Computers and Mathematics with Applications 149 (2023) 84–98

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

Blending spline surfaces over polygon mesh and their application to isogeometric analysis

Tatiana Kravetc

Department of Computer Science and Computational Engineering, Faculty of Engineering Science and Technology, UiT – The Arctic University of Norway, Norway

ARTICLE INFO

Keywords:
Isogeometric analysis
Polygonal finite elements
Blending splines
Expo-rational basis functions

ABSTRACT

Finite elements are allowed to be of a shape suitable for the specific problem. This choice defines thereafter the accuracy of the approximated solution. Moreover, flexible element shapes allow for the construction of an arbitrary domain topology. Polygon meshes are a common representation of the domain that cover any choice of the finite element shape.

Being an alternative tool for modeling and analysis, blending spline surfaces support representation on polygon grids. The blending splines have a hierarchical structure, which is obtained by generating local surfaces that cover each node support and then blended with a special type of basis functions. This type of splines in their tensor product form is suitable for application to isogeometric analysis problems. A more general representation constructed on polygonal elements can be used on a wider range of domain topology in comparison with tensor product surfaces.

In this paper we introduce a novel approach to constructing curvilinear polygon meshes in the blending spline representation in application to the isogeometric analysis context. The focus is on generating a novel special type of basis functions on a connected collection of polygons, with triangles and quadrilaterals as particular cases. The purpose of the proposed paper is to show applications of this construction to various numerical problems, as well as to generalize the approach to evaluating these basis functions on arbitrary planar domains.

1. Introduction

It is common to utilize triangular and quadrilateral elements in a finite element analysis (FEA) context. In terms of isogeometric analysis (IGA) [19] these two finite element shapes require different computational approach, since quadrilateral finite elements in many cases are naturally supported by tensor product surfaces [3], while triangular meshes require more complex representation [20], especially for smooth constrictions. Hence, the generalization of the different type finite element representations, namely, polygon meshes, is a significant topic within the isogeometric analysis context. A general approach for representing smooth polygon meshes is intended to combine computational approaches for both triangular and quadrilateral meshes.

The main goal of the usage of isogeometric approach is to avoid re-meshing while changing the geometry, or even to avoid re-meshing during the refinement process [8]. Conceptually, the isogeometric approach repeats the finite element method, i.e., the domain is initially subdivided into elements, on which a set of linearly independent basis functions is established. A linear combination of basis functions and corresponding coefficients approximates the solution of a differential equation. The main difference between these approaches is that the coefficients are not associated with nodes and, consequently, do not interpolate the solution, but approximate it. The control points, that represent the analysis suitable geometry, do not interpolate this geometry either. The analysis process is as follows: a parametric domain is initially subdivided into elements, then the simulation geometry, or, in other words, the computational domain, is approximated by the smooth spline construction, and after that the solution of the analysis problem can be found on the domain by using the same basis functions that describe the geometry. Fig. 1 illustrates a comparison between the standard FEA approach and the proposed interpretation of the IGA approach. Note that the parametric domain is a collection of convex polygons, where the set of basis functions is defined, while the computational domain consists, in general, of the isoparametric elements, in other words, curvilinear elements.

In FEA the mesh generation process impacts the solution. For example, the symmetry of the solution can be distorted due to the non-

E-mail address: tatiana.kravetc@uit.no.

https://doi.org/10.1016/j.camwa.2023.08.028
Received 26 April 2023; Received in revised form 12 August 2023; Accepted 27 August 2023

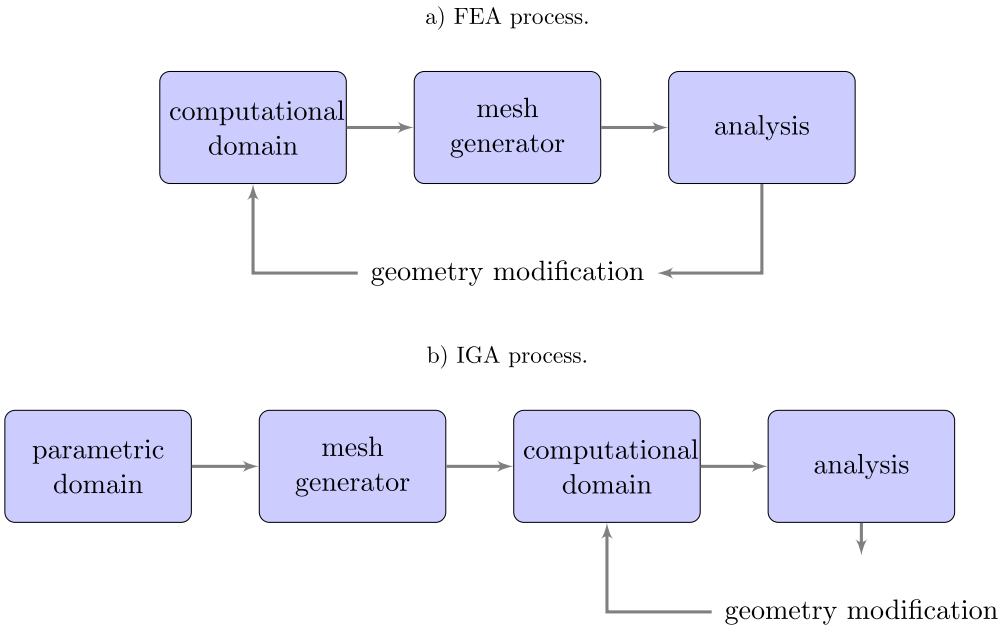


Fig. 1. FEA vs IGA.

symmetric discretization of the domain. Using the continuous representation, that the isogeometric analysis provides, we significantly reduce the discretization impact. The utilization of a smooth basis proves to be effective in various fields, including structural analysis [9,30,31], phase-field models [18], turbulence [1] and fluid-structure interaction [2,4]. The most common tool for the isogeometric analysis are non-uniform rational B-splines (NURBS) and their modifications. Continuous smooth splines provide a more accurate modeling of complex geometries and enable precise computations at the coarsest level of discretization. Additionally, they enhance computational efficiency by increasing the smoothness of elements through k-refinement.

Blending type spline construction is a tool for geometric modeling and isogeometric analysis, first introduced in [12]. The support of blending spline basis functions does not depend on their degree, which means that the basis is strictly local on the entire domain. A concept of blending splines is founded on blending basis functions, in particular, expo-rational basis functions (ERBS) and their generalizations (GERBS) [11,14]. Some related work regarding the utilization of ERBS in finite element context can be found in [13,43].

Blending spline surface is constructed by blending local geometries using a locally defined underlying basis function to form global surface patches. This process provides a hierarchical structure of the resulting surface. A smooth global surface is formed due to the overlapping of local surfaces. Local surfaces interpolate the global surface patches and contain information about surface derivatives at the grid knots. In addition, local surfaces can be manipulated independently, providing flexible modeling options.

An alternative approach for constructing blending spline surfaces is to combine underlying basis functions and basis functions that form local surfaces separately from the control points. This type of basis is called the combined expo-rational basis, it was introduced in [23], and applied to the isogeometric analysis context on tensor product meshes in [27,25].

Besides tensor product surfaces, the blending splines allow for more general surface constructions, such as triangulated surfaces [10,15,44] and polygonal surfaces [24,26]. In the existing representation, the blending triangulated surface consists of a set of connected blending triangles, where each local triangle has support on one triangle patch. That gives, in general, C^0-smoothness over the edges of the global surface. Higher continuity of the global surface can be achieved by manipulating the underlying basis functions through a conformal mapping [44] or by adjusting directional derivatives [10]. C^0-smooth triangular blending splines were applied to FEA in [22].

A brief review of existing polygonal finite element methods can be found in [34]. Polygonal finite elements [37] extend the triangular and quadrilateral finite element methods to meshes with n-sided elements ($n \geq 3$). Rational basis functions on finite element polygon construction were introduced for convex and nonconvex planar polygons by [17], where mean value coordinates are used to construct a polygonal spline space which is used in finite element modeling and analysis. An alternative implementation of a family of polygonal finite elements with conformal mapping can be found in [6]. An adaptive scheme for polygonal mesh structure is developed in [32]. A polytopal composite finite element scheme with application to solid mechanics problems is presented in [33]. Another applications of the polygonal finite element method are plane isotropic elastic problems [42] or crack propagation [21].

The proposed polygonal surface structure is obtained by overlapping of local polygonal surfaces, each of which is defined on the entire node support, i.e. neighboring elements that have one common vertex. The blending polygonal surface structure extends triangulated and tensor product blending surface constructions through the polygonal structure of the patches. The blending polygonal surface has a simple representation, without any additional structures, such as knot net [20], or extra patches [35]. The hierarchical structure of the blending surface follows directly from the local rational basis functions and their node support. Underlying expo-rational basis functions guarantee linear independence of the global basis and partition of unity at any point.

This paper examines a novel polygonal blending spline construction with application to the isogeometric analysis context. Some preliminary definitions related to this paper are given in Section 2. In Section 3 we define a blending spline construction over polygon mesh. Section 4 describes the isogeometric analysis algorithm including the construction of the computational domain. Section 5 focuses on numerical problems solved by the proposed scheme. Finally, we summarize this work and identify research questions for future development in Section 6.

2. Preliminaries

Before constructing a blending spline surface on a polygonal mesh, let us consider preliminary definitions that are used in the sequel of this paper.

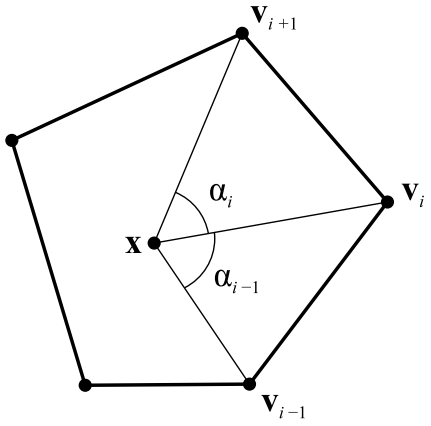


Fig. 2. Notation for mean value coordinates.

First of all, we define how to interpret the solution of a partial differential equation solved using the isogeometric analysis approach with the blending spline construction.

A parametric function that maps the domain $\Omega \subset \mathbb{R}^2$ onto $\mathbb{R}^n$ is constructed by the linear product of basis functions $\varphi_i : \Omega \to \mathbb{R}$ and corresponding coefficients $\zeta_i \in \mathbb{R}^n$, $i = 1, \ldots N$. In matrix form this can be written as

$$S = \zeta^{\mathrm{T}} \varphi, \tag{1}$$

where S is a continuous mapping $S : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^n$. Here, φ is a row vector of the ordered basis functions, ζ is a column vector of the corresponding coefficients.

The combined expo-rational basis functions, defined in [23], were developed on a foundation of the theory of the expo-rational B-splines (ERBS), first introduced in 2006 by L.T. Dechevsky et al. [12]. The concept of blending spline surfaces can be briefly described as blending of local surfaces by underlying expo-rational basis functions. Thus, the resulting surface possesses a hierarchical structure. However, this construction can be interpreted by formula (1), so that the functions φ_i, $i = 1, \ldots, N$, are built using a combination of the underlying and local basis functions.

In Section 3 we consider the evaluation of the combined expo-rational basis functions on a polygon mesh. The evaluation process consists of two steps: (i) polygon mesh generation and structuring, (ii) defining of a set of the combined expo-rational basis functions on the generated mesh, which involves the use of the local barycentric coordinate system on both elements and nodes.

2.1. General barycentric coordinates

Let $\mathrm{P} \subset \mathbb{R}^2$ be a polygon with vertices $\mathbf{v}_1$, $\mathbf{v}_2$, ..., $\mathbf{v}_n$, $n \geq 3$. According to [16], any functions $u_i : \mathrm{P} \to \mathbb{R}$, $i = 1, \ldots, n$, are called generalized barycentric coordinates if, for all $\mathbf{x} \in \mathrm{P}$, $u_i(\mathbf{x}) \geq 0$, $i = 1, \ldots, n$, and

$$\sum_{i=1}^{n} u_i(\mathbf{x}) = 1, \quad \sum_{i=1}^{n} u_i(\mathbf{x}) \mathbf{v}_i = \mathbf{x}. \tag{2}$$

For our purposes we use a type of generalized barycentric coordinates suitable for arbitrary polygons, called mean value coordinates [16], which are defined as

$$u_i(\mathbf{x}) = \frac{w_i(\mathbf{x})}{\sum\limits_{j=1}^{n} w_j(\mathbf{x})}$$

and

$$w_i(\mathbf{x}) = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{||\mathbf{v}_i - \mathbf{x}||}, \tag{3}$$

where the angles $\alpha_i = \alpha_i(\mathbf{x})$, $0 < \alpha_i < \pi$, are shown in Fig. 2.

These coordinates are used for mapping between local polygon coordinates and global coordinates on the parametric domain. The mapping between parametric domain and the Cartesian coordinate system is achieved by a linear combination of basis functions defined on the parametric domain and control points that belong to the Euclidean space.

2.2. Polygon mesh

A polygon mesh is the collection $\mathcal{P}$ of M convex polygons, or, in other words, elements $\mathbf{E}_j$, $j = 1, .., M$. Let $\Theta = \bigcup_{\mathrm{P} \in \mathcal{P}} \mathrm{P}$ be the parametric domain consisting of these polygons. We assume that the interior of any two polygons in $\mathcal{P}$ do not intersect and the intersection of any two polygons is either their common edge or common vertex if the intersection is not empty. Let also any sub-collection of the polygons having one common vertex $\mathbf{v}_i$, $i = 1, ..., m$, be called a node support.

The mesh generation algorithm, called `PolyMesher` and developed in [38], is based on the Lloyd's method [29], which allows us to establish an optimal distribution of elements and thus to construct a uniform mesh. The concept of Voronoi diagrams plays a central role in the meshing algorithm. A discretization of the domain is constructed from a Centroidal Voronoi Tesselation (CVT) which includes an approximation to the domain boundary. The approximation of the boundary is obtained by the reflection of the centroids [5]. As the result, the method offers a simple way to discretize two-dimensional geometries with convex polygons. The isoparametric formulation for polygonal finite elements is an extension of the triangular elements to convex polygons [36].

Besides the array of vertices and the connectivity matrix that describes elements, we also introduce a structure that identifies indices of vertices constituting the node supports. These "node polygons" are the supports for the basis functions that form the polygonal blending spline surface. Fig. 3 demonstrates an example of the polygon mesh and two node supports on it.

3. Blending polygonal spline space

Since blending splines by the definition blend together local surfaces, we need to define a set of local polygonal surfaces, in our case, Bézier polygons formed by polygonal Bernstein polynomials. Despite the convention that the polygon mesh consists of convex polygons, construction of Bézier polygons over non-convex polygons should be involved, because the local surfaces cover each node support consisting of several polygons. For example, Fig. 4(a) shows a polygonal support of the node $\mathbf{v}_i$.

In order to generate a basis for polygonal blending spline construction, we introduce a combined expo-rational basis, which combines Bézier polygonal basis and expo-rational basis functions.

A first component of the combined polygonal expo-rational basis is a set of Bézier basis functions $\beta_{d,\mathbf{a}}$, also denoted as β_i, of degree d defined on a polygon P_n with $n \geq 3$ sides, which is given, in accordance with [17], as

$$\beta_i = \beta_{d,\mathbf{a}}(\mathbf{x}) = \frac{d!}{a_1! a_2! \ldots a_n!} u_1^{a_1}(\mathbf{x}) u_2^{a_2}(\mathbf{x}) \ldots u_n^{a_n}(\mathbf{x}), \quad \mathbf{x} \in \mathrm{P}_n, \tag{4}$$

where $u_1, u_2, \ldots, u_n$ is a set of generalized barycentric coordinates for the given polygon, $\mathbf{a} = (a_1, a_2, \ldots, a_n)$, $a_1 + a_2 + \ldots + a_n = d$, is a multi-index. Fig. 4(c) shows a set of polygonal Bézier basis functions of degree 2 defined on a polygonal support of the node $\mathbf{v}_i$.

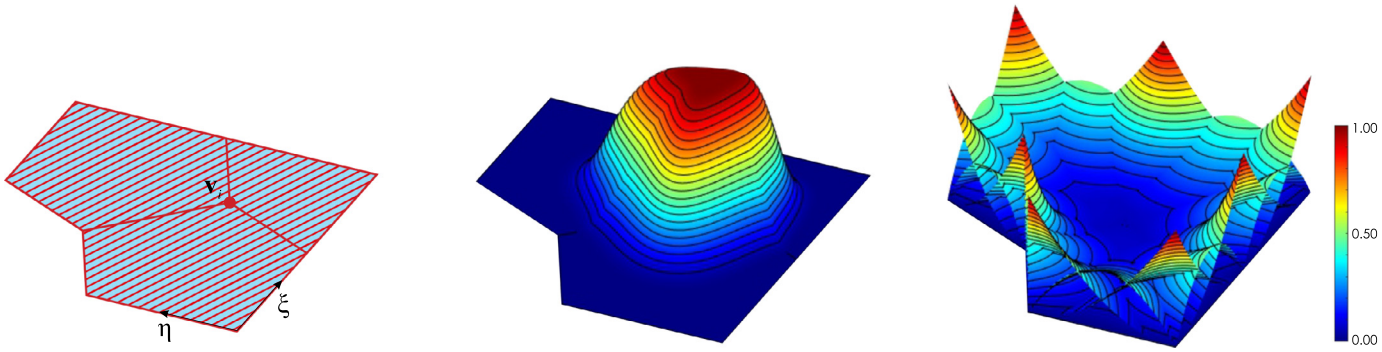
A second component of the combined polygonal expo-rational basis is a locally defined expo-rational basis function.

A univariate expo-rational basis function $B(u)$, introduced in [28], defined along a parameter $u \in (0, 1]$ is expressed as

$$B(u) = \begin{cases} \gamma \int\limits_0^u \phi(s) ds, & \text{if } 0 < u \leq 1, \\ 0, & \text{otherwise,} \end{cases} \tag{5}$$

a) $\mathbf{v}_1$ b) $\mathbf{v}_{11}$

Fig. 3. Nodes $\mathbf{v}_1$, $\mathbf{v}_{11}$ shown on the example of polygon mesh. In the blending spline surface representation each node is covered by the local polygon surface.



a) One node $\mathbf{v}_i$ and its support.

b) Underlying expo-rational basis function ψ_i.

c) Local Bernstein basis functions β_ι of the second degree.

Fig. 4. Illustration of the components of one set of the combined expo-rational basis functions (a) having their support on the node $\mathbf{v}_i$. (b) The "bell"-shaped expo-rational basis function ψ_i and (c) the corresponding polygonal Bernstein polynomials β_i of the second degree.

where

$$\phi(u) = \exp\left(-\frac{(u-1/2)^2}{u(1-u)}\right),$$

and the scaling factor

$$\gamma = \left(\int_0^1 \phi(u)du\right)^{-1}.$$

A set of expo-rational basis functions forms a "bell"-shaped basis function defined on a polygonal node support. The expo-rational basis function $\mathcal{B}(u_k(\mathbf{x}))$ in generalized barycentric coordinates u_k, $k = 1, 2, ..., n$, is defined for any $\mathbf{x} \in \mathrm{P}_n$ by

$$\mathcal{B}(u_k(\mathbf{x})) = \frac{B(u_k)}{B(u_1) + B(u_2) + ... + B(u_n)} \quad \text{for } k = 1, 2, ..., n, \tag{6}$$

where $B(u_k)$ is defined by formula (5). Thus, each $\mathcal{B}(u_k)$ is defined for each vertex of the polygonal element such that $\mathcal{B}(u_k(\mathbf{x})) = 1$ if $\mathbf{x} = \mathbf{v}_k$ and $\mathcal{B}(u_k(\mathbf{x})) = 0$ along edges that do not contain $\mathbf{v}_k$. One underlying expo-rational basis function ψ_i, which is formed as a "bell" shape, has its support on the neighbor polygons having a common vertex, and it is formed as the functions $\mathcal{B}(u_k(\mathbf{x}))$ such that they are equal to 1 at the vertex $\mathbf{v}_i$, and equal to zero along all edges of its support (except the domain boundary). This property of the underlying expo-rational basis functions provides strict locality of the basis, which is especially promising for the polygonal grids in comparison with other smooth spline representations. Fig. 4(b) shows the underlying basis function ψ_i on its support.

By analogy we generate locally defined basis functions (both expo-rational and Bernstein polynomials) for each node $\mathbf{v}_i$, $i = 1, ..., m$.

By multiplying the underlying expo-rational basis functions ψ_i with the corresponding Bézier basis functions β_i over each set of the polygons having a common vertex $\mathbf{v}_i$, we obtain the combined expo-rational basis φ. Fig. 5 illustrates some examples of these basis functions, defined on the polygon node support shown in Fig. 4(a).

The total number of combined expo-rational basis functions on each node is equal to the number of Bernstein polynomials on the corresponding local surface.

An ordered set of basis functions for each node forms a linearly independent combined expo-rational basis on the polygon mesh that can be directly utilized in the isogeometric analysis context, i.e. both geometry and physical characteristics of the system model can be derived in terms of this basis.

4. Isogeometric analysis

The basic idea of the isogeometric analysis is to construct spaces of piecewise continuous functions on the parametric domain that are easy to manipulate, and then to show that one can both approximate a computational geometry and perform analysis on this geometry by these generated functions. The initial parametric domain is partitioned into simplex, such as triangles, or, in general, polygons.

In order to illustrate the main idea of this paper, we consider a parametric domain that belongs to $\mathbb{R}^2$.

Let $\Theta \subset \mathbb{R}^2$ be a continuous parametric domain having two local coordinates $\xi, \eta \in [0, 1]$. Domain boundary is denoted as Γ_Θ.

Introduce a polygon mesh $\mathcal{P}$ that is described by a set of m vertices, or, in other words, nodes, $\mathbf{v}_i$, $i = 1, ..., m$, and a connectivity matrix that defines M polygonal elements $\mathbf{E}_j$, $j = 1, ..., M$. Each set of elements having one common vertex defines a polygonal node support.

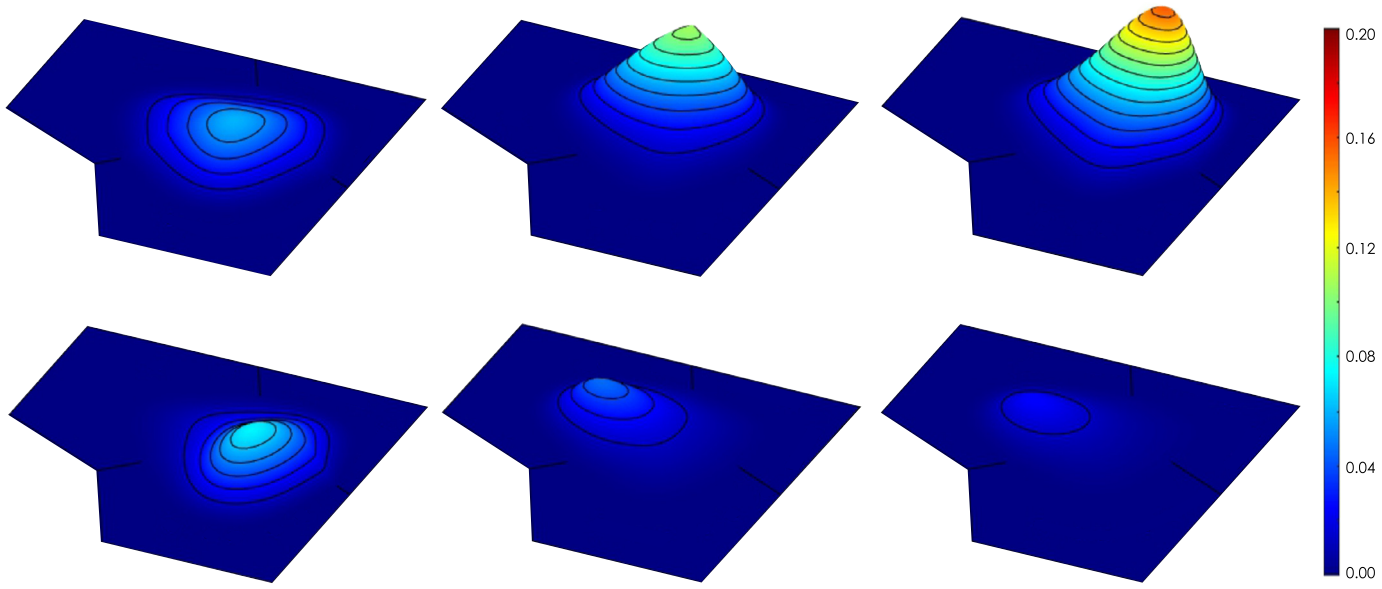


Fig. 5. Some examples of the combined expo-rational basis functions φ defined on the node support $\mathbf{v}_i$ shown in Fig. 4(a) as the result of the multiplication of the functions ψ_i (Fig. 4(b)) and β_i (Fig. 4(c)).

On the mesh we construct a combined expo-rational basis φ, where each basis function has local support. Any function $\vartheta(\Theta) \in \mathbb{R}^n$ can be uniquely approximated by the linear combination of the basis functions φ and the coefficients $\zeta \in \mathbb{R}^n$ such that

$$\vartheta_h = \zeta\varphi.$$

Let $\Omega \subset \mathbb{R}^2$ be a computational domain, i.e. the domain on which we solve the analysis problem. The computational domain boundary is denoted as Γ_Ω. The main idea of utilizing the isogeometric analysis approach here is that we approximate a computational domain boundary Γ_Ω, which is in general curvilinear, by using the same basis functions that we use for analysis. The analysis is performed on the parametric domain Θ, which allows for relatively simple computations and provides a flexible domain boundary that does not require remeshing. Thus, we denote υ_h to be an approximated computational domain. The computational domain is defined by the mapping $\upsilon_h : \Theta \subset \mathbb{R}^2 \to \Omega \subset \mathbb{R}^2$. Note that the domain boundary is preserved as $\Gamma_\Theta \to \Gamma_\Omega$.

While the computational domain is approximated, one can solve the analysis problem by using the basis φ and coordinate transformation between the parametric domain Θ and the computational domain Ω. The solution is defined by the mapping $\vartheta_h : \upsilon_h \subset \mathbb{R}^2 \to \mathbb{R}^n$.

4.1. Domain construction

One of the main differences between the standard finite element method and the isogeometric analysis in accordance with the interpretation presented in this paper is the additional stage between mesh generation and analysis. This stage can be identified as an approximation of the computational domain. Let Θ be a parametric domain, subdivided into polygons. An intermediate stage in analysis is to find a mapping between the parametric domain Θ and a continuous computational domain Ω, which is in general curvilinear.

In order to find a suitable approximation of the computational domain Ω, we solve the Laplace equation on the parametric domain Θ with the given Dirichlet boundary conditions, which is formulated as follows

$$\Delta\upsilon = 0, \quad \text{in } \Theta, \tag{7}$$

$$\upsilon = \Gamma_\Omega, \quad \text{on } \Gamma_\Theta. \tag{8}$$

Here, Γ_Θ and Γ_Ω are the boundaries of the parametric and computational domains, respectively.

The finite element method for this problem is

$$(\mathcal{A}_\Theta + \kappa\mathcal{R}_\Theta)P = \kappa r_\Theta,$$

where

$$\mathcal{A}_\Theta = \int_\Theta \nabla\varphi^{\mathrm{T}}\nabla\varphi \, d\Theta,$$

$$\mathcal{R}_\Theta = \int_{\Gamma_\Theta} \varphi^{\mathrm{T}}\varphi \, d\Gamma_\Theta, \quad r_\Theta = \int_{\Gamma_\Theta} \Gamma_\Omega \, \varphi^{\mathrm{T}} \, d\Gamma_\Theta.$$

A large valued constant κ on Γ_Θ penalizes any deviation of the solution from the given boundary condition, control points P multiplied by the basis functions φ approximate the computational domain. Thus, a mapping $\upsilon_h : \Theta \subset \mathbb{R}^2 \to \Omega \subset \mathbb{R}^2$ is expressed as

$$\upsilon_h = \varphi P. \tag{9}$$

4.2. Derivation of a linear system of equations

Let $\Omega \subset \mathbb{R}^2$ be a computational domain with boundary Γ_Ω approximated by a planar blending tensor product surface. As an example we consider the Poisson's equation

$$-\Delta\vartheta = f, \quad \text{in } \Omega, \tag{10}$$

with Robin boundary conditions,

$$\nabla\vartheta \cdot \bar{n} = \kappa(\vartheta - g_D) - g_N, \quad \text{on } \Gamma_\Omega, \tag{11}$$

where g_D and g_N are given functions, κ is a penalty constant. A large κ specifies the Dirichlet type boundary conditions, $\kappa = 0$ yields the Neumann type boundary conditions.

We also introduce the trial and test spaces

$$\mathcal{V}_g = \{\vartheta \,:\, \vartheta \in H^1(\Omega),\, \vartheta|_{\Gamma_\Omega} = g_D\}, \tag{12}$$

$$\mathcal{V} = \{\upsilon \,:\, \upsilon \in H^1(\Omega),\, \upsilon|_{\Gamma_\Omega} = 0\}.$$

Multiplying (10) with a test function $\upsilon \in \mathcal{V}$ and integrating by Green's formula, we obtain the following variation formulation: find $\vartheta \in \mathcal{V}_g$ such that $\upsilon \in \mathcal{V}$ and

$$\int_{\Omega} \nabla\vartheta \cdot \nabla v \, d\Omega + \kappa \int_{\Gamma_\Omega} \vartheta v \, d\Gamma_\Omega = \int_{\Omega} f v \, d\Omega + \int_{\Gamma_\Omega} (\kappa g_D + g_N) v \, d\Gamma_\Omega. \tag{13}$$

The combined expo-rational basis functions form a set of finite-dimensional approximations of the test and trial spaces. The computational domain is defined as a mapping $v_h : \Theta \to \Omega$ consisted of M curvilinear (in general) elements. To compute the finite element approximation of the desired function ϑ_h let $\varphi = \{\varphi_i(\xi,\eta)\}_{i=1}^{N}$ be a basis defined on the parametric domain Θ with N combined expo-rational basis functions of local degree d defined as described in Section 3. We represent a discrete solution ϑ_h in the blending spline representation, that is,

$$\vartheta_h = \sum_{i=1}^{N} \varphi_i(\xi,\eta)\zeta_i = \varphi\,\zeta, \tag{14}$$

where ζ_i, $i = 1, ..., N$, are unknown coefficients.

The test functions v_h are defined as

$$v_h = \varphi_j(\xi,\eta), \quad j = 1, ..., N. \tag{15}$$

Substituting (14) and (15) into the variational formulation (13) we define a system of N PDEs for N coefficients ζ_i, $i = 1, ..., N$. In matrix form we write this as

$$(\mathcal{A} + \mathcal{R})\zeta = b + r, \tag{16}$$

where the entries of the $N \times N$ stiffness matrix $\mathcal{A}$ and the $N \times 1$ force (load) vector b are defined as

$$\mathcal{A} = \int_{\Theta} \left(J^{-1} \begin{bmatrix} \frac{\partial\varphi}{\partial\xi} \\ \frac{\partial\varphi}{\partial\eta} \end{bmatrix} \right)^{\mathrm{T}} \left(J^{-1} \begin{bmatrix} \frac{\partial\varphi}{\partial\xi} \\ \frac{\partial\varphi}{\partial\eta} \end{bmatrix} \right) |J| \, d\xi d\eta, \tag{17}$$

$$b = \int_{\Theta} f(v_h)\varphi^{\mathrm{T}} |J| \, d\xi d\eta, \tag{18}$$

where v_h is a mapping between the parametric and the computational domains, J is the Jacobi matrix

$$J = \begin{bmatrix} \frac{\partial\varphi}{\partial\xi} \\ \frac{\partial\varphi}{\partial\eta} \end{bmatrix} P. \tag{19}$$

The generalization of the boundary conditions (11) leads to the sparse boundary matrix $\mathcal{R}$ and boundary vector r

$$\mathcal{R} = \int_{\Gamma_\Theta} \kappa\, \varphi^{\mathrm{T}}\varphi \, ||\Gamma'_\Omega|| \, d\Gamma_\Theta, \tag{20}$$

$$r = \int_{\Gamma_\Theta} (\kappa\, g_D(v_h) + g_N(v_h))\, \varphi^{\mathrm{T}} \, ||\Gamma'_\Omega|| \, d\Gamma_\Theta, \tag{21}$$

where $||\Gamma'_\Omega|| = \sqrt{(\Gamma'_\Theta P_x)^2 + (\Gamma'_\Theta P_y)^2}$, Γ'_Θ is a derivative of the basis on the parametric boundary along the corresponding parameter ξ or η, and P_x, P_y are the spatial coordinates of the computational domain control points.

Substituting (17), (18), (20) and (21) into (16) and solving this matrix equation, we finally obtain a solution of the Poisson's equation in the following form

$$\vartheta_h = \varphi\,\zeta.$$

5. Numerical examples

All the numerical examples are performed utilizing the combined expo-rational basis functions constructed on n-sided polygon meshes, where $n \geq 3$. The combined expo-rational functions are used both for the domain geometry construction and for subsequent analysis.

The purpose of these numerical examples is to demonstrate the ability of the blending type spline construction to handle various planar domains, including non-convex, curvilinear and cusp geometries.

In the numerical examples, we compare solutions for PDE problems across varied mesh grids, provide a comparison of the proposed approach with the standard finite element approach, and numerically study the convergence of the presented method.

5.1. L-shape problem

In order to compare the performance of different element shapes: triangular, quadrilateral and polygonal, we consider the Poisson's equation on an L-shape domain Ω depicted in Fig. 6(a). The boundary Γ of Ω consists of two disjoint parts Γ_N and Γ_D, representing the Neumann and Dirichlet boundary conditions, respectively. The considered problem reads

$$\begin{aligned} -\Delta\vartheta &= 0, \quad \text{in } \Omega, \\ \nabla\vartheta \cdot \bar{n} &= g, \quad \text{on } \Gamma_N, \\ \vartheta &= 0, \quad \text{on } \Gamma_D, \end{aligned} \tag{22}$$

where g is expressed in terms of the polar coordinates ρ, ϕ as

$$g = \frac{2}{3}\rho^{-\frac{4}{3}} \begin{bmatrix} x\sin\frac{2}{3}(\phi+\pi) - y\cos\frac{2}{3}(\phi+\pi) \\ y\sin\frac{2}{3}(\phi+\pi) + x\cos\frac{2}{3}(\phi+\pi) \end{bmatrix} \cdot \bar{n}. \tag{23}$$

This problem has an analytical solution $\vartheta = \rho^{\frac{2}{3}} \sin\frac{2}{3}(\phi+\pi)$.

The algorithm used for solving the problem (22) by applying the isogeometric analysis approach and the polygonal blending spline construction, is shortly described as follows. After constructing a mesh grid with the desired shape of elements (triangular, quadrilateral, or polygonal), we generate a set of combined polygonal expo-rational basis functions as described in Section 3. Then we approximate the domain with a planar blending spline surface by projecting the computational domain Ω onto the blending polygonal spline space, and by doing this we find the control points P that define the geometry of the computational domain. Finally, we apply the algorithm described in Section 4.2 and solve the matrix equation (16) with given boundary conditions (23).

Figs. 6(b)-(d) show the approximated solution ϑ_h on different uniform meshes: (b) triangular mesh, (c) polygon mesh, (d) quadrilateral mesh. Fig. 7 compares the errors $||\vartheta_h - \vartheta||_{L_2}$ on the different mesh grids. Here the both mesh refinement and local degree refinement are demonstrated. One can see how the number of degrees of freedom changes with both refinement strategies on different mesh grids. Note that the next refinement step for polygon mesh and the third degree spline, shown as a gray diamond marker in Fig. 7, gives 9206 degrees of freedom, so it is not displayed due to scaling.

One can conclude that the blending type spline construction builded on the polygon mesh does not show better performance than its triangular or quadrilateral version. Moreover, the number of degrees of freedom in the polygonal case rapidly increases, especially when the local spline degree is increasing. However, the objective of this work is to develop a general approach to constructing the combined expo-rational basis functions on the non-tensor-product meshes, where the triangular and quadrilateral meshes are the particular cases of the polygon mesh, as depicted in this example.

5.2. Eigenvalue problem on a circular membrane

In this example we compare the standard FEM approach and the blending-spline-based IGA approach applied to the eigenvalue problem on a circular membrane. We select the same refinement scheme on a triangular mesh for both approaches, but compare the calculation error with respect to the number of degrees of freedom. In addition, this example demonstrates a mapping between the square parametric domain and the circular computational domain.

Let us consider a circular membrane Ω having a radius ρ and a fixed outer boundary Γ_Ω.

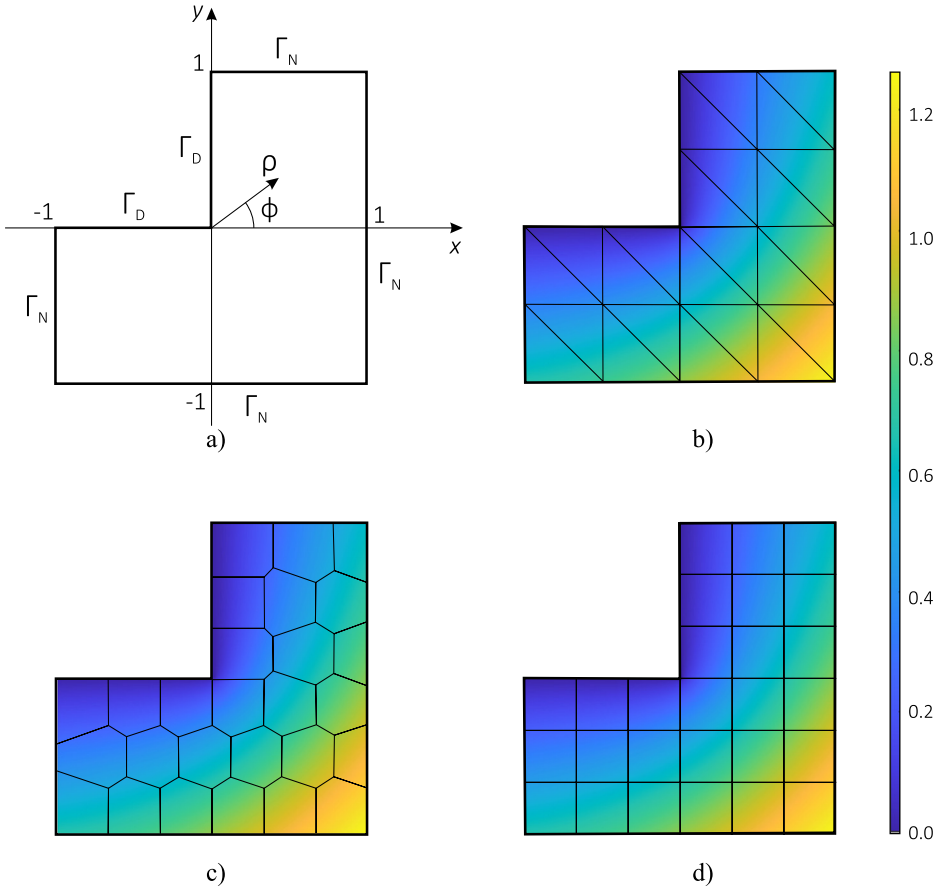


Fig. 6. (a) L-shape domain Ω with boundary parts Γ_N and Γ_D. (b) Solution of the problem (22) on a triangular mesh. (c) Solution on a polygon mesh. (d) Solution on a quadrilateral mesh.

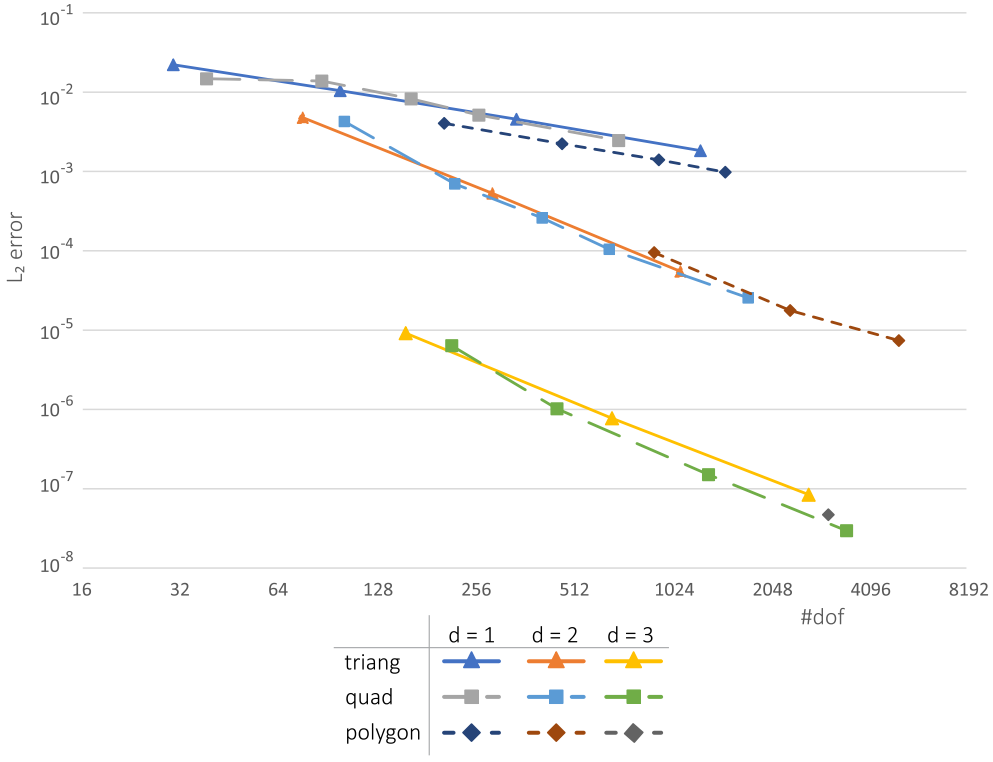


Fig. 7. Comparison of different mesh constructions. L_2 error versus a number of degrees of freedom. Three different local degrees are compared. The mesh is refined uniformly for each case.

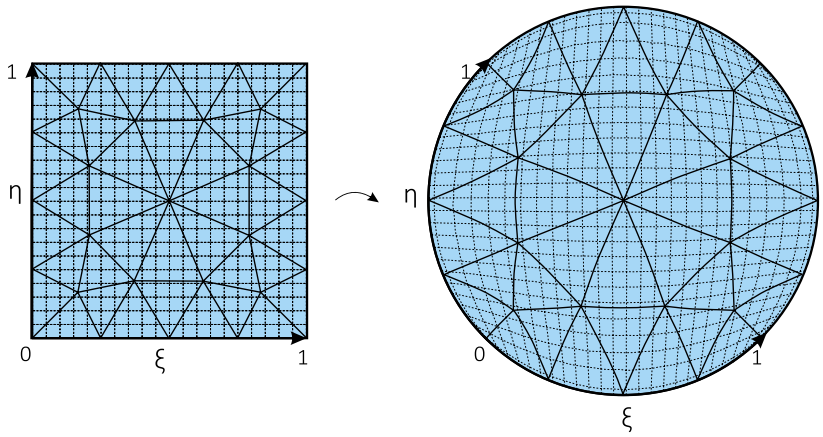


Fig. 8. A mapping between the parametric domain Θ and the curvilinear computational domain Ω approximated by the planar blending spline surface defined on a triangular mesh.

The eigenvalue problem reads as follows

$$-\Delta\vartheta = \frac{\omega^2}{c^2}\vartheta \quad \text{in } \Omega,$$
$$\vartheta = 0 \quad \text{on } \Gamma_\Omega,$$

where c is a material constant and ω is a circular frequency.

An analytical solution of the eigenvalue problem on a circular membrane with Dirichlet boundary conditions is represented by two independent orthogonal eigenfunctions, referred to as the cosine and sine modes, respectively,

$$\vartheta_C^{(\mathrm{m,n})} = \mathcal{J}_\mathrm{m}\left(\omega_{(\mathrm{m,n})}\frac{\rho}{c}\right)\cos(\mathrm{m}\phi) \quad \text{and} \quad \vartheta_S^{(\mathrm{m,n})} = \mathcal{J}_\mathrm{m}\left(\omega_{(\mathrm{m,n})}\frac{\rho}{c}\right)\sin(\mathrm{m}\phi),$$

where $\mathcal{J}_\mathrm{m}$ is a Bessel function.

In order to solve the eigenvalue problem on a circular computational domain by using the IGA approach and the blending type splines, first we introduce the parametric domain $\Theta = [0,1]^2$ with two parameters (ξ,η), and then need to approximate the computational domain Ω. Fig. 8 shows a mapping between the parametric domain Θ and the circular domain Ω obtained by solving the Laplace equation (7) with the boundary conditions (8), where the boundary Γ_Θ transforms to the corresponding circular arc of the boundary Γ_Ω. Thus, a set of points P holds the control points of $v_h : \Theta \to \Omega$.

A goal of the considered problem is to find both the function ϑ as an eigenfunction and $\frac{\omega^2}{c^2} = \lambda$ as an eigenvalue. The finite element method for this problem is formulated as: find $\vartheta_h \in \mathcal{V}_g$, where $\mathcal{V}_g$ is defined as (12), and $\lambda \in \mathbb{R}$ such that

$$(\mathcal{A} + \mathcal{R})\zeta = \lambda\mathcal{M}\zeta, \tag{24}$$

where ζ is a vector holding the coefficients of ϑ_h, and

$$\mathcal{A} = \int_\Theta \left(J^{-1}\begin{bmatrix}\frac{\partial\varphi}{\partial\xi}\\ \frac{\partial\varphi}{\partial\eta}\end{bmatrix}\right)^\mathrm{T}\left(J^{-1}\begin{bmatrix}\frac{\partial\varphi}{\partial\xi}\\ \frac{\partial\varphi}{\partial\eta}\end{bmatrix}\right)|J|\,d\xi d\eta,$$

$$\mathcal{M} = \int_\Theta \varphi^\mathrm{T}\varphi\,|J|\,d\xi d\eta,$$

where J is the Jacobi matrix

$$J = \begin{bmatrix}\frac{\partial\varphi}{\partial\xi}\\ \frac{\partial\varphi}{\partial\eta}\end{bmatrix} P,$$

the boundary matrix $\mathcal{R}$ representing the Dirichlet boundary conditions is expressed as

$$\mathcal{R} = \int_{\Gamma_\Theta} \kappa\varphi^\mathrm{T}\varphi\,||\Gamma'_\Omega||\,d\Gamma_\Theta.$$

Solving equation (24) with respect to ζ we find the coefficients, which, as a linear combination with the basis functions φ, give eigenfunctions of the eigenproblem on a circular membrane. Several solutions are shown in Fig. 9(a)-(d). In Fig. 10 we compare L_2 errors of these eigenfunctions for both FEM and IGA approaches. The errors are computed over the entire domain, not only at the nodal points. As one can see, both approaches provide similar performance with respect to the number of degrees of freedom. Note that the number of coefficients for the blending spline construction drastically increases with refinement. However, comparing the number of elements, the maximum number of elements for the FEM case, shown in Fig. 10, is 660 and the corresponding number of nodes is 361, while the blending spline construction gives 28 elements and 340 degrees of freedom/coefficients, respectively.

5.3. Linear elasticity problem

In order to numerically demonstrate the convergence of the proposed method, we consider an example of the linear elasticity problem. Let a cantilever beam of the length $L = 8$ and height $H = 2$ having a narrow rectangular cross-section be loaded at the free end $x = 0$, as shown in Fig. 11. The given characteristics are as follows: Young's elastic modulus $E = 10^5$, Poisson's ratio $\nu = 0.25$, and the load $P = 200$. According to [41], the analytical solution for the displacement field $\vartheta = (\vartheta_1, \vartheta_2)$ is the following

$$\vartheta_1 = -\frac{Px^2y}{2EI} - \frac{\nu Py^2}{6EI} + \frac{Py^3}{6IG} + \left(\frac{PL^2}{2EI} - \frac{Pc^2}{2IG}\right)y,$$

$$\vartheta_2 = \frac{\nu Pxy^2}{2EI} + \frac{Px^3}{6EI} - \frac{PL^2x}{2EI} + \frac{PL^3}{3EI},$$

where $c = H/2$, E is replaced as $\frac{E}{(1-\nu)^2}$, ν is replaced as $\frac{\nu}{1-\nu}$, $G = \frac{E}{2(1+\nu)}$, $I = \frac{2}{3}c^3$ is the moment of inertia of the cross section of the beam. The stress components are

$$\sigma_{11} = -\frac{Pxy}{I}, \quad \sigma_{22} = 0, \quad \sigma_{12} = -\frac{P}{2I}\left(c^2 - y^2\right).$$

The basic problem of linear elastostatics is to find the stress tensor σ and the displacement vector ϑ such that

$$-\nabla\cdot\sigma(\vartheta) = 0, \quad \text{in } \Omega, \tag{25a}$$
$$\sigma(\vartheta)\cdot\bar{n} = g_N, \quad \text{on } \Gamma_N, \tag{25b}$$
$$\vartheta = g_D, \quad \text{on } \Gamma_D. \tag{25c}$$

The Robin boundary conditions that mix both Neumann (25b) and Dirichlet (25c) boundary conditions have the following expression

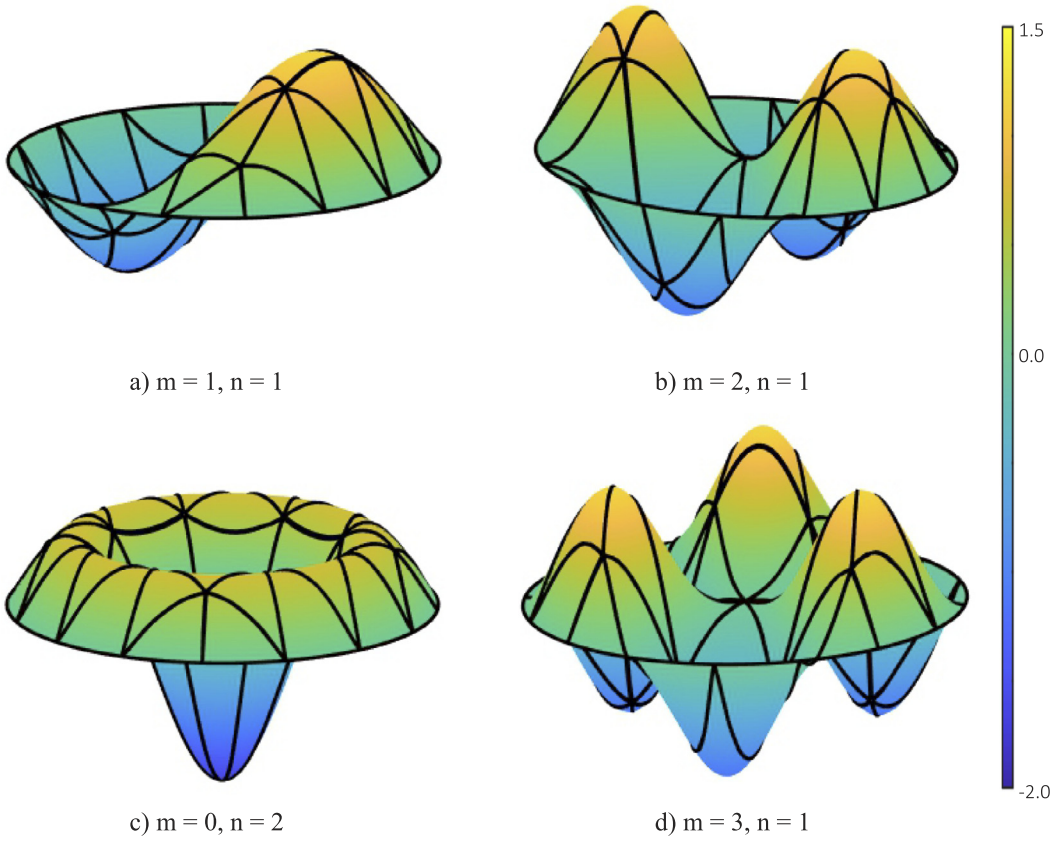


Fig. 9. Eigenfunctions of the circular membrane. Solutions are obtained by the blending spline construction of local degree 2 on a triangular mesh shown in Fig. 8.

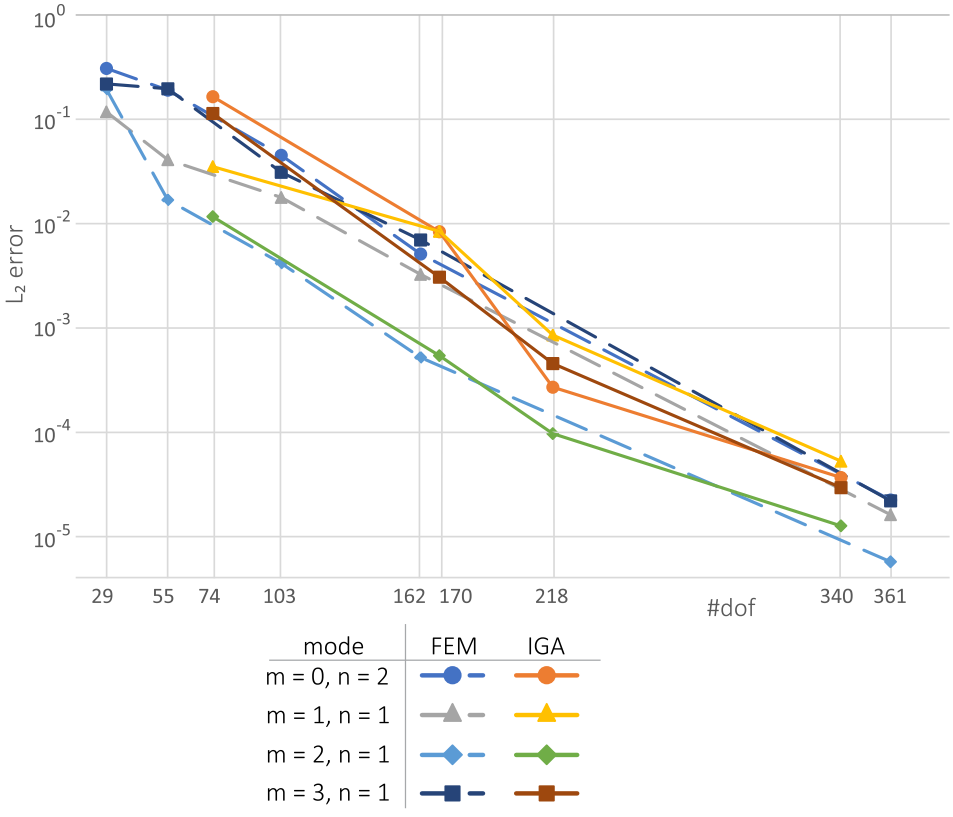


Fig. 10. Comparison of errors for a standard FEM approach with the linear basis and IGA approach on blending splines of local degree 2 constructed on a triangular mesh.

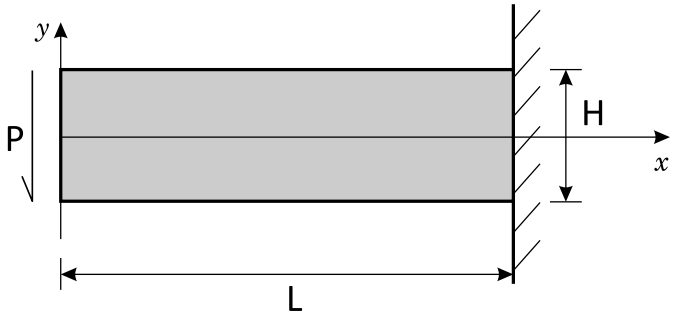


Fig. 11. Geometry of the cantilever beam.

$$-\sigma(\vartheta)\cdot\bar{n}=\kappa(\vartheta-g_D)-g_N,\quad \text{on } \Gamma=\Gamma_D\cup\Gamma_N, \tag{26}$$

where κ is a penalty constant.

Having the parametric domain Θ subdivided into elements, we evaluate the combined expo-rational basis functions φ on this domain. The mapping $v_h=\varphi Q$ defines the computational domain Ω, where Q is the set of control points, identified by projecting the space of basis functions onto the computational domain. An approximation of the displacement field ϑ_h can be represented as

$$\vartheta_h=\begin{bmatrix}(\vartheta_h)_1\\(\vartheta_h)_2\end{bmatrix}=\begin{bmatrix}\varphi_1 & 0 & \varphi_2 & 0 & \dots & \varphi_N & 0\\ 0 & \varphi_1 & 0 & \varphi_2 & \dots & 0 & \varphi_N\end{bmatrix}\begin{bmatrix}\zeta_{11}\\ \zeta_{21}\\ \zeta_{12}\\ \zeta_{22}\\ \vdots\\ \zeta_{1N}\\ \zeta_{2N}\end{bmatrix}=\bar{\varphi}\,\zeta. \tag{27}$$

The strain and stress relation takes the following matrix form

$$\varepsilon(\vartheta_h)=\begin{bmatrix}\varepsilon_{11}(\vartheta_h)\\ \varepsilon_{22}(\vartheta_h)\\ 2\varepsilon_{12}(\vartheta_h)\end{bmatrix}=\mathcal{B}\zeta, \tag{28}$$

$$\sigma(\vartheta_h)=\begin{bmatrix}\sigma_{11}(\vartheta_h)\\ \sigma_{22}(\vartheta_h)\\ \sigma_{12}(\vartheta_h)\end{bmatrix}=\mathcal{D}\mathcal{B}\zeta, \tag{29}$$

where the strain matrix $\mathcal{B}$ is given as

$$\mathcal{B}=\begin{bmatrix}\partial\varphi_1/\partial\xi & 0 & \partial\varphi_2/\partial\xi & 0 & \dots & \partial\varphi_N/\partial\xi & 0\\ 0 & \partial\varphi_1/\partial\xi & 0 & \partial\varphi_2/\partial\xi & \dots & 0 & \partial\varphi_N/\partial\xi\\ \partial\varphi_1/\partial\eta & 0 & \partial\varphi_2/\partial\eta & 0 & \dots & \partial\varphi_N/\partial\eta & 0\\ 0 & \partial\varphi_1/\partial\eta & 0 & \partial\varphi_2/\partial\eta & \dots & 0 & \partial\varphi_N/\partial\eta\end{bmatrix}$$

and

$$\mathcal{D}=\begin{bmatrix}\lambda+2\mu & \lambda & 0\\ \lambda & \lambda+2\mu & 0\\ 0 & 0 & \mu\end{bmatrix},$$

where μ and λ are the Lamé parameters.

Using expressions (28) and (29) we derive the following finite element method formulation for the problem (25a) with boundary conditions (26)

$$\left(\int_\Theta \mathcal{B}^{\mathrm{T}}\mathcal{D}\mathcal{B}\,|J|\,d\xi d\eta+\kappa\int_\Gamma\bar{\varphi}^{\mathrm{T}}\bar{\varphi}\,d\Gamma\right)\zeta=\int_\Gamma(\kappa g_D+g_N)\,\bar{\varphi}^{\mathrm{T}}\,d\Gamma. \tag{30}$$

where $\bar{\varphi}$ is a set of the combined expo-rational basis functions written in the special form as shown in formula (27). The transformation between the parametric and computational domains is achieved using the Jacobian $|J|$. Solving the system (30) with respect to the coefficients ζ, we get the deflection of the cantilever beam approximated by the blending type splines evaluated on the polygon mesh.

The solutions on the refined polygon and triangular meshes are demonstrated in Fig. 12. The combined expo-rational basis functions are evaluated on both types of meshes. The corresponding number of degrees of freedom versus the L_2 error of the displacement magnitude is shown in Fig. 13. One can see that polygons do not exhibit superior efficiency compared to triangular elements, particularly concerning the rate at which the number of degrees of freedom increases upon refinement in the polygonal case. The main weakness is in the evaluation of integrals, which requires a large number of integration points and, consequently, substantial computational expenses. There are methods aiming at enhancing numerical integration, with the gradient correction approach [33,39] being a recent advancement in this direction.

5.4. Potential flow over a wing

In this example we consider an opportunity to modify the computational domain without remeshing. The simulation example is the flow of air over a wing profile. The idea is to change the angle of attack "interactively" to see how the flow modifies. A potential equation for the airflow around the wing follows from the assumption that the velocity field ϑ is steady and irrotational, that is $\partial_t\vartheta=0$ with t time and $\nabla\times\vartheta=0$. Then there exists a scalar function γ such that $\vartheta=-\nabla\gamma$. This is called the flow potential and is given as the solution of the Laplace equation

$$-\Delta\gamma=0. \tag{31}$$

We impose the following boundary conditions

$$\begin{aligned}\bar{n}\cdot\nabla\gamma&=1,\quad \text{on } \Gamma_{\Omega_{\text{in}}},\\ \gamma&=0,\quad \text{on } \Gamma_{\Omega_{\text{out}}},\\ \bar{n}\cdot\nabla\gamma&=0,\quad \text{elsewhere.}\end{aligned} \tag{32}$$

The computational domain Ω has the boundary Γ_Ω that consists of four outer parts: $\Gamma_{\Omega_{\text{in}}}$, $\Gamma_{\Omega_{\text{out}}}$, $\Gamma_{\Omega_{\text{top}}}$, $\Gamma_{\Omega_{\text{bottom}}}$, and one inner part, which is an airfoil boundary $\Gamma_{\Omega_{\text{airfoil}}}$. The outer boundary of the domain Ω is a rectangle having length 60 and width 30. An illustration of the computational domain is shown in Fig. 14(a).

The wing profile is located in the center of the domain and is generated as a pair of parametric equations [45]

$$\Gamma_{\Omega_{\text{airfoil}}}=\begin{bmatrix}0.5+0.5\frac{|\cos(\phi)|^B}{\cos\phi}\\ \frac{T}{2}\frac{|\sin(\phi)|^B}{\sin\phi}(1-x^P)+C\sin(x^E\pi)+R\sin(2\pi x)\end{bmatrix}, \tag{33}$$

where $\phi=[0,2\pi)$ and the following parameters $B=2$, $T=0.1$, $C=0.05$, $P=1$, $E=1$, $R=0$. In addition, we implement the angle of attack using the multiplication of the parametric equations of the airfoil by the rotation matrix.

The parametric domain Θ has two parameters $(\xi,\eta)\in[0,1]^2$ and the following boundaries: $\Gamma_{\Theta_{\text{in}}}$, $\Gamma_{\Theta_{\text{out}}}$, $\Gamma_{\Theta_{\text{top}}}$, $\Gamma_{\Theta_{\text{bottom}}}$, and $\Gamma_{\Theta_{\text{airfoil}}}$, as shown in Fig. 14(b). The wing profile in the parametric domain is represented as a slit with zero thickness. Due to this configuration we can compute the partial derivatives along the boundary $\Gamma_{\Theta_{\text{airfoil}}}$ as $\frac{\partial}{\partial\xi}$.

The initial polygon mesh is generated by using the `PolyMesher` algorithm, as described in Section 2.2. Then we generate a set of combined polygonal expo-rational functions φ on the mesh, as described in Section 3. For example, in Fig. 14(b) one can see a mesh having 80 polygonal elements, and the number of basis functions of local degree 2 generated on this mesh is 8184. To obtain the slit in the mesh we split the local surfaces that cover the nodes that should be divided, into two separate local surfaces by doubling the corresponding vertices.

After generating the set of basis functions, we are ready to approximate the computational domain by solving the Laplace equation (7) with the following boundary conditions

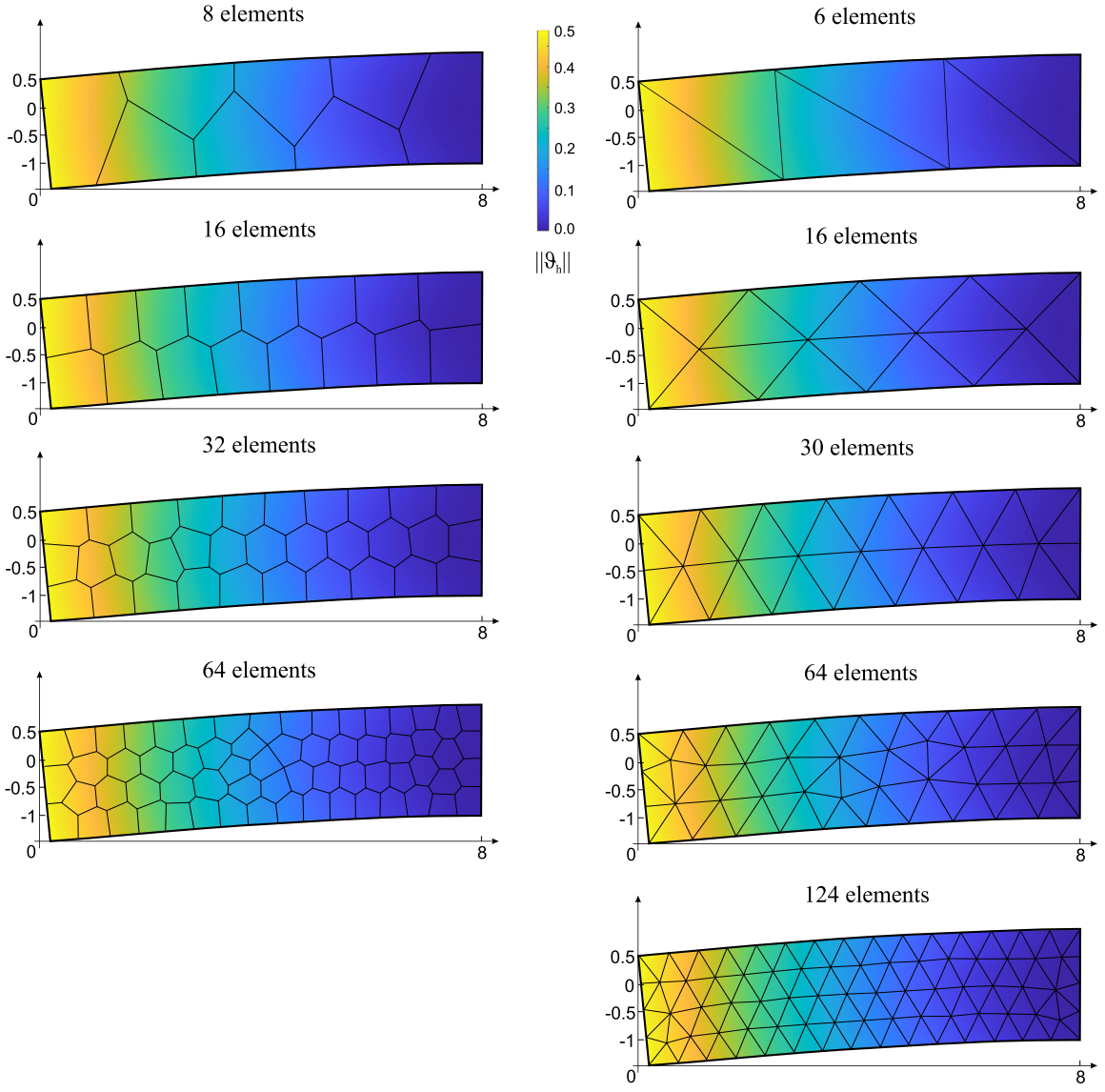


Fig. 12. Polygonal and triangular mesh refinements shown on the deflection of the cantilever beam.

$$\upsilon = \begin{cases} \Gamma_{\Omega_{\text{airfoil}}}, & \text{on } \Gamma_{\Theta_{\text{airfoil}}}, \\ \Gamma_{\Omega_{\text{in}}}, & \text{on } \Gamma_{\Theta_{\text{in}}}, \\ \Gamma_{\Omega_{\text{out}}}, & \text{on } \Gamma_{\Theta_{\text{out}}}, \\ \Gamma_{\Omega_{\text{top}}}, & \text{on } \Gamma_{\Theta_{\text{top}}}, \\ \Gamma_{\Omega_{\text{bottom}}}, & \text{on } \Gamma_{\Theta_{\text{bottom}}}. \end{cases} \tag{34}$$

Thus, we find the control points P such that by multiplying them with the basis functions φ the computational domain is approximated. The mapping $\upsilon_h : \Theta \to \Omega$ is illustrated in Fig. 14(c) and is expressed by formula (9). One can modify the computational domain by changing the boundary conditions (34). For example, we change the angle of attack of the wing by multiplying $\Gamma_{\Omega_{\text{airfoil}}}$ by the rotation matrix. Then we find new control points P and, consequently, modify the computational domain geometry without remeshing, as shown in Fig. 15(a).

Deriving the variational formulation of the problem (31)-(32), we arrive at the following matrix equation

$$(\mathcal{A} + \mathcal{R})\zeta = r, \tag{35}$$

where $\mathcal{A}$, $\mathcal{R}$ and r are defined by formulas (17), (20) and (21), respectively. Then, solving (35) with respect to the coefficients ζ, we obtain the velocity potential $\gamma_h = \varphi\zeta$. The illustration of isocontours of the computed velocity potential γ_h is shown in Fig. 15(b). In order to recompute the solution for the modified domain, it suffices to replace the new control points to the Jacobi matrix (19) and then solve the system (35) again. The initially generated basis functions φ remain the same.

In addition, one can compute and visualize the velocity field ϑ_h, which is expressed by $\vartheta_h = -\nabla\gamma_h$. In our terminology, we interpret the gradient of the velocity potential as

$$\nabla\gamma_h = \begin{bmatrix} \frac{\partial\gamma_h}{\partial x} \\ \frac{\partial\gamma_h}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial\varphi}{\partial x} \\ \frac{\partial\varphi}{\partial y} \end{bmatrix} \zeta = J^{-1} \begin{bmatrix} \frac{\partial\varphi}{\partial\xi} \\ \frac{\partial\varphi}{\partial\eta} \end{bmatrix} \zeta.$$

The result is shown in Fig. 16(a). Note that the illustrated vector field is continuous due to its spline representation, i.e. it can be evaluated at any point. Finally, pressure around the wing can be defined by $p = -||\nabla\gamma_h||^2$. Fig. 16(b) shows this pressure.

6. Conclusion

In this paper, as an alternative to tensor product spline constructions, a new polygonal blending spline construction is developed for application to the isogeometric analysis. The combined expo-rational basis generation scheme covers any convex mesh partitions: standard De-

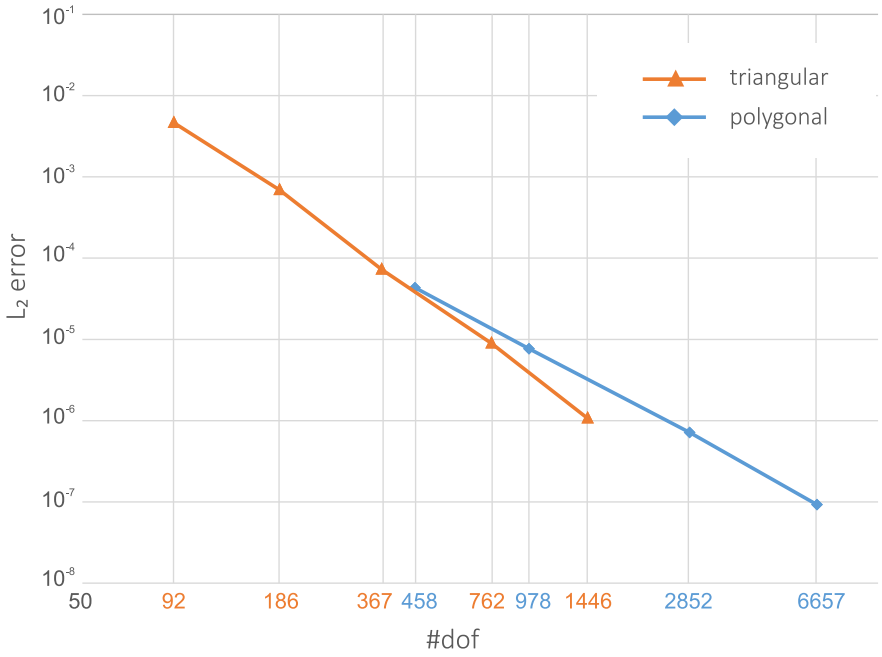


Fig. 13. Displacement error norm for the cantilever beam problem solved by using the blending-spline-based IGA approach. Two shapes of elements are considered: polygonal and triangular. The refinement scheme is shown in Fig. 12. The combined expo-rational basis functions have a second local degree.

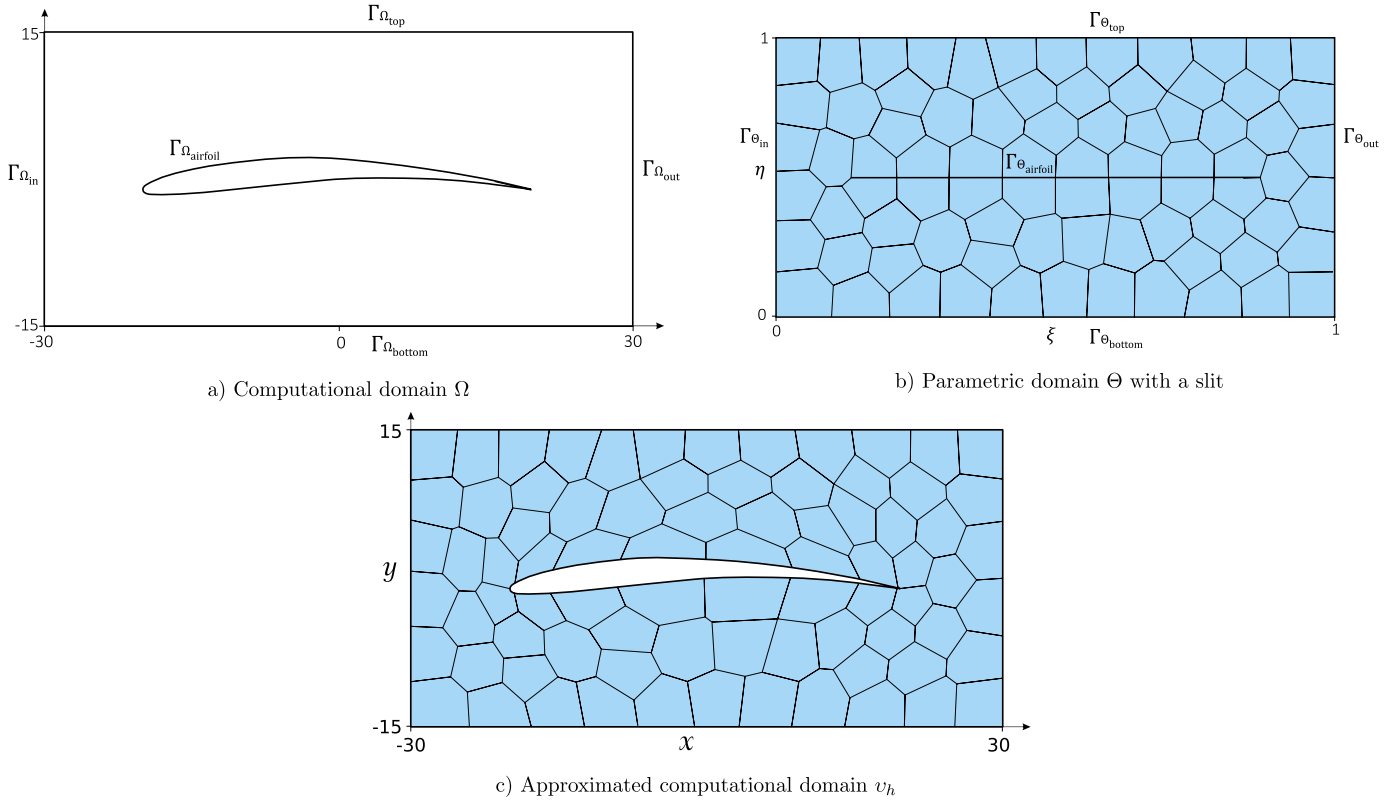
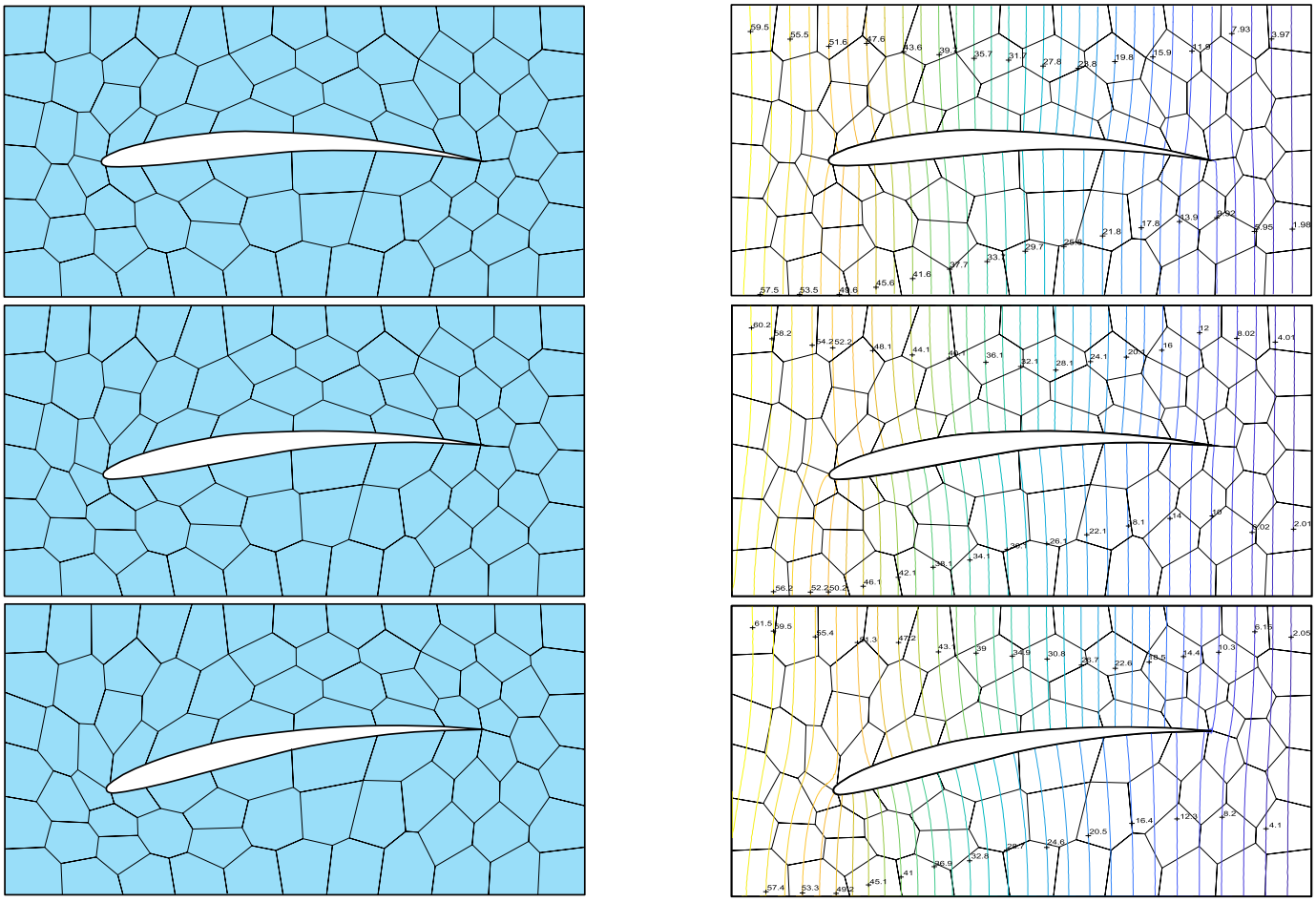


Fig. 14. A process of the domain generation. (a) Original computational domain Ω. (b) Parametric domain Θ subdivided into polygons. (c) Computational domain $v_h : \Theta \to \Omega$ approximated by using polygonal blending spline basis.

launay triangulation, quadrilateral elements, Voronoi tessellation technique. The basis that constructs the polygonal blending spline surface possesses properties that are essential for IGA purposes: linear independence, partition of unity, differentiability. In addition, the basis is strictly local, i.e. each basis function has its support only on neighboring elements that share one vertex.

The polygonal blending spline construction has a hierarchical structure. Bernstein polynomials form local surfaces that are blended by using "bell"-shaped expo-rational basis functions. The local surfaces interpolate the solution and its derivatives at the nodes. The finite elements can be manipulated independently of each other, but they are smoothly connected.



a) Computational domain υ_h b) Velocity potential γ_h

Fig. 15. Changing of the angle of attack (a) illustrated on the meshed computational domain, and (b) illustrated on the solution γ_h.

Due to the flexible polygonal configuration, one can generate free-form domains. Curvilinear domains, holes in the domain are naturally supported. Refinement can be obtained by adding new polygons, preserving all the properties of the basis.

In the presented representation, the number of degrees of freedom is affected not only by the number of sides in the mesh element, but mainly by the local degree of basis functions, which drastically increases the number of degrees of freedom.

Thus, for the proposed construction, triangular elements are preferable, because they maintain a balance between accuracy of approximation and the number of degrees of freedom. In special cases, a combination of triangular and quadrilateral elements can be used. To obtain this, the considered basis functions of a special type were presented on a generalized polygonal grid.

Furthermore, the incorporation of curvilinear smooth elements is primarily aimed at minimizing the element count within the domain. As a result, one can improve precision for various objectives, including managing domain complexity or solution accuracy. This can involve not only altering the element size but also adjusting factors such as the number of sides of an element or the local degree of the basis functions defined on that specific element.

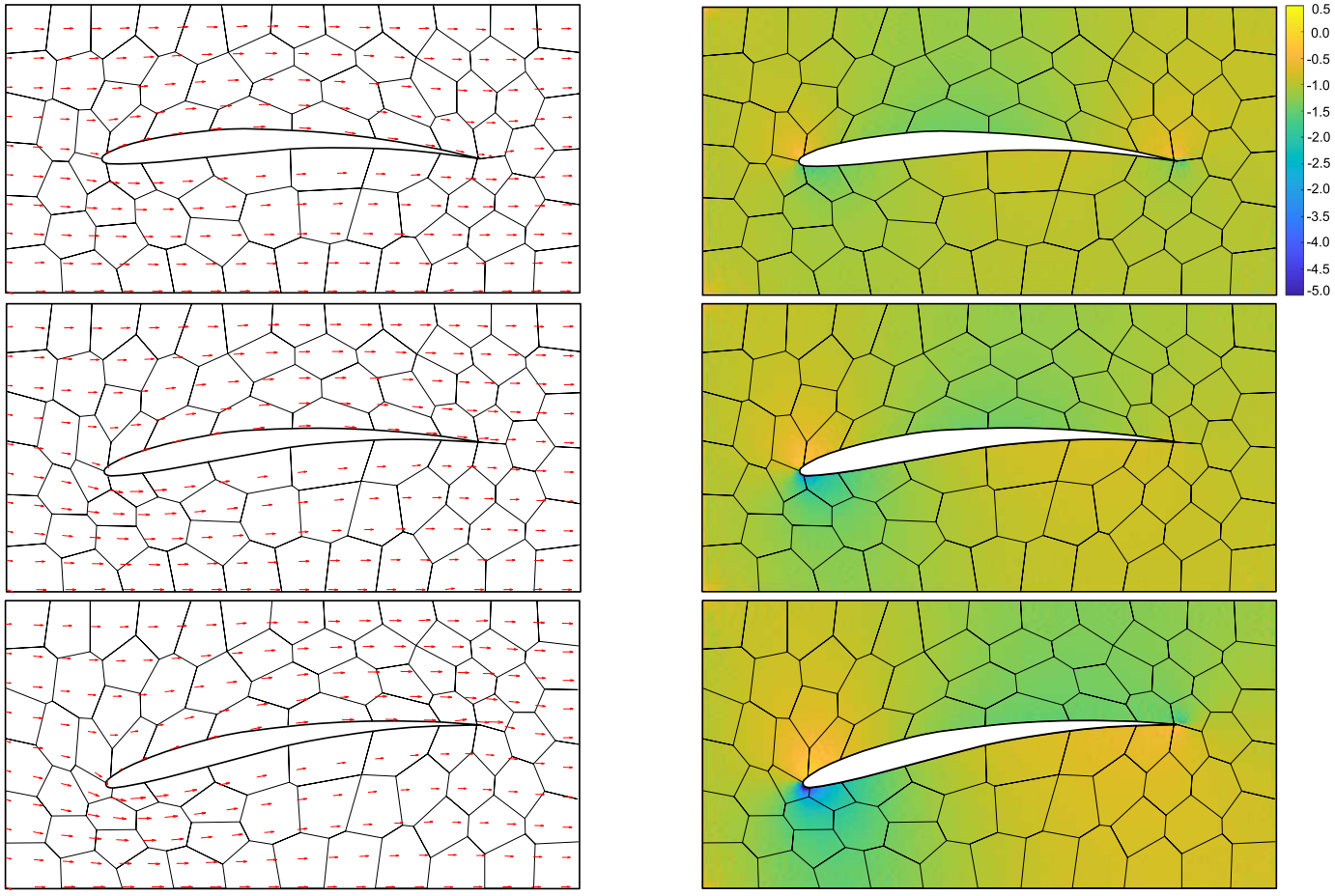
The numerical examples considered in this paper reveal the main properties of the proposed method. In the examples we describe how to build curvilinear computational domains, solve several PDE problems, provide numerical error analysis for various meshing schemes, and compare the performance of the IGA approach on polygonal blending splines with the classical FEM approach. Nonetheless, further investigations are essential, such as a patch test [40] and stability analysis. The patch test should include an examination of varied blending-spline-based constructions, including curvilinear elements, diverse n-gons, and distinct local degrees of basis functions. Concerning elasticity problems, it is crucial to show the stability of the method in incompressibility limit by using the inf-sup condition test [7]. To address these concerns, improvements in the computational process need to be performed.

Further research on this topic will be devoted to the optimization and parallelization of the proposed method, since the main weakness of the method is its computational cost. In order to achieve better accuracy, an explicit formulation of directional derivatives and pre-evaluation of the integrals are required. Another strategy aimed at enhancing the accuracy of calculations may lie in the field of numerical evaluation of integrals. Adaptation of the gradient correction method can lead to better calculations involving fewer integration points.

One potential area where the isogeometric analysis constructed on the polygonal blending splines can be utilized is adaptive design. The objective of the adaptive design is to build a geometry that satisfies given physical conditions. The isogeometric analysis allows for adaptive geometry reconstruction without re-meshing, one can manipulate the geometry completely within the analysis framework. The polygon mesh is flexible for modifications, while the local spline representation provides the ability to approximate both smooth and sharp edges.

Data availability

No data was used for the research described in the article.



a) Velocity field ϑ_h b) Pressure p

Fig. 16. Modifications of the solution of the problem (31)-(32). (a) Velocity field ϑ_h, which is represented as a continuous vector field. (b) Pressure p, which is calculated as $-||\nabla\gamma_h||^2$.

References

[1] B. Bastl, M. Brandner, J. Egermaier, K. Michálková, E. Turnerová, Isogeometric analysis for turbulent flow, Math. Comput. Simul. 145 (2018) 3–17.

[2] J. Bazilevs, Isogeometric analysis of turbulence and fluid-structure interaction, Dissertation for the Degree of Doctor of Philosophy, The University of Texas at Austin, August 2006.

[3] Y. Bazilevs, L. Beirao de Veiga, J.A. Cottrell, T.J.R. Hughes, G. Sangalli, Isogeometric analysis: approximation, stability and error estimates for h-refined meshes, Math. Models Methods Appl. Sci. 16 (2006) 1031–1090.

[4] Y. Bazilevs, V.M. Calo, T.J.R. Hughes, Y. Zhang, Isogeometric fluid–structure interaction: theory, algorithms, and computations, Comput. Mech. 43 (2008) 3–37.

[5] J.E. Bolander, S. Saito, Fracture analyses using spring networks with random geometry, Eng. Fract. Mech. 61 (5–6) (1998) 569–591.

[6] S. Chakraborty, S. Natarajan, S. Singh, D. Mahapatra, S. Bordas, Optimal numerical integration schemes for a family of polygonal finite elements with Schwarz-Christoffel conformal mapping, Int. J. Comput. Methods Eng. Sci. Mech. 19 (4) (2018) 283–304.

[7] D. Chapelle, K.J. Bathe, The inf-sup test, Comput. Struct. 47 (4) (1993) 537–545.

[8] J.A. Cottrell, T.J.R. Hughes, A. Reali, Studies of refinement and continuity in isogeometric analysis, Comput. Methods Appl. Mech. Eng. 196 (2007) 4160–4183.

[9] J.A. Cottrell, A. Reali, Y. Bazilevs, T.J.R. Hughes, Isogeometric analysis of structural vibrations, Comput. Methods Appl. Mech. Eng. 195 (2006) 5257–5296.

[10] R. Dalmo, J. Bratlie, B. Bang, A. Lakså, Smooth spline blending surface approximation over a triangulated irregular network, Int. J. Appl. Math. 27 (1) (2014) 109–119.

[11] L.T. Dechevsky, Smooth convex resolution of unity on general partitions of multidimensional domains and multivariate Hermite interpolation on scattered point sets using radial generalized expo-rational B-splines, in: G. Nikolov, R. Uluchev (Eds.), Constructive Theory of Functions, Sozopol 2010: In Memory of Borislav Bojanov, 2012, pp. 53–68.

[12] L.T. Dechevsky, A. Lakså, B. Bang, Expo-rational B-splines, Int. J. Pure Appl. Math. 27 (3) (2006) 319–367.

[13] L.T. Dechevsky, P. Zanaty, First instances of univariate and tensor-product multivariate generalized expo-rational finite elements, in: Applications of Mathematics in Engineering and Economics 2011, vol. 1410, 2011, pp. 128–138.

[14] L.T. Dechevsky, P. Zanaty Smooth GERBS, Orthogonal systems and energy minimization, in: Applications of Mathematics in Engineering and Economics 2013, vol. 1570, 2013, pp. 135–162.

[15] L.T. Dechevsky, P. Zanaty, A. Lakså, B. Bang, First instances of generalized expo-rational finite elements on triangulations, in: Applications of Mathematics and Engineering and Economics 2011, vol. 1410, 2011, pp. 49–61.

[16] M.S. Floater, Generalized barycentric coordinates and applications, Acta Numer. 24 (2015) 161–214.

[17] M.S. Floater, M.-J. Lai, Polygonal spline spaces and the numerical solution of the Poisson equation, SIAM J. Numer. Anal. 54 (2016) 797–824.

[18] H. Gomez, V.M. Calo, Y. Bazilevs, T.J.R. Hughes, Isogeometric analysis of the Cahn–Hilliard phase-field model, Comput. Methods Appl. Mech. Eng. 197 (2008) 4333–4352.

[19] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Eng. 194 (2005) 4135–4195.

[20] Y. Jia, Y. Zhang, G. Xu, X. Zhuang, T. Rabczuk, Reproducing kernel triangular B-spline-based FEM for solving PDEs, Comput. Methods Appl. Mech. Eng. 267 (2013) 342–358.

[21] A.R. Khoei, R. Yasbolaghi, S.O.R. Biabanaki, A polygonal finite element method for modeling crack propagation with minimum remeshing, Int. J. Fract. 194 (2015) 123–148.

[22] T. Kravetc, Finite element method application of ERBS triangles, Conference presentation, in: NIK2019: Norwegian Informatics Conference, Narvik, Norway, 2019.

[23] T. Kravetc, Representation and application of spline-based finite elements, Dissertation for the Degree of Doctor of Philosophy, UiT – The Arctic University of Norway, 2020.

[24] T. Kravetc, Polygonal blending splines in application to image processing, Conference presentation, in: No. 1 (2021): NIK Norwegian Informatics Conference, Tønsberg, Norway, 2021, pp. 113–135, 2021.

[25] T. Kravetc, Isogeometric analysis using a tensor product blending spline construction, J. Comput. Appl. Math. 414 (2022) 114438.
[26] T. Kravetc, Properties and applications of polygonal blending splines, Conference presentation, in: Curves and Surfaces 2022, Arcachon, France, 2022.
[27] T. Kravetc, R. Dalmo, Finite element application of ERBS extraction, J. Comput. Appl. Math. 379 (2020) 112947.
[28] A. Lakså, Blending Techniques in Curve and Surface Constructions, GEOFO - Geometriforlaget, Narvik, 2022.
[29] S.P. Lloyd, Least squares quantization in PCM, IEEE Trans. Inf. Theory 28 (2) (1982) 129–137.
[30] T.N. Nguyen, T.D. Ngo, H. Nguyen-Xuan, A novel three-variable shear deformation plate formulation: theory and isogeometric implementation, Comput. Methods Appl. Mech. Eng. 326 (2017) 376–401.
[31] T.N. Nguyen, C.H. Thai, A.-T. Luu, H. Nguyen-Xuan, J. Lee, NURBS-based postbuckling analysis of functionally graded carbon nanotube-reinforced composite shells, Comput. Methods Appl. Mech. Eng. 347 (2019) 983–1003.
[32] Son Nguyen-Hoang, H. Nguyen-Xuan, A polytree-based adaptive polygonal finite element method for topology optimization: on an adaptive polytree mesh structure for topology optimization, Int. J. Numer. Methods Eng. 110 (2016) 10.
[33] H. Nguyen-Xuan, K.N. Chau, K.N. Chau, Polytopal composite finite elements, Comput. Methods Appl. Mech. Eng. 355 (2019) 405–437.
[34] L. Perumal, A brief review on polygonal/polyhedral finite element methods, Math. Probl. Eng. 2018 (2018).
[35] P. Salvi, T. Várady, A. Rockwood, Ribbon-based transfinite surfaces, Comput. Aided Geom. Des. 31 (9) (2014) 613–630.
[36] N. Sukumar, A. Tabarraei, Conforming polygonal finite elements, Int. J. Numer. Methods Eng. 61 (12) (2004) 2045–2066.
[37] A. Tabarraei, N. Sukumar, Extended finite element method on polygonal and quadtree meshes, Comput. Methods Appl. Mech. Eng. 197 (2008) 425–438.
[38] 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 (2012) 309–328.
[39] C. Talischi, A. Pereira, I.F.M. Menezes, G.H. Paulino, Gradient correction for polygonal and polyhedral finite elements, Int. J. Numer. Methods Eng. 102 (3–4) (2015) 728–747.
[40] R.L. Taylor, J.C. Simo, O.C. Zienkiewicz, A.C.H. Chan, The patch test – a condition for assessing FEM convergence, Int. J. Numer. Methods Eng. 22 (1) (1986) 39–62.
[41] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, vol. 5(2007), McGraw-Hill, New York, 1970, pp. 157–172.
[42] H. Wang, Q.-H. Qin, Voronoi polygonal hybrid finite elements with boundary integrals for plane isotropic elastic problems, Int. J. Appl. Mech. 09 (03) (2017) 1750031.
[43] P. Zanaty, Finite element methods based on a generalized expo-rational B-splines with harmonic polynomial coefficients, Int. J. Appl. Math. 26 (3) (2013) 379–390.
[44] P. Zanaty, L.T. Dechevsky, Smooth convex partition of unity on uniform triangulation with Hermite interpolation using radial ERBS, in: Applications of Mathematics in Engineering and Economics 2012, vol. 1497, 2012, pp. 190–198.
[45] D. Ziemkiewicz, Simple analytic equation for airfoil shape description, arXiv:1701.00817v1, 2016.