




A Survey of Indicators for Mesh Quality Assessment

T. Sorgente¹  and S. Biasotti¹  and G. Manzini¹  and M. Spagnuolo¹ 

¹Istituto di Matematica Applicata e Tecnologie Informatiche ‘E. Magenes’ - Consiglio Nazionale delle Ricerche, Italy

Abstract

We analyze the joint efforts made by the geometry processing and the numerical analysis communities in the last decades to define and measure the concept of “mesh quality”. Researchers have been striving to determine how, and how much, the accuracy of a numerical simulation or a scientific computation (e.g., rendering, printing, modeling operations) depends on the particular mesh adopted to model the problem, and which geometrical features of the mesh most influence the result. The goal was to produce a mesh with good geometrical properties and the lowest possible number of elements, able to produce results in a target range of accuracy. We overview the most common quality indicators, measures, or metrics that are currently used to evaluate the goodness of a discretization and drive mesh generation or mesh coarsening/refinement processes. We analyze a number of local and global indicators, defined over two- and three-dimensional meshes with any type of elements, distinguishing between simplicial, quadrangular/hexahedral, and generic polytopal elements. We also discuss mesh optimization algorithms based on the above indicators and report common libraries for mesh analysis and quality-driven mesh optimization.

CCS Concepts

• *Computing methodologies* → *Modeling and simulation*; • *Mathematics of computing* → *Mesh generation*; *Numerical analysis*;

1. Introduction

What constitutes a good quality mesh, and exactly how fine the mesh should be, are questions that have been around since the first mesh was generated. Despite decades of research, a high variety can be found in the literature on concepts related to the *quality* of a mesh. The survey papers [She88, Tha80] cover early papers on mesh generation. In these surveys, there is little explicit mention of how to implement the first theoretical results, such as the well-celebrated angle conditions by Zlamal [Zlá68], and Babuska and Aziz [BA76]. However, in [Tha80] it is explicitly stated that elements should be roughly equilateral or instabilities may ensue. Among the first applications, it is worth mentioning the studies by Marchand, Weatherill, and Hassan on the adaptation of unstructured tetrahedral meshes for transonic viscous flow simulation [MWH97]. Liu and Joe [LJ94] developed similar geometric mesh quality indicators in the case of tetrahedral meshes, which can effectively spot geometric flaws in the mesh that can affect the numerical resolution of a PDE. Subsequent studies, cf. [Nie97, Car97, GB98, MV99], provided a more in-depth treatment of the matter, which impacted the design and development of the first mesh-generation codes. More recent surveys exist on mesh quality indicators [Knu01, SEK*07] and mesh generation and processing [BKP*10, ACK13b, BLP*13a, PCS*22].

One thing we have learned from decades of research and dozens of surveys is that the concept of quality depends on the application context. The goodness of a tool principally depends on what that

tool is used for, therefore a mesh considered “good” for one specific application may not work as fine for another. On the one hand, in computer graphics, one generally thinks of mesh quality for model reconstruction and processing, e.g. for guiding repairing operations with dihedral angles or for simplifying the model while keeping the mesh faithful to the acquired set of points [BKP*10, ACK13b]. On the other hand, in the context of simulations by means of finite element methods (FEM), element quality can have a crucial impact on error estimates and convergence rates, thus simulation speed and accuracy [Cia02], and relevant quality indicators depend on the type of simulation [Lis17]. The understanding of how the accuracy of a PDE approximation depends on the mesh is still an active field of research.

At the same time, the breathtaking explosion of the computational power available in modern computers seen in the last years opens up new possibilities, unimaginable so far. Since one of the few concepts universally accepted about mesh quality is that the finer the mesh is, the more accurate results it is likely to produce [LS16], the idea of generating finer and finer meshes, regardless of the notion of quality, becomes terribly tempting. However, there are still situations in which simply decreasing the size of the elements is not enough to guarantee accurate results. Indeed, the standard theory of spline approximation (cf. e.g. [LS07]) and the finite element method (cf. e.g. [BS08]) requires the size of an underlying partition to go to zero to demonstrate the accuracy as well as the convergence of a numerical algorithm for solving a PDE.

However, this fact is not enough to estimate how small the elements must become to be sufficient to guarantee accurate results. Just to give a trivial example, let us imagine covering a triangular domain with some squared cells: no matter how small the squares are, they will never exactly cover the domain. Decreasing the size of the cells surely leads to a better approximation, but there is no way we could ever reach the same result as if we used triangular cells instead. In a more realistic scenario, in presence of particularly complex domains or problems, we may need to exponentially increase the number of elements in order to slightly improve the accuracy of the approximation. In such contexts, it may be worth considering spending some time analyzing the domain and the problem, to generate meshes with higher quality (whatever it means) and fewer elements.

1.1. Contribution

In the literature on PDE solvers, there are plenty of definitions and criteria to measure the quality of elements and meshes. The following review is intended to cover the various flavors proposed, also in relation to the context in which these are defined and used. Both the engineering and the computer graphics communities have contributed significantly to this field in the last decades, proposing seminal ideas, theoretical insights, and practical algorithms. We believe the value of this work lies in its generality:

- It considers the widest possible class of meshes: planar, surface and volumetric, structured and unstructured, from simplicial meshes to quadrangular/hexahedral grids and generic polytopal discretizations;
- It analyzes a similarly wide range of quality indicators: local and global, geometric and algebraic, highlighting connections between such indicators and theoretical error bounds estimates or algorithms for quality optimization.

These two aspects, besides the fact of being up to date with the most advanced notions and techniques, are the main peculiarities of this work and the characteristics that differentiate it from previous STARS in mesh processing and quality indicators.

1.2. Organization

The work is organized as follows. In Section 2 we set the notation and define some basic notions and properties about meshes, polytopes, spectral operators, and quality indicators. Section 3 relates the notion of mesh quality to the discretization and interpolation error in a finite element space. We briefly review some of the historical results on the angle conditions that must be satisfied by a triangular and tetrahedral mesh to ensure a good approximation. We discuss the different impacts that small and large angles have on the interpolation and the condition number of the stiffness matrix that results from an application of the linear Galerkin FEM to a self-adjoint, second-order elliptic operator such as in the Poisson equation. In Section 4 we present the main contribution of the survey: a comparative analysis of the local quality indicators available in the literature. Such local indicators measure the geometric and algebraic properties of the single mesh elements and deal more naturally with finite element simulations. The quality of a mesh will be here related to the performance of a numerical scheme over it,

in terms of speed and accuracy. In addition to the local indicators, in Section 5 we present some examples of global quality indicators that observe the quality of the mesh from a more general perspective. They measure aspects related to the consistency of the mesh with the physical domain, the structure of the mesh, or its connectivity, and are therefore more commonly used in computer graphics problems. In Section 6 we report some applications of the above indicators, i.e., methods for improving or optimizing the quality of a mesh starting from a quality indicator defined over its elements or nodes. Section 7 contains a summary of the most common libraries and datasets for assessing the quality of a mesh and testing a quality indicator. Last, in Section 8 we draw some conclusions and final remarks.

2. Background and Notation

We introduce some basic concepts and properties about meshes, polytopes, and operators that will be used throughout the paper. We focus on two- and three-dimensional meshes containing any kind of polytopal elements with straight edges and planar faces. As a consequence, meshes with curved elements are out of the scope of the present work. Besides being of great interest for some types of applications (e.g. isogeometric analysis [HCB05]), the literature related to the quality of such types of meshes is still scarce.

2.1. Mesh Generalities

A *mesh*, often also called *tessellation*, *domain discretization*, *domain partition*, or *grid* (e.g., in the literature related to finite element methods), is a discrete approximation of an object or a domain $\Omega \subset \mathbb{R}^d$, partitioned into a finite collection of disjoint closed cells. The mesh is called *planar* if $d = 2$, or *volumetric* if $d = 3$. There exists a half-way case, i.e., meshes in \mathbb{R}^3 made by planar cells, also known as *surface* meshes. We consider them together with planar meshes, as they are locally planar, and refer to both as “2D-meshes”, as opposed to “3D-meshes”, which refers to volumetric ones.

The *cells* (or *elements*) of the mesh are subsets of \mathbb{R}^d with no holes and no self-intersections. The boundary of a cell is composed of 2-dimensional *faces* (in 3D-meshes), 1-dimensional *edges* and 0-dimensional *nodes* (or *vertices*), and two cells in a mesh can only share faces (in 3D-meshes), edges, and nodes. A mesh relative to a domain Ω with maximum edge length h (also called the *mesh size*) is noted Ω_h . We note by h_E the *diameter* of an element $E \in \Omega_h$, defined as the maximum point-to-point Euclidean distance in E . Similarly, we note by h_f the diameter of a face $f \in \partial E$ and by h_e the length of an edge $e \in \partial f$, see Figure 1. We also indicate with e_{ij} the edge shared by the faces f_i and f_j and with f_{ij} the face shared by the elements E_i and E_j .

A mesh is called *pure* if all its elements are of the same type, i.e., they have the same number of edges in 2D or faces in 3D. Common examples are triangular, quadrangular, tetrahedral, and hexahedral meshes. The number of cells adjacent to a node (equivalently, the number of incident edges in 2D, or incident faces in 3D) is called its *valence*. Meshes typically have a majority of *regular* nodes with a fixed valence and a certain number of *singular* or *irregular* nodes with different ones. A mesh is said to be *structured* if every internal

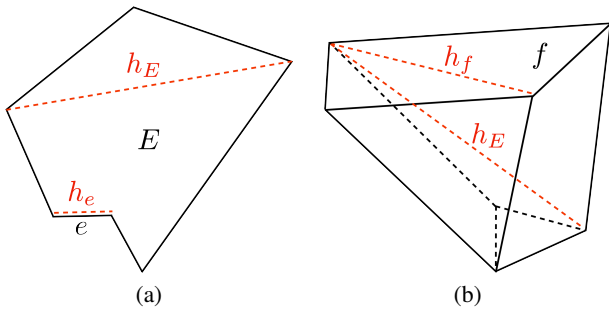


Figure 1: (a) polygon E with diameter h_E and its edge e with length h_e ; (b) polyhedron E with diameter h_E and its face f with diameter h_f .

node is regular; otherwise, the mesh is called *unstructured*. A mesh is said to be *conforming* if two adjacent cells can only share a node, a whole edge, or a whole face. Otherwise, the mesh is called *non-conforming*, and the intersections inside edges or faces are called *T-junctions*, or hanging nodes (see Figure 2).

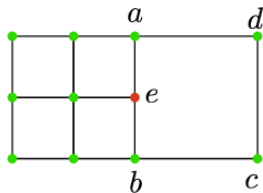


Figure 2: If the rightmost cell is defined as “ $aebcd$ ”, then the mesh is conforming but not purely quadrangular. If it is defined as “ $abcd$ ” instead, the mesh is pure but non-conforming, and the node e is a T-junction.

2.2. Polytopes Generalities

The elements in a mesh are *polytopes*, i.e., polygons for 2D-meshes and polyhedra for 3D-meshes. Following [PS85], a *polygon* is defined by a finite set of line segments such that every segment extreme is shared by exactly two edges and no subset of edges has the same property. A *polyhedron* is a finite set of plane polygons such that every edge of a polygon is shared by exactly one other polygon and no subset of polygons has the same property. As commonly assumed [PS85], we consider polyhedra such that no pair of nonadjacent faces share a point.

A polytope P is said to be *convex* if, given any two points p_1 and p_2 in P , the line segment connecting p_1 and p_2 is entirely contained in P . Two points p_1 and p_2 in P are said to be *visible* from each other if the segment (p_1, p_2) does not intersect the boundary of P . The *kernel* of P is the set of points in the interior of P from which all the points in P are visible. The first obvious consideration is that the kernel of a polytope is a convex polytope. If P is convex, its kernel coincides with its interior, because any two points inside a convex polytope are visible from each other. A polytope may also not have a kernel at all; in this case, we say that its kernel is *empty*. A polytope P is called *star-shaped* if there exists a ball from which all the

points in P are visible, see Figure 3. Note that the requirement of ‘not intersecting the boundary’ in the definition of visibility implies that a star-shapedness ball cannot have radius zero. Consider the case of a polygon \tilde{P} star-shaped with respect to a ball \tilde{B} with zero radius, i.e., a segment or a point. In particular, this means that \tilde{P} contains at least one edge \tilde{e} aligned with the points in \tilde{B} , and therefore it is not possible to connect a point in \tilde{e} to a point in \tilde{B} without touching the other points of the boundary of \tilde{P} (the same argument holds for a polyhedron). A polytope is star-shaped if and only if its kernel is not empty, therefore star-shapedness can be thought of as an indicator of the existence of a kernel. Star-shapedness is weaker than convexity, and it is often used in the literature as many theoretical results in the theory of polynomial approximation in Sobolev spaces rely on this condition [BS08, DS80].

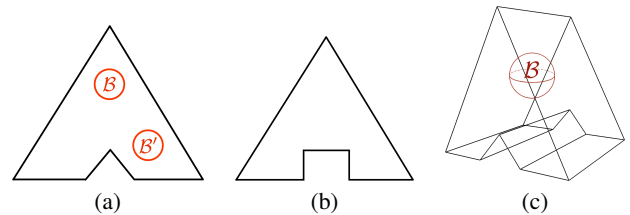


Figure 3: The polygon (a) is star-shaped with respect to the disk B but not with respect to the disk B' ; the polygon (b) is not star-shaped with respect to any disk; the polyhedron (c) is star-shaped with respect to the ball B .

For every polytope P , we can define an *inradius* r_P and an *circumradius* R_P , which are, respectively, the radii of the maximum inscribed and the minimum circumscribed circle (or sphere). We note the area and the perimeter of P as A_P and p_P respectively, and every time there is no place for ambiguity we omit the subscript P . The *solid angle* θ_i at vertex \mathbf{v}_i of a tetrahedron $T(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4)$ is defined to be the surface area formed by projecting each point on the face opposite to \mathbf{v}_i to the unit sphere centered at \mathbf{v}_i , see Figure 4. It measures how large an object appears to an observer looking at it from a point, with a natural geometrical relationship to object visualization. In contrast, the *dihedral angle* between two adjacent faces of a tetrahedron is the angle between the intersection of the two faces and a plane perpendicular to their common edge.

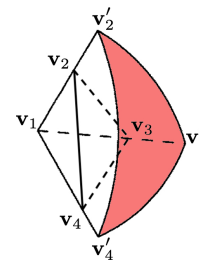


Figure 4: The solid angle θ_1 at vertex \mathbf{v}_1 of a tetrahedron $T(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4)$ is the surface area of the spherical triangle $T'(\mathbf{v}'_2, \mathbf{v}'_3, \mathbf{v}'_4)$ (colored in red), where the \mathbf{v}'_i are the projections of the \mathbf{v}_i on the unit sphere centered at \mathbf{v}_1 .

2.3. Spectral Operators

Consider a triangle with vertices $\mathbf{v}_i = (x_i, y_i)$, for $i = 1, 2, 3$ in physical space. Let ξ_i , with $0 \leq \xi_i \leq 1$ and $\xi_0 + \xi_1 + \xi_2 = 1$, be three logical space coordinates and define the mapping from logical to physical space by $\mathbf{x}(\xi) = \sum \xi_i \mathbf{v}_i$ with $\mathbf{x} \in \mathbb{R}^2$. This mapping can be explicitly written as

$$\mathbf{x} = (1 - \xi_1 - \xi_2)\mathbf{v}_0 + \xi_1\mathbf{v}_1 + \xi_2\mathbf{v}_2 = \mathcal{A}_0\mathbf{u}_0 + \mathbf{v}_0,$$

with $\mathbf{u}_0 = (\xi_1, \xi_2)^t$ and

$$\mathcal{A}_0 = \begin{bmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{bmatrix}$$

We refer to \mathcal{A}_0 as the *Jacobian matrix* of the triangle relative to its vertex \mathbf{v}_0 , because the columns of the matrix form the Jacobian of the affine map with respect to the logical variables, i.e., $\mathcal{A}_{ij} = d\mathbf{v}_i/d\xi_j$. We can define a Jacobian matrix for each vertex of the triangle:

$$\mathcal{A}_i := \begin{bmatrix} x_{i+1} - x_i & x_{i+2} - x_i \\ y_{i+1} - y_i & y_{i+2} - y_i \end{bmatrix}$$

where the indices are taken modulo 3. The Jacobian matrix is not independent of the node on which it is computed, but the determinants of the three \mathcal{A}_i coincide (and they equal twice the area of the triangle). Therefore we omit the subscript and note $\mathcal{J} := \det(\mathcal{A}_i)$, the *Jacobian* of the element. Analogous matrices can be defined for a tetrahedron with vertices $\mathbf{v}_i = (x_i, y_i, z_i)$, for $i = 1, 2, 3, 4$. However, in order to maintain the invariance of the determinant with respect to the order of the vertices, an alternate minus sign is added:

$$\mathcal{A}_i := (-1)^i \begin{bmatrix} x_{i+1} - x_i & x_{i+2} - x_i & x_{i+3} - x_i \\ y_{i+1} - y_i & y_{i+2} - y_i & y_{i+3} - y_i \\ z_{i+1} - z_i & z_{i+2} - z_i & z_{i+3} - z_i \end{bmatrix}$$

The determinant \mathcal{J} of the Jacobian matrix of a tetrahedron gives six times its volume, no matter which of the four matrices is used [Fre97]. The concept can be extended to non-simplicial polytopes like quadrangles and hexahedra, but in this case, the invariance is not guaranteed. A possible solution is to consider the different matrices \mathcal{A}_i at the corners and at the center of the polytope, and define the Jacobian as $\mathcal{J} := \min_i \det(\mathcal{A}_i)$ or as a harmonic or geometric average of the values [Knu03].

According to [Knu99], the Jacobian matrix is the fundamental theoretical object in structured mesh generation. It is fundamental because the columns of the matrix are vectors that point in the direction of the tangents to coordinate lines. Their lengths control the edge lengths of the mesh, their dot products are related to the angles between coordinate lines, and the determinant of the Jacobian matrix is simply related to the volume of the mesh element. Several quality indicators in Section 4 will be based on the condition number $\kappa(\mathcal{A}_i) = \|\mathcal{A}_i\| \|\mathcal{A}_i^{-1}\|$, or on the Jacobian \mathcal{J} . For instance, if $\mathcal{J} \leq 0$, the implied element is said to be *irregular*, and it is considered invalid in the context of finite element methods [Knu00a, MPW71]. Sometimes a distinction is made between degeneration ($\mathcal{J} = 0$) and inversion or fold-over ($\mathcal{J} < 0$). Depending on the concrete setting, such elements may lead to “inaccurate solutions or no solutions at all” [Bar96], “invalidated” solutions [RGPS11], or situations in which “calculations cannot be continued” [SB94]. Other useful entities are the *metric tensors* $\mathcal{A}_k^T \mathcal{A}_k$

relative to the Jacobian matrix \mathcal{A}_k at each vertex \mathbf{v}_k . These matrices are symmetric, and we note λ_{ij}^k the ij -th component of the k -th metric tensor. At the k -th node, the λ_{ii}^k are the squares of the lengths of the edges incident to \mathbf{v}_k , and they coincide with the eigenvalues λ_i^k of $\mathcal{A}_k^T \mathcal{A}_k$ [Knu03].

2.4. Quality Indicators

We introduce a generic notion of a quality indicator that anticipates the more rigorous and particular definitions presented in Section 4.

A quality indicator is a function defined over a mesh, capable of giving insights on the accuracy and the convergence speed of a finite element scheme applied on that mesh, before solving any numerical problem.

As we will see in the next section, quality indicators can assume very different shapes and can be formulated in terms of different properties, with different approaches. However, there has been in the literature an effort to determine some basic properties that any “good” indicator should satisfy. First of all, we point out the difference between an indicator, a measure, and a metric. We highlight certain freedom in the literature, related to this terminology. *Indicator* is a generic term, that stands for a map $f : \Omega_h \rightarrow \mathbb{R}$ defined on the mesh, its elements, or its nodes. By using the term *measure* we assume (among other more obvious properties) that f is positive and bounded. For instance, all indicators taking values in the range $[-1, 1]$ can not be called quality measures unless re-scaling them opportunely. Similarly, indicators that assume infinite values for degenerate elements are not accepted. A *metric* (or *distance*) instead, is required to be positive but not bounded, and to satisfy the triangle inequality. The triangle inequality is a particularly strong condition, that makes several indicators not be metrics but just *semi-metrics* [TV08]. It is not so obvious however that this property is more important than others in relation to the concept of mesh quality, therefore it is not always required.

A general property, that turns out to be particularly useful in surveys and comparative works, is the universality with respect to the spatial dimension and the element type. An ideal indicator should be a function computable on both triangles, tetrahedra, quadrangles, and hexahedra, as well as on generic polytopes.

Fair indicators are defined in [Fie00] to be those that possess four attributes: the ability to detect all degenerate elements, size invariance, boundedness, and normalization. An ability to detect all degenerate elements means that f yields a value of zero for two- (three-) dimensional elements that have no area (volume). Size invariance means that similar triangles (tetrahedra) or quadrilaterals (hexahedra) should yield the same value of f . Boundedness means that f cannot yield arbitrarily large values. Normalization forces f to take on positive values between zero and one, allowing better comparisons among indicators.

More properties are listed in [Knu01, Knu03], like the invariance with respect to the orientation of the element, to translations, to the nodes indexing. It is also required that f is referenced to an ideal element that describes the desired geometric configuration of the physical element and that $0 \leq f \leq 1$, with $f = 1$ if and only if the physical element attains the ideal node configuration and $f = 0$ if

and only if the physical element is degenerate. It is worth noting that size invariance may be required as a property for a good indicator, but the size of the elements (area or volume) may as well be considered a quality indicator itself.

3. Mesh Quality and the Finite Element Method

Finite element methods (FEMs) [BS08] and their many old and recent variants, such as the Discontinuous Galerkin method [BMM*00], the Mimetic Finite Difference (MFD) method [BdVLM14], the Polygonal Finite Element Method (PFEM) [ST04], the Virtual Element Method (VEM) [BadVBC*13], the Hybrid-High Order (HHO) Method [DPD19], etc., as well as the Finite Volume Methods [BM04, BM07] provide very effective numerical approximation strategies on a wide range of problems involving partial differential equations. All these methods have in common that their formulation requires a family of properly refined meshes. The numerical solution accuracy depends on several factors. The most important ones are the degree of the approximation schemes, how the mesh approximates the geometry of the computational domain, the number of elements in the mesh, their size, and their shape. The question we address in this section, i.e., what constitutes an appropriate mesh for solving an elliptic PDE such as the Poisson equation using the finite element method, is as old as the method itself. The interplay between the continuous formulation, its discretized form, and the mesh has a fundamental role in determining how accurate the numerical solution is.

It must be clear that the concept of "mesh quality" when solving real-world applications described by PDEs is inherently problem-dependent, and assessing whether a given mesh is suitable for a given application is more challenging than it would be for the most straightforward, simple academic problems. Still, reviewing the crucial mesh assumptions for the finite element method on simple academic problems makes sense, as we can use these theoretical results to establish mesh quality measurements. We will briefly review the most significant achievements concerning the concept of mesh quality when we apply the piecewise linear finite element method to solving elliptic equations on meshes of simple-shaped elements, such as triangles and quads in 2D and tetrahedra and hexahedra in 3D. For the notation and specific details concerning the FEM we refer to [BS08].

3.1. Discretization and Interpolation Errors

To discuss what a "good mesh" is in an application of the FEM, we first make a distinction between a-priori and a-posteriori analysis. A-priori analysis establishes the accuracy of the FEM depending on the degree of the method and the regularity of the solution. A-posteriori analysis determines the critical regions of the computational domain where a mesh refinement could be beneficial to improve the accuracy of the numerical solution after such a solution has been computed. We address this last topic in Section 6.

Let u be the exact solution of an elliptic problem in variational form, and u_h its numerical approximation. The *discretization error* is the difference between the "true" solution u and its finite element

approximation u_h . We define the discretization error using the L^2 norm and the H^1 seminorm

$$\|u - u_h\|_{0,\Omega}^2 := \int_{\Omega} |u - u_h|^2 d\mathbf{x}, \quad (1)$$

$$|u - u_h|_{1,\Omega}^2 := \int_{\Omega} |\nabla u - \nabla u_h|^2 d\mathbf{x}, \quad (2)$$

and the H^1 -norm

$$\|u - u_h\|_{1,\Omega}^2 := \|u - u_h\|_{0,\Omega}^2 + |u - u_h|_{1,\Omega}^2. \quad (3)$$

In the a-priori analysis, we search for an upper bound to these quantities. A standard result from the theory of the finite element method applied to the variational formulation of a self-adjoint second-order PDE indicates that the discretization error can be bounded by the *interpolation error*. The interpolation error is the error committed when we approximate some "true" function u (not necessarily the solution of a PDE) with an *interpolating function* u_I that belongs to the finite element space. The interpolation u_I shares the degrees of freedom with u (for example, the values at the mesh vertices), assuming that u is sufficiently regular. An application of the Céa's Lemma (cf. [BS08]), allows us to write the inequality

$$|u - u_h|_{1,\Omega} \leq C \inf_{v_h \in \mathcal{V}_h} |u - v_h|_{1,\Omega} \leq C |u - u_I|_{1,\Omega}. \quad (4)$$

Similarly, by using a duality argument usually referred to in the literature as the Aubin-Nische's trick [BS08], we can prove that

$$\|u - u_h\|_{0,\Omega} \leq Ch |u - u_h|_{1,\Omega}. \quad (5)$$

Therefore, if we know an upper bound on the interpolation error in the right-hand side of (4), we estimate the discretization error in the H^1 -norm still from (4). Then, using this upper bound, we estimate the discretization error in the L^2 -norm using (5).

Here, we need to make two comments. First, inequality (4) makes it possible to control the discretization error through the interpolation error. Second, constants C in these inequalities depend on specific features of the problem. For example, the constants C in (4) is determined by the ratio between the continuity constant of the right-hand side functional and the coercivity constant of the bilinear form of the variational formulation. These constant factors are required to be independent of the mesh size parameters h and the exact solution u . Indeed, in the a-priori estimates, we want to outline the dependence on h to establish a convergence rate and the dependence on a higher seminorm of the exact solution u to show the minimal regularity that u must have for this estimate to be valid. Still, the constant C may depend on the domain and the *regularity properties of the mesh*, thus hiding how the mesh impacts on numerical accuracy.

3.2. Discretization and Interpolation Errors and Mesh Quality

Following on the path drawn in [She02], it is thus natural to assess mesh quality in relation to the discretization error and the interpolation error. However, even if we make a distinction between interpolation and discretization errors, these quantities are not completely independent. As we have seen in the previous subsection, we can use the interpolation error to control the discretization error. Clearly, the discretization error can be mitigated by elements whose shape and size are selected to reduce the interpolation error.

However, the dependence of these two types of error on the shape and the size of the mesh elements is different.

On one hand, the discretization error is strongly related to the stiffness matrix: poorly conditioned matrices affect linear equation solvers by slowing them down or introducing large roundoff errors into their results. The relationship between element shape and matrix conditioning depends on the PDE being solved and the basis and test functions used to discretize it, and is reflected in the value of the constant factors C in the estimates. On the other hand, the interpolation error only depends on the approximation properties of the finite element space \mathcal{V}_h , and is, thus, independent of the partial differential equation to be solved numerically.

Both the L^2 and H^1 norms are involved and must be considered. The errors in the gradient approximation, which is measured by the H^1 -seminorm, can be surprisingly important whether the application is rendering, mapmaking, or simulation, because they can compromise accuracy or create unwanted visual artifacts, see Figure 5. Since the finite element space \mathcal{V}_h where we seek u_h , is built on a specific mesh (or sequence of meshes), the interpolation error strongly depends on the geometric properties of the elements (edges, angles, area, inradius, ...). To build a quality indicator, we assume that the interpolation process works the best on some ideal element, e.g., the equilateral triangle and the square in 2D, the equilateral tetrahedron and the cube in 3D. Then, a straightforward indication of how an element is suited to the interpolation is given by computing some sort of “distance” between the actual element and the ideal one.

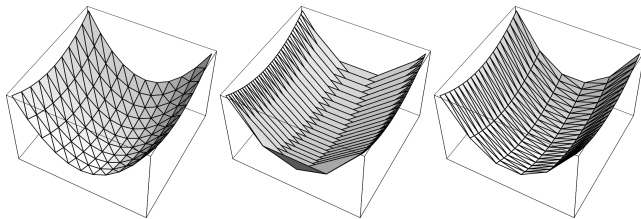


Figure 5: A visual illustration from [She02] of how large angles, but not small angles, can cause the H^1 -seminorm error to explode. In each triangulation, 200 triangles are used to render a paraboloid.

We can also define mesh quality indicators by optimizing the interpolation on a specific family of functions. Each finite dimensional space is locally defined on a mesh element and all the elemental spaces are “glued” together in a way that ensures a global regularity property of the finite element functions. For example, a conforming coupling of the elemental spaces ensures that the finite element functions are globally continuous on the domain Ω . Assuming that the elements are triangles, tetrahedra, quads or hexagons, the finite element functions are normally polynomials of a given degree on every mesh element. Berzins [Ber98] derives a mesh quality indicator from the work [Nad86], which provides a particularly suitable equation for the interpolation error when we approximate quadratic functions by piecewise linear Galerkin finite elements on triangles.

3.3. Angle Conditions

The L^2 and H^1 a priori errors estimates (1)–(2) explicitly depend on the mesh parameter h . However, their dependence on other mesh factors or parameters is not explicit. In fact, the quality of the mesh is normally hidden in the various constants C that “infest” such estimates. From such convergence estimates, we may also (erroneously) assume that decreasing the mesh size, i.e., considering a sequence of meshes with smaller and smaller elements, is sufficient to improve the accuracy of the solution. However, the constants C still depend on the geometrical shape of the elements. This latter one cannot be arbitrary, because, otherwise, the value of C , which is uniform on the whole sequence of refined meshes, would deteriorate.

We can describe the shape of an element in terms of geometrical quantities such as the internal angles, the radius of the circle or sphere that circumscribes the element, etc. As noted in [She02], several Authors in the literature were successful in determining a relationship between the constant C and the geometrical quantities of a given element, thus paving the way for the formulation of mesh indicators, many of which we will review in the next sections. The first historical results in this direction are related to the internal angles of a triangular mesh, and served as the foundation for the first mesh quality assessments. This investigation begins with Zlamal [Zlá68], Babuska and Aziz [BA76], that prove results of the form

$$\|u - u_h\|_{1,\Omega} \leq C(\theta)h|u|_{2,\Omega} \quad \text{where} \quad |u|_{2,\Omega}^2 := \sum_{i,j=1}^2 \left| \frac{\partial^2 u}{\partial x^i \partial y^j} \right|^2,$$

and $C(\theta)$ is a constant factor that depends on the internal angles of the triangles forming the mesh. Zlamal demonstrates that $C(\theta) \simeq 1/\sin(\theta_{min})$, where $\theta_{min} = \min(\theta_1, \theta_2, \theta_3)$, and θ_i , $i = 1, 2, 3$, is the angle of the i -th vertex of a triangular cell. Babuska and Aziz prove that $C(\theta) \simeq 1/\Psi(\theta)$, where $\Psi(\theta)$ is a positive and continuous function such that $\Psi(\theta) \geq \Psi(\gamma)$ for $\theta \leq \gamma < \pi$. Here, γ is a bound on the maximum angle of the mesh, so that the proper condition for triangles is the absence of large angles. Křížek’s [Kř92] extends the result of Babuska and Aziz to three-dimensional tetrahedral elements.

So, according to conventional thinking, triangular and tetrahedral elements should have no small or large angles wherever possible. However, internal angles impact differently on the discretization and the interpolation error. The interpolation error requires a maximum angle condition and is relatively insensitive to small angles. This fact had a significant impact on using anisotropic meshes for solving elliptic problems with strong anisotropies. Based on interpolation errors, Rippa [Rip92] argues convincingly that long, thin triangles provide a proper mesh for solving elliptic problems with highly anisotropic coefficients. Apel and Dobrowolski [AD92] offer a unified theory of interpolation error estimates on nonuniform meshes and application to finite elements for anisotropic problems.

The situation is completely different for the discretization error, since small angles have a major impact on the conditioning of the stiffness matrix. In fact, the condition number may arbitrarily increase if just one single angle of a triangular mesh approaches zero. We discuss this issue in the next subsection.

3.4. Condition Number of the Stiffness Matrix

The concept that we explore in this section is that poor quality elements may significantly affect the convergence and accuracy of the Krylov iterative solvers and the direct solvers that we apply to the system of linear equations resulting from the finite element discretization. The parameter measuring this difficulty is the condition number, and we expect that the higher its value, the harder it is solving such linear systems. In fact, it is possible to prove that if the condition number is big, direct methods may have excessive round-off errors, and iterative solvers may run slowly and even experience non-convergence.

Let K denote the global matrix from an application of the finite element method to the bilinear form of the variational formulation of an adjoint, second-degree PDE like, for example, the Poisson equation with homogeneous boundary conditions, and $\lambda_{max}(K)$ and $\lambda_{min}(K)$ the largest and smallest eigenvalues of K . If we include the boundary conditions, this bilinear form is symmetric, coercive, and continuous, the corresponding matrix K is symmetric and positive definite, and its eigenvalues are real, positive numbers. The condition number of K is defined as $\kappa(K) = \lambda_{max}(K)/\lambda_{min}(K)$.

The minimum eigenvalue $\lambda_{min}(K)$ is related to the size of the elements and the physical system that we want to model. In [Fri72], we find an upper bound for $\lambda_{min}(K)$ that is proportional to the area or volume of the biggest element of the mesh, and a lower bound for $\lambda_{min}(K)$ that is proportional to the area or volume of the smallest element. Therefore, we can claim that, in general, reducing the size of the elements reduces its value. However, the value of the minimum eigenvalue is not very sensitive to the elemental shape, particularly the elemental angles.

Instead, the maximum eigenvalue $\lambda_{max}(K)$ can attain large values even if there is just a single badly-shaped element E in the mesh. Let $\lambda_{max}(K^E)$ denote the biggest eigenvalue of the elemental stiffness matrix K^E . From [Fri72], we also know that

$$\max_E \lambda_{max}(K^E) \leq \lambda_{max}(K) \leq m \max_E \lambda_{max}(K^E),$$

where m is the maximum number of elements meeting at a single vertex of the mesh. Consequently, the condition number $\kappa(K)$ is roughly proportional to the biggest eigenvalue of the elemental stiffness matrices.

We can exemplify this situation by writing the elemental stiffness matrix for a triangle as

$$K^E = \frac{1}{2} \begin{bmatrix} \cot \theta_2 + \cot \theta_3 & -\cot \theta_3 & -\cot \theta_2 \\ -\cot \theta_3 & \cot \theta_1 + \cot \theta_3 & -\cot \theta_1 \\ -\cot \theta_2 & -\cot \theta_1 & \cot \theta_1 + \cot \theta_2 \end{bmatrix}$$

where $\theta_i, i = 1, 2, 3$, are the internal angles of E . We can easily see that $\lambda_{max}(K^E)$ and $\lambda_{min}(K)$ approach infinity if one of the angles θ_i , with $i = 1, 2, 3$, approaches 0 or π , because in these cases the value of $\cot \theta_i$ becomes arbitrarily large. We infer that both small and large angles deteriorate the matrix conditioning. A similar argument holds for meshes of tetrahedra; more details and the explicit formula for the stiffness matrix of a tetrahedral cell when using the linear Galerkin finite element space can be found in [She02].

Conversely, suppose that there are no badly shaped triangles in

the mesh, so that for every triangle E , we can bound $\lambda_{max}(K^E)$ from below by some small constant. In this case, we can bound $\lambda_{max}(K)$ from below by a small constant, and note that the lower bound on the smallest global eigenvalue $\lambda_{min}(K)$ is proportional to the area A_{min} of the smallest triangle, and $\kappa = \mathcal{O}(1/A_{min})$. If the triangles are of uniform size, it follows that $\kappa(K)$ is proportional to ℓ^{-2} , where we denote by ℓ the typical edge length. This deterioration of the conditioning of the matrix is, however, unavoidable, and shows that, due to the finite precision of the arithmetics, reducing the mesh size does not necessarily lead to an improvement of the numerical accuracy.

Following similar ideas, based on explicit formulas for the stiffness matrix and their eigenvalues, several authors propose mesh quality indicators for the finite element interpolation, cf. [DWZ09].

4. Element Indicators

Element quality indicators are defined element-wise and then collected into a single mesh quality score in some kind of average. As a consequence, they are often specific to particular types of elements. In this section, we separately analyze indicators defined for the following types of elements:

- *Simplicial elements (triangles and tets)*: triangular elements in a 2D-mesh with 3 nodes and 3 edges; tetrahedral elements in a 3D-mesh with 4 nodes, 6 edges, and 4 triangular faces;
- *Quads and hexes*: quadrangular elements in a 2D-mesh with 4 nodes and 4 edges; hexahedral elements in a 3D-mesh with 8 nodes, 12 edges, and 6 quadrangular faces;
- *Polytopes*: polygonal elements in a 2D-mesh that are not triangles nor quads; polyhedral elements in a 3D-mesh that are not tets nor hexes.

While a common requirement is that all mesh elements are non-degenerate (i.e., they have a positive area or volume), different numerical schemes may demand the fulfillment of additional requirements. For each type of element, we further distinguish between indicators that measure the interpolation error and those that measure the discretization error. The firsts are derived straightforwardly from error bounds or by computing some sort of distance from an ideal element (e.g., the equilateral triangle or the square). They typically involve geometric quantities like edges, areas, volumes, or diameters. The latters are derived from information about the stiffness matrix and therefore are formulated in terms of condition numbers, eigenvalues, or matrix norms.

4.1. Simplicial Elements

We start our analysis from simplicial elements because several indicators are firstly defined over them and then extended to quads/hexes or polytopes. The quality of a simplex is typically intended as its deviation from an equilateral triangle or tetrahedron (at least for isotropic problems). Pathological triangular elements can be either “needles” with one and only one angle close to 0, see Figure 6(a), or “flat elements” with one angle close to π , see Figure 6(b). The major difference between a triangle and a tet is that a new entity comes to play in the latter, namely, the solid angle. Pathological tetrahedra can be “needles” with one and only one

solid angle close to 0, see Figure 6(c), “wedges” with one face with a very small area, see Figure 6(d), or “slivers” with a vertex almost on the plane containing its opposite face, see Figure 6(e).

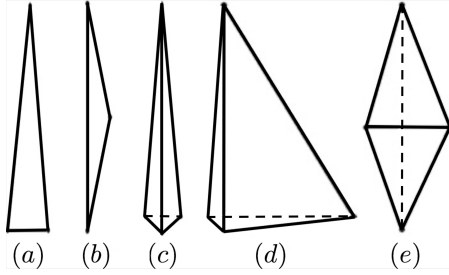


Figure 6: Examples of poorly shaped triangles and tetrahedra: (a,c) needles, (b) flat element, (d) wedge, (e) sliver.

Interpolation The simplest indicators for triangles are provided by the ratio between the inradius r and the circumradius R of the element, also called *radius* or *aspect ratio*, or the ratio between the longest edge $h_{e_{\max}}$ and the inradius:

$$\varepsilon_1 = \frac{r}{R}, \quad \varepsilon_2 = \frac{r}{h_{e_{\max}}}, \quad \varepsilon_3 = \frac{h_{e_{\max}}}{R}. \quad (6)$$

A deep analysis of these indicators is provided by [PB03], where the authors study their extremum values and asymptotic behaviors. A direct formula for computing ε_3 for a tetrahedron in terms of edge lengths and face areas is provided in [LJ94]. The edge ratio instead, intended as the ratio between the shortest and the longest edge, is not considered an interesting quality indicator as it does not vanish over flat elements [ADL*98].

Equally important is the angle regularity, which imposes that the internal angles are not too different from each other. For triangles it can be devised by measuring the sinus of the smallest angle θ_{\min} or the widest angle θ_{\max} , or a ratio between these two:

$$\vartheta_1 = \sin(\theta_{\min}), \quad \vartheta_2 = \sin(\theta_{\max}), \quad \vartheta_3 = \frac{\vartheta_1}{\vartheta_2}. \quad (7)$$

Several variations of the above indicators are overviewed in [Fie00] and [SEK*07]. The computation of the minimum solid angle of a tetrahedron is not intuitive; a widely used formula is the one introduced in [LJ94]:

$$\vartheta_4 = \min_{i=1,\dots,4} \{\sin(\theta_i/2)\}, \quad \text{where} \quad (8)$$

$$\sin(\theta_i/2) = \frac{12V}{\sqrt{\prod_{j,k \neq i, 1 \leq j < k \leq 4} [(h_{ejj} + h_{eik})^2 - h_{e_{jk}}^2]}}.$$

The minimum dihedral angle can be measured as well, but it cannot detect the degenerate case of a needle-shaped tet [ADL*98]. In fact, the dihedral angles of a tetrahedron remain more or less the same as the triangular face opposite to the pointed node is getting smaller and smaller. As already noted in Section 3.3, big angles are deleterious to the interpolation accuracy and the discretization error, while small angles may also be deleterious to the condition number of the stiffness matrix in finite element simulations [DWZ09].

Starting from tight error bounds for the interpolated function and

its gradient, accurate quality indicators are derived in [She02], for triangles:

$$\rho_1 = \frac{A}{R^2}, \quad \rho_1^\nabla = \frac{A}{(h_{e_1}h_{e_2}h_{e_3})^{2/3}}, \quad (9)$$

and for tets:

$$\rho_2 = \frac{V}{R^3}, \quad \rho_2^\nabla = \frac{V \sum_{m=1}^4 A_m}{\sum_{1 \leq i < j \leq 4} A_i A_j h_{e_{ij}}}, \quad (10)$$

where the index ∇ indicates the quality indicator relative to the quantity $\|\nabla f - \nabla g\|_\infty$. We only report here the smooth and scale-invariant versions of such measures, referring to the original paper for a more detailed analysis.

Discretization Indicators based on the Jacobian matrix have been identified as the key measures relevant in the context of FEM [EE20] and can be also used for curvilinear elements [MC21]. The algebraic shape indicator from [LJ94] is expressed as the ratio of the geometric mean to the arithmetic mean of the eigenvalues λ_i of $\mathcal{A}^T \mathcal{A}$:

$$\zeta_1 = \frac{3\mathcal{J}^{2/3}}{\text{trace}(\mathcal{A}^T \mathcal{A})} = \frac{3\sqrt[3]{\lambda_1 \lambda_2 \lambda_3}}{\lambda_1 + \lambda_2 + \lambda_3} \quad (11)$$

For this reason, it can be found under the name of *mean ratio* [ADL*98]. An equivalent version is the *Frobenius ratio*, which consists of the inverse of the condition number of matrix \mathcal{A} in the Frobenius norm [Knu01]:

$$\zeta_2 = \frac{d}{\kappa(\mathcal{A})} = \frac{d}{|\mathcal{A}|_F |\mathcal{A}^{-1}|_F}, \quad |\mathcal{A}|_F = \sqrt{\text{trace}(\mathcal{A}^T \mathcal{A})} \quad (12)$$

where d is the size of the matrix, i.e., the space dimension. The same indicator can be defined, with analog results, using any other type of matrix norm; see [PB03] for a comparison. Other specific shape quality indicators for triangles and tetrahedra based on the jacobian matrix are given in [Knu03] under the formulas

$$\zeta_4^{2D} = \frac{\sqrt{3}\mathcal{J}}{\lambda_{11} + \lambda_{22} - \lambda_{12}} \quad (13)$$

$$\zeta_4^{3D} = \frac{3\sqrt[3]{2}\mathcal{J}^{2/3}}{\frac{3}{2}(\lambda_{11} + \lambda_{22} + \lambda_{33}) - (\lambda_{12} + \lambda_{23} + \lambda_{13})}.$$

In [She02], two precise quality indicators are derived from tight bounds on the minimum and maximum eigenvalues of the stiffness matrix, respectively for triangles and tets:

$$\zeta_3^{2D} = \frac{3A}{\sum_{i=1}^3 h_{e_i}^2}, \quad \zeta_3^{3D} = \frac{4V}{\left(\sum_{i=1}^4 A_i^2\right)^{3/4}}. \quad (14)$$

Similar indicators have been found in other works: in [BX96] the quality of a triangle is set as $4\sqrt{3}A/(h_{e_1}^2 + h_{e_2}^2 + h_{e_3}^2)$, while [SP92] opts for the formulation $12\sqrt{3}A/(h_{e_1} + h_{e_2} + h_{e_3})^2$. Analogously, the quality of a tet is measured in [Lo97] by the formula $12\sqrt{3}V/(h_{e_1}^2 + h_{e_2}^2 + h_{e_3}^2)^{3/2}$. In [DWZ09] a bound on the determinant of the stiffness matrix in the more general setting of general finite element spaces and general model equations is given by $\sum_{i=1}^{n+1} A_i^2/V$.

4.2. Quads and Hexes

While quads may be more efficient than triangles from a computational point of view, their properties make them a more difficult primitive to handle compared to triangles. In addition to edge lengths, quadrilaterals have one (and only one) more degree of freedom, provided by any of their four angles [Péb04]. This extra degree of freedom gives birth to many pathological configurations, and even simple operations present a greater challenge. In Figure 7 we introduce some quantities specific to quadrilateral elements or faces. The *normal vector* associated to each vertex v_i is defined as:

$$\mathbf{n}_i = \frac{e_i \times e_{i+1}}{\|e_i \times e_{i+1}\|},$$

while the two *principal axes* of the quad are given by:

$$\mathbf{x}_1 = (\mathbf{v}_2 - \mathbf{v}_1) + (\mathbf{v}_3 - \mathbf{v}_4), \quad \mathbf{x}_2 = (\mathbf{v}_3 - \mathbf{v}_2) + (\mathbf{v}_4 - \mathbf{v}_1).$$

If the quad is not planar, the principal axes individuate a *projection plane* π . The *cross derivatives* of the map from parametric to world space are oriented along

$$\mathbf{x}_{12} = (\mathbf{v}_1 - \mathbf{v}_2) + (\mathbf{v}_3 - \mathbf{v}_4) = (\mathbf{v}_1 - \mathbf{v}_4) + (\mathbf{v}_3 - \mathbf{v}_2) = \mathbf{x}_{21}.$$

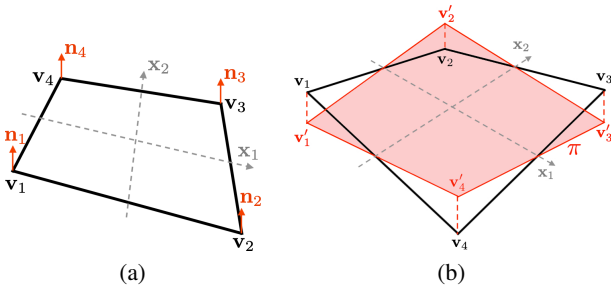


Figure 7: Elements of a quadrangle: (a) normal vectors \mathbf{n}_i and principal axes \mathbf{x}_i ; (b) projection plane π defined by the principal axes.

First of all, a quad (intended as an element itself, or as a face of a hexahedral element) is not necessarily flat, and even planar quads may be non-convex. This can be particularly crucial in architectural applications, where quad-meshes featuring this property are often categorized as *Planar-Quad* [BLP*13b]. Planarity and convexity can be simply imposed as prerequisites for all the elements in a mesh, or be the subject of some quality indicators. Robinson’s pioneering work [Rob87a] individuates four shape parameters that define a planar quadrilateral, shown in Figure 8(a-c): the *aspect ratio*, the *skew* angle and two *tapers*. If the quad is not planar we can add a fifth parameter, namely *warpage*, to measure planarity (Figure 8(d)).

Interpolation The *warping* of a quad measures its distance from being planar. Given the projection plane π_Q defined by the principal axes, the warping of Q is measured by the distance between its vertices $\mathbf{v}_1, \dots, \mathbf{v}_4$ and their projections $\mathbf{v}'_1, \dots, \mathbf{v}'_4$ onto π_Q , see Figure 7(b):

$$\varphi_1 = \max_{i=1, \dots, 4} \left\{ \sin^{-1} \left(\frac{\|\mathbf{v}_i - \mathbf{v}'_i\|}{l} \right) \right\}, \quad (15)$$

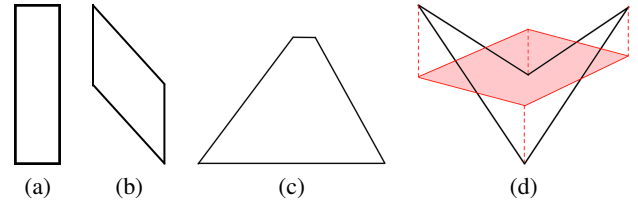


Figure 8: Quality of a quadrangular element: (a) low aspect ratio, (b) skewed element, (c) element with a significant taper along one direction

, (d) warped element.

where l is the half-length of the shortest between the two edges around v_i [Rob87b]. In the simplest sense, $\|\mathbf{v}_i - \mathbf{v}'_i\|$ is already an estimate of the warpage [Rob94]. Alternatively, warping is measured in [SEK*07] through the curvatures of the coordinate surface on which the quad lies:

$$\varphi_2 = 1 - \min \left\{ (\mathbf{n}_1 \cdot \mathbf{n}_3)^3, (\mathbf{n}_2 \cdot \mathbf{n}_4)^3 \right\}. \quad (16)$$

Other formulations involve the coefficients of the second fundamental form or the mean and Gaussian curvatures of a coordinate surface containing the quad [Lis17].

A formula for the *aspect ratio* of a quad can be derived from Indicator (6) for triangles. The opposite sides of each quad should have approximately equal lengths (this can be particularly crucial in contexts like physical simulations), or, for anisotropic approximation, a ratio best for approximation quality. However, the simple ratio between edges is not able to capture all kinds of pathologies, for instance, skewness. Another formula specific to quadrilateral quality measure has been proposed by [Rob87b] and further optimized in [Fie00]:

$$\varepsilon_1 = \max \left\{ \frac{\|\mathbf{x}_1\|}{\|\mathbf{x}_2\|}, \frac{\|\mathbf{x}_2\|}{\|\mathbf{x}_1\|} \right\}, \quad (17)$$

while the quality measured by [FG00] involves the quad diagonals d_1, d_2 and the areas A_i^t of the triangles formed by the two edges incident to the i -th vertex:

$$\varepsilon_2 = \frac{\sqrt{\sum_{i=1}^4 h_{ei}^2} \max\{h_{\max}, d_1, d_2\}}{\min_i\{A_i^t\}}. \quad (18)$$

In [Péb04] we find a specific formula to measure the aspect regularity, intended as the departure of the cell from a rhombus:

$$\varepsilon_3 = \frac{1}{1 - \left| 1 - \frac{2\alpha}{\pi} \right|}, \quad (19)$$

where α is the average between two opposite angles, no matter which. A similar *stretch* indicator is given by the ratio between the minimum edge and the maximum diagonal [SEK*07]. In [Lis17] the stretch is also formulated in terms of the covariant metric tensor.

Regarding the angle regularity, or *skewness*, internal angles should be close to 90 degrees, therefore we need to characterize the departure of a quad from a rectangle. This is measured by the absolute value of the cosine of the angle between the principal

axes [Rob87b]:

$$\vartheta_1 = \left| \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|} \cdot \frac{\mathbf{x}_2}{\|\mathbf{x}_2\|} \right|. \quad (20)$$

It can also be described through the cosine (or the cotangent) squared of the angle between the two tangent vectors defining the quad [Lis17]. As pointed out in [Knu03], such skew indicators reach their maximum also over elements that are not rectangles. To avoid this issue, the alternative indicator is proposed:

$$\vartheta_2 = \frac{4}{\sum_{i=1}^4 1/\sin(\theta_i)}. \quad (21)$$

A further quantity that can be measured in a quad is the so-called *taper*, conceived as the difference in length between opposite edges. The tapers relative to the two couples of edges in a quad are conventionally called the *x*- and *y*-component [Rob87a]. It can be quantified by the maximum ratio of cross derivative magnitude to principal axis magnitude [SEK*07]:

$$\tau = \frac{\|\mathbf{x}_{12}\|}{\min\{\|\mathbf{x}_1\|, \|\mathbf{x}_2\|\}}. \quad (22)$$

The quality indicators defined for quadrangular elements are straightforwardly extended to hexahedral cells. In [Rob94] Robinson defines three skew ratios, three tapers, and six warpages for a hexahedral element, with the same procedure used for quads. Warping, stretching, skewness, and aspect regularity of a hex can be defined by the sum, the average, or the minimum/maximum of the respective indicators on its faces [Lis17, SEK*07].

Discretization Robinson’s efforts amply demonstrate that quadrilaterals present much more complex geometric shapes to assess than triangles. This complexity perhaps motivates the predominant practice of using Jacobian determinants rather than purely geometric indicators [Fie00].

For the *shape regularity*, we already saw how the Jacobian can be simply extended by considering the minimum between the values of \mathcal{J} at the vertices and at the center of the quad. In this sense, it can be used in the same formulas as for triangles. In [Knu01] we have the equivalent of Indicator (12) for quads ($n = 4$) and hexes ($n = 8$), together with an alternative formulation ζ_2 :

$$\zeta_1 = \min_{i=1, \dots, n-1} \left\{ \frac{d}{\kappa(A_i)} \right\}, \quad \zeta_2 = \frac{9n}{\sum_{i=1}^{n-1} \kappa(A_i)^2}. \quad (23)$$

More advanced Jacobian-based indicators exist under the names of *weighted* or *scaled* Jacobian [Knu00b, Knu00a]:

$$\zeta_3 = \frac{\min_i \{\mathcal{J}_i\}}{\max_i \{\mathcal{J}_i\}}, \quad \zeta_4 = \min_i \left\{ \frac{\mathcal{J}_i}{h_{e_{i-1}} h_{e_i}} \right\}, \quad (24)$$

or the ones proposed in [Knu03] for quads and hexes:

$$\zeta_5^{2D} = \frac{8}{\sum_{i=1}^4 (\lambda_{11}^i + \lambda_{22}^i) / \mathcal{J}_i}, \quad (25)$$

$$\zeta_5^{3D} = \frac{24}{\sum_{i=1}^8 (\lambda_{11}^i + \lambda_{22}^i + \lambda_{33}^i) / \mathcal{J}_i^{2/3}}. \quad (26)$$

Such indicators provide a measure of the variation in the Jacobian across the element, and they return a general measure of the quad shape.

Another diffused shape indicator is the *Oddy metric* [OGMB88], which measures the maximum deviation of the metric tensor at the corners of the quad. It can be defined in terms of the Jacobian matrices, or through the more explicit formula given in [SEK*07] for quads (and then extended to hexes):

$$\omega = \max_{i=1, \dots, 4} \left\{ \frac{(h_{e_i}^2 - h_{e_{i+1}}^2)^2 + 4(h_{e_i} h_{e_{i+1}})^2}{2\|e_{i-1} \times e_i\|^2} \right\}. \quad (27)$$

4.3. Polytopes

The concept of the geometric quality of a cell becomes quite vague when the cell is a generic polygon or polyhedron. Most of the quality indicators defined above become meaningless when the elements have a generic number of vertices or faces, or they are at least very difficult to extend. For instance, the Jacobian can be extended to generic convex polygons (similarly to how we did for quads), but the Jacobian matrix in a polyhedron vertex is defined only if such vertex has three incident edges (e.g., a prism is fine, but a pyramid is not), and in general it is not defined for non-convex polytopes [Knu01]. Trying to delimit the field by excluding unlikely configurations, most mesh generators, and most numerical schemes only allow convex or star-shaped elements. On the other side, indicators defined for generic polygons can also be used to analyze the quality of triangular and quadrangular elements, being triangles and quads particular cases of polygons, and the same holds for polyhedra and tets/hexes.

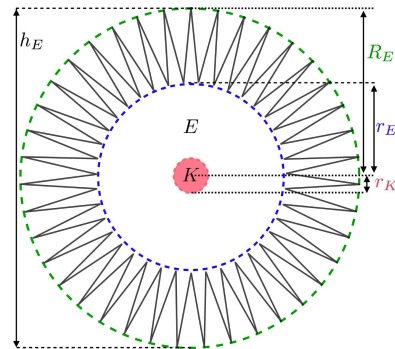


Figure 9: Polygonal element E with diameter h_E , inradius r_E , and circumradius R_E . Its kernel K has kernel radius r_K .

Interpolation The concept of *warping* can be extended naturally to polytopes. Indicator (15) remains valid, but computing the plane on which projecting the polygon (or the polygonal faces) is less immediate when the polygon is not a quad. Indicator (16) instead, can be extended to polygons by adding a proper number of pairs of normal versors \mathbf{n}_i .

The *aspect ratios* for polytopes are typically derived from Indicator (6), using the inradius r , the circumradius R , and opportunely replacing the maximum edge lengths with the diameter h_E [GRB12], see Figure 9:

$$\epsilon_1 = \frac{r}{R}, \quad \epsilon_2 = \frac{r}{h_E}, \quad \epsilon_3 = \frac{h_E}{R}. \quad (28)$$

A number of variations of the above indicators (only for polygons) are overviewed in [ABB*21], where the authors analyze the impact of the different indicators on the performance of the Virtual Element Method. Alternatively, it is always possible to decompose a polytope into triangular or tetrahedral elements obtained connecting its vertices (generating a “matching simplicial submesh” [DPD19]) and apply the simplicial shape ratio indicators.

The angle regularity of a generic polygon can be inferred by any of the Indicators (7), while the formula for the solid angle given in Indicator (8) is peculiar to tetrahedral elements.

If we allow for the presence of non-convex polytopes, the ratio between the area and the perimeter $2\pi A_E/p_E^2$ (or the volume and the surface area) becomes much more interesting than it was for simplicial elements [ABB*21]. A number of indicators involve the kernel K of the element:

$$\varphi_1 = \frac{A_K}{A_E}, \quad \varphi_1 = \frac{V_K}{V_E}, \quad (29)$$

which returns 1 for convex polytopes, a value in $(0, 1)$ for concave polytopes, and 0 for star-shaped polytopes. It is also possible to measure the ratio between the inradius r_K of the circle inscribed in K and the circumradius R_E , see Figure 9.

Some works on polygonal element methods also require that the elements in the mesh have no “short edges”, meaning that the ratio between the diameter and the shortest edge of the element is uniformly bounded across the mesh [BadVBC*13, AAB*13, Bd-VLR17, BGS17]. Recently, researchers noticed that the “no short edge” requirement can be dropped in the case of Virtual Element Methods [BS18, BadVV20]. But for other numerical methods, the situation is not completely clear and no short edges are still required at least in some theoretical analyses [GRB12]. This, therefore, arises as a sufficient condition for the good behavior of the numerical method, but it can be also intended as a quality indicator. This is the approach of [SPC*22, SBMS22b], where the authors propose a mesh quality indicator specifically designed to work in couple with the Virtual Element Method. It is derived from the four geometrical assumptions required on a mesh for the convergence of the VEM:

$$\begin{aligned} \rho_1 &= A_K/A_E, & \rho_2 &= \min\{\sqrt{A_E}, h_{e \min}\}/h_E \\ \rho_3 &= 3/\#\{e \in \partial E\}, & \rho_4 &= \min_j \frac{\min_{e \in \mathcal{I}_E^j} h_e}{\max_{e \in \mathcal{I}_E^j} h_e}, \end{aligned}$$

and it is formulated as

$$\rho = \sqrt{\frac{\rho_1 \rho_2 + \rho_1 \rho_3 + \rho_1 \rho_4}{3}}. \quad (30)$$

Its 3D version is derived straightforwardly in [SBMS22a], by simply extending the indicators to volumetric elements.

Numerous attempts have been made to measure the quality of a polygon or polyhedron as “how far it is from a regular n -gon or n -hedron” [CHSS13, Cox38, ZR04] because a unitary regular n -gon (or n -hedron) is usually considered to be of high quality in the Euclidean metric. This implies, for instance, considering the ratio between the area (volume) of an element P and the area (volume) of a regular n -gon (n -hedron) with the same perimeter (surface area) as P , or the difference between the angles of P and those of the ideal

element. Unlike triangles, however, n -gons with $n > 3$ generally are not affine similar to a single reference n -gon [HW20]. Hence the idea of measuring the quality of a polygonal mesh by comparing its elements to regular n -gons through affine mappings does not work in general. To make it work, one needs to build proper mappings connecting arbitrary polygons with the reference ones, or re-define reference polygons of “good quality”, or do both. However, polytopal element methods are defined to work over extremely generic elements, therefore there is apparently no reason to believe that the quality of a regular n -gon is the highest possible, or that they are the only configurations able to reach that quality level.

Discretization As already noted, the Jacobian matrix in a polyhedron vertex is defined only if such vertex has three incident edges. An exception can be made for the particular case of pyramidal elements, where the only vertex with valence higher than three is the top one. As a pyramid is completely defined by the tetrahedra associated with its base vertices, it is possible to define the Jacobian as an average of the Jacobians computed at the base vertices, which are all well-defined. It is therefore possible to compute Indicator (11) or (12) over “hybrid” meshes, that are mostly tetrahedral or hexahedral, with a small number of pyramids or prisms/wedges (whose vertices have all three incident edges) as transition elements to connect triangular and quadrilateral element faces [VW12, KC00, LADH22].

The *Target-matrix paradigm* is a generalization of this Jacobian-based algebraic approach to mesh quality [Knu07, Knu12]. The Jacobian matrices A_k are computed on a set of sample points and compared to the relative “Target” matrices W_k , which provide the local definition of quality. Its generality stems from the fact that the sample points can be defined for any kind of mesh and the quality indicator can be customized to match users’ requirements. In the easiest case, the sample points coincide with the vertices of a simplicial mesh, and the indicator is built on the matrix $T = AW^{-1}$ as Indicator (12):

$$\zeta = \frac{d}{\kappa(T)} = \frac{d}{|T|_F |T^{-1}|_F}. \quad (31)$$

A similar approach is adopted in [SDVG10], encoding the desired target size and shape of each element into a Riemannian metric instead of a matrix.

Another common type of polytopal meshes is Centroidal Voronoi Tessellations (CVT), which contain convex Voronoi cells with an undefined number of edges and faces. The quality of a CVT is often measured through some ad-hoc *CVT energy*. Given a CVT with centroids \mathbf{x}_i , $i = 1, \dots, n$, defined over a domain $\Omega = \bigcup_{i=1}^n \Omega_i$ endowed with a density function $\rho(\mathbf{x}) > 0$, its global energy is defined as [DFG99a]:

$$\sum_{i=1}^n \int_{\Omega_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 d\sigma.$$

The CVT energy accounts for the compactness of the cells, but it depends both on the number of cells and on the size of the shape. To overcome this problem, in [Wan17] the author proposes to define a local quality indicator derived from the dimensionless second moment (or variance) of the single polytope, which measures how

far the points inside E are spread out relative to its centroid \bar{x} :

$$\vartheta = \frac{\int_E \|x - \bar{x}\|^2 dx}{d \left(\int_E dx\right)^{(d+2)/d}}. \quad (32)$$

In this approach, the optimal element turns out to be the hexagon in 2D and the truncated octahedron in 3D.

4.4. Discussion

In Table 1 and Table 2 we summarize the main quality indicator found for 2D and 3D elements, respectively. The tables are not exhaustive: we reported a representative selection of the vast amount of the existing variants for each indicator. Moreover, we occasionally had to “force” the notation in favor of readability. In particular, the rows relative to “interpolation quality” and “shape regularity” contain all the indicators from the Interpolation and Discretization paragraphs (respectively) that did not fit into any other category.

We briefly comment on the two tables, also with respect to the considerations made in Section 2.4. Despite the name with which the indicators have been originally introduced, few of them are actually measures or metrics. For instance, they take values in ranges different from $[0, 1]$, but a deeper analysis of this aspect is beyond the scope of this work. However, according to the results presented in the respective papers, this does not prevent them to be accurate and reliable in assessing the quality of an element. Indicators reported in the rightmost columns are the most general: they are defined over any type of cell and they are often very similar between their 2D and 3D formulation. It is also notable how almost all the quality indicators for hexes are simply defined as a function of the quads quality indicators applied to their faces, exceptions made for those based on the Jacobian.

Many indicators have been shown to be equivalent when applied to the same type of elements. For instance, [LJ94] proved that radius ratio, mean ratio, and sine of solid angle are equivalent over tets, meaning that they perform similarly well (or bad) over well (or bad) shaped elements. It has also been shown how the mean ratio performs equivalently to the Frobenius ratio [Knu01], which in its turn is equivalent to the same ratio using any other matrix norm [PB03]. This is the reason why they belong to the same row in our tables. Chances are that more equivalences can be proven among other indicators, taking advantage of formulas that allow expressing algebraic quantities in terms of geometric ones and vice versa. Excluded from this point are case-specific indicators like the CVT energy or the VEM quality indicator.

The computational efforts required for some of the presented indicators are discussed in [PGH94]. There exist a few works that compare the impact of the quality indicators on various steps of the simulation process, to evaluate how much the a-priori indicators actually correlate with the eventually observed performance of a mesh. A comparison of the performance of some specific quality indicators for hexahedral meshes in terms of stability and accuracy is given in [GHX*17], with both application-dependent and independent studies. The PEmesh graphical framework [CPS22] allows the study of the correlations between the geometric properties of polygonal meshes and the numerical performances of PEM solvers. The user can design sets of polygonal meshes that increasingly stress

some geometric properties and study the correlation between the performances of an external PEM solver and the geometric properties of the input mesh. More details on available software and datasets are provided in Section 7.

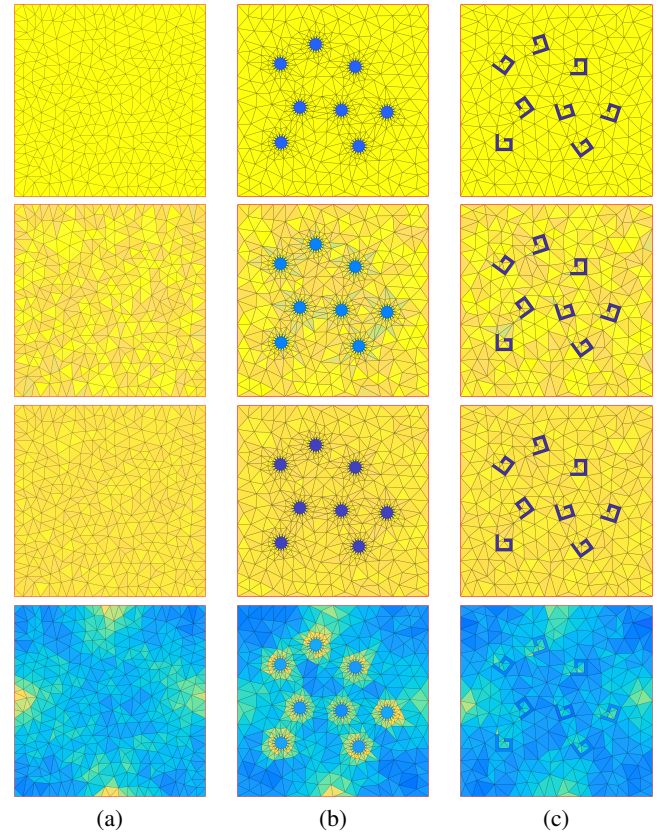


Figure 10: Comparison between different quality indicators over three polygonal meshes (a-c): each element is colored with respect to its quality from blue (low) to yellow (high). From top to bottom row: kernel-area ratio; angle ratio; VEM quality indicator; H^1 norm of the approximation error produced by the linear VEM.

In Figure 10 we present a visual comparison of the differences between some quality indicators, and their relationship with the errors produced in a numerical simulation. In column (a) we consider a mesh containing only triangular elements, in (b) we introduce some star-shaped but non-convex polygons, and in (c) some non-star-shaped ones. In the top three rows, each element is colored with respect to its quality according to, respectively, the kernel-area ratio φ_1 from (29), the angle ratio ϑ_3 from (7), and the VEM quality indicator ρ from (30). Such indicators were chosen as representatives of the aspect ratio, skewness, and interpolation quality classes in Table 1. All indicators are re-scaled in the range $[0, 1]$, value 0 (low quality) is associated with the color blue, and value 1 (high quality) with the color yellow. In the bottom row, we color the elements with respect to the H^1 error produced by the solution of a Poisson problem with homogeneous Dirichlet boundary conditions through the linear VEM (the L^2 and the L^∞ errors produce very

Table 1: Taxonomy of quality indicators for 2D elements.

	triangles	quads	polygons
warping	-	$1 - \min \left\{ (\mathbf{n}_1 \cdot \mathbf{n}_3)^3, (\mathbf{n}_2 \cdot \mathbf{n}_4)^3 \right\}$	$\max_{i=1, \dots, n} \left\{ \sin^{-1} \left(\frac{\ \mathbf{v}_i - \mathbf{v}'_i\ }{l} \right) \right\}$
aspect ratio	$\frac{r}{R}, \frac{r}{h_{e\max}}, \frac{h_{e\max}}{R}$	$\max \left\{ \frac{\mathbf{x}_1}{\mathbf{x}_2}, \frac{\mathbf{x}_2}{\mathbf{x}_1} \right\}, \left(1 - \left 1 - \frac{2\alpha}{\pi} \right \right)^{-1}$	$\frac{r}{R}, \frac{r}{h_E}, \frac{h_E}{R}, \frac{r_K}{r_E}, \frac{A_K}{A_E}$
skewness	$\sin(\theta_{\min}), \sin(\theta_{\max}), \frac{\sin(\theta_{\min})}{\sin(\theta_{\max})}$	$\left \frac{\mathbf{x}_1}{\ \mathbf{x}_1\ } \cdot \frac{\mathbf{x}_2}{\ \mathbf{x}_2\ } \right , \frac{4}{\sum_{i=1}^4 1/\sin(\theta_i)}$	$\sin(\theta_{\min}), \sin(\theta_{\max}), \frac{\sin(\theta_{\min})}{\sin(\theta_{\max})}$
taper	-	$\frac{\ \mathbf{x}_{12}\ }{\min\{\ \mathbf{x}_1\ , \ \mathbf{x}_2\ \}}$	-
interpolation quality	$\frac{A}{R^2}, \frac{A}{(h_{e1}h_{e2}h_{e3})^{2/3}}$	$\max_{i=1, \dots, 4} \left\{ \frac{(h_{ei}^2 - h_{ei+1}^2)^2 + 4(h_{ei}h_{ei+1})^2}{2\ e_{i-1} \times e_i\ ^2} \right\}$	$\sqrt{\frac{\rho_1\rho_2 + \rho_1\rho_3 + \rho_1\rho_4}{3}}$
mean ratio	$\frac{\mathcal{J}}{\text{trace}(\mathcal{A}^T \mathcal{A})}, \frac{2}{\kappa(\mathcal{A})}$	$\min_{i=1, \dots, 4} \left\{ \frac{2}{\kappa(A_i)} \right\}, \frac{36}{\sum_{i=1}^4 \kappa(A_i)^2}$	$\min_k \left\{ \frac{2}{\kappa(T_k)} \right\}$
shape regularity	$\frac{3A}{\sum_{i=1}^3 h_{ei}^2}, \frac{\sqrt{3}\mathcal{J}}{\lambda_{11} + \lambda_{22} - \lambda_{12}}$	$\frac{\min_i\{\mathcal{J}_i\}}{\max_i\{\mathcal{J}_i\}}, \frac{8}{\sum_{i=1}^4 (\lambda_{11}^i + \lambda_{22}^i)/\mathcal{J}_i}$	$\frac{\int_E \ x - \tilde{x}\ ^2 dx}{2(\int_E dx)^2}$

similar results):

$$\varepsilon(E) = -\log \frac{|u - u_h|_{1,E}}{|u|_{1,\Omega}}, \quad \forall E \in \Omega_h,$$

where the negative sign is introduced so that high error values correspond in color to low-quality elements. Moreover, ε values are re-scaled in the range $[\min_{E \in \Omega_h} \varepsilon(E), \max_{E \in \Omega_h} \varepsilon(E)]$ in order to highlight differences between the elements.

We can immediately observe how all the considered quality indicators perfectly "recognize" the pathological elements in (a) and (b), assigning them a deep blue color. Regarding the triangular elements, the kernel-area ratio holds 1 for all the triangles in all three meshes, while the angle ratio and the VEM indicator exhibit some differences among elements, more accentuated in the first case. The errors in the bottom row present similar color patterns, which depend on the function we are approximating (in this case, a sinusoidal function). Besides this, it is still appreciable how poor-quality elements produce higher errors than their neighbors, highlighting a correlation between the quality indicators and the numerical performance.

5. Mesh Indicators

Mesh quality indicators observe the features of the mesh from a more general perspective. In addition to all the possible functions that we can define collecting element indicators in some sort of average or weighted norm across all the cells, there are indicators related to features that cannot be measured element-wise. In this section, we discuss possible strategies that are in the literature to define global quality indicators. Without providing an exhaustive survey of all the existing papers, we rather give a general taste of the aspects of a mesh that may affect its quality. We refer the interested reader to the original cited literature for the details and the precise formulation of such indicators.

5.1. Consistency

We can talk about different types of consistency [Lis06]. A mesh is considered consistent with the *geometry* of the domain if it faithfully represents its boundary and eventual sharp features. It is con-

sistent with the *topology* of the domain if the numbers of holes and connected components are preserved. It is consistent with the *solution* of the physical problem if it reflects the features of the function to be approximated.

For the accurate computation of the numerical solution of a PDE defined over the domain, the requirement of consistency with the geometry is indispensable. The mesh nodes must adequately approximate the original geometry, that is, the distance between any point of the domain and the nearest mesh node must not be too large, and this distance must approach zero when the number of nodes tends to infinity. This is particularly important for nodes, edges, or faces representing the boundary of the physical domain, as it allows the boundary conditions to be applied more easily and accurately.

The concept of topology consistency traditionally refers to the connectedness properties, or which node belongs to which element. When dealing with topological consistency the location in space of each node is irrelevant, while it is crucial to have correct adjacency rules among mesh elements [Pre82]. In addition, the topology of the mesh must be coherent with that of the model. Besides simple consistency rules such as the Euler relation, the analysis of evolving surfaces, or model deformations, requires guaranteeing the consistency of the mesh with the model during the whole simulation process. This implies sampling a model consistently [CDRR04]; identifying small defects, holes, and handles [SH97, ACK13b]; removing the topological noise [WHDS04]; and even topologically replicating the boundaries during optimization [SA21].

Another type of consistency is the one with the solution to the physical problem. The distribution of the mesh nodes and the form of the mesh cells should be dependent on the features of the solution, such as preferred directions (e.g., streamlines or vector fields), localized regions of very rapid variations (i.e., regions of high gradients), boundary and interior layers (e.g., in fluid dynamics, combustion, solidification, solid mechanics, and wave propagation), areas of high solution error of the numerical approximation [Lis17]. In all these cases, it may be helpful to subdivide the domain into smaller parts and mesh it locally because the uniform refinement of the entire domain may be very costly for multidimensional computations. Features lines, when present, should be explicitly repre-

Table 2: Taxonomy of quality indicators for 3D elements.

	tetrahedra	hexahedra	polyhedra
warping	-	sum/average/min/max of the faces	$\max_{i=1,\dots,n} \left\{ \sin^{-1} \left(\frac{\ v_i - v_i'\ }{l} \right) \right\}$
aspect ratio	$\frac{r}{R}, \frac{r}{h_{e\max}}, \frac{h_{e\max}}{R}$	sum/average/min/max of the faces	$\frac{r}{R}, \frac{r}{h_E}, \frac{h_E}{R}, \frac{r_K}{r_E}, \frac{V_K}{V_E}$
skewness	$\min_{i=1,\dots,4} \{ \sin(\theta_i/2) \}$	sum/average/min/max of the faces	$\min_{i=1,\dots,n} \{ \sin(\theta_i/2) \}$
taper	-	sum/average/min/max of the faces	-
interpolation quality	$\frac{V}{R^3}, \frac{V \sum_{m=1}^4 A_m}{\sum_{1 \leq i < j \leq 4} A_i A_j h_{eij}}$	sum/average/min/max of the faces	$\sqrt{\frac{\rho_1 \rho_2 + \rho_1 \rho_3}{2}}$
mean ratio	$\frac{3\mathcal{J}^{2/3}}{\text{trace}(\mathcal{A}^T \mathcal{A})}, \frac{3}{\kappa(\mathcal{A})}$	$\min_{i=1,\dots,8} \left\{ \frac{3}{\kappa(A_i)} \right\}, \frac{72}{\sum_{i=1}^8 \kappa(A_i)^2}$	$\min_k \left\{ \frac{3}{\kappa(T_k)} \right\}$
shape regularity	$\frac{4V}{(\sum_{i=1}^4 A_i^2)^{3/4}}, \frac{3\sqrt[3]{2} \mathcal{J}^{2/3}}{\frac{3}{2}(\lambda_{11} + \lambda_{22} + \lambda_{33}) - (\lambda_{12} + \lambda_{23} + \lambda_{13})}$	$\frac{\min_i \{ \mathcal{J}_i \}}{\max_i \{ \mathcal{J}_i \}}, \frac{24}{\sum_{i=1}^8 (\lambda_{i1}^2 + \lambda_{i2}^2 + \lambda_{i3}^2) / \mathcal{J}_i^{2/3}}$	$\frac{\int_E \ x - \bar{x}\ ^2 dx}{3(\int_E dx)^{5/2}}$

sented as edge sequences. For example, sharp crease lines in mechanical objects, or lines where some attribute other than normals (e.g., color) varies. In this sense, the consistency with the solution is related to the consistency with the geometry: incorporating an input feature network into the mesh is not possible if the connectivity does not align with it, even refining the mesh [LPP*20] (see Figure 11).

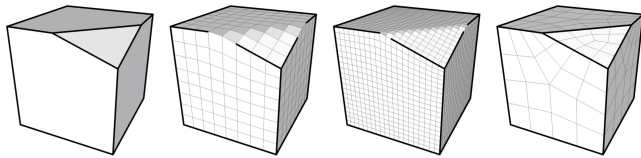


Figure 11: Key to feature preservation is the ability to align surface edges to the input network, carefully positioning mesh singularities (image from [LPP*20]).

Consistency indicators should measure the distance between the mesh and the domain, or the domain features. This is, for instance, a common theme in many mesh simplification strategies, where an explicit or implied geometric distance measure between the original and the simplified model is used to drive the simplification process [LT00]. A simple example of such measure is given by the Hausdorff distance [DD12] that was used to guarantee that the simplified model is less than a certain upper bound from the original model [RB93, KCS98]. Many other energy functions have been considered in the literature, as examples, we refer to [HDD*93, ZOG22] and many of the optimization methods listed in Section 6. Often, approaches based on this paradigm admit both modifications of the local geometry and topology of the mesh.

5.2. Structure

By the structure of a mesh, we mean the way its elements are connected to each other. A common classification of the mesh structures is the one originally developed for quad meshes [BLP*13b] and then extended to hex and hybrid meshes [PCS*22]. The authors call *regular* and *irregular* the structured and unstructured meshes already introduced in Section 2.1, and they also introduce two intermediate configurations. The *semi-regular* meshes are obtained

by gluing in a conforming way a number of regular blocks. All vertices that are internal to a block are regular, while vertices at the edges or corners of a block may be irregular. The *valence semi-regular* meshes contain a limited amount of irregular vertices, but they are not connected in a way that induces a coarse block decomposition into a few regular blocks. In some sense, irregular, valence semi-regular, semi-regular, and regular meshes can be seen as a continuum of cases, with an increasing degree of structure regularity. Depending on the applications, either a lower or higher degree of structure regularity is required, and this can be measured by the number of singular nodes in the mesh. A low number of singular nodes implies a simpler singular structure, which is more likely to allow for a block-structured mesh. Often, also the positioning of singular nodes is crucial: straight sequences of edges stemming from them (a.k.a. separatrices) should connect them in a graph that is as simple as possible, and they should appear in regions with a strong negative or positive Gaussian curvature (other than where it is needed to change resolution) [BLP*13b]. Other works weaken the concept of regular node, considering regular also nodes with valence close to the fixed value [AYZ12].

Mesh connectivity is typically assumed to be conforming (i.e., free from T-junctions, see Figure 2) and pure, see Section 2.1. T-junctions are problematic in a number of ways, for instance, they lead to configurations in which it is not trivial to enforce continuity of the finite element functions across the edges or faces. In this sense, quality indicators can be provided by the number of T-junctions, or the number of non-standard elements (for instance, the number of non-quad cells in a quad-dominant mesh). The two above requirements can sometimes be loosened [MPKZ10], but such topological freedom is not unlimited and may be bounded by the specific application [PCS*22].

5.3. Distribution

Since nodes and elements represent the only contact points between the real world and the approximated copy that is going to be reproduced through the mesh, the way they are distributed in the domain affects the quality of the final result. If a uniform distribution of the information over the domain is needed, the element size (area, volumes, or sum of the faces areas) has to be similar across the mesh. If the mesh is pure, it is possible to measure the balance (or con-

centration, or density) through simpler quantities like edge lengths or diameters. In particular, in structured meshes, the change of cell size in a certain direction or along a curve can become an indicator [Lis17].

However, it is often beneficial to let the tessellation density vary over the mesh, to adapt to local shape complexity, or to the solution of the numerical problem. In this case, the size of the elements cannot be required to be uniform across the mesh, but it is possible to ask at least for a smooth transition by imposing a gradual size change (neighboring elements cannot have too different sizes) [AC-SYD05], also called mesh gradation control [BHF98]. Moreover, in order to allow for spatial transition through different levels of resolution, extra singular nodes must be included, unless T-junctions are introduced. Therefore, in many contexts, it is desirable to achieve the right trade-off between adaptivity and connectivity.

If the physical problem itself is anisotropic, it requires mesh generation to be guided by a prescribed anisotropy field. For instance, in computational fluid dynamics, it is desirable to squeeze the elements in the direction normal to the wing of a plane since the most significant physics occurs in the limit layer [Rip92, HW20]. If the exact solution is of the type shown in Figure 12(a), a discretization like Figure 12(c) is likely to perform much better than the one in Figure 12(b), despite elements in the latter are more similar to regular polygons.

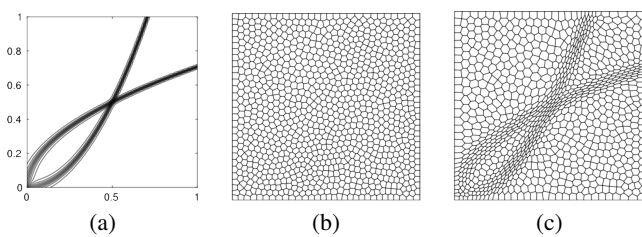


Figure 12: Example of anisotropic problem: (a) contour plot of the exact solution, (b,c) two possible discretizations [HW20].

6. Mesh Quality Optimization

The local and global quality indicators presented in Section 4 and Section 5 are commonly used to investigate and improve mesh quality. Some of them can be integrated into the mesh generation process (e.g., in advancing front methods [Lo13]), so that at the end of the pipeline, the output mesh already reaches a certain quality level. This is particularly true for 2D-meshes, and for indicators concerning properties like structure or connectivity, which may be essential requirements in some applications. However, the new placement of vertices, edges, and faces can be implicitly treated as an optimization problem. For instance, the edge collapse operation can be stated as the problem of choosing the position of the new vertex such that an associated objective function (sometimes called the “edge cost”) is minimized.

In this sense, in Computer Graphics there is a large literature of methods that set the problem of improving the quality of a mesh as an optimization problem with respect to a functional minimization. Among the others, we refer to the seminal work of Hoppe et

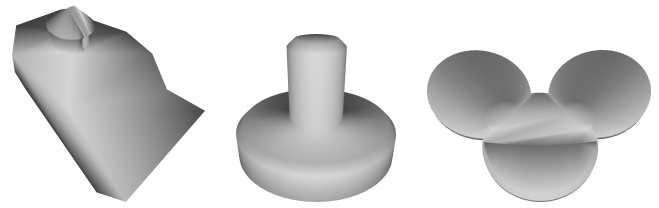


Figure 13: Impact of long and thin triangles on the model rendering (models taken from the Thingi dataset [ZJ16]). Artificial black parts are visible even in regions that are geometrically flat.

al [HDD*93]. Similarly, the majority of volumetric meshing algorithms, employ a two-step process [PCS*22]. The first step generates an initial mesh, which is expected to be dominated by well-shaped elements, but often also contains some poorly-shaped or even degenerate elements. This step is typically followed by an optimization step, whose goal is to maximize the quality of the mesh elements and remove any degeneracy while keeping the meshed domain boundary intact. As a mesh is defined by the nodal connections of the elements and the coordinates of the nodes, optimization methods can be broadly grouped into two categories [Lo14]: (i) geometrical methods that involve a change of element shapes by means of shifting of nodal points; and (ii) topological methods that involve a change in the element connections to the nodal points. However, there is no reason why topological and geometrical operations cannot be put together to form even more effective and well-balanced optimization schemes.

Broadly speaking, operations used for repairing and eliminating the noise and defects of a mesh are also included in the optimization of a mesh. In this paper we do not intend to go into detail about all processing techniques involved in mesh smoothing, repairing, etc.; for more details, the reader can consult [BKP*10, ACK13b] and the Eurographics tutorial [CAK12]. As a flavor of the relevance of mesh quality for practical applications, we mention rendering and manufacturing. For instance, the poor quality of some elements may generate a worsening of the 3D model rendering. Figure 13 shows the effect of some triangles with bad aspect ratio on the model rendering (the visualization is done with the standard per-vertex rendering available in the Meshlab suite [CCC*08]). On this type of models, even the application of a texture would require remeshing to avoid image distortion. In the case of 3D printing, STL mesh models with poor quality elements could make the printing software either fail or produce machine instructions that lead to printing failures [Att18]. In the left of Figure 14 we show a complex model originally with several heterogeneous defects; the distorted elements highlighted in the red box are then remeshed in the picture on the right, thus making the model printable.

6.1. Optimization by Geometric Operations

Improvement methods that keep the mesh connectivity fixed while changing only the locations of the mesh vertices are commonly referred to as *geometric optimization*, *smoothing*, or *untangling* methods [Owe98, SJ08]. The quality of a mesh can be improved by shifting the interior points within the domain boundary, and usually, boundary points have to be kept intact for compatibility with

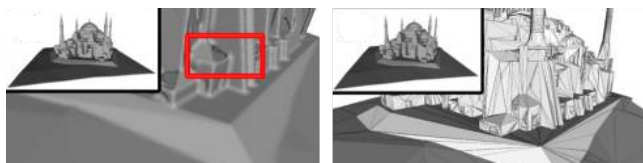


Figure 14: Left: a complex 3D model with some distorted elements (e.g., needles) highlighted in the red box. Right: after remeshing, the model can be printed. The original pictures are available in [Att18].

other meshes or for imposing the required boundary conditions for the finite element analysis.

Before the development of valid shape indicators, an early attempt to improve a triangular mesh was to shift each interior node in turn to the centroid of the polygon formed by all the nodes connected to it through an edge of the mesh. This iteration process is known as *Laplacian smoothing* [Her76], and such heuristic smoothing techniques have been successfully employed for tri and quad-meshes [EÜZ09, XN06]. Laplacian smoothing is computationally inexpensive and well-suited for meshes combining arbitrary element types. However, due to its quality-unaware node averaging scheme, Laplacian smoothing can lead to mesh quality deterioration and the generation of inverted elements.

Besides smoothing operations, vertex repositioning is able to address other needs such as equidistribution of volumes, shape improvement, or adaptive refinement [FDKMS06]. Based on a shape quality indicator, we can define a cost function over the mesh, which can be maximized as a local/global optimization problem to improve the overall mesh quality [FP00, SS09, Knu12]. In these works, the objective function is defined by measuring one or more mesh properties, such as in [KPS13], and it can be optimized by repositioning the vertices, leading to an improvement of the measured properties. The work in [FDKMS06] analyzes the approaches for the construction of a global objective function: an *all-vertex* method where the positions of all free vertices are moved simultaneously in a single iteration; and a *single-vertex* method where the position of only one vertex is modified at a time. The study, limited to tetrahedral meshes, sets the harmonic average of the mean ratio as the measure of the quality of the mesh and concludes that, of the two methods considered, the single-vertex method is better for making fast improvements, while the all-vertex one is preferable when an accurate solution of the optimization problem is necessary.

A comparison of six morphing methods that preserve the topology throughout the optimization process has been proposed in [SOS*11] and tested over four tet and four hex example problems. The quality of the elements is assessed using the scaled Jacobian indicator. Notably, the authors notice that hexahedral meshes tend to maintain higher element quality for longer than tetrahedral meshes during morphing. This fact is more important in hexahedral meshes because local remeshing of hexahedra is generally considered intractable, while being straightforward for tetrahedral meshes. Since a global optimization approach might lead to a high computational effort, a combined approach of Laplacian smoothing and local op-

timization was proposed in [Fre97], where optimization is only accomplished in problematic regions of the mesh. In addition, theoretical developments on the local optimization problem based on shape indicators for triangular and tetrahedral elements were presented by Aiffa and Flaherty [AF03].

Laplace smoothing and global optimization have been the dominating techniques in mesh optimization based on node shifting in the early days of mesh generation. A breakthrough was made when Vartziotis et al. [VAGW08] proposed the Geometric Element Transformation Method (GETMe), which is a purely geometric process to move the nodes of an element to improve its quality. The driving force behind GETMe smoothing is regularizing element transformations which, if applied iteratively, lead to more regular elements (i.e., with a higher mean ratio), see Figure 15. Such transformations for polygons with an arbitrary number of nodes have been proposed and analyzed in [VW10], and an extension to hexahedra and general polyhedral elements is detailed in [VW12].

6.2. Optimization by Topological Operations

Local topological operations such as edge/face swaps, and elimination of nodes, edges/faces, and elements, are effective means to improve the quality of a mesh. These operations can be carried out based on an appropriate shape indicator, and/or according to some topological considerations, to create a structure as much balanced as possible in the connections among nodes, edges, faces, and elements. Numerous contributions address mesh-related questions as mesh optimization problems and deal with edge collapse and edge swap operations. For instance, the seminal work [HDD*93] minimizes an energy function that explicitly models the competing desires of conciseness of representation and fidelity to the data. Starting from this method, numerous works have addressed the simplification of a model as an optimization problem, e.g., [RR96, EM99, LT00]. The related literature is wide, and we only report here some examples to give a hint on how shape indicators can be combined for mesh optimization (or also *re-meshing*). We address the interested reader to [AUGA08, ACK13b] for a more detailed treatment of the subject.

General optimization algorithms, for instance, automatically eliminate valence-three nodes (nodes with three incident edges) from a triangle mesh (Figure 16(a)). This is commonly considered a safe operation, as it reduces the number of elements and edges and the quality of the newly created triangle is typically higher than those of the old ones. Similarly, short edges smaller than a certain threshold can be eliminated by shrinking the two related triangles to line segments, and small triangles are eliminated by shrinking them to a point (Figure 17(b,c)). Analogous automatic operations can be defined for quad-, tet-, and hex-meshes [MBAE09], but only if there is little or no ambiguity on how to adjust the connectivity after an element is removed or modified.

The situation changes if we want to remove a valence-four node from a triangle mesh, as in Figure 17(a). In this case, we obtain a quadrilateral element that can be triangulated in two different ways (along the two diagonals), and we need a criterion to establish which one to choose. Quality indicators are employed in topological optimizations every time in which multiple configurations

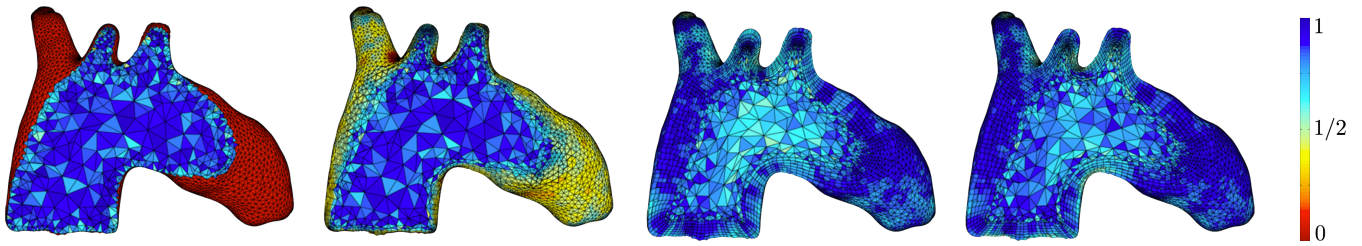


Figure 15: Initial and smoothed aorta meshes with elements colored according to their mean ratio [VW12]. From left to right: initial mesh with prismatic boundary layer; Laplace smoothing; global optimization; GETMe optimization.

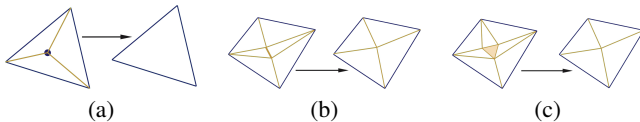


Figure 16: Triangle mesh optimization: (a) elimination of a valence-three node, (b) short edge shrinking, (c) small triangle shrinking.

are acceptable, to indicate the best one. Other examples are the diagonal swap between couples of adjacent triangles or quads, and the elimination of a valence-three node in a quad-mesh, see Figure 17(b,c).

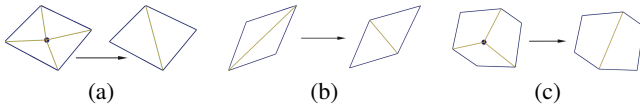


Figure 17: Triangle mesh optimization: (a) elimination of a valence-four node, (b) diagonal swap. Quad mesh optimization: (c) elimination of a valence-three node.

The preferred configurations for local transformations are those whose mean geometric qualities are maximized. Although the resulting mesh would certainly be different, the optimization by local transformations can be done with respect to any valid quality indicator. In three dimensions, there exist several methods for the optimization of tetrahedral meshes also using topological operations, e.g., [KS08, CZZ*17, MW21], while much less is available for generic polyhedra. When possible, generic polyhedra are reduced to a set of tetrahedra [VH14] and quality is assessed on the mean volume of the equivalent tetrahedrization or ad-hoc strategies are adopted, for instance, using grid-meshes [SZM12].

Pure hexahedral meshes are very often generated by mapping and sweeping methods, which are structured by the nature of their generation, and hence, there is no need to rebuild their topology. When hybrid elements are present, the application of mesh smoothing or untangling algorithms becomes more challenging, and a common assumption is that all the polyhedral elements admit a tetrahedrization [PCS*22].

Optimizing the element connections in polytopal meshes, in general, is difficult enough to leave any room for topological optimiza-

tion by a change in element types. Indeed, in such situations, other possible solutions would have already been evaluated right at the spot when the problem first arises. In other words, topological optimization of polyhedral meshes might be possible; however, it is unlikely to be cost-effective as local element swaps can only be applied to very limited locations if any, even after rigorous and time-consuming analysis [Lo14].

6.3. Recent Trends

Numerical solutions on complex geometries need to capture intricate physical features which in turn compel engineers to seek a balance between accuracy and computational cost, despite the lively growth in computing power. Unstructured meshes promise to yield this balance, offering accurate results and being preferred over their structured counterpart in certain applications [ZOG22]. On the one hand, to progress in this direction, methods are being investigated to improve the estimation of the optimization objective function; for instance, by leveraging the eigenanalysis of a Jacobian-based operator computed over proper subsets of vertices [ZOG22]. On the other hand, the increasing interest in developing polytopal element methods for the numerical discretizations of partial differential equations requires novel and effective algorithms to handle polygonal and polyhedral grids and to assess their quality.

The interest in using general polytopal meshes has grown over the years because these models are well suited for handling complex domains, featuring geometrical complexities for which the generation of good quality simplicial or hexahedral meshes can be particularly expensive or even unfeasible. Among the open problems, there is the issue of handling polytopal mesh refinement [LS16, BS22, BBD21], i.e., partitioning mesh elements into smaller elements to produce a finer grid, and agglomeration strategies [CXZ98, LWCY13, ABBV21], i.e., merging mesh elements to obtain coarser grids. In both cases, refinement and agglomeration, it is crucial to preserve the quality of the underlying mesh, because this might affect the overall performance of the method in terms of stability and accuracy. Adaptivity is a key point that guarantees reliability and efficiency. For simplicity, algorithms often privilege dealing with convex polygons or polyhedra. For instance, the refinement algorithms in [BBD21], start from a coarse mesh made of convex polygons created from a number of discrete fractures and use aspect ratio to select elements to be cut with planes, oriented according to specific criteria.

Application-oriented problems such as simulation on fracture

networks call for approaches to mesh optimization able to deal with the geometrical complexity and the size of the computational domain. Indeed, fracture networks described by the Discrete Fracture Network (DFN) model correspond to sets of intersecting planar polygons arbitrarily oriented in a three-dimensional space, representing fractures in underground rock formations, and are typically generated starting from hydraulic and geological soil models. These models might count a large number of fractures, with dimensions ranging from centimeters to kilometers, and form an intricate network of intersections. To address this problem it is necessary to use methods able to scale to different sizes, incorporate refinement strategies when necessary, and agglomerate elements far from fractures. The methods developed in [BPS13, BGPS21] cut the elements of a tessellation with the intersection segments of two fracture plans, and use an optimization approach to further split the elements that are too thin and elongated. Vertex collapses are in place too, e.g., if two vertices are close under a threshold, as well as small shifts of vertices if the mesh quality locally improves. In the terms of the previous sections, we would classify this strategy as a geometry-topology-driven optimization. Further agglomerations are possible, in particular, far from the fracture constraints. Recently, an agglomeration approach driven by the quality indicator (30) has been proposed in [SVB*22]. Extensions to whole three-dimensional domains are currently under study, with preliminary results in [BGS22].

A key aspect that is receiving increasing attention is how to keep elements aligned to some extent with the geometry of the solution. In this direction, previous studies on simplicial meshes proved that a properly generated anisotropic mesh can lead to a much more accurate solution than an isotropic one with comparable size, see [HW20] and the references therein. In this view, the shape of the elements of the tessellation is not necessarily uniform, and mesh elements with a large aspect ratio are also admitted [HW20]. Based on the intuition that it is convenient to adapt the tessellation to the physical problem, recent studies address the problem of how to deform a mesh to follow sharp physical interfaces, see for instance [MRL*23]. In this work, the topology of the tessellation does not change during the deformation process, rather it admits that elements decrease their quality measure until zero. These studies and the anisotropic mesh quality measures available in [HW20] focus on convex polygonal meshes.

Finally, we mention the ongoing investigation of the use of Convolutional Neural Networks (CNNs) for handling polygonal [AM22] and polyhedra [ADM22] grid refinement. The idea behind the approach in [AM22] is to recognize an almost regular single polygon as the representative of a set of polygons and then apply a polygon refinement strategy to the recognized element. Currently, triangular, quadrangular, pentagonal, and hexagonal polygons are used as basic shapes for recognition. Similarly, when dealing with polyhedra, a k-means algorithm is used to learn a clustered representation of the elements of an initial tessellation, on the basis of some basic convex polyhedral classes (cube, prism, tetrahedron, and other). The type of the recognized element determines the partition rule [ADM22], thus improving and replacing the elements of the original mesh. Being these approaches supervised, the quality of the polytopes recognized by the neural network is demanded to

the variety and representativeness of the polytopes used to train the model.

7. Available Resources

This section summarizes several resources that are currently available for investigating the quality of a mesh, according to the different quality indicators. We also include some datasets publicly available online, that can be used to test the accuracy of a quality indicator in predicting the performance of a simulation. These lists are, obviously, in constant evolution.

Libraries A vast amount of the quality indicators presented in Section 4.1 and Section 4.2 is detailed in the Verdict Geometric Quality Library [KET*06]. The Verdict library became so popular that its indicators are currently implemented also in more general mesh processing toolkits like Cinolib [Liv19] or HexaLab [BTP*19]. An implementation of the target-matrix paradigm is available in Mesquite [BDK*03]. Popular geometry processing libraries like CGAL [FP09], GMSH [GR09], or Geogram [LF15], also implement routines for some indicators, but they are often only defined for specific types of elements. The Mesh Repair website [ACK13a] contains a list of freely available software tools that perform the repairing of polygonal meshes with various sorts of defects and flaws. This list was published in the context of mesh repairing [ACK13b], which is somehow preliminary to mesh quality optimization, but we report it here for completeness.

The availability of commercial and open software libraries for FEMs and the related mesh generation methods is very wide, being it an established field with over 70 years of development on the subject. By way of example, we mention the *Discrete Methods & Related Tools* website [Jou01], which contains links to a wide range of commercial or open software (133 links), classified according to their functionality (FEM modelers, FEM solvers, creation of surface and volumetric meshes and grids, type of application, etc.).

More specific software that handles polygons and polyhedra are OpenMesh [BBS*02], OpenVolumeMesh [KBK13], and PolyMesher [TPPM12]. The PEMesh framework [CPS22] allows the computation of the quality indicators analyzed in [ABB*21] for generic polygons. The work by Schneider et al. proposed in the papers [SHD*18, SDG*19a, SHG*22] is condensed into the PolyFEM library [SDG*19b], a polyvalent C++ and Python FEM library. The Voro++ library is a C++ open-source software for the computation of the Voronoi diagram [Ryc08]. It contains the routines for constructing a single Voronoi cell, which is represented as a collection of vertices that are connected by edges, and to iterate the creation of cells.

Datasets The ABC dataset [KMJ*19] is a collection of about one million computer-aided design models proposed for research in geometric deep learning and applications, provided in different formats. Models were originally created with Onshape, and come in STEP and Parasolid format. Models in the Parasolid format can then be tessellated and saved in the STL format, obtaining triangle meshes with high resolution that carefully approximate the original geometry, but with low mesh quality, as the triangles may have bad aspect ratios and the sampling can be highly non-uniform. The

GMSH library is then used to create a high-quality triangle mesh in OBJ format, with a uniform vertex distribution and most triangles have angles close to 60 degrees.

A popular reference for the generation of tetrahedral meshes from a triangle soup is TetWild [HZG*18]. Despite being a tetrahedral meshing technique, TetWild is also a tetrahedral dataset, since the authors provide the output of their algorithm applied to Thingi10k [ZJ16], a dataset of triangular meshes containing ten thousand models. A recent dataset, SimJEB [WBM21], provides 381 tetrahedral meshes from CAD models. The HexaLab project [BTP*19] is a unified portal to visualize hexahedral meshes directly in a web browser as well as to download them. The recent survey [PCS*22] presents a wide list of 3D datasets that mainly contains pure hexahedral or hex-dominant meshes. Most of them are already included in Hexalab, notable exceptions are the HexMe tetrahedral dataset [BRK*22] and the MAMBO project [Led22], which stores geometric models for the study and comparison of hexahedral block meshing algorithms.

Regarding polytopal meshes, the Voro++ library also contains some examples of models tessellated with Voronoi cells. Similarly, the CVT dataset [Bur05] is a FORTRAN90 open-source software that generates a Centroidal Voronoi Tessellation from a set of vertices based on the works [DFG99b, BGPB02]. A dataset of polygonal meshes used for testing Indicator (30) is available on GitHub at [SBMS21], together with its 3D counterpart [SBMS22c].

8. Conclusions

We have systematically presented the indicators for assessing the quality of the elements of a mesh, classifying them according to the type of the element and the criteria adopted. While there is a large literature for simplicial [Knu03, SEK*07] and quadrilateral elements [BLP*13b], increasing attention is on hexahedral meshes [PCS*22], and there is still a wide room for research on polytopal ones. Then, we discussed how these indicators can be used to derive global mesh indicators and optimize the mesh in a variety of applicative contexts. Indeed, the need for high-quality meshes is much felt for running PDE simulations, but it is also crucial in computer graphics applications such as shape optimization and object manufacturing.

What we wish to leave to the reader is a more complete understanding of the complexity of this subject. Recalling what is commonly referred to as Ashby's Law, "only variety can absorb variety" [Ash57], we consider a viable system the one that can handle the variability of its environment. If a system is to be able to deal successfully with the diversity of challenges that its environment produces, then it needs to have a repertoire of responses that is (at least) as nuanced as the problems thrown up by the environment. We believe that mesh quality is not something we can completely control, in the sense that there will probably never be a single magic formula good for every context. What we can do is try to govern the quality, which means to know in advance all the different aspects of a mesh that may influence the final result, and choose which one(s) we should pay more attention to, from time to time.

Acknowledgements

We thank the chairs and the reviewers for all the very useful comments that helped us to improve the quality of our work. The authors have been financially supported by the ERC Project CHANGE, which has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement no. 694515). Open Access Funding provided by Consiglio Nazionale delle Ricerche within the CRUI-CARE Agreement.

References

- [AAB*13] AHMAD B., ALSAEDI A., BREZZI F., MARINI L. D., RUSSO A.: Equivalent projectors for virtual element methods. *Computers & Mathematics with Applications* 66 (September 2013), 376–391. 11
- [ABB*21] ATTENE M., BIASOTTI S., BERTOLUZZA S., CABIDDU D., LIVESU M., PATANÉ G., PENNACCHIO M., PRADA D., SPAGNUOLO M.: Benchmarking the geometrical robustness of a virtual element Poisson solver. *Mathematics and Computers in Simulation* 190 (2021), 1392–1414. 11, 18
- [ABBV21] ANTONIETTI P. F., BERRONE S., Busetto M., VERANI M.: Agglomeration-based geometric multigrid schemes for the virtual element method. *arXiv preprint arXiv:2112.11080* (2021). 17
- [ACK13a] ATTENE M., CAMPEN M., KOBBELT L.: Mesh Repair. <http://www.meshrepair.org>, 2013. 18
- [ACK13b] ATTENE M., CAMPEN M., KOBBELT L.: Polygon mesh repairing: An application perspective. *ACM Computing Surveys (CSUR)* 45, 2 (2013), 1–33. 1, 13, 15, 16, 18
- [ACSYD05] ALLIEZ P., COHEN-STEINER D., YVINEC M., DESBRUN M.: Variational tetrahedral meshing. In *ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), SIGGRAPH '05, Association for Computing Machinery, pp. 617–625. 15
- [AD92] APEL T., DOBROWOLSKI M.: Anisotropic interpolation with applications to the finite element method. *Computing* 47 (1992), 277–293. 6
- [ADL*98] AU P., DOMPIERRE J., LABBÉ P., LABB P., GUIBAULT F., GUIBAULT F., CAMARERO R.: Proposal of benchmarks for 3d unstructured tetrahedral mesh optimization. In *In Proceedings of the 7th International Meshing RoundTable'98* (1998), Citeseer. 8
- [ADM22] ANTONIETTI P., DASSI F., MANUZZI E.: Machine learning based refinement strategies for polyhedral grids with applications to virtual element and polyhedral discontinuous galerkin methods. *Journal of Computational Physics* 469 (2022), 111531. 18
- [AF03] AIFFA M., FLAHERTY J.: A geometrical approach to mesh smoothing. *Computer methods in applied mechanics and engineering* 192, 39–40 (2003), 4497–4514. 16
- [AM22] ANTONIETTI P. F., MANUZZI E.: Refinement of polygonal grids using convolutional neural networks with applications to polygonal discontinuous galerkin and virtual element methods. *Journal of Computational Physics* 452 (2022), 110900. 18
- [Ash57] ASHBY W. R.: *An introduction to cybernetics*. Chapman & Hall Ltd., 1957. 19
- [Att18] ATTENE M.: As-exact-as-possible repair of unprintable stl files. *Rapid Prototyping Journal* (2018). 15, 16
- [AUGA08] ALLIEZ P., UCELLI G., GOTSMAN C., ATTENE M.: Recent advances in remeshing of surfaces. *Shape analysis and structuring* (2008), 53–82. 16
- [AYZ12] AGHDAI N., YOUNESY H., ZHANG H.: 5–6–7 meshes: Remeshing and analysis. *Computers & Graphics* 36, 8 (2012), 1072–1083. 14

- [BA76] BABUŠKA I., AZIZ A. K.: On the angle condition in the finite element method. *SIAM Journal on numerical analysis* 13, 2 (1976), 214–226. 1, 6
- [BadVBC*13] BEIRÃO DA VEIGA L., BREZZI F., CANGIANI A., MANZINI G., MARINI L. D., RUSSO A.: Basic principles of virtual element methods. *Mathematical Models & Methods in Applied Sciences* 23 (2013), 119–214. 5, 11
- [BadVV20] BEIRÃO DA VEIGA L., VACCA G.: Sharper error estimates for virtual elements and a bubble-enriched version. *arXiv preprint arXiv:2005.12009* (2020). 11
- [Bar96] BARRETT K.: Jacobians for isoparametric finite elements. *Communications in numerical methods in engineering* 12, 11 (1996), 755–766. 4
- [BBD21] BERRONE S., BORIO A., D’AURIA A.: Refinement strategies for polygonal meshes applied to adaptive vem discretization. *Finite Elements in Analysis and Design* 186 (2021), 103502. 17
- [BBS*02] BISCHOFF B. S., BOTSCH M., STEINBERG S., BISCHOFF S., KOBBELT L., AACHEN R.: Openmesh—a generic and efficient polygon mesh data structure. In *In OpenSG Symposium* (2002). 18
- [BDK*03] BREWER M. L., DIACHIN L. F., KNUPP P. M., LEURENT T., MELANDER D. J.: The mesquite mesh quality improvement toolkit. In *IMR* (2003). 18
- [BdVLM14] BEIRÃO DA VEIGA L., LIPNIKOV K., MANZINI G.: *The mimetic finite difference method for elliptic problems*, vol. 11. Springer, 2014. 5
- [BdVLR17] BEIRÃO DA VEIGA L., LOVADINA C., RUSSO A.: Stability analysis for the virtual element method. *Mathematical Models and Methods in Applied Sciences* 27, 13 (2017), 2557–2594. 11
- [Ber98] BERZINS M.: A solution-based triangular and tetrahedral mesh quality indicator. *Siam Journal on Scientific Computing* 19 (1998). 6
- [BGPB02] BURKARDT J., GUNZBURGER M., PETERSON J., BRANNON R. M.: *User manual and supporting information for library of codes for centroidal Voronoi point placement and associated zeroth, first, and second moment determination*. Tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States); Sandia . . . , 2002. 19
- [BGPS21] BERRONE S., GRAPPEIN D., PIERACCINI S., SCIALÒ S.: A three-field based optimization formulation for flow simulations in networks of fractures on nonconforming meshes. *SIAM Journal on Scientific Computing* 43, 2 (2021), B381–B404. 18
- [BGS17] BRENNER S. C., GUAN Q., SUNG L.-Y.: Some estimates for virtual element methods. *Computational Methods in Applied Mathematics* 17, 4 (2017), 553–574. 11
- [BGS22] BERRONE S., GRAPPEIN D., SCIALÒ S.: 3d-1d coupling on non conforming meshes via a three-field optimization based domain decomposition. *Journal of Computational Physics* 448 (2022), 110738. 18
- [BHF98] BOROUCHE H., HECHT F., FREY P. J.: Mesh gradation control. *International Journal for Numerical Methods in Engineering* 43, 6 (1998), 1143–1165. 15
- [BKP*10] BOTSCH M., KOBBELT L., PAULY M., ALLIEZ P., LÉVY B.: *Polygon mesh processing*. CRC press, 2010. 1, 15
- [BLP*13a] BOMMES D., LÉVY B., PIETRONI N., PUPPO E., SILVA C., TARINI M., ZORIN D.: Quad-mesh generation and processing: A survey. *Computer Graphics Forum* 32, 6 (2013), 51–76. 1
- [BLP*13b] BOMMES D., LÉVY B., PIETRONI N., PUPPO E., SILVA C., TARINI M., ZORIN D.: Quad-mesh generation and processing: A survey. In *Computer Graphics Forum* (2013), vol. 32-6, Wiley Online Library, pp. 51–76. 9, 14, 19
- [BM04] BERTOLAZZI E., MANZINI G.: A cell-centered second-order accurate finite volume method for convection-diffusion problems on unstructured meshes. *Math. Mod. Methods Appl. Sci.* 14, 8 (2004), 1235–1260. 5
- [BM07] BERTOLAZZI E., MANZINI G.: On vertex reconstructions for cell-centered finite volume approximations of 2-D anisotropic diffusion problems. *Math. Mod. Methods Appl. Sci.* 17, 1 (2007), 1–32. 5
- [BMM*00] BREZZI F., MANZINI G., MARINI D., PIETRA P., RUSSO A.: Discontinuous Galerkin approximations for elliptic problems. *Numerical Methods for Partial Differential Equations* 16, 4 (2000), 365–378. 5
- [BPS13] BERRONE S., PIERACCINI S., SCIALÒ S.: A pde-constrained optimization formulation for discrete fracture network flows. *SIAM Journal on Scientific Computing* 35, 2 (2013), B487–B510. 18
- [BRK*22] BEAUFORT P.-A., REBEROL M., KALMYKOV D., LIU H., LEDOUX F., BOMMES D.: Hex me if you can. *Computer Graphics Forum* 41, 5 (2022). 19
- [BS08] BRENNER S. C., SCOTT L. R.: *The mathematical theory of finite element methods*, 3 ed. Texts in applied mathematics 15. Springer-Verlag, New York, 2008. 1, 3, 5
- [BS18] BRENNER S. C., SUNG L.-Y.: Virtual element methods on meshes with small edges or faces. *Mathematical Models and Methods in Applied Sciences* 28, 07 (2018), 1291–1336. 11
- [BS22] BAKHYALOV P., SURNACHEV M.: Method of averaged element splittings for diffusion terms discretization in vertex-centered framework. *Journal of Computational Physics* 450 (2022), 110819. 17
- [BTP*19] BRACCI M., TARINI M., PIETRONI N., LIVESU M., CIGNONI P.: Hexalab. net: An online viewer for hexahedral meshes. *Computer-Aided Design* 110 (2019), 24–36. 18, 19
- [Bur05] BURKARDT J.: CVT dataset. <https://people.sc.fsu.edu/~jburkardt/datasets/cvt/cvt.html>, 2005. 19
- [BX96] BANK R. E., XU J.: An algorithm for coarsening unstructured meshes. *Numerische Mathematik* 73, 1 (1996), 1–36. 8
- [CAK12] CAMPEN M., ATTENE M., KOBBELT L.: A practical guide to polygon mesh repairing. In *Eurographics (Tutorials)* (2012), p. 32. 15
- [Car97] CAREY G. F.: *Computational Grids: Generations, Adaptation & Solution Strategies*, 1st edition ed. Computational and Physical Processes in Mechanics and Thermal sciEnces. Taylor and Francis, Washington, DC, 1997. ISBN-10: 1560326352. 1
- [CCC*08] CIGNONI P., CALLIERI M., CORSINI M., DELLEPIANE M., GANOVELLI F., RANZUGLIA G.: MeshLab: an Open-Source Mesh Processing Tool. In *Eurographics Italian Chapter Conference* (2008), Scarano V., Chiara R. D., Erra U., (Eds.), The Eurographics Association. doi:10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2008/129-136. 15
- [CDRR04] CHENG S.-W., DEY T. K., RAMOS E. A., RAY T.: Sampling and meshing a surface with guaranteed topology and geometry. In *Proceedings of the Twentieth Annual Symposium on Computational Geometry* (New York, NY, USA, 2004), SCG ’04, Association for Computing Machinery, p. 280–289. doi:10.1145/997817.997861. 13
- [CHSS13] CHALMETA R., HURTADO F., SACRISTÁN V., SAUMELL M.: Measuring regularity of convex polygons. *Computer-Aided Design* 45, 2 (2013), 93–104. 11
- [Cia02] CIARLET P. G.: *The finite element method for elliptic problems*. SIAM, 2002. 1
- [Cox38] COXETER H. S. M.: Regular skew polyhedra in three and four dimension, and their topological analogues. *Proceedings of the London Mathematical Society* 2, 1 (1938), 33–62. 11
- [CPS22] CABIDDU D., PATANÉ G., SPAGNUOLO M.: A Graphical Framework to Study the Correlation between Geometric Design and Simulation. In *Smart Tools and Applications in Graphics - Eurographics Italian Chapter Conference* (2022), Cabiddu D., Schneider T., Allegra D., Catalano C. E., Cherchi G., Scateni R., (Eds.), The Eurographics Association. doi:10.2312/stag.20221251. 12, 18
- [CXZ98] CHAN T. F., XU J., ZIKATANOV L. T.: An agglomeration multigrid method for unstructured grids. *Contemporary mathematics* (1998). 17

- [CZZ*17] CHEN J., ZHENG J., ZHENG Y., XIAO Z., SI H., YAO Y.: Tetrahedral mesh improvement by shell transformation. *Engineering with Computers* 33, 3 (2017), 393–414. [17](#)
- [DD12] DEZA M. M., DEZA E.: *Encyclopedia of Distances*. Springer Science & Business Media, 2012. [14](#)
- [DFG99a] DU Q., FABER V., GUNZBURGER M.: Centroidal voronoi tessellations: Applications and algorithms. *SIAM review* 41, 4 (1999), 637–676. [11](#)
- [DFG99b] DU Q., FABER V., GUNZBURGER M.: Centroidal voronoi tessellations: Applications and algorithms. *SIAM Review* 41, 4 (1999), 637–676. [doi:10.1137/S0036144599352836](#). [19](#)
- [DPD19] DI PIETRO D. A., DRONIOU J.: *The Hybrid High-Order method for polytopal meshes*, vol. 19. Springer, 2019. [5, 11](#)
- [DS80] DUPONT T., SCOTT R.: Polynomial approximation of functions in sobolev spaces. *Mathematics of Computation* 34, 150 (1980), 441–463. [3](#)
- [DWZ09] DU Q., WANG D., ZHU L.: On mesh geometry and stiffness matrix conditioning for general finite element spaces. *SIAM journal on numerical analysis* 47, 2 (2009), 1421–1444. [7, 8](#)
- [EE20] ENGVALL L., EVANS J. A.: Mesh quality metrics for isogeometric bernstein–bézier discretizations. *Computer Methods in Applied Mechanics and Engineering* 371 (2020), 113305. [8](#)
- [EM99] ERIKSON C., MANOCHA D.: Gaps: General and automatic polygonal simplification. In *Proceedings of the 1999 symposium on Interactive 3D graphics* (1999), pp. 79–88. [16](#)
- [EÜZ09] ERTEN H., ÜNGÖR A., ZHAO C.: Mesh smoothing algorithms for complex geometric domains. In *Proceedings of the 18th international meshing roundtable*. Springer, 2009, pp. 175–193. [16](#)
- [FDKMS06] FREITAG DIACHIN L., KNUPP P., MUNSON T., SHONTZ S.: A comparison of two optimization methods for mesh quality improvement. *Engineering with Computers* 22, 2 (2006), 61–74. [16](#)
- [FG00] FREY P., GEORGE P.: Mesh generation. application to finite elements. hermes science publ., paris, 2000. [9](#)
- [Fie00] FIELD D. A.: Qualitative measures for initial meshes. *International Journal for Numerical Methods in Engineering* 47, 4 (2000), 887–906. [4, 8, 9, 10](#)
- [FP00] FREITAG L. A., PLASSMANN P.: Local optimization-based simplicial mesh untangling and improvement. *International Journal for Numerical Methods in Engineering* 49, 1-2 (2000), 109–125. [16](#)
- [FP09] FABRI A., PION S.: Cgal: The computational geometry algorithms library. In *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems* (2009), pp. 538–539. [18](#)
- [Fre97] FREITAG L. A.: *On combining Laplacian and optimization-based mesh smoothing techniques*. Tech. rep., Argonne National Lab., IL (United States), 1997. [4, 16](#)
- [Fri72] FRIED I.: Condition of finite element matrices generated from nonuniform meshes. *AIAA Journal* 10 (1972). Technical Note. [7](#)
- [GB98] GEORGE P.-L., BOROCHAKI H.: *Delaunay Triangulation and Meshing: Application to Finite Elements*. Technology & Engineering, Hermès, 1998. ISBN: 2866016920, 9782866016920. [1](#)
- [GHX*17] GAO X., HUANG J., XU K., PAN Z., DENG Z., CHEN G.: Evaluating hex-mesh quality metrics via correlation analysis. In *Computer Graphics Forum* (2017), vol. 36-5, Wiley Online Library, pp. 105–116. [12](#)
- [GR09] GEUZAIN C., REMACLE J.-F.: Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering* 79, 11 (2009), 1309–1331. [18](#)
- [GRB12] GILLETTE A., RAND A., BAJAJ C.: Error estimates for generalized barycentric interpolation. *Advances in computational mathematics* 37, 3 (2012), 417–439. [10, 11](#)
- [HCB05] HUGHES T. J., COTTRELL J. A., BAZILEVS Y.: Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering* 194, 39-41 (2005), 4135–4195. [2](#)
- [HDD*93] HOPPE H., DE ROSE T., DUCHAMP T., McDONALD J., STUETZLE W.: Mesh optimization. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1993), SIGGRAPH '93, Association for Computing Machinery, p. 19–26. [14, 15, 16](#)
- [Her76] HERRMANN L. R.: Laplacian-isoparametric grid generation scheme. *Journal of the Engineering Mechanics Division* 102, 5 (1976), 749–756. [16](#)
- [HW20] HUANG W., WANG Y.: Anisotropic mesh quality measures and adaptation for polygonal meshes. *Journal of Computational Physics* 410 (2020), 109368. [11, 15, 18](#)
- [HZG*18] HU Y., ZHOU Q., GAO X., JACOBSON A., ZORIN D., PANOZZO D.: Tetrahedral meshing in the wild. *ACM Trans. Graph.* 37, 4 (2018), 60–1. [19](#)
- [Jou01] JOU H.-J.: Sas - discrete methods & related tools. <http://www.sai.msu.su/sai/B/2/>, 2001. [18](#)
- [KKB13] KREMER M., BOMMES D., KOBBELT L.: Openvolumemesh—a versatile index-based data structure for 3d polytopal complexes. In *Proceedings of the 21st International Meshing Roundtable*. Springer, 2013, pp. 531–548. [18](#)
- [KC00] KWOK W., CHEN Z.: A simple and effective mesh quality metric for hexahedral and wedge elements. In *IMR* (2000), pp. 325–333. [11](#)
- [KCS98] KOBBELT L., CAMPAGNA S., SEIDEL H.-P.: A general framework for mesh decimation. In *Proceedings of the Graphics Interface 1998 Conference, June 18-20, 1998, Vancouver, BC, Canada* (June 1998), pp. 43–50. [14](#)
- [KET*06] KNUPP P. M., ERNST C., THOMPSON D. C., STIMPSON C., PEBAY P. P.: *The verdict geometric quality library*. Tech. rep., Sandia National Laboratories (SNL), Albuquerque, NM, and Livermore, CA . . . , 2006. [18](#)
- [Kir92] KIR'IZEK M.: On the maximum angle condition for linear tetrahedral elements. *SIAM Journal on Numerical Analysis* 29 (1992). [6](#)
- [KMJ*19] KOCH S., MATVEEV A., JIANG Z., WILLIAMS F., ARTEMOV A., BURNAEV E., ALEXA M., ZORIN D., PANOZZO D.: Abc: A big cad model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 9601–9611. [18](#)
- [Knu99] KNUPP P.: *Matrix norms and the condition number: A general framework to improve mesh quality via node-movement*. Tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States); Sandia . . . , 1999. [4](#)
- [Knu00a] KNUPP P. M.: Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part ii—a framework for volume mesh optimization and the condition number of the jacobian matrix. *International Journal for numerical methods in engineering* 48, 8 (2000), 1165–1185. [4, 10](#)
- [Knu00b] KNUPP P. M.: Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part i—a framework for surface mesh optimization. *International Journal for Numerical Methods in Engineering* 48, 3 (2000), 401–420. [10](#)
- [Knu01] KNUPP P. M.: Algebraic mesh quality metrics. *SIAM journal on scientific computing* 23, 1 (2001), 193–218. [1, 4, 8, 10, 12](#)
- [Knu03] KNUPP P. M.: Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elements in Analysis and Design* 39, 3 (2003), 217–241. [4, 8, 10, 19](#)
- [Knu07] KNUPP P.: *Formulation of a Target-Matrix Paradigm for Mesh Optimization*. Tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2007. [11](#)

- [Knu12] KNUPP P.: Introducing the target-matrix paradigm for mesh optimization via node-movement. *Engineering with Computers* 28, 4 (2012), 419–429. 11, 16
- [KPS13] KIM J., PANITANARAK T., SHONTZ S. M.: A multiobjective mesh optimization framework for mesh quality improvement and mesh untangling. *International Journal for Numerical Methods in Engineering* 94, 1 (2013), 20–42. 16
- [KS08] KLINGNER B. M., SHEWCHUK J. R.: Aggressive tetrahedral mesh improvement. In *Proceedings of the 16th International Meshing Roundtable* (Berlin, Heidelberg, 2008), Brewer M. L., Marcum D., (Eds.), Springer Berlin Heidelberg, pp. 3–23. 17
- [LADH22] LOBOS C., ARENAS C., DAINES E., HITSCHFELD N.: Measuring geometrical quality of different 3d linear element types. *Numerical Algorithms* 90, 1 (2022), 423–446. 11
- [Led22] LEDOUX F.; MAMBO. <https://gitlab.com/franck.ledoux/mambo>, 2022. 19
- [LF15] LÉVY B., FILBOIS A.: Geogram: a library for geometric algorithms, 2015. 18
- [Lis06] LISEIKIN V. D.: *A computational differential geometry approach to grid generation*. Springer Science & Business Media, 2006. 13
- [Lis17] LISEIKIN V. D.: *Grid generation methods*, vol. 1. Springer, 2017. 1, 9, 10, 13, 15
- [Liv19] LIVESU M.: cinolib: a generic programming header only C++ library for processing polygonal and polyhedral meshes. *Transactions on Computational Science XXXIV* (2019). <https://github.com/mlivesu/cinolib/>. 18
- [LJ94] LIU A., JOE B.: Relationship between tetrahedron shape measures. *BIT Numerical Mathematics* 34, 2 (1994), 268–287. 1, 8, 12
- [Lo97] LO S.: Optimization of tetrahedral meshes based on element shape measures. *Computers & structures* 63, 5 (1997), 951–961. 8
- [Lo13] LO D. S.: Dynamic grid for mesh generation by the advancing front method. *Computers & Structures* 123 (2013), 15–27. 15
- [Lo14] LO D. S.: *Finite element mesh generation*. CRC Press, 2014. 15, 17
- [LPP*20] LIVESU M., PIETRONI N., PUPPO E., SHEFFER A., CIGNONI P.: Loopycuts: Practical feature-preserving block decomposition for strongly hex-dominant meshing. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 121–1. 14
- [LS07] LAI M.-J., SCHUMAKER L. L.: *Spline functions on triangulations*, vol. 110. Cambridge University Press, 2007. 1
- [LS16] LAI M.-J., SLAVOV G.: On recursive refinement of convex polygons. *Computer Aided Geometric Design* 45 (2016), 83–90. 1, 17
- [LT00] LINDSTROM P., TURK G.: Image-driven simplification. *ACM Trans. Graph.* 19, 3 (jul 2000), 204–241. 14, 16
- [LWCY13] LI Z., WANG Z., CAO W., YAO L.: An aspect ratio agglomeration multigrid for unstructured grids. *International Journal for Numerical Methods in Fluids* 72, 10 (2013), 1034–1050. 17
- [MBAE09] MISZTAL M. K., BÆRENTZEN J. A., ANTON F., ERLEBEN K.: Tetrahedral mesh improvement using multi-face retriangulation. In *Proceedings of the 18th international meshing roundtable*. Springer, 2009, pp. 539–555. 16
- [MC21] MANDAD M., CAMPEN M.: Guaranteed-quality higher-order triangular meshing of 2d domains. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–14. 8
- [MPKZ10] MYLES A., PIETRONI N., KOVACS D., ZORIN D.: Feature-aligned t-meshes. *ACM Transactions on Graphics (TOG)* 29, 4 (2010), 1–11. 14
- [MPW71] MITCHELL A., PHILLIPS G., WACHSPRESS E.: Forbidden shapes in the finite element method. *IMA Journal of Applied Mathematics* 8, 2 (1971), 260–269. 4
- [MRL*23] MOËS N., REMACLE J.-F., LAMBRECHTS J., LÉ B., CHEVAUGEON N.: The extreme mesh deformation approach (x-mesh). *arXiv preprint arXiv:2111.04179* (2023). 18
- [MV99] MITCHELL S., VAVASIS S.: Quality mesh generation in three dimensions. *Proceedings of the ACM Computational Geometry Conference* (1999). 1
- [MW21] MA Y., WANG M.: An efficient method to improve the quality of tetrahedron mesh with mfr. *Scientific Reports* 11, 1 (2021), 1–20. 17
- [MWH97] MARCHANT M. J., WEATHERILL N. P., HASSAN O.: The adaptation of unstructured grids for transonic viscous flow simulation. *Finite Elements in Analysis and Design* 25, 3 (1997), 199–218. Adaptive Meshing, Part 2. 1
- [Nad86] NADLER E.: Piecewise linear best L^2 approximation on triangulations. *Journal of Approximation Theory - JAT* (1986). 6
- [Nie97] NIELSON G. M.: Tools for triangulation and tetrahedrizations and constructing functions defined over them. In *Scientific Visualization*, Nielson G. M., Hagen H., Müller H., (Eds.). IEEE Computer Society, Los Alamitos, CA, 1997, pp. 429–525. 1
- [OGMB88] ODDY A., GOLDAK J., MCDILL M., BIBBY M.: A distortion metric for isoparametric finite elements. *Transactions of the Canadian Society for Mechanical Engineering* 12, 4 (1988), 213–217. 10
- [Owe98] OWEN S. J.: A survey of unstructured mesh generation technology. *IMR* 239 (1998), 267. 15
- [PB03] PÉBAY P., BAKER T.: Analysis of triangle quality measures. *Mathematics of computation* 72, 244 (2003), 1817–1839. 8, 12
- [PCS*22] PIETRONI N., CAMPEN M., SHEFFER A., CHERCHI G., BOMMES D., GAO X., SCATENI R., LEDOUX F., REMACLE J.-F., LIVESU M.: Hex-mesh generation and processing: a survey. *ACM Transactions on Graphics* (2022). 1, 14, 15, 17, 19
- [Péb04] PÉBAY P. P.: Planar quadrilateral quality measures. *Engineering with Computers* 20, 2 (2004), 157–173. 9
- [PGH94] PARTHASARATHY V., GRAICHEN C., HATHAWAY A.: A comparison of tetrahedron quality measures. *Finite Elements in Analysis and Design* 15, 3 (1994), 255–261. 12
- [Pre82] PREISS K.: Topological consistency rules for general finite element meshes. In *CAD82*, Pipes A., (Ed.). Butterworth-Heinemann, 1982, pp. 453–460. 13
- [PS85] PREPARATA F. P., SHAMOS M. I.: *Computational Geometry: An Introduction*. Springer-Verlag, Berlin, Heidelberg, 1985. 3
- [RB93] ROSSIGNAC J., BORREL P.: Multi-resolution 3d approximations for rendering complex scenes. In *Modeling in Computer Graphics* (Berlin, Heidelberg, 1993), Falcidieno B., Kunii T. L., (Eds.), Springer Berlin Heidelberg, pp. 455–465. 14
- [RGPS11] ROCA X., GARGALLO-PEIRÓ A., SARRATE J.: Defining quality measures for high-order planar triangles and curved mesh generation. In *Proceedings of the 20th International Meshing Roundtable*. Springer, 2011, pp. 365–383. 4
- [Rip92] RIPPA S.: Long and thin triangles can be good for linear interpolation. *SIAM Journal on Numerical Analysis* 29 (1992), 257–270. 6, 15
- [Rob87a] ROBINSON J.: Cre method of element testing and the jacobian shape parameters. *Engineering Computations* (1987). 9, 10
- [Rob87b] ROBINSON J.: Some new distortion measures for quadrilaterals. *Finite Elements in Analysis and Design* 3, 3 (1987), 183–197. 9, 10
- [Rob94] ROBINSON J.: Quadrilateral and hexahedron shape parameters. *Finite elements in analysis and design* 16, 1 (1994), 43–52. 9, 10
- [RR96] RONFARD R., ROSSIGNAC J.: Full-range approximation of triangulated polyhedra. In *Computer Graphics Forum* (1996), vol. 15-3, Wiley Online Library, pp. 67–76. 16
- [Ryc08] RYCROFT C.: Voro++. <https://math.lbl.gov/voro+/about.html>, 2008. 18

- [SA21] SHAKOUR E., AMIR O.: Topology optimization with precise evolving boundaries based on iga and untrimming techniques. *Computer Methods in Applied Mechanics and Engineering* 374 (2021), 113564. doi:<https://doi.org/10.1016/j.cma.2020.113564>. 13
- [SB94] SALAGAME R. R., BELEGUNDU A. D.: Distortion, degeneracy and rezoning in finite elements—a survey. *Sadhana* 19, 2 (1994), 311–335. 4
- [SBMS21] SORGENTE T., BIASOTTI S., MANZINI G., SPAGNUOLO M.: VEM 2D Dataset. <https://github.com/TommasoSorgente/vem-2D-quality-dataset>, 2021. 19
- [SBMS22a] SORGENTE T., BIASOTTI S., MANZINI G., SPAGNUOLO M.: Polyhedral mesh quality indicator for the virtual element method. *Computers & Mathematics with Applications* 114 (2022), 151–160. 11
- [SBMS22b] SORGENTE T., BIASOTTI S., MANZINI G., SPAGNUOLO M.: The role of mesh quality and mesh quality indicators in the virtual element method. *Advances in Computational Mathematics* 48, 1 (2022), 1–34. 11
- [SBMS22c] SORGENTE T., BIASOTTI S., MANZINI G., SPAGNUOLO M.: VEM 3D Dataset. <https://github.com/TommasoSorgente/vem-3D-quality-dataset>, 2022. 19
- [SDG*19a] SCHNEIDER T., DUMAS J., GAO X., BOTSCH M., PANOZZO D., ZORIN D.: Poly-spline finite-element method. *ACM Transactions on Graphics (TOG)* 38, 3 (2019), 1–16. 18
- [SDG*19b] SCHNEIDER T., DUMAS J., GAO X., ZORIN D., PANOZZO D.: PolyFEM. <https://polyfem.github.io/>, 2019. 18
- [SDVG10] SIROIS Y., DOMPIERRE J., VALLET M.-G., GUIBAULT F.: Hybrid mesh smoothing based on riemannian metric non-conformity minimization. *Finite elements in analysis and design* 46, 1-2 (2010), 47–60. 11
- [SEK*07] STIMPSON C., ERNST C., KNUPP P., PÉBAY P., THOMPSON D.: The verdict library reference manual. *Sandia National Laboratories Technical Report* 9, 6 (2007). 1, 8, 9, 10, 19
- [SH97] STANDER B. T., HART J. C.: Guaranteeing the topology of an implicit surface polygonization for interactive modeling. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (USA, 1997)*, SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co., p. 279–286. doi:[10.1145/258734.258868](https://doi.org/10.1145/258734.258868). 13
- [SHD*18] SCHNEIDER T., HU Y., DUMAS J., GAO X., PANOZZO D., ZORIN D.: Decoupling simulation accuracy from mesh quality. *ACM Transactions on Graphics* 37, 6 (10 2018). 18
- [She88] SHEPHARD M. S.: Approaches to the automatic generation and control of finite element meshes. *Applied Mechanics Reviews* 41, 4 (1988), 169–185. 1
- [She02] SHEWCHUK J.: What is a good linear finite element? interpolation, conditioning, anisotropy, and quality measures. *University of California at Berkeley* 2002 (2002). 5, 6, 7, 8
- [SHG*22] SCHNEIDER T., HU Y., GAO X., DUMAS J., ZORIN D., PANOZZO D.: A large-scale comparison of tetrahedral and hexahedral elements for solving elliptic pdes with the finite element method. *ACM Trans. Graph.* 41, 3 (mar 2022). 18
- [SJ08] SHEPHERD J. F., JOHNSON C. R.: Hexahedral mesh generation constraints. *Engineering with Computers* 24, 3 (2008), 195–213. 15
- [SOS*11] STATEN M. L., OWEN S. J., SHONTZ S. M., SALINGER A. G., COFFEY T. S.: A comparison of mesh morphing methods for 3d shape optimization. In *Proceedings of the 20th international meshing roundtable*. Springer, 2011, pp. 293–311. 16
- [SP92] SCARLATOS L. L., PAVLIDIS T.: Optimizing triangulations by curvature equalization. In *Proceedings of the 3rd conference on Visualization '92* (1992), pp. 333–339. 8
- [SPC*22] SORGENTE T., PRADA D., CABIDDU D., BIASOTTI S., PATANE G., PENNACCHIO M., BERTOLUZZA S., MANZINI G., SPAGNUOLO M.: *VEM and the Mesh*, vol. 31 of *SEMA SIMAI Springer series*. Springer International Publishing, 2022, ch. 1, pp. 1–57. 11
- [SS09] SASTRY S. P., SHONTZ S. M.: A comparison of gradient- and hessian-based optimization methods for tetrahedral mesh quality improvement. In *Proceedings of the 18th International Meshing Roundtable*. Springer, 2009, pp. 631–648. 16
- [ST04] SUKUMAR N., TABARRAEI A.: Conforming polygonal finite elements. *International Journal for Numerical Methods in Engineering* 61, 12 (2004), 2045–2066. 5
- [SVB*22] SORGENTE T., VICINI F., BERRONE S., BIASOTTI S., MANZINI G., SPAGNUOLO M.: Mesh quality agglomeration algorithm for the virtual element method applied to discrete fracture networks, 2022. 18
- [SZM12] SUN L., ZHAO G., MA X.: Quality improvement methods for hexahedral element meshes adaptively generated using grid-based algorithm. *International Journal for Numerical Methods in Engineering* 89, 6 (2012), 726–761. 17
- [Tha80] THACKER W. C.: A brief review of techniques for generating irregular computational grids. *International Journal for Numerical Methods in Engineering* 15, 9 (1980), 1335–1341. 1
- [TPPM12] TALISCHI C., PAULINO G. H., PEREIRA A., MENEZES I. F.: Polymesher: a general-purpose mesh generator for polygonal elements written in matlab. *Structural and Multidisciplinary Optimization* 45, 3 (2012), 309–328. 18
- [TV08] TANGELDER J. W., VELTKAMP R. C.: A survey of content based 3d shape retrieval methods. *Multimedia tools and applications* 39, 3 (2008), 441–471. 4
- [VAGW08] VARTZIOTIS D., ATHANASIADIS T., GOUDAS I., WIPPER J.: Mesh smoothing using the geometric element transformation method. *Computer Methods in Applied Mechanics and Engineering* 197, 45-48 (2008), 3760–3767. 16
- [VH14] VARTZIOTIS D., HIMPEL B.: Efficient mesh optimization using the gradient flow of the mean volume. *SIAM J. Numer. Anal.* 52, 2 (2014), 1050–1075. 17
- [VW10] VARTZIOTIS D., WIPPER J.: Characteristic parameter sets and limits of circulant hermitian polygon transformations. *Linear algebra and its applications* 433, 5 (2010), 945–955. 16
- [VW12] VARTZIOTIS D., WIPPER J.: Fast smoothing of mixed volume meshes based on the effective geometric element transformation method. *Computer methods in applied mechanics and engineering* 201 (2012), 65–81. 11, 16, 17
- [Wan17] WANG L.: *Algorithms and Criteria for Volumetric Centroidal Voronoi Tessellations*. PhD thesis, Université Grenoble Alpes, 2017. 11
- [WBM21] WHALEN E., BEYENE A., MUELLER C.: Simjeb: Simulated jet engine bracket dataset. In *Computer Graphics Forum* (2021), vol. 40-5, Wiley Online Library, pp. 9–17. 19
- [WHDS04] WOOD Z., HOPPE H., DESBRUN M., SCHRÖDER P.: Removing excess topology from isosurfaces. *ACM Transactions on Graphics (TOG)* 23, 2 (2004), 190–208. 13
- [XN06] XU H., NEWMAN T. S.: An angle-based optimization approach for 2d finite element mesh smoothing. *Finite Elements in Analysis and Design* 42, 13 (2006), 1150–1164. 16
- [ZJ16] ZHOU Q., JACOBSON A.: Thingi10k: A dataset of 10,000 3D-printing models, 2016. [arXiv:1605.04797](https://arxiv.org/abs/1605.04797). 15, 19
- [Zlá68] ZLÁMAL M.: On the finite element method. *Numerische Mathematik* 12, 5 (1968), 394–409. 1, 6
- [ZOG22] ZANDSALIMY M., OLLIVIER-GOOCH C.: A novel approach to mesh optimization to stabilize unstructured finite volume simulations. *Journal of Computational Physics* 453 (2022), 110959. 14, 17
- [ZR04] ZUNIC J., ROSIN P. L.: A new convexity measure for polygons. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 7 (2004), 923–934. 11