhedral meshes, respectively.

For comparison, the same problem is solved using the finite element method on a hexahedral mesh. Topology similar to the present method is obtained. The convergence history for all three cases are illustrated in Fig. 7. As expected, a smooth monotonic convergence is obtained for all three cases and they all converge to similar final compliance values.

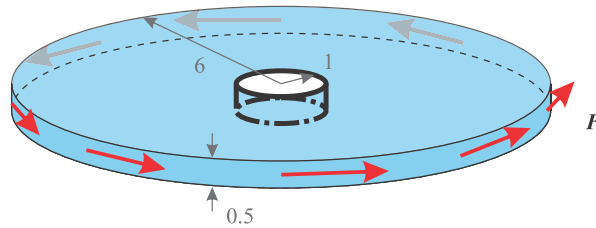


Fig. 8. Problem description for thin disc. Eight equidistant shear loads are applied along the circumference of the disc and all the nodes along the central cylindrical hole are fixed.

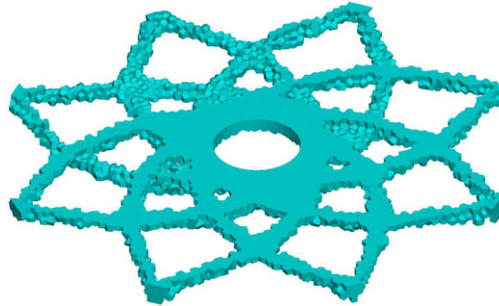


Fig. 9. Final topology for shear loaded thin disc on a 10,000 element, 55,810 nodes polyhedral mesh. The average number of vertices per element is, $\mu = 20.53$, with standard deviation of, $\sigma = 3.71$, ($J = 0.5850$).

5.2. Shear loaded thin disc

We next investigate the shear loaded thin disc problem. The thin disc domain has an external radius of 6 units with an internal cylindrical hole of radius 1 unit and has a thickness of 0.5 units (Fig. 8). Eight equidistant shear loads are applied along the circumference of the disc and all the nodes along the cylindrical hole are fixed. A polyhedral mesh of 10,000 elements (55,810 nodes) is used to discretize the design domain. A filter radius of 4% of the outer diameter is selected and a volume fraction of 0.2 is enforced.

According to Michell layout theory [2,3], an optimum structural layout is one in which the tension and compression members meet orthogonally. Such a set of orthogonal curves are known as Hencky nets [3]. The tension and compression members in topology optimization solutions should adhere to this principle. The converged topology for the shear loaded thin disc problem, obtained from our algorithm, is shown in Fig. 9 (resembles a flower). The members of the structure intersect nearly at right angles as expected, even for a coarse polyhedral mesh. The compliance of the final topology is 0.5850.

5.3. Hollow cylinder under torsional load

We now look at the torsionally loaded hollow cylinder. The design domain is in the shape of a hollow cylinder of thickness 0.1 units, height 4 units and outer diameter of 1 unit (c.f. Fig. 10). Four equidistant nodes along the bottom face are fixed and four tangential point loads are applied to corresponding nodes on the top face, effectively acting as a torsional load. A filter radius of 3% of the height of the cylinder and a volume fraction of 0.3 are prescribed. The problem is solved on three sets of meshes — two tetrahedral meshes of 9977 elements (3349 nodes); 451,584 elements (85,320 nodes) and a polyhedral mesh of 10,000 elements (79,925 nodes).

Using the polyhedral mesh, our optimization algorithm yields an elegant spiraling structure with nearly orthogonally oriented crossing members (Fig. 10(c)). For a tetrahedral mesh, with a similar number of elements as the polyhedral mesh (9977), we still obtain a similar spiraling structure (c.f. Fig. 10(b)), but the members orientation is affected by the mesh geometry and the intersecting members are not fully orthogonal. This might be because although the tetrahedral mesh has similar number of elements as polyhedral mesh, the degrees of freedom in tetrahedral mesh

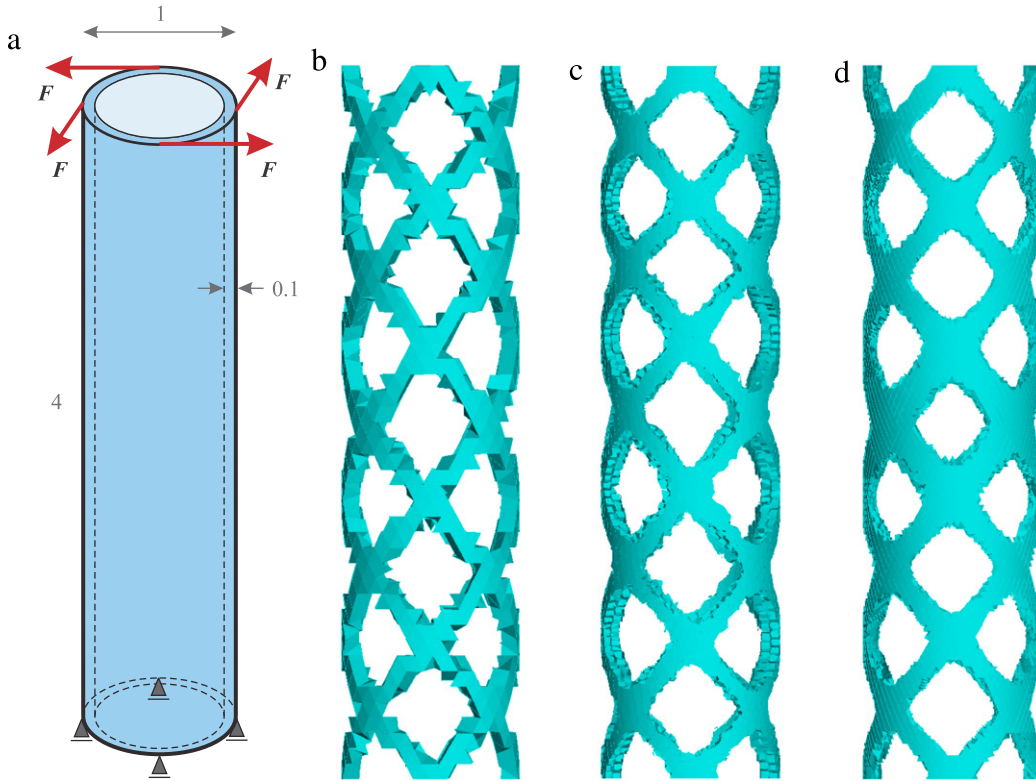


Fig. 10. Hollow cylinder under torsional load. (a) Problem description. Converged topologies for (b) Linear tetrahedral mesh of 9977 elements, 3349 nodes ($J = 1.1397$); (c) Polyhedral mesh of 10,000 elements, 79,925 nodes. On average, polyhedral elements have, $\mu = 22.57$, vertices with standard derivation of, $\sigma = 2.88$, ($J = 1.6005$); (d) Linear tetrahedral mesh of 451,584 elements, 85,320 nodes ($J = 1.2064$).

is far less than the polyhedral mesh. So we solved the hollow cylinder problem on a fine tetrahedral mesh, such that the number of nodes in the mesh are comparable to that of the polyhedral mesh. Note that, in the fine tetrahedral mesh, the number of elements has risen to 451,584. Although the fine tetrahedral mesh (Fig. 10(d)) achieves the desired member orthogonality in the optimization result, the fine mesh considerably increases the computational cost associated with operations, such as creation and storage of the filter matrix, compared to the coarse tetrahedral and polyhedral mesh. If other length scale control approaches are adopted, for example enforcing perimeter constraint, in place of filters, the computational cost associated with fine tetrahedral meshes can be considerably reduced. But the downside to this approach on tetrahedral meshes is that single node connections may arise in the designs. In terms of the cost associated with solving the governing elasticity problem, polyhedral meshes are marginally more expensive than tetrahedral meshes of comparable total degrees of freedom. This is due to that fact that polyhedrons on an average have higher number of vertices (approximately 23 for our meshes) than tetrahedrons. Finally, note that the lower compliance for the optimization result on a coarse tetrahedral mesh (Fig. 10(b)) can be attributed to the fact that tetrahedral meshes experience artificial stiffness due to shear locking phenomenon, which reduces with mesh refinement.

We would like to point out that the element stiffness matrix obtained using the linear tetrahedral finite element approach is identical to the one obtained using the current approach. The reason being that, in the present method, for elements in the shape of a tetrahedron, the contribution of the stability term is zero, because the finite dimensional space of smooth functions on element, \mathcal{W} , is identical to the space of linear deformations, \mathcal{P} . So, the only contribution to the element stiffness matrix comes from the consistency term which is the same as the one obtained from finite element analysis. Thus, the final topologies obtained from the current approach and the linear tetrahedral finite element approach should be identical, along with the convergence history. Our results are in agreement with the above statement.

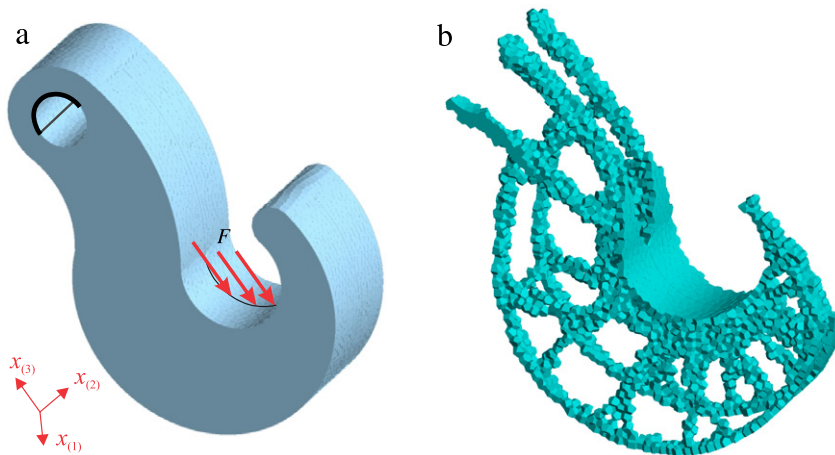


Fig. 11. Hook under line load. (a) Problem description. All the nodes along the top half of the upper cylindrical hole are fixed and line load is applied along the circular arc in the negative $x_{(3)}$ -direction, as indicated in the figure. The domain is discretized using 10,000 polyhedral elements containing 67,893 nodes. The average number of vertices per polyhedron is, $\mu = 23.97$, with standard deviation, $\sigma = 4.19$. (b) Converged topology ($J = 7.0484$).

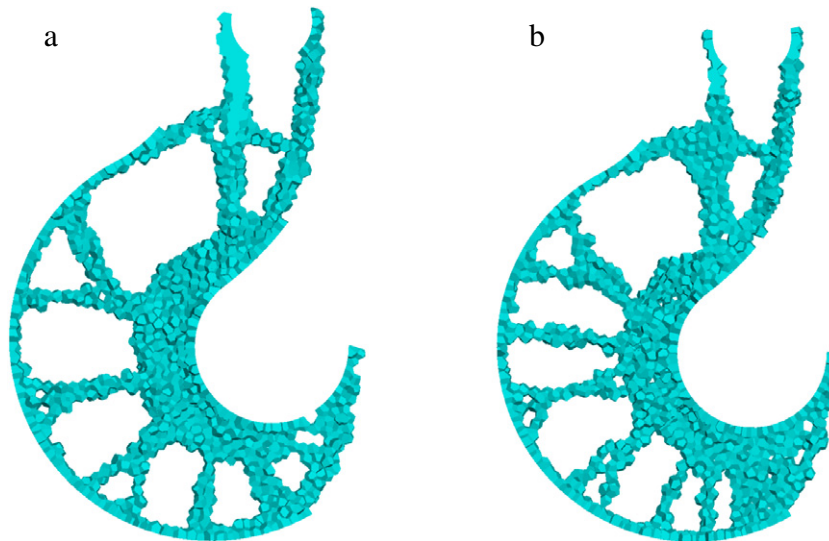


Fig. 12. Study of the influence of filters on polyhedral topology optimization for hook problem of Fig. 11. (a) With filter (front view of Fig. 11(b)). (b) Without filter.

5.4. Hook domain under line load

For the final compliance minimization problem, we investigate the hook domain subjected to a uniformly distributed line load along the negative $x_{(3)}$ -direction (Fig. 11(a)). A volume fraction of 0.1 is prescribed and a linear filter with radius equal to 2% of the maximum domain dimension is used. Using symmetry, we optimize only half the hook domain. The polyhedral mesh contains 10,000 elements (67,893 nodes). The converged topology, Fig. 11(b), has a compliance of 7.0484 and resembles the structure of a fan. The two dimensional version of the problem [45] has similar member orientations as our current three-dimensional result.

Next, we study the effect of filters on the optimization results for polyhedral meshes. Keeping all the parameters the same as before, we solve the hook problem without using any filter. As expected, without any length-scale control, more structural members, including some thin ones, appear in the solution (Fig. 12(b)). It is interesting to note that even without filtering no single node connections were observed in the design. Even though this is a favorable feature

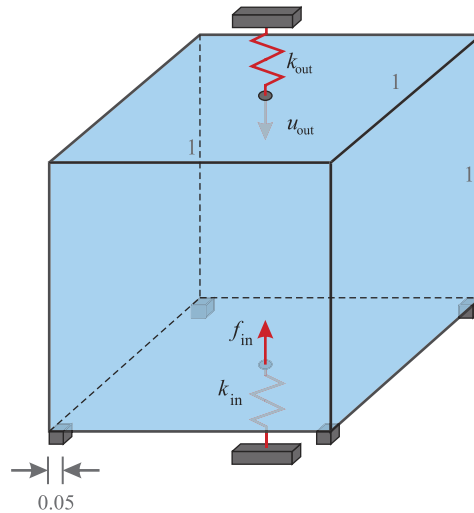


Fig. 13. Displacement inverter.

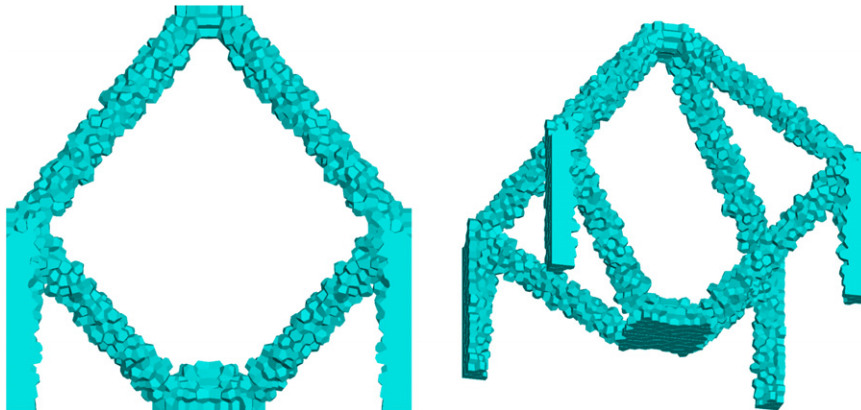


Fig. 14. Converged topology for the displacement inverter problem. Due to symmetry, only a quarter of the domain is optimized and is discretized using 10,000 polyhedral elements containing 58,785 nodes. The average number of vertices per polyhedron is, $\mu = 22.98$, with standard deviation of, $\sigma = 3.75$.

to have, from a practical point of view one needs to incorporate approaches to restrict the variation of the continuous density function such as perimeter constraint [46], gradient/slope constrained methods [47] to avoid thin members.

5.5. Displacement inverter

Apart from compliance minimization, we also investigate compliant mechanism problems [48]. First, we explore the displacement inverter problem (Fig. 13). The domain is of dimension $1 \times 1 \times 1$ and is fixed at the bottom four corners. The objective of optimization is to maximize the output displacement u_{out} on a workpiece modeled by a spring of stiffness k_{out} . The input and output spring stiffnesses, k_{in} and k_{out} , are taken to be the same as the components of the global stiffness matrix at the degrees of freedom corresponding to the input and output nodes. Taking advantage of symmetry, only a quarter of the domain is optimized and is discretized using 10,000 polyhedral elements (58,785 nodes). A volume fraction of 0.05 is prescribed. Our final topology (Fig. 14) is similar to the ones available in the literature [49]. Visually, the three-dimensional optimization result (Fig. 14) resembles to the two-dimensional optimization result [34].

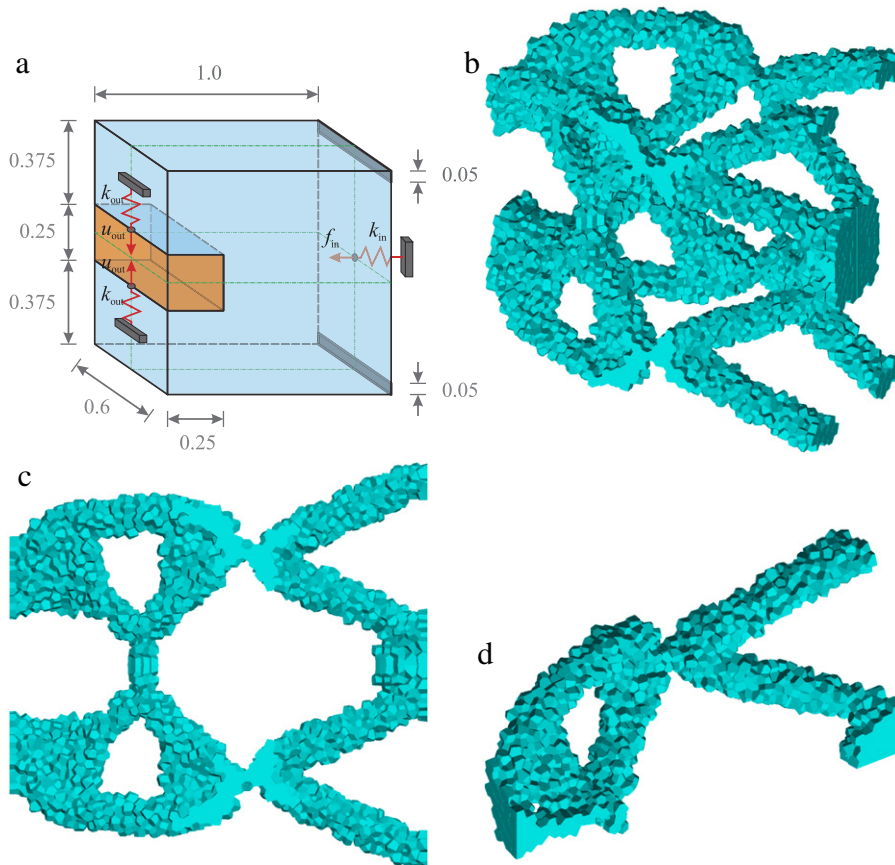


Fig. 15. Topology optimization design of gripper. A quarter of the problem is solved with a polyhedral mesh of 10,000 elements and 59,194 nodes. The average number of vertices per polyhedron is, $\mu = 23.11$, with standard deviation, $\sigma = 3.85$. Orange region is modeled as voids using passive element concept. (a) Problem description. (b) Complete design. (c) Front view. (d) Quarter section view.

5.6. Gripper

A gripper, as the name suggests, is a compliant mechanism suitable for gripping objects. The input actuator, modeled as a spring with stiffness k_{in} , and a force f_{in} , applies a horizontal load as indicated in Fig. 15 and the goal of optimization is to maximize the vertical output displacement u_{out} on a workpiece modeled by a spring of stiffness k_{out} . The problem dimensions and boundary conditions are indicated in Fig. 15. Nodes on the top and bottom section on the right face, indicated by gray color, are fixed. Using the passive element concept the orange box is modeled as void. Before each optimization cycle, the elements lying inside the orange box are identified and are assigned the minimum density corresponding to voids. The spring stiffnesses, k_{in} and k_{out} are taken to be the same as the components of the global stiffness matrix at the degrees of freedom corresponding to the input and output nodes. Due to symmetry, only a quarter of the domain is optimized and is discretized using 10,000 polyhedral elements (58,785 nodes) with a prescribed volume fraction of 0.10. The converged topology (Fig. 15) for three-dimensional optimization is a clear extension of the two-dimensional results available in reference [34]. The combination of gripping jaws and hinge mimicking narrow necks, along the middle of the design, resemble a pair of scissors. The convergence history for the gripper problem, shown in Fig. 16, indicates stable convergence.

6. Concluding remarks

This work explores unstructured polyhedral meshes for three-dimensional topology optimization. The governing elasticity problem in the optimization framework is solved using the Virtual Element Method. VEM addresses some of the challenges facing finite element approaches, such as prohibitively expensive polyhedral shape functions

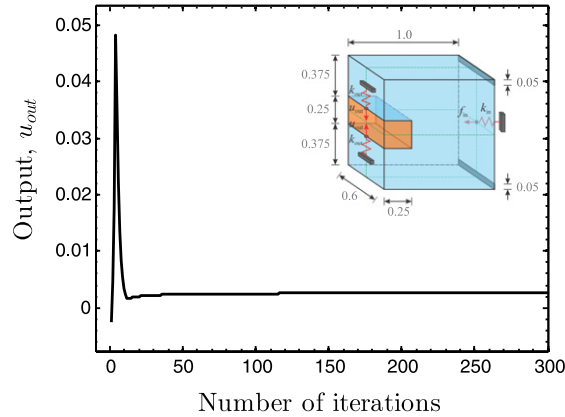


Fig. 16. Convergence history for the gripper problem.

and their derivatives computations, accurate numerical evaluation of the weak form and sensitivity to degenerate elements. Unlike other Galerkin approaches, VEM does not require explicit computation of the shape functions. Rather the stiffness matrix is constructed directly using a projection map which is defined to extract the constant strain deformation modes, thereby guaranteeing the passing of patch test and resulting in optimum convergence under mesh refinement. Here, we investigate the topology optimization of compliance minimization and compliant mechanism problems using centroidal Voronoi tessellation (CVT) based polyhedrons. The natural geometry of the polyhedral elements makes them attractive for topology optimization. We are currently exploring large scale topology optimization problems for complicated domains which may be conveniently handled using the multiresolution implementation² of the current approach [38,50].

Appendix A. Sensitivity analysis

The discretized version of (6) can be expressed as:

$$\begin{aligned} \inf_{\rho} J &= \mathbf{P}^T \mathbf{U} \\ \text{subject to: } \mathbf{K}(\rho)\mathbf{U} &= \mathbf{F}, \quad \sum_e |e| \rho_e \leq V_f |\Omega| \end{aligned} \quad (39)$$

where \mathbf{K} , \mathbf{U} and \mathbf{F} are the global stiffness matrix, global nodal displacement vector and global nodal force vector, respectively. The vector \mathbf{P} represents the global force vector \mathbf{F} for compliance minimization problem because the objective is to minimize external work. For displacement inverter and gripper problem, \mathbf{P} is a vector with all zeros except at locations corresponding to the output degree of freedom, where it is unity.

We use gradient-based optimization algorithm (OC) for solving the discrete problem (39), which requires computation of the gradient of the objective function J . Using the adjoint method [34], the sensitivity of J with respect to the design variable (element density, ρ_e), is given by:

$$\frac{\partial J}{\partial \rho_e} = -p(1 - \epsilon) \rho_e^{p-1} \boldsymbol{\lambda}_e^T \mathbf{K}_e \mathbf{u}_e. \quad (40)$$

For the compliance minimization problem (3), the vector $\boldsymbol{\lambda}_e$ is the same as the element displacement vector \mathbf{u}_e . For the displacement inverter problem (5), $\boldsymbol{\lambda}_e$ is the elemental component of $\boldsymbol{\lambda}$ that solves the adjoint system $\mathbf{K}\boldsymbol{\lambda} = \mathbf{P}$. Also, the sensitivity of the volume constraint (shown in (39)) with respect to element density is $|e|$.

Appendix B. Projection method — Filtering

In topology optimization, filters are used to enforce a length-scale in the problem and to ensure mesh-independency [51,34]. We use a linear filter which assigns a weighted average of the nearby elemental densities

² In the multiresolution approach, the elasticity and optimization problem are solved on a coarse and fine mesh, respectively.

to each element. Thus, the projected element density of an element, ρ_e , is written as:

$$\rho_e = \frac{\sum_{j \in N_e} w_{ej} \rho_j}{\sum_{j \in N_e} w_{ej}}. \quad (41)$$

Here, N_e is the set of all elements whose centers lie within a distance of r_{min} from the center of the element e under consideration. The linear weights w_{ej} are:

$$w_{ej} = \max\left(0, \frac{r_{min} - r_{ej}}{r_{min}}\right) \quad (42)$$

where r_{min} and r_{ej} are the enforced minimum member size and the distance between centroids of elements e and j , respectively.

To compute the sensitivity of the objective function J with respect to the independent design variables ρ_j , we use the chain rule:

$$\frac{\partial J}{\partial \rho_j} = \frac{\partial J}{\partial \rho_e} \frac{\partial \rho_e}{\partial \rho_j}, \quad \text{where} \quad \frac{\partial \rho_e}{\partial \rho_j} = \frac{w_{ej}}{\sum_{j \in N_e} w_{ej}}. \quad (43)$$

Other filters, such as sensitivity filters may also be used.

References

- [1] P. Dvorak, New element lops time off CFD simulations, *Mach. Des.* 78 (5) (2006) 154–155.
- [2] A.G.M. Michell, The limits of economy of material in frame-structures, *Phil. Mag.* 8 (47) (1904) 589–597.
- [3] W.S. Hemp, *Optimum Structures*, Clarendon Press, Oxford, 1973.
- [4] E.L. Wachspress, *A Rational Finite Element Basis*, Academic Press, New York, 1975.
- [5] J. Warren, Barycentric coordinates for convex polytopes, *Adv. Comput. Math.* 6 (1) (1996) 97–108.
- [6] R. Sibson, A vector identity for the Dirichlet tessellation, *Math. Proc. Camb. Phil. Soc.* 87 (1980) 151–155.
- [7] N. Sukumar, B. Moran, A.Y. Semenov, V.V. Belikov, Natural neighbor Galerkin methods, *Internat. J. Numer. Methods Engrg.* 50 (2001) 1–27.
- [8] V.V. Belikov, V.D. Ivanov, V.K. Kontorovich, S.A. Korytnik, A. Yu. Semenov, The non-Sibsonian interpolation: A new method of interpolation of the values of a function on an arbitrary set of points, *Comput. Math. Math. Phys.* 37 (1) (1997) 9–15.
- [9] H. Hiyoshi, K. Sugihara, Two generalizations of an interpolant based on Voronoi diagrams, *Int. J. Shape Model.* 5 (2) (1999) 219–231.
- [10] N.H. Christ, R. Friedberg, T.D. Lee, Weights of links and plaquettes in a random lattice, *Nuclear Phys. B* 210 (3) (1982) 337–346.
- [11] N. Sukumar, A. Tabarraei, Conforming polygonal finite elements, *Internat. J. Numer. Methods Engrg.* 61 (12) (2004) 2045–2066.
- [12] M.S. Floater, Mean value coordinates, *Comput. Aided Geom. Design* 20 (1) (2003) 19–27.
- [13] M.S. Floater, G. Kös, M. Reimers, Mean value coordinates in 3D, *Comput. Aided Geom. Design* 22 (7) (2005) 623–631.
- [14] E.A. Malsch, J.J. Lin, G. Dasgupta, Smooth two dimensional interpolants: a recipe for all polygons, *J. Graph. Tools* 10 (2) (2005).
- [15] N. Sukumar, Construction of polygonal interpolants: a maximum entropy approach, *Internat. J. Numer. Methods Engrg.* 61 (12) (2004) 2159–2181.
- [16] M. Arroyo, M. Ortiz, Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods, *Internat. J. Numer. Methods Engrg.* 65 (13) (2006) 2167–2202.
- [17] K. Hormann, N. Sukumar, Maximum entropy coordinates for arbitrary polytopes, in: *Eurographics Symposium on Geometry Processing*, vol. 27, 2008, pp. 1513–1520.
- [18] N. Sukumar, E.A. Malsch, Recent advances in the construction of polygonal finite element interpolations, *Arch. Comput. Methods Eng.* 13 (1) (2006) 129–163.
- [19] E.L. Wachspress, Rational bases for convex polyhedra, *Comput. Math. Appl.* 59 (6) (2010) 1953–1956.
- [20] P. Joshi, M. Meyer, T. Derose, B. Green, T. Sanocki, Harmonic coordinates for character articulation, *ACM Trans. Graphics* 26 (3) (2007) Art. no. 1276466.
- [21] S. Martin, P. Kaufmann, M. Botsch, M. Wicke, M. Gross, Polyhedral finite elements using harmonic basis functions, *Comput. Graph. Forum* 27 (5) (2008) 1521–1529.
- [22] J. Bishop, A displacement-based finite element formulation for general polyhedra using harmonic shape functions, *Internat. J. Numer. Methods Engrg.* 97 (1) (2014) 1–31.
- [23] R. Saxena, A. Saxena, On honeycomb representation and SIGMOID material assignment in optimal topology synthesis of compliant mechanisms, *Finite Elements in Analysis and Design* 43 (14) (2007) 1082–1098.
- [24] C. Talischi, G.H. Paulino, A. Pereira, I.F.M. Menezes, Polygonal finite elements for topology optimization: A unifying paradigm, *Internat. J. Numer. Methods Engrg.* 82 (2010) 671–698.
- [25] A.L. Gain, G.H. Paulino, Phase-field based topology optimization with polygonal elements: A finite volume approach for the evolution equation, *Struct. Multidiscip. Optim.* 46 (3) (2012) 327–342.

- [26] F. Brezzi, K. Lipnikov, M. Shashkov, V. Simoncini, A new discretization methodology for diffusion problems on generalized polyhedral meshes, *Comput. Methods Appl. Mech. Engrg.* 196 (37–40 SPEC. ISS.) (2007) 3682–3692.
- [27] L. Beirão Da Veiga, K. Lipnikov, G. Manzini, Error analysis for a mimetic discretization of the steady stokes problem on polyhedral meshes, *SIAM J. Numer. Anal.* 48 (4) (2010) 1419–1443.
- [28] L. Beirão Da Veiga, A mimetic discretization method for linear elasticity, *ESAIM Math. Model. Numer. Anal.* 44 (2) (2010) 231–250.
- [29] L. Beirão Da Veiga, F. Brezzi, L.D. Marini, Virtual elements for linear elasticity problems, *SIAM J. Numer. Anal.* 51 (2) (2013) 794–812.
- [30] A.L. Gain, C. Talischi, G.H. Paulino, On the Virtual Element Method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes, *Comput. Methods Appl. Mech. Engrg.* 282 (2014) 132–160.
- [31] M.P. Bendsøe, Optimal shape design as a material distribution problem, *Struct. Optim.* 1 (1989) 193–202.
- [32] M. Zhou, G.I.N. Rozvany, The COC algorithm, Part II: Topological, geometrical and generalized shape optimization, *Comput. Methods Appl. Mech. Engrg.* 89 (1–3) (1991) 309–336.
- [33] H.P. Mlejnek, R. Schirrmacher, An engineer’s approach to optimal material distribution and shape finding, *Comput. Methods Appl. Mech. Engrg.* 106 (1–2) (1993) 1–26.
- [34] M.P. Bendsøe, O. Sigmund, *Topology Optimization — Theory, Methods and Applications*, Springer, New York, 2003.
- [35] K. Svanberg, The method of moving asymptotes — a new method for structural optimization, *Internat. J. Numer. Methods Engrg.* 24 (2) (1987) 359–373.
- [36] C. Fleury, V. Braibant, Structural optimization: a new dual method using mixed variables, *Internat. J. Numer. Methods Engrg.* 23 (3) (1986) 409–428.
- [37] L. Beirão Da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L.D. Marini, A. Russo, Basic principles of virtual element methods, *Math. Models Methods Appl. Sci.* 23 (1) (2013) 199–214.
- [38] A.L. Gain, *Polytope-based topology optimization using a mimetic-inspired method* (Ph.D. thesis), University of Illinois, Urbana-Champaign, 2013.
- [39] F. Aurenhammer, Voronoi diagrams — a survey of a fundamental geometric data structure, *ACM Comput. Surv.* 23 (3) (1991) 345–405.
- [40] M. Yip, J. Mohle, J.E. Bolander, Automated modeling of three-dimensional structural components using irregular lattices, *Comput. Aided Civil Infrastruct. Eng.* 20 (6) (2005) 393–407.
- [41] C. Talischi, G.H. Paulino, A. Pereira, I.F.M. Menezes, PolyMesher: A general-purpose mesh generator for polygonal elements written in Matlab, *Struct. Multidiscip. Optim.* 45 (3) (2012) 309–328.
- [42] P.-O. Persson, G. Strang, A simple mesh generator in MATLAB, *SIAM Rev.* 46 (2) (2004) 329–345.
- [43] S. Lloyd, Least squares quantization in PCM, *IEEE Trans. Inform. Theory* 28 (2) (1982) 129–137.
- [44] Q. Du, D. Wang, The optimal centroidal Voronoi tessellations and the gershgorin’s conjecture in the three-dimensional space, *Comput. Math. Appl.* 49 (9–10) (2005) 1355–1373.
- [45] C. Talischi, G.H. Paulino, A. Pereira, I.F.M. Menezes, PolyTop: a Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes, *J. Struct. Multidiscip. Optim.* 45 (3) (2012) 329–357.
- [46] R.B. Haber, C.S. Jog, M.P. Bendsøe, A new approach to variable-topology shape design using a constraint on perimeter, *Struct. Optim.* 11 (1–2) (1996) 1–12.
- [47] J. Petersson, O. Sigmund, Slope constrained topology optimization, *Internat. J. Numer. Methods Engrg.* 41 (8) (1998) 1417–1434.
- [48] G.K. Ananthasuresh, S. Kota, Y. Gianchandani, *A Methodical Approach to the Design of Compliant Micromechanisms*, SC: IEEE, 1994, pp. 189–192.
- [49] O. Amir, O. Sigmund, On reducing computational effort in topology optimization: how far can we go?, *Struct. Multidiscip. Optim.* 44 (2011) 25–29.
- [50] T.H. Nguyen, G.H. Paulino, J. Song, C.H. Le, A computational paradigm for multiresolution topology optimization (MTOP), *Struct. Multidiscip. Optim.* 41 (4) (2010) 525–539.
- [51] O. Sigmund, On the design of compliant mechanisms using topology optimization, *Mech. Struct. Mach.* 25 (4) (1997) 493–524.